Network-Related Policies

Configuring vNIC Templates

vNIC Template

The vNIC LAN connectivity policy defines how a vNIC on a server connects to the LAN.

Cisco UCS Manager does not automatically create a VM-FEX port profile with the correct settings when you create a vNIC template. If you want to create a VM-FEX port profile, you must configure the target of the vNIC template as a VM. You must include this policy in a service profile for it to take effect.

You can select VLAN groups in addition to any individual VLAN while creating a vNIC template.

Note If your server has two Emulex or QLogic NICs (Cisco UCS CNA M71KR-E or Cisco UCS CNA M71KR-Q), you must configure vNIC policies for both adapters in your service profile to get a user-defined MAC address for both NICs. If you do not configure policies for both NICs, Windows still detects both of them in the PCI bus. Then because the second eth is not part of your service profile, Windows assigns it a hardware MAC address. If you then move the service profile to a different server, Windows sees additional NICs because one NIC did not have a user-defined MAC address.

Creating a vNIC Template

Before you begin

This policy requires that one or more of the following resources already exist in the system:

- Named VLAN
- MAC pool
- QoS policy
- LAN pin group
- Statistics threshold policy

Procedure

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

Step 2 Expand **LAN** > **Policies**.

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

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

Step 4 Right-click the **vNIC Templates** node and choose **Create vNIC Template**.

Step 5 In the **Create vNIC Template** dialog box:

a) In the **General** area, complete the following fields:

Name	Description
Name field	The name of the vNIC template. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Description field	A user-defined description of the template. Enter up to 256 characters. You can use any characters or spaces except ` (accent mark), \ (backslash), ^ (carat), " (double quote), = (equal sign), > (greater than), < (less than), or ' (single quote).

Name	Description
Fabric ID field	The fabric interconnect associated with the component. If you want vNICs created from this template to be able to access the second fabric interconnect if the default one is unavailable, check the **Enable Failover** check box. **Note** Do not enable vNIC fabric failover under the following circumstances: • If the Cisco UCS domain is running in Ethernet switch mode. vNIC fabric failover is not supported in that mode. If all Ethernet uplinks on one fabric interconnect fail, the vNICs do not fail over to the other. • If you plan to associate one or more vNICs created from this template to a server with an adapter that does not support fabric failover, such as the Cisco UCS 82598KR-CI 10-Gigabit Ethernet Adapter. If so, Cisco UCS Manager generates a configuration fault when you associate the service profile with the server.
Redundancy Type	The Redundancy type that you choose initiates a fabric failover using vNIC/HBA redundancy pairs. • **Primary Template**— Creates configurations that can be shared with the Secondary template. Any other shared changes on the Primary template are automatically synchronized to the Secondary template. • **Secondary Template**—All shared configurations are inherited from the Primary template. • **No Redundancy**—Legacy vNIC/vHBA template behavior. Select this option if you do not want to use redundancy.
Target list box	A list of the possible targets for vNICs created from this template. The target you choose determines whether or not Cisco UCS Manager automatically creates a VM-FEX port profile with the appropriate settings for the vNIC template. This can be one of the following: • **Adapter**—The vNICs apply to all adapters. No VM-FEX port profile is created if you choose this option. • **VM**—The vNICs apply to all virtual machines. A VM-FEX port profile is created if you choose this option.

Name	Description
Template Type field	• **Initial Template**: vNICs created from this template are not updated if the template changes. • **Updating Template**: vNICs created from this template are updated if the template changes.

b) In the **VLANs** area, use the table to select the VLAN to assign to vNICs created from this template. The table contains the following columns:

Name	Description
Select column	Check the check box in this column for each VLAN that you want to use. **Note** VLANs can not be assigned to the same vNIC.
Name column	Displays the name of the VLAN.
Native VLAN column	Click the radio button in this column to designate one of the VLANs as the native VLAN.
VLAN ID column	The unique identifier of the VLAN.

c) In the **VLAN Goups** area, use the table to select the VLAN group to assign to vNICs created from this template. The table contains the following columns:

Name	Description
Select column	Check the check box in this column for each VLAN Group that you want to use.
Name column	The name of the VLAN Group.

d) In the **Policies** area, complete the following fields:

Name	Description
CDN Source field	This can be one of the following options: • **vNIC Name**—Uses the vNIC template name of the vNIC instance as the CDN name. This is the default option. • **User Defined**—Displays the CDN Name field for you to enter a user-defined CDN name for the vNIC template. Refer to the *Cisco UCS Manager Server Management Guide* for more information on Consistent Device Naming.
CDN Name field	Enter the CDN Name. This field is displayed only when the **CDN Source - User Defined** option is selected.

Name	Description
MTU field	The maximum transmission unit, or packet size, that vNICs created from this vNIC template should use. Enter an integer between 1500 and 9000. **Note** If the vNIC template has an associated QoS policy, the MTU specified here must be equal to or less than the MTU specified in the associated QoS system class. If this MTU value exceeds the MTU value in the QoS system class, packets may be dropped during data transmission. For VIC 1400 Series and VIC 15000 Series adapters, you can change the MTU size of the vNIC from the host interface settings. When the Overlay network is configured, make sure that the new value is equal to or less than the MTU specified in the associated QoS system class or packets could be dropped during data transmission.
MAC Pool drop-down list	The MAC address pool that vNICs created from this vNIC template should use.
QoS Policy drop-down list	The quality of service policy that vNICs created from this vNIC template should use.
Network Control Policy drop-down list	The network control policy that vNICs created from this vNIC template should use.
Pin Group drop-down list	The LAN pin group that vNICs created from this vNIC template should use.
Stats Threshold Policy drop-down list	The statistics collection policy that vNICs created from this vNIC template should use.

e) In the **Connections Policies** area, complete the following fields:

Name	Description
Connection Policy radio button	Choose the type of connection policy to associate with the vNIC. This can be one of the following: • **Dynamic vNIC** • **usNIC** • **VMQ**
Connection Policy drop-down list	Choose the connection policy that the vNIC should use. The values displayed depend on the type of connection policy chosen. You can also create a new connection policy in this area.

Step 6 Click **OK**.

What to do next

Include the vNIC template in a service profile.

Creating vNIC Template Pairs

Procedure

Step 1 In the Navigation pane, click the **LAN** tab. On the **LAN** tab, expand **LAN > Policies.**

Step 2 Expand the node for the organization where you want to create the policy. If the system does not include multi-tenancy, expand the root node.

Step 3 Right-click the **vNIC Templates** node and choose **Create vNIC Template**. In the **Create vNIC Template** dialog box, assign a **Name**, **Description**, and select the **Fabric ID** for the template.

Step 4 Select the **Redundancy Type** as **Primary** or **Secondary** or **No Redundancy**. See the redundancy type descriptions below.

Step 5 Select the **Peer Redundancy Template**—to choose the name of the corresponding **Primary** or **Secondary** redundancy template to perform the template pairing from the **Primary** or **Secondary** redundancy template.

- **Primary**—Creates configurations that can be shared with the Secondary template. Any other shared changes on the Primary template are automatically synchronized to the Secondary template.
  - **VLANS**
  - **Template Type**
  - **MTU**
  - **Network Control Policies**
  - **Connection Policies**
  - **QoS Policy**
  - **Stats Threshold Policy**

  Following is a list of non-shared configurations:
  - **Fabric ID**

    Note The Fabric ID must be mutually exclusive. If you assign the Primary template to Fabric A, then Fabric B is automatically assigned to the Secondary template as part of the synchronization from the Primary template.

  - **CDN Source**
  - **MAC Pool**
  - **Description**
  - **Pin Group Policy**

- **Secondary**—

  All shared configurations are inherited from the Primary template.
- **No Redundancy**—

  Legacy vNIC template behavior.

Step 6 Click **OK**.

What to do next

After you create the vNIC redundancy template pair, you can use the redundancy template pair to create redundancy vNIC pairs for any service profile in the same organization or sub-organization.

Undo vNIC Template Pairs

You can undo the vNIC template pair by changing the Peer Redundancy Template so that there is no peer template for the Primary or the Secondary template. When you undo a vNIC template pair, the corresponding vNIC pairs also becomes undone.

Procedure

Select **not set** from the **Peer Redundancy Template** drop-down list to undo the paring between the peer Primary or Secondary redundancy template used to perform the template pairing. You can also select **None** as the **Redundancy Type** to undo the pairing.

Note If you delete one template in a pair, you are prompt to delete the other template in the pair. If you do not delete the other template in the pair, that template resets its peer reference and retains its redundancy type.

Binding a vNIC to a vNIC Template

You can bind a vNIC associated with a service profile to a vNIC template. When you bind the vNIC to a vNIC template, Cisco UCS Manager configures the vNIC with the values defined in the vNIC template. If the existing vNIC configuration does not match the vNIC template, Cisco UCS Manager reconfigures the vNIC. You can only change the configuration of a bound vNIC through the associated vNIC template. You cannot bind a vNIC to a vNIC template if the service profile that includes the vNIC is already bound to a service profile template.

Important If the vNIC is reconfigured when you bind it to a template, Cisco UCS Manager reboots the server associated with the service profile.

Procedure

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

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

Step 3 Expand the node for the organization that includes the service profile with the vNIC you want to bind.

If the system does not include multi-tenancy, expand the **root** node.

Step 4 Expand ***Service_Profile_Name*** > **vNICs**.

Step 5 Click the vNIC you want to bind to a template.

Step 6 In the **Work** pane, click the **General** tab.

Step 7 In the **Actions** area, click **Bind to a Template**.

Step 8 In the **Bind to a vNIC Template** dialog box, do the following:

a) From the **vNIC Template** drop-down list, choose the template to which you want to bind the vNIC.
b) Click **OK**.

Step 9 In the warning dialog box, click **Yes** to acknowledge that Cisco UCS Manager may need to reboot the server if the binding causes the vNIC to be reconfigured.

Unbinding a vNIC from a vNIC Template

Procedure

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

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

Step 3 Expand the node for the organization that includes the service profile with the vNIC you want to unbind.

If the system does not include multi-tenancy, expand the **root** node.

Step 4 Expand ***Service_Profile_Name*** > **vNICs**.

Step 5 Click the vNIC you want to unbind from a template.

Step 6 In the **Work** pane, click the **General** tab.

Step 7 In the **Actions** area, click **Unbind from a Template**.

Step 8 If a confirmation dialog box displays, click **Yes**.

Deleting a vNIC Template

Procedure

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

Step 2 Expand **LAN** > **Policies** > ***Organization_Name***.

Step 3 Expand the **vNIC Templates** node.

Step 4 Right-click the policy you want to delete and choose **Delete**.

Step 5 If a confirmation dialog box displays, click **Yes**.

Configuring Adapter Policies

Ethernet and Fibre Channel Adapter Policies

These policies govern the host-side behavior of the adapter, including how the adapter handles traffic. For example, you can use these policies to change default settings for the following:

- Queues
- Interrupt handling
- Performance enhancement
- RSS hash
- Failover in a cluster configuration with two fabric interconnects

Note For Fibre Channel adapter policies, the values displayed by Cisco UCS Manager may not match those displayed by applications such as QLogic SANsurfer. For example, the following values may result in an apparent mismatch between SANsurfer and Cisco UCS Manager:

- Max LUNs Per Target—SANsurfer has a maximum of 256 LUNs and does not display more than that number. Cisco UCS Manager supports a higher maximum number of LUNs. This parameter is applicable only for FC-Initiator.
- Link Down Timeout—In SANsurfer, you configure the timeout threshold for link down in seconds. In Cisco UCS Manager, you configure this value in milliseconds. Therefore, a value of 5500 ms in Cisco UCS Manager displays as 5s in SANsurfer.
- Max Data Field Size—SANsurfer has allowed values of 512, 1024, and 2048. Cisco UCS Manager allows you to set values of any size. Therefore, a value of 900 in Cisco UCS Manager displays as 512 in SANsurfer.
- LUN Queue Depth—The LUN queue depth setting is available for Windows system FC adapter policies. Queue depth is the number of commands that the HBA can send and receive in a single transmission per LUN. Windows Storport driver sets this to a default value of 20 for physical miniports and to 250 for virtual miniports. This setting adjusts the initial queue depth for all LUNs on the adapter. Valid range for this value is 1 - 254. The default LUN queue depth is 20. This feature only works with Cisco UCS Manager version 3.1(2) and higher. This parameter is applicable only for FC-Initiator.
- IO TimeOut Retry—When the target device does not respond to an IO request within the specified timeout, the FC adapter cancels the pending command then resends the same IO after the timer expires. The FC adapter valid range for this value is 1 - 59 seconds. The default IO retry timeout is 5 seconds. This feature only works with Cisco UCS Manager version 3.1(2) and higher.

Operating System Specific Adapter Policies

By default, Cisco UCS provides a set of Ethernet adapter policies and Fibre Channel adapter policies. These policies include the recommended settings for each supported server operating system. Operating systems are sensitive to the settings in these policies. Storage vendors typically require non-default adapter settings. You can find the details of these required settings on the support list provided by those vendors.

Important We recommend that you use the values in these policies for the applicable operating system. Do not modify any of the values in the default policies unless directed to do so by Cisco Technical Support.

However, if you are creating an Ethernet adapter policy for an OS (instead of using the default adapter policy), you must use the following formulas to calculate values that work for that OS.

Depending on the UCS firmware, your driver interrupt calculations may be different. Newer UCS firmware uses a calculation that differs from previous versions. Later driver release versions on Linux operating systems now use a different formula to calculate the Interrupt Count. In this formula, the Interrupt Count is the maximum of either the Transmit Queue or the Receive Queue plus 2.

Interrupt Count in Linux Adapter Policies

Drivers on Linux operating systems use differing formulas to calculate the Interrupt Count, depending on the eNIC driver version. The UCS 3.2 release increased the number of Tx and Rx queues for the eNIC driver from 8 to 256 each.

Use one of the following strategies, according to your driver version.

For Linux drivers before the UCS 3.2 firmware release, use the following formula to calculate the Interrupt Count.

Completion Queues = Transmit Queues + Receive Queues
Interrupt Count = (Completion Queues + 2) rounded up to nearest power of 2

For example, if Transmit Queues = 1 and Receive Queues = 8 then:

Completion Queues = 1 + 8 = 9
Interrupt Count = (9 + 2) rounded up to the nearest power of 2 = 16

On drivers for UCS firmware release 3.2 and higher, the Linux eNIC drivers use the following formula to calculate the Interrupt Count.

Interrupt Count = (#Tx or Rx Queues) + 2

For example:

Interrupt Count wq = 32, rq = 32, cq = 64 - then Interrupt Count = Max(32, 32) + 2 = 34
Interrupt Count wq = 64, rq = 8, cq = 72 – then Interrupt Count = Max(64, 8) + 2 = 66
Interrupt Count wq = 1, rq = 16, cq = 17 - then Interrupt count = Max(1, 16) + 2 = 18

Interrupt Count in Windows Adapter Policies

For Windows OS, the recommended adapter policy in UCS Manager for VIC 1400 series and above adapters is Win-HPN and if RDMA is used, the recommended policy is Win-HPN-SMB. For VIC 1400 series and above adapters, the recommended interrupt value setting is 512 and the Windows VIC driver takes care of allocating the required number of Interrupts.

For VIC 1300 and VIC 1200 series adapters, the recommended UCS Manager adapter policy is Windows and the Interrupt would be TX + RX + 2, rounded to closest power of 2. The maximum supported Windows queues is 8 for Rx Queues and 1 for Tx Queues.

Example for VIC 1200 and VIC 1300 series adapters:

Tx = 1, Rx = 4, CQ = 5, Interrupt = 8 (1 + 4 rounded to nearest power of 2), Enable RSS

Example for VIC 1400 series and above adapters:

Tx = 1, Rx = 4, CQ = 5, Interrupt = 512 , Enable RSS

NVMe over Fabrics using Fibre Channel

The NVM Express (NVMe) interface allows host software to communicate with a non-volatile memory subsystem. This interface is optimized for Enterprise non-volatile storage, which is typically attached as a register level interface to the PCI Express (PCIe) interface.

NVMe over Fabrics using Fibre Channel (FC-NVMe) defines a mapping protocol for applying the NVMe interface to Fibre Channel. This protocol defines how Fibre Channel services and specified Information Units (IUs) are used to perform the services defined by NVMe over a Fibre Channel fabric. NVMe initiators can access and transfer information to NVMe targets over Fibre Channel.

FC-NVMe combines the advantages of Fibre Channel and NVMe. You get the improved performance of NVMe along with the flexibility and the scalability of the shared storage architecture. Cisco UCS Manager Release 4.0(2) supports NVMe over Fabrics using Fibre Channel on UCS VIC 1400 Series adapters.

Starting with UCS Manager release 4.3(2b), NVMeoF using RDMA is supported on Cisco UCS VIC 14000 series adapters.

Starting with UCS Manager release 4.2(2), NVMeoF using Fibre Channel is supported on Cisco UCS VIC 15000 series adapters.

Cisco UCS Manager provides the recommended FC NVME Initiator adapter policies in the list of pre-configured adapter policies. To create a new FC-NVMe adapter policy, follow the steps in the *Creating a Fibre Channel Adapter Policy* section.

NVMe over Fabrics Using RDMA

NVMe over Fabrics (NVMeoF) is a communication protocol that allows one computer to access NVMe namespaces available on another computer. NVMeoF is similar to NVMe, but differs in the network-related steps involved in using the NVMeoF storage devices. The commands for discovering, connecting, and disconnecting a NVMeoF storage device are integrated into the **nvme** utility provided in Linux..

The NVMeoF fabric that Cisco supports is RDMA over Converged Ethernet version 2 (RoCEv2). RoCEv2 is a fabric protocol that runs over UDP. It requires a no-drop policy.

The eNIC RDMA driver works in conjunction with the eNIC driver, which must be loaded first when configuring NVMeoF.

Cisco UCS Manager provides the default Linux-NVMe-RoCE adapter policy for creating NVMe RoCEv2 interfaces. Do not use the default Linux adapter policy. For complete information on configuring RoCEv2 over NVMeoF, refer to the *Cisco UCS Manager Configuration Guide for RDMA over Converged Ethernet (RoCE) v2*.

NVMeoF using RDMA is supported on M5 B-Series or C-Series Servers with Cisco UCS VIC 1400 Series adapters.

Starting with UCS Manager release 4.3(2b), NVMeOF using RDMA is supported on Cisco UCS VIC 14000 series adapters.

Starting with UCS Manager release 4.2(2), NVMeOF using RDMA is supported on Cisco UCS VIC 15000 series adapters.

Accelerated Receive Flow Steering

Accelerated Receive Flow Steering (ARFS) is hardware-assisted receive flow steering that can increase CPU data cache hit rate by steering kernel level processing of packets to the CPU where the application thread consuming the packet is running.

Using ARFS can improve usage CPU efficiency and reduce network traffic latency. Each receive queue of a CPU has an interrupt associated with it. You can configure the Interrupt Service Routine (ISR) to run on a CPU. The ISR moves the packet from the receive queue to the backlog of one of the current CPUs, which processes the packet later. If the application is not running on this CPU, the CPU must copy the packet to non-local memory, which adds to latency. ARFS can reduce this latency by moving that particular stream to the receive queue of the CPU on which the application is running.

ARFS is disabled by default and can be enabled through Cisco UCS Manager. To configure ARFS, do the following:

1. Create an adapter policy with ARFS enabled.
2. Associate the adapter policy with a service profile.
3. Enable ARFS on a host:
   a. Turn off Interrupt Request Queue (IRQ) balance.
   b. Associate IRQ with different CPUs.
   c. Enable ntuple by using ethtool.

Guidelines and Limitations for Accelerated Receive Flow Steering

- ARFS supports 64 filters per vNIC
- ARFS is supported on the following adapters:
  - Cisco UCS VIC 1200 Series
  - Cisco UCS VIC 1300 Series
  - Cisco UCS VIC 1400 Series
  - Cisco UCS VIC 15000 Series
- ARFS is supported on the following Operating Systems:
  - Red Hat Enterprise Linux 6.5 and higher versions
  - Red Hat Enterprise Linux 7.0 and higher versions
  - SUSE Linux Enterprise Server 11 SP2 and higher versions
  - SUSE Linux Enterprise Server 12 SP1 - SP3
  - SUSE Linux Enterprise Server 15 and higher versions

- Ubuntu 14.04.2 and higher versions

Interrupt Coalescing

Adapters typically generate a large number of interrupts that a host CPU must service. Interrupt coalescing reduces the number of interrupts serviced by the host CPU. This is done by interrupting the host CPU only once for multiple occurrences of the same event over a configurable coalescing interval.

When interrupt coalescing is enabled for receive operations, the adapter continues to receive packets, but the host CPU does not immediately receive an interrupt for each packet. A coalescing timer starts when the first packet is received by the adapter. When the configured coalescing interval times out, the adapter generates one interrupt with the packets received during that interval. The NIC driver on the host then services the multiple packets that are received. Reduction in the number of interrupts generated reduces the time spent by the host CPU on context switches. This means that the CPU has more time to process packets, which results in better throughput and latency.

Adaptive Interrupt Coalescing

Due to the coalescing interval, the handling of received packets adds to latency. For small packets with a low packet rate, this latency increases. To avoid this increase in latency, the driver can adapt to the pattern of traffic flowing through it and adjust the interrupt coalescing interval for a better response from the server.

Adaptive interrupt coalescing (AIC) is most effective in connection-oriented low link utilization scenarios including email server, databases server, and LDAP server. It is not suited for line-rate traffic.

Guidelines and Limitations for Adaptive Interrupt Coalescing

- Adaptive Interrupt Coalescing (AIC) does not provide any reduction in latency when the link utilization is more than 80 percent.
- Enabling AIC disables static coalescing.
- AIC is supported on the following Operating Systems:
  - Red Hat Enterprise Linux 6.4 and higher versions
  - SUSE Linux Enterprise Server 11 SP2 and higher versions
  - XenServer 6.5 and higher versions
  - Ubuntu 14.04.2 and higher versions

PTP Adapter Policy

Precision Time Protocol (PTP) precisely synchronizes the server clock with other devices and peripherals on Linux operating systems. PTP must be set for each adapter, and is only supported on Cisco UCS VIC 15000 Series and later adapters.

Clocks managed by PTP follow a client-worker hierarchy, with workers synchronized to a master client The hierarchy is updated by the best master clock (BMC) algorithm, which runs on every clock. One PTP interface per adapter must be enabled to synchronize it to the grand master clock. After enabling PTP, the host must be rebooted.

The time stamping parameters displayed by **`ethtool -T int_name`** will show a field for PTP Hardware Clock. The value of `PTP Hardware Clock: 0` shows that PTP is enabled for the interface. Otherwise, it will show `PTP Hardware Clock: none`.

Note PTP Adapter Policy is not supported on Cisco UCS VIC 1400 and 14000 series adapters.

RDMA Over Converged Ethernet Overview

Remote Direct Memory Access (RDMA) improves performance by enabling direct data exchange in and out of a server. NVMe over Ethernet (NVMeoF) support for RDMA provides faster access to NVMe namespaces on another computer. RDMA over Converged Ethernet (RoCE) allows direct memory access over an Ethernet network. RoCE is a link layer protocol, and hence, it allows communication between any two hosts in the same Ethernet broadcast domain. RoCE delivers superior performance compared to traditional network socket implementations because of lower latency, lower CPU utilization and higher utilization of network bandwidth. Windows 2012 R2 and later versions use RDMA for accelerating and improving the performance of SMB file sharing and Live Migration.

Cisco UCS Manager supports RoCE for Microsoft SMB Direct. It sends additional configuration information to the adapter while creating or modifying an Ethernet adapter policy. Basic RoCE is also referred to as RoCE version 1 (RoCEv1), and is supported on UCS Manager releases from UCS Manager 2.2(4b) to 4.1(1a).

With Cisco UCS Manager 4.1(1a) and later releases, the RoCEv2 protocol is used.

RDMA Over Converged Ethernet (RoCE) v2

RDMA over Converged Ethernet version 2 (RoCEv2) is an *internet layer* protocol, which means that RoCEv2 packets can be routed. RoCEv2 allows direct memory access over the network by encapsulating an Infiniband (IB) transport packet over Ethernet.

The RoCEv2 protocol exists on top of either the UDP/IPv4 or the UDP/IPv6 protocol. The UDP destination port number 4791 has been reserved for RoCEv2. Since RoCEv2 packets are routable, the RoCEv2 protocol is sometimes called Routable RoCE.

RoCEv2 is supported on the Windows, Linux, and ESXi Operating Systems.

Guidelines for Using SMB Direct support on Windows using RDMA over converged Ethernet (RoCE) v2

General Guidelines and Limitations:

- Cisco UCS Manager release 4.1.x and later releases support Microsoft SMB Direct with RoCEv2 on Microsoft Windows Server 2019 and later. Cisco recommends that you have all KB updates from Microsoft for your Windows Server release.

Note RoCEv2 is not supported on Microsoft Windows Server 2016.

- Cisco recommends you check UCS Hardware and Software Compatibility specific to your UCS Manager release to determine support for Microsoft SMB Direct with RoCEv2 on Microsoft Windows.

- Microsoft SMB Direct with RoCEv2 is supported only with Cisco UCS VIC 1400 Seriesand 15000 Series adapters. It is not supported with UCS VIC 1200 Series and 1300 Series adapters. SMB Direct with RoCEv2 is supported on all UCS Fabric Interconnects.

Note RoCEv1 is not supported with Cisco UCS VIC 1400 Seriesand Cisco UCS VIC 15000 Series.

- RoCEv2 configuration is supported only between Cisco adapters. Interoperability between Cisco adapters and third party adapters is not supported.
- RoCEv2 supports two RoCEv2 enabled vNIC per adapter and four virtual ports per adapter interface, independent of SET switch configuration.
- RoCEv2 cannot be used on the same vNIC interface as NVGRE, NetFlow, and VMQ features.
- RoCEv2 cannot be used with usNIC.
- RoCEv2-enabled vNIC interfaces must have the no-drop QoS system class enabled in UCS Manager.
- The RoCE Properties queue pairs setting must for be a minimum of 4 queue pairs.
- Maximum number of queue pairs per adapter is 2048.
- The QoS No Drop class configuration must be properly configured on upstream switches such as Cisco Nexus 9000 series switches. QoS configurations will vary between different upstream switches.
- The maximum number of memory regions per rNIC interface is 131072.
- UCS Manager does not support fabric failover for vNICs with RoCEv2 enabled.
- SMB Direct with RoCEv2 is supported on both IPv4 and IPv6.
- RoCEv2 cannot be used with GENEVE offload.

MTU Properties:

- In older versions of the VIC driver, the MTU was derived from either a UCS Manager service profile or from the Cisco IMC vNIC MTU setting in standalone mode. This behavior changes on Cisco UCS VIC 1400 Series and later adapters, where MTU is controlled from the Windows OS Jumbo Packet advanced property. A value configured from UCS Manager or Cisco IMC has no effect.
- The RoCEv2 MTU value is always power-of-two and its maximum limit is 4096.
- RoCEv2 MTU is derived from the Ethernet MTU.
- RoCEv2 MTU is the highest power-of-two that is less than the Ethernet MTU. For example:
    - if the Ethernet value is 1500, then the RoCEv2 MTU value is 1024
    - if the Ethernet value is 4096, then the RoCEv2 MTU value is 4096
    - if the Ethernet value is 9000, then the RoCEv2 MTU value is 4096

Windows NDPKI Modes of Operation:

- Cisco's implementation of Network Direct Kernel Provider Interface (NDPKI) supports two modes of operation: Mode 1 and Mode 2. Mode 1 and Mode 2 relate to the implementation of Network Direct Kernel Provider Interface (NDKPI): Mode 1 is native RDMA, and Mode 2 involves configuration for the virtual port with RDMA. Cisco does not support NDPKI Mode 3 operation.
- The recommended default adapter policy for RoCEv2 Mode 1 is Win-HPN-SMBd .
- The recommended default adapter policy for RoCEv2 Mode 2 is MQ-SMBd.
- RoCEv2 enabled vNICs for Mode2 operation require the QoS host control policy set to full.
- Mode 2 is inclusive of Mode 1: Mode 1 must be enabled to operate Mode 2.
- On Windows, the RoCEv2 interface supports MSI & MSIx interrupt modes. By default, it is in MSIx interrupt mode. Cisco recommends you avoid changing interrupt mode when the interface is configured with RoCEv2 properties.

Downgrade Limitations: Cisco recommends you remove the RoCEv2 configuration before downgrading to any non-supported RoCEv2 release. If the configuration is not removed or disabled, downgrade will fail.

Guidelines for using NVMe over Fabrics (NVMeoF) with RoCEv2 on Linux

General Guidelines and Limitations:

- Cisco recommends you check UCS Hardware and Software Compatibility specific to your UCS Manager release to determine support for NVMeoF. NVMeoF is supported on UCS M5 and later B-Series and C-Series servers.
- NVMe over RDMA with RoCEv2 is supported with the fourth generation Cisco UCS VIC 1400 Series and UCS VIC 15000 Series adapters. NVMe over RDMA is not supported on UCS 6324 Fabric Interconnects or on UCS VIC 1200 Series and 1300 Series adapters.
- When creating RoCEv2 interfaces, use Cisco UCS Manager provided Linux-NVMe-RoCE adapter policy.

Note Do not use the default Linux Adapter policy with RoCEv2; RoCEv2 interfaces will not be created in the OS.

- When configuring RoCEv2 interfaces, use both the enic and enic_rdma binary drivers downloaded from Cisco.com and install the matched set of enic and enic_rdma drivers. Attempting to use the binary enic_rdma driver downloaded from Cisco.com with an inbox enic driver will not work.
- RoCEv2 supports maximum two RoCEv2 enabled interfaces per adapter.
- Booting from an NVMeoF namespace is not supported.
- Layer 3 routing is not supported.
- RoCEv2 does not support bonding.
- Saving a crashdump to an NVMeoF namespace during a system crash is not supported.
- NVMeoF cannot be used with usNIC, VMFEX, VxLAN, VMQ, VMMQ, NVGRE, GENEVE Offload, and DPDK features.
- Netflow monitoring is not supported on RoCEv2 interfaces.

- In the Linux-NVMe-RoCE policy, do not change values of Queue Pairs, Memory Regions, Resource Groups, and Priority settings other than to Cisco provided default values. NVMeoF functionality may not be guaranteed with different settings for Queue Pairs, Memory Regions, Resource Groups, and Priority.
- The QoS no drop class configuration must be properly configured on upstream switches such as Cisco Nexus 9000 series switches. QoS configurations will vary between different upstream switches.
- Set MTU size correctly on the VLANs and QoS policy on upstream switches.
- Spanning Tree Protocol (STP) may cause temporary loss of network connectivity when a failover or failback event occurs. To prevent this issue from occurring, disable STP on uplink switches.
- UCS Manager does not support fabric failover for vNICs with RoCEv2 enabled.

Interrupts

- Linux RoCEv2 interface supports only MSIx interrupt mode. Cisco recommends avoiding changing interrupt mode when the interface is configured with RoCEv2 properties.
- The minimum interrupt count for using RoCEv2 with Linux is 8.

Downgrade Limitations:

- Cisco recommends you remove the RoCEv2 configuration before downgrading to any non-supported RoCEv2 release.

Guidelines for using RoCEv2 Protocol in the Native ENIC driver on ESXi

General Guidelines and Limitations:

- Cisco UCS Manager release 4.2(3b) supports RoCEv2 only on ESXi 7.0 U3.
- Cisco recommends you check UCS Hardware and Software Compatibility specific to your UCS Manager release to determine support for ESXi. RoCEv2 on ESXi is supported on UCS B-Series and C-Series servers with Cisco UCS VIC 15000 Series and later adapters.
- RoCEv2 on ESXi is not supported on UCS VIC 1200, 1300 and 1400 Series adapters.
- RDMA on ESXi nENIC currently supports only ESXi NVME that is part of the ESXi kernel. The current implementation does not support the ESXi user space RDMA application.
- Multiple mac addresses and multiple VLANs are supported only on VIC 15000 Series adapters.
- RoCEv2 supports maximum two RoCEv2 enabled interfaces per adapter.
- Pvrdma, VSAN over RDMA, and iSER are not supported.
- The COS setting is not supported on UCS Manager.

Downgrade Limitations:

- Cisco recommends you remove the RoCEv2 configuration before downgrading to any non-supported RoCEv2 release.

GENEVE Offload

Cisco UCS Manager now supports Generic Network Virtualization Encapsulation (GENEVE) Offload on the ESXi platform, which allows essentially any information to be encoded in a packet and passed between tunnel endpoints. GENEVE provides the overlay capability to create isolated, multi-tenant broadcast domains across data center fabrics on UCS VIC 1400 Series and VIC 15000 Series adapters. Using the GENEVE protocol allows you to create logical networks that span physical network boundaries.

GENEVE offload is present in all Ethernet adapter policies and is disabled by default.

Refer to the NSX-T documentation for how to implement GENEVE offload end to end configuration.

Cisco recommends configuring the following values in the Ethernet adapter policy when GENEVE offload is enabled:

- Transmit Queues: 1
- TX Ring Size: 4096
- Receive Queues: 8
- RX Ring Size: 4096
- Completion Queues: 16
- Interrupts: 32

The following features are not supported when GENEVE offload is enabled on any interface:

- Azure Stack QoS
- RoCEv2

GENEVE offload-enabled interfaces do not support usNIC, Netflow, advanced filters, NetQueue, or aRFS.

Note There are exceptions for the Cisco UCS VIC 15000 series adapter. GENEVE offload-enabled interfaces on the UCS VIC 15000 series adapter also support advanced filters, NetQueue, and ARFS.

Other limitations with GENEVE offload include:

- External outer IPV6 is NOT supported with GENEVE offload.
- GENEVE offload is supported only with Cisco UCS VIC1400, and VIC 15000 series adapters. It is not supported on Cisco UCS VIC 1300 Series and 1200 Series adapters.
- GENEVE offload is supported with ESX 7.0 (NSX-T 3.0) and ESX 6.7U3(NSX-T 2.5).
- Cisco recommends that you remove the GENEVE offload configuration before downgrading to any non-supported release.

Creating an Ethernet Adapter Policy

Tip If the fields in an area do not display, click the **Expand** icon to the right of the heading.

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 multi tenancy, expand the **root** node.

Step 4 Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.

Step 5 Enter a **Name** and optional **Description** for the policy.

This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.

Step 6 (Optional) In the **Resources** area, adjust the following values:

Name	Description
Pooled radio button	Whether the queue resources are pooled or not. • **Disabled**—Pooling is disabled. • **Enabled**—Pooling is enabled. When pooling is enabled, the counts of queue resources specified in the Adapter Policy will be the total number of queues allocated across all vPorts.
Transmit Queues field	The number of transmit queue resources to allocate. Enter an integer between 1 and 1000.
Ring Size field	The number of descriptors in each transmit queue. Enter an integer between 64 and 16384. Cisco UCS VIC 1400 series adapters support maximum 4K (4096) Ring Size. Cisco UCS VIC 15000 Series and above adapters support up to 16K (16384) Ring Size.
Receive Queues field	The number of receive queue resources to allocate. Enter an integer between 1 and 1000.

Name	Description
Ring Size field	The number of descriptors in each receive queue. Enter an integer between 64 and 16384. Cisco UCS VIC 1400 series adapters support maximum 4K (4096) Ring Size. Cisco UCS VIC 15000 Series and above adapters support up to 16K (16384) Ring Size.
Completion Queues field	The number of completion queue resources to allocate. In general, the number of completion queue resources you should allocate is equal to the number of transmit queue resources plus the number of receive queue resources. Enter an integer between 1 and 2000.
Interrupts field	The number of interrupt resources to allocate. In general, this value should be equal to (Completion Queues + 2) rounded up to nearest power of 2. Enter an integer between 1 and 1024. For example, if Transmit Queues = 1 and Receive Queues = 8 then: • Completion Queues = 1 + 8 = 9 • Interrupt Count = (9 + 2) rounded up to the nearest power of 2 = 16

Step 7 (Optional) In the **Options** area, adjust the following values:

Note The RoCE Version 2 Option should be used with UCS Manager 4.2.1 and later releases.

Name	Description
Transmit Checksum Offload radio button	This can be one of the following: • **Disabled**—The CPU calculates all packet checksums. • **Enabled**—The CPU sends all packets to the hardware so that the checksum can be calculated. This option may reduce CPU overhead. **Note** This option affects only packets sent from the interface.
Receive Checksum Offload radio button	This can be one of the following: • **Disabled**—The CPU validates all packet checksums. • **Enabled**—The CPU sends all packet checksums to the hardware for validation. This option may reduce CPU overhead. **Note** This option affects only packets received by the interface.

Name	Description
TCP Segmentation Offload radio button	This can be one of the following: • **Disabled**—The CPU segments large TCP packets. • **Enabled**—The CPU sends large TCP packets to the hardware to be segmented. This option may reduce CPU overhead and increase throughput rate. **Note** This option is also known as Large Send Offload (LSO) and affects only packets sent from the interface.
TCP Large Receive Offload radio button	This can be one of the following: • **Disabled**—The CPU processes all large packets. • **Enabled**—The hardware reassembles all segmented packets before sending them to the CPU. This option may reduce CPU utilization and increase inbound throughput. **Note** This option affects only packets received by the interface.
Receive Side Scaling radio button	RSS distributes network receive processing across multiple CPUs in multiprocessor systems. This can be one of the following: • **Disabled**—Network receive processing is always handled by a single processor even if additional processors are available. • **Enabled**—Network receive processing is shared across processors whenever possible.
Accelerated Receive Flow Steering radio button	Packet processing for a flow must be performed on the local CPU. This is supported for Linux operating systems only. This can be one of the following: • **Disabled**—The CPU is not specified. • **Enabled**—Packet processing is performed on the local CPU.
Network Virtualization using Generic Routing Encapsulation radio button	Whether NVGRE overlay hardware offloads for TSO and checksum are enabled. This can be one of the following: • **Disabled**—NVGRE overlay hardware offloads are not enabled. • **Enabled**—NVGRE overlay hardware offloads are enabled. NVGRE overlay hardware offloads can be enabled when using UCS VIC 1400 Series adapters.

Name	Description
Virtual Extensible LANradio button	Whether VXLAN overlay hardware offloads for TSO and checksum are enabled. This can be one of the following: • **Disabled**—VXLAN overlay hardware offloads are not enabled. • **Enabled**—VXLAN overlay hardware offloads are enabled. VXLAN overlay hardware offloads can be enabled with RoCE and VMQ when using UCS VIC 1400 Series adapters. VXLAN overlay hardware offloads can be enabled with RoCEv2 and VMQ when using Cisco UCS VIC 1400 or VIC 15000 Series adapters
GENEVE radio button	Whether Generic Network Virtualization Encapsulation (GENEVE) overlay hardware offloads are enabled. GENEVE offload provides the overlay capability to create isolated, multi-tenant broadcast domains across data center fabrics on VIC 1400 and VIC 15000 series adapters. This can be one of the following: • **Disabled**—GENEVE overlay offloads are not enabled. • **Enabled**—GENEVE overlay offloads are enabled.
AzureStack-Host QoS radio button	Enable this feature to successfully deploy Azure Stack based solutions with RDMA enabled. • **Enabled**—Enabling AzureStack-Host QoS on an adapter allows the user to carve out traffic classes for RDMA traffic and ensure a desired portion of the bandwidth is allocated to it. • **Disabled**—Disables the AzureStack-Host QoS feature on the adapter.
Failback Timeout field	After a vNIC has started using its secondary interface, this setting controls how long the primary interface must be available before the system resumes using the primary interface for the vNIC. Enter a number of seconds between 0 and 600.

Name	Description
Interrupt Mode radio button	The preferred driver interrupt mode. This can be one of the following: • **MSI X**—Message Signaled Interrupts (MSI) with the optional extension. This is the recommended option. **Note** If you set **Interrupt Mode** as **Msi-X**, and if **pci=nomsi** parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=nomsi** would block the eNIC/fNIC driver to run in the Msi-X mode, impacting system performance. • **MSI**—MSI only. • **IN Tx**—PCI IN Tx interrupts. **Note** INTx interrupt mode is not supported with the ESX nenic driver and Windows nenic driver. MSI interrupt mode on Fibre Channel interfaces is not supported. If the MSI interrupt mode is configured for Fibre Channel interface, Fibre Channel interfaces will come up in MSIx mode.
Interrupt Coalescing Type radio button	This can be one of the following: • **Min**—The system waits for the time specified in the **Interrupt Timer** field before sending another interrupt event. • **Idle**—The system does not send an interrupt until there is a period of no activity lasting as least as long as the time specified in the **Interrupt Timer** field.
Interrupt Timer field	The time to wait between interrupts or the idle period that must be encountered before an interrupt is sent. Enter a value between 1 and 65535. To turn off interrupt coalescing, enter 0 (zero) in this field.
RoCE radio button	Whether Remote Direct Memory Access over an Ethernet network is enabled. This can be one of the following: • **Disabled**—RoCE is disabled on the Ethernet adapter. • **Enabled**—RoCE is enabled on the Ethernet adapter.
RoCE Properties area	Lists the RoCE properties. This area is enabled only if you enable RoCE.

Name	Description
Version 1 radio button	RoCE Version 1 is a link layer protocol. It allows communication between any two hosts in the same Ethernet broadcast domain. Whether RoCE Version 1 is enabled. This can be one of the following: • **Disabled**—RoCE version 1 is disabled on the Ethernet adapter. • **Enabled**—RoCE version 1 is enabled on the Ethernet adapter.
Version 2 radio button	RoCEv2 is an internet layer protocol. RoCEv2 packets can be routed. This is possible because RoCEv2 packets now include an IP and UDP header. Whether RoCE Version 2 is enabled. This can be one of the following: • **Disabled**—RoCE version 2 is disabled on the Ethernet adapter. • **Enabled**—RoCE version 2 is enabled on the Ethernet adapter. If you enable RoCE version 2, you can also set the **Priority** field.
Queue Pairs field	The number of queue pairs per adapter. Enter an integer between 1 and 8192. It is recommended that this number be an integer power of 2.
Memory Regions field	The number of memory regions per adapter. Enter an integer between 1 and 524288. It is recommended that this number be an integer power of 2.
Resource Groups field	The number of resource groups per adapter. Enter an integer between 1 and 128. It is recommended that this number be an integer power of 2 greater than or equal to the number of CPU cores on the system for optimum performance.
Priority field	Priority is set as **Platinum** by default. The supported values include: • Fibre Channel • Best Effort • Bronze • Silver • Gold • Platinum For RoCE version 2, set **Priority** as **Platinum**.

Name	Description
Advance Filter radio button	Whether Advance Filter over an Ethernet network is enabled. This can be one of the following: • **Disabled**—Advance filter is disabled on the Ethernet adapter. • **Enabled**—Advance filter is enabled on the Ethernet adapter.
Interrupt Scaling radio button	Whether Interrupt Scaling over an Ethernet network is enabled. This can be one of the following: • **Disabled**—Interrupt Scaling is disabled on the Ethernet adapter. • **Enabled**—Interrupt Scaling is enabled on the Ethernet adapter.
Adapter PTP radio button	Whether Precision Time Protocol is implemented for adapters on an Ethernet network. PTP is only available on Linux operating systems and can only be enabled for UCS VIC 15000 series and later adapters. • **Disabled**—PTP is disabled on the Ethernet adapter. • **Enabled**—PTP is enabled on the Ethernet adapter.

Step 8 Click **OK**.

Step 9 If a confirmation dialog box displays, click **Yes**.

Receive Side Scaling (RSS)

Configuring an Ethernet Adapter Policy to Enable RSS on Windows Operating Systems

To enable Receive Side Scaling (RSS) and configure an Ethernet Adapter Policy, do the following:

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 multi tenancy, expand the root node.

Step 4 Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.

Step 5 Enter a **Name** and optional **Description** for the policy. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.

Step 6 Create an Ethernet adapter policy.

In the **Resources** area, use the following values when creating the Ethernet adapter policy:

Name	Description
Pooled radio button	Whether the queue resources are pooled or not. • **Disabled**—Pooling is disabled. • **Enabled**—Pooling is enabled. When pooling is enabled, the counts of queue resources specified in the Adapter Policy will be the total number of queues allocated across all vPorts.
Transmit Queues field	The number of transmit queue resources to allocate. Enter an integer between 1 and 1000.
Ring Size field	The number of descriptors in each transmit queue. Enter an integer between 64 and 16384. Cisco UCS VIC 1400 Series and older adapters support maximum 4K (4096) Ring Size. Cisco UCS VIC 15000 Series and above adapters support up to 16K (16384) Ring Size.
Receive Queues field	The number of receive queue resources to allocate. Enter an integer between 1 and 1000.
Ring Size field	The number of descriptors in each receive queue. Enter an integer between 64 and 16384. Cisco UCS VIC 1400 Series and older adapters support maximum 4K (4096) Ring Size. Cisco UCS VIC 15000 Series and above adapters support up to 16K (16384) Ring Size.
Completion Queues field	The number of completion queue resources to allocate. In general, the number of completion queue resources you should allocate is equal to the number of transmit queue resources plus the number of receive queue resources.
Interrupts field	The number of interrupt resources to allocate. In general, this value should be equal to (Completion Queues + 2) rounded up to nearest power of 2.

In the **Options** area, use the following values:

Name	Description
Transmit Checksum Offload radio button	This can be one of the following: • **Disabled**—The CPU calculates all packet checksums. • **Enabled**—The CPU sends all packets to the hardware so that the checksum can be calculated. This option may reduce CPU overhead.
Receive Checksum Offload radio button	This can be one of the following: • **Disabled**—The CPU validates all packet checksums. • **Enabled**—The CPU sends all packet checksums to the hardware for validation. This option may reduce CPU overhead.
TCP Segmentation Offload radio button	This can be one of the following: • **Disabled**—The CPU segments large TCP packets. • **Enabled**—The CPU sends large TCP packets to the hardware to be segmented. This option may reduce CPU overhead and increase throughput rate.
TCP Large Receive Offload radio button	This can be one of the following: • **Disabled**—The CPU processes all large packets. • **Enabled**—The hardware reassembles all segmented packets before sending them to the CPU. This option may reduce CPU utilization and increase inbound throughput.
Receive Side Scaling radio button	RSS distributes network receive processing across multiple CPUs in multiprocessor systems. This can be one of the following: • **Disabled**—Network receive processing is always handled by a single processor even if additional processors are available. • **Enabled**—Network receive processing is shared across processors whenever possible. Click **Enabled** to support RSS .
Accelerated Receive Flow Steering radio button	Packet processing for a flow must be performed on the local CPU. This is supported for Linux operating systems only. This can be one of the following: • **Disabled**—The CPU is not specified. • **Enabled**—Packet processing is performed on the local CPU.

Name	Description
Network Virtualization using Generic Routing Encapsulation radio button	Whether NVGRE overlay hardware offloads for TSO and checksum are enabled. This can be one of the following: • **Disabled**—NVGRE overlay hardware offloads are not enabled. • **Enabled**—NVGRE overlay hardware offloads are enabled. NVGRE overlay hardware offloads can be enabled when using UCS VIC 1400 Series adapters.
Virtual Extensible LAN radio button	Whether VXLAN overlay hardware offloads for TSO and checksum are enabled. This can be one of the following: • **Disabled**—VXLAN overlay hardware offloads are not enabled. • **Enabled**—VXLAN overlay hardware offloads are enabled. VXLAN overlay hardware offloads can be enabled with RoCE and VMQ when using UCS VIC 1400 Series adapters. VXLAN overlay hardware offloads can be enabled with RoCEv2 and VMQ when using Cisco UCS VIC 1400 or VIC 15000 Series adapters
GENEVE	Whether Generic Network Virtualization Encapsulation (GENEVE) overlay hardware offloads are enabled. GENEVE offload provides the overlay capability to create isolated, multi-tenant broadcast domains across data center fabrics on Cisco UCS VIC 1400 or VIC 15000 series adapters. This can be one of the following: • **Disabled**—GENEVE overlay offloads are not enabled. • **Enabled**—GENEVE overlay offloads are enabled.
AzureStack-Host QoS	Enable this feature to successfully deploy Azure Stack based solutions with RDMA enabled. • **Enabled**—Enabling AzureStack-Host QoS on an adapter allows the user to carve out traffic classes for RDMA traffic and ensure a desired portion of the bandwidth is allocated to it. • **Disabled**—Disables the AzureStack-Host QoS feature on the adapter.
Failback Timeout field	After a vNIC has started using its secondary interface, this setting controls how long the primary interface must be available before the system resumes using the primary interface for the vNIC. Enter a number of seconds between 0 and 600.

Name	Description
Interrupt Mode radio button	The preferred driver interrupt mode. • **MSI X**—Message Signaled Interrupts (MSI) with the optional extension. RSS is supported only on *MSI X*. • **MSI**—MSI only. • **IN Tx**—PCI IN Tx interrupts.
Interrupt Coalescing Type radio button	This can be one of the following: • **Min**—The system waits for the time specified in the **Interrupt Timer** field before sending another interrupt event. • **Idle**—The system does not send an interrupt until there is a period of no activity lasting as least as long as the time specified in the **Interrupt Timer** field.
Interrupt Timer field	The time to wait between interrupts or the idle period that must be encountered before an interrupt is sent. Enter a value between 1 and 65535. To turn off interrupt coalescing, enter 0 (zero) in this field.
RoCE radio button	Whether Remote Direct Memory Access over an Ethernet network is enabled. This can be one of the following: • **Disabled**—RoCE is disabled on the Ethernet adapter. • **Enabled**—RoCE is enabled on the Ethernet adapter.
RoCE Properties area	Lists the RoCE properties. This area is enabled only if you enable RoCE.
Version 1 radio button	RoCE Version 1 is a link layer protocol. It allows communication between any two hosts in the same Ethernet broadcast domain. Whether RoCE Version 1 is enabled. This can be one of the following: • **Disabled**—RoCE version 1 is disabled on the Ethernet adapter. • **Enabled**—RoCE version 1 is enabled on the Ethernet adapter.
Version 2 radio button	RoCEv2 is an internet layer protocol. RoCEv2 packets can be routed. This is possible because RoCEv2 packets now include an IP and UDP header. Whether RoCE Version 2 is enabled. This can be one of the following: • **Disabled**—RoCE version 2 is disabled on the Ethernet adapter. • **Enabled**—RoCE version 2 is enabled on the Ethernet adapter. If you enable RoCE version 2, you can also set the **Priority** field.

Name	Description
Queue Pairs field	The number of queue pairs per adapter. Enter an integer between 1 and 8192. It is recommended that this number be an integer power of 2.
Priority drop-down list	Pre-defined set of Global (system wide) QoS classes. These are: • Fibre Channel • Best Effort • Bronze • Silver • Gold • Platinum For RoCE version 2, set **Priority** as **Platinum**.
Memory Regions field	The number of memory regions per adapter. Enter an integer between 1 and 524288. It is recommended that this number be an integer power of 2.
Resource Groups field	The number of resource groups per adapter. Enter an integer between 1 and 128. It is recommended that this number be an integer power of 2 greater than or equal to the number of CPU cores on the system for optimum performance.
Advance Filter radio button	Whether Advance Filter over an Ethernet network is enabled. This can be one of the following: • **Disabled**—Advance filter is disabled on the Ethernet adapter. • **Enabled**—Advance filter is enabled on the Ethernet adapter.
Interrupt Scaling radio button	Whether Interrupt Scaling over an Ethernet network is enabled. This can be one of the following: • **Disabled**—Interrupt Scaling is disabled on the Ethernet adapter. • **Enabled**—Interrupt Scaling is enabled on the Ethernet adapter.
Adapter PTP radio button	Whether Precision Time Protocol is implemented for adapters on an Ethernet network. PTP is only available on Linux operating systems and can only be enabled for Cisco UCS VIC 15000 series and later adapters. • **Disabled**—PTP is disabled on the Ethernet adapter. • **Enabled**—PTP is enabled on the Ethernet adapter.

Click **OK**. Click **Yes** to confirm the Ethernet Adapter Policy creation.

Step 7 Install RSS capable latest NENIC driver version.

Note For more information, see Cisco UCS Virtual Interface Card Drivers Installation Guide.

Step 8 Reboot the server.

Configuring an Ethernet Adapter Policy to Enable eNIC Support for RSS on VMware ESXi

Cisco UCS Manager includes eNIC support for the Receive Side Scaling (RSS) feature on ESXi 5.5 and later releases.

Procedure

Step 1 Create an Ethernet adapter policy.

Use the following parameters when creating the Ethernet adapter policy.

In the **Resources** area, set the following options:

- Transmit Queues = 1
- Receive Queues = n (up to 16)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = (# Completion Queues +2) rounded up to the nearest power of 2

In the **Options** area, set the following option:

- Receive Side Scaling (RSS) = Enabled

Step 2 Install the appropriate drivers according to the UCS Hardware and Software Compatibility.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 3 Reboot the server.

Configuring an Ethernet Adapter Policy to Enable eNIC Support for MRQS on Linux Operating Systems

Cisco UCS Manager includes eNIC support for the Multiple Receive Queue Support (MRQS) feature on Red Hat Enterprise Linux Version 6.x and SUSE Linux Enterprise Server Version 11.x.

Procedure

Step 1 Create an Ethernet adapter policy.

Use the following parameters when creating the Ethernet adapter policy:

- Transmit Queues = 1
- Receive Queues = n (up to 8)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2
- Receive Side Scaling (RSS) = Enabled
- Interrupt Mode = Msi-X

Note If you set **Interrupt Mode** as **Msi-X**, and if **pci=nomsi** parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=nomsi** would block the eNIC/fNIC driver to run in the **Msi-X** mode, impacting system performance.

Step 2 Install an eNIC driver Version 2.1.1.35 or later.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 3 Reboot the server.

Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with NVGRE

Cisco UCS Manager supports stateless offloads with NVGRE on Cisco UCS VIC 1300 Series adapters that are installed on servers running on Windows Server 2012 R2 operating systems and higher versions. NVGRE feature is also supported on servers with Cisco UCS VIC 1400 Series and Cisco UCS 15000 Series adapters running on Windows Server 2016. Stateless offloads with NVGRE cannot be used with Netflow, usNIC, or VM-FEX.

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 multi tenancy, expand the **root** node.

Step 4 Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.

a) In the **Resources** area, set the following options:

- Transmit Queues = 1
- Receive Queues = n (up to 8)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2

b) In the **Options** area, set the following options:

- Network Virtualization using Generic Routing Encapsulation = Enabled
- Interrupt Mode = Msi-X

Note If you set **Interrupt Mode** as **Msi-X**, and if **pci=nomsi** parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=nomsi** would block the eNIC/fNIC driver to run in the **Msi-X** mode, impacting system performance.

For more information on creating an Ethernet adapter policy, see Creating an Ethernet Adapter Policy, on page 19.

Step 5 Click **OK** to create the Ethernet adapter policy.

Step 6 Install an eNIC driver Version 3.0.0.8 or later.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 7 Reboot the server.

Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with VXLAN

Cisco UCS Manager supports VXLAN TSO and checksum offloads only with Cisco UCS VIC 1300 Series adapters that are running on ESXi 5.5 and later releases.

VXLAN with Receive Side-Scaling (RSS) support starts with the Cisco UCS Manager 3.1(2) release. RSS is supported with VXLAN stateless offload on VIC adapters 1300 Series and SIOC on Cisco UCS S3260 system for ESXi 5.5 and later releases.

Cisco UCS Manager 4.0(1a) Release introduces VXLAN support on servers with Cisco UCS VIC 1400 Series running ESXi 6.5 and later releases. Stateless offloads with VXLAN cannot be used with NetFlow, usNIC, VM-FEX, or Netqueue.

VXLAN support for Linux and Windows 2016 starts with Cisco UCS Manager 4.0(1a) for VIC 1400 Series adapters.

VXLAN is now supported on Cisco UCS VIC 1300, 1400, and 15000 Series adapters.

The maximum amount of receive queues may be up to 16 for Cisco UCS VIC 1300 Series and Cisco UCS 1400 Series adapters on ESXi.

Cisco UCS VIC 15000 Series adapters also support up to 16 receive queues on ESXi.

Note VXLAN stateless hardware offloads are not supported with Guest OS TCP traffic over IPv6 on UCS VIC 1300 Series adapters. However, Cisco UCS VIC 1400, and 15000 series adapters do not have this VxLAN offload limitation.

- To run VXLAN encapsulated TCP traffic over IPV6, disable the VXLAN stateless offloads feature.
- To disable the VXLAN stateless offload feature in UCS Manager, disable the Virtual Extensible LAN field in the Ethernet Adapter Policy.

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 multi tenancy, expand the **root** node.

Step 4 Right-click **Adapter Policies** and choose **Create Ethernet Adapter Policy**.

a) In the **Resources** area, set the following options:

- Transmit Queues = 1
- Receive Queues = n (up to 16)
- Completion Queues = # of Transmit Queues + # of Receive Queues
- Interrupts = # Completion Queues + 2

b) In the **Options** area, set the following options:

- Receive Side Scaling = Enabled
- Virtual Extensible LAN = Enabled
- Interrupt Mode = Msi-X

Note If you set **Interrupt Mode** as **Msi-X**, and if **pci=nomsi** parameter is enabled in `/boot/grub/grub.conf` on RHEL system, then **pci=nomsi** would block the eNIC/fNIC driver to run in the **Msi-X** mode, impacting system performance.

For more information on creating an ethernet adapter policy, see Creating an Ethernet Adapter Policy, on page 19.

Step 5 Click **OK** to create the Ethernet adapter policy.

Step 6 Install an eNIC driver Version 2.1.2.59 or later.

For more information, see the *Cisco UCS Virtual Interface Card Drivers Installation Guide*.

Step 7 Reboot the server.

Deleting an Ethernet Adapter Policy

Procedure

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

Step 2 Expand **LAN** > **Policies** > ***Organization_Name***.

Step 3 Expand the **Adapter Policies** node.

Step 4 Right-click the Ethernet adapter policy that you want to delete and choose **Delete**.

Step 5 If a confirmation dialog box displays, click **Yes**.

Configuring the Default vNIC Behavior Policy

Default vNIC Behavior Policy

Default vNIC behavior policy allows you to configure how vNICs are created for a service profile. You can choose to create vNICS manually, or you can create them automatically.

You can configure the default vNIC behavior policy to define how vNICs are created. This can be one of the following:

- **None**—Cisco UCS Manager does not create default vNICs for a service profile. All vNICs must be explicitly created.
- **HW Inherit**—If a service profile requires vNICs and none have been explicitly defined, Cisco UCS Manager creates the required vNICs based on the adapter installed in the server associated with the service profile.

Note If you do not specify a default behavior policy for vNICs, **HW Inherit** is used by default.

Configuring a Default vNIC Behavior Policy

Procedure

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

Step 2 Expand **LAN** > **Policies**.

Step 3 Expand the **root** node.

You can configure only the default vNIC behavior policy in the root organization. You cannot configure the default vNIC behavior policy in a sub-organization.

Step 4 Click **Default vNIC Behavior**.

Step 5 On the **General Tab**, in the **Properties** area, click one of the following radio buttons in the **Action** field:

- **None**—Cisco UCS Manager does not create default vNICs for a service profile. All vNICs must be explicitly created.
- **HW Inherit**—If a service profile requires vNICs and none have been explicitly defined, Cisco UCS Manager creates the required vNICs based on the adapter installed in the server associated with the service profile.

Step 6 Click **Save Changes**.

Configuring LAN Connectivity Policies

About the LAN and SAN Connectivity Policies

Connectivity policies determine the connections and the network communication resources between the server and the LAN or SAN on the network. These policies use pools to assign MAC addresses, WWNs, and WWPNs to servers and to identify the vNICs and vHBAs that the servers use to communicate with the network.

Note We do not recommend that you use static IDs in connectivity policies, because these policies are included in service profiles and service profile templates and can be used to configure multiple servers.

Privileges Required for LAN and SAN Connectivity Policies

Connectivity policies enable users without network or storage privileges to create and modify service profiles and service profile templates with network and storage connections. However, users must have the appropriate network and storage privileges to create connectivity policies.

Privileges Required to Create Connectivity Policies

Connectivity policies require the same privileges as other network and storage configurations. For example, you must have at least one of the following privileges to create connectivity policies:

- admin—Can create LAN and SAN connectivity policies
- ls-server—Can create LAN and SAN connectivity policies
- ls-network—Can create LAN connectivity policies
- ls-storage—Can create SAN connectivity policies

Privileges Required to Add Connectivity Policies to Service Profiles

After the connectivity policies have been created, a user with ls-compute privileges can include them in a service profile or service profile template. However, a user with only ls-compute privileges cannot create connectivity policies.

Interactions between Service Profiles and Connectivity Policies

You can configure the LAN and SAN connectivity for a service profile through either of the following methods:

- LAN and SAN connectivity policies that are referenced in the service profile
- Local vNICs and vHBAs that are created in the service profile
- Local vNICs and a SAN connectivity policy

- Local vHBAs and a LAN connectivity policy

Cisco UCS maintains mutual exclusivity between connectivity policies and local vNIC and vHBA configuration in the service profile. You cannot have a combination of connectivity policies and locally created vNICs or vHBAs. When you include a LAN connectivity policy in a service profile, all existing vNIC configuration is erased, and when you include a SAN connectivity policy, all existing vHBA configuration in that service profile is erased.

Creating a LAN Connectivity Policy

Procedure

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

Step 2 Expand **LAN** > **Policies**.

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

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

Step 4 Right-click **LAN Connectivity Policies** and choose **Create LAN Connectivity Policy**.

Step 5 In the **Create LAN Connectivity Policy** dialog box, enter a name and optional description.

Step 6 Do one of the following:

- To add vNICs to the LAN connectivity policy, continue with Step 7.
- To add iSCSI vNICs to the LAN connectivity policy and use iSCSI boot with the server, continue with Step 8.

Step 7 To add vNICs, click **Add** next to the plus sign and complete the following fields in the **Create vNIC** dialog box:

a) In the **Create vNIC** dialog box, enter the name, select a **MAC Address Assignment**, and check the **Use vNIC Template** check box to use an existing vNIC template.

You can also create a MAC pool from this area.

b) Choose the **Fabric ID**, select the **VLANs** that you want to use, enter the **MTU**, and choose a **Pin Group**.

Note Cisco recommends using the native VLAN 1 setting to prevent traffic interruptions if using the Cisco Nexus 1000V Series Switches because changing the native VLAN 1 setting on a vNIC causes the port to turn on and off. You can only change the native VLAN setting on a Virtual Private Cloud (VPC) secondary port, and then change the primary port on the VPC.

c) In the **Operational Parameters** area, choose a **Stats Threshold Policy**.

d) In the Adapter Performance Profile area, choose an **Adapter Policy**, **QoS Policy**, and a **Network Control Policy**.

You can also create an Ethernet adapter policy, QoS policy, and network control policy from this area.

e) In the Connection Policies area, choose the **Dynamic vNIC**, **usNIC** or **VMQ** radio button, then choose the corresponding policy.

You can also create a dynamic vNIC, usNIC, or VMQ connection policy from this area.

Note Cisco UCS 6400 Series Fabric Interconnect and Cisco UCS 6536 Fabric Interconnects do not support dynamic vNICs.

f) Click **OK**.

Step 8 If you want to use iSCSI boot with the server, click the down arrows to expand the **Add iSCSI vNICs** bar and do the following:

a) Click **Add** on the table icon bar.

b) In the **Create iSCSI vNIC** dialog box, enter the **Name** and choose the **Overlay vNIC**, **iSCSI Adapter Policy**, and **VLAN**.

You can also create an iSCSI adapter policy from this area.

Note For the Cisco UCS M81KR Virtual Interface Card and the Cisco UCS VIC-1240 Virtual Interface Card, the VLAN that you specify must be the same as the native VLAN on the overlay vNIC.

For the Cisco UCS M51KR-B Broadcom BCM57711 Adapter, the VLAN that you specify can be any VLAN assigned to the overlay vNIC.

c) In the **MAC Address Assignment** drop-down list in the **iSCSI MAC Address** area, choose one of the following:

- Leave the MAC address unassigned, select **Select (None used by default)**. Select this option if the server that will be associated with this service profile contains a Cisco UCS M81KR Virtual Interface Card adapter or a Cisco UCS VIC-1240 Virtual Interface Card.

  Important If the server that will be associated with this service profile contains a Cisco UCS NIC M51KR-B adapter, you must specify a MAC address.

- A specific MAC address, select **00:25:B5:XX:XX:XX** and enter the address in the **MAC Address** field. To verify that this address is available, click the corresponding link.
- A MAC address from a pool, select the pool name from the list. Each pool name is followed by a pair of numbers in parentheses. The first number is the number of available MAC addresses in the pool and the second is the total number of MAC addresses in the pool.

  If this Cisco UCS domain is registered with Cisco UCS Central, there might be two pool categories. **Domain Pools** are defined locally in the Cisco UCS domain and **Global Pools** are defined in Cisco UCS Central.

d) (Optional) If you want to create a MAC pool that will be available to all service profiles, click **Create MAC Pool** and complete the fields in the **Create MAC Pool** wizard.

For more information, see the *UCS Manager Storage Management Guide*, Pools chapter, Creating a MAC Pool topic.

e) Click **OK**.

Step 9 After you have created all the vNICs or iSCSI vNICs you need for the policy, click **OK**.

What to do next

Include the policy in a service profile or service profile template.

Deleting a LAN Connectivity Policy

If you delete a LAN connectivity policy that is included in a service profile, it also deletes all vNICs and iSCSI vNICs from that service profile, and disrupt LAN data traffic for the server associated with the service profile.

Procedure

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

Step 2 Expand **LAN** > **Policies** > *Organization_Name*.

Step 3 Expand the **LAN Connectivity Policies** node.

Step 4 Right-click the policy that you want to delete and choose **Delete**.

Step 5 If a confirmation dialog box displays, click **Yes**.

Creating a vNIC for a LAN Connectivity Policy

Procedure

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

Step 2 Expand **LAN** > **Policies** > *Organization_Name*.

Step 3 Expand the **LAN Connectivity Policies** node.

Step 4 Choose the policy to which you want to add a vNIC.

Step 5 In the **Work** pane, click the **General** tab.

Step 6 On the icon bar of the **vNICs** table, click **Add**.

Step 7 In the **Create vNIC** dialog box, enter the name, select a **MAC Address Assignment**, and check the **Use vNIC Template** check box if you want to use an existing vNIC template.

You can also create a MAC pool from this area.

Step 8 Choose the **Fabric ID**, select the **VLANs** that you want to use, enter the **MTU**, and choose a **Pin Group**.

You can also create a VLAN and a LAN pin group from this area.

Step 9 In the **Operational Parameters** area, choose a **Stats Threshold Policy**.

Step 10 In the Adapter Performance Profile area, choose an **Adapter Policy**, **QoS Policy**, and a **Network Control Policy**.

You can also create an Ethernet adapter policy, QoS policy, and network control policy from this area.

Step 11 In the Connection Policies area, choose the **Dynamic vNIC**, **usNIC** or **VMQ** radio button, then choose the corresponding policy.

You can also create a dynamic vNIC, usNIC, or VMQ connection policy from this area.

Note Cisco UCS 6400 Series Fabric Interconnects and Cisco UCS 6536 Fabric Interconnect do not support dynamic vNICs.

Step 12 Click **OK**.

Step 13 Click **Save Changes**.

Deleting a vNIC from a LAN Connectivity Policy

Procedure

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

Step 2 Expand **LAN** > **Policies** > ***Organization_Name***.

Step 3 Expand the **LAN Connectivity Policies** node.

Step 4 Select the policy from which you want to delete the vNIC.

Step 5 In the **Work** pane, click the **General** tab.

Step 6 In the **vNICs** table, do the following:

a) Click the vNIC you want to delete.

b) On the icon bar, click **Delete**.

Step 7 If a confirmation dialog box displays, click **Yes**.

Step 8 Click **Save Changes**.

Creating an iSCSI vNIC for a LAN Connectivity Policy

Procedure

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

Step 2 Expand **LAN** > **Policies** > ***Organization_Name***.

Step 3 Expand the **LAN Connectivity Policies** node.

Step 4 Choose the policy to which you want to add an iSCSI vNIC.

Step 5 In the **Work** pane, click the **General** tab.

Step 6 On the icon bar of the **Add iSCSI vNICs** table, click **Add**.

Step 7 In the **Create iSCSI vNIC** dialog box, complete the following fields:

Name	Description
Name field	The name of the iSCSI vNIC. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Overlay vNIC drop-down list	The LAN vNIC associated with this iSCSI vNIC, if any.

Name	Description
iSCSI Adapter Policy drop-down list	The iSCSI adapter policy associated with this iSCSI vNIC, if any.
Create iSCSI Adapter Policy link	Click this link to create a new iSCSI adapter policy that will be available to all iSCSI vNICs.
VLAN drop-down list	The virtual LAN associated with this iSCSI vNIC. The default VLAN is **default**. **Note** For the Cisco UCS M81KR Virtual Interface Card and the Cisco UCS VIC-1240 Virtual Interface Card, the VLAN that you specify must be the same as the native VLAN on the overlay vNIC. For the Cisco UCS M51KR-B Broadcom BCM57711 Adapter, the VLAN that you specify can be any VLAN assigned to the overlay vNIC.

Step 8 In the **MAC Address Assignment** drop-down list in the **iSCSI MAC Address** area, choose one of the following:

- Leave the MAC address unassigned, select **Select (None used by default)**. Select this option if the server that will be associated with this service profile contains a Cisco UCS M81KR Virtual Interface Card adapter or a Cisco UCS VIC-1240 Virtual Interface Card.

  Important If the server that will be associated with this service profile contains a Cisco UCS NIC M51KR-B adapter, you must specify a MAC address.

- A specific MAC address, select **00:25:B5:XX:XX:XX** and enter the address in the **MAC Address** field. To verify that this address is available, click the corresponding link.
- A MAC address from a pool, select the pool name from the list. Each pool name is followed by a pair of numbers in parentheses. The first number is the number of available MAC addresses in the pool and the second is the total number of MAC addresses in the pool.

  If this Cisco UCS domain is registered with Cisco UCS Central, there might be two pool categories. **Domain Pools** are defined locally in the Cisco UCS domain and **Global Pools** are defined in Cisco UCS Central.

Step 9 (Optional) If you want to create a MAC pool that will be available to all service profiles, click **Create MAC Pool** and complete the fields in the **Create MAC Pool** wizard.

For more information, see the *UCS Manager Storage Management Guide*, Pools chapter, Creating a MAC Pool topic.

Step 10 Click **OK**.

Step 11 Click **Save Changes**.

Deleting a vNIC from a LAN Connectivity Policy

Procedure

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

Step 2 Expand **LAN** > **Policies** > ***Organization_Name***.

Step 3 Expand the **LAN Connectivity Policies** node.

Step 4 Select the policy from which you want to delete the vNIC.

Step 5 In the **Work** pane, click the **General** tab.

Step 6 In the **vNICs** table, do the following:

a) Click the vNIC you want to delete.

b) On the icon bar, click **Delete**.

Step 7 If a confirmation dialog box displays, click **Yes**.

Step 8 Click **Save Changes**.

Configuring SRIOV HPN Connection Policies

Single Root I/O Virtualization HPN Connection Policy

Beginning with the release 4.3(2b), Cisco UCS Manager provides Single Root I/O Virtualization High Performance Networking (SRIOV-HPN) Connection Policy support on Cisco UCS M5, M6 and M7 servers with UCS VIC 14xx, 14xxx and 15xxx series adapters.

Single Root I/O Virtualization allows multiple VMs running a variety of guest operating systems to share a single PCIe network adapter within a host server. SR-IOV allows a VM to move data directly to and from the network adapter, bypassing the hypervisor for increased network throughput and lower server CPU burden.

To configure the SRIOV-HPN policy in a service profile, select **SRIOV-HPN** from **vNIC Adapter Policy** drop-down list.

You cannot enable the following when SRIOV-HPN is enabled:

- QinQ on the same vNIC
- VXLAN on the same vNIC
- Geneve offload on the same vNIC
- ENS on the same vNIC
- RoCE V2 on the same vNIC
- Netqueue on the same vNIC

Note

- CDN is supported on the host interface only and is not supported on the VM interface.
- Microsoft stand-alone NIC Teaming on SRIOV-HPN enabled vNICs is not supported.
- DPDK is supported on Linux VM.
- RSS is supported on the same vNIC

Creating or Viewing SRIOV HPN Connection Policy Properties

Procedure

	Command or Action	Purpose
Step 1	In the **Navigation** pane, click **LAN**.	
Step 2	Expand **Servers** > **Policies**.	
Step 3	Expand the node for the organization where you want to create and view the SRIOV-HPN Connection Policies.	If the system does not include multi-tenancy, expand the root node.
Step 4	To create SRIOV HPN Connection Policy, right click on **SRIOV HPN Connection Policies**.	
Step 5	In the **General** tab, you can view and and modify the created SRIOV HPN Connection Policy properties.	

Name	Description
Name field	The name of the policy. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Description field	Brief description of the policy.
Number of SRIOV HPN vnics field	Enter an integer between 1 and 64.

	Command or Action	Purpose
		<table><tr><th>Name</th><th>Description</th></tr><tr><td>Transmit Queues field</td><td>The number of descriptors in each transmit queue. Enter an integer between 1 and 8.</td></tr><tr><td>Receive Queues field</td><td>The number of receive queue resources to allocate. Enter an integer between 1 and 8.</td></tr><tr><td>Completion Queues field</td><td>The number of completion queue resources to allocate. In general, the number of completion queue resources you should allocate is equal to the number of transmit queue resources plus the number of receive queue resources. Enter an integer between 1 and 16.</td></tr><tr><td>Interrupt Count field</td><td>The number of interrupt resources to allocate. In general, this value should be equal to the number of completion queue resources. Enter an integer between 1 and 16.</td></tr></table>
Step 6	Click **OK** to save the changes, if any.	

Assigning SRIOV HPN Connection Policy to a vNIC

Procedure

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

Step 2 On the **Servers** tab, expand **Servers** > **Service Profile** > **root**.

Step 3 Expand the service profile that you want to configure for SRIOV-HPN policy and then click **vNICs**.

Step 4 Choose the desired vNIC.

Step 5 In the **Work Pane**, click the **General** tab.

Step 6 In the **Policies** area, select **SRIOV-HPN** from the drop-down list.

Step 7 In the **Connection Policies** area, click the **SRIOV-HPN** radio button.

The **SRIOV HPN Connection Policy** drop-down list is displayed.

Step 8 Select the desired policy from the **SRIOV HPN Connection Policy** drop-down list.

Step 9 Click **OK**.

Step 10 Click **Save Changes**.

Deleting a SRIOV HPN Connection Policy

Procedure

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

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

Step 3 Expand **SRIOV HPN Connection Policies**.

Step 4 Right-click the policy you want to delete and select **Delete**.

Step 5 If a confirmation dialog box displays, click **Yes**.

Configuring Network Control Policies

Network Control Policy

This policy configures the network control settings for the Cisco UCS domain, including the following:

- Whether the Cisco Discovery Protocol (CDP) is enabled or disabled
- How the virtual interface (VIF) behaves if no uplink port is available in end-host mode
- The action that Cisco UCS Manager takes on the remote Ethernet interface, vEthernet interface , or vFibre Channel interface when the associated border port fails
- Whether the server can use different MAC addresses when sending packets to the fabric interconnect
- Whether MAC registration occurs on a per-VNIC basis or for all VLANs

Action on Uplink Fail

By default, the **Action on Uplink Fail** property in the network control policy is configured with a value of link-down. For adapters such as the Cisco UCS M81KR Virtual Interface Card, this default behavior directs Cisco UCS Manager to bring the vEthernet or vFibre Channel interface down if the associated border port fails. For Cisco UCS systems using a non-VM-FEX capable converged network adapter that supports both Ethernet and FCoE traffic, such as Cisco UCS CNA M72KR-Q and the Cisco UCS CNA M72KR-E, this default behavior directs Cisco UCS Manager to bring the remote Ethernet interface down if the associated border port fails. In this scenario, any vFibre Channel interfaces that are bound to the remote Ethernet interface are brought down as well.

Note If your implementation includes those types of non-VM-FEX capable converged network adapters mentioned in this section and the adapter is expected to handle both Ethernet and FCoE traffic, we recommend that you configure the **Action on Uplink Fail** property with a value of warning. Note that this configuration might result in an Ethernet teaming driver not being able to detect a link failure when the border port goes down.

MAC Registration Mode

MAC addresses are installed only on the native VLAN by default, which maximizes the VLAN port count in most implementations.

Note If a trunking driver is being run on the host and the interface is in promiscuous mode, we recommend that you set the MAC Registration Mode to All VLANs.

NIC Teaming and Port Security

NIC teaming is a grouping together of network adapters to build in redundancy, and is enabled on the host. This teaming or bonding facilitates various functionalities, including load balancing across links and failover. When NIC teaming is enabled and events such as failover or reconfiguration take place, MAC address conflicts and movement may happen.

Port security, which is enabled on the fabric interconnect side, prevents MAC address movement and deletion. Therefore, you must not enable port security and NIC teaming together.

Configuring Link Layer Discovery Protocol for Fabric Interconnect vEthernet Interfaces

Cisco UCS Manager allows you to enable and disable LLDP on a vEthernet interface. You can also retrieve information about these LAN uplink neighbors. This information is useful while learning the topology of the LAN connected to the UCS system and while diagnosing any network connectivity issues from the fabric interconnect (FI). The fabric interconnect of a UCS system is connected to LAN uplink switches for LAN connectivity and to SAN uplink switches for storage connectivity. When using Cisco UCS with Cisco Application Centric Infrastructure (ACI), LAN uplinks of the fabric interconnect are connected to ACI leaf nodes. Enabling LLDP on a vEthernet interface will help the Application Policy Infrastructure Controller (APIC) to identify the servers connected to the fabric interconnect by using vCenter.

To permit the discovery of devices in a network, support for Link Layer Discovery Protocol (LLDP), a vendor-neutral device discovery protocol that is defined in the IEEE 802.1ab standard, is introduced. LLDP is a one-way protocol that allows network devices to advertise information about themselves to other devices

on the network. LLDP transmits information about the capabilities and current status of a device and its interfaces. LLDP devices use the protocol to solicit information only from other LLDP devices.

You can enable or disable LLDP on a vEthernet interface based on the Network Control Policy (NCP) that is applied on the vNIC in the service profile.

Creating a Network Control Policy

MAC address-based port security for Emulex converged Network Adapters (N20-AE0102) is not supported. When MAC address-based port security is enabled, the fabric interconnect restricts traffic to packets that contain the MAC address that it first learns. This is either the source MAC address used in the FCoE Initialization Protocol packet, or the MAC address in an ethernet packet, whichever is sent first by the adaptor. This configuration can result in either FCoE or Ethernet packets being dropped.

Procedure

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

Step 2 Expand **LAN** > **Policies**.

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

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

Step 4 Right-click the **Network Control Policies** node and select **Create Network Control Policy**.

Step 5 In the **Create Network Control Policy** dialog box, complete the required fields.

Step 6 In the LLDP area, do the following:

a) To enable the transmission of LLDP packets on an interface, click **Enabled** in the **Transmit** field.
b) To enable the reception of LLDP packets on an interface, click **Enabled** in the **Receive** field.

Step 7 In the **MAC Security** area, do the following to determine whether the server can use different MAC addresses when sending packets to the fabric interconnect:

a) Click the **Expand** icon to expand the area and display the radio buttons.
b) Click one of the following radio buttons to determine whether forged MAC addresses are allowed or denied when packets are sent from the server to the fabric interconnect:

- **Allow**— All server packets are accepted by the fabric interconnect, regardless of the MAC address associated with the packets.
- **Deny**— After the first packet has been sent to the fabric interconnect, all other packets must use the same MAC address or they will be silently rejected by the fabric interconnect. In effect, this option enables port security for the associated vNIC.

If you plan to install VMware ESX on the associated server, you must configure the **MAC Security** to **allow** for the network control policy applied to the default vNIC. If you do not configure **MAC Security** for **allow**, the ESX installation may fail because the MAC security permits only one MAC address while the installation process requires more than one MAC address.

Step 8 Click **OK**.

Deleting a Network Control Policy

Procedure

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

Step 2 Expand **LAN** > **Policies** > ***Organization_Name***.

Step 3 Expand the **Network Control Policies** node.

Step 4 Right-click the policy you want to delete and select **Delete**.

Step 5 If a confirmation dialog box displays, click **Yes**.

Configuring Multicast Policies

Multicast Policy

This policy is used to configure Internet Group Management Protocol (IGMP) snooping, IGMP querier, and IGMP source IP proxy. IGMP Snooping dynamically determines hosts in a VLAN that should be included in particular multicast transmissions. You can create, modify, and delete a multicast policy that can be associated to one or more VLANs. When a multicast policy is modified, all VLANs associated with that multicast policy are re-processed to apply the changes.

By default, IGMP snooping is enabled and IGMP querier is disabled. When IGMP snooping is enabled, the fabric interconnects send the IGMP queries only to the hosts. They do not send IGMP queries to the upstream network. To send IGMP queries to the upstream, do one of the following:

- Configure IGMP querier on the upstream fabric interconnect with IGMP snooping enabled
- Disable IGMP snooping on the upstream fabric interconnect
- Change the fabric interconnects to switch mode

By default, IGMP Source IP Proxy state is enabled. When IGMP Source IP Proxy is enabled, the fabric interconnect acts as a proxy for its hosts and manages the membership of hosts and routing devices in multicast groups. IP hosts use IGMP to report their multicast group memberships to any immediately neighboring multicast routing devices. When IGMP source IP proxy is disabled, the fabric interconnect will forward the IGMP messages from the hosts towards the upstream router or switch without any change.

The following limitations and guidelines apply to multicast policies:

- Only the default multicast policy is allowed for a global VLAN.
- If a Cisco UCS domain includes 6300 and 6200 series fabric interconnects, any multicast policy can be assigned.
- We highly recommend you use the same IGMP snooping state on the fabric interconnects and the associated LAN switches. For example, if IGMP snooping is disabled on the fabric interconnects, it should be disabled on any associated LAN switches as well.

- The option to enable or disable IGMP source IP proxy is supported on Cisco UCS 6500, UCS UCS 6400, UCS 6300, and UCS 6200 Series fabric interconnects.

Creating a Multicast Policy

Procedure

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

Step 2 Expand **LAN** > **Policies**.

Step 3 Expand the **root** node.

Step 4 Right-click the **Multicast Policies** node and select **Create Multicast Policy**.

Step 5 In the **Create Multicast Policy** dialog box, specify the name, IGMP snooping, and IGMP source IP proxy information.

Note Follow these guidelines if you choose to set IGMP Snooping querier IP addresses for a multicast policy:

a. In the Ethernet Switch-Mode configuration, you must set the querier IP addresses for each FI in the domain.

b. In the Ethernet End-Host mode, you can set the querier IP address just for FI A, and optionally for FI B as well. If an IP address is not set explicitly for FI-B, it uses the same address set for FI A.

Querier IP address can be any valid IP address. However, IP address from same subnet is required if there is a strict subnet check in the host.

Step 6 Click **OK**.

Modifying a Multicast Policy

This procedure describes how to change the IGMP snooping state, IGMP snooping querier state, and IGMP Source IP Proxy state of an existing multicast policy.

Note You cannot change the name of the multicast policy once it has been created.

Procedure

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

Step 2 Expand **LAN** > **Policies**.

Step 3 Expand the **root** node.

Step 4 Click the policy that you want to modify.

Step 5 In the work pane, edit the fields as needed.

Step 6 Click **Save Changes**.

Deleting a Multicast Policy

Note If you assigned a non-default (user-defined) multicast policy to a VLAN and then delete that multicast policy, the associated VLAN inherits the multicast policy settings from the default multicast policy until the deleted policy is re-created.

Procedure

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

Step 2 Expand **LAN** > **Policies**.

Step 3 Expand the **root** node.

Step 4 Right-click the **Multicast Policies** node and select **Delete Multicast Policy**.

Step 5 If a confirmation dialog box displays, click **Yes**.

Configuring LACP Policies

LACP Policy

Link Aggregation combines multiple network connections in parallel to increase throughput and to provide redundancy. Link aggregation control protocol (LACP) provides additional benefits for these link aggregation groups. Cisco UCS Manager enables you to configure LACP properties using LACP policy.

You can configure the following for a lacp policy:

- **Suspended-individual**: If you do not configure the ports on an upstream switch for lacp, the fabric interconnects treat all ports as uplink Ethernet ports to forward packets. You can place the lacp port in suspended state to avoid loops. When you set suspend-individual on a port-channel with LACP, if a port that is part of the port-channel does not receive PDUs from the peer port, it will go into suspended state.
- **Timer values**: You can configure rate-fast or rate-normal. In rate-fast configuration, the port is expected to receive 1 PDU every 1 second from the peer port. The time out for this is 3 seconds. In rate-normal configuration, the port is expected to receive 1 PDU every 30 seconds. The timeout for this is 90 seconds.

System creates a default LACP policy at system start up. You can modify this policy or create a new policy. You can also apply one LACP policy to multiple port-channels.

Creating a LACP Policy

Procedure

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

Step 2 Expand **LAN** > **Policies**.

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

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

Step 4 In the **Work Pane**, click **LACP Policies** tab, and click the + sign.

Step 5 In the **Create LACP Policy** dialog box, fill in the required fields.

Step 6 Click **OK**.

Modifying a LACP Policy

Procedure

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

Step 2 Expand **LAN** > **Policies**.

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

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

Step 4 In the **Work Pane**, **LACP Policies** tab, and click on the policy you want to edit.

Step 5 Click the **Properties** icon on the right.

Step 6 In the **Properties** dialog box, make the required changes and click **Apply**.

Step 7 Click **OK**.

Configuring UDLD Link Policies

Understanding UDLD

UniDirectional Link Detection (UDLD) is a Layer 2 protocol that enables devices connected through fiber-optic or twisted-pair Ethernet cables to monitor the physical configuration of the cables and detect when a unidirectional link exists. All connected devices must support UDLD for the protocol to successfully identify and disable unidirectional links. When UDLD detects a unidirectional link, it marks the link as unidirectional. Unidirectional links can cause a variety of problems, including spanning-tree topology loops.

UDLD works with the Layer 1 mechanisms to determine the physical status of a link. At Layer 1, autonegotiation takes care of physical signaling and fault detection. UDLD performs tasks that autonegotiation

cannot perform, such as detecting the identities of neighbors and shutting down misconnected interfaces. When you enable both autonegotiation and UDLD, the Layer 1 and Layer 2 detections work together to prevent physical and logical unidirectional connections and the malfunctioning of other protocols.

A unidirectional link occurs whenever traffic sent by a local device is received by its neighbor but traffic from the neighbor is not received by the local device.

Modes of Operation

UDLD supports two modes of operation: normal (the default) and aggressive. In normal mode, UDLD can detect unidirectional links due to misconnected interfaces on fiber-optic connections. In aggressive mode, UDLD can also detect unidirectional links due to one-way traffic on fiber-optic and twisted-pair links and to misconnected interfaces on fiber-optic links.

In normal mode, UDLD detects a unidirectional link when fiber strands in a fiber-optic interface are misconnected and the Layer 1 mechanisms do not detect this misconnection. If the interfaces are connected correctly but the traffic is one way, UDLD does not detect the unidirectional link because the Layer 1 mechanism, which is supposed to detect this condition, does not do so. In case, the logical link is considered undetermined, and UDLD does not disable the interface. When UDLD is in normal mode, if one of the fiber strands in a pair is disconnected and autonegotiation is active, the link does not stay up because the Layer 1 mechanisms did not detect a physical problem with the link. In this case, UDLD does not take any action, and the logical link is considered undetermined.

UDLD aggressive mode is disabled by default. Configure UDLD aggressive mode only on point-to-point links between network devices that support UDLD aggressive mode. With UDLD aggressive mode enabled, when a port on a bidirectional link that has a UDLD neighbor relationship established stops receiving UDLD packets, UDLD tries to reestablish the connection with the neighbor and administratively shuts down the affected port. UDLD in aggressive mode can also detect a unidirectional link on a point-to-point link on which no failure between the two devices is allowed. It can also detect a unidirectional link when one of the following problems exists:

- On fiber-optic or twisted-pair links, one of the interfaces cannot send or receive traffic.
- On fiber-optic or twisted-pair links, one of the interfaces is down while the other is up.
- One of the fiber strands in the cable is disconnected.

Methods to Detect Unidirectional Links

UDLD operates by using two mechanisms:

- Neighbor database maintenance

  UDLD learns about other UDLD-capable neighbors by periodically sending a hello packet (also called an advertisement or probe) on every active interface to keep each device informed about its neighbors. When the switch receives a hello message, it caches the information until the age time (hold time or time-to-live) expires. If the switch receives a new hello message before an older cache entry ages, the switch replaces the older entry with the new one.

  UDLD clears all existing cache entries for the interfaces affected by the configuration change whenever an interface is disabled and UDLD is running, whenever UDLD is disabled on an interface, or whenever the switch is reset. UDLD sends at least one message to inform the neighbors to flush the part of their caches affected by the status change. The message is intended to keep the caches synchronized.

- Event-driven detection and echoing

UDLD relies on echoing as its detection mechanism. Whenever a UDLD device learns about a new neighbor or receives a resynchronization request from an out-of-sync neighbor, it restarts the detection window on its side of the connection and sends echo messages in reply. Because this behavior is the same on all UDLD neighbors, the sender of the echoes expects to receive an echo in reply.

If the detection window ends and no valid reply message is received, the link might shut down, depending on the UDLD mode. When UDLD is in normal mode, the link might be considered undetermined and might not be shut down. When UDLD is in aggressive mode, the link is considered unidirectional, and the interface is shut down.

If UDLD in normal mode is in the advertisement or in the detection phase and all the neighbor cache entries are aged out, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbors.

If you enable aggressive mode when all the neighbors of a port have aged out either in the advertisement or in the detection phase, UDLD restarts the link-up sequence to resynchronize with any potentially out-of-sync neighbor. UDLD shuts down the port if, after the fast train of messages, the link state is still undetermined.

UDLD Configuration Guidelines

The following guidelines and recommendations apply when you configure UDLD:

- A UDLD-capable interface also cannot detect a unidirectional link if it is connected to a UDLD-incapable port of another switch.
- When configuring the mode (normal or aggressive), make sure that the same mode is configured on both sides of the link.
- UDLD should be enabled only on interfaces that are connected to UDLD capable devices. The following interface types are supported:
  - Ethernet uplink
  - FCoE uplink
  - Ethernet uplink port channel member
  - FCoE uplink port channel member

Creating a Link Profile

Procedure

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

Step 2 Expand **LAN** > **Policies** > **LAN Cloud**.

Step 3 Right-click the **Link Profile** node and choose **Create Link Profile**.

Step 4 In the **Create Link Profile** dialog box, specify the name and the UDLD link policy.

Step 5 Click **OK**.

Creating a UDLD Link Policy

Procedure

Step 1 In the **Navigation** pane, click **LAN**.
Step 2 Expand **LAN** > **Policies** > **LAN Cloud**.
Step 3 Right-click the **UDLD Link Policy** node and choose **Create UDLD Link Policy**.
Step 4 In the **Create UDLD Link Policy** dialog box, specify the name, admin state, and mode.
Step 5 Click **OK**.

Modifying the UDLD System Settings

Procedure

Step 1 In the **Navigation** pane, click **LAN**.
Step 2 Expand **LAN** > **Policies** > **LAN Cloud**.
Step 3 On the LAN tab, expand **LAN** > **Policies** > **root**.
Step 4 Expand the **Link Protocol Policy** node and click **UDLD System Settings**.
Step 5 In the **Work** pane, click the **General** tab.
Step 6 In the **Properties** area, modify the fields as needed.
Step 7 Click **Save Changes**.

Assigning a Link Profile to a Port Channel Ethernet Interface

Procedure

Step 1 In the **Navigation** pane, click **LAN**.
Step 2 Expand **LAN** > **LAN Cloud** > *Fabric* > **Port Channels**.
Step 3 Expand the port channel node and click the Eth Interface where you want to assign a link profile.
Step 4 In the **Work** pane, click the **General** tab.
Step 5 In the **Properties** area, choose the link profile that you want to assign.
Step 6 Click **Save Changes**.

Assigning a Link Profile to an Uplink Ethernet Interface

Procedure

Step 1 In the **Navigation** pane, click **LAN**.
Step 2 On the **LAN** tab, expand **LAN** > **LAN Cloud** > ***Fabric*** > **Uplink Eth Interface**.
Step 3 Click the Eth Interface where you want to assign a link profile.
Step 4 In the **Work** pane, click the **General** tab.
Step 5 In the **Properties** area, choose the link profile that you want to assign.
Step 6 Click **Save Changes**.

Assigning a Link Profile to a Port Channel FCoE Interface

Procedure

Step 1 In the **Navigation** pane, click **SAN**.
Step 2 On the **SAN** tab, expand **SAN** > **SAN Cloud** > ***Fabric*** > **FCoE Port Channels**.
Step 3 Expand the FCoE port channel node and click the FCoE Interface where you want to assign a link profile.
Step 4 In the **Work** pane, click the **General** tab.
Step 5 In the **Properties** area, choose the link profile that you want to assign.
Step 6 Click **Save Changes**.

Assigning a Link Profile to an Uplink FCoE Interface

Procedure

Step 1 In the **Navigation** pane, click **SAN**.
Step 2 On the **SAN** tab, expand **SAN** > **SAN Cloud** > ***Fabric*** > **Uplink FC Interfaces**.
Step 3 Click the FCoE interface where you want to assign a link profile.
Step 4 In the **Work** pane, click the **General** tab.
Step 5 In the **Properties** area, choose the link profile that you want to assign.
Step 6 Click **Save Changes**.

Configuring VMQ and VMMQ Connection Policies

VMQ Connection Policy

Cisco UCS Manager enables you to configure VMQ connection policy for a vNIC. VMQ provides improved network performance to the entire management operating system. Configuring a VMQ vNIC connection policy involves the following:

- Create a VMQ connection policy
- Create a static vNIC in a service profile
- Apply the VMQ connection policy to the vNIC

If you want to configure the VMQ vNIC on a service profile for a server, at least one adapter in the server must support VMQ. Make sure the servers have at least one the following adapters installed:

- UCS VIC 1200 Series
- UCS VIC 1300 Series
- UCS VIC 1400 Series
- UCS-VIC-15000 Series

The following are the supported Operating Systems for VMQ:

- Windows 2012
- Windows 2012R2
- Windows 2016
- Windows 2019
- Windows 2022

Note The Cisco UCS VIC 1400 and UCS VIC 15000 Series adapters are not supported on Windows 2012 VMQ and Windows 2012 R2 VMQ

You can apply only any one of the vNIC connection policies on a service profile at any one time. Make sure to select one of the three options such as Dynamic, usNIC or VMQ connection policy for the vNIC. When a VMQ vNIC is configured on service profile, make sure you have the following settings:

- Select SRIOV in the BIOS policy.
- Select Windows in the Adapter policy.

Creating a VMQ Connection Policy

Before you create a VMQ connection policy, consider the following:

- VMQ Tuning on the Windows Server—When an adapter is placed on a virtual switch, running the **Get-NetAdapterVmq** cmdlet displays `True` for VMQ.
- Virtual machine level—By default, VMQ is enabled on all newly deployed VMs. VMQ can be enabled or disabled on existing VMs.
- Microsoft SCVMM—VMQ must be enabled on the port profile. If not, you will not be able to successfully create the virtual switch in SCVMM.
- Microsoft Azure Stack extends the existing VMQ support for host-side virtual switch ports called vPorts to Virtual Machine Multi Queues (VMMQ). You can configure VMMQ by enabling multi queues in the VMQ Connection Policy.

For Cisco UCS VIC 1400 Series or above adapters to support VMQ functionality, the vNIC should be configured in the VMQ connection policy with the multi-queue option enabled.

Note Microsoft stand-alone NIC Teaming and Virtual Machine Queue (VMQ) support for adapters:

Microsoft stand-alone NIC teaming works only with VMQ. For Cisco UCS VIC 1400 and VIC 15000 Series adapters, the supported VMQ is VMMQ with single queue. To support VMMQ with single queue, you must create a new VMMQ adapter policy containing a 1 TQ, 1 RQ and 2 CQ combination, then assign it to the VMQ Connection Policy.

Procedure

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

Step 2 In the **LAN** tab, expand **Policies**.

Step 3 Expand the nodes for the organization where you want to create the policy. If the system does not include multitenancy, expand the root node.

Step 4 Right-click the **VMQ Connection Policies** node and select **Create VMQ Connection Policy**.

Step 5 In the **Create VMQ Connection Policy** dialog box, complete the following fields:

Name	Description
Name field	The VMQ connection policy name.
Description field	The description of the VMQ connection policy.

Name	Description
Multi Queue radio button	Whether Virtual Machine Multi-Queue (VMMQ) is enabled in the policy. With VMMQ, multiple queues are allocated to a single VM. • **Disabled**—Multi Queue is disabled and you can configure a VMQ policy. When Multi Queue is disabled, the following fields appear: • **Number of VMQs** • **Number of Interrupts** • **Enabled**—Multi Queue is enabled and the vNIC is placed into VMMQ mode. You can specify a VMMQ Adapter Policy. When Multi Queue is enabled, the following fields appear: • **Number of Sub vNICs** • **VMMQ Adapter Policy** **Note** For Cisco UCS VIC 1400 Series and above adapters, enable the Multi-Queue option to support both VMQ and VMMQ functionality. For more information on creating a VMQ Connection Policy with Multi-Queue enabled, see Creating a VMMQ Connection Policy, on page 61.
Number of VMQs field	The number of VMQs per adapter must be one more than the maximum number of VM NICs. The default value is 64.
Number of Interrupts field	The number of CPU threads or logical processors available in the server. The default value is 64. **Note** The minimum interrupt to be used is “2 x number of CPU core + 4”.

Step 6 Click **OK**.

Assigning VMQ Setting to a vNIC

Procedure

Step 1	In the **Navigation** pane, click **Servers**.
Step 2	On the **Servers** tab, expand **Servers** > **Service Profile** > **root** .
Step 3	Expand the service profile that you want to configure for VMQ and then click **vNICs**.
Step 4	In the **Work Pane**, click the **Network** tab.
Step 5	In the **vNICs** area, choose a vNIC and double-click the **Actual Order** column. **Modify vNIC** window is displayed.
Step 6	In the **Adapter Performance Profile** area of the Modify vNIC dialog box, choose **Windows** from the Adapter Policy drop-down list.
Step 7	In the **Connection Policies** area, click the **VMQ** radio button.
Step 8	Select the **VMQ Connection Policy** from the VMQ Connection Policy drop-down list.
Step 9	Click **OK**.
Step 10	Click **Save Changes**.

Enabling VMQ and NVGRE Offloading on the same vNIC

Perform the tasks in the table below to enable VMQ and NVGRE offloading on the same vNIC.

Note VMQ is not supported along with VXLAN on the same vNIC except for Cisco UCS VIC 1400 Series adapters and above. Cisco UCS VIC 1400 and VIC 15000 Series adapters support VMQ and VMMQ along with VXLAN or NVGRE on the same vNIC.

Task	Description	See
Enable normal NVGRE offloading	Perform this task by setting the corresponding flags in the adapter profile which is associated with the given vNIC. **Note** The Transmit checksum offload and TSO must be enabled for the NVGRE offloading to be effective.	Configuring an Ethernet Adapter Policy to Enable Stateless Offloads with NVGRE, on page 32

Task	Description	See
Enable VMQ	Perform this task by setting the appropriate connection policy when you add a vNIC to the service profile.	Creating a VMQ Connection Policy, on page 56 Assigning VMQ Setting to a vNIC, on page 59

VMMQ Connection Policy

Cisco UCS Manager introduces support for Virtual Machine Multi Queues (VMMQ). VMMQ allows you to configure multiple I/O queues to a single VM and, thus, distribute traffic across multiple CPU cores in a VM. VMMQ is supported on UCS VIC 1400 Series and above adapters with Windows 2016 and later versions. VMMQ with RDMA/RDMA Over Converged Ethernet (RoCEv2) mode 2 is supported from Windows 2019 and later.

The VMQ Connection Policy has an option called **Multi Queue**. When **Multi Queue** is enabled, the vNIC is placed into VMMQ mode. In this mode, you can configure sub vNICs and specify a VMMQ Adapter policy. The policy includes the aggregate queue counts for VMMQ and determines how the connectivity between VMs and Azure Stack vPorts is configured.

Enabling VMMQ on a vNIC involves the following two configurations:

- Attach an adapter-policy for the vNIC. The recommended adapter policy for VMMQ is **Win-HPN**, available in UCS Manager.
- Include a VMQ connection policy on the vNIC. The VMQ connection policy defines Tx/Rx queues for the vPorts. For the VMQ connection policy, Cisco recommends using a pre-defined multi-queue (MQ) policy, which is available in UCS Manager. Pre-defined policies are available in UCSM: **MQ** for regular VMMQ. The pre-defined policies are good for 64 sub vNICs or vPorts in pooled mode.

Note To use RDMA, you must enable RDMA in the options under the vNIC adapter policy . For RDMA, refer to the *Cisco UCS Manager Configuration Guide for RDMA over Converged Ethernet (RoCE) v2*.

There are two different ways to define the total number of queues available for vPorts. In the pooled mode, the resource counts in the VMMQ adapter policy are the totals available across vPorts. In non-pooled mode, the total available is the selected resource count from the VMMQ adapter policy * subvnic count. In VMMQ mode, these are the default queue counts:

Queue Resource	Pooled Mode	Non Pooled Mode
Transmit Queue	64	1
Receive Queue	512	8
Completion Queue	576	9

Creating a VMMQ Connection Policy, on page 61 provides detailed information about creating a VMMQ connection policy.

VMMQ Guidelines

- Each VMMQ vPort may use one Transmit Queue and multiple Receive Queues. When VMMQ is enabled, a pool of queues is created, and the host driver assigns queues to vPorts. Different vPorts may be assigned different numbers of queues based on the number of cores that the vPort will be servicing.
- VXLAN and NVGRE offloads are supported with VMMQ functionality. The option is enabled in the vNIC adapter policy and not in the sub vNIC adapter policy.
- RSS is supported on VMMQ Receive Queues, including inner packet of overlay packets.
- VMMQ vNICs support a rate limit set by the host, not from Cisco UCS Manager. COS will not be adjustable per vPort from Cisco UCS Manager.
- vNICs with the VMQ feature, specified through the VMQ Connection Policy with **Multi Queue** disabled, are not allowed on the same adapter as Multi Queue-enabled vNICs.
- FCoE and VMMQ vNICs can coexist on the same server.
- usNIC and Multi-Queue VMQ can not be enabled on the same VIC.
- Modifying the VMMQ adapter policy through the VMQ connection policy results in exceeding the maximum Completion Queue (CQ) value. Each VIC 1400 Series or above adapter supports a maximum of 2000 hardware CQ resources. If this number is exceeded, the `Out of CQ Resources` error appears in the Cisco UCS Manager GUI, and vNIC creation fails with a configuration failure at service profile association.
- Use the following PS command to enable VMMQ on the vport.

```
Set-VMNetworkAdapter -Name (vmNIC Name) -VMName (VM_NAME) -VmmqEnabled $true
-VmmqQueuePairs (Queue_Pair_Count) -VrssEnabled $true
```

Creating a VMMQ Connection Policy

A VMMQ connection policy can be created using VMQ policy with Multi Queue enabled.

Procedure

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

Step 2 In the **LAN** tab, expand **Policies**.

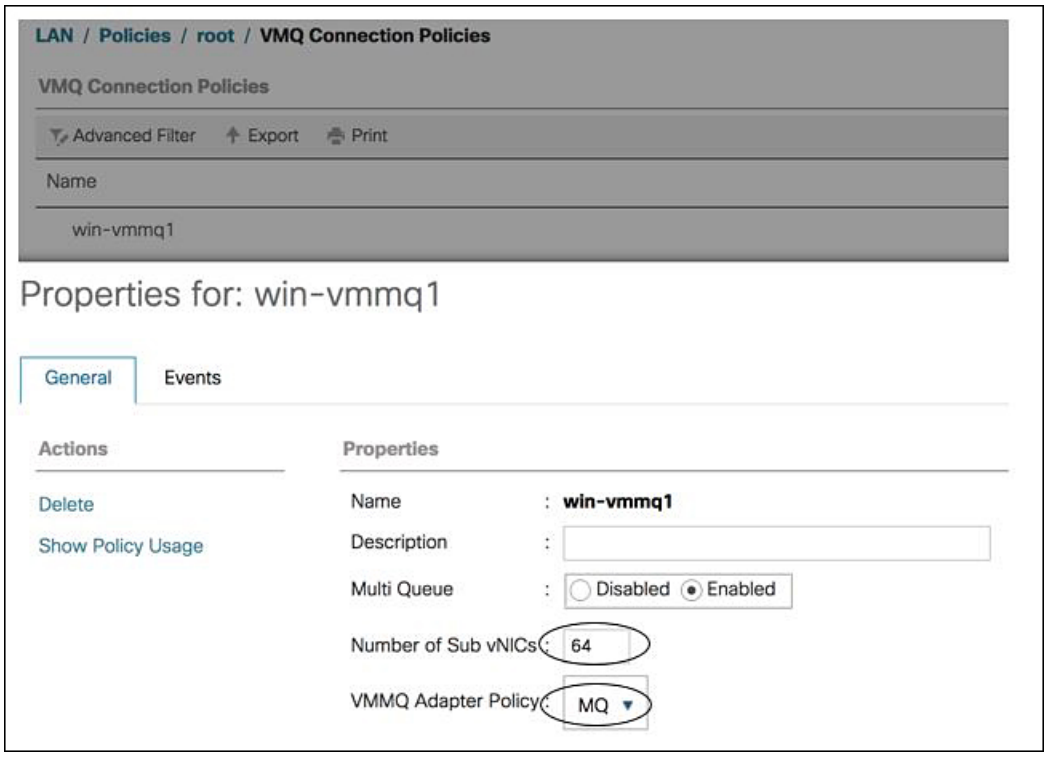
Step 3 Expand the nodes for the organization where you want to create the policy. If the system does not include multitenancy, expand the root node.

Step 4 Right-click the **VMQ Connection Policies** node and select **Create VMQ Connection Policy**.

Step 5 In the **Create VMQ Connection Policy** dialog box, complete the following fields:

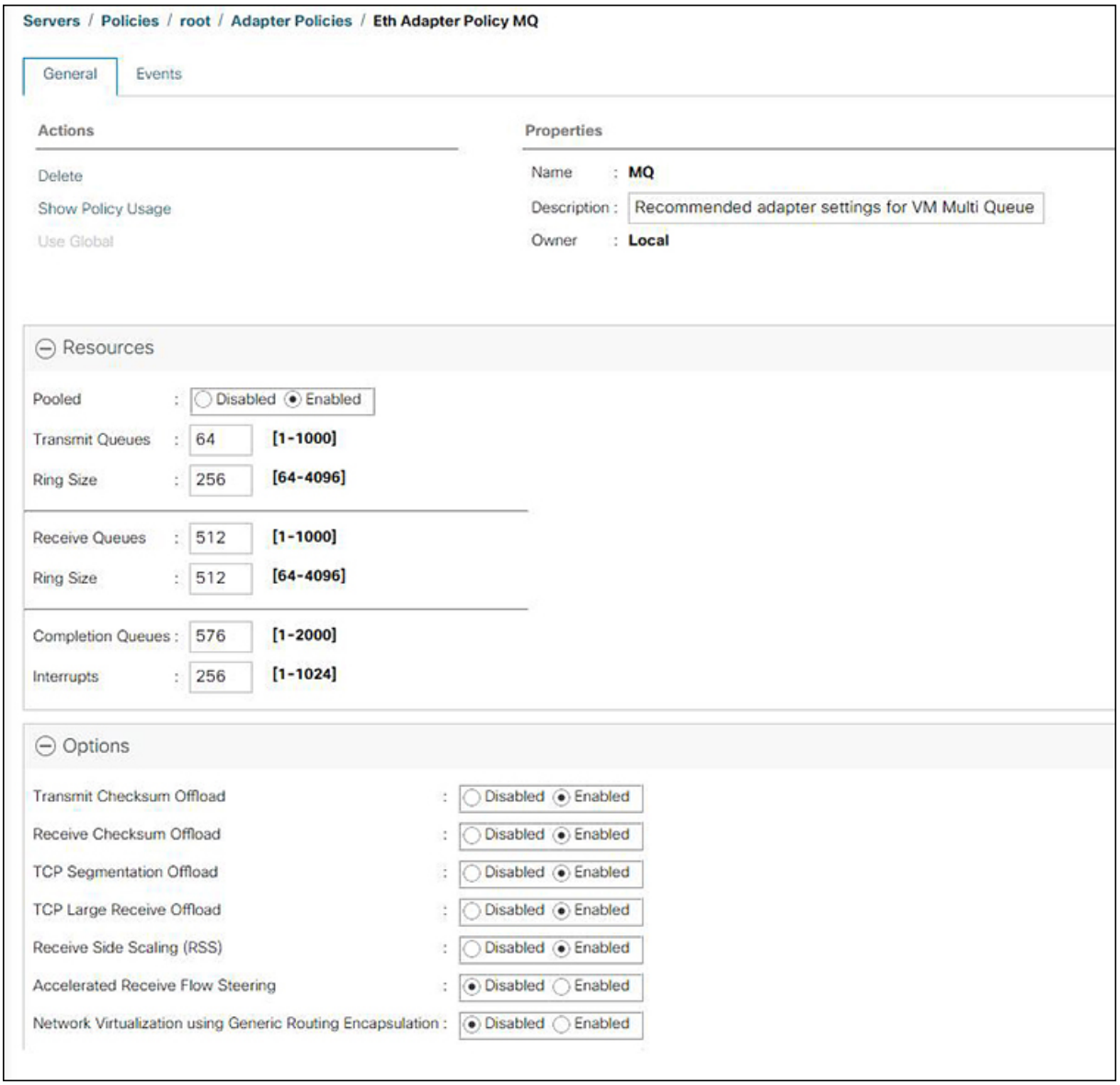
Name	Description
Name field	The VMQ connection policy name.
Description field	The description of the VMQ connection policy.

Name	Description
Multi Queue radio button	When Virtual Machine Multi-Queue (VMMQ) is enabled in the policy, multiple queues are allocated to a single vport. • **Enabled**—Multi Queue is enabled and the vNIC is placed into VMMQ mode. You can specify a VMMQ Adapter Policy. When Multi Queue is enabled, the following fields appear: • **Number of Sub vNICs** • **VMMQ Adapter Policy** **Note** For Cisco UCS VIC 1400 VIC 15000 series adapters, enable the Multi-Queue option to support both VMQ and VMMQ functionality.
Number of Sub vNICs field	Number of sub vNICs that are available for Multi Queue. The default value is 64. **Note** The TQ and RQ resource value of VMMQ adapter policy should be greater than or equal to the configured number of sub vNICs.
VMMQ Adapter Policy drop-down list	Name of the VMMQ adapter policy. Cisco recommends using the default MQ Adapter Policy. The default MQ policy includes the aggregate queue counts for VMMQ. **Note** You can also specify a custom policy designed for a specific configuration,



Step 6 Click **OK**.

Step 7 Go to the new policy under **VMQ Connection Policies**



Step 8 Set the number of **Transmit Queues** to 64 and **Receive Queues** to 8 times the transmit queues (512). The **Completion Queues** is the total of these two numbers (576).

Step 9 Set the **Interrupt** count to 256.

Step 10 Enable **Pooled** resources.

Step 11 Enable **Receive Side Scaling (RSS)**.

Step 12 Click **OK**.

What to do next

Assign a QoS policy.

Creating a QoS Policy for VMMQ

Procedure

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

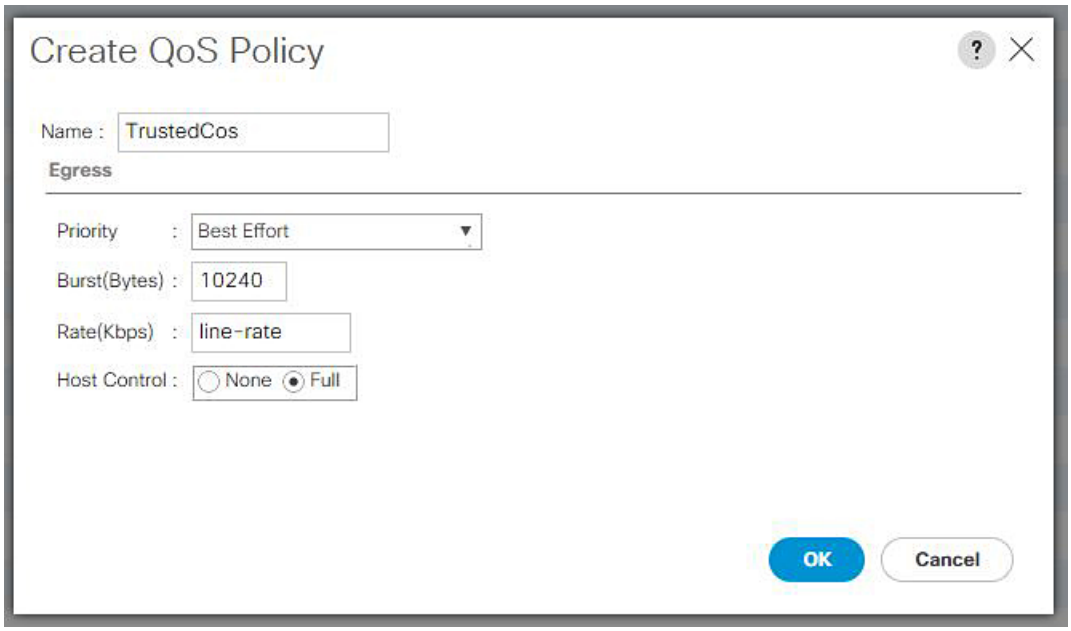
Step 2 In the **LAN** tab, expand **Policies**.

Step 3 Expand the nodes for the organization where you want to create the pool. If the system does not include multitenancy, expand the **root** node.

Step 4 Right-click the **QoS Policy** dialog box and enter the name of the policy in the **Name** field. VMMQ uses **TrustedCos** as the policy. Assign this policy to the vNIC QoS.

Step 5 Select the desired priority in the **Priority** drop-down list.

Step 6 In the **Host Control** field, click the **Full** radio button.



Step 7 Click **OK**.

What to do next

Assign the VMMQ Setting to a vNIC.

Assigning a VMMQ Setting to a vNIC

Procedure

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

Step 2 In the **Servers** tab, expand **Servers** > **Service Profiles** > **root**.

Step 3 Expand the service profile that you want to configure VMMQ and click **vNICs**.

Step 4 In the **Work Pane**, click the **Network** tab.

Step 5 In the **vNICs** area, choose the desired vNIC and double-click the **Actual Order** column.

Modify vNIC window is displayed.

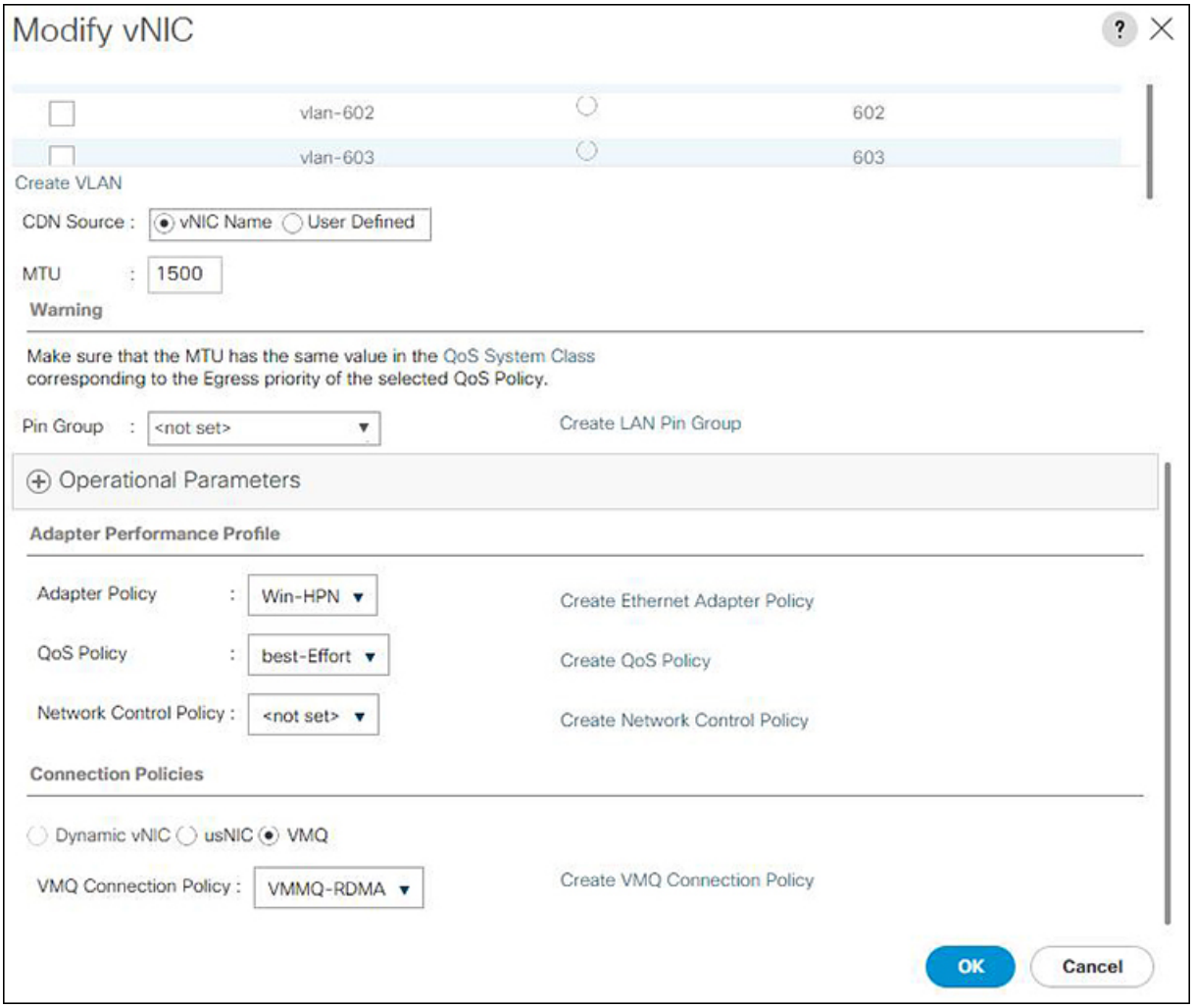
Step 6 In the **Adapter Performance Profile** area of the **Modify vNIC** dialog box, choose **WIN-HPN** from the **Adapter Policy** drop-down list.

Step 7 From the **QoS Policy** drop-down list, select the created QoS policy for VMMQ.

Step 8 In the **Connection Policies** area, click the **VMQ** radio button.

Step 9 Choose the created VMQ connection policy with Multi-Queue enabled from the **VMQ Connection Policy** drop-down list.

Step 10 Click **OK**.



Step 11 Click **Save Changes**.

NetQueue

Information About NetQueue

NetQueue improves traffic performance by providing a network adapter with multiple receive queues. These queues allow the data interrupt processing that is associated with individual virtual machines to be grouped.

Note NetQueue is supported on servers running VMware ESXi operating systems.

Configuring NetQueue

Procedure

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

Step 2 In the **LAN** tab, expand **Policies**.

Step 3 Expand the nodes for the organization where you want to create the policy. If the system does not include multitenancy, expand the root node.

Step 4 Right-click the **VMQ Connection Policies** node and select **Create VMQ Connection Policy**.

Step 5 In the **Create VMQ Connection Policy** dialog box, complete the following fields:

Name	Description
Name field	The NetQueue policy name.
Description field	The description of the NetQueue.
Multi Queue radio button	Select disabled for NetQueue.
Number of VMQs field	Enter a number between 1 to 64 to specify the number of NetQueues for this connection policy. The driver supports up to 16 NetQueues per port for standard frame configurations. **Note** VMware recommends that you use up to eight NetQueues per port for standard frame configurations.
Number of Interrupts field	The number of interrupts count of each VNIC. The value should be set to 2 x number of VMQs + 2.

Step 6

Step 7 Click **OK**.

Step 8 In the **Navigation** pane, click **Servers**.

Step 9 On the **Servers** tab, expand **Servers > Service Profiles > root**.

Step 10 Expand the service profile that you want to configure NetQueue and click vNICs.

Step 11 In the **Work** pane, click the **Network** tab.

Step 12 In the **vNICs** area, choose a vNIC and double-click the **Actual Order** column.

Modify vNIC window is displayed.

Step 13 In the **Adapter Performance Profile** area of the Modify vNIC dialog box, choose **VMWare** from the Adapter Policy drop-down list.

Step 14 In the **Connection Policies** area, click the **VMQ** radio button.

Step 15 Choose the created VMQ connection policy for NetQueue from the VMQ Connection Policy drop-down list.

Step 16 Click **OK**.

Step 17 Click **Save Changes**.

Note NetQueue should be enabled only on MSIX systems.

You should disable NetQueue on 1GB NICs.