



プロクスモックス ONTAP SAN Host Utilities

NetApp
February 24, 2026

目次

プロクスモックス	1
ProxmoxホストのONTAPサポートと機能について学ぶ	1
次の手順	1
ONTAPストレージでNVMe-oF用にProxmox VE 9.xを構成する	1
ステップ1: Proxmox VEとNVMeソフトウェアをインストールし、構成を確認する	2
ステップ2: NVMe/FCとNVMe/TCPを構成する	3
ステップ3: オプションでNVMe/FCの1MB I/Oを有効にする	9
ステップ4: NVMeブートサービスを確認する	10
ステップ5: マルチパス構成を確認する	12
手順6: 既知の問題を確認する	16
ONTAPストレージでNVMe-oF用にProxmox VE 8.xを構成する	17
ステップ1: Proxmox VEとNVMeソフトウェアをインストールし、構成を確認する	17
ステップ2: NVMe/FCとNVMe/TCPを構成する	18
ステップ3: オプションでNVMe/FCの1MB I/Oを有効にする	27
ステップ4: NVMeブートサービスを確認する	28
ステップ5: マルチパス構成を確認する	29
手順6: 既知の問題を確認する	33

プロクスマックス

ProxmoxホストのONTAPサポートと機能について学ぶ

NVMe over Fabrics (NVMe-oF) を使用したホスト構成でサポートされる機能は、ONTAPと Proxmox のバージョンによって異なります。

ONTAP機能	Proxmoxホストバージョン	ONTAPのバージョン
NVMe/TCPは完全にサポートされているエンタープライズ機能です	9.0	9.10.1以降
ネイティブのNVMe/FCとNVMe/TCPネームスペースの両方のONTAP詳細を提供します。 `nvme-cli`パッケージ。	8.0	9.10.1以降
NVMe と SCSI トラフィックは、NVMe-oF 名前空間の場合は NVMe マルチパス、SCSI LUN の場合は dm-multipath を使用して、同じホスト上でサポートされます。	8.0	9.4以降

ONTAP は、システム セットアップで実行されているONTAP のバージョンに関係なく、次の SAN ホスト機能をサポートします。

特徴	Proxmoxホストバージョン
その `nvme-cli` パッケージには自動接続スクリプトが含まれており、サードパーティのスクリプトは不要になります。	9.0
`nvme-cli` パッケージ内のネイティブudevルールは、NVMeマルチパスのラウンドロビン ロード バランシングを提供します	9.0
ネイティブNVMeマルチパスはデフォルトで有効になっています	8.0



サポートされている構成の詳細については、"[Interoperability Matrix Tool](#)"。

次の手順

Proxmox VE のバージョンが..	について学ぶ..
9シリーズ	"Proxmox VE 9.x 用の NVMe の設定"
8シリーズ	"Proxmox VE 8.x 用の NVMe の設定"

関連情報

["NVMeプロトコルの管理について学ぶ"](#)

ONTAPストレージでNVMe-oF用にProxmox VE 9.xを構成する

Proxmox VE 9.x ホストは、非対称名前空間アクセス (ANA) を備えた NVMe over Fibre

Channel (NVMe/FC) および NVMe over TCP (NVMe/TCP) プロトコルをサポートします。ANA は、iSCSI および FCP 環境における非対称論理ユニット アクセス (ALUA) と同等のマルチパス機能を提供します。

Proxmox VE 9.x 用の NVMe over Fabrics (NVMe-oF) ホストを構成する方法を学びます。詳細なサポートと機能情報については、"[ONTAPのサポートと機能](#)"。

Proxmox VE 9.x の NVMe-oF には、次の既知の制限があります。

- NVMe-FC の SAN ブート構成はサポートされていません。

ステップ1: Proxmox VEとNVMeソフトウェアをインストールし、構成を確認する

NVMe-oF 用にホストを構成するには、ホストおよび NVMe ソフトウェア パッケージをインストールし、マルチパスを有効にして、ホストの NQN 構成を確認する必要があります。

手順

1. サーバーに Proxmox VE 9.x をインストールします。インストールが完了したら、必要な Proxmox VE 9.x カーネルが実行されていることを確認します。

```
uname -r
```

Proxmox VE 9.x カーネルバージョンの例:

```
6.14.8-2-pve
```

2. 「nvme-cli」パッケージをインストールします。

```
apt list|grep nvme-cli
```

次の例は、`nvme-cli`パッケージバージョン:

```
nvme-cli/stable,now 2.13-2 amd64
```

3. をインストールします libnvme パッケージ:

```
apt list|grep libnvme
```

次の例は、`libnvme`パッケージバージョン:

```
libnvme-dev/stable 1.13-2 amd64
```

4. ホスト上で、hostnqn文字列を確認します。 /etc/nvme/hostnqn :

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

次の例は、`hostnqn`値 :

```
nqn.2014-08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a524c
```

5. ONTAPシステムで、`hostnqn`文字列が一致する `hostnqn`ONTAPアレイ上の対応するサブシステムの文字列:

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

例を示します

```
Vserver Subsystem Priority Host NQN
-----
vs_proxmox_FC_NVMeFC
    sub_176
                regular nqn.2014-
08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a4834
                regular nqn.2014-
08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a524c
2 entries were displayed
```



もし `hostnqn`文字列が一致しない場合は、`vserver modify`更新するコマンド `hostnqn`対応するONTAPストレージシステムサブシステムの文字列を `hostnqn`文字列から `/etc/nvme/hostnqn`ホスト上。

ステップ2: NVMe/FCとNVMe/TCPを構成する

Broadcom/Emulex または Marvell/QLLogic アダプタを使用して NVMe/FC を構成するか、手動の検出および接続操作を使用して NVMe/TCP を構成します。

NVMe/FC - ブロードコム/エミュレックス

Broadcom/Emulexアダプタ用にNVMe/FCを設定します。

手順

1. サポートされているアダプタモデルを使用していることを確認します。

a. モデル名を表示します。

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

次の出力が表示されます。

```
SN1700E2P  
SN1700E2P
```

b. モデルの説明を表示します:

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

次の例のような出力が表示されます。

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

2. 推奨されるBroadcomを使用していることを確認します lpfc ファームウェアおよび受信トレイドライバ:

a. ファームウェアのバージョンを表示します。

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

このコマンドはファームウェアのバージョンを返します。

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

b. 受信トレイのドライバーのバージョンを表示します。

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

次の例は、ドライバーのバージョンを示しています。

```
0:14.4.0.7
```

サポートされているアダプタドライバおよびファームウェアバージョンの最新リストについては、を参照してください"[Interoperability Matrix Tool](#)".

3. 確認します lpfc_enable_fc4_type がに設定されます 3 :

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

4. イニシエータポートを表示できることを確認します。

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

次のような出力が表示されます：

```
0x10005ced8c531948  
0x10005ced8c531949
```

5. イニシエータポートがオンラインであることを確認します。

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

次の出力が表示されます。

```
Online  
Online
```

6. NVMe/FCイニシエータポートが有効になっており、ターゲットポートが認識されることを確認します。

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

例を示します

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10005ced8c531948 WWNN x20005ced8c531948
DID x082400 ONLINE
NVME RPORT          WWPN x200ed039eac79573 WWNN x200dd039eac79573
DID x060902 TARGET DISCSRVC ONLINE
NVME RPORT          WWPN x2001d039eac79573 WWNN x2000d039eac79573
DID x060904 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000000034 Cmpl 0000000034 Abort 00000000
LS XMIT: Err 00000000  CMLP: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000142cfb Issue 0000000000142cfc OutIO
00000000000000001
      abort 00000005 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMLP: xb 00000005 Err 00000005
```

```
NVME Initiator Enabled
XRI Dist lpfc1 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc1 WWPN x10005ced8c531949 WWNN x20005ced8c531949
DID x082500 ONLINE
NVME RPORT          WWPN x2010d039eac79573 WWNN x200dd039eac79573
DID x062902 TARGET DISCSRVC ONLINE
NVME RPORT          WWPN x2007d039eac79573 WWNN x2000d039eac79573
DID x062904 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000000034 Cmpl 0000000034 Abort 00000000
LS XMIT: Err 00000000  CMLP: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000d39f1 Issue 00000000000d39f2 OutIO
00000000000000001
      abort 00000005 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMLP: xb 00000005 Err 00000005
```

NVMe/FC - マーベル/QLogic

Marvell/QLogicアダプタ用にNVMe/FCを設定します。

手順

1. サポートされているアダプタ ドライバーとファームウェア バージョンを使用していることを確認します。

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

次の例は、ドライバーとファームウェアのバージョンを示しています。

```
SN1700Q FW:v9.15.05 DVR:v10.02.09.400-k  
SN1700Q FW:v9.15.05 DVR:v10.02.09.400-k
```

2. 確認します `ql2xnvmeenable` が設定されます。これにより、MarvellアダプタをNVMe/FCイニシエーターとして機能させることができます。

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

想定される出力は1です。

NVMe/FC

NVMe/TCP プロトコルは自動接続操作をサポートしていません。代わりに、NVMe/TCPサブシステムと名前空間をNVMe/TCPコマンドで検出することができます。`connect`または`connect-all`手動で操作します。

手順

1. イニシエーター ポートが、サポートされている NVMe/TCP LIF 全体で検出ログ ページ データを取得できることを確認します。

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

例を示します

```
nvme discover -t tcp -w 192.168.165.72 -a 192.168.165.51
Discovery Log Number of Records 4, Generation counter 47
====Discovery Log Entry 0====
trtype: tcp
adrfam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 3
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:discovery
traddr: 192.168.165.51
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: 1
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:discovery
traddr: 192.168.166.50
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
====Discovery Log Entry 2====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 3
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:subsystem.sub_
176
traddr: 192.168.165.51
eflags: none
sectype: none
====Discovery Log Entry 3====
trtype: tcp
```

```
adrfam:  ipv4
subtype:  nvme subsystem
treq:    not specified
portid:  1
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:subsystem.sub_
176
traddr:  192.168.166.50
eflags:  none
sectype: none
```

2. を実行します `nvme connect-all` ノード全体でサポートされているすべてのNVMe/TCPイニシエータ/ターゲットLIFを対象としたコマンド：

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

例を示します

```
nvme connect-all -t tcp -w 192.168.166.73 -a 192.168.166.50
nvme connect-all -t tcp -w 192.168.166.73 -a 192.168.166.51
nvme connect-all -t tcp -w 192.168.165.73 -a 192.168.165.50
nvme connect-all -t tcp -w 192.168.165.73 -a 192.168.165.51
```

NVMe/TCPの設定 `ctrl_loss_tmo timeout` 自動的に「オフ」に設定されます。結果として：

- 再試行回数に制限はありません（無期限再試行）。
- 特定の設定を手動で行う必要はありません `ctrl_loss_tmo timeout` 使用時の持続時間 `nvme connect` または `nvme connect-all` コマンド（オプション-I）。
- NVMe/TCP コントローラーは、パス障害が発生した場合でもタイムアウトが発生せず、無期限に接続されたままになります。

ステップ3: オプションでNVMe/FCの1MB I/Oを有効にする

ONTAP は、識別コントローラ データで最大データ転送サイズ (MDTS) が 8 であると報告します。つまり、最大 I/O 要求サイズは 1 MB までになります。Broadcom NVMe/FCホストに1MBのI/Oリクエストを発行するには、`lpfc`の値 `lpfc_sg_seg_cnt` パラメータをデフォルト値の 64 から 256 に変更します。



この手順は、Qlogic NVMe/FCホストには適用されません。

手順

1. `lpfc_sg_seg_cnt`パラメータを256に設定します。

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

次の例のような出力が表示されます。

```
options lpfc lpfc_sg_seg_cnt=256
```

2. 実行 `update-initramfs` コマンドを実行し、ホストを再起動します。
3. の値が256であることを確認し `lpfc_sg_seg_cnt` ます。

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

ステップ4: NVMeブートサービスを確認する

その `nvmecli` パッケージはシステムの起動時に自動的に有効になります。

起動が完了したら、`nvmecli` パッケージはシステムの起動時に自動的に有効になります。

手順

1. が有効であることを確認し `nvmecli` ます。

```
systemctl status nvmecli.service
```

出力例を表示します。

```
○ nvme-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot
   Loaded: loaded (/usr/lib/systemd/system/nvme-
autoconnect.service; enabled; preset: enabled)
   Active: inactive (dead) since Thu 2025-10-16 18:00:39 IST; 6
days ago
  Invocation: e146e0b2c339432aad6e0555a528872c
    Process: 1787 ExecStart=/usr/sbin/nvme connect-all
--context=autoconnect (code=exited, status=0/SUCCESS)
   Main PID: 1787 (code=exited, status=0/SUCCESS)
  Mem peak: 2.4M
     CPU: 12ms
Oct 16 18:00:39 HPE-DL365-14-176 systemd[1]: Starting nvme-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot...
Oct 16 18:00:39 HPE-DL365-14-176 systemd[1]: nvme-
autoconnect.service: Deactivated successfully.
Oct 16 18:00:39 HPE-DL365-14-176 systemd[1]: Finished nvme-
autoconnect.service - Connect NVMe-oF subsystems automatically
during boot.
```

2. が有効であることを確認し `nvme-fc-boot-connections.service` ます。

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

出力例を表示します。

```
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 Thu 2025-10-16 18:00:35 IST; 6
days ago
  Invocation: acf73ac1ef7a402198d6ecc4d075fab0
    Process: 1173 ExecStart=/bin/sh -c echo add >
/sys/class/fc/fc_udev_device/nvme_discovery (code=exited,
status=0/SUCCESS)
   Main PID: 1173 (code=exited, status=0/SUCCESS)
   Mem peak: 2.1M
      CPU: 11ms

Oct 16 18:00:35 HPE-DL365-14-176 systemd[1]: nvme-fc-boot-
connections.service: Deactivated successfully.
Oct 16 18:00:35 HPE-DL365-14-176 systemd[1]: Finished nvme-fc-boot-
connections.service - Auto-connect to subsystems on FC-NVME devices
found during boot.
```

ステップ5: マルチパス構成を確認する

カーネル内のNVMeマルチパスステータス、ANAステータス、およびONTAP名前空間がNVMe-oF構成に対して正しいことを確認します。

手順

1. カーネル内NVMeマルチパスが有効になっていることを確認します。

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

次の出力が表示されます。

```
Y
```

2. ONTAP名前空間の適切な NVMe-oF 設定 (モデルがNetApp ONTAPコントローラに設定され、負荷分散 iopolicy がラウンドロビンに設定されているなど) がホストに正しく表示されていることを確認します。
 - a. サブシステムを表示します。

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

次の出力が表示されます。

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

b. ポリシーを表示します。

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

次の出力が表示されます。

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

3. ネームスペースが作成され、ホストで正しく検出されたことを確認します。

```
nvme list
```

例を示します

```
Node           Generic      SN           Model  
Namespace Usage           Format           FW Rev  
-----  
-----  
-----  
/dev/nvme2n1  /dev/ng2n1  81PqYFYq2aVAAAAAAAAAAB NetApp ONTAP  
Controller  0x1         17.88 GB / 171.80 GB      4 KiB + 0 B  
9.17.1
```

4. 各パスのコントローラの状態がliveであり、正しいANAステータスが設定されていることを確認します。

NVMe/FC

```
nvme list-subsys /dev/nvme3n9
```

例を示します

```
nvme-subsys3 - NQN=nqn.1992-08.com.netapp:sn.94929fdb84eb11f0b624d039eac809ba:subsystem.sub_176
                hostnqn=nqn.2014-08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a524c
\
+- nvme1 fc traddr=nn-0x200dd039eac79573:pn-0x2010d039eac79573,host_traddr=nn-0x20005ced8c531949:pn-0x10005ced8c531949 live optimized
+- nvme3 fc traddr=nn-0x200dd039eac79573:pn-0x200ed039eac79573,host_traddr=nn-0x20005ced8c531948:pn-0x10005ced8c531948 live optimized
+- nvme5 fc traddr=nn-0x200dd039eac79573:pn-0x200fd039eac79573,host_traddr=nn-0x20005ced8c531949:pn-0x10005ced8c531949 live non-optimized
+- nvme7 fc traddr=nn-0x200dd039eac79573:pn-0x2011d039eac79573,host_traddr=nn-0x20005ced8c531948:pn-0x10005ced8c531948 live non-optimized
```

NVMe/FC

```
nvme list-subsys /dev/nvme2n3
```

例を示します

```
nvme-subsys2 - NQN=nqn.1992-08.com.netapp:sn.c770be5d934811f0b624d039eac809ba:subsystem.sub_176
                hostnqn=nqn.2014-08.org.nvmexpress:uuid:39333550-3333-4753-4844-32594d4a524c
\
+- nvme2 tcp
traddr=192.168.166.50,trsvcid=4420,host_traddr=192.168.166.73,src_addr=192.168.166.73 live optimized
+- nvme4 tcp
traddr=192.168.165.51,trsvcid=4420,host_traddr=192.168.165.73,src_addr=192.168.165.73 live optimized
+- nvme6 tcp
traddr=192.168.166.51,trsvcid=4420,host_traddr=192.168.166.73,src_addr=192.168.166.73 live non-optimized
+- nvme8 tcp
traddr=192.168.165.50,trsvcid=4420,host_traddr=192.168.165.73,src_addr=192.168.165.73 live non-optimized
```

5. ネットアッププラグインで、ONTAP ネームスペースデバイスごとに正しい値が表示されていることを確認します。

列 (Column)

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

例を示します

```
Device          Vserver          Namespace Path
-----
/dev/nvme2n9    vs_proxmox_FC_NVMeFC  /vol/vol_180_data_nvme4/ns

NSID            UUID              Size
-----
1               e3d3d544-de8b-4787-93af-bfec7769e909  32.21GB
```

JSON

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

例を示します

```
{
  "Device": "/dev/nvme2n9",
  "Vserver": "vs_proxmox_FC_NVMeFC",
  "Subsystem": "sub_176",
  "Namespace_Path": "/vol/vol_180_data_nvme4/ns",
  "NSID": 9,
  "UUID": "e3d3d544-de8b-4787-93af-bfec7769e909",
  "LBA_Size": 4096,
  "Namespace_Size": 32212254720,
  "UsedBytes": 67899392,
  "Version": "9.17.1"
}
```

手順6：既知の問題を確認する

既知の問題はありません。

ONTAPストレージでNVMe-oF用にProxmox VE 8.xを構成する

Proxmox VE 8.x ホストは、非対称名前空間アクセス (ANA) を備えた NVMe over Fibre Channel (NVMe/FC) および NVMe over TCP (NVMe/TCP) プロトコルをサポートします。ANA は、iSCSI および FCP 環境における非対称論理ユニット アクセス (ALUA) と同等のマルチパス機能を提供します。

Proxmox VE 8.x 用に NVMe over Fabrics (NVMe-oF) ホストを構成する方法を学びます。詳細なサポートと機能情報については、"[ONTAPのサポートと機能](#)"。

Proxmox VE 8.x の NVMe-oF には、次の既知の制限があります。

- NVMe-FC の SAN ブート構成はサポートされていません。

ステップ1: Proxmox VEとNVMeソフトウェアをインストールし、構成を確認する

NVMe-oF 用にホストを構成するには、ホストおよび NVMe ソフトウェア パッケージをインストールし、マルチパスを有効にして、ホストの NQN 構成を確認する必要があります。

手順

1. サーバーに Proxmox 8.x をインストールします。インストールが完了したら、指定された Proxmox 8.x カーネルが実行されていることを確認します。

```
uname -r
```

次の例は、Proxmox カーネルのバージョンを示しています。

```
6.8.12-10-pve
```

2. 「nvme-cli」パッケージをインストールします。

```
apt list|grep nvme-cli
```

次の例は、`nvme-cli`パッケージバージョン:

```
nvme-cli/oldstable,now 2.4+really2.3-3 amd64
```

3. をインストールします libnvme パッケージ:

```
apt list|grep libnvme
```

次の例は、`libnvme`パッケージバージョン:

```
libnvme1/oldstable,now 1.3-1+deb12u1 amd64
```

4. ホスト上で、`hostnqn`文字列を確認します。 `/etc/nvme/hostnqn` :

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

次の例は、`hostnqn`値 :

```
nqn.2014-08.org.nvmeexpress:uuid:1536c9a6-f954-11ea-b24d-0a94efb46eaf
```

5. ONTAPシステムで、`hostnqn`文字列が一致する `hostnqn`ONTAPアレイ上の対応するサブシステムの文字列:

```
::> vserver nvme subsystem host show -vserver proxmox_120_122
```

例を示します

```
Vserver Subsystem Priority Host NQN
-----
proxmox_120_122
proxmox_120_122
                regular nqn.2014-
08.org.nvmeexpress:uuid:1536c9a6-f954-11ea-b24d-0a94efb46eaf
                regular nqn.2014-
08.org.nvmeexpress:uuid:991a7476-f9bf-11ea-8b73-0a94efb46c3b
proxmox_120_122_tcp
                regular nqn.2014-
08.org.nvmeexpress:uuid:1536c9a6-f954-11ea-b24d-0a94efb46eaf
                regular nqn.2014-08.org.nvmeexpress:uuid:991a7476-
f9bf-11ea-8b73-0a94efb46c3b
2 entries were displayed.
```



もし `hostnqn`文字列が一致しない場合は、`vserver modify`更新するコマンド `hostnqn`対応するONTAPストレージシステムサブシステムの文字列を `hostnqn`文字列から`/etc/nvme/hostnqn`ホスト上。

ステップ2: NVMe/FCとNVMe/TCPを構成する

Broadcom/Emulex または Marvell/QLogic アダプタを使用して NVMe/FC を構成するか、手動の検出および接続操作を使用して NVMe/TCP を構成します。

NVMe/FC - ブロードコム/エミュレックス

Broadcom/Emulexアダプタ用にNVMe/FCを設定します。

手順

1. サポートされているアダプタモデルを使用していることを確認します。

- a. モデル名を表示します。

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

次の出力が表示されます。

```
LPe35002-M2  
LPe35002-M2
```

- b. モデルの説明を表示します。

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

次の例のような出力が表示されます。

```
Emulex LPe35002-M2 2-Port 32Gb Fibre Channel Adapter  
Emulex LPe35002-M2 2-Port 32Gb Fibre Channel Adapter
```

2. 推奨されるBroadcomを使用していることを確認します lpfc ファームウェアおよび受信トレイドライバ:

- a. ファームウェアのバージョンを表示します。

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

このコマンドはファームウェアのバージョンを返します。

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

- b. 受信トレイのドライバーのバージョンを表示します。

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

次の例は、ドライバーのバージョンを示しています。

```
0:14.2.0.17
```

サポートされているアダプタドライバおよびファームウェアバージョンの最新リストについては、を参照してください"[Interoperability Matrix Tool](#)".

3. 確認します lpfc_enable_fc4_type がに設定されます 3 :

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

4. イニシエータポートを表示できることを確認します。

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

次のような出力が表示されます：

```
0x100000109b95467e  
0x100000109b95467f
```

5. イニシエータポートがオンラインであることを確認します。

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

次の出力が表示されます。

```
Online  
Online
```

6. NVMe/FCイニシエータポートが有効になっており、ターゲットポートが認識されることを確認します。

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

例を示します

```
NVME Initiator Enabled
XRI Dist lpfc0 Total 6144 IO 5894 ELS 250
NVME LPORT lpfc0 WWPN x10005ced8c531948 WWNN x20005ced8c531948
DID x082400
ONLINE
NVME RPORT WWPN x200ed039eac79573 WWNN x200dd039eac79573 DID
x060902
TARGET DISCSRV ONLINE
NVME RPORT WWPN x2001d039eac79573 WWNN x2000d039eac79573 DID
x060904
TARGET DISCSRV ONLINE
NVME Statistics LS: Xmt 0000000034 Cmpl 0000000034 Abort
00000000 LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 0000000000142cfb Issue 0000000000142cfc OutIO
0000000000000001 abort 00000005 noxri 00000000 nondlp 00000000
qdepth 00000000 wqerr 00000000 err 00000000 FCP CMPL: xb
00000005 Err 00000005 NVME Initiator Enabled XRI Dist lpfc1
Total 6144 IO 5894 ELS 250 NVME LPORT lpfc1 WWPN
x10005ced8c531949 WWNN x20005ced8c531949 DID x082500
ONLINE
NVME RPORT WWPN x2010d039eac79573 WWNN x200dd039eac79573 DID
x062902
TARGET DISCSRV ONLINE
NVME RPORT WWPN x2007d039eac79573 WWNN x2000d039eac79573 DID
x062904
TARGET DISCSRV ONLINE
NVME Statistics LS: Xmt 0000000034 Cmpl 0000000034 Abort
00000000 LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000d39f1 Issue 00000000000d39f2 OutIO
0000000000000001 abort 00000005 noxri 00000000 nondlp 00000000
qdepth 00000000 wqerr 00000000 err 00000000 FCP CMPL: xb
00000005 Err 00000005
```

NVMe/FC - マーベル/QLogic

Marvell/QLogicアダプタ用にNVMe/FCを設定します。

1. サポートされているアダプタ ドライバーとファームウェア バージョンを使用していることを確認します。

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

次の例は、ドライバーとファームウェアのバージョンを示しています。

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

2. 確認します `ql2xnvmeenable` が設定されます。これにより、MarvellアダプタをNVMe/FCイニシエーターとして機能させることができます。

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

想定される出力は1です。

NVMe/FC

NVMe/TCP プロトコルは自動接続操作をサポートしていません。代わりに、NVMe/TCPサブシステムと名前空間をNVMe/TCPコマンドで検出することができます。`connect`または`connect-all`手動で操作します。

手順

1. イニシエーターポートが、サポートされているNVMe/TCP LIF全体で検出ログページデータを取得できることを確認します。

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

例を示します

```
nvme discover -t tcp -w 192.168.2.22 -a 192.168.2.30

Discovery Log Number of Records 12, Generation counter 13
====Discovery Log Entry 0=====
trtype: tcp
adrfam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 10
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr: 192.168.2.30
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: 9
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr: 192.168.1.30
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: 12
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr: 192.168.2.25
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: 11
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:discovery
traddr: 192.168.1.25
eflags: explicit discovery connections, duplicate discovery
information
sectype: none
====Discovery Log Entry 4====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 10
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr: 192.168.2.30
eflags: none
sectype: none
====Discovery Log Entry 5====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 9
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr: 192.168.1.30
eflags: none
sectype: none
====Discovery Log Entry 6====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 12
trsvcid: 4420
subnqn: nqn.1992-
```

```
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr: 192.168.2.25
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 11
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122
traddr: 192.168.1.25
eflags: none
sectype: none
=====Discovery Log Entry 8=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 10
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr: 192.168.2.30
eflags: none
sectype: none
=====Discovery Log Entry 9=====
trtype: tcp
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 9
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr: 192.168.1.30
eflags: none
sectype: none
=====Discovery Log Entry 10=====
trtype: tcp
```

```
adrfam:  ipv4
subtype:  nvme subsystem
treq:    not specified
portid:  12
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr:  192.168.2.25
eflags:  none
sectype: none
=====Discovery Log Entry 11=====
trtype:  tcp
adrfam:  ipv4
subtype:  nvme subsystem
treq:    not specified
portid:  11
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox
mox_120_122_tcp
traddr:  192.168.1.25
eflags:  none
sectype: none
```

2. 他の NVMe/TCP イニシエーターとターゲット LIF の組み合わせで検出ログ ページ データを正常に取得できることを確認します。

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

例を示します

```
nvme discover -t tcp -w 192.168.1.22 -a 192.168.1.30
nvme discover -t tcp -w 192.168.2.22 -a 192.168.2.30
nvme discover -t tcp -w 192.168.1.22 -a 192.168.1.25
nvme discover -t tcp -w 192.168.2.22 -a 192.168.2.25
```

3. を実行します `nvme connect-all` ノード全体でサポートされているすべての NVMe/TCP イニシエーター/ターゲット LIF を対象としたコマンド：

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

例を示します

```
nvme connect-all -t tcp -w 192.168.1.22 -a 192.168.1.30
nvme connect-all -t tcp -w 192.168.2.22 -a 192.168.2.30
nvme connect-all -t tcp -w 192.168.1.22 -a 192.168.1.25
nvme connect-all -t tcp -w 192.168.2.22 -a 192.168.2.25
```

NVMe/TCPの設定 `ctrl_loss_tmo timeout` 自動的に「オフ」に設定されます。結果として：

- 再試行回数に制限はありません（無期限再試行）。
- 特定の設定を手動で行う必要はありません `ctrl_loss_tmo timeout` 使用時の持続時間 `nvme connect` または `nvme connect-all` コマンド（オプション -l）。
- NVMe/TCP コントローラーは、パス障害が発生した場合でもタイムアウトが発生せず、無期限に接続されたままになります。

ステップ3: オプションでNVMe/FCの1MB I/Oを有効にする

ONTAP は、識別コントローラ データで最大データ転送サイズ (MDTS) が 8 であると報告します。つまり、最大 I/O 要求サイズは 1 MB までになります。Broadcom NVMe/FCホストに1MBのI/Oリクエストを発行するには、`lpfc`の値 `lpfc_sg_seg_cnt` パラメータをデフォルト値の 64 から 256 に変更します。



この手順は、Qlogic NVMe/FCホストには適用されません。

手順

1. `lpfc_sg_seg_cnt` パラメータを256に設定します。

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

次の例のような出力が表示されます。

```
options lpfc lpfc_sg_seg_cnt=256
```

2. 実行 `update-initramfs` コマンドを実行し、ホストを再起動します。
3. の値が256であることを確認し `lpfc_sg_seg_cnt` ます。

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

ステップ4: NVMeブートサービスを確認する

Proxmox 8.xでは、`nvme-fc-boot-connections.service`そして`nvme-fc-autoconnect.service` NVMe/FCに含まれるブートサービス`nvme-cli`パッケージはシステムの起動時に自動的に有効になります。

起動が完了したら、`nvme-fc-boot-connections.service`そして`nvme-fc-autoconnect.service`ブートサービスが有効になっています。

手順

1. が有効であることを確認し`nvme-fc-autoconnect.service`ます。

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

出力例を表示します。

```
○ nvme-fc-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot
   Loaded: loaded (/lib/systemd/system/nvme-fc-autoconnect.service;
enabled; preset: enabled)
   Active: inactive (dead) since Fri 2025-11-21 19:59:10 IST; 8s
ago
   Process: 256613 ExecStartPre=/sbin/modprobe nvme-fabrics
(code=exited, status=0/SUCCESS)
   Process: 256614 ExecStart=/usr/sbin/nvme connect-all
(code=exited, status=0/SUCCESS)
   Main PID: 256614 (code=exited, status=0/SUCCESS)
   CPU: 18ms
Nov 21 19:59:07 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Starting nvme-fc-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot...
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in nvme[256614]:
Failed to write to /dev/nvme-fabrics: Invalid argument
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in nvme[256614]:
Failed to write to /dev/nvme-fabrics: Invalid argument
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in systemd[1]: nvme-fc-
autoconnect.service: Deactivated successfully.
Nov 21 19:59:10 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Finished nvme-fc-autoconnect.service - Connect NVMe-oF subsystems
automatically during boot.
```

2. が有効であることを確認し`nvme-fc-boot-connections.service`ます。

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

出力例を表示します。

```
o nvme-fc-boot-connections.service - Auto-connect to subsystems on
FC-NVME devices found during boot
   Loaded: loaded (/lib/systemd/system/nvme-fc-boot-
connections.service; enabled; preset: enabled)
   Active: inactive (dead) since Thu 2025-11-20 17:48:29 IST; 1
day 2h ago
   Process: 1381 ExecStart=/bin/sh -c echo add >
/sys/class/fc/fc_udev_device/nvme_discovery (code=exited,
status=0/SUCCESS)
   Main PID: 1381 (code=exited, status=0/SUCCESS)
   CPU: 3ms

Nov 20 17:48:29 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Starting nvme-fc-boot-connections.service - Auto-connect to
subsystems on FC-NVME devices found during boot..
Nov 20 17:48:29 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
nvme-fc-boot-connections.service: Deactivated successfully.
Nov 20 17:48:29 SR665-14-122.lab.eng.btc.netapp.in systemd[1]:
Finished nvme-fc-boot-connections.service - Auto-connect to
subsystems on FC-NVME devices found during boot...
```

ステップ5: マルチパス構成を確認する

カーネル内のNVMeマルチパスステータス、ANAステータス、およびONTAP名前空間がNVMe-oF構成に対して正しいことを確認します。

手順

1. カーネル内NVMeマルチパスが有効になっていることを確認します。

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

次の出力が表示されます。

```
Y
```

2. ONTAP名前空間の適切な NVMe-oF 設定 (モデルがNetApp ONTAPコントローラに設定され、負荷分散 iopolicy がラウンドロビンに設定されているなど) がホストに正しく表示されていることを確認します。

- a. サブシステムを表示します。

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

次の出力が表示されます。

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

b. ポリシーを表示します。

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

次の出力が表示されます。

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

3. ネームスペースが作成され、ホストで正しく検出されたことを確認します。

```
nvme list
```

例を示します

```
Node           Generic           SN  
Model          Namespace  Usage  
Format        FW Rev  
-----  
-----  
-----  
/dev/nvme2n20  /dev/ng2n20      81K13BUDdygsAAAAAAG  
NetApp ONTAP Controller 10      5.56 GB /  
91.27 GB      4 KiB + 0 B    9.18.1
```

4. 各パスのコントローラの状態がliveであり、正しいANAステータスが設定されていることを確認します。

NVMe/FC

```
nvme list-subsys /dev/nvme2n20
```

例を示します

```
nvme-subsys2 - NQN= nqn.1992-  
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox  
mox_120_122_tcp  
\  
+- nvme1 fc traddr=nn-0x200dd039eac79573:pn-  
0x2010d039eac79573,host_traddr=nn-0x20005ced8c531949:pn-  
0x10005ced8c531949 live optimized  
+- nvme3 fc traddr=nn-0x200dd039eac79573:pn-  
0x200ed039eac79573,host_traddr=nn-0x20005ced8c531948:pn-  
0x10005ced8c531948 live optimized  
+- nvme5 fc traddr=nn-0x200dd039eac79573:pn-  
0x200fd039eac79573,host_traddr=nn-0x20005ced8c531949:pn-  
0x10005ced8c531949 live non-optimized  
+- nvme7 fc traddr=nn-0x200dd039eac79573:pn-  
0x2011d039eac79573,host_traddr=nn-0x20005ced8c531948:pn-  
0x10005ced8c531948 live non-optimized
```

NVMe/FC

```
nvme list-subsys /dev/nvme2n3
```

例を示します

```
nvme-subsys2 - NQN= qn.1992-  
08.com.netapp:sn.ae9f2d55a7ec11ef8751d039ea9e891c:subsystem.prox  
mox_120_122_tcp  
\  
+- nvme2 tcp  
traddr=192.168.1.30,trsvcid=4420,host_traddr=192.168.1.22,src_ad  
dr=192.168.1.22 live optimized  
+- nvme4 tcp  
traddr=192.168.2.30,trsvcid=4420,host_traddr=192.168.2.22,src_ad  
dr=192.168.2.22 live optimized  
+- nvme6 tcp  
traddr=192.168.1.25,trsvcid=4420,host_traddr=192.168.1.22,src_ad  
dr=192.168.1.22 live non-optimized  
+- nvme8 tcp  
traddr=192.168.2.25,trsvcid=4420,host_traddr=192.168.2.22,src_ad  
dr=192.168.2.22 live non-optimized
```

5. ネットアッププラグインで、ONTAP ネームスペースデバイスごとに正しい値が表示されていることを確認します。

列 (Column)

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

例を示します

```
Device          Vserver          Namespace Path
-----
/dev/nvme2n11   proxmox_120_122 / /vol/vm120_tcp1/ns

NSID            UUID              Size
-----
1               5aefea74-f0cf-4794-a7e9-e113c4659aca  37.58GB
```

JSON

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

例を示します

```
{
  "Device": "/dev/nvme2n11",
  "Vserver": "proxmox_120_122",
  "Namespace_Path": "/vol/vm120_tcp1/ns",
  "NSID": 1,
  "UUID": "5aefea74-f0cf-4794-a7e9-e113c4659aca",
  "Size": "37.58GB",
  "LBA_Data_Size": 4096,
  "Namespace_Size": 32212254720
}
```

手順6：既知の問題を確認する

既知の問題はありません。

著作権に関する情報

Copyright © 2026 NetApp, Inc. All Rights Reserved. Printed in the U.S.このドキュメントは著作権によって保護されています。著作権所有者の書面による事前承諾がある場合を除き、画像媒体、電子媒体、および写真複写、記録媒体、テープ媒体、電子検索システムへの組み込みを含む機械媒体など、いかなる形式および方法による複製も禁止します。

ネットアップの著作物から派生したソフトウェアは、次に示す使用許諾条項および免責条項の対象となります。

このソフトウェアは、ネットアップによって「現状のまま」提供されています。ネットアップは明示的な保証、または商品性および特定目的に対する適合性の暗示的保証を含み、かつこれに限定されないいかなる暗示的な保証も行いません。ネットアップは、代替品または代替サービスの調達、使用不能、データ損失、利益損失、業務中断を含み、かつこれに限定されない、このソフトウェアの使用により生じたすべての直接的損害、間接的損害、偶発的損害、特別損害、懲罰的損害、必然的損害の発生に対して、損失の発生の可能性が通知されていたとしても、その発生理由、根拠とする責任論、契約の有無、厳格責任、不法行為（過失またはそうでない場合を含む）にかかわらず、一切の責任を負いません。

ネットアップは、ここに記載されているすべての製品に対する変更を随時、予告なく行う権利を保有します。ネットアップによる明示的な書面による合意がある場合を除き、ここに記載されている製品の使用により生じる責任および義務に対して、ネットアップは責任を負いません。この製品の使用または購入は、ネットアップの特許権、商標権、または他の知的所有権に基づくライセンスの供与とはみなされません。

このマニュアルに記載されている製品は、1つ以上の米国特許、その他の国の特許、および出願中の特許によって保護されている場合があります。

権利の制限について：政府による使用、複製、開示は、DFARS 252.227-7013（2014年2月）およびFAR 5252.227-19（2007年12月）のRights in Technical Data -Noncommercial Items（技術データ - 非商用品目に関する諸権利）条項の(b)(3)項、に規定された制限が適用されます。

本書に含まれるデータは商用製品および / または商用サービス（FAR 2.101の定義に基づく）に関係し、データの所有権はNetApp, Inc.にあります。本契約に基づき提供されるすべてのネットアップの技術データおよびコンピュータソフトウェアは、商用目的であり、私費のみで開発されたものです。米国政府は本データに対し、非独占的かつ移転およびサブライセンス不可で、全世界を対象とする取り消し不能の制限付き使用权を有し、本データの提供の根拠となった米国政府契約に関連し、当該契約の裏付けとする場合にのみ本データを使用できます。前述の場合を除き、NetApp, Inc.の書面による許可を事前に得ることなく、本データを使用、開示、転載、改変するほか、上演または展示することはできません。国防総省にかかる米国政府のデータ使用权については、DFARS 252.227-7015(b)項（2014年2月）で定められた権利のみが認められます。

商標に関する情報

NetApp、NetAppのロゴ、<http://www.netapp.com/TM>に記載されているマークは、NetApp, Inc.の商標です。その他の会社名と製品名は、それを所有する各社の商標である場合があります。