



Oracle Linux の場合

ONTAP SAN Host Utilities

NetApp
January 26, 2026

目次

Oracle Linux の場合	1
Oracle LinuxホストのONTAPサポートと機能について学ぶ	1
次の手順	2
ONTAPストレージ用にNVMe-oFを使用してOracle Linux 9.xを構成する	2
手順1：必要に応じてSANブートを有効にします。	2
ステップ2: Oracle Linux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3: NVMe/FCとNVMe/TCPを構成する	4
ステップ4: オプションとして、udevルールのiopolicyを変更します。	12
ステップ5: オプションでNVMe/FCの1MB I/Oを有効にする	13
ステップ6: NVMeブートサービスを確認する	14
ステップ7: マルチパス構成を確認する	15
ステップ8: 安全なインバンド認証を設定する	20
手順9：既知の問題を確認する	25
ONTAPストレージ用にNVMe-oFを使用してOracle Linux 8.xを構成する	25
ステップ1: Oracle LinuxとNVMeソフトウェアをインストールし、構成を確認する	26
ステップ2: NVMe/FCとNVMe/TCPを構成する	28
ステップ3: オプションでNVMe/FCの1MB I/Oを有効にする	35
ステップ4: マルチパス構成を確認する	36
ステップ5: オプションで1MBのI/Oサイズを有効にする	40
手順6：既知の問題を確認する	41
ONTAPストレージ用にNVMe-oFを使用してOracle Linux 7.xを構成する	41
ステップ1: Oracle LinuxとNVMeソフトウェアをインストールし、構成を確認する	42
ステップ2: NVMe/FCを構成する	43
ステップ3: オプションでNVMe/FCの1MB I/Oを有効にする	45
ステップ4: マルチパス構成を確認する	46
ステップ5: 既知の問題を確認する	48

Oracle Linux の場合

Oracle Linux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および Oracle Linux のバージョンによって異なります。

特徴	Oracle Linuxホストバージョン	ONTAPのバージョン
Oracle LinuxホストとONTAPコントローラ間のNVMe/TCP経由の安全なインバンド認証がサポートされています。	9.4以降	9.12.1以降
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以降	9.10.1以降
NVMe/TCPはネイティブを使用して名前空間を提供します `nvme-cli` パッケージ	8.2以降	9.10.1以降
NVMe と SCSI トライフィックは、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。	7.7以降	9.4以降



ネットアップ `sanlun` ホストユーティリティはNVMe-oFをサポートしていません。代わりに、ネイティブのNetAppプラグインを使用できます。`nvme-cli` すべての NVMe-oF トランスポート用。

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	Oracle Linuxホストバージョン
`nvme-cli` パッケージ内のネイティブudevルールは、NVMeマルチパスのキュー深度ロード バランシングを提供します	9.6以降
SANポートはNVMe/FCプロトコルを使用して有効化されます	9.5以降
NVMe名前空間のカーネル内NVMeマルチパスはデフォルトで有効になっています	8.3以降
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。	8.3以降
`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	8.3以降



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

Oracle Linux バージョンが..の場合	について学ぶ..
9シリーズ	"Oracle Linux 9.x 用の NVMe の構成"
8シリーズ	"Oracle Linux 8.x 用の NVMe の構成"
7シリーズ	"Oracle Linux 7.x 用の NVMe の構成"

関連情報

["NVMeプロトコルの管理について学ぶ"](#)

ONTAPストレージ用にNVMe-oFを使用してOracle Linux 9.xを構成する

Oracle Linux 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。

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

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

- その `nvme disconnect-all` このコマンドはルートファイルシステムとデータファイルシステムの両方を切断し、システムが不安定になる可能性があります。 NVMe-TCP または NVMe-FC 名前空間を介して SAN から起動するシステムではこれを発行しないでください。

手順1：必要に応じてSANブートを有効にします。

SAN ブートを使用するようにホストを構成すると、展開が簡素化され、スケーラビリティが向上します。使用["Interoperability Matrix Tool"](#)Linux OS、ホスト バス アダプタ (HBA)、HBA フームウェア、HBA ブート BIOS、およびONTAPバージョンが SAN ブートをサポートしていることを確認します。

手順

1. ["NVMe名前空間を作成し、ホストにマッピングする"](#)。
2. SAN ブート名前空間がマップされているポートに対して、サーバー BIOS で SAN ブートを有効にします。

HBA BIOS を有効にする方法については、ベンダー固有のマニュアルを参照してください。

3. ホストを再起動し、OS が起動して実行されていることを確認します。

ステップ2: Oracle LinuxとNVMeソフトウェアをインストールし、構成を確認する

サポートされている Oracle Linux 9.x ソフトウェアの最小バージョンを検証するには、次の手順に従います。

手順

1. サーバーに Oracle Linux 9.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 Oracle Linux 9.x カーネルが実行されていることを確認します。

```
uname -r
```

Oracle Linuxカーネルバージョンの例:

```
6.12.0-1.23.3.2.el9uek.x86_64
```

2. 「nvme-cli」パッケージをインストールします。

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

次の例は、`nvme-cli`パッケージバージョン:

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

3. をインストールします libnvme パッケージ:

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

次の例は、`libnvme`パッケージバージョン:

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

4. Oracle Linux 9.xホストで、`hostnqn`文字列 `/etc/nvme/hostnqn`:

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

次の例は、`hostnqn`バージョン:

```
nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
```

5. ONTAPシステムで、`hostnqn`文字列が一致する `hostnqn`ONTAPストレージシステム上の対応するサブシステムの文字列:

```
vserver nvme subsystem host show -vserver vs_203
```

例を示します

Vserver	Subsystem	Priority	Host	NQN
vs_203	Nvme1	regular	nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb	
	Nvme10	regular	nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb	
	Nvme11	regular	nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb	
	Nvme12	regular	nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb	
	Nvme13	regular	nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb	
	Nvme14	regular	nqn.2014-08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb	



文字列が一致しない場合 `hostnqn` は、コマンドを使用して、対応するONTAPアレイサブシステムの文字列を更新し、`hostnqn` ホストののの文字列 `/etc/nvme/hostnqn` と一致させる `hostnqn` ことができます `vserver modify`。

ステップ3: 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 - プロードコム/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。

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

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

次の例はファームウェアのバージョンを示しています。

```
14.4.576.17, sli-4:6:d  
14.4.576.17, 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。

```
0:14.4.0.8
```

+

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>
```

次の例はポート ID を示しています。

```
0x2100f4c7aa9d7c5c  
0x2100f4c7aa9d7c5d
```

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 lpfco Total 6144 IO 5894 ELS 250
NVME LPORT lpfco WWPN x100000620b3c0869 WWNN x200000620b3c0869
DID x080e00 ONLINE
NVME RPORT WWPN x2001d039eabac36f WWNN x2000d039eabac36f
DID x021401 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x20e2d039eabac36f WWNN x20e1d039eabac36f
DID x02141f TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2011d039eabac36f WWNN x2010d039eabac36f
DID x021429 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2002d039eabac36f WWNN x2000d039eabac36f
DID x021003 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x20e4d039eabac36f WWNN x20e1d039eabac36f
DID x02100f TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2012d039eabac36f WWNN x2010d039eabac36f
DID x021015 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
LS: Xmt 0000027ccf Cmpl 0000027cca Abort 00000014
LS XMIT: Err 00000005 CMPL: xb 00000014 Err 00000014
Total FCP Cmpl 0000000000613ff Issue 0000000000613fc OutIO
fffffffffffffd
    abort 00000007 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 000000a Err 000000d
```

```
NVME Initiator Enabled
XRI Dist lpfcl Total 6144 IO 5894 ELS 250
NVME LPORT lpfcl WWPN x100000620b3c086a WWNN x200000620b3c086a
DID x080000 ONLINE
NVME RPORT WWPN x2004d039eabac36f WWNN x2000d039eabac36f
DID x021501 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x20e3d039eabac36f WWNN x20e1d039eabac36f
DID x02150f TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2014d039eabac36f WWNN x2010d039eabac36f
DID x021515 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2003d039eabac36f WWNN x2000d039eabac36f
DID x02110b TARGET DISCSRVC ONLINE
NVME RPORT WWPN x20e5d039eabac36f WWNN x20e1d039eabac36f
DID x02111f TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2013d039eabac36f WWNN x2010d039eabac36f
DID x021129 TARGET DISCSRVC ONLINE
```

```
NVME Statistics
```

```
LS: Xmt 0000027ca3 Cmpl 0000027ca2 Abort 00000017
LS XMIT: Err 00000001 CMPL: xb 00000017 Err 00000017
Total FCP Cmpl 000000000006369d Issue 000000000006369a OutIO
fffffffffffffd
    abort 00000007 noxri 00000000 nondlp 00000011 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000008 Err 0000000c
```

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とファームウェアのバージョンが実行され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:v9.15.03 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. イニシエータポートがサポートされているNVMe/TCP LIFの検出口ログページの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 192.168.30.10 -a 192.168.30.58
=====Discovery Log Entry 0=====
trtype: tcp
adrifam: ipv4
subtype: current discovery subsystem
treq: not specified
portid: 8
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr: 192.168.31.99
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: 6
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr: 192.168.30.99
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: 7
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr: 192.168.31.98
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:  5
trsvcid: 8009
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:discovery
traddr:  192.168.30.98
eflags:  explicit discovery connections, duplicate discovery
information
sectype: none
=====Discovery Log Entry 4=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  8
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr:  192.168.31.99
eflags:  none
sectype: none
=====Discovery Log Entry 5=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  6
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr:  192.168.30.99
eflags:  none
sectype: none
=====Discovery Log Entry 6=====
trtype:  tcp
adrifam:  ipv4
subtype: nvme subsystem
treq:    not specified
portid:  7
trsvcid: 4420
subnqn:  nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
```

```
traddr: 192.168.31.98
eflags: none
sectype: none
=====Discovery Log Entry 7=====
trtype: tcp
adrfrm: ipv4
subtype: nvme subsystem
treq: not specified
portid: 5
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.064a9b19b3ee11f09dcad039eabac370:subsystem.subs
ys_kvm
traddr: 192.168.30.98
eflags: none
sectype: none
```

2. NVMe/TCPイニシエータとターゲットLIFの他の組み合わせで、検出ログページの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 192.168.30.10 -a 192.168.30.58
nvme discover -t tcp -w 192.168.30.10 -a 192.168.30.59
nvme discover -t tcp -w 192.168.31.10 -a 192.168.31.58
nvme discover -t tcp -w 192.168.31.10 -a 192.168.31.59
```

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/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 192.168.30.10 -a 192.168.30.58
nvme connect-all -t tcp -w 192.168.30.10 -a 192.168.30.59
nvme connect-all -t tcp -w 192.168.31.10 -a 192.168.31.58
nvme connect-all -t tcp -w 192.168.31.10 -a 192.168.31.59
```

Oracle Linux 9.4以降、NVMe/TCPの設定は`ctrl_loss_tmo timeout`自動的に「オフ」に設定されます。その結果、次のようにになります

- 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4: オプションとして、udevルールのiopolicyを変更します。

Oracle Linux 9.xホストはNVMe-oFのデフォルトのiopolicyを次のように設定します。`round-robin`。Oracle Linux 9.6以降では、iopolicyを次のように変更できます。`queue-depth` udev ルール ファイルを変更します。

手順

- ルート権限でテキスト エディターで udev ルール ファイルを開きます。

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

次の出力が表示されます。

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

- NetApp ONTAPコントローラの iopolicy を設定する行を見つけます。

次の例は、ルールの例を示しています。

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

- ルールを修正して`round-robin`なる`queue-depth`:

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

4. udev ルールを再読み込みし、変更を適用します。

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

5. サブシステムの現在の iopolicy を確認します。<subsystem>を置き換えます。例: nvme-subsys0。

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

次の出力が表示されます。

```
queue-depth.
```



新しい iopolicy は、一致する NetApp ONTAP コントローラ デバイスに自動的に適用されます。
再起動は不要です。

ステップ5: 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. コマンドを実行し dracut -f、ホストをリブートします。
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6: NVMeブートサービスを確認する

Oracle Linux 9.5以降では、`nvmefc-boot-connections.service` そして `nvmf-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、`nvmefc-boot-connections.service` そして `nvmf-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 `nvmf-autoconnect.service` ます。

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

出力例を表示します。

```
nvmf-autoconnect.service - Connect NVMe-oF subsystems automatically
during boot
   Loaded: loaded (/usr/lib/systemd/system/nvmf-
autoconnect.service; enabled; preset: disabled)
   Active: inactive (dead) since Tue 2025-10-07 09:48:11 EDT; 1
week 0 days ago
     Main PID: 2620 (code=exited, status=0/SUCCESS)
       CPU: 19ms

Oct 07 09:48:11 R650xs-13-211 systemd[1]: Starting Connect NVMe-oF
subsystems automatically during boot...
Oct 07 09:48:11 R650xs-13-211 systemd[1]: nvmf-autoconnect.service:
Deactivated successfully.
Oct 07 09:48:11 R650xs-13-211 systemd[1]: Finished Connect NVMe-oF
subsystems automatically during boot.
```

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

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

出力例を表示します。

```
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 Tue 2025-10-07 09:47:07 EDT; 1 week 0 days ago
     Main PID: 1651 (code=exited, status=0/SUCCESS)
       CPU: 14ms

Oct 07 09:47:07 R650xs-13-211 systemd[1]: Starting Auto-connect to subsystems on FC-NVME devices found during boot...
Oct 07 09:47:07 R650xs-13-211 systemd[1]: nvmefc-boot-connections.service: Deactivated successfully.
Oct 07 09:47:07 R650xs-13-211 systemd[1]: Finished Auto-connect to subsystems on FC-NVME devices found during boot.
```

ステップ7: 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ネームスペースが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. 該当するONTAPネームスペースの適切なNVMe-oF設定（modelをNetApp ONTAPコントローラに設定し、load balancing iopolicyをラウンドロビンに設定するなど）がホストに正しく反映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-subsy*/*iopolicy
```

iopolicy に設定された値が表示されます。次に例を示します。

```
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 Namespace	Generic Usage	SN	Format	Model FW Rev
/dev/nvme102n1 Controller	0x1	81LLqNYTindCAAAAAAAk	2.25 GB / 5.37 GB	NetApp ONTAP 9.17.1
/dev/nvme102n2 Controller	0x2	81LLqNYTindCAAAAAAAk	2.25 GB / 5.37 GB	NetApp ONTAP 9.17.1
/dev/nvme106n1 Controller	0x1	81LLqNYTindCAAAAAAAs	2.25 GB / 5.37 GB	NetApp ONTAP 9.17.1
/dev/nvme106n2 Controller	0x2	81LLqNYTindCAAAAAAAs	2.25 GB / 5.37 GB	NetApp ONTAP 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/nvme4n5
```

例を示します

```
nvme-subsys4 - NQN=nqn.1992-  
08.com.netapp:sn.f9c6d0cb4fef11f08579d039eaa8138c:discovery  
hostnqn=nqn.2014-08.org.nvmexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-  
3a68dd61a1cb \ +- nvme2 fc traddr=nn-0x201ad039eabac36f:pn-  
0x201bd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-  
0x2100f4c7aa9d7c5c live optimized  
+- nvme8 fc traddr=nn-0x201ad039eabac36f:pn-  
0x201dd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-  
0x2100f4c7aa9d7c5d live non-optimized  
+- nvme2 fc traddr=nn-0x201ad039eabac36f:pn-  
0x201bd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-  
0x2100f4c7aa9d7c5c live non-optimized  
+- nvme8 fc traddr=nn-0x201ad039eabac36f:pn-  
0x201dd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-  
0x2100f4c7aa9d7c5d live optimized
```

NVMe/FC

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

例を示します

```
nvme-subsy98 - NQN=nqn.1992-
08.com.netapp:sn.f9c6d0cb4fef11f08579d039eaa8138c:subsystem.Nvme
9
hostnqn=nqn.2014-
08.org.nvmeexpress:uuid:b1d95cd0-1f7c-11ec-b8d1-3a68dd61a1cb
\
+- nvme100 fc traddr=nn-0x201ad039eabac36f:pn-
0x201dd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-
0x2100f4c7aa9d7c5d live non-optimized
+- nvme101 fc traddr=nn-0x201ad039eabac36f:pn-
0x201cd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-
0x2100f4c7aa9d7c5c live non-optimized
+- nvme98 fc traddr=nn-0x201ad039eabac36f:pn-
0x201bd039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5c:pn-
0x2100f4c7aa9d7c5c live optimized
+- nvme99 fc traddr=nn-0x201ad039eabac36f:pn-
0x201ed039eabac36f,host_traddr=nn-0x2000f4c7aa9d7c5d:pn-
0x2100f4c7aa9d7c5d live optimized
[root@SR630-13-203 ~]#
```

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 Size	Vserver	Namespace Path	NSID	UUID
/dev/nvme102n1 00e760c9-e4ca-4d9f-b1d4-e9a930bf53c0	vs_203	/vol/Nvmevol35/ns35	1	
/dev/nvme102n2 1fa97524-7dc2-4dbc-b4cf-5dda9e7095c0	vs_203	/vol/Nvmevol183/ns83	2	5.37GB

JSON

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

例を示します

```
{  
  "ONTAPdevices": [  
    {  
      "Device": "/dev/nvme11n1",  
      "Vserver": "vs_203",  
      "Namespace_Path": "/vol/Nvmevol16/ns16",  
      "NSID": 1,  
      "UUID": "18a88771-8b5b-4eb7-bff0-2ae261f488e4",  
      "LBA_Size": 4096,  
      "Namespace_Size": 5368709120,  
      "UsedBytes": 2262282240,  
      "Version": "9.17.1"  
    }  
  ]  
}
```

ステップ8: 安全なインバンド認証を設定する

Oracle Linux 9.x ホストとONTAPコントローラ間の NVMe/TCP 経由の安全なインバンド認証がサポートされます。

安全な認証を設定するには、各ホストまたはコントローラを DH-HMAC-CHAP キーに関連付ける必要があります。 DH-HMAC-CHAP キーは、NVMe ホストまたはコントローラの NQN と管理者によって設定された認証シークレットの組み合わせです。ピアを認証するには、NVMe ホストまたはコントローラはピアに関連付けられたキーを認識する必要があります。

手順

CLI または構成 JSON ファイルを使用して、安全なインバンド認証を設定します。異なるサブシステムに異なる dhchap キーを指定する必要がある場合は、構成 JSON ファイルを使用します。

CLI の使用

CLIを使用してセキュアなインバンド認証を設定します。

1. ホストNQNを取得します。

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

2. Linux ホストの dhchap キーを生成します。

コマンドパラメータの出力を次に示し `gen-dhchap-key` ます。

```
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
```

次の例では、HMACが3に設定されたランダムDHCHAPキー (SHA-512) が生成されます。

```
# nvme gen-dhchap-key -m 3 -n nqn.2014-
08.org.nvmexpress:uuid:4c4c4544-0056-5410-8048-c4c04f425633
DHHC-
1:03:xhAfAD5IVLZDxiVbmFEOA5JZ3F/ERqTXhHzZQJKgkYkTbP19dhRyVtr4dBD+SG
iAJ03by4FbnVtov1Lmk+86+nNc6k=:
```

3. ONTAPコントローラで、ホストを追加し、両方のDHCHAPキーを指定します。

```
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. ホストは、単方向と双方向の2種類の認証方式をサポートします。ホストで、ONTAPコントローラに接続し、選択した認証方式に基づいてDHCHAPキーを指定します。

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

5. 検証する nvme connect authentication ホストとコントローラのDHCHAPキーを確認してコマンドを実行します。

- a. ホストDHCHAPキーを確認します。

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

に、単方向設定の出力例を示します。

```
cat /sys/class/nvme-subsystem/nvme-subsys1/nvme*/dhchap_secret
DHHC-
1:03:Y5VkkESgmtTGNDX842qemNpFK6BXYYVwwnqErgt3IQKP5Fbjje\JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\GwKoRp6AwGkw=:
DHHC-
1:03:Y5VkkESgmtTGNDX842qemNpFK6BXYYVwwnqErgt3IQKP5Fbjje\JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\GwKoRp6AwGkw=:
DHHC-
1:03:Y5VkkESgmtTGNDX842qemNpFK6BXYYVwwnqErgt3IQKP5Fbjje\JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\GwKoRp6AwGkw=:
DHHC-
1:03:Y5VkkESgmtTGNDX842qemNpFK6BXYYVwwnqErgt3IQKP5Fbjje\JSBOjG
5Ea3NBLRfuiAuUSDUto6eY\GwKoRp6AwGkw=:
```

- b. コントローラのDHCHAPキーを確認します。

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

に、双方向設定の出力例を示します。

```
cat /sys/class/nvme-subsystem/nvme-
subsys6/nvme*/dhchap_ctrl_secret
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
DHHC-
1:03:frpLlTrnOYtcWDxPzq4ccxU1UrH2FjV7hYw5s2XEDB+lo+TjMsOwHR\/
FtM0nBBidx+gdoyUcC5s6hOOtTLDGcz0Kbs=:
```

JSON

ONTAPコントローラ構成で複数のNVMeサブシステムを使用できる場合は、コマンドでファイルを `nvme connect-all` 使用できます `~/etc/nvme/config.json`。

使用 `-o` JSON ファイルを生成するオプション。その他の構文オプションについては、`nvme connect - all` のマニュアルページを参照してください。

1. JSON ファイルを設定します。

例を示します

```
[  
  {  
    "hostnqn": "nqn.2014-08.org.nvmeexpress:uuid:4c4c4544-0056-  
5410-8048-c4c04f425633",  
    "hostid": "4c4c4544-0056-5410-8048-c4c04f425633",  
    "dhchap_key": "DHHC-  
1:01:nFg06gV0FNpXqoiLOF0L+swULQpZU/PjU9v/McDeJHjTZFlF:",  
    "subsystems": [  
      {  
        "nqn": "nqn.1992-  
08.com.netapp:sn.09035a8d8c8011f0ac0fd039eabac370:subsystem.subs  
ys",  
        "ports": [  
          {  
            "transport": "tcp",  
            "traddr": "192.168.30.69",  
            "host_traddr": "192.168.30.10",  
            "trsvcid": "4420",  
            "dhchap_ctrl_key": "DHHC-  
1:03:n3F8d+bvxKW/s+lEhqXaOohI2sxrQ9iLutzduuFq49JgdjjaFtTpDSO9kQ1  
/bvZj+Bo3rdHh3xPxEp6a4xyhcRyqdds=":  
          }  
        ]  
      }  
    ]  
  }  
]
```



上記の例では、はに対応し、は `dhchap_key` に対応 `dhchap_secret` し `dhchap_ctrl_key` `dhchap_ctrl_secret` ます。

2. config jsonファイルを使用してONTAPコントローラに接続します。

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

3. 各サブシステムの各コントローラでDHCHAPシークレットが有効になっていることを確認します。

- a. ホストDHCHAPキーを確認します。

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

次の例は、dhchap キーを示しています。

```
DHHC-1:01:nFg06gV0FNpXqoiLOF0L+swULQpZU/PjU9v/McDeJHjTZF1F:
```

b. コントローラのDHCHAPキーを確認します。

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

次の例のような出力が表示されます。

```
DHHC-
1:03:n3F8d+bvxKW/s+1EhqXaOohI2sxrQ9iLutzduuFq49JgdjjaFtTpDS09kQ1/bvZ
j+Bo3rdHh3xPXeP6a4xyhcRyqdds=:
```

手順9：既知の問題を確認する

既知の問題はありません。

ONTAPストレージ用にNVMe-oFを使用してOracle Linux 8.xを構成する

Oracle Linux 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。

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

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

- NVMe-oF プロトコルを使用した SAN ブートはサポートされていません。
- NetApp sanlun ホスト ユーティリティのサポートは、Oracle Linux 8.x ホスト上の NVMe-oF では使用できません。代わりに、ネイティブに含まれるNetAppプラグインを利用できます。`nvme-cli` すべての NVMe-oF トランスポート用のパッケージ。
- Oracle Linux 8.2 以前では、ネイティブ NVMe/FC 自動接続スクリプトは nvme-cli パッケージで使用できません。HBA ベンダーが提供する外部自動接続スクリプトを使用します。
- Oracle Linux 8.2 以前では、NVMe マルチパスのラウンドロビン ロード バランシングはデフォルトで有効になっていません。この機能を有効にするには、[udevルールの作成](#)。

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

サポートされている Oracle Linux 8.x ソフトウェアの最小バージョンを検証するには、次の手順に従います。

手順

1. サーバーに Oracle Linux 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 Oracle Linux 8.x カーネルが実行されていることを確認します。

```
uname -r
```

Oracle Linuxカーネルバージョンの例:

```
5.15.0-206.153.7.1.el8uek.x86_64
```

2. 「nvme-cli」パッケージをインストールします。

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

次の例は、`nvme-cli`パッケージバージョン:

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

3. Oracle Linux 8.2以前の場合、次の文字列を別のudevルールとして追加します。

/lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules。これにより、NVMe マルチパスのラウンドロビン ロード バランシングが可能になります。

```
cat /lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules
Enable round-robin for NetApp ONTAP
ACTION=="add", SUBSYSTEMS=="nvme-subsystem", ATTRS{model}=="NetApp ONTAP
Controller", ATTR{iopolicy}="round-robin"
```

4. Oracle Linux 8.xホストで、`hostnqn`文字列 `/etc/nvme/hostnqn`:

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

次の例は、`hostnqn`バージョン:

```
nqn.2014-08.org.nvmeexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
```

5. ONTAPシステムで、`hostnqn`文字列が一致する`hostnqn`ONTAPストレージシステム上の対応するサブ

システムの文字列:

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

例を示します

```
Vserver Subsystem Priority Host NQN
----- -----
-----
vs_coexistence_LPE36002
    nvme
        regular    nqn.2014-
08.org.nvmeexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
    nvme1
        regular    nqn.2014-
08.org.nvmeexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
    nvme2
        regular    nqn.2014-
08.org.nvmeexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
    nvme3
        regular    nqn.2014-
08.org.nvmeexpress:uuid:edd38060-00f7-47aa-a9dc-4d8ae0cd969a
4 entries were displayed.
```



文字列が一致しない場合 `hostnqn` は、コマンドを使用し `vserver modify` て、対応するONTAPアレイサブシステムの文字列をホストののの文字列 `/etc/nvme/hostnqn` と一致するように `hostnqn` 更新し `hostnqn` ます。

6. オプションとして、同じホスト上でNVMeとSCSIの共存トラフィックを実行するために、NetApp はONTAPネームスペースにカーネル内NVMeマルチパスを使用することを推奨しています。`dm-multipath` それぞれONTAP LUN 用です。これにより、ONTAPネームスペースが除外されます。`dm-multipath` 防止する `dm-multipath` ONTAPネームスペースデバイスを要求できなくなります。
 - a. 追加する `enable_foreign` 設定する `/etc/multipath.conf` ファイル。

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

- b. 再起動する `multipathd` 新しい設定を適用するデーモン。

```
systemctl restart multipathd
```

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。

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。

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

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

次の例はファームウェアのバージョンを示しています。

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

```
0:14.2.0.13
```

+

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. が「3」に設定されていることを確認し `lpfc_enable_fc4_type` ます。

```
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>
```

次の例はポート ID を示しています。

```
0x100000109bf0449c  
0x100000109bf0449d
```

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 lpfco Total 6144 IO 5894 ELS 250
NVME LPORT lpfco WWPN x100000109bf0449c WWNN x200000109bf0449c
DID x061500 ONLINE
NVME RPORT WWPN x200bd039eab31e9c WWNN x2005d039eab31e9c
DID x020e06 TARGET DISCSRVC ONLINE
NVME RPORT WWPN x2006d039eab31e9c WWNN x2005d039eab31e9c
DID x020a0a TARGET DISCSRVC ONLINE
NVME Statistics
LS: Xmt 000000002c Cmpl 000000002c Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 000000000008ffe8 Issue 000000000008ffb9 OutIO
fffffffffffffd1
    abort 0000000c noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 0000000c Err 0000000c
NVME Initiator Enabled
XRI Dist lpfcl Total 6144 IO 5894 ELS 250
NVME LPORT lpfcl WWPN x100000109bf0449d WWNN x200000109bf0449d
DID x062d00 ONLINE
NVME RPORT WWPN x201fd039eab31e9c WWNN x2005d039eab31e9c
DID x02090a TARGET DISCSRVC ONLINE
NVME RPORT WWPN x200cd039eab31e9c WWNN x2005d039eab31e9c
DID x020d06 TARGET DISCSRVC ONLINE
NVME Statistics
LS: Xmt 000000041 Cmpl 000000041 Abort 00000000
LS XMIT: Err 00000000 CMPL: xb 00000000 Err 00000000
Total FCP Cmpl 00000000000936bf Issue 000000000009369a OutIO
fffffffffffffdb
    abort 00000016 noxri 00000000 nondlp 00000000 qdepth
00000000 wqerr 00000000 err 00000000
FCP CMPL: xb 00000016 Err 00000016
```

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とファームウェアのバージョンが実行され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。

```
QLE2772 FW:v9.15.00 DVR:v10.02.09.100-k
QLE2772 FW:v9.15.00 DVR:v10.02.09.100-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。

TCP

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。

1. イニシエータポートがサポートされているNVMe/TCP LIFの検出口グページの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 192.168.6.1 -a 192.168.6.24 Discovery
Log Number of Records 20, Generation counter 45
=====Discovery Log Entry 0=====
trtype: tcp
adrfrm: ipv4
subtype: unrecognized
treq: not specified
portid: 6
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr: 192.168.6.25
sectype: none
=====Discovery Log Entry 1=====
trtype: tcp
adrfrm: ipv4
subtype: unrecognized
treq: not specified
portid: 1
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr: 192.168.5.24
sectype: none
=====Discovery Log Entry 2=====
trtype: tcp
adrfrm: ipv4
subtype: unrecognized
treq: not specified
portid: 4
trsvcid: 8009
subnqn: nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr: 192.168.6.24
sectype: none
=====Discovery Log Entry 3=====
trtype: tcp
adrfrm: ipv4
subtype: unrecognized
treq: not specified
portid: 2
trsvcid: 8009
subnqn: nqn.1992-
```

```

08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:discovery
traddr: 192.168.5.25
sectype: none
=====Discovery Log Entry 4=====
trtype: tcp
adrifam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 6
trsvcid: 4420
subnqn: nqn.1992-
08.com.netapp:sn.e6c438e66ac211ef9ab8d039eab31e9d:subsystem.nvme
_tcp_4
traddr: 192.168.6.25
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.e6c438e66ac211ef9ab8d039eab31e9d:subsystem.nvme
_tcp_4
.....

```

2. NVMe/TCPイニシエータとターゲットLIFの他のすべての組み合わせで、検出口グページの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 192.168.6.1 -a 192.168.6.24
nvme discover -t tcp -w 192.168.6.1 -a 192.168.6.25
nvme discover -t tcp -w 192.168.5.1 -a 192.168.5.24
nvme discover -t tcp -w 192.168.5.1 -a 192.168.5.25

```

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/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 -l
<ctrl_loss_timeout_in_seconds>
```

例を示します

```
nvme connect-all -t tcp -w 192.168.5.1 -a 192.168.5.24
-l -1
nvme connect-all -t tcp -w 192.168.5.1 -a 192.168.5.25
-l -1
nvme connect-all -t tcp -w 192.168.6.1 -a 192.168.6.24
-l -1
nvme connect-all -t tcp -w 192.168.6.1 -a 192.168.6.25
-l -1
```

NetAppは、`ctrl-loss-tmo option`に`-1`NVMe/TCP イニシエーターは、パス損失が発生した場合に無期限に再接続を試行します。

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1MB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. コマンドを実行し dracut -f、ホストをリブートします。
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: 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ネームスペースが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. 該当するONTAPネームスペースの適切なNVMe-oF設定（modelをNetApp ONTAPコントローラに設定し、load balancing iopolicyをラウンドロビンに設定するなど）がホストに正しく反映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	SN	Model
/dev/nvme0n1	814vWBNRwf9HAAAAAAAB	NetApp ONTAP Controller
/dev/nvme0n2	814vWBNRwf9HAAAAAAAB	NetApp ONTAP Controller
/dev/nvme0n3	814vWBNRwf9HAAAAAAAB	NetApp ONTAP Controller

Namespace	Usage	Format	FW	Rev
1	85.90 GB	/ 85.90 GB	4 KiB + 0 B	FFFFFFFF
2	85.90 GB	/ 85.90 GB	24 KiB + 0 B	FFFFFFFF
3	85.90 GB	/ 85.90 GB	4 KiB + 0 B	FFFFFFFF

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 list-subsy /dev/nvme0n1
```

NVMe/FCの例を表示

```
nvme-subsy0 - NQN=nqn.1992- 08.com.netapp:  
4b4d82566aab11ef9ab8d039eab31e9d:subsystem.nvme\  
+- nvme1 fc traddr=nn-0x2038d039eab31e9c:pn-0x203ad039eab31e9c  
host_traddr=nn-0x200034800d756a89:pn-0x210034800d756a89 live  
optimized  
+- nvme2 fc traddr=nn-0x2038d039eab31e9c:pn-0x203cd039eab31e9c  
host_traddr=nn-0x200034800d756a88:pn-0x210034800d756a88 live  
optimized  
+- nvme3 fc traddr=nn-0x2038d039eab31e9c:pn-0x203ed039eab31e9c  
host_traddr=nn-0x200034800d756a89:pn-0x210034800d756a89 live non-optimized  
+- nvme7 fc traddr=nn-0x2038d039eab31e9c:pn-0x2039d039eab31e9c  
host_traddr=nn-0x200034800d756a88:pn-0x210034800d756a88 live non-optimized
```

NVMe/TCPの例を示す

```
nvme-subsy0 - NQN=nqn.1992- 08.com.netapp:  
sn.e6c438e66ac211ef9ab8d039eab31e9d:subsystem.nvme_tcp_4  
\  
+- nvme1 tcp traddr=192.168.5.25 trsvcid=4420  
host_traddr=192.168.5.1 src_addr=192.168.5.1 live optimized  
+- nvme10 tcp traddr=192.168.6.24 trsvcid=4420  
host_traddr=192.168.6.1 src_addr=192.168.6.1 live optimized  
+- nvme2 tcp traddr=192.168.5.24 trsvcid=4420  
host_traddr=192.168.5.1 src_addr=192.168.5.1 live non-optimized  
+- nvme9 tcp traddr=192.168.6.25 trsvcid=4420  
host_traddr=192.168.6.1 src_addr=192.168.6.1 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 NSID UUID	Vserver	Namespace Path Size
/dev/nvme0n1	vs_coexistence_QLE2772	
/vol/fcnvme_1_1_0/fcnvme_ns	1	159f9f88-be00-4828-aef6-197d289d4bd9 10.74GB
/dev/nvme0n2	vs_coexistence_QLE2772	
/vol/fcnvme_1_1_1/fcnvme_ns	2	2c1ef769-10c0-497d-86d7-e84811ed2df6 10.74GB
/dev/nvme0n3	vs_coexistence_QLE2772	
/vol/fcnvme_1_1_2/fcnvme_ns	3	9b49bf1a-8a08-4fa8-baf0-6ec6332ad5a4 10.74GB

JSON

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

例を示します

```
{  
  "ONTAPdevices" : [  
    {  
      "Device" : "/dev/nvme0n1",  
      "Vserver" : "vs_coexistence_QLE2772",  
      "Namespace_Path" : "/vol/fcnvme_1_1_0/fcnvme_ns",  
      "NSID" : 1,  
      "UUID" : "159f9f88-be00-4828-aef6-197d289d4bd9",  
      "Size" : "10.74GB",  
      "LBA_Data_Size" : 4096,  
      "Namespace_Size" : 2621440  
    },  
    {  
      "Device" : "/dev/nvme0n2",  
      "Vserver" : "vs_coexistence_QLE2772",  
      "Namespace_Path" : "/vol/fcnvme_1_1_1/fcnvme_ns",  
      "NSID" : 2,  
      "UUID" : "2c1ef769-10c0-497d-86d7-e84811ed2df6",  
      "Size" : "10.74GB",  
      "LBA_Data_Size" : 4096,  
      "Namespace_Size" : 2621440  
    },  
    {  
      "Device" : "/dev/nvme0n4",  
      "Vserver" : "vs_coexistence_QLE2772",  
      "Namespace_Path" : "/vol/fcnvme_1_1_3/fcnvme_ns",  
      "NSID" : 4,  
      "UUID" : "f3572189-2968-41bc-972a-9ee442dfaed7",  
      "Size" : "10.74GB",  
      "LBA_Data_Size" : 4096,  
      "Namespace_Size" : 2621440  
    },  
  ]  
}
```

ステップ5: オプションで1MBのI/Oサイズを有効にする

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1MB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. コマンドを実行し `dracut -f`、ホストをリブートします。

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
```

手順6：既知の問題を確認する

既知の問題は次のとおりです:

NetApp バグ ID	タイトル	説明
"1479047"	Oracle Linux 8.x NVMe-oFホストは重複した永続検出コントローラ (PDC) を作成します	NVMe-oF ホストでは、 <code>nvme discover -p</code> コマンドを使用して PDC を作成できます。このコマンドを使用する場合、イニシエーターとターゲットの組み合わせごとに 1 つの PDC のみを作成する必要があります。ただし、NVMe-oF ホストで Oracle Linux 8.x を実行している場合は、重複した PDC が毎回作成されます。`nvme discover -p` 実行されます。これにより、ホストとターゲットの両方でリソースが不要に使用されることになります。

ONTAPストレージ用にNVMe-oFを使用してOracle Linux 7.xを構成する

Oracle Linux 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。

Oracle Linux 7.x 用に NVMe over Fabrics (NVMe-oF) ホストを構成する方法を学習します。詳細なサポートと機能情報については、["Oracle Linux ONTAPのサポートと機能"](#)。

Oracle Linux 7.x の NVMe-oF には、次の既知の制限があります。

- NVMe-oF プロトコルを使用した SAN ブートはサポートされていません。
- NetApp sanlun ホスト ユーティリティのサポートは、Oracle Linux 7.x ホスト上の NVMe-oF では使用できません。代わりに、ネイティブに含まれるNetAppプラグインを利用できます。`nvme-cli`すべての NVMe-oF トランスポート用のパッケージ。
- NVMe/FC 標準の自動接続スクリプトは、nvme-CLI パッケージでは使用できません。HBA ベンダーが提供する外部自動接続スクリプトを使用します。
- NVMe マルチパスでは、ラウンドロビン ロード バランシングはデフォルトで有効になっていません。この機能を有効にするには、udev ルールを記述します。

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

サポートされている Oracle Linux 7.x ソフトウェアの最小バージョンを検証するには、次の手順に従います。

手順

1. サーバーに Oracle Linux 7.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 Oracle Linux 7.x カーネルが実行されていることを確認します。

```
uname -r
```

Oracle Linuxカーネルバージョンの例:

```
5.4.17-2011.6.2.el7uek.x86_64
```

2. 「nvme-cli」パッケージをインストールします。

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

次の例は、`nvme-cli`パッケージバージョン:

```
nvme-cli-1.8.1-3.el7.x86_64
```

3. 次の文字列を別のudevルールとして追加します。/lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules。これにより、NVMe マルチパスのラウンドロビン ロード バランシングが可能になります。

```
cat /lib/udev/rules.d/71-nvme-iopolicy-netapp-ONTAP.rules
Enable round-robin for NetApp ONTAP
ACTION=="add", SUBSYSTEMS=="nvme-subsystem", ATTRS{model}=="NetApp ONTAP Controller", ATTR{iopolicy}="round-robin"
```

4. Oracle Linux 7.xホストで、hostnqn`文字列 `/etc/nvme/hostnqn:

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

次の例は、`hostnqn`バージョン:

```
nqn.2014-08.org.nvmeexpress:uuid:497ad959-e6d0-4987-8dc2-a89267400874
```

5. ONTAPシステムで、`hostnqn`文字列が一致する`hostnqn`ONTAPストレージシステム上の対応するサブシステムの文字列:

```
*> vserver nvme subsystem host show -vserver vs_nvme_10
```

例を示します

```
Vserver Subsystem Host NQN
-----
ol_157_nvme_ss_10_0
nqn.2014-08.org.nvmeexpress:uuid:497ad959-e6d0-4987-8dc2-a89267400874
```



文字列が一致しない場合 `hostnqn` は、コマンドを使用し `vserver modify` で、対応するONTAPアレイサブシステムの文字列をホストののの文字列 `/etc/nvme/hostnqn` と一致するように `hostnqn` 更新し `hostnqn` ます。

6. ホストをリブートします。

ステップ2: NVMe/FCを構成する

Broadcom/Emulex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。

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

2. が「3」に設定されていることを確認し `lpfc_enable_fc4_type` ます。

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

3. 推奨される lpfc 自動接続スクリプトをインストールします。

```
rpm -ivh nvmefc-connect-12.8.264.0-1.noarch.rpm
```

4. 自動接続スクリプトがインストールされていることを確認します。

```
rpm -qa | grep nvmefc
```

次の出力が表示されます。

```
nvmefc-connect-12.8.264.0-1.noarch
```

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。

- ポート名を表示します:

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

次の出力が表示されます。

```
0x10000090fae0ec61  
0x10000090fae0ec62
```

- ポート名を表示します:

```
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 NVME 2947 SCSI 2947 ELS 250  
NVME LPORT lpfc0 WWPN x10000090fae0ec61 WWNN x20000090fae0ec61 DID  
x012000 ONLINE  
NVME RPORT WWPN x202d00a098c80f09 WWNN x202c00a098c80f09 DID x010201  
TARGET DISCSRVC ONLINE  
NVME RPORT WWPN x203100a098c80f09 WWNN x202c00a098c80f09 DID x010601  
TARGET DISCSRVC ONLINE
```

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1MB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. コマンドを実行し dracut -f、ホストをリブートします。
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: 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ネームスペースが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. 該当するONTAPネームスペースの適切なNVMe-oF設定（modelをNetApp ONTAPコントローラに設定し、load balancing iopolicyをラウンドロビンに設定するなど）がホストに正しく反映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 SN Model Namespace Usage Format FW Rev
-----
/dev/nvme0n1 80BADBKnB/JvAAAAAAAC NetApp ONTAP Controller 1 53.69 GB
/ 53.69 GB 4 KiB + 0 B FFFFFFFF
```

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 list-subsy /dev/nvme0n1
```

例を示します

```
Nvme-subsyf0 - NQN=nqn.1992-
08.com.netapp:sn.341541339b9511e8a9b500a098c80f09:subsystem.ol_157_n
vme_ss_10_0
\
+- nvme0 fc traddr=nn-0x202c00a098c80f09:pn-0x202d00a098c80f09
host_traddr=nn-0x20000090fae0ec61:pn-0x10000090fae0ec61 live
optimized
+- nvme1 fc traddr=nn-0x207300a098dfdd91:pn-0x207600a098dfdd91
host_traddr=nn-0x200000109b1c1204:pn-0x100000109b1c1204 live
inaccessible
+- nvme2 fc traddr=nn-0x207300a098dfdd91:pn-0x207500a098dfdd91
host_traddr=nn-0x200000109b1c1205:pn-0x100000109b1c1205 live
optimized
+- nvme3 fc traddr=nn-0x207300a098dfdd91:pn-0x207700a098dfdd91 host
traddr=nn-0x200000109b1c1205:pn-0x100000109b1c1205 live inaccessible
```

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	NSID	UUID	Size
/dev/nvme0n1	vs_nvme_10					
/vol/rhel_141_vol_10_0/ol_157_ns_10_0				1	55baf453-f629-4a18-9364-b6aee3f50dad	53.69GB

JSON

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

例を示します

```
{
  "ONTAPdevices" : [
    {
      "Device" : "/dev/nvme0n1",
      "Vserver" : "vs_nvme_10",
      "Namespace_Path" :
      "/vol/rhel_141_vol_10_0/ol_157_ns_10_0",
      "NSID" : 1,
      "UUID" : "55baf453-f629-4a18-9364-b6aee3f50dad",
      "Size" : "53.69GB",
      "LBA_Data_Size" : 4096,
      "Namespace_Size" : 13107200
    }
  ]
}
```

ステップ5: 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隨時、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 5225.227-19（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権を有し、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権については、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。