

Cisco UCS Manager VIC Configuration Guide, Release 6.0

First Published: 2025-09-02

Last Modified: 2026-04-02

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

© 2025–2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

Preface vii

Audience vii

Conventions vii

Related Cisco UCS Documentation ix

Documentation Feedback ix

CHAPTER 1

New and Changed Information 1

New and Changed Information 1

CHAPTER 2

Overview 3

Overview of VIC Configuration Guide 3

RDMA Over Converged Ethernet (RoCE) v2 3

NDIS Poll Mode for Windows 3

NFS Over RDMA 4

Single Root I/O Virtualization 4

CHAPTER 3

Guidelines, Limitations, and Requirements 5

RoCEv2 for Windows 5

Guidelines for Using SMB Direct support using RoCEv2 5

Windows Requirements 7

RoCEv2 for Linux 7

Guidelines for Using NVMe over Fabrics (NVMeoF) with RoCEv2 7

Linux Requirements 8

NFS Over RDMA 9

Guidelines for Using NFS over RDMA 9

Linux Requirements 10

RoCEv2 for ESXi 11
Guidelines for Using RoCEv2 Protocol in the Native ENIC driver on ESXi 11
ESXi nENIC RDMA Requirements 11
SRIOV for ESXi 12
Guidelines and Limitations 12
ESXi Requirements 12
SRIOV for Linux 13
Guidelines and Limitations 13
Linux Requirements 13
Express Data Path (XDP) 14
Express Data Path (XDP) for Linux 14
Guidelines and Limitations 14
Linux Requirements 14
NDIS Poll Mode support for Windows 15
Guidelines for Using NDIS Poll Mode for Windows 15

CHAPTER 4 **Configuring RDMA Over Converged Ethernet Using RoCEv2** 17
Configuring RoCEv2 in Windows 17
Configuring RoCEv2 Modes 1 and 2 in Windows 17
Configuring SMB Direct Mode 1 on Cisco UCS Manager 17
Configuring SMB Direct Mode 1 on the Host System 18
Configuring Mode 2 on Cisco UCS Manager 20
Configuring SMB Direct Mode 2 on the Host System 21
Configuring RoCEv2 in Linux 23
Configuring NVMeoF Using RoCEv2 on Cisco UCS Manager 23
Enabling SRIOV BIOS Policy 24
Configuring NVMeoF Using RoCEv2 on the Host 25
Installing Cisco enic and enic_rdma Drivers 26
Discovering the NVMe Target 27
Setting Up Device Mapper Multipath 28
Deleting the RoCEv2 Interface Using Cisco UCS Manager 29
Configuring RoCEv2 in ESXi 30
Configuring NVMEoF using RoCEv2 for ESXi on UCS Manager 30
Installing NENIC Driver 31

ESXi NVMe RDMA Host Side Configuration 32
NENIC RDMA Functionality 32
Create Network Connectivity Switches 33
Create VMHBA Ports in ESXi 34
Displaying vmnic and vmrdma Interfaces 34
NVMe Fabrics and Namespace Discovery 37
Using the UCS Manager CLI to Configure or Delete the RoCEv2 Interface 38
Configure Windows SMB Direct RoCEv2 Interface using UCS Manager CLI 38
Deleting the Windows RoCEv2 Interface Using the CLI for UCS Manager 39
Configuring the Linux RoCEv2 Interface Using the UCS Manager CLI 40
Deleting the Linux RoCEv2 Interface Using the UCS Manager CLI 41
Configuring the VMware ESXi RoCEv2 Interface Using the UCS Manager CLI 42
Deleting the ESXi RoCEv2 Interface Using UCS Manager 43

CHAPTER 5 **Configuring NFS Over RDMA Using ROCEv2** 45
Configuring NFS over RDMA on Cisco UCS Manager 45
Deleting the NFS over RDMA Interface Using Cisco UCS Manager 46
Configuring the NFS over RDMA Interface Using the UCS Manager CLI 47
Deleting the NFS over RDMA Interface Using the Cisco UCS Manager CLI 48
Configuring NFS over RDMA on the Linux Host (AMD) 49
Configuring NFS over RDMA on the Linux Host (Intel) 50
Installing Cisco eNIC and enic_rdma Drivers 51
Mounting and Verifying NFS over RDMA Volumes 52

CHAPTER 6 **Configuring Single Root I/O Virtualization (SR-IOV)** 55
Configuring BIOS and Cisco UCS Manager Parameters 55
Enabling BIOS Parameters 55
Enabling SR-IOV VFs using Cisco UCS Manager GUI 56
Disabling SR-IOV VFs Using Cisco UCS Manager GUI 57
Enabling SR-IOV VFs using Cisco UCS Manager CLI 58
Disabling SR-IOV VFs using Cisco UCS Manager CLI 59
Configuring SR-IOV VFs on the ESXi Host Server 60
Installing Cisco eNIC Driver 60
Verifying the Total Number of SR-IOV VFs Per Ports on the Host 60

Creating SR-IOV VFs on the Host 61
Configuring the Switch 62
Creating a Virtual Port 64
Creating a New Virtual Machine (VM) 64
Adding SR-IOV VF on the Virtual Machine 65
Installing OS on Guest VM on ESXi 65
Configuring SR-IOV VFs on the Linux Host Server 66
Installing Cisco eNIC Driver 66
Verifying the Total number of SR-IOV VFs per Port on the Host 67
Creating SR-IOV VFs on the Host 67
Creating a New Virtual Machine (VM) 69
Adding SR-IOV VF on the Virtual Machine 70



Preface

- Audience, on page vii
- Conventions, on page vii
- Related Cisco UCS Documentation, on page ix
- Documentation Feedback, on page ix

Audience

This guide is intended primarily for data center administrators with responsibilities and expertise in one or more of the following:

- Server administration
- Storage administration
- Network administration
- Network security

Conventions

Text Type	Indication
GUI elements	GUI elements such as tab titles, area names, and field labels appear in **this font**. Main titles such as window, dialog box, and wizard titles appear in **this font**.
Document titles	Document titles appear in *this font*.
TUI elements	In a Text-based User Interface, text the system displays appears in `this font`.
System output	Terminal sessions and information that the system displays appear in `this font`.
CLI commands	CLI command keywords appear in **this font**. Variables in a CLI command appear in *this font*.
[ ]	Elements in square brackets are optional.

Text Type	Indication
{x | y | z}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.



Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the document.



Tip Means *the following information will help you solve a problem*. The tips information might not be troubleshooting or even an action, but could be useful information, similar to a Timesaver.



Timesaver Means *the described action saves time*. You can save time by performing the action described in the paragraph.



Caution Means *reader be careful*. In this situation, you might perform an action that could result in equipment damage or loss of data.



Warning IMPORTANT SAFETY INSTRUCTIONS

This warning symbol means danger. You are in a situation that could cause bodily injury. Before you work on any equipment, be aware of the hazards involved with electrical circuitry and be familiar with standard practices for preventing accidents. Use the statement number provided at the end of each warning to locate its translation in the translated safety warnings that accompanied this device.

SAVE THESE INSTRUCTIONS

Related Cisco UCS Documentation

Documentation Roadmaps

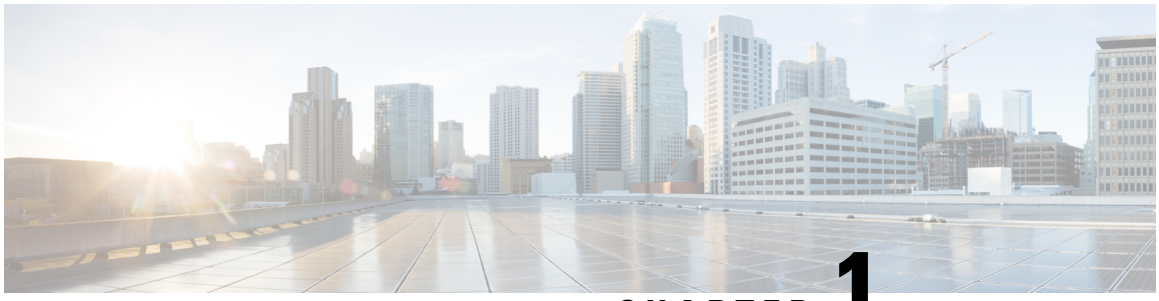
For a complete list of all B-Series documentation, see the *Cisco UCS B-Series Servers Documentation Roadmap* available at the following URL: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/overview/guide/UCS_roadmap.html

For a complete list of all C-Series documentation, see the *Cisco UCS C-Series Servers Documentation Roadmapdoc roadmap* available at the following URL: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/overview/guide/ucs_rack_roadmap.html.

For information on supported firmware versions and supported UCS Manager versions for the rack servers that are integrated with the UCS Manager for management, refer to Release Bundle Contents for Cisco UCS Software.

Documentation Feedback

To provide technical feedback on this document, or to report an error or omission, please send your comments to ucs-docfeedback@external.cisco.com. We appreciate your feedback.



CHAPTER 1

New and Changed Information

- New and Changed Information, on page 1

New and Changed Information

The following table provides an overview of the significant changes to this guide for this current release. The table does not provide an exhaustive list of all changes made to this guide or of all new features in this release.

Table 1: New Features and Changed Behavior in Cisco UCS Manager, Release 6.0(3a)

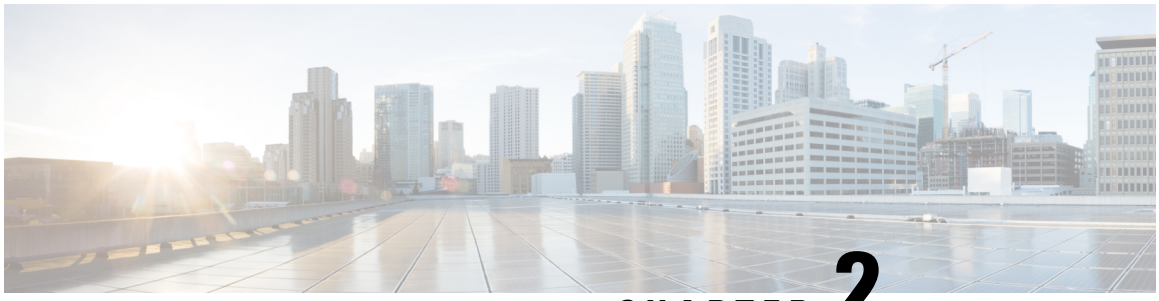
Feature	Description	Where Documented
RoCEv2 with direct-attached Pure Storage Arrays	Cisco UCS Manager now supports NVMe over FabrMA using RoCEv2 for supported Pure Storage arrays connected directly to fabric interconnect appliance ports or port channels. Configure the Platinum no-drop QoS system class with CoS 5 and DSCP 46, set the appliance-interface priority to Platinum, enable Match DSCP, and use the existing Linux or ESXi host configuration workflow.	Configuring NVMeoF Using RoCEv2 on Cisco UCS Manager, on page 23
Express Data Path (XDP) Support for Linux	Cisco UCS Manager now supports Express Data Path (XDP) on Cisco UCS VIC 1400, 14000, and 15000 Series adapters running supported Linux operating systems. This enhancement improves packet-processing efficiency and throughput by enabling the Cisco eNIC driver to process packets earlier in the Linux networking stack. XDP is configured on the Linux host; no Cisco UCS Manager configuration is required.	• Express Data Path (XDP) for Linux, on page 14 • Guidelines and Limitations, on page 14 • Linux Requirements, on page 14

Table 2: New Features and Changed Behavior in Cisco UCS Manager, Release 6.0(2b)

Feature	Description	Where Documented
NFS over RDMA	Added support for NFS over RDMA using RoCEv2. This feature is supported on Cisco UCS VIC 15000 Series adapters running on Red Hat Enterprise Linux (RHEL) 9.7 or later and 10.1 or later.	• NFS Over RDMA, on page 4 • Guidelines for Using NFS over RDMA, on page 9 • Linux Requirements, on page 10 • Configuring NFS over RDMA on Cisco UCS Manager, on page 45 • Deleting the NFS over RDMA Interface Using Cisco UCS Manager, on page 46 • Configuring the NFS over RDMA Interface Using the UCS Manager CLI, on page 47 • Deleting the NFS over RDMA Interface Using the Cisco UCS Manager CLI, on page 48 • Configuring NFS over RDMA on the Linux Host (AMD), on page 49 • Installing Cisco eNIC and enic_rdma Drivers, on page 51 • Mounting and Verifying NFS over RDMA Volumes, on page 52

Table 3: New Features and Changed Behavior in Cisco UCS Manager, Release 6.0(1b)

Feature	Description	Where Documented
NDIS Poll Mode	Added support for NDIS Poll Mode on Windows Server 2025 and later. This feature allows the operating system to manage network processing directly, enhancing performance and efficiency. Supported on Cisco UCS VIC 15000 Series adapters.	• NDIS Poll Mode for Windows, on page 3 • Guidelines for Using NDIS Poll Mode for Windows, on page 15



CHAPTER 2

Overview

- Overview of VIC Configuration Guide , on page 3
- RDMA Over Converged Ethernet (RoCE) v2 , on page 3
- NDIS Poll Mode for Windows, on page 3
- NFS Over RDMA, on page 4
- Single Root I/O Virtualization, on page 4

Overview of VIC Configuration Guide

A Cisco UCS network adapter can be installed to provide options for I/O consolidation and virtualization support. This guide contains configuration details on RDMA over Converged Ethernet version 2 (RoCEv2) , NDIS Poll Mode, and Single Root I/O Virtualization (SR-IOV).

RDMA Over Converged Ethernet (RoCE) v2

RDMA over Converged Ethernet version 2, or RoCEv2, is an IP-based RDMA protocol that allows RoCEv2 packets to be routed across Layer 3 networks.

RoCEv2 enables direct memory access over the network by encapsulating InfiniBand transport packets over Ethernet using UDP/IP. The RoCEv2 protocol runs over either UDP/IPv4 or UDP/IPv6. UDP destination port 4791 is reserved for RoCEv2. Because RoCEv2 packets are routable, the protocol is sometimes referred to as Routable RoCE.

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

NDIS Poll Mode for Windows

Network Driver Interface Specification (NDIS) Poll Mode is an advanced traffic processing mechanism designed to optimize how the operating system manages network data. NDIS Poll Mode allows the operating system to control how network traffic is processed, providing improvements over older methods that relied on Deferred Procedure Calls (DPCs). By using the NDIS layer, the operating system manages network processing directly, which enhances performance, stability, and efficiency, especially during periods of heavy network activity.

NFS Over RDMA

Network File System (NFS) over Remote Direct Memory Access (RDMA) optimizes network storage performance by allowing the NFS protocol to operate over high-speed, RDMA-capable fabrics. By bypassing traditional network stack processing, this feature significantly reduces Central Processing Unit (CPU) overhead, thereby enhancing overall system efficiency.

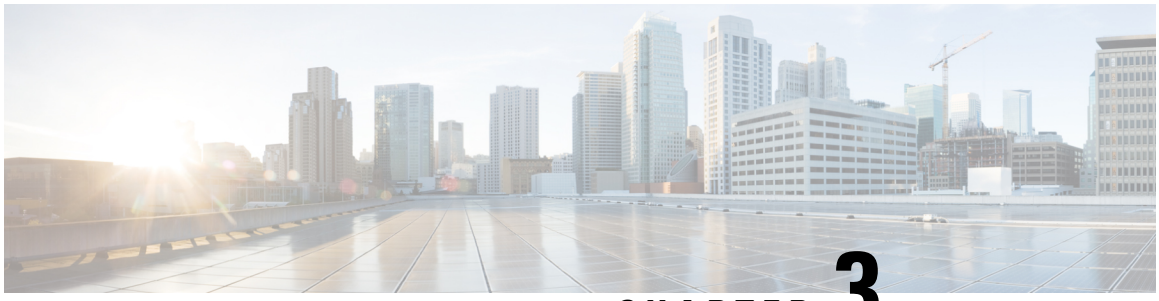
Single Root I/O Virtualization

Single Root I/O Virtualization (SR-IOV) 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. Recent x86 server processors include chipset enhancements, such as Intel VT-x technology, that facilitate direct memory transfers and other operations required by SR-IOV.

The SR-IOV specification defines two device types:

- Physical Function (PF)—Essentially a static vNIC, a PF is a full PCIe device that includes SR-IOV capabilities. PFs are discovered, managed, and configured as normal PCIe devices. A single PF can provide management and configuration for a set of virtual functions (VFs).
- Virtual Function (VF)—A VF is a full or lightweight virtual PCIe device that provides at least the necessary resources for data movements. A VF is not managed directly but is derived from and managed through a PF. One or more VFs can be assigned to a VM.

SR-IOV is defined and maintained by the Peripheral Component Interconnect Special Interest Group (PCI-SIG), an industry organization that is chartered to develop and manage the PCI standard.



CHAPTER 3

Guidelines, Limitations, and Requirements

- RoCEv2 for Windows, on page 5
- RoCEv2 for Linux, on page 7
- NFS Over RDMA, on page 9
- RoCEv2 for ESXi, on page 11
- SRIOV for ESXi, on page 12
- SRIOV for Linux, on page 13
- Express Data Path (XDP), on page 14
- NDIS Poll Mode support for Windows, on page 15

RoCEv2 for Windows

Guidelines for Using SMB Direct support using RoCEv2

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. See Windows Requirements, on page 7.

Note RoCEv2 is not supported on Microsoft Windows Server 2016.

- Cisco recommends you check UCS Hardware and Software Compatibility specific to your Cisco 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 Series, 14000 Series, and 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 Series, Cisco UCS VIC 14000 Series, and 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.
- Support for RoCEv2 protocol for Windows 2019 NDKPI mode 1 and mode 2, with both IPV4 and IPV6.
- 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 maximum number of memory regions per rNIC interface is 131072.
- Cisco 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.
- Set the QoS No Drop CoS value to 5. (For example: platinum enable with no drop).

  For detailed configuration on upstream Nexus switches, see FlashStack Oracle RAC 19c with NVMe over RoCE CVD.
- RoCEv2 cannot be used with usNIC.

MTU Properties

- In older versions of the VIC driver, the MTU was derived from either a Cisco UCS Manager service profile or from the Cisco IMC vNIC MTU setting in non-cluster setup. This behavior changes on Cisco UCS VIC 1400 Series and later adapters, where MTU is controlled from the Windows OS Network Direct Maximum Transmission Unit advanced property. A value configured from Cisco UCS Manager or Cisco IMC has no effect.
- RoCEv2 MTU is derived from the Ethernet MTU.
- RoCEv2 MTU is always a power-of-two value and must not exceed the Ethernet MTU.

Windows NDPKI Modes of Operation

- The 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.

Windows Requirements

Configuration and use of RDMA over Converged Ethernet for RoCEv2 in Windows Server requires the following:

- Windows 2019 and later versions with latest Microsoft updates
- UCS Manager release 4.1.1 or later
- VIC Driver version 5.4.0.x or later
- UCS M5 or later generation of X-Series, B-Series ,or C-Series servers equipped with VIC 1400, 14000, or 15000 Series adapters.



Note All Powershell commands or advanced property configurations are common across all Windows versions unless explicitly mentioned.

RoCEv2 for Linux

Guidelines for Using NVMe over Fabrics (NVMeoF) with RoCEv2

General Guidelines and Limitations

- Cisco recommends you check UCS Hardware and Software Compatibility specific to your Cisco UCS Manager release to determine support for NVMeoF.
- NVMe over RDMA with RoCEv2 is supported with Cisco UCS VIC 1400 Series, 14000, and Cisco UCS VIC 15000 Series adapters. NVMe over RDMA is not supported on Cisco UCS 6324 Fabric Interconnects or on Cisco UCS VIC 1200 Series and Cisco 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.
- RoCEv2 cannot be used with GENEVE offload.
- 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, Netflow, and DPDK features.
- Set the QoS No Drop CoS value to 5. (For example: platinum enable with no drop).

  For detailed configuration on upstream Nexus switches, see FlashStack Oracle RAC 19c with NVMe over RoCE CVD.
- Set the MTU size consistently on all VLANs and set the appropriate QoS policies on upstream switches. The MTU must not exceed the maximum supported value of the storage arrays.
- 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.
- Netflow monitoring is not supported on RoCEv2 interfaces.
- 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.
- Cisco UCS Manager does not support fabric failover for vNICs with RoCEv2 enabled.

Interrupt Requirements

- Linux RoCEv2 interfaces support only MSI-X interrupt mode. Do not change the interrupt mode of an interface configured for RoCEv2.

  A minimum of eight interrupts is required for RoCEv2 on Linux.

A minimum of eight interrupts is required for RoCEv2 on Linux.

Downgrade Limitations

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

Linux Requirements

Configuration and use of RoCEv2 in Linux requires the following:

- InfiniBand kernel API module ib_core
- Red Hat Enterprise Linux 8.x, 9.x, and 10.x versions
- Cisco UCS Manager release 4.1.1 or later
- Minimum VIC firmware 5.1(1x) for IPv4 support and 5.1(2x) for IPv6 support
- Cisco UCS M5 and later B-Series, C-Series, and X-Series servers using Cisco UCS VIC 1400, 14000, or 15000 Series adapters
- eNIC driver version 4.0.0.10-802.34 or later
- enic_rdma driver version 1.0.0.10-802.34 or later
- A storage array that supports NVMeoF connection

NFS Over RDMA

Guidelines for Using NFS over RDMA

Network File System (NFS) over Remote Direct Memory Access (RDMA) enables the NFS protocol to run over high-speed RDMA-capable fabrics using RoCEv2, significantly reducing Central Processing Unit (CPU) overhead.

General Guidelines and Limitations

- Cisco recommends you check UCS Hardware and Software Compatibility specific to your Cisco UCS Manager release to determine support for NFS.
- NFS over RDMA is supported on Cisco UCS X-Series, B-Series, and C-Series M6 and later servers equipped with Cisco UCS VIC 15000 Series adapters.
- NFS over RDMA is supported for NFS version 4.1 and later (depending on the ONTAP version).
- NFS over RDMA supports a maximum of two RDMA-enabled interfaces per Cisco VIC adapter.
- When creating RDMA-capable vNICs, use Cisco UCS Manager provided Linux-NVMe-RoCE adapter policy. Do not use the default Linux Adapter policy.
- In Linux-NVMe-RoCE adapter policy, do not change values of Queue Pairs, Memory Regions, Resource Groups, and Priority settings other than to Cisco provided default values.
- When configuring NFS over RDMA using RoCEv2 interfaces, install the matched set of enic and enic_rdma binary drivers downloaded from Cisco.com. Attempting to use the binary enic_rdma driver with an inbox enic driver is not supported.
- Set the QoS CoS value to 'no drop', 5 (Platinum). For detailed configuration on upstream Nexus switches, see FlashStack Oracle RAC 19c with NVMe over RoCE CVD.
- Set the MTU size consistently on all VLANs and set the appropriate QoS policies on upstream switches. The MTU must not exceed the maximum supported value of the storage arrays.

- NFS over RDMA does not support fabric failover, Layer 3 routing, bonding, Netflow monitoring, or GENEVE offload.
- NFS over RDMA cannot be used with usNIC, VMFEX, VxLAN, VMQ, VMMQ, NVGRE, and DPDK features.
- Booting from NFS over RDMA and saving a crashdump to NFS over RDMA during a system crash are not supported.

Interrupts

- NFS Over RDMA supports only MSIx interrupt mode.
- The minimum interrupt count is 256.
- Cisco recommends not to change the interrupt mode once NFS Over RDMA interface is configured.

Downgrade Limitations

Before downgrading to a non-supported release version, it is recommended to remove any RDMA-capable vNIC configurations.

Linux Requirements

Configuration and use of Network File System (NFS) over Remote Direct Memory Access (RDMA) in Linux requires the following:

- Red Hat Enterprise Linux (RHEL) 9.7 or later, and 10.1 or later
- NetApp storage arrays (ONTAP 9.17.1P1 or later)
- Cisco UCS Manager release 6.0.2b or later
- Minimum VIC firmware 5.4(2x)
- InfiniBand kernel API module ib_core
- eNIC driver version 4.10.366.0-1233.0 or later
- `enic_rdma` driver version 1.10.366.0-1233.0 or later

Note You must install a matched set of `enic` and `enic_rdma` drivers. Using the binary `enic_rdma` driver with an inbox `enic` driver is not supported.

- A storage array that supports NFS Over RDMA connection

Note Do not use the default Linux Adapter Policy; RDMA interfaces will not be created in the Linux OS.

RoCEv2 for ESXi

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

General Guidelines and Limitations

- Cisco UCS Manager release 4.2(3b) and later.
- ESXi 7.0 U3, ESXi 8.0, ESXi 8.0 U1, ESXi 8.0 U2, ESXi 8.0 U3, and ESXi 9.0 and later.
- Cisco recommends you check UCS Hardware and Software Compatibility specific to your Cisco UCS Manager release to determine support for ESXi. RoCEv2 on ESXi is supported on Cisco UCS X-Series, 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, 1400, 14000 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 and GENEVE offload can be configured on different interfaces on the same Cisco UCS VIC adapter.
- RoCEv2 cannot be used on the same vNIC interface with VXLAN, Geneve Offload, QinQ, and VMQ.
- RoCEv2 supports maximum two RoCEv2 enabled interfaces per adapter.
- Pvrdma, VSAN over RDMA, and iSER are not supported.
- Set the QoS No Drop CoS value to 5. (For example: platinum enable with no drop).

  For detailed configuration on upstream Nexus switches, see FlashStack Oracle RAC 19c with NVMe over RoCE CVD.

Downgrade Limitations

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

ESXi nENIC RDMA Requirements

Configuration and use of RoCEv2 in ESXi requires the following:

- VMware ESXi 7.0 U3, 8.0, 9.0, and later
- Cisco UCS Manager release 4.2.3 or later
- Cisco VMware nENIC driver version 2.0.10.0 for ESXi 7.0 U3 and 2.0.11.0 for ESXi 8.0 and later provide both standard eNIC and RDMA support
- A storage array that supports NVMeoF connection. Currently, tested and supported on Pure Storage with Cisco Nexus 9300 Series switches.

SRIOV for ESXi

Guidelines and Limitations

- Cisco recommends that you check UCS Hardware and Software Compatibility specific to your Cisco UCS Manager release to determine support for SR-IOV.
- SR-IOV is supported with Cisco UCS VIC 1400 series, 15000 series, and later series adapters. SR-IOV is not supported on Cisco UCS VIC 1200 and 1300 series adapters.
- SR-IOV is supported with Cisco UCS AMD®/Intel® based C-Series, B-Series, and X-Series servers.
- SR-IOV cannot be configured on the same vNIC with VXLAN, Geneve Offload, QinQ, VMQ/VMMQ, RoCE, or usNIC.
- aRFS is not supported on SR-IOV VF.
- iSCSI boot is not supported on SR-IOV VF.
- DPDK on SRIOV VF is not supported when the host has Linux OS.
- SR-IOV interface supports MSIx interrupt mode.
- Precision Time Protocol (PTP) is not supported on SR-IOV VF.
- Cisco recommends not do downgrade the adapter firmware to lower than 5.3(2.32) and to remove SR-IOV related configurations before downgrading Cisco UCS Manger to non-supported SR-IOV release.
- For Cisco UCS VIC 1400/14000, Receive Side Scaling (RSS) must be enabled on PF to support VF RSS.

Note RSS turned off on PF disables the RSS on all VFs.

- For Cisco UCS VIC 15000 series adapters, turning off the RSS on PF works on all the VFs.

Note The PF and VF RSS are independent of each other. The VF driver enables and configures RSS on VF, when there are multiple RQs.

- On ESXi hosts configured with SR-IOV vNICs and Virtual Machines (VMs) utilizing enumerated Virtual Functions (VFs), the system might experience a Purple Screen of Death (PSOD) during cold or warm reboot operations.

ESXi Requirements

Configuration and use of SR-IOV in ESXi requires the following:

- Cisco UCS Manager release 4.3(2b) or later
- Cisco VIC firmware version 5.3(2.32) or later

- VMware ESXi 7.0 U3, 8.0, 9.0, and later
- VMs with RHEL 8.7 or later and RHEL 9.0 or later
- Cisco VMware nENIC driver version 2.0.10.0 for ESXi 7.0 U3, 2.0.11.0 for ESXi 8.0 U3, and 2.0.18.0 for ESXi 9.0 or later
- Cisco RHEL ENIC driver version 4.4.0.1-930.10 or later

SRIOV for Linux

Guidelines and Limitations

- Cisco recommends that you check UCS Hardware and Software Compatibility specific to your Cisco UCS Manager release to determine the support for SR-IOV.
- SR-IOV is supported with Cisco UCS VIC 1400, 14000, 15000 series adapters. SR-IOV is not supported on Cisco UCS VIC 1200 and 1300 series adapters.
- SR-IOV is supported with AMD®/Intel® based Cisco UCS C-Series, B-Series, and X-Series servers.
- SR-IOV is not supported in Physical NIC mode.
- SR-IOV does not support VLAN Access mode.
- SR-IOV cannot be configured on the same vNIC with VXLAN, Geneve Offload, QinQ, VMQ/VMMQ, RoCE, or usNIC.
- aRFS is not supported on SR-IOV VF.
- iSCSI boot is not supported on SR-IOV VF.
- DPDK on SRIOV VF is not supported when the host has Linux OS.
- SR-IOV interface supports MSIx interrupt mode.
- Precision Time Protocol (PTP) is not supported on SR-IOV VF.
- Cisco recommends that you do not downgrade the adapter firmware to lower than 5.3(2.32) and to remove SR-IOV related configurations before downgrading Cisco UCS Manager to non-supported SR-IOV release.
- On Linux hosts with SR-IOV configured on vNICs and Virtual Functions (VFs) enumerated, VFs are not persistent across reboots and must be recreated after each system reboot.

Linux Requirements

Configuration and use of SR-IOV in Linux requires the following:

- Host OS: Red Hat Enterprise Linux 8.10 or later, 9.4 or later, Ubuntu 22.0.4.2 LTS or later
- Guest OS: Red Hat Enterprise Linux 8.10, 9.4, Ubuntu 22.0.4.2 LTS or later
- Virtualization Packages installed on the host

- eNIC driver version 4.7.0.5-1076.6 or later
- Cisco UCS Manager Release 4.3(5a) or later
- Cisco VIC firmware 5.3(4.75) or later

Express Data Path (XDP)

Express Data Path (XDP) for Linux

Express Data Path (XDP) enables supported Cisco UCS Virtual Interface Card (VIC) adapters to process packets earlier in the Linux networking stack through the Cisco eNIC driver. This capability improves packet-processing throughput and reduces networking overhead. XDP is supported on servers managed by Cisco UCS Manager, while its configuration and operation are managed at the Linux host level through the operating system, kernel, and Cisco eNIC driver.

Guidelines and Limitations

- Starting with Cisco UCS Manager Release 6.0(3a), the Cisco eNIC driver supports Express Data Path (XDP) on Cisco UCS VIC 1400, 14000, and 15000 Series adapters.
- XDP operates through the supported Linux operating system, kernel, and Cisco eNIC driver stack rather than through the server management platform. Therefore, XDP support does not depend on the server management mode.
- Configure and manage XDP on the Linux host. No Cisco UCS Manager GUI or CLI configuration is required.

Linux Requirements

XDP support requires the following hardware and software:

- Cisco UCS VIC 15000 Series adapter
- Linux kernel version 6.12 or later
- One of the following operating systems:
  - Red Hat Enterprise Linux 10.x or later
  - Rocky Linux 10.x or later
  - Ubuntu 24.04.4 or later
- Cisco eNIC driver version `6.0.3.XXXXXX` or later

NDIS Poll Mode support for Windows

Guidelines for Using NDIS Poll Mode for Windows

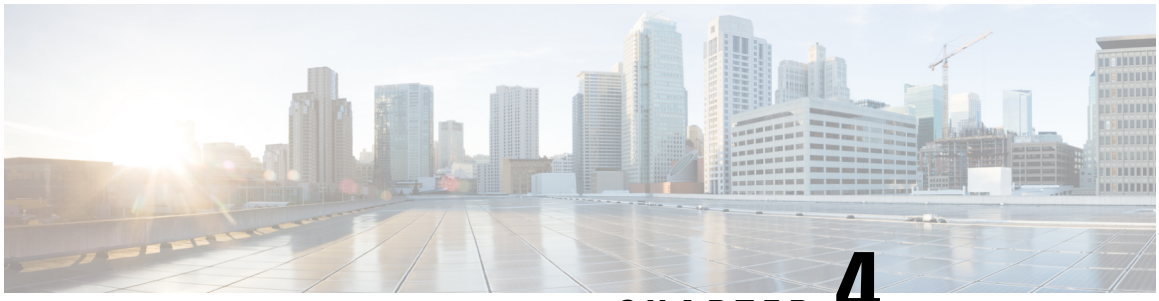
NDIS Poll Mode is enabled by default in supported Cisco VIC driver packages. To modify settings, use the `Set-NetAdapterAdvancedProperty` PowerShell cmdlet or navigate to the **Advanced** tab in the Device Manager properties for the specific Cisco VIC adapter.

Guidelines

- Supported on Windows Server 2025 and later versions
- Supported on nENIC version 5.15.17.4
- Supported on Cisco UCS VIC 15000 series adapters

Note NDIS Poll Mode is enabled by default in the supported driver packages. No additional configuration is required for the feature to function as intended.

For additional information on NDIS Poll Mode and its underlying mechanisms in Windows drivers, refer to Microsoft's Windows Driver and Network documentation.



CHAPTER 4

Configuring RDMA Over Converged Ethernet Using RoCEv2

- Configuring RoCEv2 in Windows , on page 17
- Configuring RoCEv2 in Linux, on page 23
- Configuring RoCEv2 in ESXi, on page 30
- Using the UCS Manager CLI to Configure or Delete the RoCEv2 Interface , on page 38

Configuring RoCEv2 in Windows

Configuring RoCEv2 Modes 1 and 2 in Windows

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

To configure RoCEv2 mode 1, you must:

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

RoCEv2 Mode 1 must be configured before configuring Mode 2.

To configure RoCEv2 mode 2, you must:

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

Configuring SMB Direct Mode 1 on Cisco UCS Manager

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

Before you begin

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

Procedure

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

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

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

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

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

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

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

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

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

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

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

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

Step 9 Click **Save Changes**.

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

What to do next

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

Configuring SMB Direct Mode 1 on the Host System

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

Before you begin

Configure RoCEv2 for Mode 1 in Cisco UCS Manager.

Procedure

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

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

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

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

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

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

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

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

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

`Set-NetOffloadGlobalSetting -NetworkDirect enabled`

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

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> Get-SmbClientNetworkInterface
Interface Index    RSS Capable    RDKA Capable    Speed    IpAddresses    Friendly Name
---------------    ------------   ------------   -------   -----------  ---------------
14                 True           False          40 Gbps    {10.37.60.162}    vEthernet
(vswitch)
26                 True           True           40 Gbps    {10.37.60.158}    vEthernet
(vpl)
9                  True           True           40 Gbps    {50.37.61.23}     Ethernet 2
5                  False          False          40 Gbps    {169.254.10.S}    Ethernet
(Kernel Debugger)
8                  True           False          40 Gbps    {169.254.4.26}    Ethernet 3
PS C:\Users\Administrator>
```

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

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

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

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> netstat -xan
Active NetworkDirect Connections, Listeners, SharedEndpoints
```

```
Mode    IfIndex    Type    Local Address    Foreign Address    PID
Kernel    9        Listener  50.37.61.23:445    NA             0
Kernel    26       Listener  10.37.60.158:445   NA             0
PS C:\Users\Administrator>
```

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

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

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

```
PS C:\Users\Administrator> nctstat -xan
Active NetworkDirect Connections, Listeners, SharedEndpoints
Mode    IfIndex    Type    Local Address        Foreign Address    PID
Kernel   4    Connection    50.37.61.22:445    50.37.61.71:2240    0
Kernel   4    Connection    50.37.61.22:445    50.37.61.71:2496    0
Kernel   11   Connection    50.37.61.122:445   50.37.61.71:2752    0
Kernel   11   Connection    50.37.61.122:445   50.37.61.71:3008    0
Kernel   32   Connection    10.37.60.155:445   50.37.60.61:49092   0
Kernel   32   Connection    10.37.60.155:445   50.37.60.61:49348   0
Kernel   26   Connection    50.37.60.32:445    50.37.60.61:48580   0
Kernel   26   Connection    50.37.60.32:445    50.37.60.61:48836   0
Kernel   4    Listener      50.37.61.22:445    NA                  0
Kernel   11   Listener      50.37.61.122:445   NA                  0
Kernel   32   Listener      10.37.60.155:445   NA                  0
Kernel   26   Listener      50.37.60.32:445    NA                  0
```

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

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

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

Configuring Mode 2 on Cisco UCS Manager

You will apply the VMQ Connection Policy as vmmq.

Before you begin

Configure RoCEv2 Policies in Mode 1.

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

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

- Resource Groups: 2
- Priority: Platinum

Create a VMQ connection policy with the following values:

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

Procedure

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

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

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

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

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

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

Step 5 Click **Save Changes**.

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

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

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

Step 9 Click **Save Changes**.

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

What to do next

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

Configuring SMB Direct Mode 2 on the Host System

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

Before you begin

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

Procedure

Step 1 Go to the Hyper-V switch manager.

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

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

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

Bring up the Powershell interface.

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

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

enable-netAdapterRdma -name "vEthernet (vp1"

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

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

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

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

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

```
get-netadapterrdma

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

You will see the new vport when you again enter

```
get-netadapterrdma
```

b) Add a vport:

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

You see the new vport when you again enter:

```
get-netadapterrdma
```

c) Enable the RDMA on the vport:

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

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

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

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

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

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

Step 6 Finally, verify the Mode 2 configuration.

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

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

b) Start any RDMA I/O in the file share in smb-client.

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

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

Configuring RoCEv2 in Linux

Configuring NVMeoF Using RoCEv2 on Cisco UCS Manager

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

Before you begin

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

Direct-attached Pure Storage configuration

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

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

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

Procedure

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

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

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

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

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

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

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

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

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

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

Step 7 Click **OK**.

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

Step 8 Click **Save Changes**.

What to do next

Enabling SRIOV BIOS Policy, on page 24

Enabling SRIOV BIOS Policy

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

Procedure

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

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

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

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

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

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

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

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

Step 8 Click **Save Changes**.

Configuring NVMeoF Using RoCEv2 on the Host

Before you begin

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

Procedure

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

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

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

Step 3 Save the file.

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

- For Legacy boot:

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

- For UEFI boot:

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

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

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

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

Note its inclusion at the end of the output.

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

What to do next

Download the enic and enic_rdma drivers.

Installing Cisco enic and enic_rdma Drivers

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

Procedure

Step 1 Install the enic and enic_rdma rpm packages:

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

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

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

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

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

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

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

Step 4 Load the vme-rdma kernel module:

```
# modprobe nvme-rdma
```

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

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

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

What to do next

Discover targets and connect to NVMe namespaces. If your system needs multipath access to the storage, please go to the section for Setting Up Device Mapper Multipath, on page 28.

Discovering the NVMe Target

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

Before you begin

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



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

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

Procedure

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

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

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

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

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

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

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

For example, to discover the target at 50.2.85.200:

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

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

Note

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

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

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

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

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

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

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

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

Setting Up Device Mapper Multipath

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

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

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

Procedure

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

Step 2 Enable and start multipathd:

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

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

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

Step 4 Flush with the updated multipath device maps.

```
# multipath -F
```

Step 5 Restart multipath service:

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

Step 6 Rescan multipath devices:

```
# multipath -v2
```

Step 7 Check the multipath status:

```
# multipath -ll
```

Deleting the RoCEv2 Interface Using Cisco UCS Manager

Use these steps to remove the RoCE v2 interface

Procedure

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

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

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

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

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

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

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

c) Click OK.

Step 5 Click **Save Changes**.

Configuring RoCEv2 in ESXi

Configuring NVMEoF using RoCEv2 for ESXi on UCS Manager

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

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

Before you begin

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

Direct-attached Pure Storage configuration

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

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

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

Procedure

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

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

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

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

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

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

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

Perform the below steps to modify the vNIC policy:

a) Name the new VNIC.

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

Step 6 Click **Save Changes**.

What to do next

Install the NENIC Driver.

Installing NENIC Driver

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

These steps assume this is a new installation.

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

Procedure

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

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

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

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

or

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

Example:

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

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

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

Upgrading NENIC Driver

a. To upgrade NENIC driver, enter the command:

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

or

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

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

What to do next

Configure the ESXi Host side NVMe RDMA.

ESXi NVMe RDMA Host Side Configuration

NENIC RDMA Functionality

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

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

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

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

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

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

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

Create Network Connectivity Switches

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

Before you begin

Ensure that you have downloaded and installed the NENIC driver.

Procedure

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

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

Step 3 Click on **Add Networking**.

The available network adapter connection types are:

- **Vmkernel Network Adapter**

  Creates a new VMkernel adapter to handle host management traffic

- **Physical Network Adapter**

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

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

  Creates a new port group for virtual machine networking.

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

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

Step 6 Add physical adapters to the new standard switch.

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

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

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

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

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

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

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

f) Configure IPV4 and/or IPV6 settings.

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

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

What to do next

Create vmhba ports on top of vmrdma ports.

Create VMHBA Ports in ESXi

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

Before you begin

Create the adapter ports for storage connectivity.

Procedure

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

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

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

Add Software Adapter dialog box is displayed.

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

Step 5 Click **OK**

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

What to do next

Configure NVMe.

Displaying vmnic and vmrdma Interfaces

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

Before you begin

Create Network Adapters and VHBA ports.

Procedure

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

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

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

esxcli network nic list

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

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

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

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

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

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

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

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

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

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

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

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

```
[root@RackServer:~] esxcli storage core adapter list
HBA Name Driver   Link State UID                                   Capabilities
Description
-------- ------   ---------- ----------------------------------- --------------------
-------------------------------------
vmhbao   nfnic    link-down  fc.10002cf89b798dbe:20002cf89b798dbe Second Level Lun ID
(0000:1d:00.2) Cisco Corporation Cisco
                                                                                    UCS
 VIC Fnic Controller
vmhba1   vmw_ahci link-n/a   sata.vmhba1
(0000:00:11.5) Intel Corporation Lewisburg

SATA AHCI Controller
vmhba2   nfnic    link-down  fc.10002cf89b798dbf:20002cf89b798dbf Second Level Lun ID
(0000:1d:00.3) Cisco Corporation Cisco
                                                                                    UCS
 VIC Fnic Controller
vmhba3   nfnic    link-down  fc.10002cf89b51b33c:20002cf89b51b33c Second Level Lun ID
(0000:63:00.2) Cisco Corporation Cisco
                                                                                    UCS
 VIC Fnic Controller
vmhba4   nfnic    link-down  fc.10002cf89b51b33d:20002cf89b51b33d Second Level Lun ID
(0000:63:00.3) Cisco Corporation Cisco
                                                                                    UCS
 VIC Fnic Controller
vmhba5   lsi_mr3  link-n/a   sas.5cc167e9732f9b00
(0000:3c:00.0) Broadcom Cisco 126 Modular

Raid Controller with 2GB cache
vmhba64  nvmerdma link-n/a   rdma.vmnic2:2c: f8:9b:79:8d:bc                        VMware
 NVMe over RDMA Storage Adapter on vmrdma0
vmhba65  nvmerdma link-n/a   rdma.vmnic3:2c:f8:9b:79:8d:bd                        VMware
 NVMe over RDMA Storage Adapter on vmrdma1
```

What to do next

Configure NVME.

NVMe Fabrics and Namespace Discovery

This procedure is performed through the ESXi command line interface.

Before you begin

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

Procedure

Step 1 Check and enable NVMe on the vmrdma device.

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

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

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

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

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

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

You will see output on the NVMe controller.

Step 3 Perform NVMe fabric interconnect.

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

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

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

`esxcli nvme controller list RDMA -d vmrdma0`

```
[root@RackServer:~] esxcli nvme controller list
Name                                    Controller Number Adapter  Transport Type Is
Online
----------------------------------------  -----------------  --------  --------------  ---------
nqn.2010-06.com.purestorage: flasharray.  258               vmhba64  RDMA           true
5ab274df5b161455#vmhba64#50.2.84.100:4420
nqn.2010-06.com.purestorage: flasharray.  259               vmhba65  RDMA           true
Sab274df5b161455#vmhba65#50.2.83.100:4420
[root@RackServer:~] esxcli nvme namespace list
Name                                Controller Number Namespace ID Block Size Capacity in
 MB
----------------------------------- ----------------  ------------ ---------- --------------
```

```
eui.00e6d65b65a8f34824a9374e00011745 258               71493        512        102400
eui.00e6d65b65a8f34024a9374e00011745 259               71493        512        102400
```

Example

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

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

flasharray:2dp1239anjkl484
[root@RackServer:~] esxcli nvme fabrics discover -a vmhba64 -l 50.2.84.100 p 4420 -s
nq.210-06.com.
purestorage:flasharray:2dp1239anjkl484 Controller already connected
```

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

Configure Windows SMB Direct RoCEv2 Interface using UCS Manager CLI

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

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

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

Enter the vnic mode for the specified vNIC.

Step 4 To configure Windows SMBDirect RoCEv2 Mode 1:

Example:

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

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

Step 5 To configure Windows SMBDirect RoCEv2 Mode 2:

Example:

```
UCS-A# scope org
UCS-A /org # create vmq-conn-policy policy name
UCS-A /org/vmq-conn-policy* # set multi-queue enabled
UCS-A /org/vmq-conn-policy* # set vmmq-sub-vnic-count 64
UCS-A /org/vmq-conn-policy* # set vmmq-adaptor-profile-name MQ-SMBd
UCS-A /org/vmq-conn-policy* # commit-buffer
UCS-A /org/vmq-conn-policy #
```

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

Step 6 **Example:**

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

Commit the transaction to the system configuration.

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

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

vNIC:

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


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

Deleting the Windows RoCEv2 Interface Using the CLI for UCS Manager

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

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

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

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

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

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

Step 5 **Example:**

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

Commit the transaction to the system configuration.

What to do next

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

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

vNIC:

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


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

Configuring the Linux RoCEv2 Interface Using the UCS Manager CLI

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

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

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

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

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

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

Step 5 **Example:**

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

Commit the transaction to the system configuration.

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

Example

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

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

Deleting the Linux RoCEv2 Interface Using the UCS Manager CLI

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

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

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

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

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

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

Step 5 **Example:**

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

Commit the transaction to the system configuration.

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

Example

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

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

Configuring the VMware ESXi RoCEv2 Interface Using the UCS Manager CLI

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

Before you begin

You must log in with admin privileges.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

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

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

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

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

Step 5 **Example:**

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

Commit the transaction to the system configuration.

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

Example

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

vNIC:
    Name               Fabric ID Dynamic MAC Addr   Virtualization Preference
    ------------------ --------- ------------------ -------------------------
    eth00              A B       00:25:B5:3A:84:00  NONE
    eth01              A         00:25:B5:3A:84:01  NONE
    eth02              B         00:25:B5:3A:84:02  NONE
UCS-A /org/service-profile # scope vnic eth01
UCS-A /org/service-profile/vnic # set adapter-policy VMWareNVMeRoCEv2
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

Deleting the ESXi RoCEv2 Interface Using UCS Manager

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

Procedure

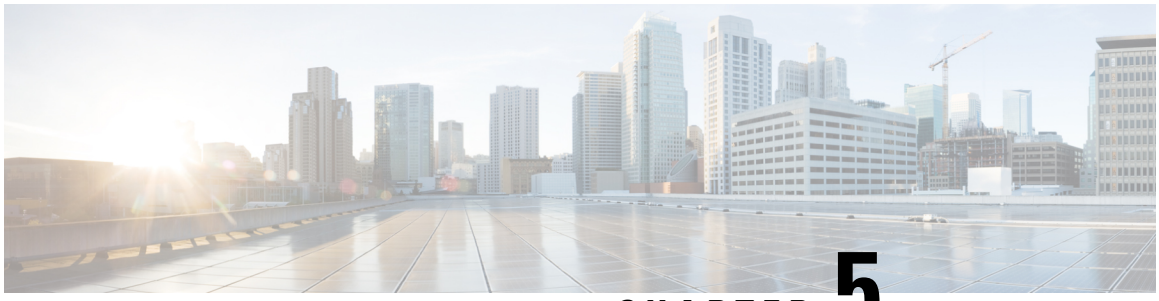
Step 1 In the **Navigation** pane, click **Servers**.

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

Step 3 Expand the node for the profile to delete.

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

Step 5 Click **Save Changes**.



CHAPTER 5

Configuring NFS Over RDMA Using ROCEv2

- Configuring NFS over RDMA on Cisco UCS Manager, on page 45
- Deleting the NFS over RDMA Interface Using Cisco UCS Manager, on page 46
- Configuring the NFS over RDMA Interface Using the UCS Manager CLI, on page 47
- Deleting the NFS over RDMA Interface Using the Cisco UCS Manager CLI, on page 48
- Configuring NFS over RDMA on the Linux Host (AMD), on page 49
- Configuring NFS over RDMA on the Linux Host (Intel), on page 50
- Installing Cisco eNIC and enic_rdma Drivers, on page 51
- Mounting and Verifying NFS over RDMA Volumes, on page 52

Configuring NFS over RDMA on Cisco UCS Manager

Use the following steps to configure the Remote Direct Memory Access (RDMA) over Converged Ethernet version 2 (RoCEv2) interface on Cisco UCS Manager to support Network File System (NFS) over RDMA.

Before you begin

The **Linux-NVMe-RoCE** adapter policy is already defined in the system.

Procedure

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

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

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

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

Step 4 Select the desired **Service Profile**.

Step 5 Click **vNICs** and go to the **Network** tab in the work area.

Step 6 Modify the vNIC policy:

a) In the **vNICs** area, select the desired vNIC and click **Modify**

b) In the **Modify vNIC** dialog box, scroll down to the **Adapter Performance Profile** area.

c) Click the **Adapter Policy** drop-down and choose **`Linux-NVMe-RoCE`**.

d) Click **OK**.

Step 7 Click **Save Changes**.

Step 8 Select **Reboot** to apply the changes to the server.

The Cisco UCS Manager applies the new adapter policy to the server, and the server initiates a reboot to finalize the configuration.

What to do next

After the server reboots, proceed to configure the host-side NFS over RDMA settings as described in the host configuration procedure.

Deleting the NFS over RDMA Interface Using Cisco UCS Manager

Use the following steps to remove the Network File System (NFS) over Remote Direct Memory Access (RDMA) configuration from a specific vNIC.

Procedure

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

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

Step 3 Expand the node for the organization where the service profile is located. (If the system does not include multitenancy, expand the root node).

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

a) Click on **vNICs** and go to the **Network** tab in the work area.
b) Scroll down to the desired vNIC, select it, and click **Modify**.
c) In the popup dialog box, scroll down to the **Adapter Performance Profile** area.
d) From the **Adapter Policy** drop-down list, choose **Linux** (or the appropriate standard adapter policy for your environment) to remove the `Linux-NVMe-RoCE` configuration.
e) Click **OK**.

Step 5 Click **Save Changes** and reboot the server.

What to do next

The Cisco UCS Manager applies the standard adapter policy to the server and reboot.

Configuring the NFS over RDMA Interface Using the UCS Manager CLI

Use the following steps to configure the Network File System (NFS) over Remote Direct Memory Access (RDMA) for Linux in the Cisco UCS Manager Command Line Interface (CLI).

Before you begin

- Cisco UCS Manager is installed and accessible.
- The server is associated with a Service Profile.
- The **Linux-NVMe-RoCE** adapter policy is already defined in the system.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

```
UCS-A /org/service-profile # scope vnic eth0
```

Enter the vnic mode for the specified vNIC (in this example, `eth0`).

Step 4 **Example:**

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

Specifies the adapter policy for the vNIC used for NFS over RDMA.

Step 5 **Example:**

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

Commit the transaction to the system configuration.

The Cisco UCS Manager applies the new adapter policy to the server, and the server initiates a reboot to finalize the configuration.

Example

This example shows how to configure the *NFS over RDMA* adapter policy on the `eth01` vNIC:

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

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

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

What to do next

- The Cisco UCS Manager applies the new adapter policy to the server, and the server initiates a reboot to finalize the configuration.
- After the server reboots, proceed to configure the host-side NFS over RDMA settings.

Deleting the NFS over RDMA Interface Using the Cisco UCS Manager CLI

Use the following steps to remove the Network File System (NFS) over Remote Direct Memory Access (RDMA) interface configuration in the Cisco UCS Manager CLI by reverting the adapter policy to the default Linux policy.

Before you begin

You must be logged in with administrative privileges.

Procedure

Step 1 **Example:**

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

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

Step 2 **Example:**

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

Display the vNICs available on the server.

Step 3 **Example:**

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

Enter the vnic mode for the specified vNIC.

Step 4 **Example:**

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

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

Step 5 **Example:**

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

Commit the transaction to the system configuration.

Step 6 Reboot the server to fully remove the RDMA-enabled interface from the host operating system.

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

Example

Removing the NFS over RDMA Interface

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

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

What to do next

Reboot the server to remove *Linux-NVMe-RoCE* adapter policy and apply the new policy.

Configuring NFS over RDMA on the Linux Host (AMD)

Before you begin

- Ensure the server is configured with RoCEv2 vNICs.
- Ensure you have administrative (root or sudo) privileges to the Linux host.
- Verify that the host system is AMD-based.

Use this procedure to configure NFS over RDMA on the Red Hat Enterprise Linux (RHEL) host for AMD-based systems.

Procedure

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

Step 2 Add `amd_iommu=on` to the end of the line for `GRUB_CMDLINE_LINUX` as shown in the following sample:

Sample:

```
# cat /etc/default/grub
                          GRUB_CMDLINE_LINUX="... rhgb quiet amd_iommu=on"
```

Step 3 Save the file.

Step 4 Generate the new GRUB configuration file (grub.cfg) for UEFI systems:

For UEFI boot:

Sample:

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

Step 5 Reboot the server to apply the changes.

```
# reboot
```

Note: The server must be rebooted for the IOMMU changes to take effect.

Step 6 Verify that the server booted with the `amd_iommu=on` option.

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

The output should confirm the inclusion of `amd_iommu=on` at the end of the line.

Sample Output:

```
BOOT_IMAGE=(hd0,gpt2)/vmlinuz-6.12.0-124.8.1.el10_1.x86_64 ... rhgb quiet amd_iommu=on
```

The IOMMU option is enabled, and the system is prepared for the installation of the necessary RDMA drivers.

What to do next

- Proceed to install the downloaded `enic` and `enic_rdma` drivers.
- Configure your NFS mount points to utilize the RDMA interface as described in the mounting procedure.

Configuring NFS over RDMA on the Linux Host (Intel)

Before you begin

- Ensure the server is configured with RoCEv2 vNICs.
- Ensure you have administrative (root or sudo) privileges to the Linux host.
- Verify that the host system is Intel-based.

Use this procedure to configure NFS over RDMA on the Red Hat Enterprise Linux (RHEL) host (Intel-based system).

Procedure

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

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

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

Step 3 Save the file.

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

- For Legacy boot:

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

- For UEFI boot:

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

Step 5 Reboot the server.

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

Step 6 Verify that the server booted with the `intel_iommu=on` option: `# cat /proc/cmdline | grep iommu. The output should confirm the inclusion of intel_iommu=on at the end of the line.`

```
Sample Output: The output should confirm the inclusion of intel_iommu=on at the end of the
 line.

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

The IOMMU option is enabled, and the system is prepared for the installation of the necessary RDMA drivers.

What to do next

- Proceed to install the downloaded `enic` and `enic_rdma` drivers.
- Configure your NFS mount points to utilize the RDMA interface.

Installing Cisco eNIC and enic_rdma Drivers

Use the following steps to install Cisco eNIC and enic_rdma Drivers for NFS over RDMA.

Before you begin

- Ensure you have downloaded the matched set of eNIC and enic_rdma drivers from the Cisco support portal.

- Verify that the target server is installed with the supported RHEL version.
- Ensure you have administrative (root or sudo) privileges to perform driver installation.

Note Using the `enic_rdma` driver binary with an inbox driver is not supported. Ensure you are using the matched set of drivers provided by Cisco.

Procedure

Step 1 Install the eNIC and enic_rdma driver package on the host: `# rpm -ivh kmod-enic-<version>.x86_64.rpm kmod-enic_rdma-<version>.x86_64.rpm`

Step 2 Reboot the server to load the drivers into the running kernel: `# reboot`

Step 3 Verify that the drivers are loaded and RoCEv2 is enabled by checking the system logs: `# dmesg | grep enic_rdma`.

The system should display the following output:

```
[    1.562913] enic_rdma: Cisco VIC Ethernet NIC RDMA Driver, ver 1.10.366.0-1233.0 init
[    1.687502] enic 0000:75:00.0 enp117s0f0: enic_rdma: FW v4 RoCEv2 enabled
[    1.771968] enic 0000:75:00.1 enp117s0f1: enic_rdma: FW v4 RoCEv2 enabled
```

The eNIC and enic_rdma drivers are successfully installed and loaded. The system logs confirm that the Cisco VIC interfaces are initialized and RoCEv2 functionality is enabled on the respective ports.

What to do next

After verifying the driver installation, proceed to configure the NFS over RDMA mount points.

Mounting and Verifying NFS over RDMA Volumes

This procedure provides instructions for mounting and verifying NFS volumes using the RDMA protocol on the host operating system.

Before you begin

- Ensure the entire network infrastructure is configured to support NFS over RDMA traffic. The configuration must be consistent across the Cisco VIC adapter, upstream network switches, and the storage target.
- Confirm Cisco VIC firmware and drivers (**enic** and **enic_rdma**) are up to date and compatible with RDMA and RoCEv2.
- Ensure MTU is set appropriately on the VIC, host OS, and network switches to optimize RDMA performance.

- Confirm Cisco UCS Manager or IMC is configured correctly for RoCEv2, including enabling RoCE properties on the vNIC and disabling incompatible failover features.
- Validate that RDMA interfaces are up and configured with correct IP addressing matching the storage network.

Procedure

Step 1 Create a local directory to serve as the mount point:

```
# mkdir /<mount_point>
```

Sample

```
# mkdir /mnt/nfs_volume1
```

Step 2 Mount the NFS volume using the RDMA protocol:

- **IPv4:**

  ```
  mount -o proto=rdma <storage IPV4>:/<nfs volume>/<mount_point>
  ```

  Sample

  ```
  mount -o proto=rdma 192.168.1.100:/mnt/nfs_volume1
  ```

- **IPv6:**

  ```
  mount -o proto=rdma6 '[IPv6_address]:/<volume path>'/<mount_point>
  ```

  Sample

  ```
  mount -o proto=rdma6 '[20:34:d0:27:ea:19:70:50]:/<volume path>'/<mount point>
  ```

  Replace <IPv4_address>/<IPv6_address>, <volume_path>, and <mount_point> with your actual IP addresses, NFS export path, and desired local mount directory.

Step 3 Verify that the NFS over RDMA mount is created and active. Run the following command to filter the mount list for RDMA-specific connections:

```
# mount | grep rdma
```

Sample Output

```
50.19.40.5:/mnt/nfs_volume1 type nfs4 (rw, relatime, vers=4.1, rsize=65536, wsize=65536,
namlen=255, hard, fatal_neterrors=none, proto=rdma, nconnect=4, port=20049, timeo=600,
retrans=2, sec=sys, clientaddr=, local_lock=none, addr=)
```

Step 4 Verify the volume capacity and mount status using the df command. Use the `df` command to ensure the volume is correctly mapped and reporting the expected storage capacity:

```
# df -h
```

Sample Output

```
<target IP>:/nfs_vol    501G  3.6G  497G   1% /mnt/nfs_volume1
```

Step 5 Check the RDMA statistics to verify traffic flow:

This command provides NFS statistics on the mount, confirming RDMA usage and showing incrementing counts during active traffic.

```
# mountstats --xprt <mount_point>
```

Step 6 To unmount the volume, use the following command:

```
# umount <mount_point>
```

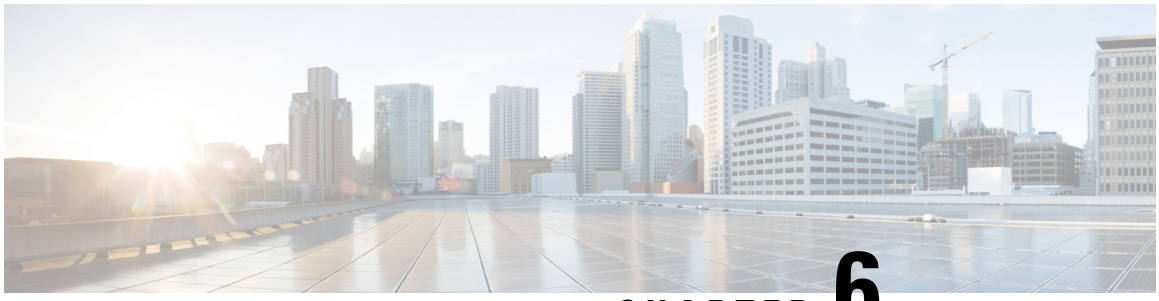
What to do next

After successfully mounting and verifying the NFS over RDMA volume, perform the following tasks:

- Configure Persistent Mounting: To ensure the volume mounts automatically after a system reboot, add an entry to the `/etc/fstab` file. *Example entry:*

  ```
  <storage_ip>:/<volume_path> <mount_point> nfs rdma,nconnect=4,x-systemd.automount 0
          0
  ```

- Configure Application Workloads: Update your application or database configurations to point to the new `<mount_point>` to begin utilizing the high-throughput, low-latency RDMA storage.



CHAPTER 6

Configuring Single Root I/O Virtualization (SR-IOV)

- Configuring BIOS and Cisco UCS Manager Parameters , on page 55
- Configuring SR-IOV VFs on the ESXi Host Server , on page 60
- Configuring SR-IOV VFs on the Linux Host Server , on page 66

Configuring BIOS and Cisco UCS Manager Parameters

Enabling BIOS Parameters

Before you begin

- Ensure your BIOS policy is set up with the following options:
  - For Intel based servers, enable **Intel VT for directed IO** under **Intel Directed IO** tab.
  - For AMD based servers, enable **IOMMU** and **SVM Mode** under **Processor** tab and enable **SR-IOV** under **LOM and PCIe Slots BIOS Settings** tab.

  To update BIOS options, see, Cisco UCS Manager Server Management Guide.
- You must have a service profile already created for SR-IOV configuration. To create a Service Profile see Cisco UCS Manager Server Management Guide. Once the Service Profile is created, follow the steps in this procedure to enable the BIOS policy.

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 for which you want to enable SR-IOV BIOS parameters.

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

Step 4 Click the service profile for which you want to enable SR-IOV BIOS parameters.

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

Step 6 On the **Policies** tab, expand **BIOS Policy**.

Step 7 From the **BIOS Policy** drop-down list, select the BIOS policy that you have created for SR-IOV configuration.

Ensure that the BIOS policy selected satisfies the pre-requisites for this procedure.

Step 8 Save changes and click **Yes** to reboot the server.

Enabling SR-IOV VFs using Cisco UCS Manager GUI

To enable SR-IOV from Cisco UCS Manager, you must

- Create an SRIOV HPN Connection Policy with desired number of VFs.
- Assign the SRIOV HPN Connection Policy to a Service Profile.

Before you begin

- Ensure that the required BIOS options are enabled before performing this procedure.

Procedure

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

Step 2 Expand **Policies** > **root**.

Step 3 To create SRIOV HPN Connection Policy, right click **SRIOV HPN Connection Policies**.

Step 4 You can view 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.
Transmit Queues field	The number of descriptors in each transmit queue. Enter an integer between 1 and 8.
Receive Queues field	The number of receive queue resources to allocate. Enter an integer between 1 and 8.

Name	Description
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 16.
Interrupt Count field	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.

Step 5 Provide the policy name with the desire number of SRIOV HPN vNICs and click **OK** to create **SRIOV HPN Connection Policy**.

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

Step 7 Expand **Servers** > **Service Profiles**.

Step 8 Expand the node and service profile for the organization that contains the service profile for SR-IOV configuration.

Step 9 Click the desired service profile for which you wish to apply the SR-IOV VFs.

Step 10 Expand **vNIC** and select the vNIC for which you wish to apply the SR-IOV VFs.

Step 11 In the work pane, select the **General** tab.

Step 12 At the **Adapter Policy** drop-down list, select **SRIOV-HPN**.

Step 13 Under the **Connection Policies** radio buttons, select **SRIOV-HPN**.

Step 14 From the **SRIOV HPN Connection Policy** drop-down list, select the policy you have already created for SR-IOV configuration.

Step 15 Save changes and click **Yes** to reboot the server.

Disabling SR-IOV VFs Using Cisco UCS Manager GUI

Procedure

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

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

Expand the node and service profile for the organization that contains the service profile for SR-IOV configuration.

Step 3 Click the service profile from which you wish to remove the SR-IOV VFs.

Step 4 Expand vNIC and select the vNIC for which you wish to disable the SR-IOV VFs.

Step 5 In the work pane, select the **General** tab.

Step 6 Under the **Connection Policies** radio button options, select **SRIOV-HPN**.

Step 7 From the **SRIOV HPN Connection Policy** drop-down list, select **not set** to remove the SR-IOV connection policy.

Step 8 Save changes and click **Yes** to reboot the server.

Enabling SR-IOV VFs using Cisco UCS Manager CLI

To enable SR-IOV from Cisco UCS Manager, you must

- Create an SRIOV HPN Connection Policy with desired number of VFs.
- Assign the SRIOV HPN Connection Policy to a Service Profile.

Before you begin

- Ensure that the required BIOS options are enabled before performing this procedure.

Procedure

Step 1 UCS-A # **scope org** *org-name*

Enters organization mode for the specified organization. To enter the root organization mode, type / as the org-name.

Step 2 UCS-A /org # **create sriov-hpn-conn-policy** *policy-name*

Specifies the name for the SRIOV HPN connection policy.

Step 3 UCS-A /org/sriov-hpn-conn-policy* # **set sriov-hpn-count** *sriov hpn count*

Specifies the SRIOV HPN vNICs count for the SRIOV HPN connection policy. Enter an integer between 1 and 64.

Step 4 UCS-A /org/sriov-hpn-conn-policy* # **set transmit-queue-count** *transmit queue count*

Specifies the transmit queue count for the SRIOV HPN connection policy. Enter an integer between 1 and 8.

Step 5 UCS-A /org/sriov-hpn-conn-policy* # **set receive-queue-count** *receive queue count*

Specifies the receive queue count for the SRIOV HPN connection policy. Enter an integer between 1 and 8.

Step 6 UCS-A /org/sriov-hpn-conn-policy* # **set completion-queue-count** *completion-queue count*

Specifies the completion queue count for the SRIOV HPN connection policy. 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.

Step 7 UCS-A /org/sriov-hpn-conn-policy* # **set interrupt-queue-count** *interrupt queue count*

Specifies the interrupt count for the SRIOV HPN connection policy. In general, this value should be equal to the number of completion queue resources. Enter an integer between 1 and 16.

Step 8 UCS-A /org/sriov-hpn-conn-policy* # **commit-buffer**

Commits the transaction to the system.

Step 9 UCS-A /org/sriov-hpn-conn-policy* # **exit**

Step 10 UCS-A /org # **scope service-profile** *profile-name-for-sriov-config*

Enters the service profile for the organization that contains the service profile for SR-IOV configuration.

Step 11 UCS-A# scope /org/service-profile # **scope vnic** *eth0/eth1*

Select a vNIC for which you wish to apply the SR-IOV VFs.

Step 12 UCS-A /org/service-profile/vnic # **set adapter-policy SRIOV-HPN**

Sets the adapter policy as **SRIOV HPN**

Step 13 UCS-A /org/service-profile/vnic # **enter sriov-hpn-conn-policy-ref** *sriov_hpn_connection_policy_name*

Assigns the SRIOV HPN connection policy created previously to the vNIC.

Step 14 UCS-A /org/service-profile/vnic/sriov-hpn-conn-policy-ref* # **commit-buffer**

Commits the transaction to the system.

Disabling SR-IOV VFs using Cisco UCS Manager CLI

To disable the SRIOV VFs, you must delete the associated SRIOV HPN connection policy.

Procedure

Step 1 UCS-A # **scope org** *org-name*

Enters organization mode for the specified organization. To enter the root organization mode, type / as the org-name.

Step 2 UCS-A /org # **scope service-profile** *service_profile_name*

Enter the service profile withing which you wish to disable the SRIOV VFs.

Step 3 UCS-A /org/service-profile # **scope vnic** *eth0/eth1*

Select a vNIC for which you wish to apply the SR-IOV VFs.

Step 4 UCS-A /org/service-profile/vnic # **delete sriov-hpn-conn-policy-ref** *sriov_hpn_connection_policy_name*

Deletes the SRIOV HPN Connection policy. This disables the SRIOV VFs.

Step 5 UCS-A /org/service-profile/vnic* # **commit-buffer**

Commits the transaction to the system.

Configuring SR-IOV VFs on the ESXi Host Server

Installing Cisco eNIC Driver

Before you begin

Ensure that the required BIOS parameters and SR-IOV VFs configurations are completed.

Procedure

Step 1 Install the enic driver on the host.

The following example shows the installation of eNIC driver on ESXi:

```
[root@localhost:/vmfs/volumes/645c8bdd-c655e553-8ba0-e8d32272f6c0] esxcli software vib
install -v /vmfs/volumes/C240M7-Standalone/nenic-2.0.10.0-1OEM.800.1.0.20143090.x86_64.vib
--no-sig-check
Installation Result
Message: The update completed successfully, but the system needs to be rebooted for the
changes to be effective.
VIBs Installed: CIS_bootbank_nenic_2.0.10.0-1OEM.800.1.0.20143090
VIBs Removed: CIS_bootbank_nenic_2.0.11.0-1OEM.800.1.0.20143090
VIBs Skipped:
Reboot Required: true
DPU Results:
[root@localhost:/vmfs/volumes/645c8bdd-c655e553-8ba0-e8d32272f6c0]
```

Step 2 Reboot the server to load the enic driver into the running kernel.

Step 3 After reboot, execute the command `esxcli software vib list | grep nenic` to check the driver version.

Verifying the Total Number of SR-IOV VFs Per Ports on the Host

You can verify the total number of SR-IOV VFs in the following two ways:

Procedure

Step 1 Verify by logging into the VMware ESXi Host Client.:

- Login to the VMware ESXi Host Client.
- Execute the following command to check the vNIC with SR-IOV capability:

```
root@localhost:~] esxcli network sriovnic list
Name    PCI Device    Driver  Link  Speed  Duplex  MAC Address         MTU  Description
-----------------------------------------------------------------------------------------------
vmnic0  0000:1b:00.0  nenic   Up    50000  Full    f4:ee:31:30:80:40  1500  Cisco Systems
 Inc Cisco VIC Ethernet NIC
```

The following output shows the number of VF configured on vNIC:

```
[root@localhost:~] esxcli network sriovnic vf list -n vmnic0
VF ID  Active  PCI Address     Owner World ID
0   false  00000:027:00.1   -
1   false  00000:027:00.2   -
2   false  00000:027:00.3   -
3   false  00000:027:00.4   -
4   false  00000:027:00.5   -
5   false  00000:027:00.6   -
6   false  00000:027:00.7   -
7   false  00000:027:01.0   -
```

Step 2 Alternatively, you can also access your host from vSphere vCenter Client.

For more information on configuring SR-IOV VFs on the host, see Creating SR-IOV VFs on the Host.

After you reboot the host server, do the following:

- Login to the ESXi Host Client, and choose **Networking** > **Virtual Switches**.
- Click **Add Standard Virtual Switch**.
- Add a switch name in the **vSwitch Name** field, select the vmnic with SR-IOV capability, and click **Add**.
- In the **Port Groups** tab, click **Add Port Group**.
- In the **Add Port Group** dialog-box, add a new port group and select the switch from the **Virtual Switch** drop-down.

Creating SR-IOV VFs on the Host

Procedure

Step 1 Login to your VMware ESXi Host Client.

Alternatively, you can also access your host from vSphere vCenter Client and browse to **Configure** > **Networking** > **Physical adapters**.

Step 2 Go to **Host** > **Manage** and select the **Hardware** tab.

Step 3 Select **PCI Devices** from the list.

Step 4 From the drop-down list, select **SR-IOV Capable**.

The list shows all the SR-IOV capable devices.

Step 5 Select the vNIC for which you wish to create the VFs.

Step 6 Click **Configure SR-IOV**.

Configure SR-IOV for Cisco VIC Ethernet NIC window is displayed.

Step 7 Perform the following:

Field	Description
Enabled radio button	Select **Yes** to enable the configuration.
Virtual functions field	Number of VFs as configured on SRIOV connection policy that are available for the configuration. Enter an integer between 1 and 64.

Step 8 Click **Save** and then reboot the host server.

Configuring the Switch

Before you begin

Ensure that the SR-IOV VFs are configured.

Procedure

Step 1 Login to your VMware ESXi Host Client.

Step 2 Navigate to **Host** > **Networking** and select the **Virtual switches** tab.

Step 3 Click **Add Standard Virtual Switch**.

Step 4 Enter the name for the switch.

Step 5 Select a SR-IOV Capable Vmnic from the list.

Step 6 Click **Add**.

Step 7 Complete the following:

Field	Description
vSwitch Name field	Enter a suitable name for the virtual switch.
MTU field	Enter the maximum transmission unit. The default is 1500 bytes.
Uplink 1 drop-down list	From the drop-down list, select the PCIe devices for which you created the SR-IOVs.
Link Discovery	From the drop-down list, select the **Mode** and the **Protocol**. **Note** These fields remain as default.

Field	Description
Security	Choose from the following options: • **Promiscuous mode**—**Accept**, **Reject**, or **Inherit from vSwitch**. • **MAC address changes**—**Accept**, **Reject**, or **Inherit from vSwitch**. **Forged trasmits**—**Accept**, **Reject**, or **Inherit from vSwitch**.
NIC teaming	Choose from the following: • **Load balancing**—From the drop-down list choose the Load balancing. Values are: **Inherit from vSwitch**, • **Network failover detection**—From the drop-down list choose the network failover detection. Values are: **Inherit from vSwitch**, • **Notify switches**—Choose the notify switches. Values are **Yes**, **No**, **Inherit from vSwitch**. • **Fallback**—Choose the fallback. Values are **Yes**, **No**, **Inherit from vSwitch**. • **Override failover order**—From the drop-down list choose the override failover order. Values are **Yes** or **No**, • **Failover order**—Choose the failover order.
Traffic Shaping	Perform the following: • **Status**—Choose the status. Values are **Enabled**, **Disabled**, **Inherit from vSwitch**. • **Average bandwidth**—Enter the average bandwidth. • **Peek bandwidth**—Enter the peek bandwidth. • **Burst size**—Enter the burst size. **Note** Traffic shaping policy is applied to the traffic of each virtual network adapter attached to the virtual switch.

What to do next

Creating a Virtual Port, on page 64

Creating a Virtual Port

Before you begin

Ensure that the SR-IOV VFs are configured.

Procedure

Step 1 Login to your VMware ESXi Host Client.

Step 2 Go to **Host** > **Networking** and select the **Port Groups** tab.

Step 3 Click **Add port group**.

Add port group-New port group window is displayed

Step 4 Complete the following:

Field	Description
Name field	Enter a suitable name for the virtual port.
VLAN ID field	Enter the VLAN ID.
Virtual Switch drop-down list	From the drop-down list, select the virtual switch.
Security	Choose from the following options: • **Promiscuous mode**—**Accept**, **Reject**, or **Inherit from vSwitch**. • **MAC address changes**—**Accept**, **Reject**, or **Inherit from vSwitch**. **Forged trasmits**—**Accept**, **Reject**, or **Inherit from vSwitch**.

Step 5 Click **Add**.

Creating a New Virtual Machine (VM)

Before you begin

- Host with Desktop Environment
- sudo user with admin rights
- Virtualization packages are installed
- OS ISO image is copied to the host server

Procedure

For more information, refer to Installing OS on Guest VM on ESXi, on page 65.

Adding SR-IOV VF on the Virtual Machine

Before you begin

Power off the Virtual Machine.

Procedure

Step 1 In the Virtual Machine Manager, right-click on the Virtual Machine and select **Open**.

Step 2 Click the **Show Virtual Hardware Detail** icon next to **Monitor** icon.

Step 3 Click **Add Hardware**.

Step 4 In the **Add New Virtual Hardware** window, select **PCI Host Device**. Under the **PCI Device Details** tab, assign a created SR-IOV VF to the Virtual Machine.

Step 5 Click **Finish**.

Step 6 Power on the Virtual Machine.

What to do next

You can now log into the virtual machine, install Cisco eNIC driver 4.7.0.5-1076.6 or later version, reboot the virtual machine, and then use the ip link command to verify the added SR-IOV VF.

Installing OS on Guest VM on ESXi

Before you begin

Upload the Linux operating system ISO on the datastore.

Procedure

Step 1 Right-click the host node and navigate to **vCenter** > **New Virtual machine**.

Step 2 Select a **Creation Type** > **Create New Virtual Machine**, and click **Next**.

Step 3 Enter a name for the folder, and click **Next**.

Step 4 Select a compute resource, choose a node and click **Next**.

Step 5 Select Storage and check the datastore radio-button, and click **Next**.

Step 6 Select the compatibility ESXi 8.0 or later and click **Next**.

Step 7 Select a guest OS version as **RHEL Linux9 (64-bit)**, and click **Next**.

Step 8 Customize the hardware set **CPU** to 2, and **Memory values** to 4 GB.

Step 9 Expand the **Memory** tab, and check **Reserve all guest memory (All locket)** check box.

Step 10 Select **New CD/DVD Drive** *(Datastore ISO file)*, and check the **Connect At Power On** check box.

Step 11 Under **CD/DVD Media**, browse and select the Linux ISO image and click **Next**.

Step 12 Click **Finish**.

Configuring SR-IOV VFs on the Linux Host Server

Installing Cisco eNIC Driver

Before you begin

Ensure that the required BIOS parameters and SR-IOV VFs configurations are completed.

Procedure

Step 1 Install the enic driver on the host.

Following example shows the installation of eNIC driver on RHEL:

```
[user@rack-111 drivers]# rpm -ivh kmod-enic-4.7.0.5-1076.6.rhel9u4_5.14.0_427.13.1.x86_64.rpm
Verifying...                          ################################# [100%]
Preparing...                          ################################# [100%]
Updating / installing...
   1:kmod-enic-4.7.0.5-1076.6.rhel9u4_################################# [100%]
[user@rack-111 drivers]#
```

Step 2 Reboot the server to load the enic driver into the running kernel.

Step 3 Execute **modinfo enic** to check enic driver is loaded.

Following example shows the output of **modinfo enic** command:

```
[user@rack-111 drivers]# modinfo enic
filename:       /lib/modules/5.14.0-427.13.1.el9_4.x86_64/extra/enic/enic.ko
version:        4.7.0.5-1076.6
retpoline:      Y
license:        GPL v2
author:         Scott Feldman scofeldm@cisco.com
description:    Cisco VIC Ethernet NIC Driver
rhelversion:    9.4
srcversion:     3A1B1E81C9641925B34D1B2
alias:          pci:v00001137d000002B7sv*sd*bc*sc*i*
alias:          pci:v00001137d00000071sv*sd*bc*sc*i*
alias:          pci:v00001137d00000044sv*sd*bc*sc*i*
alias:          pci:v00001137d00000043sv*sd*bc*sc*i*
depends:
retpoline:      Y
name:           enic
vermagic:       5.14.0-427.13.1.el9_4.x86_64 SMP preempt mod_unload modversions
sig_id:         PKCS#7
signer:         Cisco UCS Driver Signing REL Cert
```

```
sig_key:        D0:54:9A:88:88:DD:0E:7A
sig_hashalgo:   sha256
signature:      89:9C:DA:53:D1:FF:0A:DA:98:9A:7F:AF:63:29:66:EB:FF:0C:D6:65:
                39:6C:15:40:30:6E:99:4B:2C:F0:54:2E:EB:A4:8A:33:D5:9C:41:7A:
                A4:DB:C8:52:55:74:3A:68:F3:22:36:7B:2A:7C:7C:40:8B:7F:6D:9E:
                A5:CF:06:F1:23:42:E6:60:DB:78:0E:46:C9:0C:BC:06:9B:02:A0:AA:
                5A:FC:36:A3:FB:B0:FE:76:F2:EB:2F:AD:AD:84:89:61:30:7D:E9:2F:
                5D:E1:3E:EA:7C:10:B2:42:94:CD:4F:74:19:A6:16:FE:75:B6:78:49:
                E8:F0:4A:A9:01:BB:92:44:A9:FE:C7:CE:DB:E8:F5:08:AF:36:1E:5F:
                30:D3:B1:5F:70:62:56:6F:C2:38:8E:F2:88:28:0F:44:29:E5:44:66:
                34:B7:5C:A7:5E:21:C3:5D:42:D8:C0:87:CA:40:5E:C4:C0:2C:DA:26:
                D2:25:9B:58:A8:84:C6:A6:41:B3:24:9C:D7:E6:4A:79:42:00:32:82:
                7A:CB:36:D8:79:1D:41:1A:9E:1C:A8:0D:39:6D:C8:F1:0D:44:FA:00:
                93:1E:A3:C9:61:AA:DE:25:4A:38:68:C3:9C:14:55:5B:D3:AC:1C:85:
                00:FE:57:F1:DE:F7:A8:04:64:0E:5D:35:D8:AF:CF:A4
parm:           rxcopybreak:Maximum size of packet that is copied to a new buffer on receive
 (uint)
[user@rack-111 drivers]#
```

Verifying the Total number of SR-IOV VFs per Port on the Host

Before you begin

Ensure that Cisco eNIC driver is installed.

Procedure

Log into the host server and run the following command and replace *interface_name* with actual interface name on the host.

 # cat /sys/class/net/*interface_name*/device/sriov_totalvfs

Example

Following example shows the total number for SR-IOV VFs created from SRIOV HPN Connection Policy on p1p1 interface:

```
[user@rack-111 ~]# cat /sys/class/net/p1p1/device/sriov_totalvfs
32
[user@rack-111 ~]#
```

Creating SR-IOV VFs on the Host

Enabling SR-IOV VFs from SRIOV HPN Connection Policy does not create SR-IOV VFs on the host by default. To create SR-IOV VFs on the host, use the following procedure:

Procedure

Step 1 Execute the following command to create SR-IOV VFs on the host:

 # echo number_of_sriov_devices > /sys/class/net/*sriov interface_name***/device/sriov_numvfs**

Example:

Following example shows the creation of 6 SR-IOV VFs on p1p1 interface.

```
[user@rack-111 ~]# echo 6 > /sys/class/net/p1p1/device/sriov_numvfs
[user@rack-111 ~]#
```

Step 2 Execute the following command to verify the SR-IOV VFs created:

 # cat /sys/class/net/interface_name/device/sriov_numvfs

Example:

Following example shows the verification of SR-IOV VFs on p1p1 interface.

```
[user@rack-111 ~]# cat /sys/class/net/p1p1/device/sriov_numvfs
6
[user@rack-111 ~]#
```

Step 3 (Optional) Alternatively, IP link command shows created SR-IOV VFs.

 # ip link show interface_name

Example:

Following example shows created 6 SR-IOV VFs on p1p1 interface.

```
[user@rack-111 ~]# ip link show pipl
2: plpl:  <BROADCAST, MULTICAST, UP, LOWER_UP>mtu 9000 qdisc mq state UP mode
DEFAULT group default qlen 10 00
link/ether 98: a2:c0:66:32:80 brd ff:ff:ff:ff:ff:ff
vf 0 link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff, spoof checking off,
link-state auto, trust off, query_rss off
vf 1 link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff, spoof checking off,
link-state auto, trust off, query_rss off
vf 2 link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff, spoof checking off,
link-state auto, trust off, query_rss off
vf 3 link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff, spoof checking off,
link-state auto, trust off, query_rss off
vf 4 link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff, spoof checking off,
link-state auto, trust off, query_rss off
vf 5 link/ether 00:00:00:00:00:00 brd ff:ff:ff:ff:ff:ff, spoof checking off,
link-state auto, trust off, query_rss off
altname enp9s0
altname eno5
[user@rack-111 ~]#
```

Note

After the host server reboots, the created SR-IOV VFs are removed from the host. By adding the command from Step 1 to rc.local file, the same number of SR-IOV VFs can be created each time the host server boots up.

What to do next

You can create a new virtual machine.

Creating a New Virtual Machine (VM)

Before you begin

- Host with Desktop Environment
- sudo user with admin rights
- Virtualization packages are installed
- OS ISO image is copied to the host server

Procedure

Step 1 Verify the virtualization is enabled on the host server by using this command.

 # lscpu | grep Virtualization

Example:

This example shows the Intel's virtualization technology VT-x is enabled.

```
[user@rack-111 ~]$ lscpu | grep Virtualization
Virtualization: VT-x
[user@rack-111 ~]$
```

Step 2 Verify the KVM modules are loaded by using this command.

 # lsmod | grep kvm

Example:

This example shows KVM modules are loaded in the host server.

```
[user@rack-111 ~]$ lsmod | grep kvm
kvm_intel    409600    8
kvm         1134592    1 kvm_intel
irqbypass      6384    290 vfio_pci_core, kvm
[user@rack-111 ~]$
```

Step 3 Type **virt-manager** command at the terminal to launch Virtual Machine Manager GUI.

Step 4 At the Virtual Machine Manager, click **File** > **New Virtual Machine** to create a new virtual machine.

Step 5 At **New VM window**, select **Local install media (ISO image or CDROM)** option and click **Forward**.

Step 6 At **Choose ISO or CDROM install media**, click **Browse**.

Step 7 At **Locate ISO media volume** window, click **Browser Local**.

Step 8 Go to the folder that has ISO image. Select ISO image and click **Open**.

Step 9 Click **Forward**.

Step 10 Select the desire Memory and CPU settings for the VM and click **Forward**.

Step 11 Choose the VM's disk image size and click **Forward**.

Step 12 Enter a name for the VM in the **Name** field and click **Finish**.

You may monitor the OS installation progress.

Adding SR-IOV VF on the Virtual Machine

Before you begin

Power off the Virtual Machine.

Procedure

Step 1 In the Virtual Machine Manager, right-click on the Virtual Machine and select **Open**.

Step 2 Click the **Show Virtual Hardware Detail** icon next to **Monitor** icon.

Step 3 Click **Add Hardware**.

Step 4 In the **Add New Virtual Hardware** window, select **PCI Host Device**. Under the **PCI Device Details** tab, assign a created SR-IOV VF to the Virtual Machine.

Step 5 Click **Finish**.

Step 6 Power on the Virtual Machine.

What to do next

You can now log into the virtual machine, install Cisco eNIC driver 4.7.0.5-1076.6 or later version, reboot the virtual machine, and then use the ip link command to verify the added SR-IOV VF.