



コンバージドイーサネット (RoCE) バージョン 2 上の RDMA の構成

- [Windows で RoCEv2 を構成 \(1 ページ\)](#)
- [Linux で RoCEv2 を構成 \(8 ページ\)](#)
- [ESXi での RoCEv2 の構成 \(15 ページ\)](#)
- [UCS Manager CLI を使用した RoCEv2 インターフェイスの構成または削除 \(24 ページ\)](#)

Windows で RoCEv2 を構成

Windows での RoCEv2 モード 1 および 2 の構成の概要

Windows プラットフォームでの Remote Direct Memory Access (RDMA) over Converged Ethernet version 2 (RoCEv2) の設定では、最初に RoCEv2 モード 1、次に RoCEv2 モード 2 を設定する必要があります。モード 1 と 2 は、ネットワーク ダイレクト カーネル プロバイダ インターフェイス (NDKPI) の実装に関連しています。モード 1 はネイティブ RDMA、モード 2 には RDMA を使用する仮想ポートの設定が含まれています。

RoCEv2 モード 1 を設定するには、次の操作が必要です。

- サービス クラス (CoS) システム クラスで非ドロップ クラスを設定します。デフォルトでは、CoS 5 のプラチナが Cisco Unified Computing System (UCS) Manager のデフォルトです。
- Cisco UCS Manager でモード 1 のイーサネット アダプタ ポリシーを設定します。
- ホスト システムでモード 1 を設定します。

モード 2 を設定する前に、RoCEv2 モード 1 を設定する必要があります。

RoCEv2 モード 2 を設定するには、次の操作が必要です。

RoCEv2 のイーサネット仮想マシン キュー (VMQ) 接続ポリシーを作成するか、Cisco UCS Manager メッセージ キュー (MQ) サーバ メッセージ ブロック (SMB) 直接 (SMBd) ポリシーを使用します。

UCS Manager での SMB Direct モード1の設定

RDMA パケット ドロップの可能性を回避するには、ネットワーク全体で同じ非ドロップ COS が設定されていることを確認してください。

始める前に

UCSM QoS ポリシーで非ドロップクラスを設定し、RDMA でサポートされているインターフェイスに使用します。[LAN] > [LAN クラウド (LAN Cloud)] > [QoS システム クラス (QoS System Class)] に移動し、CoS 5 で [優先順位 (Priority) プラチナ] を有効にします。

手順

-
- ステップ 1** [Navigation] ペインで [Servers] をクリックします。
- ステップ 2** [Servers] > [Policies] の順に展開します。
- ステップ 3** ポリシーを作成する組織のノードを展開します。
- システムにマルチテナント機能が備えられていない場合は、[ルート (root)] ノードを展開します。
- ステップ 4** [アダプタ ポリシー (Adapter Policies)] を展開し、WIN-Hpn-smbd の既存のアダプタ ポリシーを選択します。
- ユーザー定義のアダプタ ポリシーを使用している場合は、次の設定手順を使用します。
- [全般 (General)] タブで、[RoCE] までスクロールし、[有効 (Enabled)] オプションボタンをクリックします。
 - [RoCE プロパティ (RoCE Properties)] フィールドの [バージョン 1 (Version 1)] の下で、[無効 (Disabled)] オプションボタンをクリックします。[バージョン 2 (Version 2)] については、[有効 (Enabled)] オプションボタンをクリックします。
 - [キュー ペア (Queue Pairs)] には、256 と入力します。
 - [メモリ領域 (Memory Regions)] には、131072 と入力します。
 - リソースグループの場合は、「2」を入力します。
 - [優先度 (Priority)] については、ドロップダウンから [プラチナ非ドロップ COS (Platinum No-Drop COS)] を選択します。
 - [変更の保存 (Save Changes)] をクリックします。
- ステップ 5** 次に、イーサネット アダプタ ポリシーを作成します。[ナビゲーション] ペインで、[LAN] をクリックします。
- ステップ 6** [LAN] > [ポリシー (Policies)] を展開します。
- ステップ 7** [vNIC Templates] ノードを右クリックし、[Create vNIC Template] を選択します。
- ステップ 8** [全般 (General)] タブの下で [vNIC プロパティ (vNIC Properties)] に移動し、次のように vNIC ポリシーの設定を変更します。
- MTU を 1500 または 4096 に設定します。

- b) アダプタ ポリシーの場合は、**[Win-HPN-SMBd]** を選択します。
- c) **[QoS ポリシー (QoS policy)]** の場合は、**[プラチナ (Platinum)]** を指定します。

ステップ 9 **[変更の保存 (Save Changes)]** をクリックします。

ステップ 10 変更を保存後、Cisco UCS Manager で再起動が指示されます。システムをリブートします。

次のタスク

サーバが復帰したら、ホスト上で RoCEv2 モード 1 を設定します。

ホスト システムでの SMB ダイレクト モード 1 の設定

この手順を実行して、2 個のホスト インターフェイスで smb クライアントと smb サーバ間の接続を構成します。これらのサーバのそれぞれに対して、smb クライアントおよび smb サーバで、RoCEv2 対応 vNIC を構成します。

始める前に

Cisco UCS Manager でモード 1 の RoCEv2 を構成します。

手順

ステップ 1 Windows ホストで、**[デバイス マネージャ (Device Manager)]** に移動し、適切な Cisco VIC インターネット インターフェイスを選択します。

ステップ 2 **[アドバンスド (Advanced)]** タブを選択し、**[ネットワーク ダイレクト機能 (Network Direct Functionality)]** プロパティが **[有効 (Enabled)]** になっていることを確認します。そうでない場合は有効にし、**[OK]** をクリックします。

smb サーバと smb クライアントの両方の vNICs に対してこの手順を実行します。

ステップ 3 **[ツール (Tools)]** > **[コンピュータ管理 (Computer Management)]** > **[デバイス マネージャ (Device Manager)]** > **[ネットワーク アダプタ (Network Adapter)]** > **[VIC ネットワーク アダプタ (VIC Network Adapter)]** をクリックし > **[プロパティ (Properties)]** > **[アドバンスド (Advanced)]** > **[ネットワーク ダイレクト機能 (Network Direct Functionality)]** に移動します。smb サーバと smb クライアント両方の vNICs に対してこの操作を実行します。

ステップ 4 PowerShell を使用して、ホスト オペレーティング システムで RoCE が有効になっていることを確認します。

Get-NetOffloadGlobalSetting コマンドを実行して、**NetworkDirect** が有効になっていることを確認します。

```
PS C:\Users\Administrator> Get-NetOffloadGlobalSetting

ReceiveSideScaling      : Enabled
ReceiveSegmentCoalescing : Enabled
Chimney                  : Disabled
TaskOffload              : Enabled
```

```

NetworkDirect           : Enabled
NetworkDirectAcrossIPSubnets : Blocked
PacketCoalescingFilter  : Disabled

```



(注) **NetworkDirect** 設定が無効と表示されている場合、次のコマンドを使用して有効にします：

Set-NetOffloadGlobalSetting -NetworkDirect enabled

ステップ 5 Powershell を起動し、**get -SmbClientNetworkInterface** コマンドを実行します。

```

PS C:\Users\Administrator>
PS C:\Users\Administrator> Get-SmbClientNetworkInterface

```

Interface	Index	RSS Capable	RDMA Capable	Speed	IpAddresses	Friendly Name
(vswitch)	14	True	False	40 Gbps	{10.37.60.162}	vEthernet
(vpl)	26	True	True	40 Gbps	{10.37.60.158}	vEthernet
	9	True	True	40 Gbps	{50.37.61.23}	Ethernet
	2					
	5	False	False	40 Gbps	{169.254.10.S}	Ethernet
(Kernel Debugger)	8	True	False	40 Gbps	{169.254.4.26}	Ethernet
	3					

```

PS C:\Users\Administrator>

```

ステップ 6 **enable - netadapterrdma [-name] ["Ethernetname"]** と入力します

ステップ 7 ホストでの全体的な RoCEv2 モード 1 の設定を確認します。

- a) Powershell コマンド **netstat-xan** を使用して、smb クライアントと smb サーバ Windows ホストの両方のリスナーを確認します。リスナーはコマンド出力に表示されます。

```

PS C:\Users\Administrator>
PS C:\Users\Administrator> netstat -xan
Active NetworkDirect Connections, Listeners, SharedEndpoints
Mode   IfIndex   Type    Local Address   Foreign Address   PID
-----
Kernel    9        Listener 50.37.61.23:445   NA                 0
Kernel   26        Listener 10.37.60.158:445  NA                 0
PS C:\Users\Administrator>

```

- b) **smb-client** サーバ ファイル共有に移動し、I/O 操作を開始します。
- c) パフォーマンス モニタに移動し、RDMA アクティビティが表示されていることを確認します。

ステップ 8 Powershell コマンドウィンドウで、**netstat-xan** 出力コマンドを使用して接続エントリをチェックして、表示されていることを確認します。コマンドプロンプトから **netstat-xan** を実行することもできます。接続エントリが **netstat xan** 出力に表示されている場合は、クライアントとサーバの間で RoCEv2 モード 1 接続が正しく確立されています。

```

PS C:\Users\Administrator> netstat -xan
Active NetworkDirect Connections, Listeners, SharedEndpoints
Mode   IfIndex   Type    Local Address   Foreign Address   PID
-----
Kernel    4        Connection 50.37.61.22:445   50.37.61.71:2240   0
Kernel    4        Connection 50.37.61.22:445   50.37.61.71:2496   0
Kernel   11        Connection 50.37.61.122:445  50.37.61.71:2752   0
Kernel   11        Connection 50.37.61.122:445  50.37.61.71:3008   0

```

Kernel	32	Connection	10.37.60.155:445	50.37.60.61:49092	0
Kernel	32	Connection	10.37.60.155:445	50.37.60.61:49348	0
Kernel	26	Connection	50.37.60.32:445	50.37.60.61:48580	0
Kernel	26	Connection	50.37.60.32:445	50.37.60.61:48836	0
Kernel	4	Listener	50.37.61.22:445	NA	0
Kernel	11	Listener	50.37.61.122:445	NA	0
Kernel	32	Listener	10.37.60.155:445	NA	0
Kernel	26	Listener	50.37.60.32:445	NA	0

ステップ 9 デフォルトでは、Microsoft の SMB ダイレクトは RDMA インターフェイスごとに 2 個の RDMA 接続を確立します。RDMA インターフェイスごとに RDMA 接続数を 1 個または複数の接続数に変更できます。

例えば、RDMA 接続数を 4 個に増やすには、PowerShell で次のコマンドを実行します：

```
PS C:\Users\Administrator> Set-ItemProperty -Path `
"HKLM:\SYSTEM\CurrentControlSet\Services
\LanmanWorkstation\Parameters" ConnectionCountPerRdmaNetworkInterface -Type DWORD -Value
4 -Force
```

UCS Manager でのモード 2 の設定

VMQ 接続ポリシーは vmmq として適用されます。

始める前に

モード 1 で RoCEv2 ポリシーを設定します。

事前定義されたデフォルトのアダプタ ポリシー「MQ-SMBd」を使用するか、または次の推奨される RoCE 固有のパラメータを使用してユーザー定義のイーサネット アダプタ ポリシーを設定します。

- RoCE：有効
- バージョン 1：無効
- バージョン 2：有効
- キュー ペア：256
- メモリ領域：65536
- リソース グループ：2
- 優先順位：プラチナ

次の値を使用して VMQ 接続ポリシーを作成します。

- マルチ キュー：有効
- サブ vNIC の数：16
- VMMQ アダプタ ポリシー：MQ-SMBd

手順

-
- ステップ 1 [Navigation] ペインで [Servers] をクリックします。
 - ステップ 2 [Servers] > [Service Profiles] の順に展開します。
 - ステップ 3 [サービス プロファイル (Service Profiles)] > [vNIC] を展開し、設定する VMQ 接続ポリシー プロファイルを選択します。
 - ステップ 4 [全般 (General)] タブの下にある [vNIC プロパティ (vNIC Properties)] に移動し、[ポリシー (Policies)] 領域までスクロールします。vNIC ポリシーの設定を次のように変更します。
 - a) アダプタ ポリシーの場合は、必ず **Win-HPN-SMBd** を使用するか、またはモード 1 の前に設定したアダプタ ポリシーを使用してください。
 - b) QoS ポリシーの場合は、[**best-effort**] を選択します。
 - ステップ 5 [Save Changes] をクリックします。
 - ステップ 6 [ナビゲーション] ペインで、[LAN] をクリックします。
 - ステップ 7 [LAN] > [ポリシー (Policies)] > [QoS ポリシー ベスト エフォート (QoS Policy Best Effort)] を展開します。
 - ステップ 8 [ホスト制御 (Host Control)] を [フル (Full)] に設定します。
 - ステップ 9 [Save Changes] をクリックします。
 - ステップ 10 変更を保存後、Cisco UCS Manager で再起動が指示されます。インターフェイスを再起動します。
-

次のタスク

サーバが復帰したら、ホストでモード 2 を設定します。

ホスト システムでの SMB ダイレクト モード 2 の設定

このタスクでは、Windows Server 2019 とそれ以降と互換性のある Hyper-V 仮想化ソフトウェアを使用します。

始める前に

- Cisco UCS マネージャ とホストの両方に対して、RoCEv2 モード 2 の接続を設定して確認します。
- Cisco UCS Manager でモード 2 の RoCEv2 を構成します。
- Windows ホスト サーバで Hyper-V を有効にします。

手順

ステップ 1 Hyper-V スイッチ マネージャに移動します。

ステップ 2 RoCEv2 対応イーサネット インターフェイスの新しい仮想ネットワーク スイッチ (vSwitch) を作成します。

- a) **[外部ネットワーク (External Network)]** を選択し、**[VIC イーサネット インターフェイス 2 (VIC Ethernet Interface 2)]** および **[管理オペレーティング システムでこのネットワーク アダプタの共有を許可する (Allow management operating system to share this network adapter)]** を選択します。
- b) **[OK]** をクリックして、仮想スイッチを作成します。

Powershell インターフェイスを起動します。

ステップ 3 デフォルト以外の vPort を設定し、次の Powershell コマンドを使用して RDMA を有効にします。

```
add-vmNetworkAdapter -switchname vswitch -name vpl -managementOS
```

```
enable-netAdapterRdma -name "vEthernet (vpl)"
```

```
PS C:\Users\Administrator>
```

```
PS C:\Users\Administrator> add - vmNet workAdapter -switchName vswitch -name vpl  
-managementOS
```

```
PS C:\Users\Administrator> enable-netAdapterRdma -name "vEthernet (vpl)"
```

```
PS C:\Users\Administrator>
```

- a) 次の Powershell コマンドを使用して、設定スイッチを構成します。

```
new-vmSwitch -name setswitch -netAdapterName "Ethernet x" -enableEmbeddedTeam $true
```

これにより、スイッチが作成されます。インターフェイスを表示するには、次を使用します。

```
get-netadapterrdma
```

```
add-vmNetworkAdapter -switchname setswitch -name svpl
```

再度入力すると、新しい vport が表示されます。

```
get-netadapterrdma
```

- b) vport を追加します：

```
add-vmNetworkAdapter -switchname setswitch -name svpl
```

再度入力すると、新しい vport が表示されます。

```
get-netadapterrdma
```

- c) vport で RDMA を有効にします。

```
enable-netAdapterRdma -name "vEthernet (svpl)"
```

ステップ 4 両方のサーバの RDMA 対応 vport で IPv4 アドレスを構成します。

ステップ 5 smb サーバで共有を作成し、smb クライアントで共有をマッピングします。

- a) ホストシステムの smb クライアントおよび smb サーバの場合は、前述のように RoCEv2 対応 vNIC を設定します。
- b) 両方のサーバに同じ IP サブネットと同じ固有の VLAN を使用して、両方のサーバでプライマリ ファブリックとサブ vNICs の IPv4 アドレスを設定します。
- c) smb サーバで共有を作成し、smb クライアントで共有をマッピングします。

ステップ 6 最後に、モード 2 の設定を確認します。

- a) Powershell コマンド **netstat-xan** を使用して、リスナーとそれらに関連付けられている IP アドレスを表示します。

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> netstat -xan
Active NetworkDirect Connections, Listeners, SharedEndpoints
Mode IfIndex Type Local Address Foreign Address PID
Kernel 9 Listener 50.37.61.23:445 NA 0
Kernel 26 Listener 10.37.60.158:445 NA 0
PS C:\Users\Administrator>
```

- b) smb クライアントのファイル共有で RDMA I/O を開始します。
- c) **Netstat-xan** コマンドを再度発行し、接続エントリが表示されていることを確認します。

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> netstat -xan
Active NetworkDirect Connections, Listeners, SharedEndpoints
Mode IfIndex Type Local Address Foreign Address PID
Kernel 9 Connection 50.37.61.23:192 50.37.61.184:445 0
Kernel 9 Connection 50.37.61.23:448 50.37.61.184:445 0
Kernel 9 Connection 50.37.61.23:704 50.37.61.214:445 0
Kernel 9 Connection 50.37.61.23:960 50.37.61.214:445 0
Kernel 9 Connection 50.37.61.23:1216 50.37.61.224:44 05
Kernel 9 Connection 50.37.61.23:1472 50.37.61.224:445 0
Kernel 9 Connection 50.37.61.23:1728 50.37.61.234:445 0
Kernel 9 Connection 50.37.61.23:1984 50.37.61.234:445 0
Kernel 9 Listener 50.37.61.23:445 NA
Kernel 26 Listener 10.37.60.158:445 NA
PS C:\Users\Administrator>
```

Linux で RoCEv2 を構成

Cisco UCS Manager での RoCEv2 を使用した NVMeoF の構成

Cisco UCS Manager で RoCEv2 インターフェイスを構成するには、次の手順を実行します。

手順

ステップ 1 [Navigation] ペインで [Servers] をクリックします。

ステップ 2 [Servers] > [Service Profiles] の順に展開します。

ステップ 3 ポリシーを作成する組織のノードを展開します。

システムにマルチテナント機能が備えられていない場合は、[ルート (root)] ノードを展開します。

ステップ 4 vNICs をクリックして、作業領域の **[ネットワーク (Network)]** タブに移動します。

下の手順に従い vNIC ポリシーを変更します。

- a) **[ネットワーク (Network)]** タブで、希望の vNIC までスクロールして、**[変更 (Modify)]** をクリックします。
- b) ポップアップダイアログボックスが表示されます。**[アダプタ パフォーマンス プロファイル (Adapter Performance Profile)]** 領域までスクロールし、**[アダプタ ポリシー (Adapter Policy)]** ドロップダウンをクリックします。ドロップダウン リストから **[Linux-NVMe-RoCE]** を選択します。
- c) **[OK]** をクリックします。

ステップ 5 **[Save Changes]** をクリックします。

次のタスク

[SRIOV BIOS ポリシーの有効化 \(9 ページ\)](#)

SRIOV BIOS ポリシーの有効化

Linux カーネルで IOMMU を有効にする前に、次の手順を実行して、SRIOV BIOS ポリシーでサーバのサービス プロファイルを設定します。

手順

ステップ 1 [Navigation] ペインで **[Servers]** をクリックします。

ステップ 2 **[サーバ (Servers)] > [サービス プロファイル (Service Profiles)]** を展開します。

ステップ 3 ポリシーを作成する組織のノードを展開します。

システムにマルチテナント機能が備えられていない場合は、[ルート (root)] ノードを展開します。

ステップ 4 BIOS ポリシーを有効にするサービス プロファイル ノードを選択します。

ステップ 5 [作業 (Work)] ペインで、**[ポリシー (Policy)]** を選択します。

ステップ 6 [ポリシー (Policies)] 領域で、**[BIOS ポリシー (BIOS Policy)]** を展開します。

ステップ 7 **[BIOS ポリシー (BIOS Policy)]** ドロップダウン リストから、デフォルトの SRIOV ポリシーを選択します。

ステップ 8 **[Save Changes]** をクリックします。

ホストで RoCEv2 を使用しての NVMeoF の構成

始める前に

RoCEv2 vNIC および SRIOV 対応 BIOS ポリシーを使用してサーバを設定します。

手順

ステップ 1 編集のために /etc/default/grub ファイルを開きます。

ステップ 2 次のサンプル ファイルに示すように、GRUB_CMDLINE_LINUX の行の末尾に intel_iommu = on を追加します。

```
sample /etc/default/grub configuration file after adding intel_iommu=on:
# cat /etc/default/grub
GRUB_TIMEOUT=5
GRUB_DISTRIBUTOR="$(sed 's, release .*$,g' /etc/system-release)"
GRUB_DEFAULT=saved
GRUB_DISABLE_SUBMENU=true
GRUB_TERMINAL_OUTPUT="console"
GRUB_CMDLINE_LINUX="crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap biosdevname=1
rhgb quiet intel_iommu=on
GRUB_DISABLE_RECOVERY="true"
```

ステップ 3 ファイルを保存します。

ステップ 4 ファイルを保存した後、次のコマンドを実行して新しい grub.cfg ファイルを生成します：

- レガシー ブートの場合：

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

- UEFI ブートの場合：

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```

ステップ 5 サーバをリブートします。IOMMU を有効にした後で、変更を反映するためにサーバを再起動します。

ステップ 6 出力ファイルをチェックして、サーバが intel_iommu = on オプションで起動していることを確認します。

```
cat /proc/cmdline | grep iommu
```

出力の最後に含まれることに注意してください。

```
[root@localhost basic-setup]# cat /proc/cmdline | grep iommu
BOOT_IMAGE=vmlinux-3.10.0-957.27.2.el7.x86_64 root=/dev/mapper/rhel-
root ro crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap rhgb
quiet intel_iommu=on LANG=en_US.UTF-8
```

次のタスク

enic および enic_rdma ドライバをダウンロードします。

Cisco enic および enic_rdma ドライバのインストール

enic_rdma ドライバには enic ドライバが必要です。enic および enic_rdma ドライバをインストールする場合は、Cisco.com で一致する enic および enic_rdma ドライバのセットをダウンロードして使用してください。inbox enic ドライバを使用して Cisco.com からダウンロードしたバイナリ enic_rdma ドライバを使用しようとしても、機能しません。

手順

ステップ 1 enic および enic_rdma rpm パッケージをインストールします。

```
# rpm -ivh kmod-enic-<version>.x86_64.rpm kmod-enic rdma-<version>.x86_64.rpm
```



(注) enic_rdma のインストール中に、enic_rdmalibnvdimm モジュールは、RHEL 7.7 へのインストールに失敗することがあります。nvdimm-security.conf dracut モジュールは add_drivers 値にスペースを必要とするためです。回避策については、次のリンクの指示に従ってください。

<https://access.redhat.com/solutions/4386041>

https://bugzilla.redhat.com/show_bug.cgi?id=1740383

ステップ 2 enic_rdma ドライバはインストールされていますが、動作中のカーネルでロードされません。サーバを再起動して、実行中のカーネルに enic_rdma ドライバをロードします。

ステップ 3 enic_rdma ドライバと RoCE v2 インターフェイスのインストールを確認します。

```
# dmesg | grep enic_rdma
[ 4.025979] enic_rdma: Cisco VIC Ethernet NIC RDMA Driver, ver 1.0.0.6-802.21 init
[ 4.052792] enic 0000:62:00.1 eth1: enic_rdma: IPv4 RoCEv2 enabled
[ 4.081032] enic 0000:62:00.2 eth2: enic_rdma: IPv4 RoCEv2 enabled
```

ステップ 4 nvme-rdma カーネル モジュールをロードします。

```
# modprobe nvme-rdma
```

サーバの再起動後に、nvme-rdma カーネルモジュールがアンロードされます。サーバの再起動ごとに nvme-rdma カーネルモジュールをロードするには、次を使用して nvme_rdma.conf ファイルを作成します。

```
# echo nvme_rdma > /etc/modules-load.d/nvme_rdma.conf
```



(注) インストール後の enic_rdma の詳細については、rpm -q -l kmod-enic_rdma コマンドを使用して README ファイルを抽出します。

次のタスク

ターゲットを検出し、NVMe ネームスペースに接続します。システムでストレージへのマルチパスアクセスが必要な場合は、[デバイス マAPPER マルチパスの設定 \(13 ページ\)](#) についてのセクションを参照してください。

NVMe ターゲットの検出

NVMe のターゲットを検出し、NVMe ネームスペースを接続するには、次の手順を使用します。

始める前に

まだインストールされていない場合は、**nvme cli** バージョン 1.6 以降をインストールします。



(注) **nvme-cli** バージョン 1.7 以降がインストールされている場合は、下のステップ 2 はスキップします。

RoCEv2 インターフェイスで IP アドレスを設定し、インターフェイスがターゲット IP に対して ping を実行できることを確認します。

手順

ステップ 1 /etc で nvme フォルダを作成し、ホスト nqn を手動で生成します。

```
# mkdir /etc/nvme
# nvme gen-hostnqn > /etc/nvme/hostnqn
```

ステップ 2 settos.sh ファイルを作成し、IB フレームでプライオリティフロー制御 (PFC) を設定するスクリプトを実行します。



(注) NVMeoF トラフィックの送信に失敗しないようにするには、サーバを再起動することによりこのスクリプトを作成して実行する必要があります。

```
# cat settos.sh
#!/bin/bash
for f in `ls /sys/class/infiniband`;
do
    echo "setting TOS for IB interface:" $f
    mkdir -p /sys/kernel/config/rdma_cm/$f/ports/1
    echo 186 > /sys/kernel/config/rdma_cm/$f/ports/1/default_roce_tos
done
```

ステップ 3 次のコマンドを入力して、NVMe ターゲットを検出します。

```
nvme discover --transport=rdma --traddr=<IP address of transport target port>
```

例えば、50.2.85.200 でターゲットを検出するには、次のようにします。

```
# nvme discover --transport=rdma --traddr=50.2.85.200

Discovery Log Number of Records 1, Generation counter 2
=====Discovery Log Entry 0=====
trtype: rdma
adrfam: ipv4
subtype: nvme subsystem
trreq: not required
portid: 3
trsvcid: 4420
subnqn: nqn.2010-06.com.purestorage:flasharray.9a703295ee2954e
traddr: 50.2.85.200
rdma_prtype: roce-v2
rdma_qptype: connected
rdma_cms: rdma-cm
rdma_pkey: 0x0000
```



(注) IPv6 を使用して NVMe ターゲットを検出するには、traddr オプションの次に IPv6 ターゲットアドレスを指定します。

ステップ 4 次のコマンドを入力して、検出された NVMe ターゲットに接続します。

```
nvme connect --transport=rdma --traddr=<IP address of transport target port>> -n <subnqn
value from nvme discover>
```

例えば、50.2.85.200 のターゲットと上記の subnqn 値を検出するには、次の手順を実行します。

```
# nvme connect --transport=rdma --traddr=50.2.85.200 -n
nqn.2010-06.com.purestorage:flasharray.
9a703295ee2954e
```



(注) IPv6 を使用して検出した NVMe ターゲットに接続するには、traddr オプションの次に IPv6 ターゲットアドレスを指定します。

ステップ 5 nvme list コマンドを使用して、マッピングされたネームスペースを確認します。

```
# nvme list
Node          SN              Model              Namespace Usage      Format
FW Rev
-----
-----
/dev/nvme0n1 09A703295EE2954E Pure Storage FlashArray 72656 4.29 GB/4.29 GB 512 B + 0 B
99.9.9
/dev/nvme0n2 09A703295EE2954E Pure Storage FlashArray 72657 5.37 GB/5.37 GB 512 B + 0 B
99.9.9
```

デバイス マッパー マルチパスの設定

システムがデバイス マッパー マルチパス (DM マルチパス) を使用して構成されている場合は、次の手順に従ってデバイス マッパー マルチパスをセットアップします。



- (注) Red Hat Enterprise Linux (RHEL) での NVMe-over-RDMA 構成の場合は、デバイスマッパー (DM) マルチパスの代わりに NVMe ネイティブ マルチパスを使用します。
- **RHEL 8.x** : NVMe ネイティブ マルチパスは DM マルチパスの代替として使用できます。
 - **RHEL 9.x** : NVMe ネイティブ マルチパスは NVMe-over-RDMA の推奨構成です。
 - **RHEL 10.x 以降** : NVMe ネイティブ マルチパスは Red Hat ディストリビューションのデフォルト設定です。
 - ネイティブ マルチパスは、パフォーマンスを最適化するために、デフォルトで NUMA 対応ポリシーを使用します。管理者は、特定のアプリケーションワークロードで必要な場合に、マルチパス ポリシーをデフォルトの NUMA 対応設定からラウンドロビンポリシーに再設定できます。

手順

ステップ 1 まだインストールされていない場合は、device-mapper-multipath パッケージをインストールします。

ステップ 2 Multipathd を有効にして開始します。

```
# mpathconf --enable --with_multipathd y
```

ステップ 3 etc/multipath.conf ファイルを編集して、次の値を使用します。

```
defaults {
    polling_interval    10
    path_selector       "queue-length 0"
    path_grouping_policy multibus
    fast_io_fail_tmo    10
    no_path_retry       0
    features            0
    dev_loss_tmo        60
    user_friendly_names yes
}
```

ステップ 4 更新されたマルチパス デバイス マップを使用してフラッシュします。

```
# multipath -F
```

ステップ 5 マルチパス サービスを再起動します。

```
# systemctl restart multipathd.service
```

ステップ 6 マルチパス デバイスを再スキャンします。

```
# multipath -v2
```

ステップ 7 マルチパス ステータスを確認します。

```
# multipath -ll
```

Cisco UCS Manager を使用した RoCEv2 インターフェイスの削除

RoCE v2 インターフェイスを削除するには、次の手順を実行します。

手順

ステップ 1 [ナビゲーション] ペインで、[サーバ] をクリックします。

ステップ 2 [サーバ (Servers)] > [サービス プロファイル (Service Profiles)] を展開します。

ステップ 3 ポリシーを作成する組織のノードを展開します。

システムにマルチテナント機能が備えられていない場合は、[ルート (root)] ノードを展開します。

ステップ 4 下の手順に従い vNIC ポリシーを変更します。

- [ネットワーク (Network)]** タブで、希望の vNIC までスクロールして、**[変更 (Modify)]** をクリックします。
- ポップアップ ダイアログ ボックスが表示されます。**[ポリシー (Policies)]** エリアまで下にスクロールして、**[アダプタ ポリシー (Adapter Policy)]** ドロップダウン リストから **[Linux]** を選択します。
- [OK]** をクリックします。

ステップ 5 **[Save Changes]** をクリックします。

ESXi での RoCEv2 の構成

UCS Manager での ESXi の RoCEv2 を使用した NVMeoF の構成

UCS Manager には、動作パラメータが事前に入力されたデフォルトのアダプタポリシーが含まれているため、アダプタポリシーを手動で作成する必要はありません。ただし、RoCEv2 インターフェイスを作成する必要があります。

UCS Manager で RoCEv2 インターフェイスを設定するには、次の手順を実行します。

手順

ステップ 1 [Navigation] ペインで [Servers] をクリックします。

ステップ 2 [Servers] > [Service Profiles] の順に展開します。

ステップ 3 ポリシーを作成する組織のノードを展開します。

システムにマルチテナント機能が備えられていない場合は、[ルート (root)] ノードを展開します。

ステップ 4 作成した RDMA サービス プロファイルをクリックし、サービス プロファイルを展開します。

ステップ 5 [vNIC] を右クリックし、[vNIC の作成 (Create vNIC)] を選択して新しい vNIC を作成します。

[vNIC の作成 (Create vNIC)] ポップアップ メニューが表示されます。

vNIC ポリシーを変更するには、次の手順を実行します。

- 新しい VNIC に名前を付けます。
- [MAC アドレス (MAC address)] ドロップダウンで、ドロップダウンの OUI またはドメインプールを使用した手動からオプションを選択します。
- 使用する VLAN をドロップダウン リストから選択します。
- アダプタ パフォーマンス プロファイルで、[VMWareNVMeRoCEv2] という名前のデフォルトのアダプタ ポリシーを選択します。
- [OK] をクリックします。これで、インターフェイスは 1 つのポート用に設定されました。

ステップ 6 [Save Changes] をクリックします。

次のタスク

NENIC ドライバをインストールします。

NENIC ドライバーをインストール

RDMA ドライバを含む eNIC ドライバは、複合パッケージとして使用できます。cisco.com で eNIC ドライバをダウンロードし使用します。

これらの手順は、これが新規にインストールされていることを前提としています。



(注) この例では /tmp の 場所を使用していますが、ESX コンソール シェルにアクセス可能な任意の場所にファイルを配置できます。

手順

ステップ 1 eNIC VIB またはオフライン バンドルを ESX サーバにコピーします。次の例では、Linux scp ユーティリティを使用して、ローカル システムから 10.10.10.10 にある ESX サーバにファイルをコピーし、ロケーション /tmp を使用します。

```
scp nenic-2.0.4.0-10EM.700.1.0.15843807.x86_64.vib root@
10.10.10.10:/tmp
```

ステップ 2 フルパスを指定して、次のコマンドを発行します。

```
esxcli software vib install -v {VIBFILE}
```

または

```
esxcli software vib install -d {OFFLINE_BUNDLE}
```

例：

```
esxcli software vib install -v /tmp/nenic-2.0.4.0-10EM.
700.1.0.15843807.x86_64.vib
```



(注) VIB の署名に使用されている証明書によっては、ホスト許容レベルの変更が必要になる場合があります。これを行うには、次のコマンドを使用します：`esxcli software accept set --level=<level>`

インストールされる VIB のタイプによっては、ESX をメンテナンス モードにする必要があります。これは、VI クライアントを介して実行するか、上記の `esxcli` コマンドに `--maintenance-mode` オプションを追加することで実行できます。

NENIC ドライバのアップグレード

1. NENIC ドライバをアップグレードするには、コマンドを入力します：

```
esxcli software vib update -v {VIBFILE}
```

または

```
esxcli software vib update -d {OFFLINE_BUNDLE}
```

2. 上記の手順 1 を使用して、`enic` VIB またはオフラインバンドルを ESX サーバにコピーします。

次のタスク

ESXi ホスト側 NVMe RDMA を構成します。

ESXi NVMe RDMA のホスト側の構成

NENIC RDMA の機能

Linux と ESXi での RDMA の使用例の違い：

- ESXi では、物理インターフェイス (vmnic) の MAC は RoCEv2 トラフィックに使用されません。代わりに、VMkernelポート (vmk) の MAC が使用されます。

発信 RoCE パケットはイーサネット送信元 MAC フィールドの vmrk MAC を使用します。着信 RoCE パケットは、イーサネット接続先 mac フィールドの vmk MAC を使用します。vmk MAC アドレスは、作成時に vmk インターフェイスに割り当てられる VMware MAC アドレスです。

- Linux では、物理インターフェイス MAC が RoCE パケットの送信元 MAC アドレスフィールドで使用されます。この Linux MAC は通常、Cisco UCS Manager を使用して VNIC に構成された Cisco MAC アドレスです。

ホストに ssh で接続し、**esxcli network ip interface list** コマンドを使用すると、MAC アドレスを確認できます。

```
vmko
Name: vmko
MAC Address: 2c:f8:9b:a1:4c:e7
Enabled: true
Portset: vSwitch0
Portgroup: Management Network
Netstack Instance: defaultTcpipStack
VDS Name: N/A
VDS UUID: N/A
VDS Port: N/A
VDS Connection: -1
Opaque Network ID: N/A
Opaque Network Type: N/A
External ID: N/A
MTU: 1500
TSO MSS: 65535
RXDispQueue Size: 2
Port ID: 67108881
```

ホスト、仮想マシン、および VMkernel トラフィックにネットワーク接続を提供するには、vSphere 標準スイッチを作成する必要があります。作成する接続タイプに応じて、VMkernel アダプタを使用して新しい vSphere 標準スイッチを作成するか、物理ネットワークアダプタのみを新しいスイッチに接続するか、または仮想マシンポートグループを使用してスイッチを作成することができます。

ネットワーク接続スイッチの作成

次の手順に従って、ホスト、仮想マシン、および VMkernel トラフィックへネットワーク接続を提供するための vSphere 標準スイッチを作成します。

始める前に

NENIC ドライバをダウンロードおよびインストールしていることを確認します。

手順

ステップ 1 vSphere Web Client で、ホストに移動します。

ステップ 2 [構成 (Configure)] タブで、[ネットワークキング (Networking)] を展開し、[仮想スイッチ (Virtual Switches)] を選択します。

ステップ 3 [ネットワークの追加 (Add Networking)] をクリックします。

使用可能なネットワーク アダプタの接続タイプは次のとおりです。

- **Vmkernel ネットワーク アダプタ**

ホスト管理トラフィックを処理する新しい Vmkernel アダプタを作成します。

- **物理ネットワーク アダプタ**

物理ネットワーク アダプタを新しい、または既存の標準スイッチに追加します。

- **標準スイッチの仮想マシン ポート グループ**

仮想マシン ネットワーキング用の新しいポート グループを作成します。

ステップ 4 接続タイプ [Vmkernel ネットワーク アダプタ (Vmkernel Network Adapter)] を選択します。

ステップ 5 [新しい標準スイッチ (New standard switch)] を選択し、[次へ (Next)] をクリックします。

ステップ 6 物理ネットワーク アダプタを新しい標準スイッチに追加します。

- [割り当て済みアダプタ (Assigned Adapters)] で、[新しいアダプタ (New Adapters)] を選択します。
- リストから 1 つ以上のアダプタをセレクトし、[OK] をクリックします。スループットの向上を促し、冗長性を得るため、[アクティブ (Active)] リストで少なくとも 2 つの物理ネットワーク アダプタを追加してください。
- (オプション) 上矢印キーと下矢印キーで、[割り当て済みアダプタ (Assigned Adapters)] リスト内のアダプタの位置を変更します。
- [次へ (Next)] をクリックします。

ステップ 7 VMadapter またはポートグループ用に作成した新しい標準スイッチに対し、アダプタまたはポートグループの接続設定を入力します。

- VMkernel アダプタのトラフィック タイプを表すラベルを入力します。
- ネットワーク トラフィックのルーティングで Vmkernel が使用する VLAN を識別するための、VLAN ID を設定します。
- [IPv4]、[IPv6]、またはその両方を選択します。
- ドロップダウン メニューから MTU サイズを選択します。特定の MTU サイズを入力する場合は、[カスタム (Custom)] を選択します。最大 MTU サイズは 9000 バイトです。



(注) 1500 より大きい MTU を設定すれば、ジャンボ フレームを有効にすることができます。

- VMkernel アダプタの TCP/IP スタックを設定した後、TCP/IP スタックを選択します。デフォルトの TCP/IP スタックを使用するには、使用可能なサービスから選択します。



(注) Vmkernel アダプタの TCP/IP スタックは、後から変更できないことに注意してください。

f) IPV4 または IPV6 設定、あるいはその両方を構成します。

ステップ 8 [完了の準備] ページで、**[完了 (Finish)]** をクリックします。

ステップ 9 次の結果に示すように、vSphere クライアントで NVMe RDMA を使用して VM アダプタまたはポート グループの VMkernel ポートを確認します。

次のタスク

vmrdma ポートの上に vmhba ポートを作成します。

ESXi での VMHBA ポートの作成

vmrdma アダプタ ポートの上に vmhba ポートを作成するには、次の手順に従います。

始める前に

ストレージ接続用のアダプタ ポートを作成します。

手順

ステップ 1 ESXi ホストが接続されている vCenter に移動します。

ステップ 2 **[ホスト (Host)] > [構成 (Configure)] > [ストレージアダプタ (Storage adapters)]** の順にクリックします。

ステップ 3 **[+ ソフトウェア アダプタの追加 (Add Software Adapter)]** をクリックします。

[ソフトウェアアダプタの追加 (Add Software Adapter)] ダイアログ ボックスが表示されます。

ステップ 4 **[RDMA アダプタ上にソフトウェア NVMe を追加 (Add software NVMe over RDMA adapter)]** と、使用する vmrdma ポートを選択します。

ステップ 5 **[OK]** をクリックします。

RDMA ストレージアダプタ上の VMware NVMe の vmhba ポートは、表示されます。

次のタスク

NVMe を構成します。

vmnic および vmrdma インターフェイスの表示

ESXi は、ホストに構成された各 enic VNIC に対して vmnic インターフェイスを作成します。

始める前に

ネットワーク アダプタと VHBA ポートを作成します。

手順

ステップ 1 `ssh` を使用してホスト システムにアクセスします。

ステップ 2 `esxcfg-nics -l` と入力して、ESXi 上の `vmnic` を一覧表示します。

Name	PCI	Driver	Link	Speed	Duplex	MAC Address	MTU	Description
vmnic0	0000:3b:00.0	ixgben	Down	0Mbps	Half	2c:f8:9b:a1:4c:e6	1500	Intel(R) Ethernet Controller X550
vmnic1	0000:36:00.1	ixgben	Up	1000Mbps	Full	2c:f8:9b:a1:4c:e7	1500	Intel(R) Ethernet Controller X550
vmnic2	0000:1d:00.0	nenic	Up	50000Mbps	Full	2c:f8:9b:79:8d:bc	1500	Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic3	0000:1d:00.1	nenic	Up	50000Mbps	Full	2c:f8:9b:79:8d:bd	1500	Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic4	0000:63:00.0	nenic	Down	0Mbps	Half	2c:f8:9b:51:b3:3a	1500	Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic5	0000:63:00.1	nenic	Down	0Mbps	Half	2c:f8:9b:51:b3:3b	1500	Cisco Systems Inc Cisco VIC Ethernet NIC

esxcli network nic list

Name	PCI	Driver	Admin	Status	Link	Status	Speed	Duplex	MAC Address	MTU
vmnic0	0000:3b:00.0	ixgben	Up		Down		0	Half	2c:f8:9b:a1:4c:e6	1500
vmnic1	0000:36:00.1	ixgben	Up		Up		1000	Full	2c:f8:9b:a1:4c:e7	1500
vmnic2	0000:1d:00.0	nenic	Up		Up		50000	Full	2c:f8:9b:79:8d:bc	1500
vmnic3	0000:1d:00.1	nenic	Up		Up		50000	Full	2c:f8:9b:79:8d:bd	1500
vmnic4	0000:63:00.0	nenic	Up		Down		0	Half	2c:f8:9b:51:b3:3a	1500
vmnic5	0000:63:00.1	nenic	Up		Down		0	Half	2c:f8:9b:51:b3:3b	1500

`enic` ドライバが RDMA 対応 VNIC の RDMA デバイスを ESXi に登録すると、ESXi は `vmrdma` デバイスを作成し、対応する `vmnic` にリンクします。

ステップ 3 `esxcli rdma device list` を使用して、`vmrdma` デバイスを一覧表示します。

```
[root@RackServer:~] esxcli rdma device list
Name      Driver State  MTU   Speed  Paired Uplink Description
-----
vmrdma0   nenic  Active  4096  50 Gbps vmnic1  Cisco UCS VIC 15XXX (A0)
vmrdma1   nenic  Active  4096  50 Gbps vmnic2  Cisco UCS VIC 15XXX (A0)
[root@StockholmRackServer:~] esxcli rdma device vmknics list
Device    Vmknics NetStack
-----
vmrdma0   vmk1    defaultTcpipStack
vmrdma1   vmk2    defaultTcpipStack
```

ステップ 4 `esxcli rdma device list` を使用して、vmrdma インターフェイスでサポートされているプロトコルを確認します。

enic の場合、RoCE v2 がサポートされている唯一のプロトコルであることがリストから分かります。このコマンドの出力は、VNIC の RoCEv2 設定と一致しているはずです。

ステップ 5 `esxcli rdma device protocol list` を使用して、vmrdma インターフェイスでサポートされているプロトコルを確認します。

enic の場合、RoCE v2 がサポートされている唯一のプロトコルであることがリストから分かります。このコマンドの出力は、VNIC の RoCEv2 設定と一致しているはずです。

```
[root@RackServer:~] esxcli rdma protocol list
Device  RoCE v1 RoCE v2 iWARP
-----
vmrdma0 false  true  false
vmrdma1 false  true  false
```

ステップ 6 `esxcli nvme adapter list` を使用して、NVMe アダプタと、それが構成されている vmrdma および vmnic インターフェイスを一覧表示します。

```
[root@RackServer:~] esxcli nvme adapter list
Adapter Adapter Qualified Name Transport Type Driver Associated Devices
-----
vmhba64 aqn: nvmerdma:2c-f8-9b-79-8d-bc RDMA nvmerdma vmrdmaR, vmnic2
vmhba65 aqn: nvmerdma:2c-f8-9b-79-8d-bd RDMA nvmerdma vmrdma1, vmnic3
```

ステップ 7 `esxcli storage core adapter list` を使用して、システム内のすべての vmhbas を一覧表示できます。

```
[root@RackServer:~] esxcli storage core adapter list
HBA Name Driver Link State UID Capabilities
Description
-----
vmhba0 nfnic link-down fc.10002cf89b798dbe:20002cf89b798dbe Second Level Lun ID
(0000:1d:00.2) Cisco Corporation Cisco

UCS VIC Fnic Controller
vmhba1 vmw_ahci link-n/a sata.vmhba1
(0000:00:11.5) Intel Corporation Lewisburg

SATA AHCI Controller
vmhba2 nfnic link-down fc.10002cf89b798dbf:20002cf89b798dbf Second Level Lun ID
(0000:1d:00.3) Cisco Corporation Cisco

UCS VIC Fnic Controller
vmhba3 nfnic link-down fc.10002cf89b51b33c:20002cf89b51b33c Second Level Lun ID
(0000:63:00.2) Cisco Corporation Cisco

UCS VIC Fnic Controller
vmhba4 nfnic link-down fc.10002cf89b51b33d:20002cf89b51b33d Second Level Lun ID
(0000:63:00.3) Cisco Corporation Cisco

UCS VIC Fnic Controller
vmhba5 lsi_mr3 link-n/a sas.5cc167e9732f9b00
(0000:3c:00.0) Broadcom Cisco 126 Modular

Raid Controller with 2GB cache
vmhba64 nvmerdma link-n/a rdma.vmnic2:2c: f8:9b:79:8d:bc
```

```
VMware NVMe over RDMA Storage Adapter on vmrdma0
vmhba65 nvmerdma link-n/a rdma.vmnic3:2c:f8:9b:79:8d:bd
VMware NVMe over RDMA Storage Adapter on vmrdma1
```

次のタスク

NVMe を構成します。

NVMe ファブリックと名前空間の検出

この手順は、ESXi コマンドライン インターフェイスを使用して実行します。

始める前に

アダプタの VMHBA で NVMe を作成して構成します。アダプタの最大数は 2 で、ベスト プラクティスは両方をフォールトトレランス用に構成することです。

手順

ステップ 1 vmrdma デバイスの NVMe をチェックして有効にします。

```
esxcli nvme fabrics enable -p RDMA -d vmrdma0
```

NVMe が有効になっているかどうかを示すメッセージが表示されます。

ステップ 2 次のコマンドを入力して、アレイ上の NVMe ファブリックを検出します。

```
esxcli nvme fabrics discover -a vmhba64 -l transport_address
```

esxcli nvme fabrics discover -a vmhba64 -l 50.2.84.100 の実行結果を示す図

出力は、トランスポート タイプ、アドレス ファミリー、サブシステム タイプ、コントローラ ID、管理キュー、最大サイズ、トランスポート アドレス、トランスポート サービス ID、およびサブシステム NQN の情報を表示します。

NVMe コントローラに出力が表示されます。

ステップ 3 NVMe ファブリック インターコネクトを実行します。

```
esxcli nvme fabrics discover -a vmhba64 -l transport_address p Transport
Service ID -s Subsystem NQN
```

ステップ 4 手順 1 ～ 4 を繰り返して、2 番目のアダプタを構成します。

ステップ 5 コントローラ リストを表示して、NVMe コントローラが存在し、動作していることを確認します。

```
esxcli nvme controller list RDMA -d vmrdma0
```

```
[root@RackServer:~] esxcli nvme controller list
Name                                     Controller Number Adapter Transport Type Is
```

```
Online
-----
nqn.2010-06.com.purestorage: flasharray. 258 vmhba64 RDMA true
5ab274df5b161455#vmhba64#50.2.84.100:4420
nqn.2010-06.com.purestorage: flasharray. 259 vmhba65 RDMA true
Sab274df5b161455#vmhba65#50.2.83.100:4420
[root@RackServer:~] esxcli nvme namespace list
Name Controller Number Namespace ID Block Size Capacity
in MB
-----
eui.00e6d65b65a8f34824a9374e00011745 258 71493 512 102400
eui.00e6d65b65a8f34024a9374e00011745 259 71493 512 102400
```

例

次の例は、サーバ上で実行された `esxcli discovery` コマンドを示しています。

```
[root@RackServer:~] esxcli nvme fabrics enable -p RDMA -d vmrdma0 NVMe already
enabled on vmrdma0
[root@RackServer:~] esxcli nvme fabrics discover -a vmhba64 -l 50.2.84.100
Transport Address Subsystem Controller Admin Queue Transport Transport Subsystem NQN
Type Family Type ID Max Size Address Service ID
-----
RDMA IPV4 NVM 65535 31 50.2.84.100 4420 nq.210-06.com.
purestorage:
flasharray:2dp1239anjkl484
[root@RackServer:~] esxcli nvme fabrics discover -a vmhba64 -l 50.2.84.100 p 4420 -s
nq.210-06.com.
purestorage:flasharray:2dp1239anjkl484 Controller already connected
```

UCS Manager CLI を使用した RoCEv2 インターフェイスの構成または削除

UCS Manager CLI を使用した Windows SMB Direct RoCEv2 インターフェイスの設定

Cisco UCS Manager CLI で RoCEv2 インターフェイスを設定するには、次の手順を使用します。

始める前に

管理権限を持つユーザーとしてログインする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	例 : UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id	指定されたシャーシ、ブレード、または UCS 管理ラック サーバ ID のサービス プロファイルを入力します。
ステップ 2	例 : UCS-A /org/service-profile # show vnic	サーバ上で使用可能な vNIC が表示されます。
ステップ 3	例 : UCS-A /org/service-profile # scope vnic vnic name	指定された vNIC の vNIC モードを開始します。
ステップ 4	Windows SMBDirect RoCEv2 モード1 を設定するには、次の手順を実行します。 例 : UCS-A /org/service-profile/vnic # set adapter-policy Win-HPN-SMBd	RoCEv2 モード 1 の Windows SMBDirect RoCEv2 アダプタ ポリシーを指定します。
ステップ 5	Windows SMBDirect RoCEv2 モード2 を設定するには、次の手順を実行します。 例 : UCS-A# scope org UCS-A /org # create vmq-conn-policy policy name UCS-A /org/vmq-conn-policy* # set multi-queue enabled UCS-A /org/vmq-conn-policy* # set vmmq-sub-vnic-count 64 UCS-A /org/vmq-conn-policy* # set vmmq-adaptor-profile-name MQ-SMBd UCS-A /org/vmq-conn-policy* # commit-buffer UCS-A /org/vmq-conn-policy #	VMQ 接続ポリシーを作成し、アダプタ ポリシー MQ-SMBd を割り当てた後、Windowsモード 2 を設定します。
ステップ 6	例 : UCS-A /org/service-profile/vnic* # commit-buffer	トランザクションをシステム設定にコミットします。

次の例は、RoCEv2 Win-HPN-SMBd アダプタ ポリシーの設定方法を示しています。

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic
```

vNIC:

```
Name      Fabric ID      Dynamic MAC Addr  Virtualization Preference
-----
eth00      A B            00:25:B5:3A:84:00  NONE
```

```
eth01      A      00:25:B5:3A:84:01      NONE
eth02      B      00:25:B5:3A:84:02      NONE
```

```
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy Win-HPN-SMBd
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

UCS Manager の CLI を使用した Windows RoCEv2 インターフェイスの削除

Cisco UCS Manager CLI で Windows RoCEv2 インターフェイスを削除するには、次の手順に従います。

始める前に

管理権限を持つユーザーとしてログインする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	例 : UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id	指定されたシャーシ、ブレード、または UCS 管理ラック サーバ ID のサービス プロファイルを入力します。
ステップ 2	例 : UCS-A /org/service-profile # show vnic	サーバ上で使用可能な vNIC が表示されます。
ステップ 3	例 : UCS-A /org/service-profile # scope vnic vnic name	指定された vNIC の vNIC モードを開始します。
ステップ 4	例 : UCS-A /org/service-profile/vnic # set adapter-policy Windows	デフォルトの Windows アダプタ ポリシーを設定して、Windows RoCEv2 アダプタ ポリシーを削除します。
ステップ 5	例 : UCS-A /org/service-profile/vnic* # commit-buffer	トランザクションをシステム設定にコミットします。

次のタスク

次に、Windows の eth01 vNIC で RoCEv2 インターフェイスを削除する例を示します。

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic
```

vNIC:

Name	Fabric ID	Dynamic MAC Addr	Virtualization Preference
eth00	A B	00:25:B5:3A:84:00	NONE
eth01	A	00:25:B5:3A:84:01	NONE
eth02	B	00:25:B5:3A:84:02	NONE

```
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy Windows
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

UCS Manager CLI を使用した Linux の RoCEv2 インターフェイスの設定

Cisco UCS Manager CLI で Linux の RoCEv2 インターフェイスを設定するには、次の手順を使用します。

始める前に

管理権限を持つユーザーとしてログインする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	例 : UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id	指定されたシャーシ、ブレード、または UCS 管理ラック サーバ ID のサービス プロファイルを入力します。
ステップ 2	例 : UCS-A /org/service-profile # show vnic	サーバ上で使用可能な vNIC が表示されます。
ステップ 3	例 : UCS-A /org/service-profile # scope vnic vnic name	指定された vNIC の vNIC モードを開始します。
ステップ 4	例 : UCS-A /org/service-profile/vnic # set adapter-policy Linux-NVMe-RoCE	NVMeoF に使用する vNIC のアダプタ ポリシーとして Linux NVMe-RoCE を指定します。
ステップ 5	例 : UCS-A /org/service-profile/vnic* # commit-buffer	トランザクションをシステム設定にコミットします。

次に、eth01 vNIC で RoCEv2 Linux アダプタ ポリシーを設定する例を示します。

例

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic

vNIC:
-----
Name                Fabric ID Dynamic MAC Addr    Virtualization Preference
-----
eth00                A B          00:25:B5:3A:84:00    NONE
eth01                A            00:25:B5:3A:84:01    NONE
eth02                B            00:25:B5:3A:84:02    NONE
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy Linux-NVMe-RoCE
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

UCS Manager CLI を使用した Linux の RoCEv2 インターフェイスの削除

Cisco UCS Manager CLI で Linux の RoCEv2 インターフェイスを削除するには、次の手順を使用します。

始める前に

管理権限を持つユーザーとしてログインする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	例 : UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id	指定されたシャーシ、ブレード、または UCS 管理ラック サーバ ID のサービス プロファイルを入力します。
ステップ 2	例 : UCS-A /org/service-profile # show vnic	サーバ上で使用可能な vNIC が表示されます。
ステップ 3	例 : UCS-A /org/service-profile # scope vnic vnic name	指定された vNIC の vNIC モードを開始します。
ステップ 4	例 : UCS-A /org/service-profile/vnic # set adapter-policy Linux	デフォルトの Linux アダプタ ポリシーを設定して、Linux NVMe-RoCE ポリシーを削除します。
ステップ 5	例 : UCS-A /org/service-profile/vnic* # commit-buffer	トランザクションをシステム設定にコミットします。

次に、Linux の eth01 vNIC で RoCEv2 インターフェイスを削除する例を示します。

例

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic

vNIC:
-----
Name                Fabric ID Dynamic MAC Addr    Virtualization Preference
-----
eth00               A B        00:25:B5:3A:84:00  NONE
eth01               A          00:25:B5:3A:84:01  NONE
eth02               B          00:25:B5:3A:84:02  NONE
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy Linux
UCS-A /org/service-profile/vnic* # commit-buffer
```

UCS Manager CLI を使用した VMware ESXi RoCEv2 インターフェイスの設定

Cisco UCS Manager CLI で VMware ESXi の RoCEv2 インターフェイスを設定するには、次の手順に従います。

始める前に

管理権限を持つユーザーとしてログインする必要があります。

手順

	コマンドまたはアクション	目的
ステップ 1	例 : UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id	指定されたシャーシ、ブレード、または UCS 管理ラック サーバ ID のサービス プロファイルを入力します。
ステップ 2	例 : UCS-A /org/service-profile # show vnic	サーバ上で使用可能な vNIC が表示されます。
ステップ 3	例 : UCS-A /org/service-profile # scope vnic vnic name	指定された vNIC の vNIC モードを開始します。
ステップ 4	例 : UCS-A /org/service-profile/vnic # set adapter-policy VMWareNVMeRoCEv2	NVMeoF に使用する vNIC のアダプタ ポリシーとして VMware NVMe-RoCE を指定します。
ステップ 5	例 : UCS-A /org/service-profile/vnic* # commit-buffer	トランザクションをシステム設定にコミットします。

次に、eth01 vNIC で RoCEv2 VMware のアダプタ ポリシーを設定する例を示します。

例

```
UCS-A# scope service-profile server 1/1
UCS-A /org/service-profile # show vnic
```

vNIC:

Name	Fabric ID	Dynamic MAC Addr	Virtualization Preference
eth00	A B	00:25:B5:3A:84:00	NONE
eth01	A	00:25:B5:3A:84:01	NONE
eth02	B	00:25:B5:3A:84:02	NONE

```
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy VMWareNVMeRoCEv2
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

UCS Manager を使用した ESXi RoCEv2 インターフェイスの削除

特定のポートの RoCEv2 インターフェイスを削除するには、次の手順を実行します。

手順

- ステップ 1** [Navigation (ナビゲーション)] ペインで [Servers (サーバ)] をクリックします。
- ステップ 2** [サーバ (Servers)] > [サービス プロファイル (Service Profiles)] を展開します。
- ステップ 3** 削除するプロファイルのノードを展開します。
- ステップ 4** [vNIC] をクリックし、目的のインターフェイスを選択します。右クリックして、ドロップダウンから [削除 (Delete)] を選択します。
- ステップ 5** [Save Changes] をクリックします。

翻訳について

このドキュメントは、米国シスコ発行ドキュメントの参考和訳です。リンク情報につきましては、日本語版掲載時点で、英語版にアップデートがあり、リンク先のページが移動/変更されている場合がありますことをご了承ください。あくまでも参考和訳となりますので、正式な内容については米国サイトのドキュメントを参照ください。