

Configuring RDMA Over Converged Ethernet Using RoCEv2

- Configuring RoCEv2 in Windows , on page 1
- Configuring RoCEv2 in Linux, on page 7
- Configuring RoCEv2 in ESXi, on page 14
- Using the UCS Manager CLI to Configure or Delete the RoCEv2 Interface , on page 22

Configuring RoCEv2 in Windows

Configuring RoCEv2 Modes 1 and 2 in Windows

Configuration of Remote Direct Memory Access (RDMA) over Converged Ethernet version 2 (RoCEv2) on the Windows platform requires first configuring RoCEv2 Mode 1, then configuring RoCEv2 Mode 2. Modes 1 and 2 relate to the implementation of the Network Direct Kernel Provider Interface (NDKPI): Mode 1 is native RDMA, and Mode 2 involves configuration for the virtual port with RDMA.

To configure RoCEv2 mode 1, you must:

- Configure a no-drop class in the Class of Service (CoS) System Class. By default, Platinum with CoS 5 is a default in Cisco Unified Computing System (UCS) Manager.
- Configure an Ethernet adapter policy for Mode 1 in Cisco UCS Manager.
- Configure Mode 1 on the host system.

RoCEv2 Mode 1 must be configured before configuring Mode 2.

To configure RoCEv2 mode 2, you must:

Either create an Ethernet Virtual Machine Queue (VMQ) connection policy for RoCEv2 or use the Cisco UCS Manager Message Queue (MQ)-Server Message Block (SMB) direct (SMBd) policy.

Configuring SMB Direct Mode 1 on Cisco UCS Manager

To avoid possible RDMA packet drops, make sure same no-drop COS is configured across the network.

Before you begin

Configure a no-drop class in UCSM QoS Policies and use it for RDMA supported interfaces. Go to **LAN** > **LAN Cloud** > **QoS System Class** and enable **Priority** Platinum with CoS 5.

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Policies**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Expand **Adapter Policies** and choose the existing adapter policy for Win-HPN-SMBd.

If using a user-defined adapter policy, use the configuration steps below.

a) On the **General** tab, scroll down to **RoCE** and click the **Enabled** radio button.
b) In the **RoCE Properties** field, under **Version 1**, click the **Disabled** radio button. For **Version 2**, click the **Enabled** radio button.
c) For **Queue Pairs**, enter `256`.
d) For **Memory Regions**, enter `131072`.
e) For **Resource Groups**, enter 2.
f) For **Priority**, choose **Platinum No-Drop COS.** from the dropdown.
g) Click **Save Change**s.

Step 5 Next, create an Ethernet Adapter Policy. In the Navigation pane, click **LAN**.

Step 6 Expand **LAN** > **Policies**.

Step 7 Right-click the **vNIC Templates** node and choose **Create vNIC Template**.

Step 8 Go to **vNIC Properties** under the General tab and modify the vNIC policy settings as follows:

a) Set **MTU** to `1500` or `4096` or `9158`.
b) For the Adapter Policy, select `Win-HPN-SMBd`
c) For the **QoS policy**, specify `Platinum`.

Step 9 Click **Save Changes**.

Step 10 After you save the changes, Cisco UCS Manager will prompt you to reboot. Reboot the system.

What to do next

When the server comes back up, configure RoCEv2 mode 1 on the Host.

Configuring SMB Direct Mode 1 on the Host System

Perform this procedure to configure a connection between smb-client and smb-server on two host interfaces. For each of these servers, smb-client, and smb-server, configure the RoCEv2-enabled vNIC.

Before you begin

Configure RoCEv2 for Mode 1 in Cisco UCS Manager.

Procedure

Step 1 In the Windows host, go to the **Device Manager** and select the appropriate Cisco VIC Internet Interface.

Step 2 Select the **Advanced** tab and verify that the **Network Direct Functionality** property is **Enabled**. If not, enable it and click **OK**.

Perform this step for both the smb-server and smb-client vNICs.

Step 3 Go to **Tools** > **Computer Management** > **Device Manager** > **Network Adapter** > click **VIC Network Adapter** > **Properties** > **Advanced** > **Network Direct Functionality**. Perform this operation for both the smb-server and smb-client vNICs.

Step 4 Verify that RoCE is enabled on the host operating system using PowerShell.

Execute the **`Get-NetOffloadGlobalSetting`** command to verify that **NetworkDirect** is enabled:

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

ReceiveSideScaling           : Enabled
ReceiveSegmentCoalescing     : Enabled
Chimney                      : Disabled
TaskOffload                  : Enabled
NetworkDirect                : Enabled
NetworkDirectAcrossIPSubnets : Blocked
PacketCoalescingFilter       : Disabled
```

Note
If the **NetworkDirect** setting is showing as disabled, enable it using the following command:

`Set-NetOffloadGlobalSetting -NetworkDirect enabled`

Step 5 Bring up the Powershell and execute the **`get -SmbClientNetworkInterface`** command.

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> Get-SmbClientNetworkInterface
Interface Index    RSS Capable    RDKA Capable    Speed    IpAddresses    Friendly Name
---------------    ------------   ------------    -------  -----------  ---------------
14                 True           False           40 Gbps    {10.37.60.162}    vEthernet
(vswitch)
26                 True           True            40 Gbps    {10.37.60.158}    vEthernet
(vpl)
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>
```

Step 6 Enter **enable - netadapterrdma [-name] ["Ethernetname"]**

Step 7 Verify the overall RoCEv2 Mode 1 configuration at the host:

a) Use the Powershell command **netstat -xan** to verify the listeners in both the smb-client and smb-server Windows host; listeners will be shown in the command output.

```
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) Go to the smb-client server fileshare and start an I/O operation.

c) Go to the performance monitor and check that it displays the RDMA activity.

Step 8 In the Powershell command window, check the connection entries with the **netstat -xan** output command to make sure they are displayed. You can also run **netstat -xan** from the command prompt. If the connection entry shows up in netstat-xan output, the RoCEv2 mode1 connections are correctly established between client and server.

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

Step 9 By default, Microsoft's SMB Direct establishes two RDMA connections per RDMA interface. You can change the number of RDMA connections per RDMA interface to one or any number of connections.

For example, to increase the number of RDMA connections to 4, execute the following command in PowerShell:

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

Configuring Mode 2 on Cisco UCS Manager

You will apply the VMQ Connection Policy as vmmq.

Before you begin

Configure RoCEv2 Policies in Mode 1.

Use the pre-defined default adapter policy “MQ-SMBd”, or configure a user-defined Ethernet adapter policy with the following recommended RoCE-specific parameters:

- RoCE: Enabled
- Version 1: disabled
- Version 2: enabled
- Queue Pairs: 256
- Memory Regions: 65536

- Resource Groups: 2
- Priority: Platinum

Create a VMQ connection policy with the following values:

- Multi queue: Enabled
- Number of sub-vNIC: 16
- VMMQ adapter policy: MQ-SMBd

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand **Service Profiles** > **vNICs** and choose the VMQ Connection policy profile to configure.

Step 4 Go to **vNIC Properties** under the General tab and scroll down to the Policies area. Modify the vNIC policy settings as follows:

a) For the Adapter Policy, make sure it uses **`Win-HPN-SMBd`** or the adapter policy configured earlier for Mode 1.

b) For the **QoS policy**, select **`best-effort`**.

Step 5 Click **Save Changes**.

Step 6 In the Navigation pane, click **LAN**.

Step 7 Expand **LAN** > **Policies** > **QoS Policy Best Effort**.

Step 8 Set **Host Control** to **`Full`**.

Step 9 Click **Save Changes**.

Step 10 After you save the changes, Cisco UCS Manager will prompt you to reboot. Reboot the interface.

What to do next

When the server comes back up, configure Mode 2 on the Host.

Configuring SMB Direct Mode 2 on the Host System

This task uses Hyper-V virtualization software that is compatible with Windows Server 2019 and later.

Before you begin

- Configure and confirm the connection for RoCEv2 Mode 2 for both the Cisco UCS Manager and Host.
- Configure RoCEv2 Mode 2 in Cisco UCS Manager.
- Enable Hyper-V at the Windows host server.

Procedure

Step 1 Go to the Hyper-V switch manager.

Step 2 Create a new Virtual Network Switch (vswitch) for theRoCEv2-enabled Ethernet interface.

a) Choose **External Network** and select **VIC Ethernet Interface 2** and **Allow management operating system to share this network adapter**.

b) Click **OK** to create the virtual switch.

Bring up the Powershell interface.

Step 3 Configure the non-default vport and enable RDMA with the following Powershell commands:

```
add-vmNetworkAdapter -switchname vswitch -name vp1 -managementOS

enable-netAdapterRdma -name "vEthernet (vp1"

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) Configure the set-switch using the following Powershell command.

```
new-vmswitch -name setswitch -netAdapterName “Ethernet x” -enableEmbeddedTeam $true
```

This creates the switch. Use the following to display the interfaces:

```
get-netadapterrdma

add-vmNetworkAdapter -switchname setswtch -name svp1
```

You will see the new vport when you again enter

```
get-netadapterrdma
```

b) Add a vport:

```
add-vmNetworkAdapter -switchname setswtch -name svp1
```

You see the new vport when you again enter:

```
get-netadapterrdma
```

c) Enable the RDMA on the vport:

```
enable-netAdapterRdma -name “vEthernet (svp1)”
```

Step 4 Configure the IPv4 addresses on the RDMA enabled vport in both servers.

Step 5 Create a share in smb-server and map the share in the smb-client.

a) For smb-client and smb-server in the host system, configure the RoCEv2-enabled vNIC as described above.

b) Configure the IPv4 addresses of the primary fabric and sub-vNICs in both servers, using the same IP subnet and same unique VLAN for both.

c) Create a share in smb-server and map the share in the smb-client.

Step 6 Finally, verify the Mode 2 configuration.

a) Use the Powershell command **netstat -xan** to display listeners and their associated IP addresses.

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> netstat -xan
Active    NetworkDirect Connections, Listeners, SharedEndpoints
Mode    Iflndex    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) Start any RDMA I/O in the file share in smb-client.

c) Issue the **netstat -xan** command again and check for the connection entries to verify they are displayed.

```
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>
```

Configuring RoCEv2 in Linux

Configuring NVMeoF Using RoCEv2 on Cisco UCS Manager

Use these steps to configure the RoCEv2 interface on Cisco UCS Manager.

Before you begin

- Configure a no-drop QoS system class for RoCEv2 traffic.
- Create a QoS policy that uses the no-drop system class.
- Verify that the server, Cisco UCS VIC, host operating system, VIC firmware, and drivers are supported. For more information, see Cisco UCS Hardware and Software Compatibility.

Direct-attached Pure Storage configuration

For a supported Pure Storage array connected directly to a fabric interconnect appliance port or appliance port channel, Cisco recommends the following configuration:

- Configure the Platinum no-drop QoS system class with **CoS** 5 and **DSCP** 46.
- Configure an Ethernet interface as an appliance port, or create an appliance port channel. Set **Priority** to **Platinum** and **Match DSCP** to **Yes**.
- Use access mode for the storage VLAN. If you use trunk mode, add the storage VLAN to the allowed VLAN list and configure the same VLAN as the native VLAN.

For instructions on configuring QoS system classes, appliance ports, and appliance port channels, see the Cisco UCS Manager Network Management Guide.

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Click on **vNICs** and go to the **Network** tab in the work area.

Step 5 In the **vNICs** area, select the vNIC that you want to configure, and click **Modify**.

Step 6 In the **Modify vNIC** dialog box, scroll to the **Adapter Performance Profile** area and configure the following fields:

a) From the **Adapter Policy** drop-down list, choose **Linux-NVMe-RoCE**.

b) From the **QoS Policy** drop-down list, choose the QoS policy that uses the no-drop system class configured for RoCEv2.

For example, if **Platinum** is configured globally as the no-drop system class with **CoS 5**, choose a QoS policy that uses the Platinum priority.

Step 7 Click **OK**.

Note
Cisco UCS Manager supports a maximum MTU of 9158 bytes.

Step 8 Click **Save Changes**.

What to do next

Enabling SRIOV BIOS Policy, on page 8

Enabling SRIOV BIOS Policy

Use these steps to configure the server's service profile with the SRIOV BIOS policy before enabling the IOMMU in the Linux kernel.

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Select the service profile node where you want to enable the BIOS Policy.

Step 5 In the Work pane, select **Policies** tab.

Step 6 In the Policies Area, expand **BIOS Policy**.

Step 7 Choose the default SRIOV policy from the **BIOS Policy** drop-down list.

Step 8 Click **Save Changes**.

Configuring NVMeoF Using RoCEv2 on the Host

Before you begin

Configure the server with RoCEv2 vNIC and the SRIOV-enabled BIOS policy.

Procedure

Step 1 Open the `/etc/default/grub` file for editing.

Step 2 Add `intel_iommu=on` to the end of the line for `GRUB_CMDLINE_LINUX`as shown in the sample file below.

```
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"
```

Step 3 Save the file.

Step 4 After saving the file, run the following command to generate a new grub.cfg file:

- For Legacy boot:

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

- For UEFI boot:

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

Step 5 Reboot the server. You must reboot your server for the changes to take after enabling IOMMU.

Step 6 Verify that the server booted with the `intel_iommu=on` option by checking the output file.

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

Note its inclusion at the end of the output.

```
[root@localhost basic-setup]# cat /proc/cmdline | grep iommu
BOOT_IMAGE=/vmlinuz-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
```

What to do next

Download the enic and enic_rdma drivers.

Installing Cisco enic and enic_rdma Drivers

The enic_rdma driver requires enic driver. When installing enic and enic_rdma drivers, download and use the matched set of enic and enic_rdma drivers on Cisco.com. Attempting to use the binary enic_rdma driver downloaded from Cisco.com with an inbox enic driver, will not work.

Procedure

Step 1 Install the enic and enic_rdma rpm packages:

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

Note
During enic_rdma installation, the enic_rdmalibnvdimm module may fail to install on RHEL 7.7 because the `nvdimm-security.conf` dracut module needs spaces in the `add_drivers` value. For workaround, please follow the instruction from the following links:

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

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

Step 2 The enic_rdma driver is now installed but not loaded in the running kernel. Reboot the server to load enic_rdma driver into the running kernel.

Step 3 Verify the installation of enic_rdma driver and RoCE v2 interface:

```
# 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
```

Step 4 Load the vme-rdma kernel module:

```
# modprobe nvme-rdma
```

After server reboot, nvme-rdma kernel module is unloaded. To load nvme-rdma kernel module every server reboot, create nvme_rdma.conf file using:

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

Note
For more information about enic_rdma after installation, use the **`rpm -q -l kmod-enic_rdma`** command to extract the README file.

What to do next

Discover targets and connect to NVMe namespaces. If your system needs multipath access to the storage, please go to the section for Setting Up Device Mapper Multipath, on page 12.

Discovering the NVMe Target

Use this procedure to discover the NVMe target and connect NVMe namespaces.

Before you begin

Install **nvme-cli** version 1.6 or later if it is not installed already.



Note Skip to Step 2 below if nvme-cli version 1.7 or later is installed.

Configure the IP address on the RoCE v2 interface and make sure the interface can ping the target IP.

Procedure

Step 1 Create an nvme folder in /etc, then manually generate host nqn.

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

Step 2 Create a settos.sh file and run the script to set priority flow control (PFC) in IB frames.

Note
To avoid failure of sending NVMeoF traffic, you *must* create and run this script after *every* server reboot.

```
# 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
```

Step 3 Discover the NVMe target by entering the following command.

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

For example, to discover the target at 50.2.85.200:

```
# 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
treq:    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
```

Note

To discover the NVMe target using IPv6, put the IPv6 target address next to the `traddr` option.

Step 4 Connect to the discovered NVMe target by entering the following command.

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

For example, to discover the target at 50.2.85.200 and the subnqn value found above:

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

Note
To connect to the discovered NVMe target using IPv6, put the IPv6 target address next to the `traddr` option.

Step 5 Use the **nvme list** command to check mapped namespaces:

```
# 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
```

Setting Up Device Mapper Multipath

If your system is configured with Device Mapper multipathing (DM Multipath), use the following steps to set up Device Mapper multipath.

Note For NVMe-over-RDMA configurations on Red Hat Enterprise Linux (RHEL), use NVMe Native Multipath instead of Device Mapper (DM) Multipath.

- **RHEL 8.x**: NVMe Native Multipath is available as an alternative to DM-Multipath.
- **RHEL 9.x**: NVMe Native Multipath is the recommended configuration for NVMe-over-RDMA.
- **RHEL 10.x and later**: NVMe Native Multipath is the default configuration for Red Hat distributions.
- Native multipath uses a NUMA-aware policy by default to optimize performance. Administrators can reconfigure the multipath policy from the default NUMA-aware setting to a round-robin policy at the host level if required by specific application workloads.

Procedure

Step 1 Install the `device-mapper-multipath` package if it is not installed already

Step 2 Enable and start multipathd:

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

Step 3 Edit the etc/multipath.conf file to use the following values :

```
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
}
```

Step 4 Flush with the updated multipath device maps.

```
# multipath -F
```

Step 5 Restart multipath service:

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

Step 6 Rescan multipath devices:

```
# multipath -v2
```

Step 7 Check the multipath status:

```
# multipath -ll
```

Deleting the RoCEv2 Interface Using Cisco UCS Manager

Use these steps to remove the RoCE v2 interface

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Modify the vNIC policy, according to the steps below.

a) On the **Network** tab, scroll down to the desired vNIC and click on it, then click **Modify**.

b) A popup dialog box will be displayed. Scroll down to the **Policies** area, and choose **Linux** from the **Adapter Policy** drop-down list.

c) Click OK.

Step 5 Click **Save Changes**.

Configuring RoCEv2 in ESXi

Configuring NVMEoF using RoCEv2 for ESXi on UCS Manager

UCS Manager contains a default adapter policy that is prepopulated with operational parameters, so you do not need to manually create the adapter policy. However, you do need to create the RoCEv2 interface.

Use these steps to configure the RoCEv2 interface on UCS Manager.

Before you begin

- Configure a no-drop QoS system class for RoCEv2 traffic.
- Create a QoS policy that uses the no-drop system class.
- Verify that the server, Cisco UCS VIC, host operating system, VIC firmware, and drivers are supported. For more information, see Cisco UCS Hardware and Software Compatibility.

Direct-attached Pure Storage configuration

For a supported Pure Storage array connected directly to a fabric interconnect appliance port or appliance port channel, Cisco recommends the following configuration:

- Configure the Platinum no-drop QoS system class with **CoS** 5 and **DSCP** 46.
- Configure an Ethernet interface as an appliance port, or create an appliance port channel. Set **Priority** to **Platinum** and **Match DSCP** to **Yes**.
- Use access mode for the storage VLAN. If you use trunk mode, add the storage VLAN to the allowed VLAN list and configure the same VLAN as the native VLAN.

For instructions on configuring QoS system classes, appliance ports, and appliance port channels, see the Cisco UCS Manager Network Management Guide.

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the organization where you want to create the policy.

If the system does not include multitenancy, expand the **root** node.

Step 4 Click on a RDMA service profile you created and expand the service profile.

Step 5 Right-click on **vNICs** and choose **Create vNIC** to create a new vNIC.

The **Create VNIC** pop-up menu is displayed.

Perform the below steps to modify the vNIC policy:

a) Name the new VNIC.

b) On the **MAC address** drop-down, select the option from Manual using OUI or Domain Pools in the drop-down.
c) Select which VLAN you want use from the list.
d) In the Adapter Performance Profile, select the default adapter policy named `VMWareNVMeRoCEv2`.
e) Click **OK**. The interface is now configured for one port.

Step 6 Click **Save Changes**.

What to do next

Install the NENIC Driver.

Installing NENIC Driver

The eNIC drivers, which contain the RDMA driver, are available as a combined package. Download and use the eNIC driver on cisco.com.

These steps assume this is a new installation.

Note While this example uses the `/tmp` location, you can place the file anywhere that is accessible to the ESX console shell.

Procedure

Step 1 Copy the eNIC VIB or offline bundle to the ESX server. The example below uses the Linux **scp** utility to copy the file from a local system to an ESX server located at 10.10.10.10: and uses the location /tmp.

```
scp nenic-2.0.4.0-1OEM.700.1.0.15843807.x86_64.vib root@
10.10.10.10:/tmp
```

Step 2 Specifying the full path, issue the command shown below.

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

or

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

Example:

```
esxcli software vib install -v /tmp/nenic-2.0.4.0-1OEM.
700.1.0.15843807.x86_64.vib
```

Note
Depending on the certificate used to sign the VIB, you may need to change the host acceptance level. To do this, use the command: `esxcli software acceptance set --level=<level>`

Depending on the type of VIB being installed, you may need to put ESX into maintenance mode. This can be done through the VI Client, or by adding the `--maintenance-mode` option to the above `esxcli` command.

Upgrading NENIC Driver

a. To upgrade NENIC driver, enter the command:

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

or

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

b. Copy the enic VIB or offline bundle to the ESX server using Step 1 given above.

What to do next

Configure the ESXi Host side NVMe RDMA.

ESXi NVMe RDMA Host Side Configuration

NENIC RDMA Functionality

Differences between the use case for RDMA on Linux and ESXi:

- In ESXi, the physical interface (vmnic) MAC is not used for RoCEv2 traffic. Instead, the VMkernel port (vmk) MAC is used.

  Outgoing RoCE packets use the vmk MAC in the Ethernet source MAC field, and incoming RoCE packets use the vmk MAC in the Ethernet destination mac field. The vmk MAC address is a VMware MAC address assigned to the vmk interface when it is created.

- In Linux, the physical interface MAC is used in source MAC address field in the RoCE packets. This Linux MAC is usually a Cisco MAC address configured to the VNIC using Cisco UCS Manager.

If you ssh into the host and use the **esxcli network ip interface list** command, you can see the MAC address.

```
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
```

You must create a vSphere Standard Switch to provide network connectivity for hosts, virtual machines, and VMkernel traffic. Depending on the connection type that you want to create, you can create a new vSphere Standard Switch with a VMkernel adapter, only connect physical network adapters to the new switch, or create the switch with a virtual machine port group.

Create Network Connectivity Switches

Use these steps to create a vSphere Standard Switch to provide network connectivity for hosts, virtual machines, and to VMkernel traffic.

Before you begin

Ensure that you have downloaded and installed the NENIC driver.

Procedure

Step 1 In the vSphere Client, navigate to the host.

Step 2 On the **Configure** tab, expand **Networking** and select **Virtual Switches.**

Step 3 Click on **Add Networking**.

The available network adapter connection types are:

- **Vmkernel Network Adapter**

  Creates a new VMkernel adapter to handle host management traffic

- **Physical Network Adapter**

  Adds physical network adapters to a new or existing standard switch.

- **Virtual Machine Port Group for a Standard Switch**

  Creates a new port group for virtual machine networking.

Step 4 Select connection type **Vmkernel Network Adapter**.

Step 5 Select **New Standard Switch** and click **Next**.

Step 6 Add physical adapters to the new standard switch.

a) Under **Assigned Adapters**, select **New Adapters**.
b) Select one or more adapters from the list and click **OK**. To promote higher throughput and create redundancy, add two or more physical network adapters to the Active list.
c) (Optional) Use the up and down arrow keys to change the position of the adapter in the Assigned Adapters list.
d) Click **Next**.

Step 7 For the new standard switch you just created for the VMadapter or a port group, enter the connection settings for the adapter or port group.

a) Enter a label that represents the traffic type for the VMkernel adapter.
b) Set a VLAN ID to identify the VLAN the VMkernel uses for routing network traffic.
c) Select IPV4 or IPV6 or both.
d) Select an MTU size from the drop-down menu. Select Custom if you wish to enter a specific MTU size. The maximum MTU size is 9000 bytes.

   Note
   You can enable Jumbo Frames by setting an MTU greater than 1500.

e) After setting the TCP/IP stack for the VMkernel adapter, select a TCP/IP stack.

To use the default TCP/IP stack, select it from the available services.

Note
Be aware that the TCP/IP stack for the VMkernel adapter cannot be changed later.

f) Configure IPV4 and/or IPV6 settings.

Step 8 On the **Ready to Complete** page, click **Finish**.

Step 9 Check the VMkernel ports for the VM Adapters or port groups with NVMe RDMA in the vSphere client, as shown in the Results below.

What to do next

Create vmhba ports on top of vmrdma ports.

Create VMHBA Ports in ESXi

Use the following steps for creating vmhba ports on top of the vmrdma adapter ports.

Before you begin

Create the adapter ports for storage connectivity.

Procedure

Step 1 Go to vCenter where your ESXi host is connected.

Step 2 Click on **Host**>**Configure**>**Storage adapters**.

Step 3 Click **+Add Software Adapter**.

Add Software Adapter dialog box is displayed.

Step 4 Select **Add software NVMe over RDMA adapter** and the vmrdma port you want to use.

Step 5 Click **OK**

The vmhba ports for the VMware NVMe over RDMA storage adapter will be shown.

What to do next

Configure NVMe.

Displaying vmnic and vmrdma Interfaces

ESXi creates a vmnic interface for each enic VNIC configured to the host.

Before you begin

Create Network Adapters and VHBA ports.

Procedure

Step 1 Use **ssh** to access the host system.

Step 2 Enter **esxcfg-nics -l** to list the vmnics on ESXi.

```
Name   PCI          Driver  Link  Speed     Duplex  MAC Address       MTU  Description
vmnico 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
Venic5 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
Description
vmnico 0000:3b:00.0 ixgben  Up           Down        0     Half   2c:f8:9b:a1:4c:e6 1500
Intel(R) Ethernet Controller X550
vmnic1 0000:36:00.1 ixgben  Up           Up          1000  Full   2c:f8:9b:a1:4c:e7 1500
Intel(R) Ethernet Controller X550
vmnic2 0000:1d:00.0 nenic   Up           Up          50000 Full   2c:f8:9b:79:8d:bc 1500
Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic3 0000:1d:00.1 nenic   Up           Up          50000 Full   2c:f8:9b:79:8d:bd 1500
Cisco Systems Inc Cisco VIC Ethernet NIC
vmnic4 0000:63:00.0 nenic   Up           Down        0     Half   2c:f8:9b:51:b3:3a 1500
Cisco Systems Inc Cisco VIC Ethernet NIC
Venic5 0000:63:00.1 nenic   Up           Down        0     Half   2c:f8:9b:51:b3:3b 1500
Cisco Systems Inc Cisco VIC Ethernet NIC
```

When the enic driver registers with ESXi the RDMA device for a RDMA capable VNIC, ESXi creates a vmrdma device and links it to the corresponding vmnic.

Step 3 Use **esxcli rdma device list** to list the vmrdma devices.

```
[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)
vmrdmal nenic  Active 4096 50 Gbps vmnic2        Cisco UCS VIC 15XXX (A0)
[root@StockholmRackServer:~] esxcli rdma device vmknic list
Device  Vmknic NetStack
------- ------ --------
vmrdma0 vmk1   defaultTcpipStack
vmrdma1 vmk2   defaultTcpipStack
```

Step 4 Use **esxcli rdma device list** to check the protocols supported by the vmrdma interface.

For enic, RoCE v2 will be the only protocol supported from this list. The output of this command should match the RoCEv2 configuration on the VNIC.

Step 5 Use **esxcli rdma device protocol list** to check the protocols supported by the vmrdma interface.

For enic RoCE v2 will be the only protocol supported from this list. The output of this command should match the RoCEv2 configuration on the VNIC.

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

Step 6 Use **esxcli nvme adapter list** to list the NVMe adapters and the vmrdma and vmnic interfaces it is configured on.

```
[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
```

Step 7 All vmhbas in the system can be listed using **esxcli storage core adapter list**.

```
[root@RackServer:~] esxcli storage core adapter list
HBA Name Driver   Link State UID                                   Capabilities
Description
-------- ------   ---------- ----------------------------------- --------------------
-------------------------------------
vmhbao   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
```

What to do next

Configure NVME.

NVMe Fabrics and Namespace Discovery

This procedure is performed through the ESXi command line interface.

Before you begin

Create and configure NVMe on the adapter's VMHBAs. The maximum number of adapters is two, and it is a best practice to configure both for fault tolerance.

Procedure

Step 1 Check and enable NVMe on the vmrdma device.

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

The system should return a message showing if NVMe is enabled.

Step 2 Discover the NVMe fabric on the array by entering the following command:

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

figure with **`esxcli nvme fabrics discover -a vmhba64 -l 50.2.84.100`**

The output lists the following information: Transport Type, Address Family, Subsystem Type, Controller ID, Admin Queue, Max Size, Transport Address, Transport Service ID, and Subsystem NQN

You will see output on the NVMe controller.

Step 3 Perform NVMe fabric interconnect.

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

Step 4 Repeat steps 1 through 4 to configure the second adapter.

Step 5 Display the controller list to verify the NVMe controller is present and operating.

`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
```

Example

The following example shows esxcli discovery commands executed on the server.

```
[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
```

Using the UCS Manager CLI to Configure or Delete the RoCEv2 Interface

Configure Windows SMB Direct RoCEv2 Interface using UCS Manager CLI

Use the following steps to configure the RoCEv2 interface in the Cisco UCS Manager CLI.

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic vnic name
```

Enter the vnic mode for the specified vNIC.

Step 4 To configure Windows SMBDirect RoCEv2 Mode 1:

Example:

```
 UCS-A /org/service-profile/vnic # set adapter-policy Win-HPN-SMBd
```

Specifies a Windows SMBDirect RoCEv2 adapter policy for RoCEv2 Mode 1.

Step 5 To configure Windows SMBDirect RoCEv2 Mode 2:

Example:

```
 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 #
```

Configures Windows Mode 2, after creating a VMQ connection policy and assigning the adapter policy **MQ-SMBd**:

Step 6 **Example:**

```
 UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

This example shows how to configure the RoCEv2 Win-HPN-SMBd adapter policy:

```
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 #
```

Deleting the Windows RoCEv2 Interface Using the CLI for UCS Manager

Use the following steps to delete the Windows RoCEv2 interface in the Cisco UCS Manager CLI.

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic vnic name
```

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

```
 UCS-A /org/service-profile/vnic # set adapter-policy Windows
```

Removes the Windows RoCEv2 adapter policy by setting the default Windows adapter policy.

Step 5 **Example:**

```
 UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

What to do next

This example shows how to remove the RoCEv2 interface on the eth01 vNIC on Windows.

```
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 #
```

Configuring the Linux RoCEv2 Interface Using the UCS Manager CLI

Use the following steps to configure the RoCEv2 interface for Linux in the Cisco UCS Manager CLI.

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic vnic name
```

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

```
 UCS-A /org/service-profile/vnic # set adapter-policy Linux-NVMe-RoCE
```

Specify Linux-NVMe-RoCE as the adapter policy for the vNIC that you want to use for NVMeoF.

Step 5 **Example:**

```
 UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

This example shows how to configure the RoCEv2 Linux adapter policy on the eth01 vNIC:

Example

```
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 #
```

Deleting the Linux RoCEv2 Interface Using the UCS Manager CLI

Use the following steps to delete the Linux RoCEv2 interface in the Cisco UCS Manager CLI.

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic vnic name
```

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

```
 UCS-A /org/service-profile/vnic # set adapter-policy Linux
```

Removes Linux-NVMe-RoCE policy by setting the default Linux adapter policy.

Step 5 **Example:**

```
 UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

This example shows how to remove the RoCEv2 interface on the eth01 vNIC on Linux.

Example

```
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
```

Configuring the VMware ESXi RoCEv2 Interface Using the UCS Manager CLI

Use the following steps to configure the RoCEv2 interface for VMware ESXi in the Cisco UCS Manager CLI.

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

```
UCS-A # scope service-profile server chassis-id / blade-id or rack_server-id
```

Enter the service profile for the specified chassis, blade or UCS managed rack server ID.

Step 2 **Example:**

```
UCS-A /org/service-profile # show vnic
```

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic vnic name
```

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

```
 UCS-A /org/service-profile/vnic # set adapter-policy VMWareNVMeRoCEv2
```

Specify VMWareNVMeRoCEv2 as the adapter policy for the vNIC that you want to use for NVMeoF.

Step 5 **Example:**

```
 UCS-A /org/service-profile/vnic* # commit-buffer
```

Commit the transaction to the system configuration.

This example shows how to configure the RoCEv2 VMware adapter policy on the eth01 vNIC:

Example

```
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 #
```

Deleting the ESXi RoCEv2 Interface Using UCS Manager

Use these steps to remove the RoCE v2 interface for a specific port.

Procedure

Step 1 In the **Navigation** pane, click **Servers**.

Step 2 Expand **Servers** > **Service Profiles**.

Step 3 Expand the node for the profile to delete.

Step 4 Click on **vNICs** and select the desired interface. Right click and select **Delete** from the dropdown.

Step 5 Click **Save Changes**.