



Marvell[®] Converged Network Adapters 41000 Series Adapters

Converged Network Adapters

User's Guide



Third party information brought to
you courtesy of Dell.

THIS DOCUMENT AND THE INFORMATION FURNISHED IN THIS DOCUMENT ARE PROVIDED "AS IS" WITHOUT ANY WARRANTY. MARVELL AND ITS AFFILIATES EXPRESSLY DISCLAIM AND MAKE NO WARRANTIES OR GUARANTEES, WHETHER EXPRESS, ORAL, IMPLIED, STATUTORY, ARISING BY OPERATION OF LAW, OR AS A RESULT OF USAGE OF TRADE, COURSE OF DEALING, OR COURSE OF PERFORMANCE, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.

This document, including any software or firmware referenced in this document, is owned by Marvell or Marvell's licensors, and is protected by intellectual property laws. No license, express or implied, to any Marvell intellectual property rights is granted by this document. The information furnished in this document is provided for reference purposes only for use with Marvell products. It is the user's own responsibility to design or build products with this information. Marvell products are not authorized for use as critical components in medical devices, military systems, life or critical support devices, or related systems. Marvell is not liable, in whole or in part, and the user will indemnify and hold Marvell harmless for any claim, damage, or other liability related to any such use of Marvell products.

Marvell assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. You may not use or facilitate the use of this document in connection with any infringement or other legal analysis concerning the Marvell products disclosed herein. Marvell and the Marvell logo are registered trademarks of Marvell or its affiliates. Please visit www.marvell.com for a complete list of Marvell trademarks and guidelines for use of such trademarks. Other names and brands may be claimed as the property of others.

Copyright

Copyright © 2021. Marvell and/or its affiliates. All rights reserved.

Table of Contents

Preface

Supported Products	xix
Intended Audience	xx
What Is in This Guide	xx
Documentation Conventions	xxi
Legal Notices	xxiv
Laser Safety—FDA Notice.	xxiv
Agency Certification.	xxiv
EMI and EMC Requirements	xxiv
KCC: Class A	xxv
VCCI: Class A.	xxvi
Product Safety Compliance.	xxvi

1 Product Overview

Functional Description	1
Features	1
Adapter Specifications.	3
Physical Characteristics	3
Standards Specifications.	3

2 Hardware Installation

System Requirements	5
Hardware Requirements	5
Software Requirements	6
Safety Precautions	6
Preinstallation Checklist	7
Installing the Adapter	7

3 Driver Installation

Installing Linux Driver Software	9
Installing the Linux Drivers Without RDMA	11
Removing the Linux Drivers	11
Installing Linux Drivers Using the src RPM Package	13

Installing Linux Drivers Using the kmp/kmod RPM Package . . .	14
Installing Linux Drivers Using the TAR File.	15
Installing the Linux Drivers with RDMA	15
Linux Driver Optional Parameters	16
Linux Driver Operation Defaults	17
Linux Driver Messages	17
Statistics	19
Linux RoCE MTU.	19
Importing a Public Key for Secure Boot.	20
Installing Windows Driver Software	21
Installing the Windows Drivers	21
Running the DUP in the GUI.	21
DUP Installation Options.	27
DUP Installation Examples	28
Removing the Windows Drivers	28
Managing Adapter Properties	28
Setting Power Management Options.	30
Configuring the Communication Protocol to Use with QCC GUI, QCC PowerKit, and QCS CLI	30
Link Configuration in Windows	31
Link Control Mode	32
Link Speed and Duplex	33
FEC Mode	33
Installing VMware Driver Software	34
VMware Drivers and Driver Packages.	35
Installing VMware Drivers	36
Installing or Upgrading the Standard Driver	36
Installing the Enhanced Network Stack Poll Mode Driver.	37
VMware NIC Driver Optional Parameters	39
VMware Driver Parameter Defaults.	40
Removing the VMware Driver	42
FCoE Support	43
iSCSI Support	43
Installing Citrix Hypervisor Driver Software	44
4	Upgrading the Firmware
Running the DUP by Double-Clicking	47
Running the DUP from a Command Line.	49
Running the DUP Using the .bin File	50

5	Adapter Preboot Configuration	
	Getting Started	53
	Default NPar/NParEP Mode Numbering	56
	PCI Device IDs	60
	Displaying Firmware Image Properties	60
	Configuring Device-level Parameters.	61
	Configuring NIC Parameters	62
	Configuring Data Center Bridging	65
	Configuring FCoE Boot	67
	Configuring iSCSI Boot	68
	Configuring Partitions.	73
	Partitioning for VMware ESXi 6.7 and ESXi 7.0	77
6	Boot from SAN Configuration	
	iSCSI Boot from SAN	79
	iSCSI Out-of-Box and Inbox Support.	80
	iSCSI Preboot Configuration	80
	Setting the BIOS Boot Mode to UEFI	81
	Enabling NPar and the iSCSI HBA.	83
	Configuring the Storage Target.	83
	Selecting the iSCSI UEFI Boot Protocol.	84
	Configuring iSCSI Boot Options	85
	Configuring the DHCP Server to Support iSCSI Boot	97
	Configuring iSCSI Boot from SAN on Windows	101
	Before You Begin	102
	Selecting the Preferred iSCSI Boot Mode	102
	Configuring iSCSI General Parameters	102
	Configuring the iSCSI Initiator	103
	Configuring the iSCSI Targets	104
	Detecting the iSCSI LUN and Injecting the Marvell Drivers	104
	Configuring iSCSI Boot from SAN on Linux	106
	Configuring iSCSI Boot from SAN for RHEL 7.8 and Later	107
	Configuring iSCSI Boot from SAN for SLES 12 SP3 and Later	109
	Configuring iSCSI Boot from SAN for SLES 15 SP1 and Later	109
	Configuring iSCSI Boot from SAN on VMware	110
	Setting the UEFI Main Configuration	110
	Configuring the System BIOS for iSCSI Boot (L2)	112
	Mapping the CD or DVD for OS Installation	114

FCoE Boot from SAN	116
FCoE Out-of-Box and Inbox Support.	116
FCoE Preboot Configuration	117
Specifying the BIOS Boot Protocol.	117
Enabling NPar and the FCoE HBA.	117
Configuring Adapter UEFI Boot Mode	118
Configuring FCoE Boot from SAN on Windows	123
Windows Server 2016 and 2019/Azure Stack HCI FCoE Boot Installation	123
Configuring FCoE on Windows	124
FCoE Crash Dump on Windows.	124
Injecting (Slipstreaming) Adapter Drivers into Windows Image Files.	125
Configuring FCoE Boot from SAN on Linux	125
Prerequisites for Linux FCoE Boot from SAN.	125
Configuring Linux FCoE Boot from SAN	126
Configuring FCoE Boot from SAN on VMware	129
Injecting (Slipstreaming) ESXi Adapter Drivers into Image Files Installing the Customized ESXi ISO	129 130
Viewing Boot Statistics	133

7 **RoCE Configuration**

Supported Operating Systems and OFED	134
Planning for RoCE	135
Preparing the Adapter	136
Preparing the Ethernet Switch	136
Configuring the Cisco Nexus 6000 Ethernet Switch	137
Configuring the Dell Z9100 Ethernet Switch for RoCE	138
Configuring RoCE on the Adapter for Windows Server	140
Viewing RDMA Counters	144
Configuring RoCE for SR-IOV VF Devices (VF RDMA)	149
Configuration Instructions.	149
Limitations	157
Configuring RoCE on the Adapter for Linux	158
RoCE Configuration for RHEL.	158
RoCE Configuration for SLES	159
Verifying the RoCE Configuration on Linux.	160
vLAN Interfaces and GID Index Values.	162

RoCE v2 Configuration for Linux	163
Identifying the RoCE v2 GID Index or Address	163
Verifying the RoCE v1 or RoCE v2 GID Index and Address from sys and class Parameters	164
Verifying the RoCE v1 or RoCE v2 Function Through perftest Applications	165
Configuring RoCE for SR-IOV VF Devices (VF RDMA)	168
Enumerating VFs for L2 and RDMA	169
Number of VFs Supported for RDMA	171
Limitations	172
Configuring RoCE on the Adapter for VMware ESX	173
Configuring RDMA Interfaces	173
Configuring MTU	175
RoCE Mode and Statistics	175
Configuring a Paravirtual RDMA Device (PVRDMA)	176
Configuring the RoCE Namespace	179
Configuring DCQCN	180
DCQCN Terminology	180
DCQCN Overview	181
DCB-related Parameters	182
Global Settings on RDMA Traffic	182
Setting vLAN Priority on RDMA Traffic	182
Setting ECN on RDMA Traffic	182
Setting DSCP on RDMA Traffic	182
Configuring DSCP-PFC	183
Enabling DCQCN	183
Configuring CNP	183
DCQCN Algorithm Parameters	183
MAC Statistics	184
Script Example	184
Limitations	185

8 iWARP Configuration

Preparing the Adapter for iWARP	186
Configuring iWARP on Windows	187
Configuring iWARP on Linux	191
Installing the Driver	191
Configuring iWARP and RoCE	191
Detecting the Device	192
Supported iWARP Applications	193

	Running PerfTest for iWARP	194
	Configuring NFS-RDMA	195
9	iSER Configuration	
	Before You Begin	197
	Configuring iSER for RHEL	198
	Configuring iSER for SLES 15 and Later	201
	Using iSER with iWARP on RHEL and SLES	202
	Optimizing Linux Performance	203
	Configuring CPUs to Maximum Performance Mode	203
	Configuring Kernel sysctl Settings	204
	Configuring IRQ Affinity Settings	204
	Configuring Block Device Staging	204
	Configuring iSER on ESXi 6.7 and ESXi 7.0	205
	Before You Begin	205
	Configuring iSER for ESXi 6.7 and ESXi 7.0	205
10	iSCSI Configuration	
	iSCSI Boot	209
	iSCSI Offload in Windows Server	210
	Installing Marvell Drivers	210
	Installing the Microsoft iSCSI Initiator	210
	Configuring Microsoft Initiator to Use Marvell's iSCSI Offload	210
	iSCSI Offload FAQs	216
	Windows Server 2016 and 2019/Azure Stack HCI iSCSI Boot Installation	217
	iSCSI Crash Dump	218
	iSCSI Offload in Linux Environments	218
	Differences from bnx2i	219
	Configuring qedi.ko	219
	Verifying iSCSI Interfaces in Linux	219
	iSCSI Offload in VMware ESXi	221
11	FCoE Configuration	
	Configuring Linux FCoE Offload	228
	Differences Between qedf and bnx2fc	229
	Configuring qedf.ko	229
	Verifying FCoE Devices in Linux	230

12	SR-IOV Configuration	
	Configuring SR-IOV on Windows	232
	Configuring SR-IOV on Linux	239
	Enabling IOMMU for SR-IOV in UEFI-based Linux OS Installations.	244
	Configuring SR-IOV on VMware	245
13	NVMe-oF Configuration with RDMA	
	Installing Device Drivers on Both Servers	252
	Configuring the Target Server	253
	Configuring the Initiator Server.	255
	Preconditioning the Target Server	256
	Testing the NVMe-oF Devices	257
	Optimizing Performance.	258
	IRQ Affinity (multi_rss-affin.sh)	259
	CPU Frequency (cpufreq.sh).	260
	Configuring NVMe-oF on ESXi 7.0.	260
14	VXLAN Configuration	
	Configuring VXLAN in Linux.	262
	Configuring VXLAN in VMware	264
	Configuring VXLAN in Windows Server 2016	265
	Enabling VXLAN Offload on the Adapter.	265
	Deploying a Software Defined Network.	266
15	Windows Server 2016	
	Configuring RoCE Interfaces with Hyper-V	267
	Creating a Hyper-V Virtual Switch with an RDMA NIC	268
	Adding a vLAN ID to Host Virtual NIC	269
	Verifying If RoCE is Enabled	270
	Adding Host Virtual NICs (Virtual Ports)	271
	Mapping the SMB Drive.	271
	IPv4 Network Drive Mapping	271
	IPv6 Network Drive Mapping	271
	Running RoCE Traffic	272
	Using IPv6 Addressing for RoCE (and iWARP) on SMB Direct/S2D	274
	RoCE over Switch Embedded Teaming	274
	Creating a Hyper-V Virtual Switch with SET and RDMA Virtual NICs	275
	Enabling RDMA on SET	275

Assigning a vLAN ID on SET	276
Running RDMA Traffic on SET	276
Configuring QoS for RoCE	276
Configuring QoS by Disabling DCBX on the Adapter	276
Configuring QoS by Enabling DCBX on the Adapter	281
Configuring VMMQ	284
Enabling VMMQ on the Adapter	285
Creating a Virtual Machine Switch with or Without SR-IOV	285
Enabling VMMQ on the Virtual Machine Switch	286
Getting the Virtual Machine Switch Capability	287
Creating a VM and Enabling VMMQ on VMNetworkAdapters in the VM	287
Enabling and Disabling VMMQ on a Management NIC	288
Monitoring Traffic Statistics	288
Configuring Storage Spaces Direct	288
Configuring the Hardware	289
Deploying a Hyper-Converged System	289
Deploying the Operating System	290
Configuring the Network	290
Configuring Storage Spaces Direct	292

16

Windows Server 2019/ Azure Stack HCI

RSSv2 for Hyper-V	296
RSSv2 Description	296
Known Event Log Errors	297
Windows Server 2019/Azure Stack HCI Behaviors	297
VMMQ Is Enabled by Default	297
Inbox Driver Network Direct (RDMA) Is Disabled by Default	298
New Adapter Properties	298
Max Queue Pairs (L2) Per VPort	298
Network Direct Technology	299
Virtualization Resources	299
VMQ and VMMQ Default Accelerations	300
Single VPort Pool	300

17	Troubleshooting	
	Troubleshooting Checklist	302
	Verifying that Current Drivers Are Loaded	303
	Verifying Drivers in Windows	303
	Verifying Drivers in Linux	303
	Verifying Drivers in VMware	304
	Testing Network Connectivity	304
	Testing Network Connectivity for Windows	304
	Testing Network Connectivity for Linux	305
	Atypical FCoE Configurations	305
	Atypical Configurations	306
	Multiple FCoE VLANs	306
	Fallback FCoE VLAN	306
	Multiple FCFs in a Single Fabric	306
	Multiple Fabric (With One or More FCFs)	306
	Troubleshooting Atypical FCoE Configurations	307
	Windows	307
	Linux	307
	VMware ESX (Driver v2.x.y.z and Later)	307
	VMware ESX (Driver v1.x.y.z and Later)	308
	Linux-specific Issues	308
	Miscellaneous Issues	309
	Collecting Debug Data	309
A	Adapter LEDs	
B	Cables and Optical Modules	
	Supported Specifications	311
	Tested Cables and Optical Modules	312
	Known Issue: Using SmartAN Mode with Invalid FEC Configuration for 25G DAC	316
	Tested Switches	317
C	Dell Z9100 Switch Configuration	

D	VMware ESXi Enhanced Networking Stack Support	
	Overview	320
	What is Enhanced Network Stack?	320
	Poll Mode Driver	321
	Capabilities	321
	Features	322
	Limitations	322
	Installing and Configuring an ENS-capable N-VDS	322
	Prerequisites	323
	Preparing the Host to Become Part of the NSX-T Fabric	323
	Creating an Uplink Profile	325
	Configuring the Transport Zone	327
	Creating an IP Pool	328
	Creating the Transport Node	330
	Creating the Logical Switch	332
	Unified Enhanced Network Stack (UENS)	335
	Configuring ENS Using a Command Line Interface	336
	Reference Documents	337
E	Feature Constraints	
F	Revision History	
	Glossary	

List of Figures

Figure		Page
3-1	Dell Update Package Window	22
3-2	QLogic InstallShield Wizard: Welcome Window	22
3-3	QLogic InstallShield Wizard: License Agreement Window.	23
3-4	InstallShield Wizard: Setup Type Window	24
3-5	InstallShield Wizard: Custom Setup Window	25
3-6	InstallShield Wizard: Ready to Install the Program Window	25
3-7	InstallShield Wizard: Completed Window	26
3-8	Dell Update Package Window	27
3-9	Setting Advanced Adapter Properties	29
3-10	Power Management Options	30
3-11	Setting Driver Controlled Mode	32
3-12	Setting the Link Speed and Duplex Property	33
3-13	Setting the FEC Mode Property	34
3-14	Starting in Shell Mode	44
3-15	Installing the qede, qed, qedi, and qedf Drivers.	44
3-16	Loading the Device Driver	45
3-17	Locating the Device Driver.	45
3-18	Driver Installed Successfully	45
3-19	Loading the OOB Driver.	46
4-1	Dell Update Package: Splash Screen	47
4-2	Dell Update Package: Loading New Firmware	48
4-3	Dell Update Package: Installation Results	48
4-4	Dell Update Package: Finish Installation	49
4-5	DUP Command Line Options.	50
5-1	System Setup.	53
5-2	System Setup: Device Settings	53
5-3	Main Configuration Page	54
5-4	Main Configuration Page, Setting Partitioning Mode to NPar	54
5-5	Firmware Image Properties	60
5-6	Device Level Configuration	61
5-7	NIC Configuration.	62
5-8	System Setup: Data Center Bridging (DCB) Settings	66
5-9	FCoE General Parameters.	67
5-10	FCoE Target Configuration.	68
5-11	iSCSI General Parameters.	70
5-12	iSCSI Initiator Configuration Parameters	71
5-13	iSCSI First Target Parameters	71
5-14	iSCSI Second Target Parameters	72
5-15	NIC Partitioning Configuration, Global Bandwidth Allocation.	73
5-16	Global Bandwidth Allocation Page	74
5-17	Partition 1 Configuration.	75
5-18	Partition 2 Configuration: FCoE Offload.	76
5-19	Partition 3 Configuration: iSCSI Offload.	77

5-20	Partition 4 Configuration	77
6-1	System Setup: Boot Settings	82
6-2	System Setup: Device Settings	83
6-3	System Setup: NIC Configuration	84
6-4	System Setup: NIC Configuration, Boot Protocol.	85
6-5	System Setup: iSCSI Configuration	86
6-6	System Setup: Selecting General Parameters.	86
6-7	System Setup: iSCSI General Parameters	87
6-8	System Setup: Selecting iSCSI Initiator Parameters	89
6-9	System Setup: iSCSI Initiator Parameters	90
6-10	System Setup: Selecting iSCSI First Target Parameters	91
6-11	System Setup: iSCSI First Target Parameters	92
6-12	System Setup: iSCSI Second Target Parameters	93
6-13	System Setup: Saving iSCSI Changes	94
6-14	System Setup: iSCSI General Parameters	96
6-15	System Setup: iSCSI General Parameters, VLAN ID	101
6-16	Detecting the iSCSI LUN Using UEFI Shell (Version 2).	105
6-17	Windows Setup: Selecting Installation Destination	105
6-18	Windows Setup: Selecting Driver to Install	106
6-19	Integrated NIC: Device Level Configuration for VMware	111
6-20	Integrated NIC: Partition 2 Configuration for VMware	112
6-21	Integrated NIC: System BIOS, Boot Settings for VMware	112
6-22	Integrated NIC: System BIOS, Connection 1 Settings for VMware	113
6-23	Integrated NIC: System BIOS, Connection 1 Settings (Target) for VMware	114
6-24	VMware iSCSI BFS: Selecting a Disk to Install	115
6-25	VMware iSCSI Boot from SAN Successful	115
6-26	System Setup: Selecting Device Settings	118
6-27	System Setup: Device Settings, Port Selection	119
6-28	System Setup: NIC Configuration	120
6-29	System Setup: FCoE Mode Enabled	121
6-30	System Setup: FCoE General Parameters	122
6-31	System Setup: FCoE General Parameters	123
6-32	Prompt for Out-of-Box Installation	127
6-33	Red Hat Enterprise Linux 7.4 Configuration.	128
6-34	ESXi-Customizer Dialog Box	130
6-35	Select a VMware Disk to Install	131
6-36	VMware USB Boot Options	132
6-37	Boot Session Information	133
7-1	Configuring RoCE Properties.	142
7-2	Add Counters Dialog Box.	144
7-3	Performance Monitor: 41000 Series Adapters' Counters.	146
7-4	Setting an External New Virtual Network Switch	150
7-5	Setting SR-IOV for New Virtual Switch	151
7-6	VM Settings	152
7-7	Enabling VLAN to the Network Adapter	153

7-8	Enabling SR-IOV for the Network Adapter	154
7-9	Upgrading Drivers in VM	155
7-10	Enabling RDMA on the VMNIC	156
7-11	RDMA Traffic	157
7-12	Switch Settings, Server	167
7-13	Switch Settings, Client	167
7-14	Configuring RDMA_CM Applications: Server	168
7-15	Configuring RDMA_CM Applications: Client	168
7-16	Configuring a New Distributed Switch	177
7-17	Assigning a vmknics for PVRDMA	178
7-18	Setting the Firewall Rule	178
8-1	Windows PowerShell Command: Get-NetAdapterRdma	188
8-2	Windows PowerShell Command: Get-NetOffloadGlobalSetting	188
8-3	Perfmon: Add Counters	189
8-4	Perfmon: Verifying iWARP Traffic	190
9-1	RDMA Ping Successful	199
9-2	iSER Portal Instances	199
9-3	Iface Transport Confirmed	200
9-4	Checking for New iSCSI Device	201
9-5	LIO Target Configuration	202
10-1	iSCSI Initiator Properties, Configuration Page	211
10-2	iSCSI Initiator Node Name Change	211
10-3	iSCSI Initiator—Discover Target Portal	212
10-4	Target Portal IP Address	213
10-5	Selecting the Initiator IP Address	214
10-6	Connecting to the iSCSI Target	215
10-7	Connect To Target Dialog Box	216
10-8	Enabling iSCSI Offload Mode in the BIOS	222
10-9	Creating a VMkernel Adapter	222
10-10	Binding the VMkernel Adapter to the iSCSI Partition	223
10-11	Adding Send Target Information	223
10-12	LUN Discovery	224
10-13	Datastore Type	224
10-14	Selecting a LUN for the Datastore	225
10-15	VMFS Version	225
10-16	Partition Configuration Information	226
10-17	Datastore Properties	226
12-1	System Setup for SR-IOV: Integrated Devices	233
12-2	System Setup for SR-IOV: Device Level Configuration	233
12-3	Adapter Properties, Advanced: Enabling SR-IOV	234
12-4	Virtual Switch Manager: Enabling SR-IOV	235
12-5	Settings for VM: Enabling SR-IOV	237
12-6	Device Manager: VM with QLogic Adapter	238
12-7	Windows PowerShell Command: Get-NetadapterSriovVf	238
12-8	System Setup: Processor Settings for SR-IOV	239

12-9	System Setup for SR-IOV: Integrated Devices	240
12-10	Editing the grub.conf File for SR-IOV	241
12-11	Command Output for sriov_numvfs	242
12-12	Command Output for ip link show Command	242
12-13	RHEL68 Virtual Machine	243
12-14	Add New Virtual Hardware	244
12-15	VMware Host Edit Settings	248
13-1	NVMe-oF Network	251
13-2	Subsystem NQN	255
13-3	Confirm NVMe-oF Connection	256
13-4	FIO Utility Installation	257
14-1	Advanced Properties: Enabling VXLAN	265
15-1	Enabling RDMA in Host Virtual NIC	268
15-2	Hyper-V Virtual Ethernet Adapter Properties	269
15-3	Windows PowerShell Command: Get-VMNetworkAdapter	270
15-4	Windows PowerShell Command: Get-NetAdapterRdma	270
15-5	Add Counters Dialog Box	273
15-6	Performance Monitor Shows RoCE Traffic	273
15-7	Windows PowerShell Command: New-VMSwitch	275
15-8	Windows PowerShell Command: Get-NetAdapter	275
15-9	Advanced Properties: Enable QoS	277
15-10	Advanced Properties: Setting VLAN ID	278
15-11	Advanced Properties: Enabling QoS	282
15-12	Advanced Properties: Setting VLAN ID	283
15-13	Advanced Properties: Enabling Virtual Switch RSS	285
15-14	Virtual Switch Manager	286
15-15	Windows PowerShell Command: Get-VMSwitch	287
15-16	Example Hardware Configuration	289
16-1	RSSv2 Event Log Error	297
D-1	ENS Stack Block Diagram	321
D-2	NSX Home Page	323
D-3	Nodes Menu	324
D-4	Add Host Window	324
D-5	NSX Host Added	325
D-6	New Uplink Profile Window	326
D-7	Transport Zones Menu	327
D-8	New Transport Zone	328
D-9	Groups Menu	329
D-10	Add New IP Pool Window	329
D-11	Nodes Menu	330
D-12	Add Transport Node Window	330
D-13	Add Transport Node—ADD N-VDS Window	331
D-14	NSX-T Home Window	332
D-15	Advanced Networking & Security Window	333

D-16	Add New Logical Switch Window	333
D-17	Edit Settings, Virtual Hardware Window	334

List of Tables

Table		Page
2-1	Host Hardware Requirements	5
2-2	Minimum Host Operating System Requirements	6
3-1	41000 Series Adapters Linux Drivers	10
3-2	qede Driver Optional Parameters	16
3-3	Linux Driver Operation Defaults	17
3-4	Linux Driver Debug Verbosity Values	17
3-5	VMware Drivers	35
3-6	VMware NIC Driver Optional Parameters	39
3-7	VMware Driver Parameter Defaults	40
5-1	Adapter Properties	55
5-2	Dual-Port QL41xx2 Default Mode	56
5-3	Dual-Port QL41xx2 NParEP Mode	57
5-4	Quad-Port QL41xx4 Default Mode	58
5-5	Quad-Port QL41xx4 NParEP Mode	59
5-6	QL41xxx PCI Device IDs	60
6-1	iSCSI Out-of-Box and Inbox Boot from SAN Support	80
6-2	iSCSI General Parameters	88
6-3	DHCP Option 17 Parameter Definitions	97
6-4	DHCP Option 43 Sub-option Definitions	98
6-5	DHCP Option 17 Sub-option Definitions	100
6-6	FCoE Out-of-Box and Inbox Boot from SAN Support	116
7-1	OS Support for RoCE v1, RoCE v2, iWARP and iSER	134
7-2	Advanced Properties for RoCE	141
7-3	Marvell FastLinQ RDMA Error Counters	146
7-4	Supported Linux OSs for VF RDMA	169
7-5	DCQCN Algorithm Parameters	183
13-1	Target Parameters	253
16-1	Windows 2019/Azure Stack HCI Virtualization Resources for 41000 Series Adapters	299
16-2	Windows 2019/Azure Stack HCI VMQ and VMMQ Accelerations	300
17-1	Collecting Debug Data Commands	309
A-1	Adapter Port Link and Activity LEDs	310
B-1	Tested Cables and Optical Modules	312
B-2	Switches Tested for Interoperability	317

Preface

This preface lists the supported products, specifies the intended audience, explains the typographic conventions used in this guide, and describes legal notices.

Supported Products

This user's guide describes the following Marvell products:

- QL41112HFCU-DE 10Gb Converged Network Adapter, full-height bracket
- QL41112HLCU-DE 10Gb Converged Network Adapter, low-profile bracket
- QL41132HFRJ-DE 10Gb NIC Adapter, full-height bracket
- QL41132HLRJ-DE 10Gb NIC Adapter, low-profile bracket
- QL41132HQCU-DE 10Gb NIC Adapter
- QL41132HQRJ-DE 10Gb NIC Adapter
- QL41154HQRJ-DE 10Gb Converged Network Adapter
- QL41154HQCU-DE 10Gb Converged Network Adapter
- QL41162HFRJ-DE 10Gb Converged Network Adapter, full-height bracket
- QL41162HLRJ-DE 10Gb Converged Network Adapter, low-profile bracket
- QL41162HMRJ-DE 10Gb Converged Network Adapter
- QL41164HMCU-DE 10Gb Converged Network Adapter
- QL41164HMRJ-DE 10Gb Converged Network Adapter
- QL41164HFRJ-DE 10Gb Converged Network Adapter, full-height bracket
- QL41164HLRJ-DE 10Gb Converged Network Adapter, low-profile bracket
- QL41164HFCU-DE 10Gb Converged Network Adapter, full-height bracket
- QL41232HFCU-DE 10/25Gb NIC Adapter, full-height bracket
- QL41232HLCU-DE 10/25Gb NIC Adapter, low-profile bracket
- QL41232HMKR-DE 10/25Gb NIC Adapter
- QL41232HQCU-DE 10/25Gb NIC Adapter

- QL41262HFCU-DE 10/25Gb Converged Network Adapter, full-height bracket
- QL41262HLCU-DE 10/25Gb Converged Network Adapter, low-profile bracket
- QL41262HMCU-DE 10/25Gb Converged Network
- QL41262HMKR-DE 10/25Gb Converged Network Adapter
- QL41264HMCU-DE 10/25Gb Converged Network Adapter

Intended Audience

This guide is intended for system administrators and other technical staff members responsible for configuring and managing adapters installed on Dell® PowerEdge® servers in Windows®, Linux®, or VMware® environments.

What Is in This Guide

Following this preface, the remainder of this guide is organized into the following chapters and appendices:

- [Chapter 1 Product Overview](#) provides a product functional description, a list of features, and the adapter specifications.
- [Chapter 2 Hardware Installation](#) describes how to install the adapter, including the list of system requirements and a preinstallation checklist.
- [Chapter 3 Driver Installation](#) describes the installation of the adapter drivers on Windows, Linux, and VMware.
- [Chapter 4 Upgrading the Firmware](#) describes how to use the Dell Update Package (DUP) to upgrade the adapter firmware.
- [Chapter 5 Adapter Preboot Configuration](#) describes the preboot adapter configuration tasks using the Human Infrastructure Interface (HII) application.
- [Chapter 6 Boot from SAN Configuration](#) covers boot from SAN configuration for both iSCSI and FCoE.
- [Chapter 7 RoCE Configuration](#) describes how to configure the adapter, the Ethernet switch, and the host to use RDMA over converged Ethernet (RoCE).
- [Chapter 8 iWARP Configuration](#) provides procedures for configuring Internet wide area RDMA protocol (iWARP) on Windows, Linux, and VMware ESXi 6.7/7.0 systems.
- [Chapter 9 iSER Configuration](#) describes how to configure iSCSI Extensions for RDMA (iSER) for Linux RHEL, SLES, Ubuntu, and ESXi 6.7/7.0.

- [Chapter 10 iSCSI Configuration](#) describes iSCSI boot and iSCSI offload for Windows and Linux.
- [Chapter 11 FCoE Configuration](#) covers configuring Linux FCoE offload.
- [Chapter 12 SR-IOV Configuration](#) provides procedures for configuring single root input/output virtualization (SR-IOV) on Windows, Linux, and VMware systems.
- [Chapter 13 NVMe-oF Configuration with RDMA](#) demonstrates how to configure NVMe-oF on a simple network for 41000 Series Adapters.
- [Chapter 14 VXLAN Configuration](#) describes how to configure VXLAN for Linux, VMware, and Windows Server 2016.
- [Chapter 15 Windows Server 2016](#) describes features common to both Windows Server 2016 and Windows Server 2019/Azure Stack HCI.
- [Chapter 16 Windows Server 2019/ Azure Stack HCI](#) describes the Windows Server 2019/Azure Stack HCI features.
- [Chapter 17 Troubleshooting](#) describes a variety of troubleshooting methods and resources.
- [Appendix A Adapter LEDs](#) lists the adapter LEDs and their significance.
- [Appendix B Cables and Optical Modules](#) lists the cables, optical modules, and switches that the 41000 Series Adapters support.
- [Appendix C Dell Z9100 Switch Configuration](#) describes how to configure the Dell Z9100 switch port for 25Gbps.
- [Appendix D VMware ESXi Enhanced Networking Stack Support](#) describes how to use the 41000 Series Adapter as a virtual NIC (vNIC) in a VMware hypervisor environment to support an NSX-Transformer (NSX-T) managed Virtual Distribution Switch (N-VDS).
- [Appendix E Feature Constraints](#) provides information about feature constraints implemented in the current release.
- [Appendix F Revision History](#) describes the changes made in this revision of the guide.

At the end of this guide is a glossary of terms.

Documentation Conventions

This guide uses the following documentation conventions:

- **NOTE** provides additional information.
- **CAUTION** without an alert symbol indicates the presence of a hazard that could cause damage to equipment or loss of data.

- **CAUTION** with an alert symbol indicates the presence of a hazard that could cause minor or moderate injury.
- **WARNING** indicates the presence of a hazard that could cause serious injury or death.
- Text in blue font indicates a hyperlink (jump) to a figure, table, or section in this guide, and links to Web sites are shown in underlined blue. For example:
 - ❑ [Table 9-2](#) lists problems related to the user interface and remote agent.
 - ❑ See “[Installation Checklist](#)” on page 6.
 - ❑ For more information, visit www.marvell.com.
- Text in **bold** font indicates user interface elements such as a menu items, buttons, check boxes, or column headings. For example:
 - ❑ Click the **Start** button, point to **Programs**, point to **Accessories**, and then click **Command Prompt**.
 - ❑ Under **Notification Options**, select the **Warning Alarms** check box.
- Text in `Courier` font indicates a file name, directory path, or command line text. For example:
 - ❑ To return to the root directory from anywhere in the file structure:
Type `cd/ root` and press ENTER.
 - ❑ Issue the following command: `sh ./install.bin`.
- Key names and key strokes are indicated with UPPERCASE:
 - ❑ Press CTRL+P.
 - ❑ Press the UP ARROW key.
- Text in *italics* indicates terms, emphasis, variables, or document titles. For example:
 - ❑ What are *shortcut keys*?
 - ❑ To enter the date type *mm/dd/yyyy* (where *mm* is the month, *dd* is the day, and *yyyy* is the year).
- Topic titles between quotation marks identify related topics either within this manual or in the online help, which is also referred to as *the help system* throughout this document.

- Command line interface (CLI) command syntax conventions include the following:
 - Plain text indicates items that you must type as shown. For example:
 - `qaucli -pr nic -ei`
 - `< >` (angle brackets) indicate a variable whose value you must specify. For example:
 - `<serial_number>`

NOTE

For CLI commands only, variable names are always indicated using angle brackets instead of *italics*.

- `[]` (square brackets) indicate an optional parameter. For example:
 - `[<file_name>]` means specify a file name, or omit it to select the default file name.
- `|` (vertical bar) indicates mutually exclusive options; select one option only. For example:
 - `on|off`
 - `1|2|3|4`
- `...` (ellipsis) indicates that the preceding item may be repeated. For example:
 - `x...` means *one* or more instances of `x`.
 - `[x...]` means *zero* or more instances of `x`.
- `⋮` (vertical ellipsis) within command example output indicate where portions of repetitious output data have been intentionally omitted.
- `()` (parentheses) and `{ }` (braces) are used to avoid logical ambiguity. For example:
 - `a|b c` is ambiguous
 - `{(a|b) c}` means *a* or *b*, followed by *c*
 - `{a|(b c)}` means either *a*, or *b c*

Legal Notices

Legal notices covered in this section include laser safety (FDA notice), agency certification, and product safety compliance.

Laser Safety—FDA Notice

This product complies with DHHS Rules 21CFR Chapter I, Subchapter J. This product has been designed and manufactured according to IEC60825-1 on the safety label of laser product.

CLASS I LASER

Class 1 Laser Product	Caution —Class 1 laser radiation when open Do not view directly with optical instruments
Appareil laser de classe 1	Attention —Radiation laser de classe 1 Ne pas regarder directement avec des instruments optiques
Produkt der Laser Klasse 1	Vorsicht —Laserstrahlung der Klasse 1 bei geöffneter Abdeckung Direktes Ansehen mit optischen Instrumenten vermeiden
Luokan 1 Laserlaite	Varoitus —Luokan 1 lasersäteilyä, kun laite on auki Älä katso suoraan laitteeseen käyttämällä optisia instrumenttejä

Agency Certification

The following sections summarize the EMC and EMI test specifications performed on the 41000 Series Adapters to comply with emission, immunity, and product safety standards.

EMI and EMC Requirements

FCC Part 15 compliance: Class A

FCC compliance information statement: This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

ICES-003 Compliance: Class A

This Class A digital apparatus complies with Canadian ICES-003. Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

CE Mark 2014/30/EU, 2014/35/EU EMC Directive Compliance:

EN55032:2012/ CISPR 32:2015 Class A

EN55024: 2010

EN61000-3-2: Harmonic Current Emission

EN61000-3-3: Voltage Fluctuation and Flicker

Immunity Standards

EN61000-4-2: ESD
EN61000-4-3: RF Electro Magnetic Field
EN61000-4-4: Fast Transient/Burst
EN61000-4-5: Fast Surge Common/ Differential
EN61000-4-6: RF Conducted Susceptibility
EN61000-4-8: Power Frequency Magnetic Field
EN61000-4-11: Voltage Dips and Interrupt

VCCI: 2015-04; Class A

AS/NZS; CISPR 32: 2015 Class A

CNS 13438: 2006 Class A

KCC: Class A

Korea RRA Class A Certified



Product Name/Model: Converged Network Adapters and Intelligent Ethernet Adapters
Certification holder: QLogic Corporation
Manufactured date: Refer to date code listed on product
Manufacturer/Country of origin: QLogic Corporation/USA

A class equipment
(Business purpose
info/telecommunications
equipment)

As this equipment has undergone EMC registration for business purpose, the seller and/or the buyer is asked to beware of this point and in case a wrongful sale or purchase has been made, it is asked that a change to household use be made.

Korean Language Format—Class A

A급 기기 (업무용 정보통신기기)

이 기기는 업무용으로 전자파적합등록을 한 기기이오니 판매자 또는 사용자는 이 점을 주의하시기 바라며, 만약 잘못판매 또는 구입하였을 때에는 가정용으로 교환하시기 바랍니다.

VCCI: Class A

This is a Class A product based on the standard of the Voluntary Control Council for Interference (VCCI). If this equipment is used in a domestic environment, radio interference may occur, in which case the user may be required to take corrective actions.

<p>この装置は、クラスA情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。</p> <p>VCCI-A</p>

Product Safety Compliance

UL, cUL product safety:

UL 60950-1 (2nd Edition) A1 + A2 2014-10-14

CSA C22.2 No.60950-1-07 (2nd Edition) A1 +A2 2014-10

Use only with listed ITE or equivalent.

Complies with 21 CFR 1040.10 and 1040.11, 2014/30/EU, 2014/35/EU.

2006/95/EC low voltage directive:

TUV EN60950-1:2006+A11+A1+A12+A2 2nd Edition

TUV IEC 60950-1: 2005 2nd Edition Am1: 2009 + Am2: 2013 CB

TUV IEC 62368 2nd and 3rd Edition CB

CB Certified to IEC 60950-1 2nd Edition

1 Product Overview

This chapter provides the following information for the 41000 Series Adapters:

- [Functional Description](#)
- [Features](#)
- [“Adapter Specifications” on page 3](#)

Functional Description

The Marvell FastLinQ 41000 Series Adapters include 10 and 25Gb Converged Network Adapters and Intelligent Ethernet Adapters that are designed to perform accelerated data networking for server systems. The 41000 Series Adapter includes a 10/25Gb Ethernet MAC with full-duplex capability.

Using the operating system’s teaming feature, you can split your network into virtual LANs (vLANs), as well as group multiple network adapters together into teams to provide network load balancing and fault tolerance. For more information about teaming, see your operating system documentation.

Features

The 41000 Series Adapters provide the following features. Some features may not be available on all adapters:

- NIC partitioning (NPar)/NPar extended partitioning (NParEP)
- Single-chip solution:
 - ❑ 10/25Gb MAC
 - ❑ SerDes interface for direct attach copper (DAC) transceiver connection
 - ❑ PCI Express® (PCIe®) 3.0 x8
 - ❑ Zero copy capable hardware
- Performance features:
 - ❑ TCP, IP, UDP checksum offloads
 - ❑ TCP segmentation offload (TSO)
 - ❑ Large segment offload (LSO)

- Generic segment offload (GSO)
- Large receive offload (LRO)
- Receive segment coalescing (RSC)
- Microsoft® dynamic virtual machine queue (VMQ), and Linux Multiqueue
- Adaptive interrupts:
 - Transmit/receive side scaling (TSS/RSS)
 - Stateless offloads for Network Virtualization using Generic Routing Encapsulation (NVGRE) and virtual LAN (VXLAN) L2/L3 GRE tunneled traffic¹
- Manageability:
 - System management bus (SMB) controller
 - Advanced Configuration and Power Interface* (ACPI) 1.1a compliant (multiple power modes)
 - Network controller-sideband interface (NC-SI) support
- Advanced network features:
 - Jumbo frames (up to 9,600 bytes). The OS and the link partner must support jumbo frames.
 - Virtual LANs (VLANs)
 - Flow control (IEEE Std 802.3x)
- Logical link control (IEEE Std 802.2)
- High-speed on-chip reduced instruction set computer (RISC) processor
- Integrated 96KB frame buffer memory (not applicable to all models)
- 1,024 classification filters (not applicable to all models)
- Support for multicast addresses through 128-bit hashing hardware function
- Support for VMDirectPath I/O over PCI physical functions

FastLinQ 41000 Series Adapters support VMDirectPath I/O in Linux and ESX environments. VMDirectPath I/O is not supported in Windows environments.

FastLinQ 41000 Series Adapters can be assigned to virtual machines for PCI pass-through operation. However, due to function level dependencies, all PCIe functions associated with an adapter (in default, NPar, or NParEP modes) must be assigned to the same virtual machine. Sharing PCIe physical functions across the hypervisor and any virtual machines; or sharing them across more than one virtual machine, is not supported.

¹ This feature requires OS or Hypervisor support to use the offloads.

- Serial flash NVRAM memory
- *PCI Power Management Interface* (v1.1)
- 64-bit base address register (BAR) support
- EM64T processor support
- SR-IOV Virtual Functions (VF)²
- iSCSI (software (SW) and hardware (HW)) and FCoE boot support³

Adapter Specifications

The 41000 Series Adapter specifications include the adapter's physical characteristics and standards-compliance references.

Physical Characteristics

The 41000 Series Adapters are standard PCIe cards and ship with either a full-height or a low-profile bracket for use in a standard PCIe slot.

Standards Specifications

Supported standards specifications include:

- *PCI Express Base Specification*, rev. 3.1
- *PCI Express Card Electromechanical Specification*, rev. 3.0
- *PCI Bus Power Management Interface Specification*, rev. 1.2
- IEEE Specifications:
 - ❑ *802.1ad* (QinQ)
 - ❑ *802.1AX* (Link Aggregation)
 - ❑ *802.1p* (Priority Encoding)
 - ❑ *802.1q* (VLAN)
 - ❑ *802.3-2015 IEEE Standard for Ethernet* (flow control)
 - ❑ *802.3-2015 Clause 78 Energy Efficient Ethernet (EEE)*

² Hardware support limit of SR-IOV VFs varies. The limit may be lower in some OS environments; refer to the appropriate section for your OS

³ On Dell 41000 Series Adapters, FCoE-Offload and FCoE-Offload remote boot require enabling NPar/NParEP mode and also enabling FCoE-Offload on the second partition or physical function (PF) of the desired boot or usage port. On Dell 41000 Series Adapters, iSCSI-Offload (HW) and iSCSI-Offload (HW) remote boot require enabling NPar/NParEP mode and also enabling iSCSI-Offload on the third partition or PF of the desired boot or usage port. iSCSI (SW) and iSCSI (SW) remote boot do not require enabling NPar/NParEP mode and use the always enabled first Ethernet partition or PF of the desired boot or usage port. PXE remote boot does not require enabling NPar/NParEP mode and uses the always enabled first Ethernet partition or PF of the desired boot port. Only one remote boot mode (PXE or FCoE or iSCSI HW or iSCSI SW) or offload type (FCoE or iSCSI HW) can be selected and utilized per physical port.

- 1588-2002 PTPv1 (Precision Time Protocol)
- 1588-2008 PTPv2
- IPv4 (RFQ 791)
- IPv6 (RFQ 2460)

2 Hardware Installation

This chapter provides the following hardware installation information:

- [System Requirements](#)
- [“Safety Precautions” on page 6](#)
- [“Preinstallation Checklist” on page 7](#)
- [“Installing the Adapter” on page 7](#)

System Requirements

Before you install a Marvell 41000 Series Adapter, verify that your system meets the hardware and operating system requirements shown in the following sections.

Hardware Requirements

The hardware requirements for the 41000 Series Adapter are listed in [Table 2-1](#).

Table 2-1. Host Hardware Requirements

Hardware	Requirement
Architecture	IA-32 or EMT64 that meets operating system requirements
PCIe	PCIe Gen 2 x8 (2x10G NIC) PCIe Gen 3 x8 (2x25G NIC) Full dual-port 25Gb bandwidth is supported on PCIe Gen 3 x8 or faster slots.
Memory	8GB RAM (minimum)
Cables and Optical Modules	The 41000 Series Adapters have been tested for interoperability with a variety of 1G, 10G, and 25G cables and optical modules. See “Tested Cables and Optical Modules” on page 312 .

Software Requirements

The minimum software requirements at the time of publication are listed in [Table 2-2](#).

Table 2-2. Minimum Host Operating System Requirements

Operating System	Requirement
Windows Server	2016, 2019, Azure Stack HCI
Linux	RHEL® 7.8, 7.9, 8.2, 8.3 SLES® 15 SP1, SP2 Ubuntu 20.04
VMware	ESXi 6.7 U3, ESXi 7.0 U1
XenServer	Citrix Hypervisor 7.2 CU2 LTSR, 8.2 LTSR

NOTE

[Table 2-2](#) denotes the minimum host OS requirements. For a complete list of supported operating systems, refer to the appropriate *Read Me* and *Release Notes*.

Safety Precautions

WARNING

The adapter is being installed in a system that operates with voltages that can be lethal. Before you open the case of your system, observe the following precautions to protect yourself and to prevent damage to the system components.

- Remove any metallic objects or jewelry from your hands and wrists.
 - Make sure to use only insulated or nonconducting tools.
 - Verify that the system is powered OFF and is unplugged before you touch internal components.
 - Install or remove adapters in a static-free environment. The use of a properly grounded wrist strap or other personal antistatic devices and an antistatic mat is strongly recommended.
-

Preinstallation Checklist

Before installing the adapter, complete the following:

1. Verify that the system meets the hardware and software requirements listed under [“System Requirements” on page 5](#).
2. Verify that the system is using the latest BIOS.

NOTE

If you acquired the adapter software from the Marvell Web site, verify the path to the adapter driver files.

3. If the system is active, shut it down.
4. When system shutdown is complete, turn off the power and unplug the power cord.
5. Remove the adapter from its shipping package and place it on an anti-static surface.
6. Check the adapter for visible signs of damage, particularly on the edge connector. Never attempt to install a damaged adapter.

Installing the Adapter

The following instructions apply to installing the Marvell 41000 Series Adapters in most systems. For details about performing these tasks, refer to the manuals that were supplied with the system.

To install the adapter:

1. Review [“Safety Precautions” on page 6](#) and [“Preinstallation Checklist” on page 7](#). Before you install the adapter, ensure that the system power is OFF, the power cord is unplugged from the power outlet, and that you are following proper electrical grounding procedures.
2. Open the system case, and select the slot that matches the adapter size, which can be PCIe Gen 2 x8 or PCIe Gen 3 x8. A lesser-width adapter can be seated into a greater-width slot (x8 in an x16), but a greater-width adapter cannot be seated into a lesser-width slot (x8 in an x4). If you do not know how to identify a PCIe slot, refer to your system documentation.
3. Remove the blank cover-plate from the slot that you selected.
4. Align the adapter connector edge with the PCIe connector slot in the system.

5. Applying even pressure at both corners of the card, push the adapter card into the slot until it is firmly seated. When the adapter is properly seated, the adapter port connectors are aligned with the slot opening, and the adapter faceplate is flush against the system chassis.

CAUTION

Do not use excessive force when seating the card, because this may damage the system or the adapter. If you have difficulty seating the adapter, remove it, realign it, and try again.

6. Secure the adapter with the adapter clip or screw.
7. Close the system case and disconnect any personal anti-static devices.

3 Driver Installation

This chapter provides the following information about driver installation:

- [Installing Linux Driver Software](#)
- [“Installing Windows Driver Software” on page 21](#)
- [“Installing VMware Driver Software” on page 34](#)
- [“Installing Citrix Hypervisor Driver Software” on page 44](#)

Installing Linux Driver Software

This section describes how to install Linux drivers with or without remote direct memory access (RDMA). It also describes the Linux driver optional parameters, default values, messages, statistics, and public key for Secure Boot.

- [Installing the Linux Drivers Without RDMA](#)
- [Installing the Linux Drivers with RDMA](#)
- [Linux Driver Optional Parameters](#)
- [Linux Driver Operation Defaults](#)
- [Linux Driver Messages](#)
- [Statistics](#)
- [Importing a Public Key for Secure Boot](#)

The 41000 Series Adapter Linux drivers and supporting documentation are available on the Dell Support page:

dell.support.com

Table 3-1 describes the 41000 Series Adapter Linux drivers.

Table 3-1. 41000 Series Adapters Linux Drivers

Linux Driver	Description
qed	The qed core driver module directly controls the firmware, handles interrupts, and provides the low-level API for the protocol specific driver set. The qed interfaces with the qede, qedr, qedi, and qedf drivers. The Linux core module manages all PCI device resources (registers, host interface queues, and so on). The qed core module requires Linux kernel version 2.6.32 or later. Testing was concentrated on the x86_64 architecture.
qede	Linux Ethernet driver for the 41000 Series Adapter. This driver directly controls the hardware and is responsible for sending and receiving Ethernet packets on behalf of the Linux host networking stack. This driver also receives and processes device interrupts on behalf of itself (for L2 networking). The qede driver requires Linux kernel version 2.6.32 or later. Testing was concentrated on the x86_64 architecture.
qedr	Linux RDMA (RoCE and iWARP) driver that works in the OpenFabrics Enterprise Distribution (OFED™) environment, which is available in all major Linux distributions. This driver works in conjunction with the qed core module and the qede Ethernet driver. RDMA user space applications also require that the libqedr (part of the rdma-core, the RDMA user stack) user library is installed on the server.
qedi	Linux iSCSI-Offload driver for the 41000 Series Adapters. This driver works with the Open iSCSI library.
qedf	Linux FCoE-Offload driver for the 41000 Series Adapters. This driver works with Open FCoE library.

Install the Linux drivers using either a source Red Hat® Package Manager (RPM) package or a kmod RPM package. The RHEL RPM packages are as follows:

- qlgc-fastlinq-<version>.<OS>.src.rpm
- qlgc-fastlinq-kmp-default-<version>.<arch>.rpm

The SLES source and kmp RPM packages are as follows:

- qlgc-fastlinq-<version>.<OS>.src.rpm
- qlgc-fastlinq-kmp-default-<version>.<OS>.<arch>.rpm

The following kernel module (kmod) RPM installs Linux drivers on SLES hosts running the Xen Hypervisor:

- qlgc-fastlinq-kmp-xen-<version>.<OS>.<arch>.rpm

The following source RPM installs the RDMA library code on RHEL and SLES hosts:

- qlgc-libqedr-<version>.<OS>.<arch>.src.rpm

The following source code TAR BZip2 (BZ2) compressed file installs Linux drivers on RHEL and SLES hosts:

- `fastlinq-<version>.tar.bz2`

NOTE

For network installations through NFS, FTP, or HTTP (using a network boot disk), you may require a driver disk that contains the qede driver. Compile the Linux boot drivers by modifying the makefile and the make environment.

Installing the Linux Drivers Without RDMA

To install the Linux drivers without RDMA:

1. Download the 41000 Series Adapter Linux drivers from Dell:
dell.support.com
2. Remove the existing Linux drivers, as described in “[Removing the Linux Drivers](#)” on page 11.
3. Install the new Linux drivers using one of the following methods:
 - [Installing Linux Drivers Using the src RPM Package](#)
 - [Installing Linux Drivers Using the kmp/kmod RPM Package](#)
 - [Installing Linux Drivers Using the TAR File](#)

Removing the Linux Drivers

There are two procedures for removing Linux drivers: one for a non-RDMA environment and another for an RDMA environment. Choose the procedure that matches your environment.

To remove Linux drivers in a non-RDMA environment, unload and remove the drivers:

Follow the procedure that relates to the original installation method and the OS.

- If the Linux drivers were installed using an RPM package, issue the following commands:

```
rmmod qede
rmmod qed
depmod -a
rpm -e qlgc-fastlinq-kmp-default-<version>.<arch>
```

- If the Linux drivers were installed using a TAR file, issue the following commands:

```
rmmod qede
```

```
rmmmod qed  
depmod -a
```

- ❑ For RHEL:

```
cd /lib/modules/<version>/extra/qlgc-fastlinq  
rm -rf qed.ko qede.ko qedr.ko
```

- ❑ For SLES:

```
cd /lib/modules/<version>/updates/qlgc-fastlinq  
rm -rf qed.ko qede.ko qedr.ko
```

To remove Linux drivers in a non-RDMA environment:

1. To get the path to the currently installed drivers, issue the following command:

```
modinfo <driver name>
```

2. Unload and remove the Linux drivers.

- ❑ If the Linux drivers were installed using an RPM package, issue the following commands:

```
modprobe -r qede  
depmod -a  
rpm -e qlgc-fastlinq-kmp-default-<version>.<arch>
```

- ❑ If the Linux drivers were installed using a TAR file, issue the following commands:

```
modprobe -r qede  
depmod -a
```

NOTE

If the `qedr` is present, issue the `modprobe -r qedr` command instead.

3. Delete the `qed.ko`, `qede.ko`, and `qedr.ko` files from the directory in which they reside. For example, in SLES, issue the following commands:

```
cd /lib/modules/<version>/updates/qlgc-fastlinq  
rm -rf qed.ko  
rm -rf qede.ko  
rm -rf qedr.ko  
depmod -a
```

To remove Linux drivers in an RDMA environment:

1. To get the path to the installed drivers, issue the following command:

```
modinfo <driver name>
```

2. Unload and remove the Linux drivers.

```
modprobe -r qedr  
modprobe -r qede  
modprobe -r qed  
depmod -a
```

3. Remove the driver module files:

- ❑ If the drivers were installed using an RPM package, issue the following command:

```
rpm -e qlgc-fastlinq-kmp-default-<version>.<arch>
```

- ❑ If the drivers were installed using a TAR file, issue the following commands for your operating system:

For RHEL:

```
cd /lib/modules/<version>/extra/qlgc-fastlinq  
rm -rf qed.ko qede.ko qedr.ko
```

For SLES:

```
cd /lib/modules/<version>/updates/qlgc-fastlinq  
rm -rf qed.ko qede.ko qedr.ko
```

Installing Linux Drivers Using the src RPM Package

To install Linux drivers using the src RPM package:

1. Issue the following at a command prompt:

```
rpm -ivh RPMS/<arch>/qlgc-fastlinq-<version>.src.rpm
```

2. Change the directory to the RPM path and build the binary RPM for the kernel.

NOTE

For RHEL 8, install the `kernel-rpm-nacros` and `kernel-abi-whitelists` packages before building the binary RPM package.

For SLES15 SP2, install `kernel-source` and `kernel-syms` (using either the software manager or Zypper tool) before building the binary RPM package.

For RHEL:

```
cd /root/rpmbuild  
rpmbuild -bb SPECS/fastlinq-<version>.spec
```

For SLES:

```
cd /usr/src/packages  
rpmbuild -bb SPECS/fastlinq-<version>.spec
```

3. Install the newly compiled RPM:

```
rpm -ivh RPMS/<arch>/qlgc-fastlinq-<version>.<arch>.rpm
```

NOTE

The `--force` option may be needed on some Linux distributions if conflicts are reported.

The drivers will be installed in the following paths.

For SLES:

```
/lib/modules/<version>/updates/qlgc-fastlinq
```

For RHEL:

```
/lib/modules/<version>/extra/qlgc-fastlinq
```

4. Turn on all ethX interfaces as follows:

```
ifconfig <ethX> up
```
5. For SLES, use YaST to configure the Ethernet interfaces to automatically start at boot by setting a static IP address or enabling DHCP on the interface.

Installing Linux Drivers Using the kmp/kmod RPM Package

To install kmod RPM package:

1. Issue the following command at a command prompt:

```
rpm -ivh qlgc-fastlinq-<version>.<arch>.rpm
```

2. Reload the driver:

```
modprobe -r qede  
modprobe qede
```

Installing Linux Drivers Using the TAR File

To install Linux drivers using the TAR file:

1. Create a directory and extract the TAR files to the directory:
2. Change to the recently created directory, and then install the drivers:

```
tar xjvf fastlinq-<version>.tar.bz2
```

```
cd fastlinq-<version>  
make clean; make install
```

The qed and qede drivers will be installed in the following paths.

For SLES:

```
/lib/modules/<version>/updates/qlgc-fastlinq
```

For RHEL:

```
/lib/modules/<version>/extra/qlgc-fastlinq
```

3. Test the drivers by loading them (unload the existing drivers first, if necessary):

```
rmmod qede  
rmmod qed  
modprobe qed  
modprobe qede
```

Installing the Linux Drivers with RDMA

For information on iWARP, see [Chapter 8 iWARP Configuration](#).

To install Linux drivers in an inbox OFED environment:

1. Download the 41000 Series Adapter Linux drivers from the Dell:
dell.support.com
2. Configure RoCE on the adapter, as described in “[Configuring RoCE on the Adapter for Linux](#)” on page 158.
3. Remove existing Linux drivers, as described in “[Removing the Linux Drivers](#)” on page 11.
4. Install the new Linux drivers using one of the following methods:
 - [Installing Linux Drivers Using the kmp/kmod RPM Package](#)
 - [Installing Linux Drivers Using the TAR File](#)

5. Install libqedr libraries to work with RDMA user space applications. The libqedr RPM is available only for inbox OFED. You must select which RDMA (RoCE, RoCEv2, or iWARP) is used in UEFI until concurrent RoCE+iWARP capability is supported in the firmware). None is enabled by default. Issue the following command:

```
rpm -ivh qlgc-libqedr-<version>.<arch>.rpm
```

6. To build and install the libqedr user space library in Linux Red Hat 7.3 and earlier, issue the following command:

```
'make libqedr_install'
```

7. Test the drivers in Linux Red Hat 7.3 and earlier by loading them as follows:

```
modprobe qedr  
make install_libqedr
```

Linux Driver Optional Parameters

Table 3-2 describes the optional parameters for the qede driver.

Table 3-2. qede Driver Optional Parameters

Parameter	Description
debug	Controls driver verbosity level similar to <code>ethtool -s <dev> msglvl.</code>
int_mode	Controls interrupt mode other than MSI-X.
gro_disable	The device supports generic receive offload (GRO) in the hardware. In older Linux kernels, GRO was a software-only feature. As such, disabling GRO would not reach the driver, nor could it be dynamically disabled in the device through the <code>ethtool</code> callback method. Therefore, this module parameter was provided. Newer kernels offer hardware GRO as an <code>ethtool</code> feature, which can perform the same functionality dynamically.
err_flags_override	A bitmap for disabling or forcing the actions taken in case of a hardware error: <ul style="list-style-type: none">■ bit #31 – An enable bit for this bitmask■ bit #0 – Prevent hardware attentions from being reasserted■ bit #1 – Collect debug data■ bit #2 – Trigger a recovery process■ bit #3 – Call WARN to get a call trace of the flow that led to the error

Linux Driver Operation Defaults

Table 3-3 lists the qed and qede Linux driver operation defaults.

Table 3-3. Linux Driver Operation Defaults

Operation	qed Driver Default	qede Driver Default
Speed	Auto-negotiation with speed advertised	Auto-negotiation with speed advertised
MSI/MSI-X	Enabled	Enabled
Flow Control	—	Auto-negotiation with RX and TX advertised
MTU	—	1500 (range is 46–9600)
Rx Ring Size	—	1000
Tx Ring Size	—	4078 (range is 128–8191)
Coalesce Rx Microseconds	—	24 (range is 0–255)
Coalesce Tx Microseconds	—	48
TSO	—	Enabled

Linux Driver Messages

To set the Linux driver message detail level, issue one of the following commands:

- `ethtool -s <interface> msglvl <value>`
- `modprobe qede debug=<value>`

Where `<value>` represents bits 0–15, which are standard Linux networking values, and bits 16 and greater are driver-specific.

Table 3-4 defines the `<value>` options for the various outputs. This information is also available in the `verbosity_log_level.txt` file in the package documentation folder.

Table 3-4. Linux Driver Debug Verbosity Values

Name	Value	Description
Debug verbose in netdev		
NETIF_MSG_DRV	0x0001	Information about the network debug feature infrastructure in the kernel.
NETIF_MSG_PROBE	0x0002	Information about the driver module probe

Table 3-4. Linux Driver Debug Verbosity Values (Continued)

Name	Value	Description
NETIF_MSG_LINK	0x0004	Information about the network interface link
NETIF_MSG_TIMER	0x0008	Information about the timer infrastructure
NETIF_MSG_IFDOWN	0x0010	Information about link down flow
NETIF_MSG_IFUP	0x0020	Information about link up flow
NETIF_MSG_RX_ERR	0x0040	Information about receive errors
NETIF_MSG_TX_ERR	0x0080	Information about transmit errors
NETIF_MSG_TX_UEUED	0x0100	Information about transmission queues in fast path
NETIF_MSG_INTR	0x0200	Information about assertion, attentions, and interrupt handling
NETIF_MSG_TX_DONE	0x0400	Information about transmission completion
NETIF_MSG_RX_STATUS	0x0800	Information about receive queues in fast path
NETIF_MSG_PKTDATA	0x1000	Information about packet/buffer data
NETIF_MSG_HW	0x2000	Information about hardware data (for example, register access)
NETIF_MSG_WOL	0x4000	Information about wake on LAN (WoL)
Debug verbose in qed/qede		
QED_MSG_SPQ	0x10000	Information about slow path queues' settings
QED_MSG_STATS	0x20000	Information about driver statistics
QED_MSG_DCB	0x40000	Information about data center bridging exchange (DCBX) configuration and settings
QED_MSG_IOV	0x80000	Information about SR-IOV. If virtual machines (VMs) are being used, Marvell recommends setting this value on both VMs and the hypervisor.
QED_MSG_SP	0x100000	Information about slow path (all ramrods and mail-boxes)
QED_MSG_STORAGE	0x200000	Information about storage
QED_MSG_OOO	0x200000	Information about out-of-order packets/buffers
QED_MSG_CXT	0x800000	Information about context manager (for example, context identifiers (CIDs))
QED_MSG_LL2	0x1000000	Information about light L2

Table 3-4. Linux Driver Debug Verbosity Values (Continued)

Name	Value	Description
QED_MSG_ILT	0x2000000	Information about the internal lookup table (ILT)
QED_MSG_RDMA	0x4000000	Information about RDMA (RoCE and iWARP)
QED_MSG_DEBUG	0x8000000	Information about debug feature infrastructure
Debug verbose in qedr		
QEDR_MSG_INIT	0x10000	Information about initialization
QEDR_MSG_FAIL	0x10000	Information about failure
QEDR_MSG_CQ	0x20000	Information about the completion queue
QEDR_MSG_RQ	0x40000	Information about the receive queue
QEDR_MSG_SQ	0x80000	Information about the send queue
QEDR_MSG_QP	0xC0000	Information about the queue pairs
QEDR_MSG_MR	0x100000	Information about the memory region
QEDR_MSG_GSI	0x200000	Information about the general service interface
QEDR_MSG_MISC	0x400000	Information about miscellaneous topics
QEDR_MSG_SRQ	0x800000	Information about the shared receive queue
QEDR_MSG_IWARP	0x1000000	Information about iWARP

Statistics

To view detailed statistics and configuration information, use the `ethtool` utility. See the `ethtool` man page for more information.

To collect statistics, issue the `ethtool -S` command.

Linux RoCE MTU

For optimal performance, the per-Ethernet physical function RoCE MTU size should be 4,096. To achieve this size, set the per L2 Ethernet physical function MTU size larger than 4,096. Additionally, set the network and target ports to an equivalent L2 Ethernet MTU size (to prevent these packets from fragmenting or dropping).

The RoCE MTU size is automatically set to the largest supported size; that is, smaller than the current L2 Ethernet MTU size of that physical function. On Linux, this RoCE MTU value is automatically changed (when the L2 Ethernet MTU value is changed) on the same Ethernet physical function.

Importing a Public Key for Secure Boot

Linux drivers require that you import and enroll the QLogic public key to load the drivers in a Secure Boot environment. Before you begin, ensure that your server supports Secure Boot.

This section provides two methods for importing and enrolling the public key.

To import and enroll the QLogic public key:

1. Download the public key from the following Web page:

<http://ldriver.qlogic.com/Module-public-key/>

2. To install the public key, issue the following command:

```
# mokutil --root-pw --import cert.der
```

Where the `--root-pw` option enables direct use of the root user.

3. Reboot the system.
4. Review the list of certificates that are prepared to be enrolled:

```
# mokutil --list-new
```

5. Reboot the system again.
6. When the shim launches MokManager, enter the root password to confirm the certificate importation to the Machine Owner Key (MOK) list.
7. To determine if the newly imported key was enrolled:

```
# mokutil --list-enrolled
```

To launch MOK manually and enroll the QLogic public key:

1. Issue the following command:

```
# reboot
```

2. In the **GRUB 2** menu, press the C key.

3. Issue the following commands:

```
chainloader $efibootdir/MokManager.efi  
- boot
```

4. Select **Enroll key from disk**.
5. Navigate to the `cert.der` file and then press ENTER.
6. Follow the instructions to enroll the key. Generally this includes pressing the 0 (zero) key and then pressing the Y key to confirm.

NOTE

The firmware menu may provide more methods to add a new key to the Signature Database.

For additional information about Secure Boot, refer to the following Web page:

https://www.suse.com/documentation/sled-12/book_sle_admin/data/sec_uefi_secboot.html

Installing Windows Driver Software

For information on iWARP, see [Chapter 8 iWARP Configuration](#).

- [Installing the Windows Drivers](#)
- [Removing the Windows Drivers](#)
- [Managing Adapter Properties](#)
- [Setting Power Management Options](#)
- [Link Configuration in Windows](#)

Installing the Windows Drivers

Install Windows driver software using the Dell Update Package (DUP):

- [Running the DUP in the GUI](#)
- [DUP Installation Options](#)
- [DUP Installation Examples](#)

Running the DUP in the GUI

To run the DUP in the GUI:

1. Double-click the icon representing the Dell Update Package file.

NOTE

The actual file name of the Dell Update Package varies.

2. In the Dell Update Package window (Figure 3-1), click **Install**.

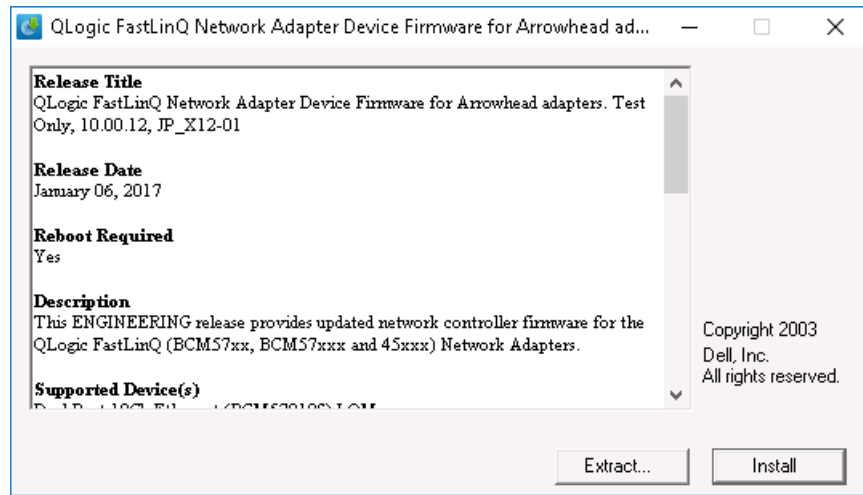


Figure 3-1. Dell Update Package Window

3. In the QLogic Super Installer—InstallShield® Wizard's Welcome window (Figure 3-2), click **Next**.

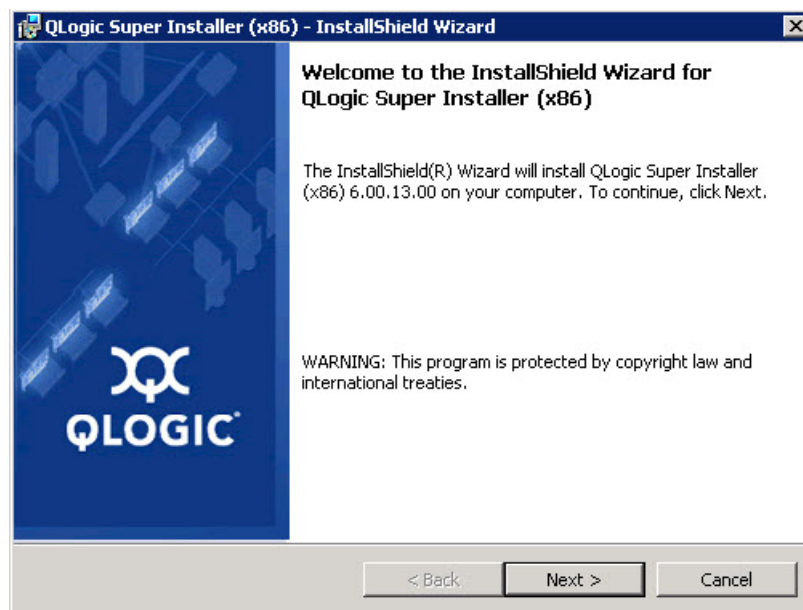


Figure 3-2. QLogic InstallShield Wizard: Welcome Window

4. Complete the following in the wizard's License Agreement window (Figure 3-3):
 - a. Read the End User Software License Agreement.
 - b. To continue, select **I accept the terms in the license agreement**.
 - c. Click **Next**.

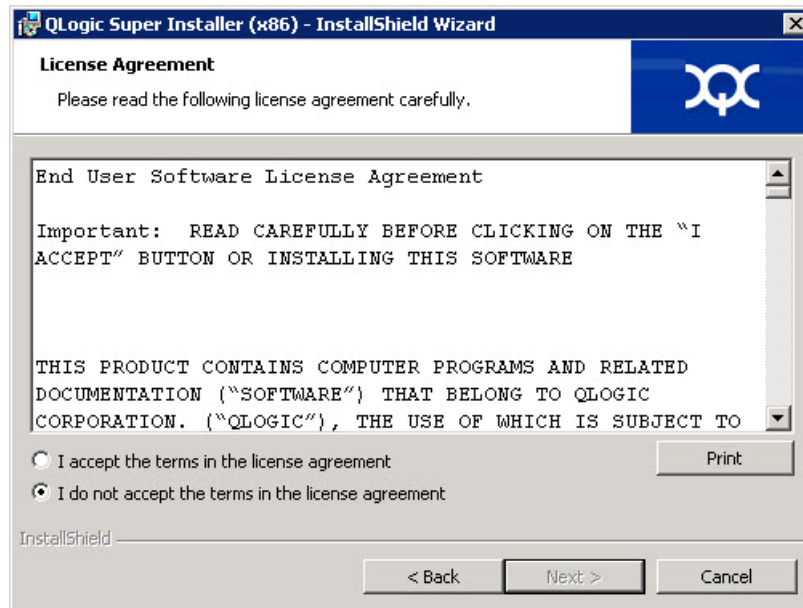


Figure 3-3. QLogic InstallShield Wizard: License Agreement Window

5. Complete the wizard's Setup Type window (Figure 3-4) as follows:
 - a. Select one of the following setup types:
 - Click **Complete** to install all program features.
 - Click **Custom** to manually select the features to be installed.
 - b. To continue, click **Next**.

If you clicked **Complete**, proceed directly to [Step 6b](#).

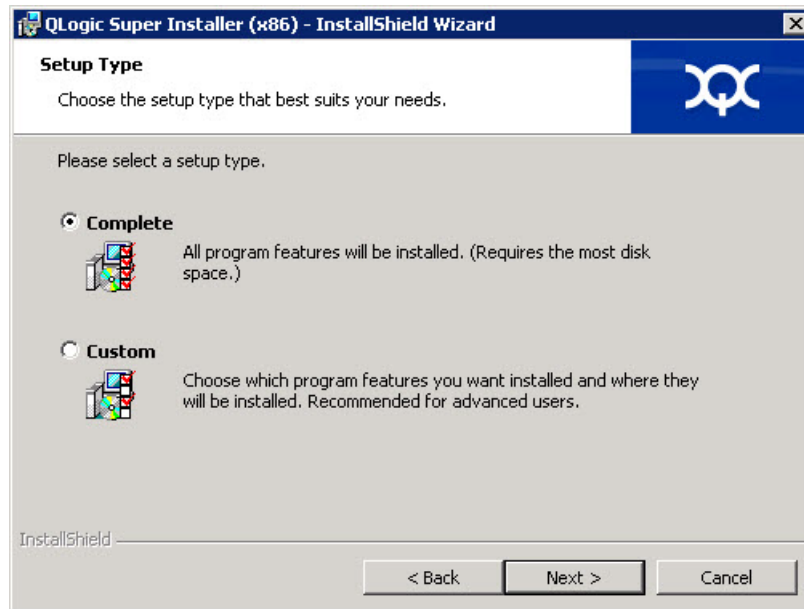


Figure 3-4. InstallShield Wizard: Setup Type Window

6. If you selected **Custom** in [Step 5](#), complete the Custom Setup window ([Figure 3-5](#)) as follows:
 - a. Select the features to install. By default, all features are selected. To change a feature's install setting, click the icon next to it, and then select one of the following options:
 - **This feature will be installed on the local hard drive**—Marks the feature for installation without affecting any of its subfeatures.
 - **This feature, and all subfeatures, will be installed on the local hard drive**—Marks the feature and all of its subfeatures for installation.
 - **This feature will not be available**—Prevents the feature from being installed.
 - b. Click **Next** to continue.

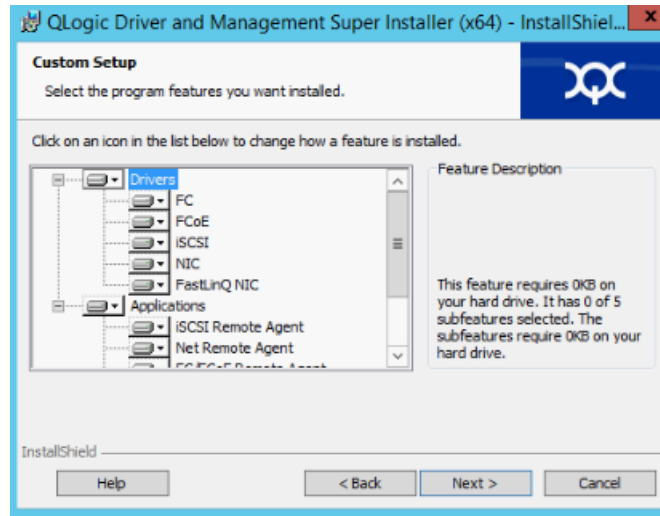


Figure 3-5. InstallShield Wizard: Custom Setup Window

7. In the InstallShield Wizard's Ready To Install window (Figure 3-6), click **Install**. The InstallShield Wizard installs the QLogic Adapter drivers and Management Software Installer.

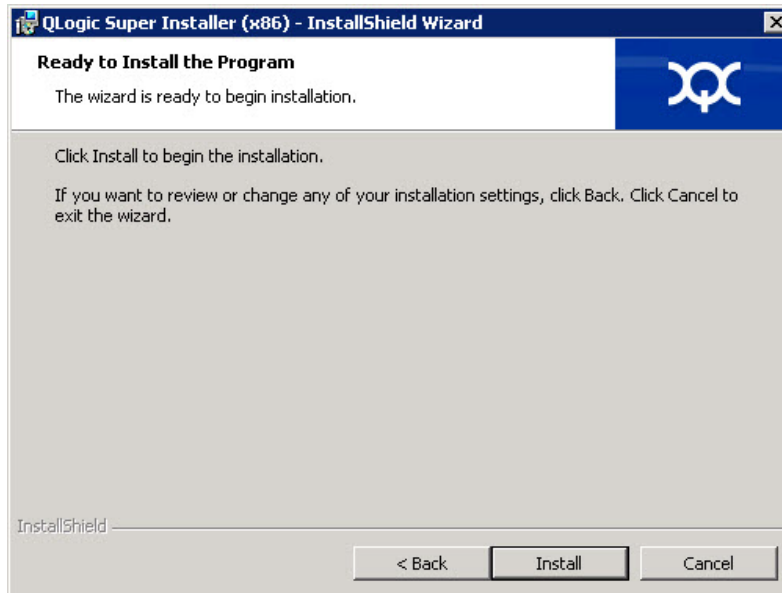


Figure 3-6. InstallShield Wizard: Ready to Install the Program Window

8. When the installation is complete, the InstallShield Wizard Completed window appears (Figure 3-7). Click **Finish** to dismiss the installer.

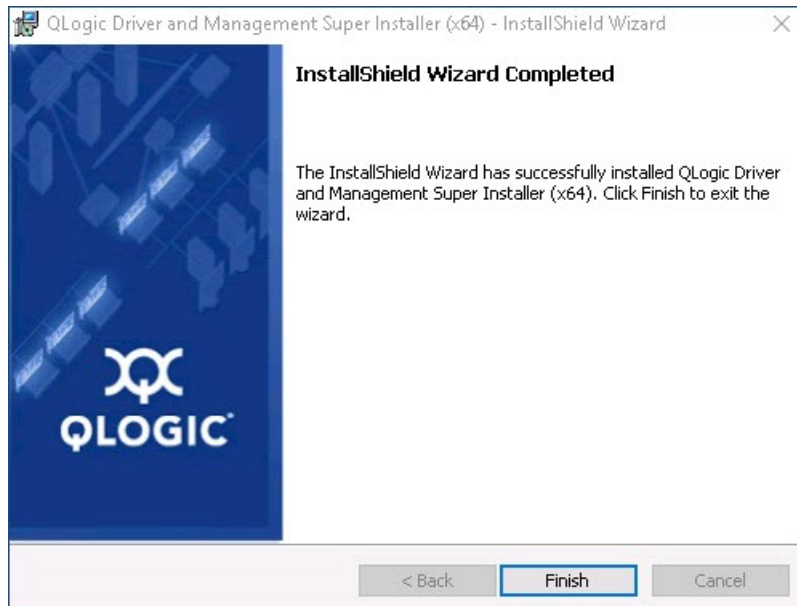


Figure 3-7. InstallShield Wizard: Completed Window

9. In the Dell Update Package window (Figure 3-8), “Update installer operation was successful” indicates completion.
 - (Optional) To open the log file, click **View Installation Log**. The log file shows the progress of the DUP installation, any previous installed versions, any error messages, and other information about the installation.
 - To close the Update Package window, click **CLOSE**.

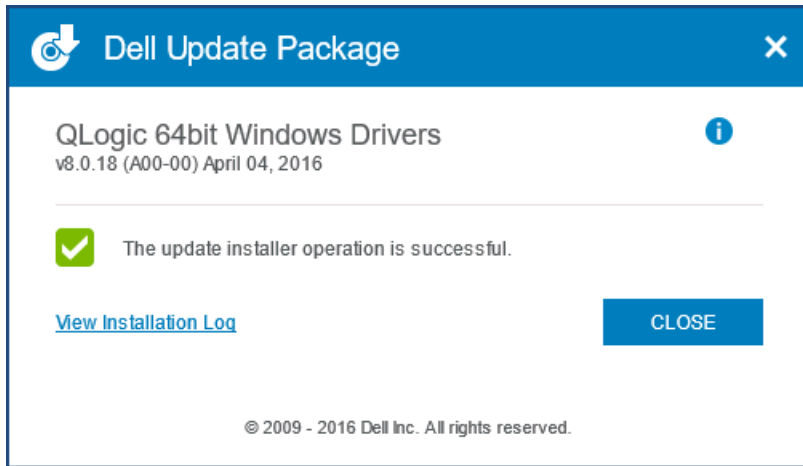


Figure 3-8. Dell Update Package Window

DUP Installation Options

To customize the DUP installation behavior, use the following command line options.

- To extract only the driver components to a directory:

`/drivers=<path>`

NOTE

This command requires the `/s` option.

- To install or update only the driver components:

`/driveronly`

NOTE

This command requires the `/s` option.

- (Advanced) Use the `/passthrough` option to send all text following `/passthrough` directly to the QLogic installation software of the DUP. This mode suppresses any provided GUIs, but not necessarily those of the QLogic software.

`/passthrough`

- (Advanced) To return a coded description of this DUP's supported features:

`/capabilities`

NOTE

This command requires the `/s` option.

DUP Installation Examples

The following examples show how to use the installation options.

To update the system silently:

```
<DUP_file_name>.exe /s
```

To extract the update contents to the `C:\mydir\` directory:

```
<DUP_file_name>.exe /s /e=C:\mydir
```

To extract the driver components to the `C:\mydir\` directory:

```
<DUP_file_name>.exe /s /drivers=C:\mydir
```

To install only the driver components:

```
<DUP_file_name>.exe /s /driveronly
```

To change from the default log location to `C:\my path with spaces\log.txt`:

```
<DUP_file_name>.exe /l="C:\my path with spaces\log.txt"
```

Removing the Windows Drivers

To remove the Windows drivers:

1. In the Control Panel, click **Programs**, and then click **Programs and Features**.
2. In the list of programs, select **QLogic FastLinQ Driver Installer**, and then click **Uninstall**.
3. Follow the instructions to remove the drivers.

Managing Adapter Properties

To view or change the 41000 Series Adapter properties:

1. In the Control Panel, click **Device Manager**.
2. On the properties of the selected adapter, click the **Advanced** tab.
3. On the Advanced page ([Figure 3-9](#)), select an item under **Property** and then change the **Value** for that item as needed.

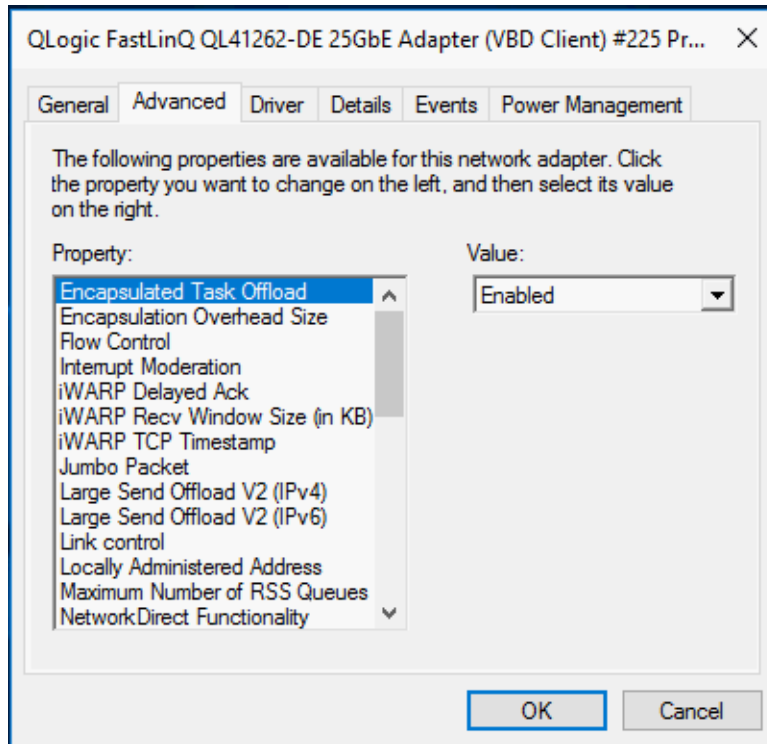


Figure 3-9. Setting Advanced Adapter Properties

Setting Power Management Options

You can set power management options to allow the operating system to turn off the controller to save power or to allow the controller to wake up the computer. If the device is busy (servicing a call, for example), the operating system will not shut down the device. The operating system attempts to shut down every possible device only when the computer attempts to go into hibernation. To have the controller remain on at all times, do not select the **Allow the computer to turn off the device to save power** check box (Figure 3-10).

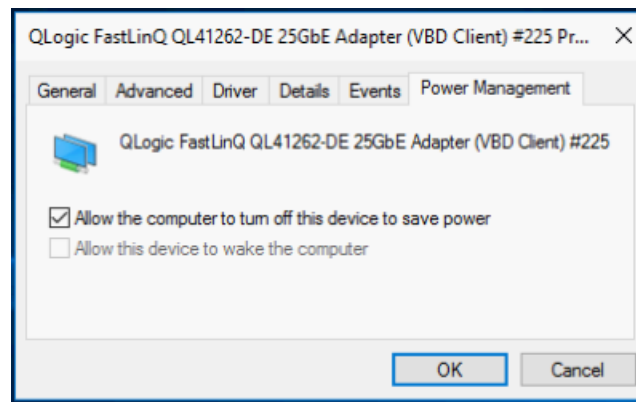


Figure 3-10. Power Management Options

NOTE

- The Power Management page is available only for servers that support power management.
 - Do not select the **Allow the computer to turn off the device to save power** check box for any adapter that is a member of a team.
-

Configuring the Communication Protocol to Use with QCC GUI, QCC PowerKit, and QCS CLI

There are two main components of the QCC GUI, QCC PowerKit, and QCS CLI management applications: the RPC agent and the client software. An RPC agent is installed on a server, or managed host, that contains one or more Converged Network Adapters. The RPC agent collects information on the Converged Network Adapters and makes it available for retrieval from a management PC on which the client software is installed. The client software enables viewing information from the RPC agent and configuring the Converged Network Adapters. The management software includes QCC GUI and QCS CLI.

A communication protocol enables communication between the RPC agent and the client software. Depending on the mix of operating systems (Linux, Windows, or both) on the clients and managed hosts in your network, you can choose an appropriate utility.

For installation instructions for these management applications, refer to the following documents on the Marvell Web site:

- *User's Guide, QLogic Control Suite CLI* (part number BC0054511-00)
- *User's Guide, PowerShell* (part number BC0054518-00)
- *Installation Guide, QConvergeConsole GUI* (part number SN0051105-00)

Link Configuration in Windows

Link configuration can be done in the Windows OS with three different parameters, which are available for configuration on the Advanced tab of the Device Manager page.

Link Control Mode

There are two modes for controlling link configuration:

- **Preboot Controlled** is the default mode. In this mode, the driver uses the link configuration from the device, which is configurable from preboot components. This mode ignores the link parameters on the Advanced tab.
- **Driver Controlled** mode should be set when you want to configure the link settings from Advanced tab of the Device Manager (as shown in [Figure 3-11](#)).

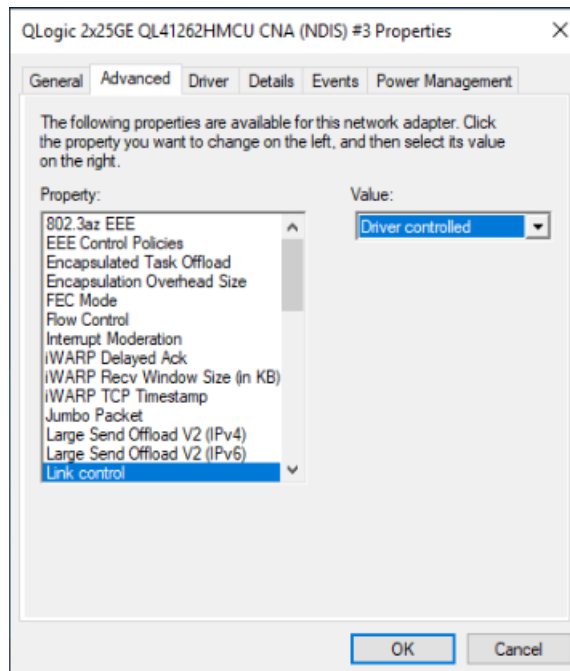


Figure 3-11. Setting Driver Controlled Mode

Link Speed and Duplex

The Speed & Duplex property (on the Advanced tab of the Device Manager) can be configured to any selection in the Value menu (see [Figure 3-12](#)).

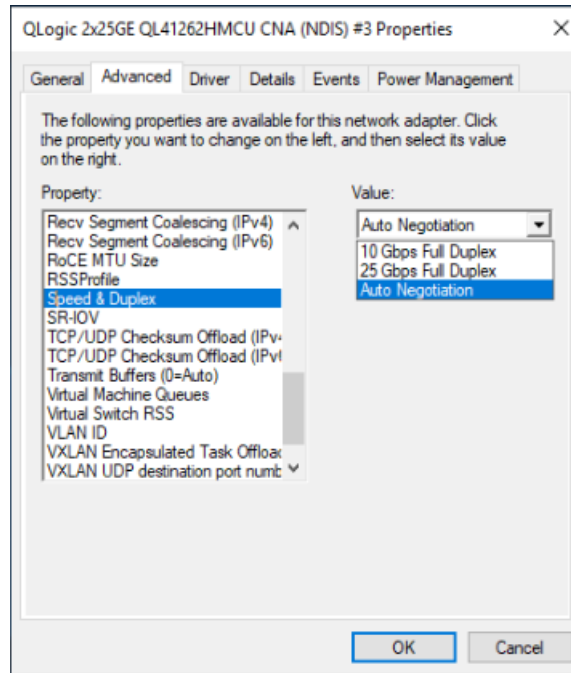


Figure 3-12. Setting the Link Speed and Duplex Property

This configuration is effective only when the link control property is set to Driver controlled (see [Figure 3-11](#)).

FEC Mode

FEC mode configuration at the OS level involves three driver advanced properties.

To set FEC mode:

1. Set Link Control. On the Advanced tab of the Device Manager:
 - a. In the Property menu, select **Link control**.
 - b. In the Value menu, select **Driver controlled**.See [Figure 3-11](#) for an example.
2. Set Speed & Duplex. On the Advanced tab of the Device Manager:
 - a. In the Property menu, select **Speed & Duplex**.
 - b. In the Value menu, select a fixed speed.

FEC mode configuration is active only when Speed & Duplex is set to a fixed speed. Setting this property to Auto Negotiation disables FEC configuration.

3. Set FEC Mode. On the Advanced tab of the Device Manager:
 - a. In the Property menu, select **FEC Mode**.
 - b. In the Value menu, select a valid value (see [Figure 3-13](#)).

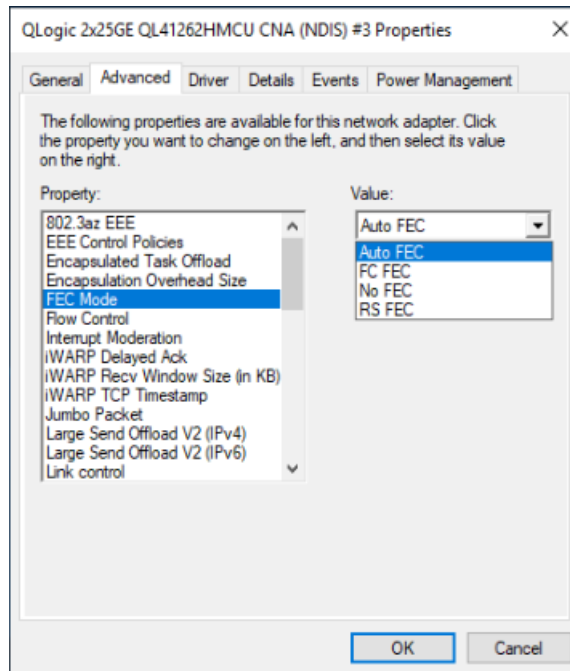


Figure 3-13. Setting the FEC Mode Property

This property is in effect only when [Step 1](#) and [Step 2](#) have been completed. All FEC modes are not valid for each media; you must know the valid modes for your specific media. If the wrong FEC mode value is set, the link goes down.

Installing VMware Driver Software

This section describes the qedentv VMware ESXi driver for the 41000 Series Adapters:

- [VMware Drivers and Driver Packages](#)
- [Installing VMware Drivers](#)
- [VMware NIC Driver Optional Parameters](#)

- [VMware Driver Parameter Defaults](#)
- [Removing the VMware Driver](#)
- [FCoE Support](#)
- [iSCSI Support](#)

VMware Drivers and Driver Packages

Table 3-5 lists the VMware ESXi drivers for the protocols.

Table 3-5. VMware Drivers

VMware Driver	Description
qedentv	Native networking driver
qedrntv	Native RDMA-Offload (RoCE and RoCEv2) driver
qedf	Native FCoE-Offload driver
qedil	Legacy iSCSI-Offload driver
qedi	Native iSCSI-Offload driver (ESXi 6.7 and later) ^a
qedentv_ens	Enhanced Network Stack (ENS) poll mode driver (PMD)

^a For ESXi 6.7 and 7.0, the NIC, RoCE, FCoE, and iSCSI drivers have been combined as a single component package. This package can be installed using standard ESXi installation methods and commands.

The ESXi drivers are included as individual driver packages and are not bundled together, except as noted.

The VMware drivers are available for download only from the VMware web site:

https://www.vmware.com/resources/compatibility/search.php?deviceCategory=io&details=1&keyword=QL41&page=1&display_interval=10&sortColumn=Partner&sortOrder=Asc

Install individual drivers using either:

- Standard ESXi package installation commands (see [Installing VMware Drivers](#))
- Procedures in the individual driver Read Me files
- Procedures in the following VMware KB article:
https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2137853

You should install the NIC driver first, followed by the storage drivers.

Installing VMware Drivers

This section provides instructions for:

- “Installing or Upgrading the Standard Driver” on page 36
- “Installing the Enhanced Network Stack Poll Mode Driver” on page 37

Installing or Upgrading the Standard Driver

You can use the driver ZIP file to install a new driver or update an existing driver. Be sure to install the entire driver set from the same driver ZIP file. Mixing drivers from different ZIP files will cause problems.

To install the VMware driver:

1. Download the VMware driver for the 41000 Series Adapter from the VMware support page:

www.vmware.com/support.html

2. Power up the ESX host, and then log into an account with administrator authority.
3. Use the Linux scp utility to copy the driver bundle from a local system into the `/tmp` directory on an ESX server with IP address 10.10.10.10. For example, issue the following command:

```
# scp qedentv-bundle-2.0.3.zip root@10.10.10.10:/tmp
```

You can place the file anywhere that is accessible to the ESX console shell.

4. Place the host in maintenance mode by issuing the following command:

```
# esxcli --maintenance-mode
```

NOTE

The maximum number of supported qedentv Ethernet interfaces on an ESXi host is 32 because the vmkernel allows only 32 interfaces to register for management callback.

5. Select one of the following installation options:

- ❑ **Option 1:** Install the driver bundle (which will install all of the driver VIBs at one time) by issuing the following command:

```
# esxcli software vib install -d /tmp/qedentv-2.0.3.zip
```

- ❑ **Option 2:** Install the `.vib` directly on an ESX server using either the CLI or the VMware Update Manager (VUM). To do this, unzip the driver ZIP file, and then extract the `.vib` file.

- To install the `.vib` file using the CLI, issue the following command. Be sure to specify the full `.vib` file path:

```
# esxcli software vib install -v /tmp/qedentv-1.0.3.11-10EM.550.0.0.1331820.x86_64.vib
```

- To install the `.vib` file using the VUM, see the knowledge base article here:

[Updating an ESXi/ESX host using VMware vCenter Update Manager 4.x and 5.x \(1019545\)](#)

To upgrade the existing driver bundle:

- Issue the following command:

```
# esxcli software vib update -d /tmp/qedentv-bundle-2.0.3.zip
```

To upgrade an individual driver:

Follow the steps for a new installation (see **To install the VMware driver**), except replace the command in Option 1 with the following:

```
# esxcli software vib update -v /tmp/qedentv-1.0.3.11-10EM.550.0.0.1331820.x86_64.vib
```

Installing the Enhanced Network Stack Poll Mode Driver

The Enhanced Network Stack (ENS) Poll Mode Driver (PMD) allows the 41000 Series Adapter to be used as a virtual NIC (vNIC) attached to an NSX-Transformers (NSX-T) managed virtual distribution switch (N-VDS). The ENS feature is supported on VMware ESXi 6.7. For more information about ENS, see [“VMware ESXi Enhanced Networking Stack Support” on page 320](#).

NOTE

Before installing the ENS driver, perform the steps in [“Installing or Upgrading the Standard Driver” on page 36](#) to install the NIC driver (qedentv).

To install the PMD:

1. Install the ENS PMD (qedentv-ens) VMware installation bundle (VIB) by issuing the following command:

```
esxcli software vib install -v /tmp/  
qedentv-ens-3.40.2.1-1OEM.670.0.0.8169922.x86_64.vib
```

2. Reboot the host.
3. Issue the following command and check the output:

```
[root@ens-iovp-host:~] esxcfg-nics -e
```

Name	Driver	ENS Capable	ENS Driven	MAC Address	Description
vmnic4	qedentv	True	False	94:f1:28:b4:9d:02	QLogic Corp. QLogic FastLinQ QL41xxx 1/10/25 GbE Ethernet Adapter
vmnic5	qedentv	True	False	94:f1:28:b4:9d:03	QLogic Corp. QLogic FastLinQ QL41xxx 1/10/25 GbE Ethernet Adapter

Ensure that the `ENS Capable` field is set to `True`, indicating that the ENS driver VIB is installed on that host.

In this example, the `ENS Driven` field is `False`, indicating that there is an ENS driver VIB installed on that host. The interfaces are ENS capable, but not ENS driven.

The ENS driver claims only the devices that are connected to the N-VDS virtual switch (vSwitch).

4. To make the interfaces ENS driven, configure the N-VDS switch using the NSX-T manager, and then add these uplinks to that N-VDS switch by following the instructions in [“Installing and Configuring an ENS-capable N-VDS” on page 322](#).
5. After the uplinks are successfully added to the N-VDS switch, issue the following command and check the output to ensure that `ENS Driven` is set to `True`.

```
[root@ens-iovp-host:~] esxcfg-nics -e
```

Name	Driver	ENS Capable	ENS Driven	MAC Address	Description
vmnic4	qedentv	True	True	94:f1:28:b4:9d:02	QLogic Corp. QLogic FastLinQ QL41xxx 1/10/25GbE Ethernet Adapter
vmnic5	qedentv	True	True	94:f1:28:b4:9d:03	QLogic Corp. QLogic FastLinQ QL41xxx 1/10/25GbE Ethernet Adapter

6. Issue the following command and check the output to ensure that the VMware NIC interfaces are claimed by the ENS PMD, `qedentv_ens`.

```
[root@ens-iovp-host:~] esxcfg-nics-1 2020-05-27 08:53:45
Name PCI Driver Link Speed Duplex MAC Address MTU Description
vmnic4 0000:88:00.0 qedentv_ens Up 25000Mbps Full 94:f1:28:b4:9d:02 1500 QLogic Corp.
QLogic FastLinQ QL41xxx 1/10/25 GbE Ethernet Adapter
vmnic5 0000:88:00.1 qedentv_ens Up 25000Mbps Full 94:f1:28:b4:9d:03 1500 QLogic Corp.
QLogic FastLinQ QL41xxx 1/10/25 GbE Ethernet Adapter
```

VMware NIC Driver Optional Parameters

Table 3-6 describes the optional parameters that can be supplied as command line arguments to the `esxcfg-module` command.

Table 3-6. VMware NIC Driver Optional Parameters

Parameter	Description
<code>hw_vlan</code>	Globally enables (1) or disables (0) hardware vLAN insertion and removal. Disable this parameter when the upper layer needs to send or receive fully formed packets. <code>hw_vlan=1</code> is the default.
<code>num_queues</code>	Specifies the number of TX/RX queue pairs. <code>num_queues</code> can be 1-11 or one of the following: <ul style="list-style-type: none"> ■ -1 allows the driver to determine the optimal number of queue pairs (default). ■ 0 uses the default queue. You can specify multiple values delimited by commas for multiport or multi-function configurations.
<code>multi_rx_filters</code>	Specifies the number of RX filters per RX queue, excluding the default queue. <code>multi_rx_filters</code> can be 1-4 or one of the following values: <ul style="list-style-type: none"> ■ -1 uses the default number of RX filters per queue. ■ 0 disables RX filters.
<code>disable_tpa</code>	Enables (0) or disables (1) the TPA (LRO) feature. <code>disable_tpa=0</code> is the default.
<code>max_vfs</code>	Specifies the number of virtual functions (VFs) per physical function (PF). <code>max_vfs</code> can be 0 (disabled) or 64 VFs on a single port (enabled). The 64 VF maximum support for ESXi is an OS resource allocation constraint.

Table 3-6. VMware NIC Driver Optional Parameters (Continued)

Parameter	Description
RSS	<p>Specifies the number of receive side scaling queues used by the host or virtual extensible LAN (VXLAN) tunneled traffic for a PF. RSS can be 2, 3, 4, or one of the following values:</p> <ul style="list-style-type: none"> ■ -1 uses the default number of queues. ■ 0 or 1 disables RSS queues. <p>You can specify multiple values delimited by commas for multiport or multi-function configurations.</p>
debug	<p>Specifies the level of data that the driver records in the <code>vmkernel</code> log file. <code>debug</code> can have the following values, shown in increasing amounts of data:</p> <ul style="list-style-type: none"> ■ <code>0x80000000</code> indicates Notice level. ■ <code>0x40000000</code> indicates Information level (includes the Notice level). ■ <code>0x3FFFFFFF</code> indicates Verbose level for all driver submodules (includes the Information and Notice levels).
auto_fw_reset	<p>Enables (1) or disables (0) the driver automatic firmware recovery capability. When this parameter is enabled, the driver attempts to recover from events such as transmit timeouts, firmware asserts, and adapter parity errors. The default is <code>auto_fw_reset=1</code>.</p>
vxlan_filter_en	<p>Enables (1) or disables (0) the VXLAN filtering based on the outer MAC, the inner MAC, and the VXLAN network (VNI), directly matching traffic to a specific queue. The default is <code>vxlan_filter_en=1</code>. You can specify multiple values delimited by commas for multiport or multifunction configurations.</p>
enable_vxlan_offld	<p>Enables (1) or disables (0) the VXLAN tunneled traffic checksum offload and TCP segmentation offload (TSO) capability. The default is <code>enable_vxlan_offld=1</code>. You can specify multiple values delimited by commas for multiport or multifunction configurations.</p>

VMware Driver Parameter Defaults

Table 3-7 lists the VMware driver parameter default values.

Table 3-7. VMware Driver Parameter Defaults

Parameter	Default
Speed	<p>Autonegotiation with all speeds advertised. The speed parameter must be the same on all ports. If autonegotiation is enabled on the device, all of the device ports will use autonegotiation.</p>

Table 3-7. VMware Driver Parameter Defaults (Continued)

Parameter	Default
Flow Control	Autonegotiation with RX and TX advertised
MTU	1,500 (range 46–9,600) (ESXi 7.0 range 60–9,190)
Rx Ring Size	8,192 (range 128–8,192)
Tx Ring Size	8,192 (range 128–8,192)
MSI-X	Enabled
Transmit Send Offload (TSO)	Enabled
Large Receive Offload (LRO)	Enabled
RSS	Enabled (four RX queues)
HW VLAN	Enabled
Number of Queues	Enabled (eight RX/TX queue pairs)
Wake on LAN (WoL)	Disabled
multi_rss	Enabled
DRSS	Enabled (four RX queues)

When the `RSS` and `DRSS` module parameters are used simultaneously, the values provided to `DRSS` take precedence. In addition, both `RSS` and `DRSS` use the value that was provided for `DRSS`.

To disable `RSS` on ESXi 6.7/7.0, disable both the `multi_rss` and `RSS` parameters. For example:

```
esxcfg-module -s 'multi_rss=0 RSS=0,0' qedentv
```

On ESXi 6.7 or later, you can control the number of `RSS` engines by varying the number of queues, as described in the following examples.

Example 1. In this example, DRSS is disabled. (By default, four RSS queues are used by drivers).

<code>num_queues = 4–7</code>	Enables one RSS engine
<code>num_queues = 8–11</code>	Enables two RSS engines
<code>num_queues = 12–15</code>	Enables three RSS engines
<code>num_queues = 16– maximum</code>	Enables four RSS engines

Example 2. In this example, DRSS is enabled (that is, `DRSS=4, 4`).

<code>num_queues = 3–6</code>	Enables one RSS engine
<code>num_queues = 7–10</code>	Enables two RSS engines
<code>num_queues = 11–14</code>	Enables three RSS engines
<code>num_queues = 15–maximum</code>	Enables four RSS engines

Following are the maximum and minimum RSS engine and RSS queues that are currently supported by the `qedentv` driver on ESXi 7.0:

Default number of RSS engines for a PF in single function mode	= 4
Default number of RSS engines for a PF in multifunction mode	= 2
Default number of secondary queues in an RSS engine	= 3
Maximum number of secondary queues in an RSS engine	= 5
Maximum number of total RSS queues in an RSS engine	= 6
Maximum number of RX queues supported for a PF	= 32
Maximum number of TX queues supported for a PF	= 32

In multifunction mode (switch independent or switch dependent), the first four PFs have the default number of RSS engines enabled if enough hardware or eCore resources are available. On the remaining PFs, RSS is disabled.

In SRI-OV only configuration, the behavior the same as single function mode.

In SR-IOV over NPar configuration (switch independent or switch dependent), RSS is disabled.

Removing the VMware Driver

To remove the `.vib` file (`qedentv`), issue the following command:

```
# esxcli software vib remove --vibName qedentv
```


To remove the driver, issue the following command:

```
# vmkload_mod -u qedentv
```

FCoE Support

The Marvell VMware FCoE qedf driver included in the VMware software package supports Marvell FastLinQ FCoE converged network interface controllers (C-NICs). The driver is a kernel-mode driver that provides a translation layer between the VMware SCSI stack and the Marvell FCoE firmware and hardware. The FCoE and DCB feature set is supported on VMware ESXi 5.0 and later.

To enable FCoE-Offload mode, see the *Application Note, Enabling Storage Offloads on Dell and Marvell FastLinQ 41000 Series Adapters* at <https://www.marvell.com/documents/5aa5otcbkr0im3ynera3/>.

iSCSI Support

The Marvell VMware iSCSI qedil Host Bus Adapter (HBA) driver, similar to qedf, is a kernel mode driver that provides a translation layer between the VMware SCSI stack and the Marvell iSCSI firmware and hardware. The qedil driver leverages the services provided by the VMware iscsid infrastructure for session management and IP services.

To enable iSCSI-Offload mode, see the *Application Note, Enabling Storage Offloads on Dell and Marvell FastLinQ 41000 Series Adapters* at <https://www.marvell.com/documents/5aa5otcbkr0im3ynera3/>.

NOTE

The iSCSI interface (iSCSI offload) supported by the 41000 Series Adapters is a dependent hardware interface that relies on networking services, iSCSI configuration, and management interfaces provided by VMware. The iSCSI interface includes two components: a network adapter and an iSCSI engine on the same interface. The iSCSI engine appears on the list of storage adapters as an iSCSI adapter (vmhba). For services such as ARP and DHCP needed by iSCSI, the iSCSI vmhba uses the services of the vmnic device created by the qedil driver. The vmnic is a thin dummy implementation intended to provide L2 functionality for iSCSI to operate. This is not a comprehensive L2 solution. Do not use this implementation to carry regular networking traffic; that is, do not assign this to a VM as a network adapter. The actual NIC interfaces on the adapter will be claimed by the qedentv driver, which is a fully-functional NIC driver. See the NOTE in “iSCSI Offload in VMware ESXi” on page 221 for a list of iSCSI Offload limitations.

Installing Citrix Hypervisor Driver Software

This section describes how to install the Citrix driver on a XenServer operating system using the driver update disk (DUD).

NOTE

The procedures in this section apply only to Citrix XenServer 8.2 and later distributions.

These procedures use both the DUD and the OS installation disk.

To install the Citrix hypervisor driver:

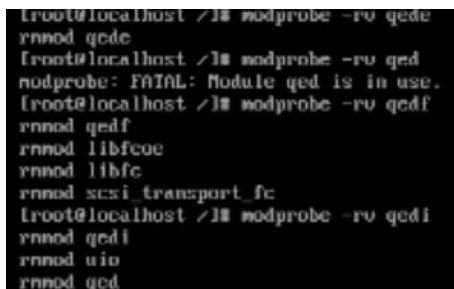
1. Insert the XenServer installation CD and begin the installation in shell mode (see [Figure 3-14](#)).



```
install
no-serial
safe
multipath
*shell
```

Figure 3-14. Starting in Shell Mode

2. When the system boots to shell mode, upload the inbox `qede`, `qed`, `qedi`, and `qedf` drivers (see [Figure 3-15](#)).



```
[root@localhost ~]# modprobe -rv qede
rmod qede
[root@localhost ~]# modprobe -rv qed
modprobe: FATAL: Module qed is in use.
[root@localhost ~]# modprobe -rv qedf
rmod qedf
rmod libfcoc
rmod libfc
rmod scsi transport fc
[root@localhost ~]# modprobe -rv qedi
rmod qedi
rmod uio
rmod qed
```

Figure 3-15. Installing the `qede`, `qed`, `qedi`, and `qedf` Drivers

3. Type `exit`, and then press ENTER, to return to the GUI installer.

4. Insert the DUD CD/ISO. The GUI Welcome screen appears (see [Figure 3-16](#)).

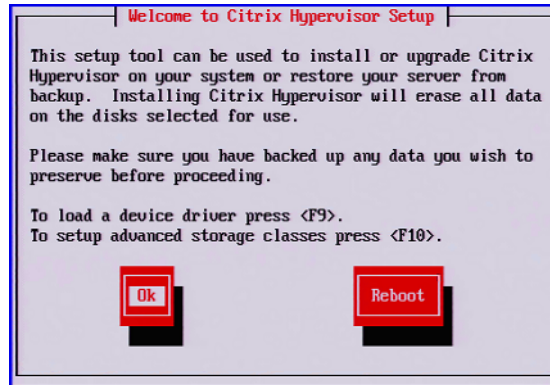


Figure 3-16. Loading the Device Driver

Press F9 to load the driver.

The Load Repository window appears (see [Figure 3-17](#).)

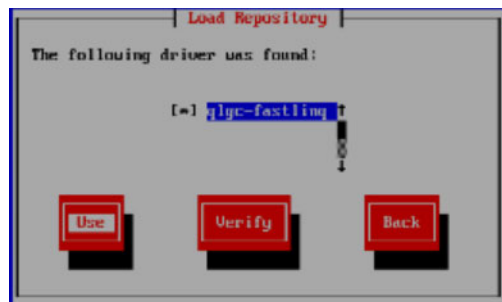


Figure 3-17. Locating the Device Driver

5. Click **Use**.
The Drivers Loaded window appears (see [Figure 3-18](#)).



Figure 3-18. Driver Installed Successfully

6. Press ALT+F2 to return to shell mode, and then load the out-of-box (OOB) driver (see [Figure 3-19](#)).

```
root@localhost ~# modprobe -u qede
insmod /lib/modules/4.19.0-1/kernel/lib/crc8.ko
insmod /lib/modules/4.19.0-1/updates/qed.ko
insmod /lib/modules/4.19.0-1/updates/qede.ko
```

Figure 3-19. Loading the OOB Driver

7. Press ALT+F1 to return to the GUI installer, and then continue the installation.
Do **not** remove the driver CD/ISO.
8. When prompted, skip the supplemental package installation.
9. When prompted, reboot the system after removing the OS installer CD and the DUD.

The hypervisor should boot with the new driver installed.

4 Upgrading the Firmware

This chapter provides information about upgrading the firmware using the Dell Update Package (DUP).

The firmware DUP is a Flash update utility only; it is not used for adapter configuration. You can run the firmware DUP by double-clicking the executable file. Alternatively, you can run the firmware DUP from the command line with several supported command line options.

- [Running the DUP by Double-Clicking](#)
- [“Running the DUP from a Command Line” on page 49](#)
- [“Running the DUP Using the .bin File” on page 50 \(Linux only\)](#)

Running the DUP by Double-Clicking

To run the firmware DUP by double-clicking the executable file:

1. Double-click the icon representing the firmware Dell Update Package file.
2. The Dell Update Package splash screen appears, as shown in [Figure 4-1](#). Click **Install** to continue.

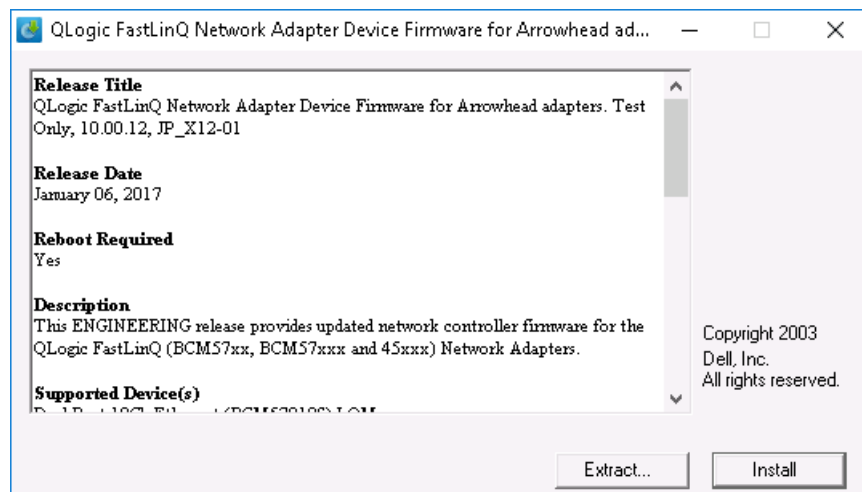


Figure 4-1. Dell Update Package: Splash Screen

3. Follow the on-screen instructions. In the Warning dialog box, click **Yes** to continue the installation.

The installer indicates that it is loading the new firmware, as shown in [Figure 4-2](#).

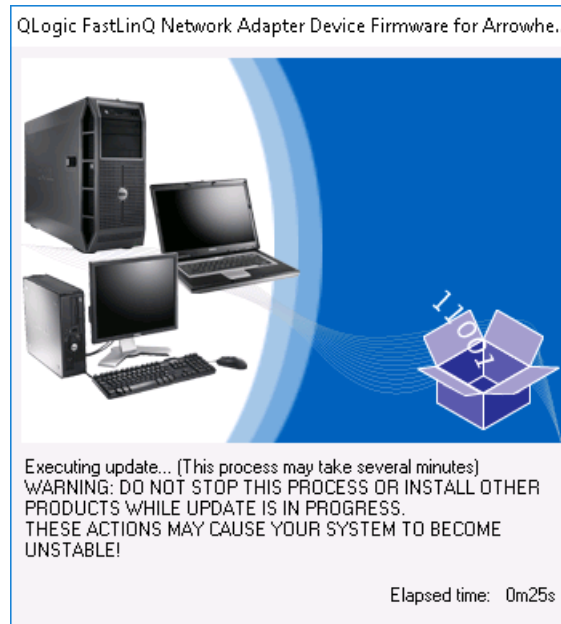


Figure 4-2. Dell Update Package: Loading New Firmware

When complete, the installer indicates the result of the installation, as shown in [Figure 4-3](#).

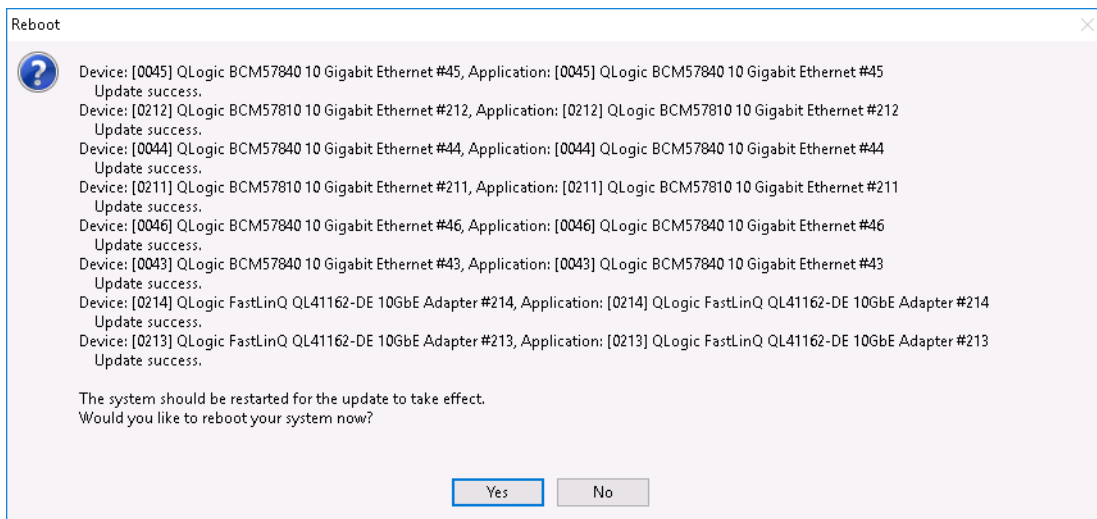


Figure 4-3. Dell Update Package: Installation Results

Figure 4-5 shows the options that you can use to customize the Dell Update Package installation.

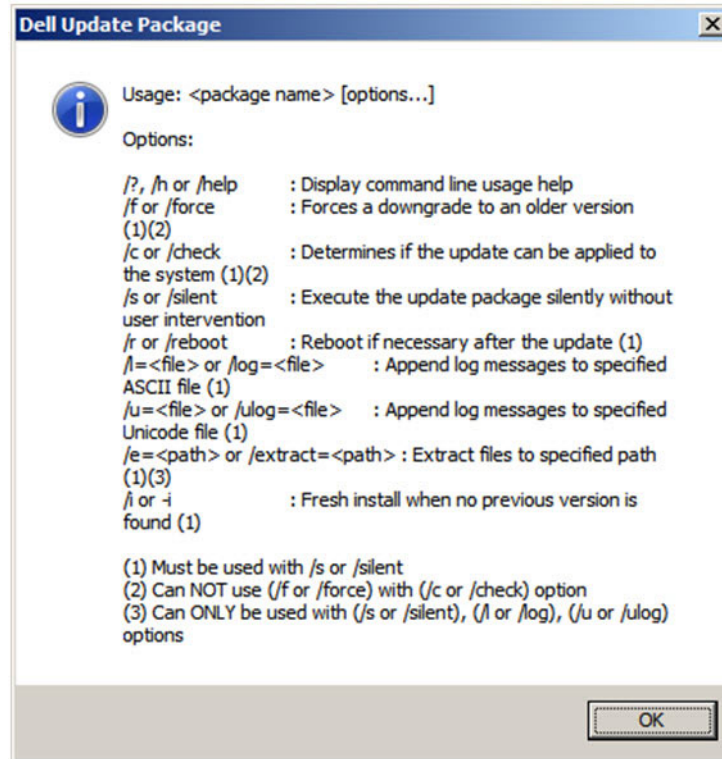


Figure 4-5. DUP Command Line Options

Running the DUP Using the .bin File

The following procedure is supported only on Linux OSs with the following minimum driver requirements:

- qed Linux driver version 8.3.4.0 and later
- qede Linux driver version 8.3.4.0 and later
- qede (inbox) Linux driver version 8.33.0.20 on RHEL 7.6, RHEL 8.0, SLES 12.4, SLES 15.1, and later

To update the DUP using the .bin file:

1. Copy the `Network_Firmware_NJCX1_LN_X.Y.Z.BIN` file to the system or server.
2. Change the file type into an executable file as follows:

```
chmod 777 Network_Firmware_NJCX1_LN_X.Y.Z.BIN
```


3. To start the update process, issue the following command:
`./Network_Firmware_NJCX1_LN_X.Y.Z.BIN`
4. After the firmware is updated, reboot the system.

Example output from the SUT during the DUP update:

```
./Network_Firmware_NJCX1_LN_08.07.26.BIN
Collecting inventory...
Running validation...
BCM57810 10 Gigabit Ethernet rev 10 (p2p1)
The version of this Update Package is the same as the currently installed
version.
Software application name: BCM57810 10 Gigabit Ethernet rev 10 (p2p1)
Package version: 08.07.26
Installed version: 08.07.26
BCM57810 10 Gigabit Ethernet rev 10 (p2p2)
The version of this Update Package is the same as the currently installed
version.
Software application name: BCM57810 10 Gigabit Ethernet rev 10 (p2p2)
Package version: 08.07.26
Installed version: 08.07.26
Continue? Y/N:Y
Y entered; update was forced by user
Executing update...
WARNING: DO NOT STOP THIS PROCESS OR INSTALL OTHER DELL PRODUCTS WHILE UPDATE
IS IN PROGRESS.
THESE ACTIONS MAY CAUSE YOUR SYSTEM TO BECOME UNSTABLE!
.....
Device: BCM57810 10 Gigabit Ethernet rev 10 (p2p1)
  Application: BCM57810 10 Gigabit Ethernet rev 10 (p2p1)
  Update success.
Device: BCM57810 10 Gigabit Ethernet rev 10 (p2p2)
  Application: BCM57810 10 Gigabit Ethernet rev 10 (p2p2)
  Update success.
Would you like to reboot your system now?
Continue? Y/N:Y
```

5 Adapter Preboot Configuration

During the host boot process, you have the opportunity to pause and perform adapter management tasks using the Human Infrastructure Interface (HII) application. These tasks include the following:

- [“Getting Started” on page 53](#)
- [“Displaying Firmware Image Properties” on page 60](#)
- [“Configuring Device-level Parameters” on page 61](#)
- [“Configuring NIC Parameters” on page 62](#)
- [“Configuring Data Center Bridging” on page 65](#)
- [“Configuring FCoE Boot” on page 67](#)
- [“Configuring iSCSI Boot” on page 68](#)
- [“Configuring Partitions” on page 73](#)

NOTE

The HII screen shots in this chapter are representative and may not match the screens that you see on your system.

Getting Started

To start the HII application:

1. Open the System Setup window for your platform. For information about launching the System Setup, consult the user guide for your system.
2. In the System Setup window ([Figure 5-1](#)), select **Device Settings**, and then press ENTER.

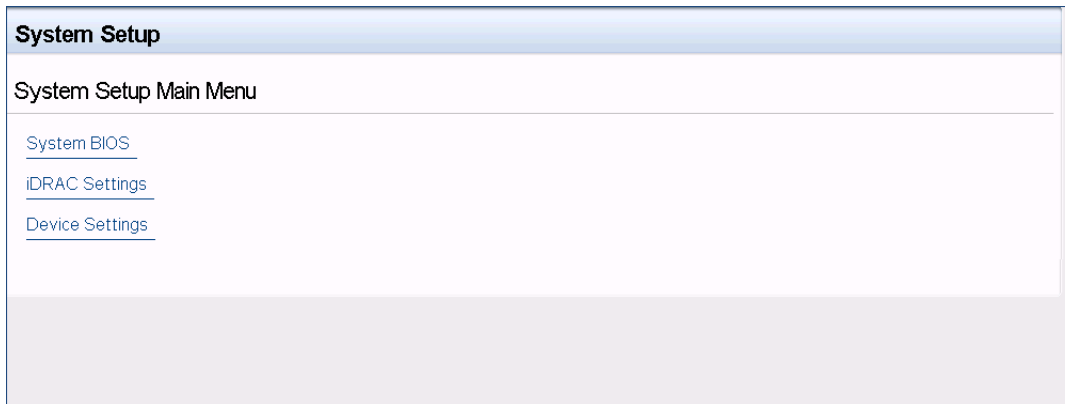


Figure 5-1. System Setup

3. In the Device Settings window ([Figure 5-2](#)), select the 41000 Series Adapter port that you want to configure, and then press ENTER.

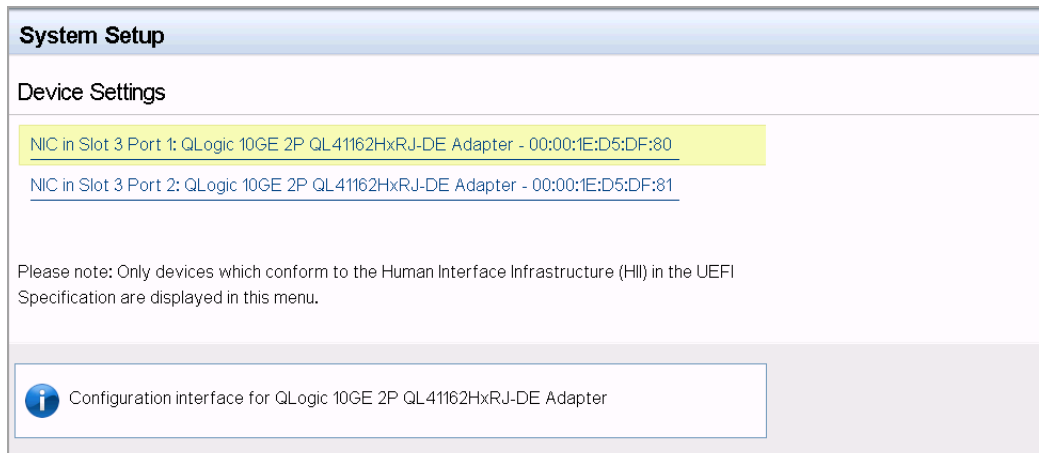


Figure 5-2. System Setup: Device Settings

The Main Configuration Page (Figure 5-3) presents the adapter management options where you can set the partitioning mode.

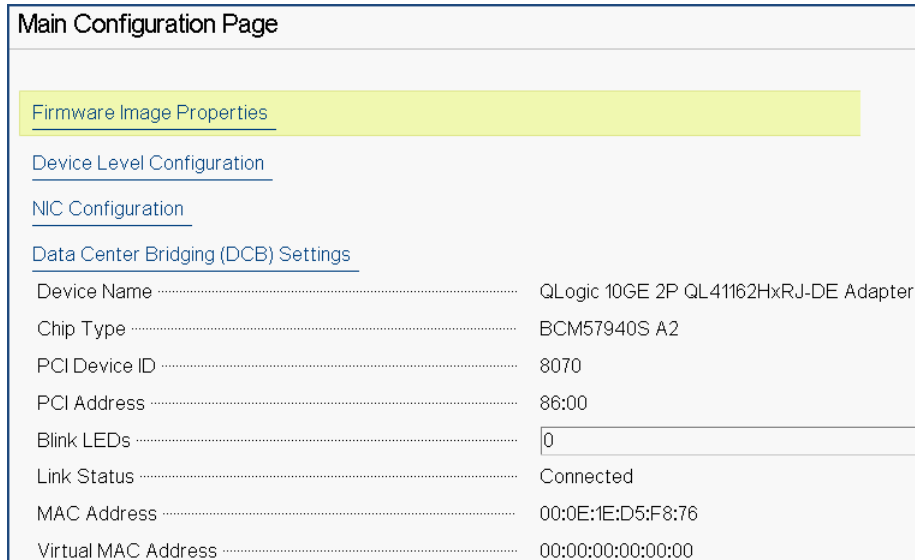


Figure 5-3. Main Configuration Page

4. Under **Device Level Configuration**, set the **Partitioning Mode** to **NPAR** to add the **NIC Partitioning Configuration** option to the Main Configuration Page, as shown in Figure 5-4.

NOTE

NPar is not available on ports with a maximum speed of 1G.

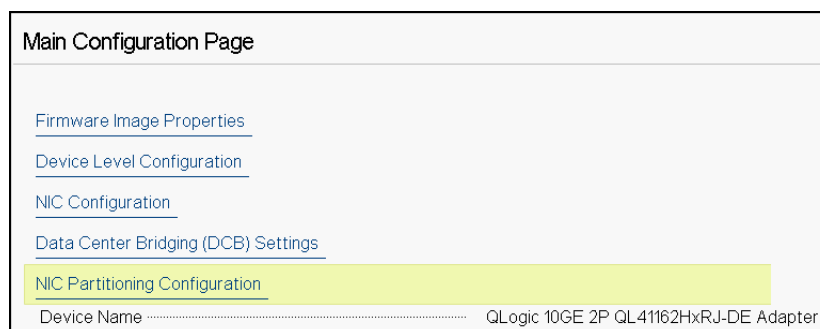


Figure 5-4. Main Configuration Page, Setting Partitioning Mode to NPar

In Figure 5-3 and Figure 5-4, the Main Configuration Page shows the following:

- **Firmware Image Properties** (see “Displaying Firmware Image Properties” on page 60)

- **Device Level Configuration** (see “[Configuring Device-level Parameters](#)” on page 61)
- **NIC Configuration** (see “[Configuring NIC Parameters](#)” on page 62)
- **iSCSI Configuration** (if iSCSI remote boot is allowed by enabling iSCSI offload in NPar mode on the port’s third partition) (see “[Configuring iSCSI Boot](#)” on page 68)
- **FCoE Configuration** (if FCoE boot from SAN is allowed by enabling FCoE offload in NPar mode on the port’s second partition) (see “[Configuring FCoE Boot](#)” on page 67)
- **Data Center Bridging (DCB) Settings** (see “[Configuring Data Center Bridging](#)” on page 65)
- **NIC Partitioning Configuration** (if NPAR is selected on the Device Level Configuration page) (see “[Configuring Partitions](#)” on page 73)

In addition, the Main Configuration Page presents the adapter properties listed in [Table 5-1](#).

Table 5-1. Adapter Properties

Adapter Property	Description
Device Name	Factory-assigned device name
Chip Type	ASIC version
PCI Device ID	Unique vendor-specific PCI device ID
PCI Address	PCI device address in bus-device function format
Blink LEDs	User-defined blink count for the port LED
Link Status	External link status
MAC Address	Manufacturer-assigned permanent device MAC address
Virtual MAC Address	User-defined device MAC address
iSCSI MAC Address ^a	Manufacturer-assigned permanent device iSCSI Offload MAC address
iSCSI Virtual MAC Address ^a	User-defined device iSCSI Offload MAC address
FCoE MAC Address ^b	Manufacturer-assigned permanent device FCoE Offload MAC address
FCoE Virtual MAC Address ^b	User-defined device FCoE Offload MAC address

Table 5-1. Adapter Properties (Continued)

Adapter Property	Description
FCoE WWPN ^b	Manufacturer-assigned permanent device FCoE Offload WWPN (world wide port name)
FCoE Virtual WWPN ^b	User-defined device FCoE Offload WWPN
FCoE WWNN ^b	Manufacturer-assigned permanent device FCoE Offload WWNN (world wide node name)
FCoE Virtual WWNN ^b	User-defined device FCoE Offload WWNN

^a This property is visible only if **iSCSI Offload** is enabled on the NIC Partitioning Configuration page.

^b This property is visible only if **FCoE Offload** is enabled on the NIC Partitioning Configuration page.

Default NPar/NParEP Mode Numbering

The 41000 Series Adapter uses the following PCI Bus:Device.Function numbering and MAC Addressing for Dual Port (Table 5-2 and Table 5-3) and Quad Port modes (Table 5-4 and Table 5-5). The MAC addresses do not normally not start at ...:00, but is shown as such in these examples to make the MAC address incrementation more obvious. The *aa:bb:cc:dd:ee:XX* numbering are place holders for the actual values. The PCI Bus:Device.Function numbering normally has the Device number as 00 or 01, while the Function numbering always starts at 00, and increments from there.

Table 5-2. Dual-Port QL41xx2 Default Mode

	Allowed Mode	Ethernet PF	DCBX LLDP ^a
Port 0	Bus:Device.Function Numbering	xx:00.00	
	MAC Addressing	aa.bb.cc.dd.ee.00	aa.bb.cc.dd.ee.10
Port 1	Bus:Device.Function Numbering	xx:00.01	
	MAC Addressing	aa.bb.cc.dd.ee.01	aa.bb.cc.dd.ee.11

^a DCB mode only

Table 5-3. Dual-Port QL41xx2 NParEP Mode

Partition Number		1	2	3	4	5 ^a	6 ^a	7 ^a	8 ^a	—
Allowed Mode		Ethernet PF	Disabled or Ethernet or FCoE-Offload ^b PF	Disabled or Ethernet or iSCSI-Offload ^b PF	Disabled or Ethernet PF	Disabled or Ethernet PF	Disabled or Ethernet PF	Disabled or Ethernet PF	Disabled or Ethernet PF	DCBX LLDP ^c
Port 0	Bus:Device. Function Numbering	xx:00.00	xx:00.02	xx:00.04	xx:00.06	xx:01.00	xx:01.02	xx:01.04	xx:01.06	—
	MAC Addressing	aa.bb.cc.dd.ee.00	aa.bb.cc.dd.ee.02	aa.bb.cc.dd.ee.04	aa.bb.cc.dd.ee.06	aa.bb.cc.dd.ee.08	aa.bb.cc.dd.ee.0a	aa.bb.cc.dd.ee.0c	aa.bb.cc.dd.ee.0e	aa.bb.cc.dd.ee.10
Port 1	Bus:Device. Function Numbering	xx:00.01	xx:00.03	xx:00.05	xx:00.07	xx:01.01	xx:01.03	xx:01.05	xx:01.07	—
	MAC Addressing	aa.bb.cc.dd.ee.01	aa.bb.cc.dd.ee.03	aa.bb.cc.dd.ee.05	aa.bb.cc.dd.ee.07	aa.bb.cc.dd.ee.09	aa.bb.cc.dd.ee.0b	aa.bb.cc.dd.ee.0d	aa.bb.cc.dd.ee.0f	aa.bb.cc.dd.ee.11

^a In NPar (four PFs per physical port) mode, these are hidden.

^b CNA only

^c DCB mode only

Table 5-4. Quad-Port QL41xx4 Default Mode

	Allowed Mode	Ethernet PF	DCBX LLDP ^a
Port 0	Bus:Device.Function Numbering	xx:00.00	—
	MAC Addressing	aa.bb.cc.dd.ee.00	aa.bb.cc.dd.ee.10
Port 1	Bus:Device.Function Numbering	xx:00.01	—
	MAC Addressing	aa.bb.cc.dd.ee.01	aa.bb.cc.dd.ee.11
Port 2	Bus:Device.Function Numbering	xx:00.02	—
	MAC Addressing	aa.bb.cc.dd.ee.02	aa.bb.cc.dd.ee.12
Port 3	Bus:Device.Function Numbering	xx:00.03	—
	MAC Addressing	aa.bb.cc.dd.ee.03	aa.bb.cc.dd.ee.13

^a DCB mode only

Table 5-5. Quad-Port QL41xx4 NParEP Mode

Partition Number		1	2	3 ^a	4 ^a	—
		Ethernet PF	Disabled or Ethernet or (FCoE-Offload or iSCSI-Offload) ^b PF	Disabled or Ethernet PF	Disabled or Ethernet PF	DCBX LLDP ^c
Port 0	Bus:Device.Function Numbering	xx:00.00	xx:00.04	xx:01.00	xx:01.04	—
	MAC Addressing	aa.bb.cc.dd.ee.00	aa.bb.cc.dd.ee.04	aa.bb.cc.dd.ee.08	aa.bb.cc.dd.ee.0c	aa.bb.cc.dd.ee.10
Port 1	Bus:Device.Function Numbering	xx:00.01	xx:00.05	xx:01.01	xx:01.05	—
	MAC Addressing	aa.bb.cc.dd.ee.01	aa.bb.cc.dd.ee.05	aa.bb.cc.dd.ee.09	aa.bb.cc.dd.ee.0d	aa.bb.cc.dd.ee.11
Port 2	Bus:Device.Function Numbering	xx:00.02	xx:00.06	xx:01.02	xx:01.06	—
	MAC Addressing	aa.bb.cc.dd.ee.02	aa.bb.cc.dd.ee.06	aa.bb.cc.dd.ee.0a	aa.bb.cc.dd.ee.0e	aa.bb.cc.dd.ee.12
Port 3	Bus:Device.Function Numbering	xx:00.03	xx:00.07	xx:01.03	xx:01.07	—
	MAC Addressing	aa.bb.cc.dd.ee.03	aa.bb.cc.dd.ee.07	aa.bb.cc.dd.ee.0b	aa.bb.cc.dd.ee.0f	aa.bb.cc.dd.ee.13

On 2x1GBASE-T (RJ-45) plus 2x10G or 10G/25G interfaced rack NDCs (QL41624HMCU and QL41264HMRJ), the two 1GBASE-T ports (3 and 4) have only a single Ethernet PF (the other PFs are hidden).

^a In NPar (two PFs per physical port), these are hidden.

^b CNA only

^c DCB mode only

PCI Device IDs

The 41000 Series Adapter uses the PCI device IDs in [Table 5-6](#).

Table 5-6. QL41xxx PCI Device IDs

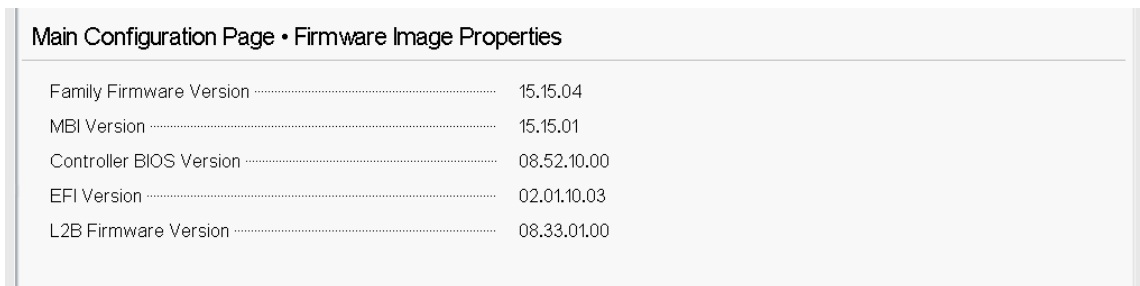
Function Type		
	Ethernet ^a	0x8070
	FCoE-Offload	0x8080
	iSCSI-Offload	0x8084
	SR-IOV VF	0x8090

^a RDMA-Offload (RoCE/iWARP) is under Ethernet

Displaying Firmware Image Properties

To view the properties for the firmware image, select **Firmware Image Properties** on the Main Configuration Page, and then press ENTER. The Firmware Image Properties page ([Figure 5-5](#)) specifies the following view-only data:

- **Family Firmware Version** is the multiboot image version, which comprises several firmware component images.
- **MBI Version** is the Marvell FastLinQ bundle image version that is active on the device.
- **Controller BIOS Version** is the management firmware version.
- **EFI Driver Version** is the extensible firmware interface (EFI) driver version.
- **L2B Firmware Version** is the NIC offload firmware version for boot.



Main Configuration Page • Firmware Image Properties

Family Firmware Version	15.15.04
MBI Version	15.15.01
Controller BIOS Version	08.52.10.00
EFI Version	02.01.10.03
L2B Firmware Version	08.33.01.00

Figure 5-5. Firmware Image Properties

Configuring Device-level Parameters

NOTE

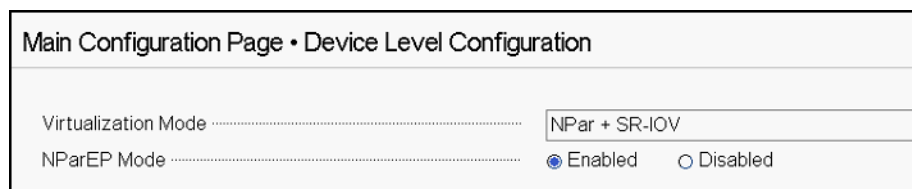
The iSCSI physical functions (PFs) are listed when the iSCSI Offload feature is enabled in NPar mode only. The FCoE PFs are listed when the FCoE Offload feature is enabled in NPar mode only. Not all adapter models support iSCSI Offload and FCoE Offload. Only one offload can be enabled per port, and only in NPar mode.

Device-level configuration includes the following parameters:

- **Virtualization Mode**
- **NParEP Mode**

To configure device-level parameters:

1. On the Main Configuration Page, select **Device Level Configuration** (see [Figure 5-3 on page 54](#)), and then press ENTER.
2. On the **Device Level Configuration** page, select values for the device-level parameters, as shown in [Figure 5-6](#).



Main Configuration Page • Device Level Configuration

Virtualization Mode	NPar + SR-IOV
NParEP Mode	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled

Figure 5-6. Device Level Configuration

NOTE

QL41264HMCU-DE (part number 5V6Y4) and QL41264HMRJ-DE (part number 0D1WT) adapters show support for NPar, SR-IOV and NParEP in the Device Level Configuration, though these features are not supported on 1Gbps ports 3 and 4.

3. For **Virtualization Mode**, select one of the following modes to apply to all adapter ports:
 - None** (default) specifies that no virtualization mode is enabled.
 - NPar** sets the adapter to switch-independent NIC partitioning mode.
 - SR-IOV** sets the adapter to SR-IOV mode.
 - NPar + SR-IOV** sets the adapter to SR-IOV over NPar mode.

4. **NParEP Mode** configures the maximum quantity of partitions per adapter. This parameter is visible when you select either **NPar** or **NPar + SR-IOV** as the **Virtualization Mode** in [Step 2](#).
 - Enabled** allows you to configure up to 16 partitions per adapter.
 - Disabled** allows you to configure up to 8 partitions per adapter.
5. Click **Back**.
6. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

Configuring NIC Parameters

NIC configuration includes setting the following parameters:

- **Link Speed**
- **NIC + RDMA Mode**
- **RDMA Protocol Support**
- **Boot Mode**
- **FEC Mode**
- **Energy Efficient Ethernet**
- **Virtual LAN Mode**
- **Virtual LAN ID**

To configure NIC parameters:

1. On the Main Configuration Page, select **NIC Configuration** ([Figure 5-3 on page 54](#)), and then click **Finish**.

[Figure 5-7](#) shows the NIC Configuration page.

The screenshot shows the 'Main Configuration Page • NIC Configuration' interface. It contains several configuration options with radio buttons and a text input field:

Link Speed	<input checked="" type="radio"/> Auto Negotiated	<input type="radio"/> 1 Gbps	<input type="radio"/> 10 Gbps	<input type="radio"/> 25 Gbps	<input type="radio"/> SmartAN
NIC + RDMA Mode	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled			
RDMA Protocol Support	<input checked="" type="radio"/> RoCE	<input type="radio"/> iWARP	<input type="radio"/> iWARP + RoCE		
Boot Mode	<input type="radio"/> PXE	<input checked="" type="radio"/> iSCSI	<input type="radio"/> Disabled		
Energy Efficient Ethernet	<input type="text" value="Optimal Power and Performance"/>				
Virtual LAN Mode	<input type="radio"/> Enabled	<input checked="" type="radio"/> Disabled			
Virtual LAN ID	<input type="text" value="1"/>				

Figure 5-7. NIC Configuration

2. Select one of the following **Link Speed** options for the selected port. Not all speed selections are available on all adapters.
 - Auto Negotiated** enables Auto Negotiation mode on the port. FEC mode selection is not available for this speed mode.
 - 1 Gbps** enables 1GbE fixed speed mode on the port. This mode is intended only for 1GbE interfaces and should not be configured for adapter interfaces that operate at other speeds. FEC mode selection is not available for this speed mode. This mode is not available on all adapters.
 - 10 Gbps** enables 10GbE fixed speed mode on the port. This mode is not available on all adapters.
 - 25 Gbps** enables 25GbE fixed speed mode on the port. This mode is not available on all adapters.
 - SmartAN** (Default) enables FastLinQ SmartAN™ link speed mode on the port. No FEC mode selection is available for this speed mode. The **SmartAN** setting cycles through all possible link speeds and FEC modes until a link is established. This mode is intended for use only with 25G interfaces. This mode is not available on all adapters.
3. For **NIC + RDMA Mode**, select either **Enabled** or **Disabled** for RDMA on the port. This setting applies to all partitions of the port, if in NPar mode.
4. **FEC Mode** is visible when **25 Gbps** fixed speed mode is selected as the **Link Speed** in [Step 2](#). For **FEC Mode**, select one of the following options. Not all FEC modes are available on all adapters.
 - None** disables all FEC modes.
 - Fire Code** enables Fire Code (BASE-R) FEC mode.
 - Reed Solomon** enables Reed Solomon FEC mode.
 - Auto** enables the port to cycle through **None**, **Fire Code**, and **Reed Solomon** FEC modes (at that link speed) in a round-robin fashion, until a link is established.
5. The **RDMA Protocol Support** setting applies to all partitions of the port, if in NPar mode. This setting appears if the **NIC + RDMA Mode** in [Step 3](#) is set to **Enabled**. **RDMA Protocol Support** options include the following:
 - RoCE** enables RoCE mode on this port.
 - iWARP** enables iWARP mode on this port.
 - iWARP + RoCE** enables iWARP and RoCE modes on this port. This is the default. Additional configuration for Linux is required for this option as described in [“Configuring iWARP and RoCE” on page 191](#).

6. For **Boot Mode**, select one of the following values:
 - PXE** enables PXE boot.
 - FCoE** enables FCoE boot from SAN over the hardware offload pathway. The **FCoE** mode is available only if **FCoE Offload** is enabled on the second partition in NPar mode (see [“Configuring Partitions” on page 73](#)).
 - iSCSI** enables iSCSI remote boot over the hardware offload pathway. The **iSCSI** mode is available only if **iSCSI Offload** is enabled on the third partition in NPar mode (see [“Configuring Partitions” on page 73](#)).
 - Disabled** prevents this port from being used as a remote boot source.
7. The **Energy Efficient Ethernet (EEE)** parameter is visible only on 10GBASE-T or 10GBASE-T RJ45 interfaced adapters. Select from the following EEE options:
 - Disabled** disables EEE on this port.
 - Optimal Power and Performance** enables EEE in optimal power and performance mode on this port.
 - Maximum Power Savings** enables EEE in maximum power savings mode on this port.
 - Maximum Performance** enables EEE in maximum performance mode on this port.
8. The **Virtual LAN Mode** parameter applies to the entire port when in PXE remote install mode. It is not persistent after a PXE remote install finishes. Select from the following vLAN options:
 - Enabled** enables vLAN mode on this port for PXE remote install mode.
 - Disabled** disables vLAN mode on this port.
9. The **Virtual LAN ID** parameter specifies the vLAN tag ID to be used on this port for PXE remote install mode. This setting applies only when **Virtual LAN Mode** is enabled in the previous step.
10. Click **Back**.
11. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

To configure the port to use RDMA:

NOTE

Follow these steps to enable RDMA on all partitions of an NPar mode port.

1. Set **NIC + RDMA Mode** to **Enabled**.
2. Click **Back**.
3. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

To configure the port's boot mode:

1. For a UEFI PXE remote installation, select **PXE** as the **Boot Mode**.
2. Click **Back**.
3. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

To configure the port's PXE remote install to use a vLAN:

NOTE

This vLAN is not persistent after the PXE remote install is finished.

1. Set the **Virtual LAN Mode** to **Enabled**.
2. In the **Virtual LAN ID** box, enter the number to be used.
3. Click **Back**.
4. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

Configuring Data Center Bridging

The data center bridging (DCB) settings comprise the DCBX protocol and the RoCE priority.

To configure the DCB settings:

1. On the Main Configuration Page ([Figure 5-3 on page 54](#)), select **Data Center Bridging (DCB) Settings**, and then click **Finish**.
2. On the Data Center Bridging (DCB) Settings page ([Figure 5-8](#)), select the appropriate **DCBX Protocol** option:
 - Disabled** disables DCBX on this port.

- CEE** enables the legacy Converged Enhanced Ethernet (CEE) protocol DCBX mode on this port.
 - IEEE** enables the IEEE DCBX protocol on this port.
 - Dynamic** enables dynamic application of either the CEE or IEEE protocol to match the attached link partner.
3. On the Data Center Bridging (DCB) Settings page, enter the **RoCE v1 Priority** as a value from **0–7**. This setting indicates the DCB traffic class priority number used for RoCE traffic and should match the number used by the DCB-enabled switching network for RoCE traffic. Typically, 0 is used for the default lossy traffic class, 3 is used for the FCoE traffic class, and 4 is used for the lossless iSCSI-TLV over DCB traffic class.

Main Configuration Page • Data Center Bridging (DCB) Settings	
DCBX Protocol	Disabled
RoCE v1 Priority	0

Figure 5-8. System Setup: Data Center Bridging (DCB) Settings

4. Click **Back**.
5. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

NOTE

When DCBX is enabled, the adapter periodically sends link layer discovery protocol (LLDP) packets with a dedicated unicast address that serves as the source MAC address. This LLDP MAC address is different from the factory-assigned adapter Ethernet MAC address. If you examine the MAC address table for the switch port that is connected to the adapter, you will see two MAC addresses: one for LLDP packets and one for the adapter Ethernet interface.

Configuring FCoE Boot

NOTE

The FCoE Boot Configuration Menu is only visible if **FCoE Offload Mode** is enabled on the second partition in NPar mode (see [Figure 5-18 on page 76](#)). It is not visible in non-NPar mode.

On QL41164Hxxx quad-port adapters, FCoE and iSCSI storage offloads are only enabled on partition 2.

To enable FCoE-Offload mode, see the *Application Note, Enabling Storage Offloads on Dell and Marvell FastLinQ 41000 Series Adapters* at <https://www.marvell.com/documents/5aa5otcbkr0im3ynera3/>.

To configure the FCoE boot configuration parameters:

1. On the Main Configuration Page, select **FCoE Boot Configuration Menu**, and then select the following as needed:
 - FCoE General Parameters** ([Figure 5-9](#))
 - FCoE Target Configuration** ([Figure 5-10](#))
2. Press ENTER.
3. Choose values for the FCoE General or FCoE Target Configuration parameters.

Main Configuration Page • FCoE Configuration • FCoE General Parameters	
Fabric Discovery Retry Count	<input type="text" value="5"/>
LUN Busy Retry Count	<input type="text" value="5"/>

Figure 5-9. FCoE General Parameters

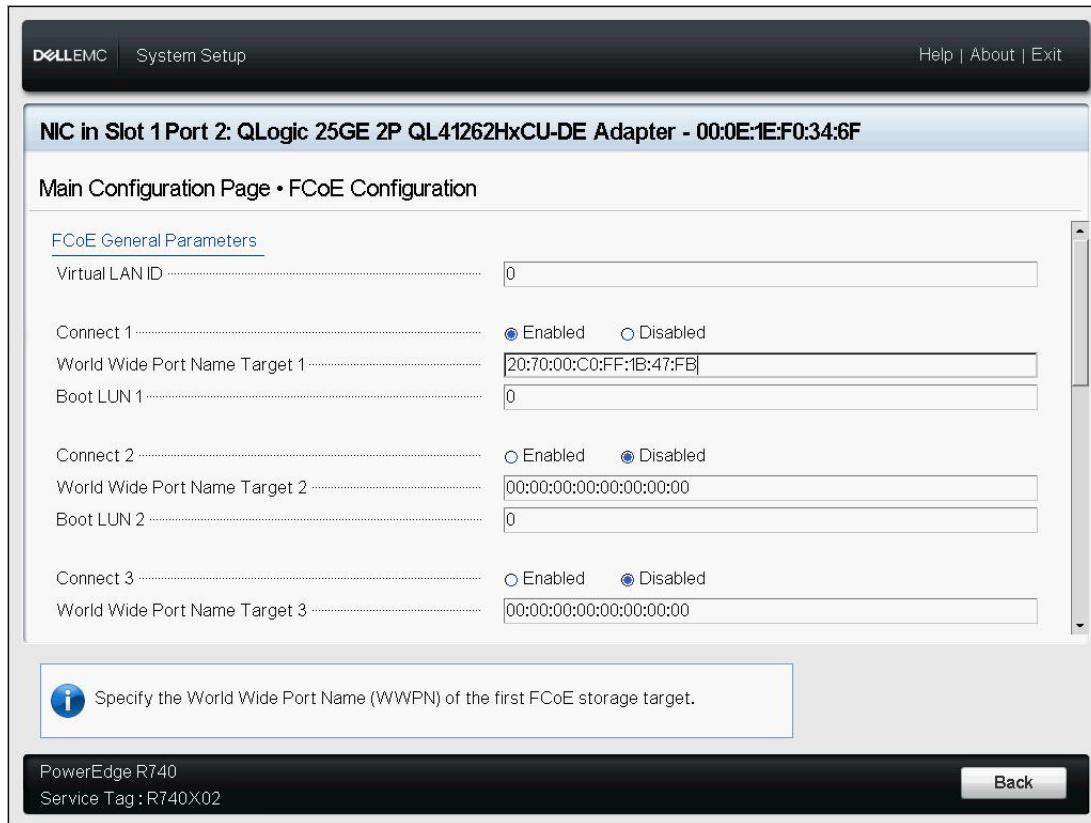


Figure 5-10. FCoE Target Configuration

4. Click **Back**.
5. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

Configuring iSCSI Boot

NOTE

The iSCSI Boot Configuration Menu is only visible if **iSCSI Offload Mode** is enabled on the third partition in NPar mode (see [Figure 5-19 on page 77](#)). It is not visible in non-NPar mode.

On QL41164Hxxx quad-port adapters, FCoE and iSCSI storage offloads are only enabled on partition 2.

To enable FCoE-Offload mode, see the *Application Note, Enabling Storage Offloads on Dell and Marvell FastLinQ 41000 Series Adapters* at <https://www.marvell.com/documents/5aa5otcbkr0im3ynera3/>.

To configure the iSCSI boot configuration parameters:

1. On the Main Configuration Page, select **iSCSI Boot Configuration Menu**, and then select one of the following options:
 - iSCSI General Configuration**
 - iSCSI Initiator Configuration**
 - iSCSI First Target Configuration**
 - iSCSI Second Target Configuration**
2. Press ENTER.
3. Choose values for the appropriate iSCSI configuration parameters:
 - iSCSI General Parameters** ([Figure 5-11 on page 70](#))
 - TCP/IP Parameters Via DHCP
 - iSCSI Parameters Via DHCP
 - CHAP Authentication
 - CHAP Mutual Authentication
 - IP Version
 - ARP Redirect
 - DHCP Request Timeout
 - Target Login Timeout
 - DHCP Vendor ID
 - iSCSI Initiator Parameters** ([Figure 5-12 on page 71](#))
 - IPv4 Address
 - IPv4 Subnet Mask
 - IPv4 Default Gateway
 - IPv4 Primary DNS
 - IPv4 Secondary DNS
 - VLAN ID
 - iSCSI Name
 - CHAP ID
 - CHAP Secret
 - iSCSI First Target Parameters** ([Figure 5-13 on page 71](#))
 - Connect
 - IPv4 Address
 - TCP Port
 - Boot LUN
 - iSCSI Name
 - CHAP ID
 - CHAP Secret

❑ **iSCSI Second Target Parameters** (Figure 5-14 on page 72)

- Connect
- IPv4 Address
- TCP Port
- Boot LUN
- iSCSI Name
- CHAP ID
- CHAP Secret

4. Click **Back**.
5. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

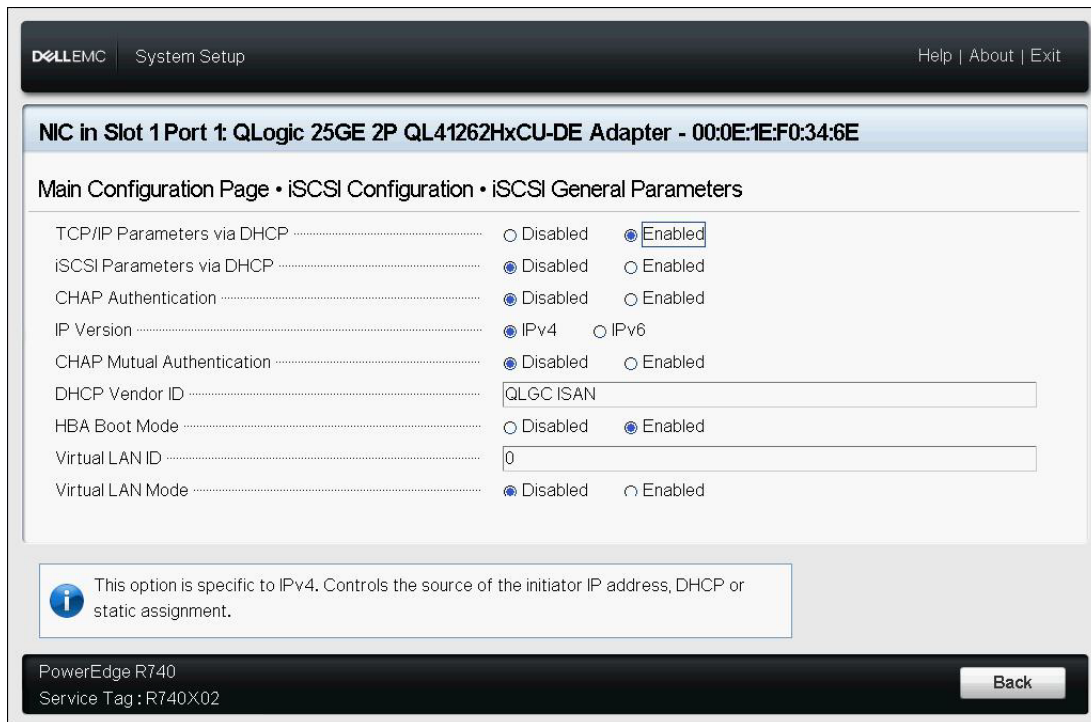


Figure 5-11. iSCSI General Parameters

The screenshot shows the 'iSCSI Initiator Parameters' configuration page. At the top, the header includes 'Dell EMC System Setup' and 'Help | About | Exit'. Below the header, the NIC information is displayed: 'NIC in Slot 1 Port 1: QLogic 25GE 2P QL41262HxCU-DE Adapter - 00:0E:1E:F0:34:6E'. The main configuration area is titled 'Main Configuration Page • iSCSI Configuration • iSCSI Initiator Parameters'. It contains several input fields: IPv4 Address (192.168.100.145), Subnet Mask (255.255.255.0), IPv4 Default Gateway (0.0.0.0), IPv4 Primary DNS (0.0.0.0), IPv4 Secondary DNS (0.0.0.0), iSCSI Name (iqn.1994-02.com.qlogic.iscsi:fastlinqboot), CHAP ID, and CHAP Secret. A blue information icon with a text box below it reads: 'Specify the iSCSI Qualified Name (IQN) of the initiator.' The footer shows 'PowerEdge R740' and 'Service Tag : R740X02' on the left, and a 'Back' button on the right.

Figure 5-12. iSCSI Initiator Configuration Parameters

The screenshot shows the 'iSCSI First Target Parameters' configuration page. At the top, the header includes 'Dell EMC System Setup' and 'Help | About | Exit'. Below the header, the NIC information is displayed: 'NIC in Slot 1 Port 1: QLogic 25GE 2P QL41262HxCU-DE Adapter - 00:0E:1E:F0:34:6E'. The main configuration area is titled 'Main Configuration Page • iSCSI Configuration • iSCSI First Target Parameters'. It contains several input fields: a 'Connect' checkbox (radio buttons for Disabled and Enabled, with 'Enabled' selected), IPv4 Address (192.168.100.9), TCP Port (3260), Boot LUN (1), iSCSI Name (iqn.2002-03.com.compellent:5000d31000ee1246), CHAP ID, and CHAP Secret. A blue information icon with a text box below it reads: 'Specify the IPV4 address of the first iSCSI target.' The footer shows 'PowerEdge R740' and 'Service Tag : R740X02' on the left, and a 'Back' button on the right.

Figure 5-13. iSCSI First Target Parameters

The screenshot shows the 'iSCSI Second Target Parameters' configuration page within the Dell EMC System Setup utility. The interface includes a top navigation bar with the Dell EMC logo, 'System Setup', and links for 'Help | About | Exit'. Below this, a header identifies the network interface as 'NIC in Slot 1 Port 1: QLogic 25GE 2P QL41262HxCU-DE Adapter - 00:0E:1E:F0:34:6E'. The main configuration area is titled 'Main Configuration Page • iSCSI Configuration • iSCSI Second Target Parameters' and contains the following fields:

- Connect:** Radio buttons for 'Disabled' (selected) and 'Enabled'.
- IPv4 Address:** Text input field containing '0.0.0.0'.
- TCP Port:** Text input field containing '3260'.
- Boot LUN:** Text input field containing '2'.
- iSCSI Name:** Empty text input field.
- CHAP ID:** Empty text input field.
- CHAP Secret:** Empty text input field.

An information box below the fields states: 'Specify the iSCSI Qualified Name (IQN) of the second iSCSI storage target.' The bottom of the window shows system information: 'PowerEdge R740' and 'Service Tag : R740X02', along with a 'Back' button.

Figure 5-14. iSCSI Second Target Parameters

Configuring Partitions

You can configure bandwidth ranges for each partition on the adapter.

To configure the maximum and minimum bandwidth allocations:

1. On the Main Configuration Page, select **NIC Partitioning Configuration**, and then press ENTER.
2. On the Partitions Configuration page ([Figure 5-15](#)), select **Global Bandwidth Allocation**.

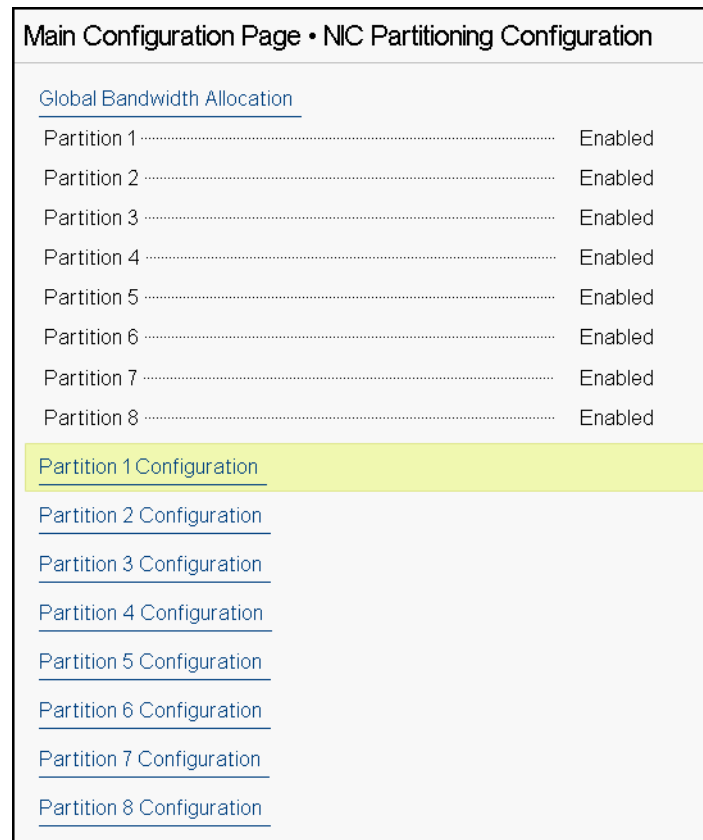


Figure 5-15. NIC Partitioning Configuration, Global Bandwidth Allocation

3. On the Global Bandwidth Allocation page (Figure 5-16), click each partition minimum and maximum TX bandwidth field for which you want to allocate bandwidth. There are eight partitions per port in dual-port mode.

The screenshot shows a configuration page titled "Main Configuration Page • NIC Partitioning Configuration • Global Bandwidth Allocation". It contains two sections of input fields. The first section lists "Partition 1 Minimum TX Bandwidth" through "Partition 8 Minimum TX Bandwidth", each with a text input field containing the value "0". The second section lists "Partition 1 Maximum TX Bandwidth", "Partition 2 Maximum TX Bandwidth", and "Partition 3 Maximum TX Bandwidth", each with a text input field containing the value "100". At the bottom, there is an information icon and a text box stating: "Minimum Bandwidth represents the minimum transmit bandwidth of the partition as percentage of the full physical port link speed. The Minimum ... (Press <F1> for more help)".

Figure 5-16. Global Bandwidth Allocation Page

- ❑ **Partition *n* Minimum TX Bandwidth** is the minimum transmit bandwidth of the selected partition expressed as a percentage of the maximum physical port link speed. Values can be 0–100. When DCBX ETS mode is enabled, the per-traffic class DCBX ETS minimum bandwidth value is used simultaneously with the per-partition minimum TX bandwidth value. The total of the minimum TX bandwidth values of all partitions on a single port must equal 100 or be all zeros.

Setting the TX minimum bandwidth to all zeros is similar to equally dividing the available bandwidth over every active partition; however, the bandwidth is dynamically allocated over all actively sending partitions. A zero value (when one or more of the other values are set to a non-zero value) allocates a minimum of one percent to that partition, when congestion (from all of the partitions) is restricting TX bandwidth.

- ❑ **Partition *n* Maximum TX Bandwidth** is the maximum transmit bandwidth of the selected partition expressed as a percentage of the maximum physical port link speed. Values can be 1–100. The per-partition maximum TX bandwidth value applies regardless of the DCBX ETS mode setting.

Type a value in each selected field, and then click **Back**.

4. When prompted, click **Yes** to save the changes. Changes take effect after a system reset.

To configure partitions:

1. To examine a specific partition configuration, on the NIC Partitioning Configuration page (Figure 5-15 on page 73), select **Partition *n* Configuration**. If NParEP is not enabled, only four partitions exist per port.
2. To configure the first partition, select **Partition 1 Configuration** to open the Partition 1 Configuration page (Figure 5-17), which shows the following parameters:
 - NIC Mode** (always enabled)
 - PCI Device ID**
 - PCI (bus) Address**
 - MAC Address**
 - Virtual MAC Address**

If NParEP is not enabled, only four partitions per port are available. On non-offload-capable adapters, the **FCoE Mode** and **iSCSI Mode** options and information are not displayed.

Main Configuration Page • NIC Partitioning Configuration • Partition 1 Configuration	
NIC Mode	Enabled
PCI Device ID	8070
PCI Address	86:00
MAC Address	00:0E:1E:D5:F8:76
Virtual MAC Address	00:00:00:00:00:00

Figure 5-17. Partition 1 Configuration

3. To configure the second partition, select **Partition 2 Configuration** to open the Partition 2 Configuration page. If FCoE Offload is present, the Partition 2 Configuration (Figure 5-18) shows the following parameters:
 - NIC Mode** enables or disables the L2 Ethernet NIC personality on Partitions 2 and greater. To disable any of the remaining partitions, set the **NIC Mode** to **Disabled**. To disable offload-capable partitions, disable both the **NIC Mode** and respective offload mode.
 - FCoE Mode** enables or disables the FCoE-Offload personality on the second partition. If you enable this mode on the second partition, you should disable **NIC Mode**. Because only one offload is available per port, if FCoE-Offload is enabled on the port's second partition, iSCSI-Offload cannot be enabled on the third partition of that same NPar mode port. Not all adapters support **FCoE Mode**.

- iSCSI Mode** enables or disables the iSCSI-Offload personality on the third partition. If you enable this mode on the third partition, you should disable **NIC Mode**. Because only one offload is available per port, if iSCSI-Offload is enabled on the port's third partition, FCoE-Offload cannot be enabled on the second partition of that same NPar mode port. Not all adapters support **iSCSI Mode**.
- FIP MAC Address**¹
- Virtual FIP MAC Address**¹
- World Wide Port Name**¹
- Virtual World Wide Port Name**¹
- World Wide Node Name**¹
- Virtual World Wide Node Name**¹
- PCI Device ID**
- PCI (bus) Address**

Main Configuration Page • NIC Partitioning Configuration • Partition 2 Configuration		
NIC Mode	<input type="radio"/> Enabled	<input checked="" type="radio"/> Disabled
FCoE Mode	<input checked="" type="radio"/> Enabled	<input type="radio"/> Disabled
FIP MAC Address	00:0E:1E:D5:F8:78	
Virtual FIP MAC Address	00:00:00:00:00:00	
World Wide Port Name	20:01:00:0E:1E:D5:F8:78	
Virtual World Wide Port Name	00:00:00:00:00:00:00:00	
World Wide Node Name	20:00:00:0E:1E:D5:F8:78	
Virtual World Wide Node Name	00:00:00:00:00:00:00:00	
PCI Device ID	8070	
PCI Address	86:02	

Figure 5-18. Partition 2 Configuration: FCoE Offload

4. To configure the third partition, select **Partition 3 Configuration** to open the Partition 3 Configuration page (Figure 5-19). If iSCSI Offload is present, the Partition 3 Configuration shows the **Personality** as **iSCSI** (Figure 5-19) and the following additional parameters:
 - NIC Mode (Disabled)**
 - iSCSI Offload Mode (Enabled)**
 - iSCSI Offload MAC Address**²
 - Virtual iSCSI Offload MAC Address**²

¹ This parameter is only present on the second partition of an NPar mode port of FCoE offload-capable adapters.

² This parameter is only present on the third partition of an NPar mode port of iSCSI offload-capable adapters.

- PCI Device ID**
- PCI Address**

Main Configuration Page • NIC Partitioning Configuration • Partition 3 Configuration	
NIC Mode	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled
iSCSI Offload Mode	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled
iSCSI Offload MAC Address	00:0E:1E:D5:F8:7A
Virtual iSCSI Offload MAC Address	00:00:00:00:00:00
PCI Device ID	8070
PCI Address	86:04

Figure 5-19. Partition 3 Configuration: iSCSI Offload

5. To configure the remaining Ethernet partitions, including the previous (if not offload-enabled), open the page for a partition 2 or greater partition (see [Figure 5-20](#)).
 - NIC Mode (Enabled or Disabled)**. When disabled, the partition is hidden such that it does not appear to the OS if fewer than the maximum quantity of partitions (or PCI PFs) are detected.
 - PCI Device ID**
 - PCI Address**
 - MAC Address**
 - Virtual MAC Address**

Main Configuration Page • NIC Partitioning Configuration • Partition 4 Configuration	
NIC Mode	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled
PCI Device ID	8070
PCI Address	86:06
MAC Address	00:0E:1E:D5:F8:7C
Virtual MAC Address	00:00:00:00:00:00

Figure 5-20. Partition 4 Configuration

Partitioning for VMware ESXi 6.7 and ESXi 7.0

If the following conditions exist on a system running either VMware ESXi 6.7 or ESXi 7.0, you must uninstall and reinstall the drivers:

- The adapter is configured to enable NPar with all NIC partitions.
- The adapter is in Single Function mode.
- The configuration is saved and the system is rebooted.

- Storage partitions are enabled (by converting one of the NIC partitions as storage) while drivers are already installed on the system.
- Partition 2 is changed to FCoE.
- The configuration is saved and the system is rebooted again.

Driver re-installation is required because the storage functions may keep the `vmnicX` enumeration rather than `vmhbaX`, as shown when you issue the following command on the system:

```
# esxcfg-scsidevs -a
vmnic4 qedf          link-up   fc.2000000e1ed6fa2a:2001000e1ed6fa2a
(0000:19:00.2) QLogic Corp. QLogic FastLinQ QL41xxx Series 10/25 GbE
Controller (FCoE)
vmhba0 lsi_mr3        link-n/a  sas.51866da071fa9100
(0000:18:00.0) Avago (LSI) PERC H330 Mini
vmnic10 qedf         link-up   fc.2000000e1ef249f8:2001000e1ef249f8
(0000:d8:00.2) QLogic Corp. QLogic FastLinQ QL41xxx Series 10/25 GbE
Controller (FCoE)
vmhba1 vmw_ahci       link-n/a  sata.vmhba1
(0000:00:11.5) Intel Corporation Lewisburg SSATA Controller [AHCI mode]
vmhba2 vmw_ahci       link-n/a  sata.vmhba2
(0000:00:17.0) Intel Corporation Lewisburg SATA Controller [AHCI mode]
vmhba32 qedil         online    iscsi.vmhba32          QLogic
FastLinQ QL41xxx Series 10/25 GbE Controller (iSCSI)
vmhba33 qedil         online    iscsi.vmhba33          QLogic
FastLinQ QL41xxx Series 10/25 GbE Controller (iSCSI)
```

In the preceding command output, notice that `vmnic4` and `vmnic10` are actually storage adapter ports. To prevent this behavior, you should enable storage functions at the same time that you configure the adapter for NPar mode.

For example, assuming that the adapter is in Single Function mode by default, you should:

1. Enable NPar mode.
2. Change Partition 2 to FCoE.
3. Save and reboot.

6 Boot from SAN Configuration

SAN boot enables deployment of diskless servers in an environment where the boot disk is located on storage connected to the SAN. The server (initiator) communicates with the storage device (target) through the SAN using the Marvell Converged Network Adapter (CNA) Host Bus Adapter (HBA).

To enable FCoE-Offload mode, see the *Application Note, Enabling Storage Offloads on Dell and Marvell FastLinQ 41000 Series Adapters* at <https://www.marvell.com/documents/5aa5otcbkr0im3ynera3/>.

This chapter covers boot from SAN configuration for both iSCSI and FCoE:

- [iSCSI Boot from SAN](#)
- [“FCoE Boot from SAN” on page 116](#)

iSCSI Boot from SAN

Marvell 41000 Series gigabit Ethernet (GbE) adapters support iSCSI boot to enable network boot of operating systems to diskless systems. iSCSI boot allows a Windows, Linux, or VMware operating system to boot from an iSCSI target machine located remotely over a standard IP network.

This section provides the following configuration information about iSCSI boot from SAN:

- [iSCSI Out-of-Box and Inbox Support](#)
- [iSCSI Preboot Configuration](#)
- [Configuring iSCSI Boot from SAN on Windows](#)
- [Configuring iSCSI Boot from SAN on Linux](#)
- [Configuring iSCSI Boot from SAN on VMware](#)

iSCSI Out-of-Box and Inbox Support

Table 6-1 lists the operating systems' inbox and out-of-box support for iSCSI boot from SAN (BFS).

Table 6-1. iSCSI Out-of-Box and Inbox Boot from SAN Support

OS Version	Out-of-Box		Inbox	
	SW iSCSI BFS Support	Hardware Offload iSCSI BFS Support	SW iSCSI BFS Support	Hardware Offload iSCSI BFS Support
Windows 2016 ^a	Yes	Yes	Yes	No
Windows 2019	Yes	Yes	Yes	Yes
Azure Stack HCI	Yes	Yes	Yes	Yes
RHEL 7.8	Yes	Yes	Yes	Yes
RHEL 7.9	Yes	Yes	Yes	Yes
RHEL 8.2	Yes	Yes	Yes	Yes
RHEL 8.3	Yes	Yes	Yes	Yes
SLES 15 SP1, SP2	Yes	Yes	Yes	Yes
VMware ESXi 6.7 U2 ^b	Yes	No	Yes	No
VMware ESXi 7.0	Yes	No	Yes	No

^a Windows Server 2016 does not support the inbox iSCSI driver for hardware offload.

^b ESXi out-of-box and inbox do not support native hardware offload iSCSI boot. The system will perform a SW boot and connection and then will transition to hardware offload.

iSCSI Preboot Configuration

For both Windows and Linux operating systems, configure iSCSI boot with **UEFI iSCSI HBA** (offload path with the Marvell offload iSCSI driver). Set this option using Boot Protocol, under **Port Level Configuration**. To support iSCSI boot, first enable the iSCSI HBA in the UEFI HII and then set the boot protocol accordingly.

For both Windows and Linux operating systems, iSCSI boot can be configured to boot with two distinctive methods:

- **iSCSI SW** (also known as non-offload path with Microsoft/Open-iSCSI initiator)

Follow the Dell BIOS guide for iSCSI software installation.

- **iSCSI HW** (offload path with the Marvell FastLinQ offload iSCSI driver). This option can be set using **Boot Mode**.

iSCSI hardware installation instructions start in [“Enabling NPar and the iSCSI HBA” on page 83](#).

For VMware ESXi operating systems, only the iSCSI SW method is supported.

iSCSI preboot information in this section includes:

- [Setting the BIOS Boot Mode to UEFI](#)
- [Enabling NPar and the iSCSI HBA](#)
- [Selecting the iSCSI UEFI Boot Protocol](#)
- [Configuring the Storage Target](#)
- [Configuring iSCSI Boot Options](#)
- [Configuring the DHCP Server to Support iSCSI Boot](#)

Setting the BIOS Boot Mode to UEFI

To configure the boot mode:

1. Restart the system.
2. Access the System BIOS menu.
3. For the **Boot Mode** setting, select **UEFI** (see [Figure 6-1](#)).

NOTE

SAN boot is supported in UEFI environment only. Make sure the system boot option is UEFI, and not legacy.

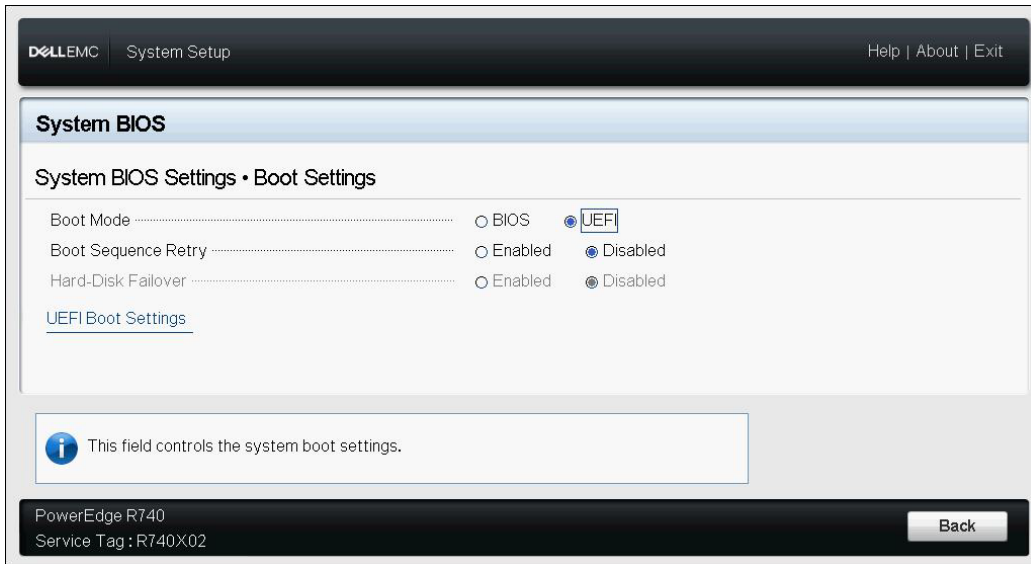


Figure 6-1. System Setup: Boot Settings

Enabling NPar and the iSCSI HBA

To enable NPar and the iSCSI HBA:

1. In the System Setup, Device Settings, select the QLogic device (Figure 6-2). Refer to the OEM user guide on accessing the PCI device configuration menu.

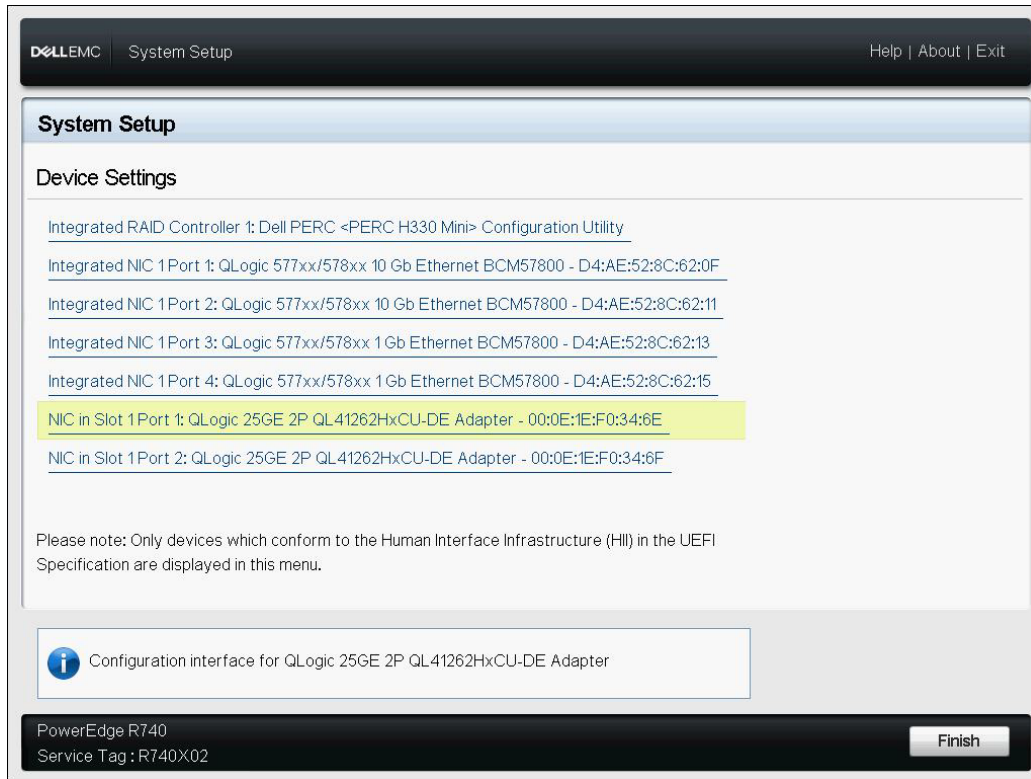


Figure 6-2. System Setup: Device Settings

2. Enable NPar.

Configuring the Storage Target

Configuring the storage target varies by target vendors. For information on configuring the storage target, refer to the documentation provided by the vendor.

To configure the storage target:

1. Select the appropriate procedure based on your storage target, either:
 - Create a storage target using software such as SANBlaze® or Linux-IO (LIO™) Target.
 - Create a vdisk or volume using a target array such as EqualLogic® or EMC®.
2. Create a virtual disk.

Selecting the iSCSI UEFI Boot Protocol

Before selecting the preferred boot mode, ensure that the **Device Level Configuration** menu setting is **Enable NPAR** and that the **NIC Partitioning Configuration** menu setting is **Enable iSCSI HBA**.

The **Boot Mode** option is listed under **NIC Configuration** (Figure 6-3) for the adapter, and the setting is port specific. Refer to the OEM user manual for direction on accessing the device-level configuration menu under UEFI HII.

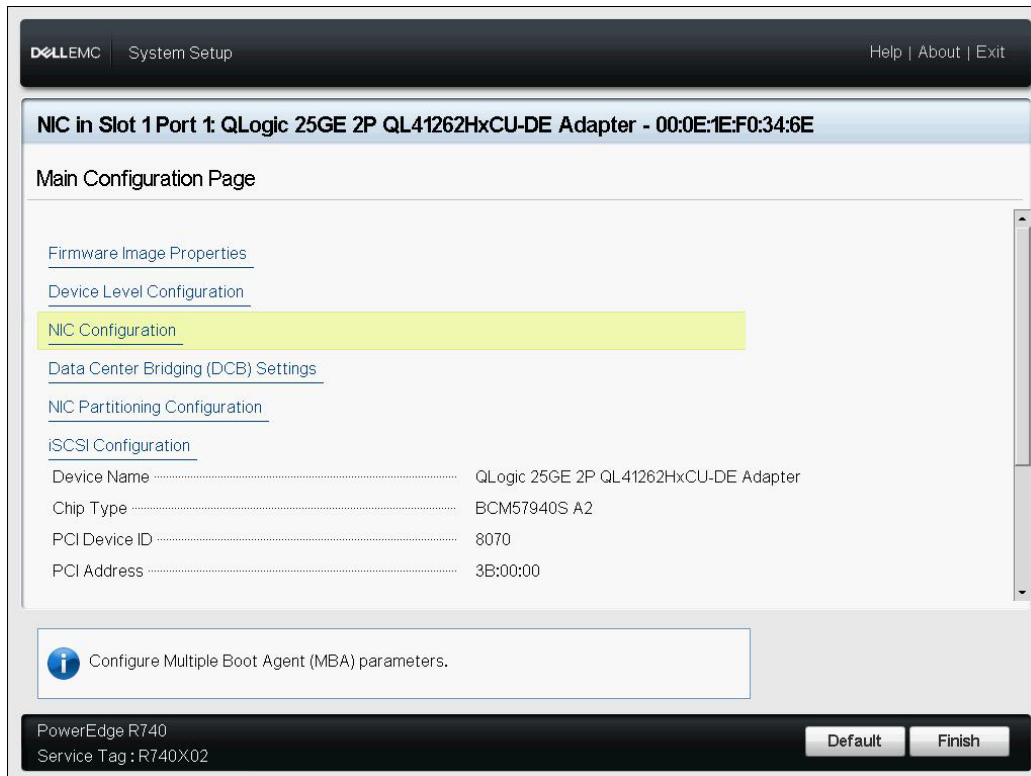


Figure 6-3. System Setup: NIC Configuration

NOTE

Boot from SAN is supported only in NPar mode and is configured in UEFI, and not in legacy BIOS.

1. On the NIC Configuration page (Figure 6-4), for the **Boot Protocol** option, select **UEFI iSCSI HBA** (requires NPar mode).

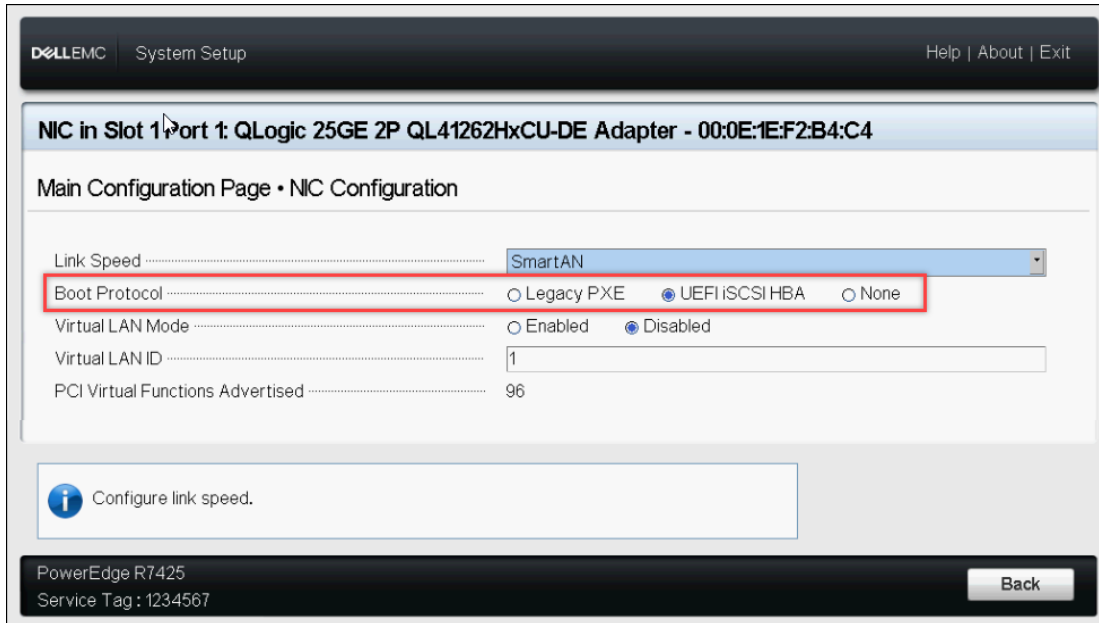


Figure 6-4. System Setup: NIC Configuration, Boot Protocol

NOTE

Use the **Virtual LAN Mode** and **Virtual LAN ID** options on this page only for PXE boot. If a vLAN is needed for UEFI iSCSI HBA boot mode, see [Step 3](#) of [Static iSCSI Boot Configuration](#).

Configuring iSCSI Boot Options

iSCSI boot configuration options include:

- [Static iSCSI Boot Configuration](#)
- [Dynamic iSCSI Boot Configuration](#)
- [Enabling CHAP Authentication](#)

Static iSCSI Boot Configuration

In a static configuration, you must enter data for the following:

- Initiator IP address
- Initiator IQN
- Target parameters (obtained in [“Configuring the Storage Target”](#) on page 83)

For information on configuration options, see [Table 6-2](#) on page 88.

To configure the iSCSI boot parameters using static configuration:

1. In the Device HII **Main Configuration Page**, select **iSCSI Configuration** (Figure 6-5), and then press ENTER.

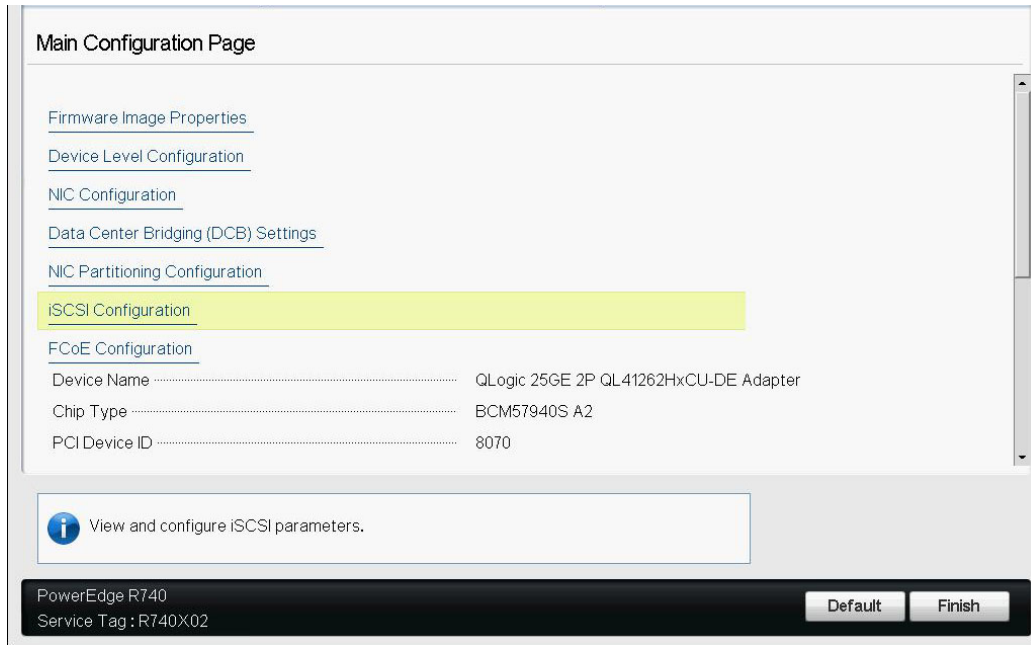


Figure 6-5. System Setup: iSCSI Configuration

2. On the **iSCSI Configuration** page, select **iSCSI General Parameters** (Figure 6-6), and then press ENTER.

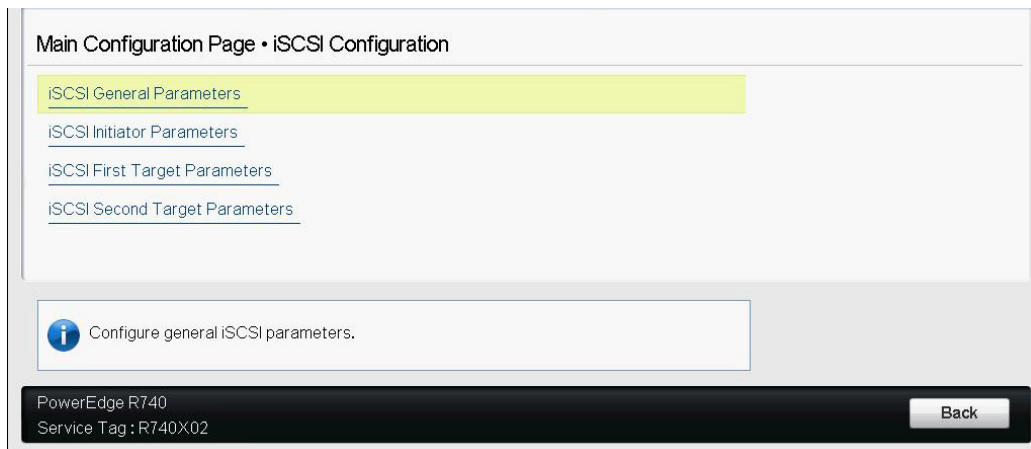


Figure 6-6. System Setup: Selecting General Parameters

3. On the iSCSI General Parameters page (Figure 6-7), press the DOWN ARROW key to select a parameter, and then press the ENTER key to input the following values (Table 6-2 on page 88 provides descriptions of these parameters):
 - TCP/IP Parameters via DHCP: Disabled**
 - iSCSI Parameters via DHCP: Disabled**
 - CHAP Authentication: As required**
 - IP Version: As required (IPv4 or IPv6)**
 - CHAP Mutual Authentication: As required**
 - DHCP Vendor ID: Not applicable for static configuration**
 - HBA Boot Mode: As required**
 - Virtual LAN ID: Default value or as required**
 - Virtual LAN Mode: As required**

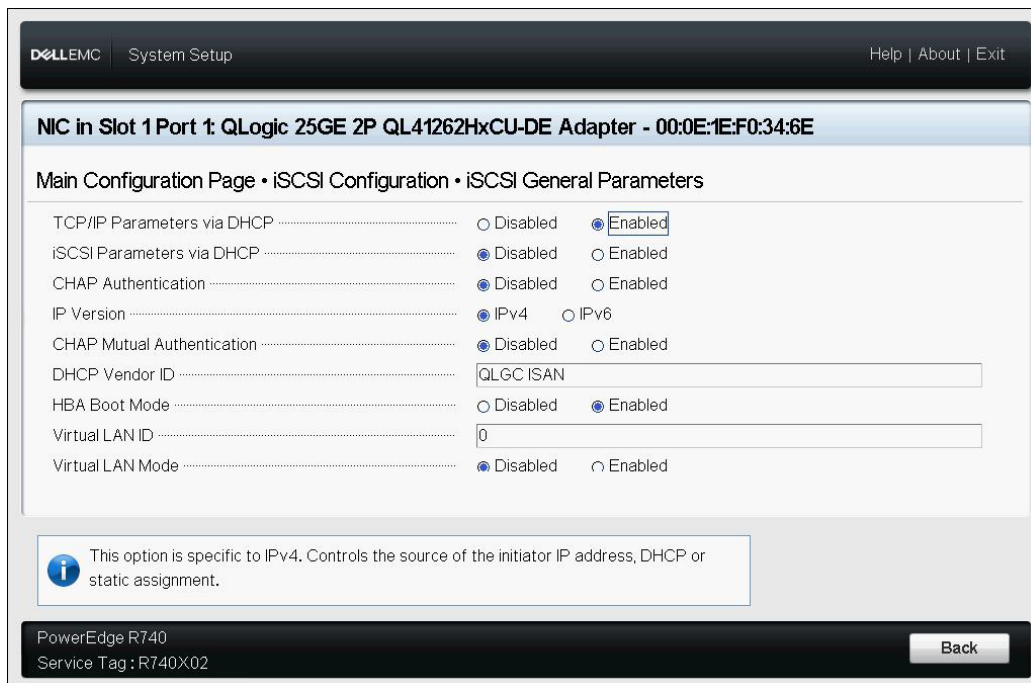


Figure 6-7. System Setup: iSCSI General Parameters

Table 6-2. iSCSI General Parameters

Option	Description
TCP/IP Parameters via DHCP	This option is specific to IPv4. Controls whether the iSCSI boot host software acquires the IP address information using DHCP (Enabled) or using a static IP configuration (Disabled).
iSCSI Parameters via DHCP	Controls whether the iSCSI boot host software acquires its iSCSI target parameters using DHCP (Enabled) or through a static configuration (Disabled). The static information is entered on the iSCSI Initiator Parameters Configuration page.
CHAP Authentication	Controls whether the iSCSI boot host software uses CHAP authentication when connecting to the iSCSI target. If CHAP Authentication is enabled, configure the CHAP ID and CHAP Secret on the iSCSI Initiator Parameters Configuration page.
IP Version	This option is specific to IPv6. Toggles between IPv4 and IPv6. All IP settings are lost if you switch from one protocol version to another.
CHAP Mutual Authentication	Controls whether the iSCSI boot host software acquires its iSCSI target parameters using DHCP (Enabled) or through a static configuration (Disabled). The static information is entered on the iSCSI Initiator Parameters Configuration page.
DHCP Vendor ID	Controls how the iSCSI boot host software interprets the Vendor Class ID field used during DHCP. If the Vendor Class ID field in the DHCP offer packet matches the value in the field, the iSCSI boot host software looks into the DHCP Option 43 fields for the required iSCSI boot extensions. If DHCP is disabled, this value does not need to be set.
HBA Boot Mode	Controls whether SW or Offload is enabled or disabled. For Offload, this option is unavailable (grayed out). For information about SW (non-offload), refer to the Dell BIOS configuration.
Virtual LAN ID	vLAN ID range is 1–4094.
Virtual LAN Mode	Enables or disables vLAN.

4. Return to the iSCSI Configuration page, and then press the ESC key.

5. Select **iSCSI Initiator Parameters** (Figure 6-8), and then press ENTER.

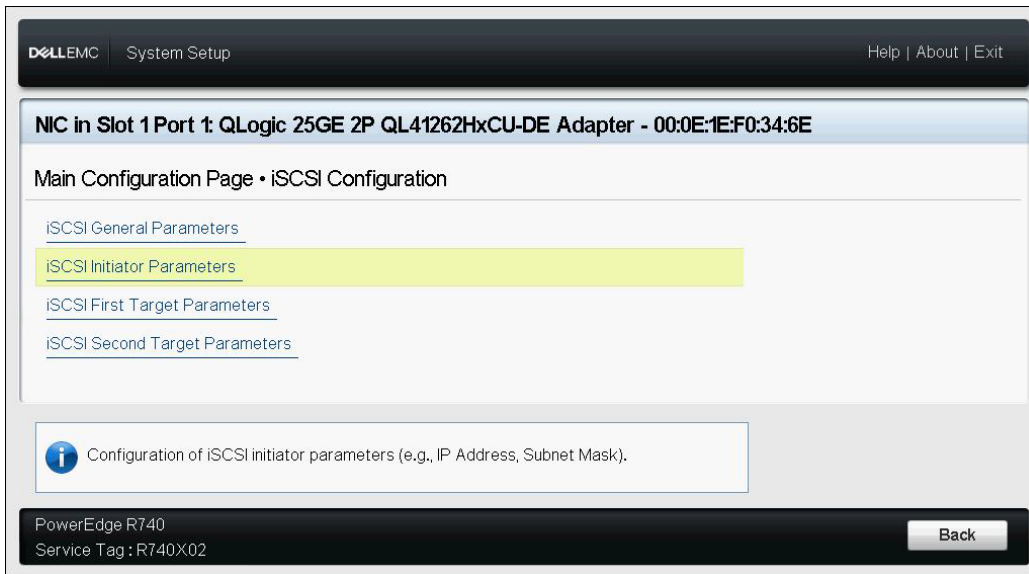


Figure 6-8. System Setup: Selecting iSCSI Initiator Parameters

6. On the iSCSI Initiator Parameters page (Figure 6-9), select the following parameters, and then type a value for each:
 - IPv4* Address**
 - Subnet Mask**
 - IPv4* Default Gateway**
 - IPv4* Primary DNS**
 - IPv4* Secondary DNS**
 - iSCSI Name.** Corresponds to the iSCSI initiator name to be used by the client system.
 - CHAP ID**
 - CHAP Secret**

NOTE

For the preceding items with asterisks (*), note the following:

- The label will change to **IPv6** or **IPv4** (default) based on the IP version set on the iSCSI General Parameters page (Figure 6-7 on page 87).
- Carefully enter the IP address. There is no error-checking performed against the IP address to check for duplicates, incorrect segment, or network assignment.

The screenshot shows the 'System Setup' interface for a Dell EMC PowerEdge R740. The title bar includes 'DELL EMC System Setup' and 'Help | About | Exit'. The main heading is 'NIC in Slot 1 Port 1: QLogic 25GE 2P QL41262HxCU-DE Adapter - 00:0E:1E:F0:34:6E'. Below this, the breadcrumb path is 'Main Configuration Page • iSCSI Configuration • iSCSI Initiator Parameters'. The configuration fields are as follows:

IPv4 Address	0.0.0.0
Subnet Mask	0.0.0.0
IPv4 Default Gateway	0.0.0.0
IPv4 Primary DNS	0.0.0.0
IPv4 Secondary DNS	0.0.0.0
iSCSI Name	iqn.1994-02.com.qlogic.iscsi:fastlinqboot
CHAP ID	
CHAP Secret	

Below the fields is an information icon and the text: 'Specify the iSCSI Qualified Name (IQN) of the initiator.' At the bottom left, it says 'PowerEdge R740 Service Tag : R740X02'. At the bottom right, there is a 'Back' button.

Figure 6-9. System Setup: iSCSI Initiator Parameters

7. Return to the iSCSI Configuration page, and then press ESC.

8. Select **iSCSI First Target Parameters** (Figure 6-10), and then press ENTER.

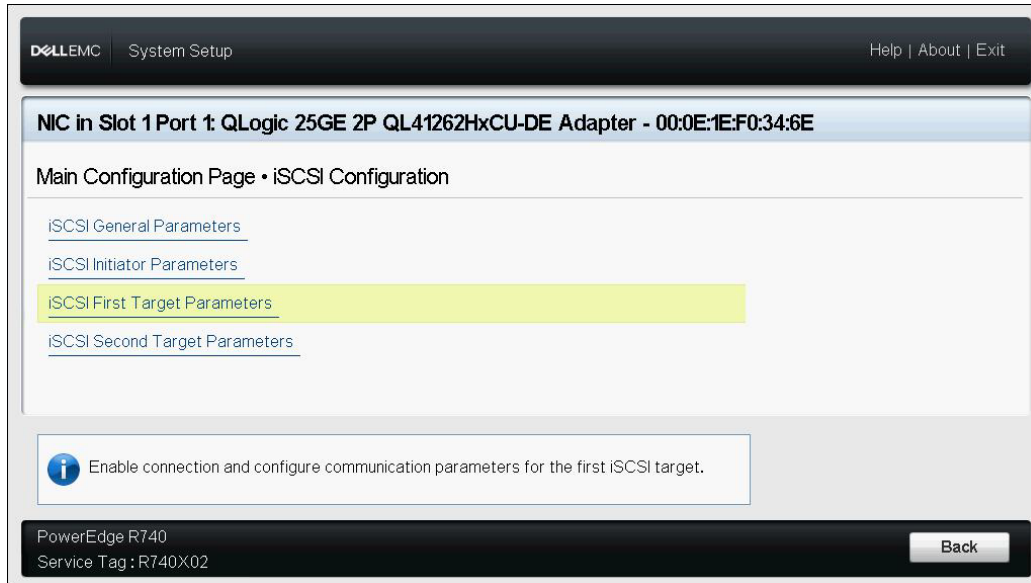


Figure 6-10. System Setup: Selecting iSCSI First Target Parameters

9. On the iSCSI First Target Parameters page, set the **Connect** option to **Enabled** for the iSCSI target.
10. Type values for the following parameters for the iSCSI target, and then press ENTER:
 - IPv4* Address**
 - TCP Port**
 - Boot LUN**
 - iSCSI Name**
 - CHAP ID**
 - CHAP Secret**

NOTE

For the preceding parameters with an asterisk (*), the label will change to **IPv6** or **IPv4** (default) based on IP version set on the iSCSI General Parameters page, as shown in Figure 6-11.

The screenshot shows the Dell EMC System Setup utility interface. At the top, it says "DELL EMC System Setup" and "Help | About | Exit". Below that, a header identifies the network interface: "NIC in Slot 1 Port 1: QLogic 25GE 2P QL41262HxCU-DE Adapter - 00:0E:1E:F0:34:6E". The main configuration page is titled "Main Configuration Page • iSCSI Configuration • iSCSI First Target Parameters". It contains several fields: "Connect" with radio buttons for "Disabled" and "Enabled" (selected); "IPv4 Address" with the value "192.168.100.9"; "TCP Port" with the value "3260"; "Boot LUN" with the value "1"; "iSCSI Name" (empty); "CHAP ID" (empty); and "CHAP Secret" (empty). Below these fields is an information box with an 'i' icon and the text "Specify the iSCSI Qualified Name (IQN) of the first iSCSI storage target." At the bottom left, it shows "PowerEdge R740" and "Service Tag : R740X02". At the bottom right, there is a "Back" button.

Figure 6-11. System Setup: iSCSI First Target Parameters

11. Return to the iSCSI Boot Configuration page, and then press ESC.

12. If you want to configure a second iSCSI target device, select **iSCSI Second Target Parameters** (Figure 6-12), and enter the parameter values as you did in Step 10. This second target is used if the first target cannot be connected to. Otherwise, proceed to Step 13.

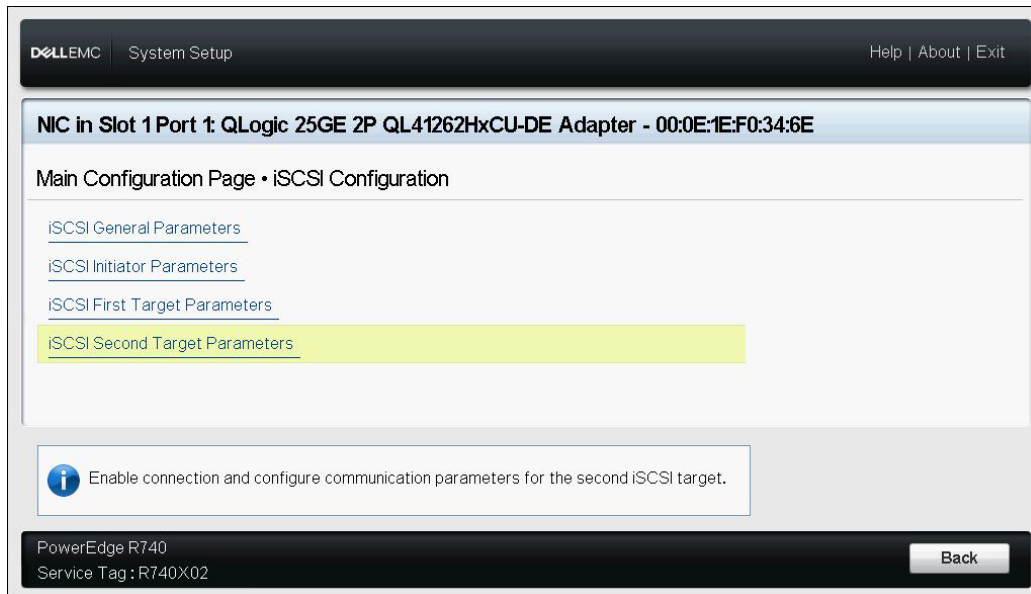


Figure 6-12. System Setup: iSCSI Second Target Parameters

13. Press ESC once, and a second time to exit.
14. Click **Yes** to save changes, or follow the OEM guidelines to save the device-level configuration. For example, click **Yes** to confirm the setting change (Figure 6-13).

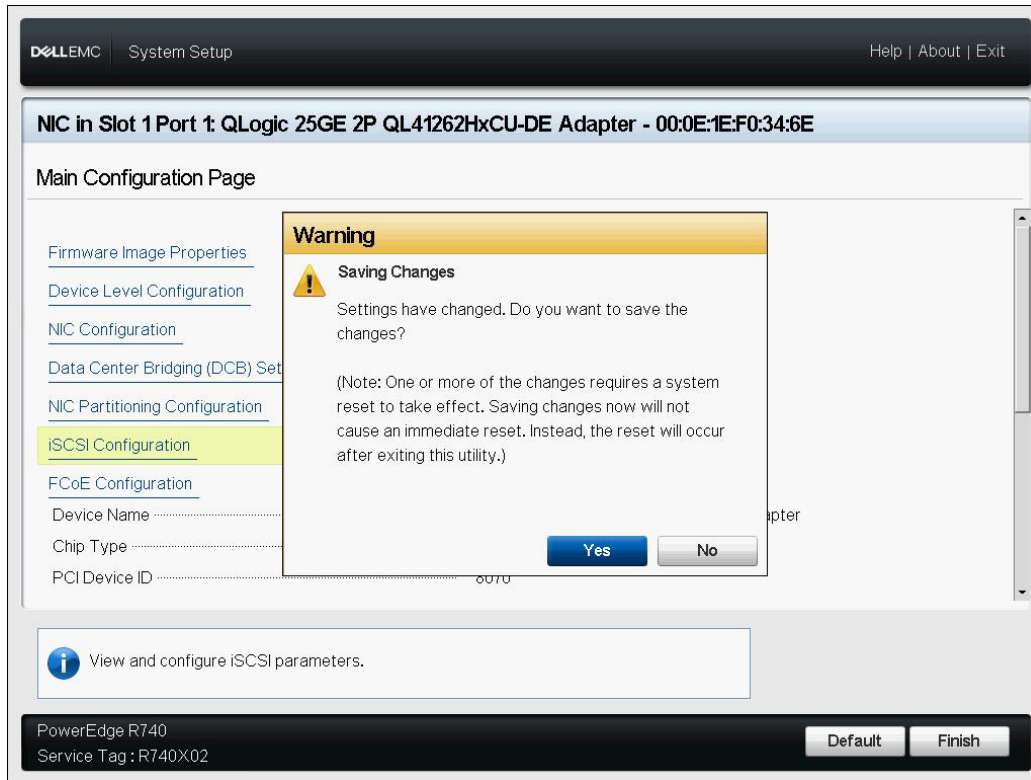


Figure 6-13. System Setup: Saving iSCSI Changes

15. After all changes have been made, reboot the system to apply the changes to the adapter's running configuration.

Dynamic iSCSI Boot Configuration

In a dynamic configuration, ensure that the system's IP address and target (or initiator) information are provided by a DHCP server (see IPv4 and IPv6 configurations in ["Configuring the DHCP Server to Support iSCSI Boot" on page 97](#)).

Any settings for the following parameters are ignored and do not need to be cleared (with the exception of the initiator iSCSI name for IPv4, CHAP ID, and CHAP secret for IPv6):

- Initiator Parameters
- First Target Parameters or Second Target Parameters

For information on configuration options, see [Table 6-2 on page 88](#).

NOTE

When using a DHCP server, the DNS server entries are overwritten by the values provided by the DHCP server. This override occurs even if the locally provided values are valid and the DHCP server provides no DNS server information. When the DHCP server provides no DNS server information, both the primary and secondary DNS server values are set to 0.0.0.0. When the Windows OS takes over, the Microsoft iSCSI initiator retrieves the iSCSI initiator parameters and statically configures the appropriate registries. It will overwrite whatever is configured. Because the DHCP daemon runs in the Windows environment as a user process, all TCP/IP parameters must be statically configured before the stack comes up in the iSCSI boot environment.

If DHCP Option 17 is used, the target information is provided by the DHCP server, and the initiator iSCSI name is retrieved from the value programmed from the Initiator Parameters window. If no value was selected, the controller defaults to the following name:

```
iqn.1995-05.com.qlogic.<11.22.33.44.55.66>.iscsiboot
```

The string 11.22.33.44.55.66 corresponds to the controller's MAC address. If DHCP Option 43 (IPv4 only) is used, any settings on the following windows are ignored and do not need to be cleared:

- Initiator Parameters
- First Target Parameters, or Second Target Parameters

To configure the iSCSI boot parameters using dynamic configuration:

- On the iSCSI General Parameters page, set the following options, as shown in [Figure 6-14](#):
 - TCP/IP Parameters via DHCP:** Enabled
 - iSCSI Parameters via DHCP:** Enabled
 - CHAP Authentication:** As required
 - IP Version:** As required (IPv4 or IPv6)
 - CHAP Mutual Authentication:** As required
 - DHCP Vendor ID:** As required
 - HBA Boot Mode:** As required
 - Virtual LAN ID:** As required
 - Virtual LAN Mode:** As required¹

¹ **Virtual LAN Mode** is not necessarily required when using a dynamic (externally provided) configuration from the DHCP server.

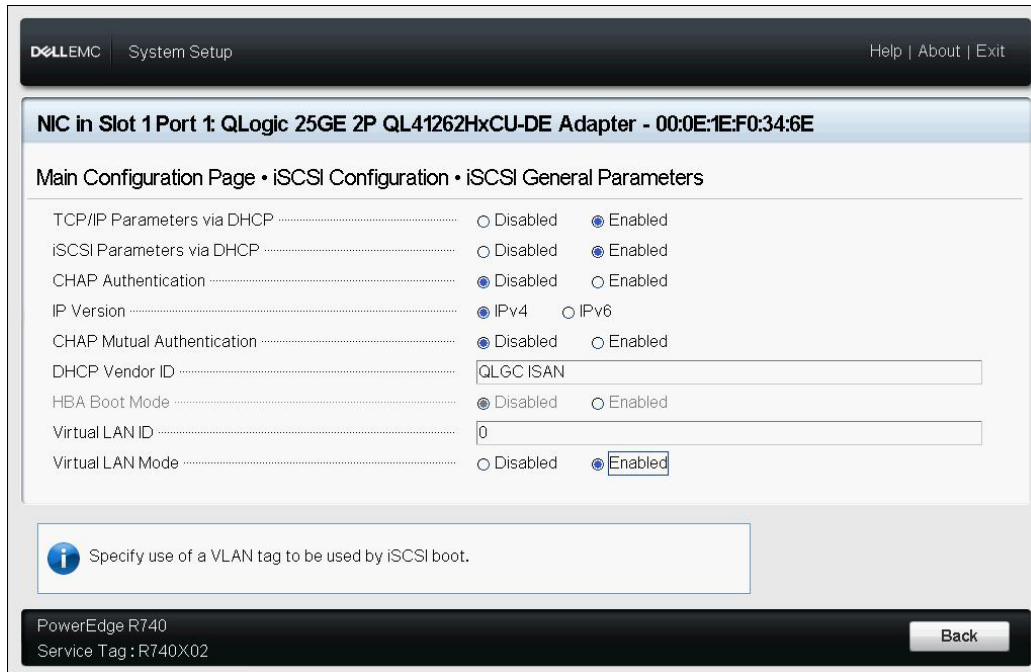


Figure 6-14. System Setup: iSCSI General Parameters

Enabling CHAP Authentication

Ensure that the CHAP authentication is enabled on the target.

To enable CHAP authentication:

1. Go to the iSCSI General Parameters page.
2. Set **CHAP Authentication** to **Enabled**.
3. In the Initiator Parameters window, type values for the following:
 - CHAP ID** (up to 255 characters)
 - CHAP Secret** (if authentication is required; must be 12 to 16 characters in length)
4. Press ESC to return to the iSCSI Boot Configuration page.
5. On the **iSCSI Boot Configuration Menu**, select **iSCSI First Target Parameters**.
6. In the iSCSI First Target Parameters window, type values used when configuring the iSCSI target:
 - CHAP ID** (optional if two-way CHAP)
 - CHAP Secret** (optional if two-way CHAP; must be 12 to 16 characters in length or longer)

7. Press ESC to return to the iSCSI Boot Configuration Menu.
8. Press ESC, and then select confirm **Save Configuration**.

Configuring the DHCP Server to Support iSCSI Boot

The DHCP server is an optional component, and is only necessary if you will be doing a dynamic iSCSI boot configuration setup (see “[Dynamic iSCSI Boot Configuration](#)” on page 94).

Configuring the DHCP server to support iSCSI boot differs for IPv4 and IPv6:

- [DHCP iSCSI Boot Configurations for IPv4](#)
- [Configuring the DHCP Server](#)
- [Configuring DHCP iSCSI Boot for IPv6](#)
- [Configuring vLANs for iSCSI Boot](#)

DHCP iSCSI Boot Configurations for IPv4

DHCP includes several options that provide configuration information to the DHCP client. For iSCSI boot, Marvell FastLinQ adapters support the following DHCP configurations:

- [DHCP Option 17, Root Path](#)
- [DHCP Option 43, Vendor-specific Information](#)

DHCP Option 17, Root Path

Option 17 is used to pass the iSCSI target information to the iSCSI client.

The format of the root path, as defined in IETF RFC 4173, is:

```
"iscsi:"<servername>":"<protocol>":"<port>":"<LUN>":"<targetname>"
```

[Table 6-3](#) lists the DHCP Option 17 parameters.

Table 6-3. DHCP Option 17 Parameter Definitions

Parameter	Definition
"iscsi:"	A literal string
<servername>	IP address or fully qualified domain name (FQDN) of the iSCSI target
":"	Separator
<protocol>	IP protocol used to access the iSCSI target. Because only TCP is currently supported, the protocol is 6.
<port>	Port number associated with the protocol. The standard port number for iSCSI is 3260.

Table 6-3. DHCP Option 17 Parameter Definitions (Continued)

Parameter	Definition
<LUN>	Logical unit number to use on the iSCSI target. The value of the LUN must be represented in hexadecimal format. A LUN with an ID of 64 must be configured as 40 within the Option 17 parameter on the DHCP server.
<targetname>	Target name in either IQN or EUI format. For details on both IQN and EUI formats, refer to RFC 3720. An example IQN name is <code>iqn.1995-05.com.QLogic:iscsi-target</code> .

DHCP Option 43, Vendor-specific Information

DHCP Option 43 (vendor-specific information) provides more configuration options to the iSCSI client than does DHCP Option 17. In this configuration, three additional sub-options are provided that assign the initiator IQN to the iSCSI boot client, along with two iSCSI target IQNs that can be used for booting. The format for the iSCSI target IQN is the same as that of DHCP Option 17, while the iSCSI initiator IQN is simply the initiator's IQN.

NOTE

DHCP Option 43 is supported on IPv4 only.

Table 6-4 lists the DHCP Option 43 sub-options.

Table 6-4. DHCP Option 43 Sub-option Definitions

Sub-option	Definition
201	First iSCSI target information in the standard root path format: <code>"iscsi:"<servername>":"<protocol>":"<port>":"<LUN>": "<targetname>"</code>
202	Second iSCSI target information in the standard root path format: <code>"iscsi:"<servername>":"<protocol>":"<port>":"<LUN>": "<targetname>"</code>
203	iSCSI initiator IQN

Using DHCP Option 43 requires more configuration than DHCP Option 17, but it provides a richer environment and more configuration options. You should use DHCP Option 43 when performing dynamic iSCSI boot configuration.

Configuring the DHCP Server

Configure the DHCP server to support either Option 16, 17, or 43.

NOTE

The format of DHCPv6 Option 16 and Option 17 are fully defined in RFC 3315.

If you use Option 43, you must also configure Option 60. The value of Option 60 must match the DHCP Vendor ID value, QLGC ISAN, as shown in the **iSCSI General Parameters** of the iSCSI Boot Configuration page.

Configuring DHCP iSCSI Boot for IPv6

The DHCPv6 server can provide several options, including stateless or stateful IP configuration, as well as information for the DHCPv6 client. For iSCSI boot, Marvell FastLinQ adapters support the following DHCP configurations:

- [DHCPv6 Option 16, Vendor Class Option](#)
- [DHCPv6 Option 17, Vendor-Specific Information](#)

NOTE

The DHCPv6 standard Root Path option is not yet available. Marvell suggests using Option 16 or Option 17 for dynamic iSCSI boot IPv6 support.

DHCPv6 Option 16, Vendor Class Option

DHCPv6 Option 16 (vendor class option) must be present and must contain a string that matches your configured DHCP Vendor ID parameter. The DHCP Vendor ID value is QLGC ISAN, as shown in the **General Parameters** of the iSCSI Boot Configuration menu.

The content of Option 16 should be `<2-byte length> <DHCP Vendor ID>`.

DHCPv6 Option 17, Vendor-Specific Information

DHCPv6 Option 17 (vendor-specific information) provides more configuration options to the iSCSI client. In this configuration, three additional sub-options are provided that assign the initiator IQN to the iSCSI boot client, along with two iSCSI target IQNs that can be used for booting.

Table 6-5 lists the DHCP Option 17 sub-options.

Table 6-5. DHCP Option 17 Sub-option Definitions

Sub-option	Definition
201	First iSCSI target information in the standard root path format: "iscsi:" [<servername>] ":" <protocol> ":" <port> ":" <LUN> " : " <targetname> "
202	Second iSCSI target information in the standard root path format: "iscsi:" [<servername>] ":" <protocol> ":" <port> ":" <LUN> " : " <targetname> "
203	iSCSI initiator IQN

Brackets [] are required for the IPv6 addresses.

The format of Option 17 should be:

```
<2-byte Option Number 201|202|203> <2-byte length> <data>
```

Configuring vLANs for iSCSI Boot

iSCSI traffic on the network may be isolated in a Layer 2 vLAN to segregate it from general traffic. If this is the case, make the iSCSI interface on the adapter a member of that vLAN.

To configure vLAN for iSCSI boot:

1. Go to the **iSCSI Configuration Page** for the port.
2. Select **iSCSI General Parameters**.

3. Select **VLAN ID** to enter and set the vLAN value, as shown in [Figure 6-15](#).

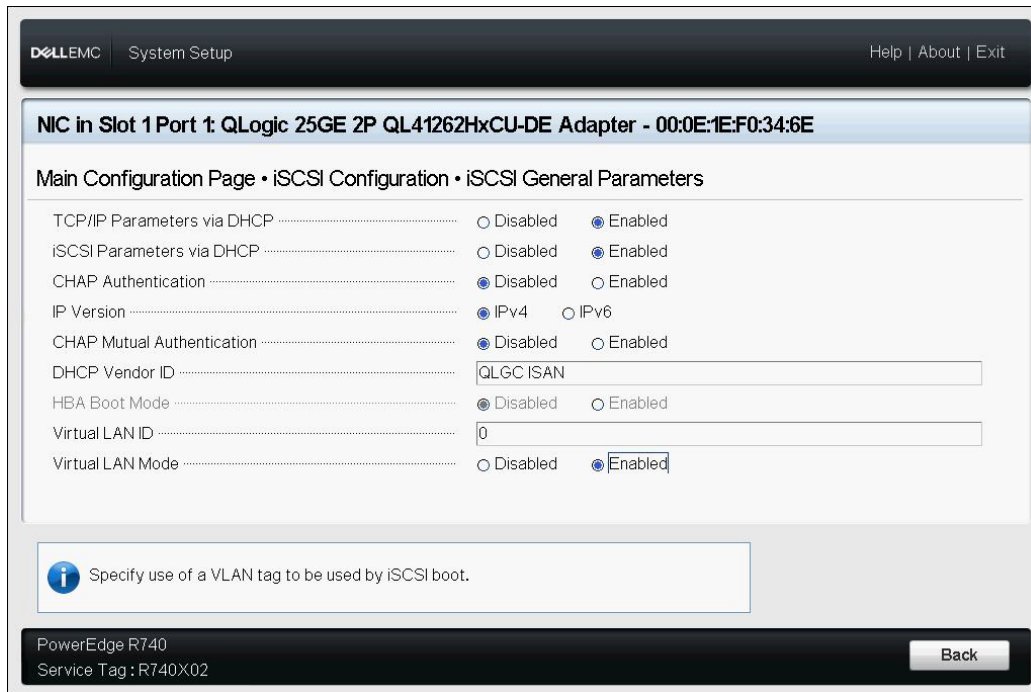


Figure 6-15. System Setup: iSCSI General Parameters, VLAN ID

Configuring iSCSI Boot from SAN on Windows

Adapters support iSCSI boot to enable network boot of operating systems to diskless systems. iSCSI boot allows a Windows operating system to boot from an iSCSI target machine located remotely over a standard IP network. You can set the L4 iSCSI option (offload path with Marvell offload iSCSI driver) by opening the **NIC Configuration** menu and setting the **Boot Protocol** to **UEFI iSCSI**.

iSCSI boot from SAN for Windows information includes the following:

- [Before You Begin](#)
- [Selecting the Preferred iSCSI Boot Mode](#)
- [Configuring iSCSI General Parameters](#)
- [Configuring the iSCSI Initiator](#)
- [Configuring the iSCSI Targets](#)
- [Detecting the iSCSI LUN and Injecting the Marvell Drivers](#)

Before You Begin

Before you begin configuring iSCSI boot from SAN on a Windows machine, note the following:

- iSCSI boot is only supported for NPar with **NParEP Mode**. Before configuring iSCSI boot:
 1. Access the Device Level Configuration page.
 2. Set the **Virtualization Mode** to **Npar**.
 3. Set the **NParEP Mode** to **Enabled**.
- The server boot mode must be UEFI.
- The PermitTotalPortShutdown diagnostic feature can take down a port link when enabled. This feature cannot be used on ports configured to boot from SAN.
- iSCSI boot on 41000 Series Adapters is not supported in legacy BIOS.
- Marvell recommends that you disable the Integrated RAID Controller.

Selecting the Preferred iSCSI Boot Mode

To select the iSCSI boot mode on Windows:

1. On the NIC Partitioning Configuration page for a selected partition, set the **iSCSI Offload Mode** to **Enabled**.
2. On the NIC Configuration page, set the **Boot Protocol** option to **UEFI iSCSI HBA**.

Configuring iSCSI General Parameters

Configure the Marvell iSCSI boot software for either static or dynamic configuration. For configuration options available from the General Parameters window, see [Table 6-2 on page 88](#), which lists parameters for both IPv4 and IPv6.

To set the iSCSI general parameters on Windows:

1. From the Main Configuration page, select **iSCSI Configuration**, and then select **iSCSI General Parameters**.
2. On the iSCSI General Parameters page (see [Figure 6-7 on page 87](#)), press the DOWN ARROW key to select a parameter, and then press the ENTER key to input the following values (see [Table 6-2 on page 88](#) provides descriptions of these parameters):
 - TCP/IP Parameters via DHCP: Disabled** (for static iSCSI boot), or **Enabled** (for dynamic iSCSI boot)
 - iSCSI Parameters via DHCP: Disabled**
 - CHAP Authentication: As required**

- IP Version:** As required (IPv4 or IPv6)
- Virtual LAN ID:** (Optional) You can isolate iSCSI traffic on the network in a Layer 2 vLAN to segregate it from general traffic. To segregate traffic, make the iSCSI interface on the adapter a member of the Layer 2 vLAN by setting this value.

Configuring the iSCSI Initiator

To set the iSCSI initiator parameters on Windows:

1. From the Main Configuration page, select **iSCSI Configuration**, and then select **iSCSI Initiator Parameters**.
2. On the iSCSI Initiator Parameters page (see [Figure 6-9 on page 90](#)), select the following parameters, and then type a value for each:
 - IPv4* Address**
 - Subnet Mask**
 - IPv4* Default Gateway**
 - IPv4* Primary DNS**
 - IPv4* Secondary DNS**
 - Virtual LAN ID:** (Optional) You can isolate iSCSI traffic on the network in a Layer 2 vLAN to segregate it from general traffic. To segregate traffic, make the iSCSI interface on the adapter a member of the Layer 2 vLAN by setting this value.
 - iSCSI Name.** Corresponds to the iSCSI initiator name to be used by the client system.
 - CHAP ID**
 - CHAP Secret**

NOTE

For the preceding items with asterisks (*), note the following:

- The label will change to **IPv6** or **IPv4** (default) based on the IP version set on the iSCSI General Parameters page (see [Figure 6-7 on page 87](#)).
- Carefully enter the IP address. There is no error-checking performed against the IP address to check for duplicates, incorrect segment, or network assignment.

3. Select **iSCSI First Target Parameters** ([Figure 6-10 on page 91](#)), and then press ENTER.

Configuring the iSCSI Targets

You can set up the iSCSI first target, second target, or both at once.

To set the iSCSI target parameters on Windows:

1. From the Main Configuration page, select **iSCSI Configuration**, and then select **iSCSI First Target Parameters**.
2. On the iSCSI First Target Parameters page, set the **Connect** option to **Enabled** for the iSCSI target.
3. Type values for the following parameters for the iSCSI target, and then press ENTER:
 - IPv4* Address**
 - TCP Port**
 - Boot LUN**
 - iSCSI Name**
 - CHAP ID**
 - CHAP Secret**

NOTE

For the preceding parameters with an asterisk (*), the label will change to **IPv6** or **IPv4** (default) based on IP version set on the iSCSI General Parameters page, as shown in [Figure 6-7 on page 87](#).

4. If you want to configure a second iSCSI target device, select **iSCSI Second Target Parameters** ([Figure 6-12 on page 93](#)), and enter the parameter values as you did in [Step 3](#). This second target is used if the first target cannot be connected to. Otherwise, proceed to [Step 5](#).
5. In the Warning dialog box, click **Yes** to save the changes, or follow the OEM guidelines to save the device-level configuration.

Detecting the iSCSI LUN and Injecting the Marvell Drivers

1. Reboot the system, access the HII, and determine if the iSCSI LUN is detected. Issue the following UEFI Shell (version 2) script command:

```
map -r -b
```

The output from the preceding command shown in [Figure 6-16](#) indicates that the iSCSI LUN was detected successfully at the preboot level.

```
BLK19: Alias(s) :  
PciRoot (0x3) /Pci (0x0,0x0) /Pci (0x0,0x4) /MAC (000E1ED6624C,0x0) /iSCSI (iqn  
.1986-03.com.hp:storage.p2000g3.13491b47fb,0x0,0x0,None,None,None,TCP)  
BLK21: Alias(s) :  
PciRoot (0x3) /Pci (0x0,0x0) /Pci (0x0,0x4) /MAC (000E1ED6624C,0x0) /iSCSI (iqn  
.1986-03.com.hp:storage.p2000g3.13491b47fb,0x0,0x0,None,None,None,TCP) /HD (2,GPT,  
1910807F-AA79-4DD9-8D5E-4EE6ABADC920,0x4E800,0x403B00)  
BLK22: Alias(s) :  
PciRoot (0x3) /Pci (0x0,0x0) /Pci (0x0,0x4) /MAC (000E1ED6624C,0x0) /iSCSI (iqn  
.1986-03.com.hp:storage.p2000g3.13491b47fb,0x0,0x0,None,None,None,TCP) /HD (3,GPP  
ress ENTER to continue or 'Q' break: _
```

Figure 6-16. Detecting the iSCSI LUN Using UEFI Shell (Version 2)

2. On the newly detected iSCSI LUN, select an installation source such as using a WDS server, mounting the .ISO with an integrated Dell Remote Access Controller (iDRAC), or using a CD/DVD.
3. In the Windows Setup window ([Figure 6-17](#)), select the drive name on which to install the driver.

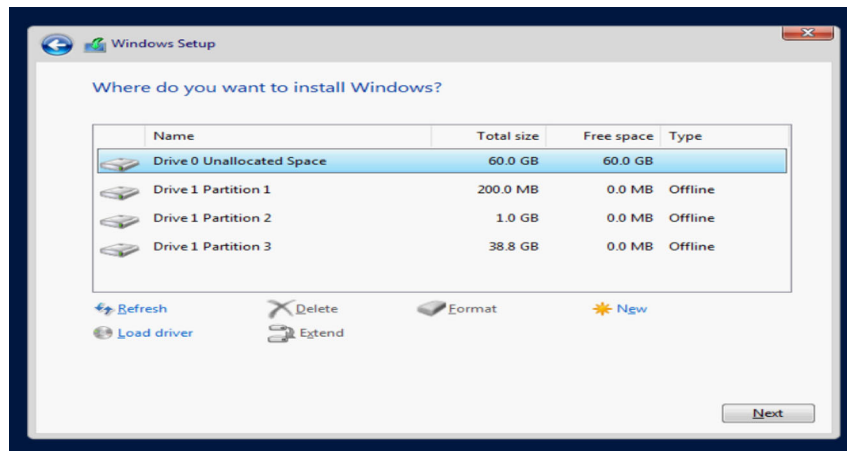


Figure 6-17. Windows Setup: Selecting Installation Destination

4. Inject the latest Marvell drivers by mounting drivers in the virtual media:
 - a. Click **Load driver**, and then click **Browse** (see [Figure 6-18](#)).

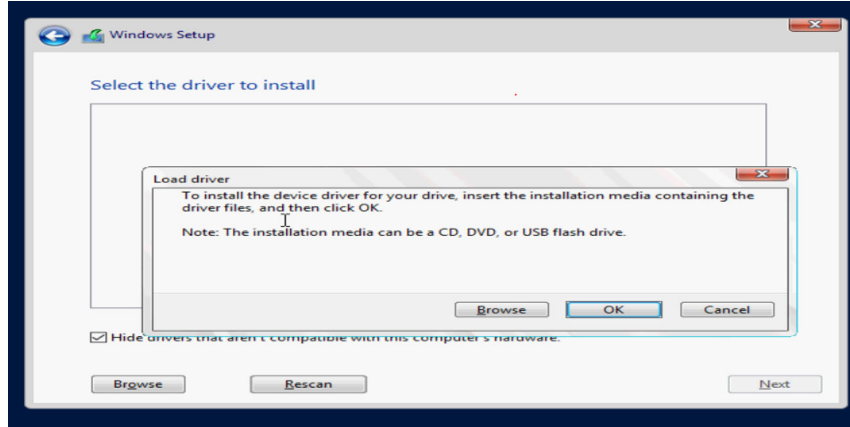


Figure 6-18. Windows Setup: Selecting Driver to Install

- b. Navigate to the driver location and choose the qevbd driver.
 - c. Choose the adapter on which to install the driver, and then click **Next** to continue.
5. Repeat [Step 4](#) to load the qeios driver (Marvell L4 iSCSI driver).
6. After injecting the qevbd and qeios drivers, click **Next** to begin installation on the iSCSI LUN. Then follow the on-screen instructions.

The server will undergo a reboot multiple times as part of the installation process, and then will boot up from the iSCSI boot from SAN LUN.
7. If it does not automatically boot, access the **Boot Menu** and select the specific port boot entry to boot from the iSCSI LUN.

Configuring iSCSI Boot from SAN on Linux

This section provides iSCSI boot from SAN procedures for the following Linux distributions:

- [Configuring iSCSI Boot from SAN for RHEL 7.8 and Later](#)
- [Configuring iSCSI Boot from SAN for SLES 12 SP3 and Later](#)
- [Configuring iSCSI Boot from SAN for SLES 15 SP1 and Later](#)

Configuring iSCSI Boot from SAN for RHEL 7.8 and Later

To install RHEL 7.8 and later:

1. Boot from the RHEL 7.x installation media with the iSCSI target already connected in UEFI.

```
Install Red Hat Enterprise Linux 7.x
```

```
Test this media & install Red Hat Enterprise 7.x
```

```
Troubleshooting -->
```

```
Use the UP and DOWN keys to change the selection
```

```
Press 'e' to edit the selected item or 'c' for a command  
prompt
```

2. To install an out-of-box driver, press the E key. Otherwise, proceed to [Step 6](#).
3. Select the kernel line, and then press the E key.
4. Issue the following command, and then press CTRL+X to start.

```
inst.dd modprobe.blacklist=qed,qede,qedr,qedi,qedf
```

The installation process prompts you to install the out-of-box driver.

5. If required for your setup, load the FastLinQ driver update disk when prompted for additional driver disks. Otherwise, if you have no other driver update disks to install, press the C key.
6. Continue with the installation. You can skip the media test. Click **Next** to continue with the installation.
7. In the Configuration window, select the language to use during the installation process, and then click **Continue**.
8. In the Installation Summary window, click **Installation Destination**. The disk label is *sda*, indicating a single-path installation. If you configured multipath, the disk has a device mapper label.
9. In the **Specialized & Network Disks** section, select the iSCSI LUN.
10. Type the root user's password, and then click **Next** to complete the installation.
11. Reboot the server, and then add the following parameters in the command line:

```
rd.iscsi.firmware  
rd.driver.pre=qed,qedi (to load all drivers pre=qed,qedi,qede,qedf)  
selinux=0
```
12. After a successful system boot, edit the `/etc/modprobe.d/anaconda-blacklist.conf` file to remove the blacklist entry for the selected driver.

13. Edit the `/etc/default/grub` file as follows:
 - a. Locate the string in double quotes as shown in the following example. The command line is a specific reference to help find the string.

```
GRUB_CMDLINE_LINUX="iscsi_firmware"
```
 - b. The command line may contain other parameters that can remain. Change only the `iscsi_firmware` string as follows:

```
GRUB_CMDLINE_LINUX="rd.iscsi.firmware selinux=0"
```
14. Create a new `grub.cfg` file by issuing the following command:

```
# grub2-mkconfig -o /boot/efi/EFI/redhat/grub.cfg
```
15. Rebuild the ramdisk by issuing the `dracut -f` command, and then reboot.

NOTE

When installing iSCSI BFS in Linux with a multipath I/O (MPIO) configuration and a single path active, use the following settings in the `multipath.conf` file:

```
defaults {  
    find_multipaths yes  
    user_friendly_names yes  
    polling_interval 5  
    fast_io_fail_tmo 5  
    dev_loss_tmo 10  
    checker_timeout 15  
    no_path_retry queue  
}
```

These suggested settings are tunable and provided as guidance for iSCSI BFS to be operational.

For more information, contact the appropriate OS vendor.

Configuring iSCSI Boot from SAN for SLES 12 SP3 and Later

To install SLES 12 SP3 and later:

1. Boot from the SLES 12 SP3 installation media with the iSCSI target pre-configured and connected in UEFI.
2. Update the latest driver package by adding the `dud=1` parameter in the installer command parameter. The driver update disk is required because the necessary iSCSI drivers are not in box.

NOTE

For SLES 12 SP3 only: If the server is configured for Multi-Function mode (NPar), you must provide the following additional parameters as part of this step:

```
dud=1 brokenmodules=qed,qedi,qedf,qede withiscsi=1  
[BOOT_IMAGE=/boot/x86_64/loader/linux dud=1  
brokenmodules=qed,qedi,qedf,qede withiscsi=1]
```

3. Complete the installation steps specified by the SLES 12 SP3 OS.

Known Issue in DHCP Configuration

In DHCP configuration for SLES 12 SP3 and later, the first boot after an OS installation may fail if the initiator IP address acquired from the DHCP server is in a different range than the target IP address. To resolve this issue, boot the OS using static configuration, update the latest `iscsiuio` out-of-box RPM, rebuild the `initrd`, and then reboot the OS using DHCP configuration. The OS should now boot successfully.

Configuring iSCSI Boot from SAN for SLES 15 SP1 and Later

To install SLES15 SP1 and later:

1. Boot from the SLES15 installation media with the iSCSI target already connected in UEFI.

```
Install SLES15SP1  
Test this media & install SLES15SP1  
Troubleshooting -->
```

```
Use the UP and DOWN keys to change the selection  
Press 'e' to edit the selected item or 'c' for a command  
prompt
```

2. To install an out-of-box driver, perform the following steps. Otherwise, go to [Step 3](#).
 - a. Press the E key.
 - b. Select the kernel line, and then press the E key.
 - c. Issue the following command, and then press ENTER.

```
dud=1
```

The installation process prompts you to install the out-of-box driver.
 - d. If required for your setup, load the FastLinQ driver update disk when prompted for additional driver disks.
Otherwise, if you have no other driver update disks to install, press the C key.
3. Continue with the installation.
The iSCSI Configuration menu appears.
4. Click and verify the iSCSI session details.
5. Select **OK**, follow the on-screen instructions, and start the installation.

Configuring iSCSI Boot from SAN on VMware

Because VMware does not natively support iSCSI boot from SAN offload, you must configure BFS through the software in preboot, and then transition to offload upon OS driver loads. For more information, see [“Enabling NPar and the iSCSI HBA” on page 83](#).

In VMware ESXi, iSCSI BFS configuration procedures include:

- [Setting the UEFI Main Configuration](#)
- [Configuring the System BIOS for iSCSI Boot \(L2\)](#)
- [Mapping the CD or DVD for OS Installation](#)

Setting the UEFI Main Configuration

To configure iSCSI boot from SAN on VMware:

1. Plug the 41000 Series Adapter into a Dell 14G server. For example, plug a PCIE and LOM (four ports or two ports) into an R740 server.
2. In the HII, go to **System Setup**, select **Device Settings**, and then select an integrated NIC port to configure. Click **Finish**.
3. On the **Main Configuration Page**, select **NIC Partitioning Configuration**, and then click **Finish**.

4. On the **Main Configuration Page**, select **Firmware Image Properties**, view the non-configurable information, and then click **Finish**.
5. On the **Main Configuration Page**, select **Device Level Configuration**.
6. Complete the Device Level Configuration page (see [Figure 6-19](#)) as follows:
 - a. For **Virtualization Mode**, select either **None**, **NPar**, or **NPar_EP** for IBFT installation through the NIC interface.
 - b. For **NParEP Mode**, select **Disabled**.
 - c. For **UEFI Driver Debug Level**, select **10**.

Integrated NIC 1 Port 1: QLogic 4x10GE QL41164HMRJ CNA - 00:0E:1E:D2:7D:64

Main Configuration Page • Device Level Configuration

Virtualization Mode	NPar
NParEP Mode	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled
UEFI Driver Debug Level	10

Figure 6-19. Integrated NIC: Device Level Configuration for VMware

7. Go to the **Main Configuration Page** and select **NIC Partitioning Configuration**.
8. On the NIC Partitioning Configuration page, select **Partition 1 Configuration**.
9. Complete the Partition 1 Configuration page as follows:
 - a. For **Link Speed**, select either **Auto Neg**, **10Gbps**, or **1Gbps**.
 - b. Ensure that the link is up.
 - c. For **Boot Protocol**, select **None**.
 - d. For **Virtual LAN Mode**, select **Disabled**.
10. On the NIC Partitioning Configuration page, select **Partition 2 Configuration**.
11. Complete the Partition 2 Configuration page (see [Figure 6-20](#)) as follows:
 - a. For **FCoE Mode**, select **Disabled**.
 - b. For **iSCSI Offload Mode**, select **Disabled**.

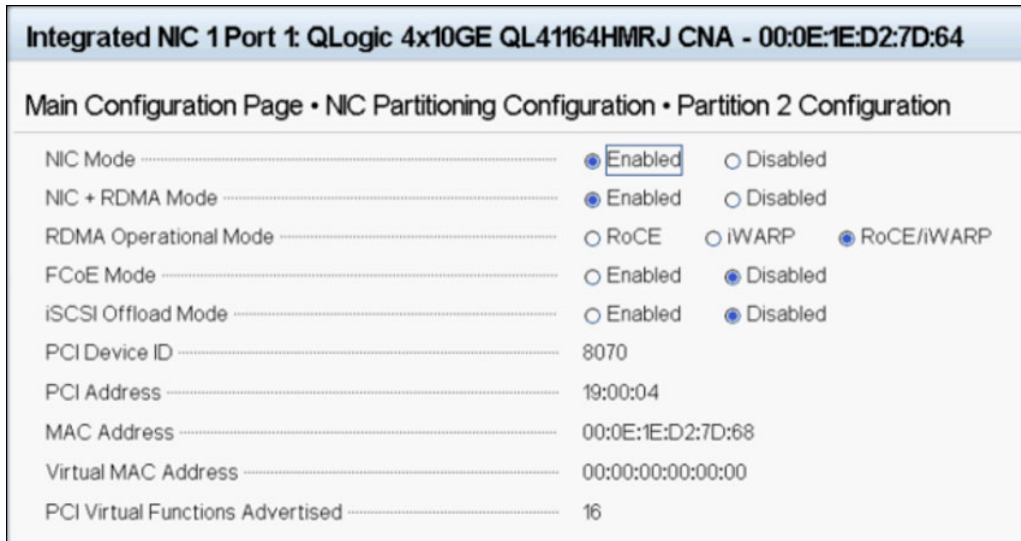


Figure 6-20. Integrated NIC: Partition 2 Configuration for VMware

Configuring the System BIOS for iSCSI Boot (L2)

To configure the System BIOS on VMware:

1. On the System BIOS Settings page, select **Boot Settings**.
2. Complete the Boot Settings page as shown in [Figure 6-21](#).

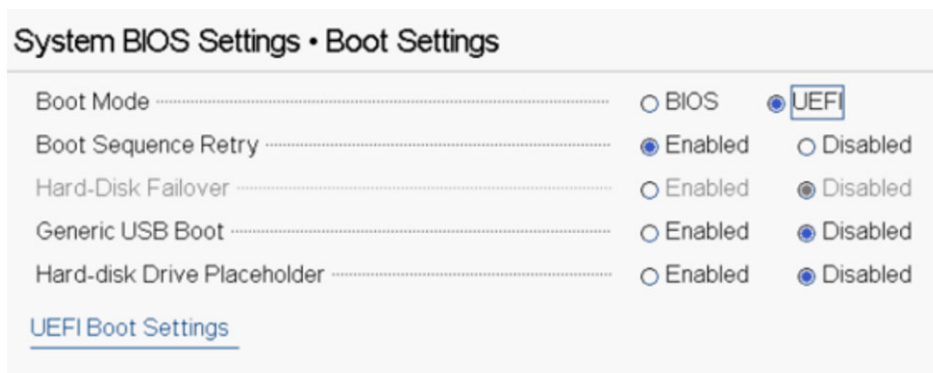


Figure 6-21. Integrated NIC: System BIOS, Boot Settings for VMware

3. On the System BIOS Settings page, select **Network Settings**.
4. On the Network Settings page under **UEFI iSCSI Settings**:
 - a. For **iSCSI Device1**, select **Enabled**.
 - b. Select **UEFI Boot Settings**.

5. On the iSCSI Device1 Settings page:
 - a. For **Connection 1**, select **Enabled**.
 - b. Select **Connection 1 Settings**.
6. On the Connection 1 Settings page (see [Figure 6-22](#)):
 - a. For **Interface**, select the adapter port on which to test the iSCSI boot firmware table (IBFT) boot from SAN.
 - b. For **Protocol**, select either **IPv4** or **IPv6**.
 - c. For **VLAN**, select either **Disabled** (the default) or **Enabled** (if you want to test IBFT with a VLAN).
 - d. For **DHCP**, select **Enabled** (if the IP address is from the DHCP server) or **Disabled** (to use static IP configuration).
 - e. For **Target info via DHCP**, select **Disabled**.

The screenshot displays the 'System BIOS' settings for 'Connection 1 Settings'. The settings are as follows:

Setting	Value
Interface	Integrated NIC 1 Port 1 Partition 1
Protocol	<input checked="" type="radio"/> IPv4 <input type="radio"/> IPv6
VLAN	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled
VLAN ID	100
VLAN Priority	0
Retry Count	3
Timeout	10000
DHCP	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled
Initiator IP Address	192.169.100.39
Initiator Subnet Mask	255.255.0.0
Initiator Gateway	192.169.100.1
Target info via DHCP	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled

Figure 6-22. Integrated NIC: System BIOS, Connection 1 Settings for VMware

7. Complete the target details, and for **Authentication Type**, select either **CHAP** (to set CHAP details) or **None** (the default). [Figure 6-23](#) shows an example.

The screenshot shows the 'System BIOS' settings for 'Connection 1 Settings'. The 'Target info via DHCP' option is disabled. The 'Target Name' is 'iqn.2000-05.com.3pardata:20210002ac010f9', 'Target IP Address' is '192.168.17.254', 'Target Port' is '3260', and 'Target Boot Lun' is '0'. The 'Authentication Type' is set to 'None', and the 'CHAP Type' is 'Mutual'. The 'CHAP Name' is 'preboot', 'CHAP Secret' is '123456789123', 'Reverse CHAP Name' is 'preboot1', and 'Reverse CHAP Secret' is '987654321123'.

System BIOS	
System BIOS Settings • Network Settings • Connection 1 Settings	
Target info via DHCP	<input type="radio"/> Enabled <input checked="" type="radio"/> Disabled
Target Name	<input type="text" value="iqn.2000-05.com.3pardata:20210002ac010f9"/>
Target IP Address	<input type="text" value="192.168.17.254"/>
Target Port	<input type="text" value="3260"/>
Target Boot Lun	<input type="text" value="0"/>
ISID	<input type="text"/>
Authentication Type	<input type="radio"/> CHAP <input checked="" type="radio"/> None
CHAP Type	<input type="radio"/> One Way <input checked="" type="radio"/> Mutual
CHAP Name	<input type="text" value="preboot"/>
CHAP Secret	<input type="text" value="123456789123"/>
Reverse CHAP Name	<input type="text" value="preboot1"/>
Reverse CHAP Secret	<input type="text" value="987654321123"/>

Figure 6-23. Integrated NIC: System BIOS, Connection 1 Settings (Target) for VMware

8. Save all configuration changes, and then reboot the server.
9. During system boot up, press the F11 key to start the Boot Manager.
10. In the Boot Manager under **Boot Menu**, **Select UEFI Boot Option**, select the **Embedded SATA Port AHCI Controller**.

Mapping the CD or DVD for OS Installation

To map the CD or DVD:

1. Create a customized ISO image using the ESXi-Customizer and inject the latest bundle or VIB.
2. Map the ISO to the server virtual console's virtual media.
3. On the virtual optical drive, load the ISO file.
4. After the ISO is loaded successfully, press the F11 key.

5. On the Select a Disk To Install Or Upgrade window, under **Storage Device**, select the **3PARdata W** disk, and then press the ENTER key. [Figure 6-24](#) shows an example.

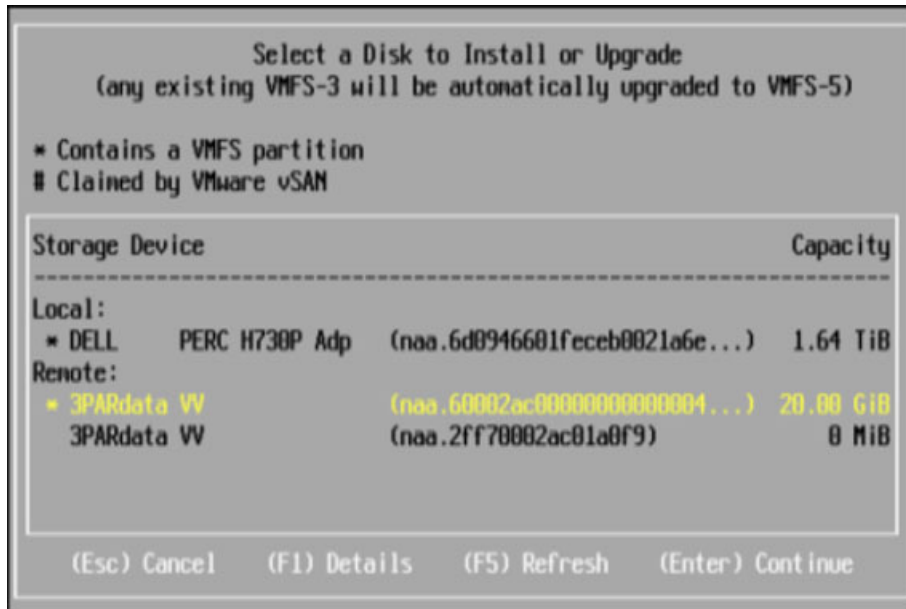


Figure 6-24. VMware iSCSI BFS: Selecting a Disk to Install

6. Start installation of the ESXi OS on the remote iSCSI LUN.
7. After the ESXi OS installation completes successfully, the system boots to the OS, as shown in [Figure 6-25](#).

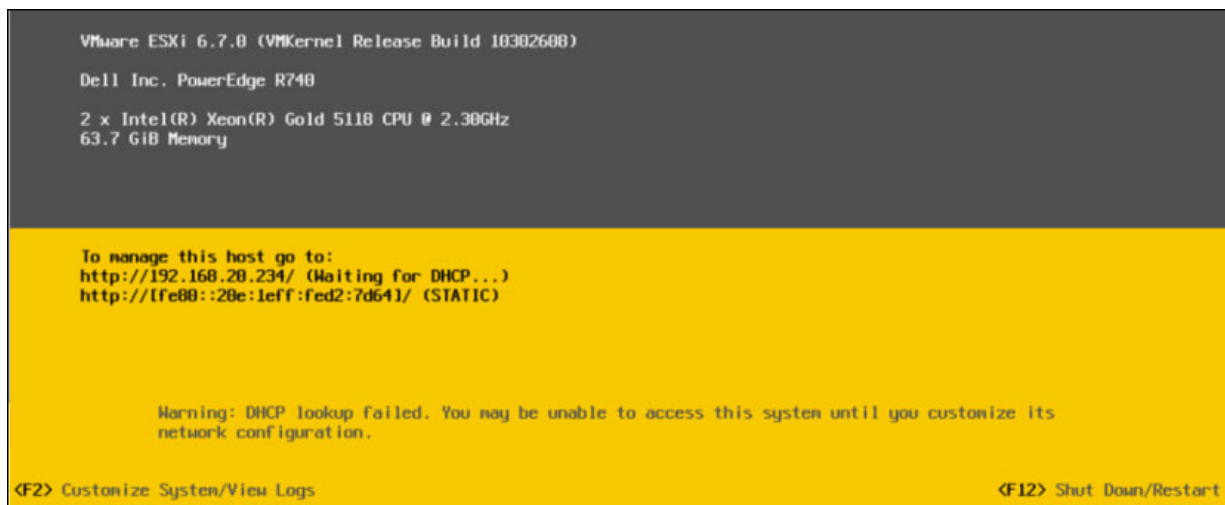


Figure 6-25. VMware iSCSI Boot from SAN Successful

FCoE Boot from SAN

Marvell 41000 Series Adapters support FCoE boot to enable network boot of operating systems to diskless systems. FCoE boot allows a Windows, Linux, or VMware operating system to boot from a Fibre Channel or FCoE target machine located remotely over an FCoE supporting network. You can set the FCoE option (offload path with Marvell offload FCoE driver) by opening the **NIC Configuration** menu and setting the **Boot Protocol** option to **FCoE**.

NOTE

The PermitTotalPortShutdown diagnostic feature can take down a port link when enabled. This feature cannot be used on ports configured to boot from SAN.

This section provides the following configuration information about FCoE boot from SAN:

- [FCoE Out-of-Box and Inbox Support](#)
- [FCoE Preboot Configuration](#)
- [Configuring FCoE Boot from SAN on Windows](#)
- [Configuring FCoE Boot from SAN on Linux](#)
- [Configuring FCoE Boot from SAN on VMware](#)

FCoE Out-of-Box and Inbox Support

[Table 6-6](#) lists the operating systems' inbox and out-of-box support for FCoE boot from SAN (BFS).

Table 6-6. FCoE Out-of-Box and Inbox Boot from SAN Support

OS Version	<u>Out-of-Box</u>	<u>Inbox</u>
	Hardware Offload FCoE BFS Support	Hardware Offload FCoE BFS Support
Windows 2016	Yes	No
Windows 2019	Yes	Yes

Table 6-6. FCoE Out-of-Box and Inbox Boot from SAN Support (Continued)

OS Version	<u>Out-of-Box</u>	<u>Inbox</u>
	Hardware Offload FCoE BFS Support	Hardware Offload FCoE BFS Support
Azure Stack HCI	Yes	Yes
RHEL 7.8	Yes	Yes
RHEL 7.9	Yes	Yes
RHEL 8.2	Yes	Yes
RHEL 8.3	Yes	Yes
SLES 15 SP1, SP2	Yes	Yes
VMware ESXi 6.7 U3	Yes	No
VMware ESXi 7.0 U1	Yes	No

FCoE Preboot Configuration

This section describes the installation and boot procedures for the Windows, Linux, and ESXi operating systems. To prepare the system BIOS, modify the system boot order and specify the BIOS boot protocol, if required.

NOTE

FCoE boot from SAN is supported on ESXi 5.5 and later. Not all adapter versions support FCoE and FCoE boot from SAN.

Specifying the BIOS Boot Protocol

FCoE boot from SAN is supported in UEFI mode only. Set the platform in boot mode (protocol) using the system BIOS configuration to UEFI.

NOTE

FCoE BFS is not supported in legacy BIOS mode.

Enabling NPar and the FCoE HBA

To use NPar to configure the FCoE HBA for storage offloads, follow the instructions in the *Application Note, Enabling Storage Offload on Dell and Marvell FastLinQ 41000 Series Adapters*, available on the Marvell Web Site.

Configuring Adapter UEFI Boot Mode

To configure the boot mode to FCOE:

1. Restart the system.
2. Press the OEM hot key to enter System Setup (Figure 6-26). This is also known as UEFI HII.

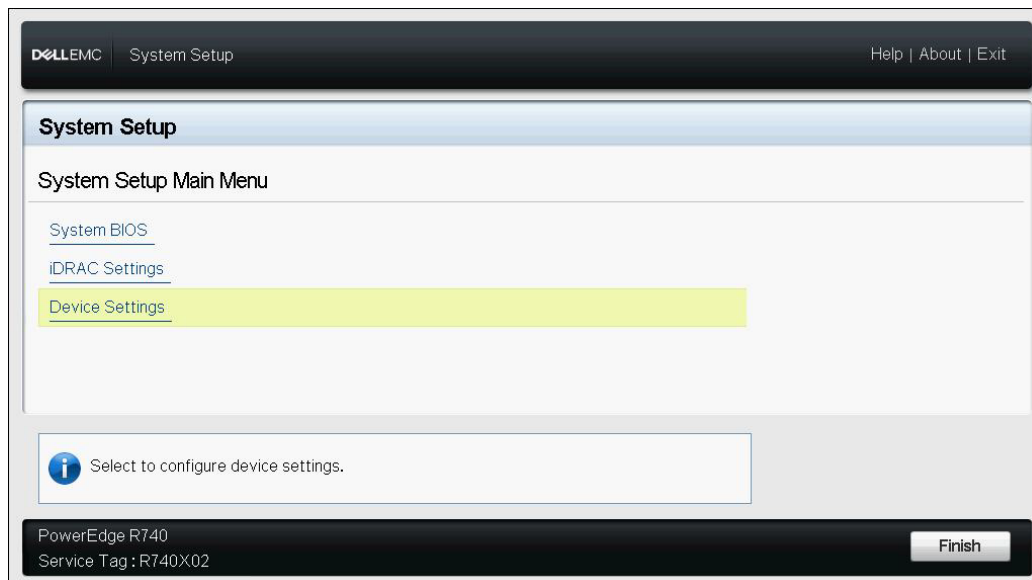


Figure 6-26. System Setup: Selecting Device Settings

NOTE

SAN boot is supported in the UEFI environment only. Make sure the system boot option is UEFI, and not legacy.

3. On the Device Settings page, select the Marvell FastLinQ adapter (Figure 6-27).

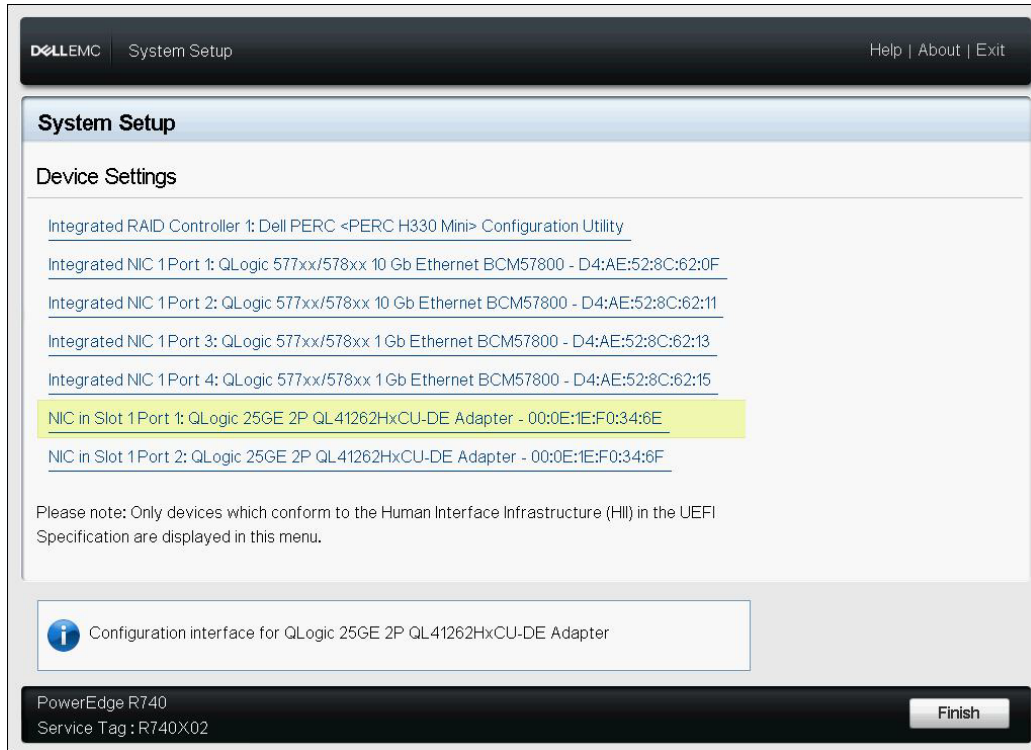


Figure 6-27. System Setup: Device Settings, Port Selection

4. On the Main Configuration Page, select **NIC Configuration** (Figure 6-28), and then press ENTER.

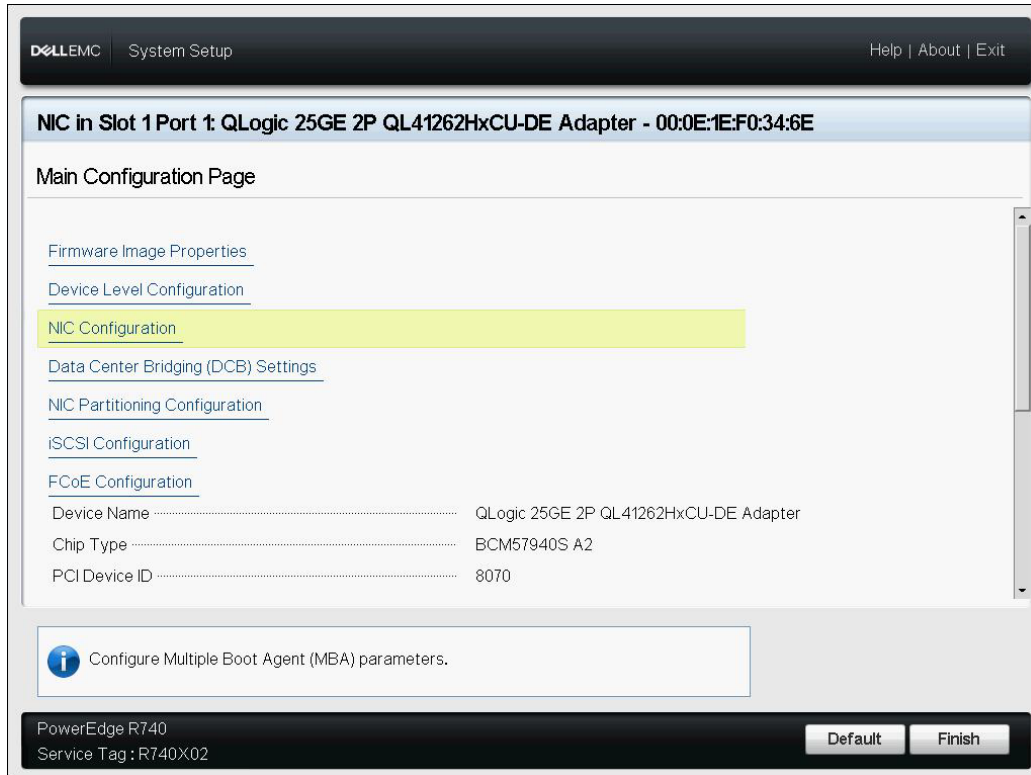


Figure 6-28. System Setup: NIC Configuration

5. On the NIC Configuration page, select **Boot Mode**, press ENTER, and then select **FCoE** as a preferred boot mode.

NOTE

FCoE is not listed as a boot option if the **FCoE Mode** feature is disabled at the port level. If the **Boot Mode** preferred is **FCoE**, make sure the **FCoE Mode** feature is enabled as shown in Figure 6-29 (and described in *Application Note, Enabling Storage Offload on Dell and Marvell FastLinQ 41000 Series Adapters*, available on the Marvell Web Site). Not all adapter versions support FCoE.

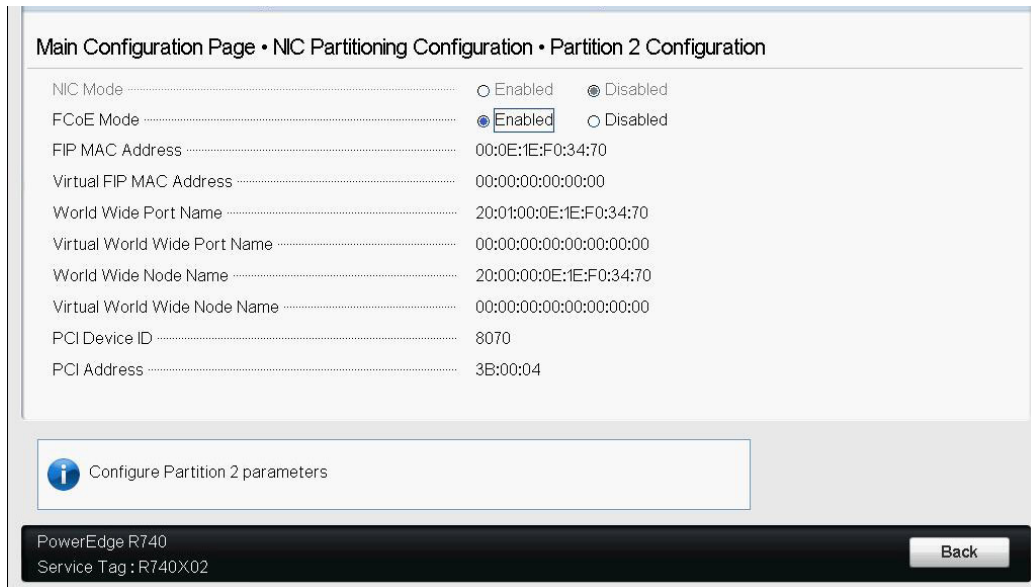


Figure 6-29. System Setup: FCoE Mode Enabled

To configure the FCoE boot parameters:

1. On the Device UEFI HII Main Configuration Page, select **FCoE Configuration**, and then press ENTER.
2. On the FCoE Configuration Page, select **FCoE General Parameters**, and then press ENTER.
3. On the FCoE General Parameters page ([Figure 6-30](#)), press the UP ARROW and DOWN ARROW keys to select a parameter, and then press ENTER to select and input the following values:
 - Fabric Discovery Retry Count:** Default value or as required
 - LUN Busy Retry Count:** Default value or as required

Main Configuration Page • FCoE Configuration • FCoE General Parameters

Fabric Discovery Retry Count 3

LUN Busy Retry Count 3

i Specify the retry count for FCoE fabric discovery. Value must be in range 0 to 60.

PowerEdge R740
Service Tag : R740X02

Back

Figure 6-30. System Setup: FCoE General Parameters

4. Return to the FCoE Configuration page.
5. Press ESC, and then select **FCoE Target Parameters**.
6. Press ENTER.
7. In the **FCoE General Parameters Menu**, enable **Connect** to the preferred FCoE target.
8. Type values for the following parameters ([Figure 6-31](#)) for the FCoE target, and then press ENTER:
 - World Wide Port Name Target n**
 - Boot LUN n**

Where the value of n is between 1 and 8, enabling you to configure 8 FCoE targets.

Main Configuration Page • FCoE Configuration

FCoE General Parameters

Virtual LAN ID 0

Connect 1 Enabled Disabled

World Wide Port Name Target 1 50:00:00:00:00:00:01

Boot LUN 1 1

Connect 2 Enabled Disabled

World Wide Port Name Target 2 50:00:00:00:00:00:02

Boot LUN 2 2

Connect 3 Enabled Disabled

World Wide Port Name Target 3 50:00:00:00:00:00:03

Configure general parameters that apply to all FCoE functionality.

PowerEdge R740
Service Tag : R740X02 Back

Figure 6-31. System Setup: FCoE General Parameters

Configuring FCoE Boot from SAN on Windows

FCoE boot from SAN information for Windows includes:

- [Windows Server 2016 and 2019/Azure Stack HCI FCoE Boot Installation](#)
- [Configuring FCoE on Windows](#)
- [FCoE Crash Dump on Windows](#)
- [Injecting \(Slipstreaming\) Adapter Drivers into Windows Image Files](#)

Windows Server 2016 and 2019/Azure Stack HCI FCoE Boot Installation

For Windows Server 2016 and 2019/Azure Stack HCI boot from SAN installation, Marvell requires the use of a “slipstream” DVD, or ISO image, with the latest Marvell drivers injected. See [“Injecting \(Slipstreaming\) Adapter Drivers into Windows Image Files”](#) on page 125.

The following procedure prepares the image for installation and booting in FCoE mode.

To set up Windows Server 2016 and 2019/Azure Stack HCI FCoE boot:

1. Remove any local hard drives on the system to be booted (remote system).
2. Prepare the Windows OS installation media by following the slipstreaming steps in [“Injecting \(Slipstreaming\) Adapter Drivers into Windows Image Files”](#) on page 125.

3. Load the latest Marvell FCoE boot images into the adapter NVRAM.
4. Configure the FCoE target to allow a connection from the remote device. Ensure that the target has sufficient disk space to hold the new OS installation.
5. Configure the UEFI HII to set the FCoE boot type on the required adapter port, correct initiator, and target parameters for FCoE boot.
6. Save the settings and reboot the system. The remote system should connect to the FCoE target, and then boot from the DVD-ROM device.
7. Boot from DVD and begin installation.
8. Follow the on-screen instructions.

On the window that shows the list of disks available for the installation, the FCoE target disk should be visible. This target is a disk connected through the FCoE boot protocol, located in the remote FCoE target.
9. To proceed with Windows Server 2016/2019/Azure Stack HCI installation, select **Next**, and then follow the on-screen instructions. The server will undergo a reboot multiple times as part of the installation process.
10. After the server boots to the OS, you should run the driver installer to complete the Marvell drivers and application installation.

Configuring FCoE on Windows

By default, DCB is enabled on 41000 FCoE- and DCB-compatible C-NICs. Marvell 41000 FCoE requires a DCB-enabled interface. For Windows operating systems, use QConvergeConsole GUI or a command line utility to configure the DCB parameters.

FCoE Crash Dump on Windows

Crash dump functionality is currently supported for FCoE boot for the FastLinQ 41000 Series Adapters.

No additional configuration is required for FCoE crash-dump generation when in FCoE boot mode.

Injecting (Slipstreaming) Adapter Drivers into Windows Image Files

To inject adapter drivers into the Windows image files:

1. Obtain the latest driver package for the applicable Windows Server version (2016 or 2019/Azure Stack HCI).
2. Extract the driver package to a working directory:
 - a. Open a command line session and navigate to the folder that contains the driver package.
 - b. To extract the driver Dell Update Package (DUP), issue the following command:

```
start /wait NameOfDup.exe /s /drivers=<folder path>
```
3. Download the Windows Assessment and Deployment Kit (ADK) version 10 from Microsoft:
<https://developer.microsoft.com/en-us/windows/hardware/windows-assessment-deployment-kit>
4. Follow the Microsoft “Add and Remove Drivers to an offline Windows Image article” and inject the OOB driver extracted on [Step 2](#), part [b](#). See <https://docs.microsoft.com/en-us/windows-hardware/manufacture/desktop/add-and-remove-drivers-to-an-offline-windows-image>

Configuring FCoE Boot from SAN on Linux

FCoE boot from SAN configuration for Linux covers the following:

- [Prerequisites for Linux FCoE Boot from SAN](#)
- [Configuring Linux FCoE Boot from SAN](#)

Prerequisites for Linux FCoE Boot from SAN

The following are required for Linux FCoE boot from SAN to function correctly with the Marvell FastLinQ 41000 10/25GbE Controller.

General

You no longer need to use the FCoE disk tabs in the Red Hat and SUSE installers because the FCoE interfaces are not exposed from the network interface and are automatically activated by the qedf driver.

SLES 12 and SLES 15

- Driver update disk is recommended for SLES 12 SP 3 and later.
- The installer parameter `dud=1` is required to ensure that the installer will ask for the driver update disk.

- Do not use the installer parameter `withfcoe=1` because the software FCoE will conflict with the hardware offload if network interfaces from `qede` are exposed.

Configuring Linux FCoE Boot from SAN

This section provides FCoE boot from SAN procedures for the following Linux distributions:

- [Configuring FCoE Boot from SAN for RHEL 7.4 and Later](#)
- [Configuring FCoE Boot from SAN for SLES 12 SP3 and Later](#)
- [Turning Off Ildpad](#)
- [Using an FCoE Boot Device as a kdump Target](#)

Configuring FCoE Boot from SAN for RHEL 7.4 and Later

To install RHEL 7.4 and later:

1. Boot from the RHEL 7.x installation media with the FCoE target already connected in UEFI.

```
Install Red Hat Enterprise Linux 7.x
Test this media & install Red Hat Enterprise 7.x
Troubleshooting -->
```

```
Use the UP and DOWN keys to change the selection
Press 'e' to edit the selected item or 'c' for a command
prompt
```

2. To install an out-of-box driver, press the E key. Otherwise, proceed to [Step 6](#).
3. Select the `kernel` line, and then press the E key to edit the line.
4. Issue the following command, and then press ENTER:

```
inst.dd modprobe.blacklist=qed,qede,qedr,qedi,qedf
```

5. The installation process prompts you to install the out-of-box driver as shown in [Figure 6-32](#).

```
Starting Driver Update Disk UI on tty1...
[ OK ] Started Show Plymouth Boot Screen.
[ OK ] Reached target Paths.
[ OK ] Reached target Basic System.
[ OK ] Started Device-Mapper Multipath Device Controller.
Starting Open-iSCSI...
[ OK ] Started Open-iSCSI.
Starting dracut initqueue hook...
[ OK ] Created slice system-driverxx2dupdates.slice.
Starting Driver Update Disk UI on tty1...
DD: starting interactive mode

(Page 1 of 1) Driver disk device selection
  /DEVICE  TYPE  LABEL  UUID
  1) sda1   ntfs   LABEL  1A90FE4090FE2245
  2) sda2   ufat   A6FF-80A4
  3) sda4   ntfs   7490015F900128E6
  4) sr0    iso9660 2017-07-11-01-39-24-00
# to select, 'r'-refresh, or 'c'-continue: r

(Page 1 of 1) Driver disk device selection
  /DEVICE  TYPE  LABEL  UUID
  1) sda1   ntfs   Recovery 1A90FE4090FE2245
  2) sda2   ufat   A6FF-80A4
  3) sda4   ntfs   7490015F900128E6
  4) sr0    iso9660 CDROM    2017-07-11-13-08-37-00
# to select, 'r'-refresh, or 'c'-continue: 4
DD: Examining /dev/sr0
mount: /dev/sr0 is write-protected, mounting read-only

(Page 1 of 1) Select drivers to install
  1) [ ] /media/DD-1/rpms/x86_64/kmod-qlgc-fastlinq-8.22.0.0-1.rhel17u4.x86_64.rpm
# to toggle selection, or 'c'-continue: 1

(Page 1 of 1) Select drivers to install
  1) [x] /media/DD-1/rpms/x86_64/kmod-qlgc-fastlinq-8.22.0.0-1.rhel17u4.x86_64.rpm
# to toggle selection, or 'c'-continue: c
DD: Extracting: kmod-qlgc-fastlinq

(Page 1 of 1) Driver disk device selection
  /DEVICE  TYPE  LABEL  UUID
  1) sda1   ntfs   Recovery 1A90FE4090FE2245
  2) sda2   ufat   A6FF-80A4
  3) sda4   ntfs   7490015F900128E6
  4) sr0    iso9660 CDROM    2017-07-11-13-08-37-00
# to select, 'r'-refresh, or 'c'-continue:
```

Figure 6-32. Prompt for Out-of-Box Installation

6. If required for your setup, load the FastLinQ driver update disk when prompted for additional driver disks. Otherwise, press the C key if you have no other driver update disks to install.
7. Continue with the installation. You can skip the media test. Click **Next** to continue with the installation.

8. In the Configuration window (Figure 6-33), select the language to use during the installation process, and then click **Continue**.

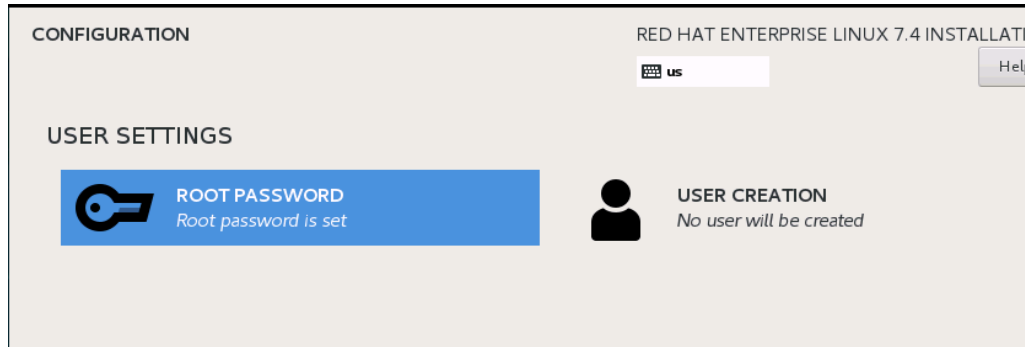


Figure 6-33. Red Hat Enterprise Linux 7.4 Configuration

9. In the Installation Summary window, click **Installation Destination**. The disk label is *sda*, indicating a single-path installation. If you configured multipath, the disk has a device mapper label.
10. In the **Specialized & Network Disks** section, select the FCoE LUN.
11. Type the root user's password, and then click **Next** to complete the installation.
12. During the first boot, add the following kernel command line to fall into shell:

```
rd.driver.pre=qed,qede,qedr,qedf,qedi
```
13. After a successful system boot, edit the `/etc/modprobe.d/anaconda-blacklist.conf` file to remove the blacklist entry for the selected driver.
14. Rebuild the ramdisk by issuing the `dracut -f` command, and then reboot.
15. Turn off the `lldpad` and `fcoe` services that are used for software FCoE. (If they are active, they can interfere with the typical operation of the hardware offload FCoE.) Issue the appropriate commands based on your operating system:
 - ❑ For RHEL 7 and SLES 15:

```
# systemctl stop fcoe
# systemctl stop lldpad
# systemctl disable fcoe
# systemctl disable lldpad
```

Configuring FCoE Boot from SAN for SLES 12 SP3 and Later

No additional steps, other than injecting DUD for out-of-box driver, are necessary to perform boot from SAN installations when using SLES 12 SP3.

Turning Off lldpad

After completing the installation and booting to the OS, turn off the lldpad and fcoe services that are used for software FCoE. (If they are active, they can interfere with the normal operation of the hardware offload FCoE.) Issue the following commands:

```
# service fcoe stop
# service lldpad stop
# chkconfig fcoe off
# chkconfig lldpad off
```

Using an FCoE Boot Device as a kdump Target

When using a device exposed using the qedf driver as a kdump target for crash dumps, Marvell recommends that you specify the kdump `crashkernel` memory parameter at the kernel command line to be a minimum of 512MB. Otherwise, the kernel crash dump may not succeed.

For details on how to set the kdump `crashkernel` size, refer to your Linux distribution documentation.

Configuring FCoE Boot from SAN on VMware

For VMware ESXi boot from SAN installation, Marvell requires that you use a customized ESXi ISO image that is built with the latest Marvell Converged Network Adapter bundle injected. This section covers the following VMware FCoE boot from SAN procedures.

- [Injecting \(Slipstreaming\) ESXi Adapter Drivers into Image Files](#)
- [Installing the Customized ESXi ISO](#)

Injecting (Slipstreaming) ESXi Adapter Drivers into Image Files

This procedure uses ESXi-Customizer tool v2.7.2 as an example, but you can use any ESXi customizer.

To inject adapter drivers into an ESXi image file:

1. Download ESXi-Customizer v2.7.2 or later.
2. Go to the `ESXi customizer` directory.
3. Issue the `ESXi-Customizer.cmd` command.

4. In the ESXi-Customizer dialog box, click **Browse** to complete the following.
 - a. Select the original VMware ESXi ISO file.
 - b. Select either the Marvell FCoE driver VIB file or the Marvell offline qedentv bundle ZIP file.
 - c. For the working directory, select the folder in which the customized ISO needs to be created.
 - d. Click **Run**.

Figure 6-34 shows an example.

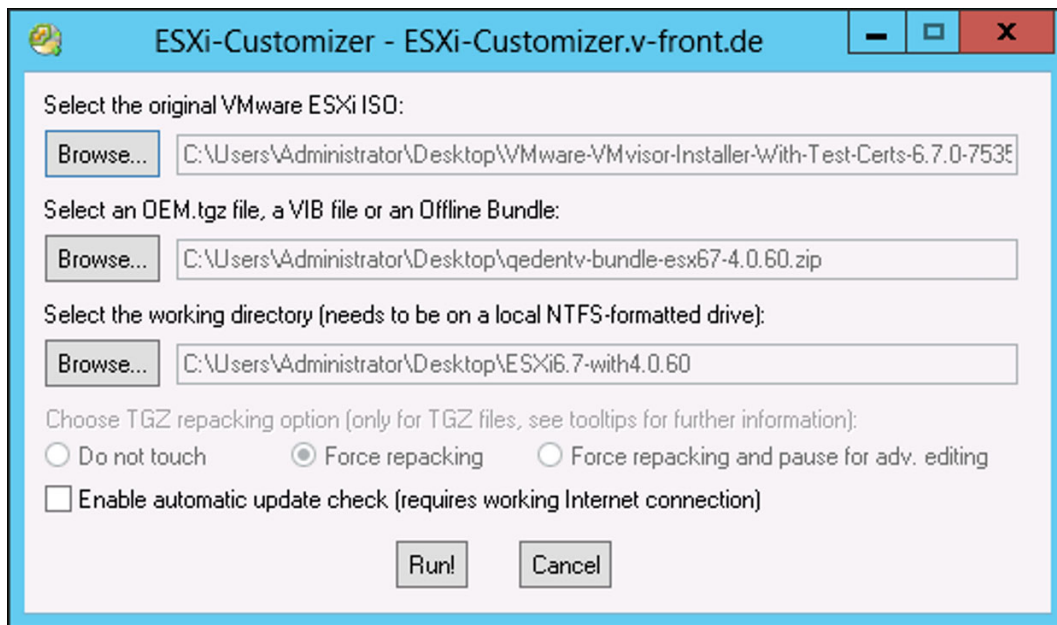


Figure 6-34. ESXi-Customizer Dialog Box

5. Burn a DVD that contains the customized ISO build located in the working directory specified in [Step 4c](#).
6. Use the new DVD to install the ESXi OS.

Installing the Customized ESXi ISO

1. Load the latest Marvell FCOE boot images into the adapter NVRAM.
2. Configure the FCOE target to allow a valid connection with the remote machine. Ensure that the target has sufficient free disk space to hold the new OS installation.
3. Configure the UEFI HII to set the FCOE boot type on the required adapter port, the correct initiator, and the target parameters for FCOE boot.

4. Save the settings and reboot the system.
The initiator should connect to an FCOE target and then boot the system from the DVD-ROM device.
5. Boot from the DVD and begin installation.
6. Follow the on-screen instructions.

On the window that shows the list of disks available for the installation, the FCOE target disk should be visible because the injected Converged Network Adapter bundle is inside the customized ESXi ISO. [Figure 6-35](#) shows an example.

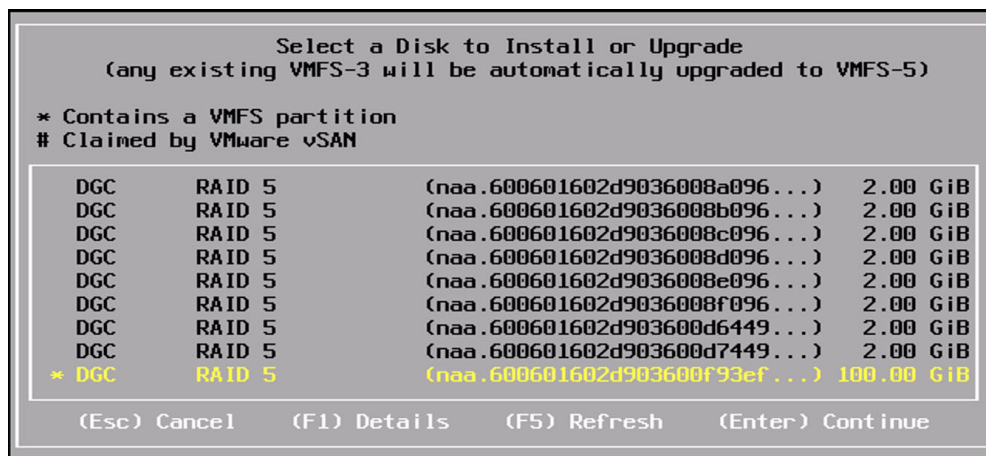


Figure 6-35. Select a VMware Disk to Install

7. Select the LUN on which ESXi can install, and then press ENTER.
8. On the next window, click **Next**, and then follow the on-screen instructions.
9. When installation completes, reboot the server and eject the DVD.
10. During the server boot, press the F9 key to access the **One-Time Boot Menu**, and then select **Boot media to QLogic adapter port**.
11. Under **Boot Menu**, select the newly installed ESXi to load through boot from SAN.

Figure 6-36 provides two examples.

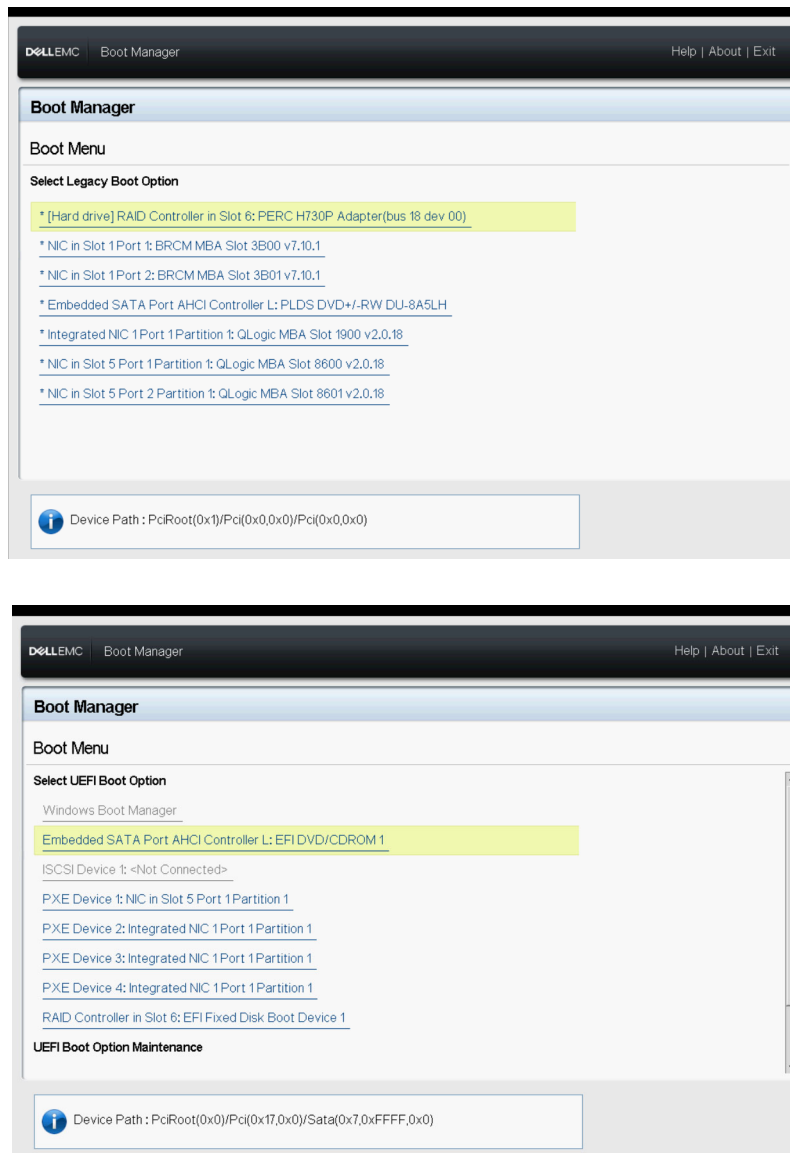


Figure 6-36. VMware USB Boot Options

Viewing Boot Statistics

To view boot statistics for PXE, iSCSI, and FCoE, on the Main Configuration Page, click Boot Session Information. [Figure 6-37](#) shows an example.



Figure 6-37. Boot Session Information

7 RoCE Configuration

This chapter describes RDMA over converged Ethernet (RoCE v1 and v2) configuration on the 41000 Series Adapter, the Ethernet switch, and the Windows, Linux, or VMware host, including:

- [Supported Operating Systems and OFED](#)
- [“Planning for RoCE” on page 135](#)
- [“Preparing the Adapter” on page 136](#)
- [“Preparing the Ethernet Switch” on page 136](#)
- [“Configuring RoCE on the Adapter for Windows Server” on page 140](#)
- [“Configuring RoCE on the Adapter for Linux” on page 158](#)
- [“Configuring RoCE on the Adapter for VMware ESX” on page 173](#)
- [“Configuring DCQCN” on page 180](#)

NOTE

Some RoCE features may not be fully enabled in the current release.

Supported Operating Systems and OFED

[Table 7-1](#) shows the operating system support for RoCE v1, RoCE v2, iWARP, and iSER. OpenFabrics Enterprise Distribution (OFED) is not supported for any OS.

Table 7-1. OS Support for RoCE v1, RoCE v2, iWARP and iSER

Operating System	Inbox
Windows Server 2016	No
Windows Server	RoCE v1, RoCE v2, iWARP
2019/Azure Stack HCI	RoCE v1, RoCE v2, iWARP
RHEL 7.8	RoCE v1, RoCE v2, iWARP, iSER, NVMe-oF
RHEL 7.9	RoCE v1, RoCE v2, iWARP, iSER, NVMe-oF

Table 7-1. OS Support for RoCE v1, RoCE v2, iWARP and iSER (Continued)

Operating System	Inbox
RHEL 8.2	RoCE v1, RoCE v2, iWARP, iSER, NVMe-oF
RHEL 8.3	RoCE v1, RoCE v2, iWARP, iSER, NVMe-oF
SLES 15 SP1	RoCE v1, RoCE v2, iWARP, iSER, NVMe-oF
SLES 15 SP2	RoCE v1, RoCE v2, iWARP, iSER, NVMe-oF
VMware ESXi 6.7 U3	RoCE v1, RoCE v2
VMware ESXi 7.0 U1	RoCE v1, RoCE v2

Planning for RoCE

As you prepare to implement RoCE, consider the following limitations:

- If you are using the inbox OFED, the operating system should be the same on the server and client systems. Some of the applications may work between different operating systems, but there is no guarantee. This is an OFED limitation.
- For OFED applications (most often perftest applications), server and client applications should use the same options and values. Problems can arise if the operating system and the perftest application have different versions. To confirm the perftest version, issue the following command:

```
# ib_send_bw --version
```
- Building libqedr in inbox OFED for older distributions (RHEL 7.4 and earlier) requires installing libibverbs-devel. For later versions of RHEL, use the inbox rdma-core of the distribution.
- Running user space applications in inbox OFED requires installing the InfiniBand® Support group, by yum groupinstall “InfiniBand Support” that contains libibcm, libibverbs, and more.
- OFED and RDMA applications that depend on libibverbs also require the Marvell RDMA user space library, libqedr, which is available in inbox in all relevant distributions as part of the rdma-core. Install libqedr using the libqedr RPM or source packages.
- View the current RDMA statistics and counters under debugfs:

```
/sys/kernel/debug/qedr
```
- RoCE supports only little endian.

Preparing the Adapter

Follow these steps to enable DCBX and specify the RoCE priority using the HII management application. For information about the HII application, see [Chapter 5 Adapter Preboot Configuration](#).

These steps cause the device to add the DCBX application `tlvs` (for RoCE/RoCEv2) to its suggested DCBX configuration when negotiating with the switch. This application works for persistent out-of-the-box DCBX with no additional configuration required. However, it limits RoCE to use a single DCBX priority for RoCE. If multiple DCB priorities are needed for RoCE, use the dynamic configuration from the host through `debugfs_conf.sh`, `debugfs_dump.sh`, and `debugfs_edit.sh` (all available in the `add-ons` folder in the FastLinQ package).

To prepare the adapter:

1. On the Main Configuration Page, select **Data Center Bridging (DCB) Settings**, and then click **Finish**.
2. In the Data Center Bridging (DCB) Settings window, click the **DCBX Protocol** option. The 41000 Series Adapter supports both CEE and IEEE protocols. This value should match the corresponding value on the DCB switch. In this example, select **CEE** or **Dynamic**.
3. In the **RoCE Priority** box, type a priority value. This value should match the corresponding value on the DCB switch. In this example, type `5`. Typically, `0` is used for the default lossy traffic class, `3` is used for the FCoE traffic class, and `4` is used for lossless iSCSI-TLV over DCB traffic class.
4. Click **Back**.
5. When prompted, click **Yes** to save the changes. Changes will take effect after a system reset.

For Windows, you can configure DCBX using the HII or QoS method. The configuration shown in this section is through HII. For QoS, refer to [“Configuring QoS for RoCE” on page 276](#).

Preparing the Ethernet Switch

This section describes how to configure a Cisco® Nexus® 6000 Ethernet Switch and a Dell Z9100 Ethernet Switch for RoCE.

- [Configuring the Cisco Nexus 6000 Ethernet Switch](#)
- [Configuring the Dell Z9100 Ethernet Switch for RoCE](#)

Configuring the Cisco Nexus 6000 Ethernet Switch

Steps for configuring the Cisco Nexus 6000 Ethernet Switch for RoCE include configuring class maps, configuring policy maps, applying the policy, and assigning a vLAN ID to the switch port.

To configure the Cisco switch:

1. Open a config terminal session as follows:

```
Switch# config terminal view  
switch(config)#
```
2. Configure the quality of service (QoS) class map and set the RoCE priority (cos) to match the adapter (5) as follows:

```
switch(config)# class-map type qos class-roce  
switch(config)# match cos 5
```
3. Configure queuing class maps as follows:

```
switch(config)# class-map type queuing class-roce  
switch(config)# match qos-group 3
```
4. Configure network QoS class maps as follows:

```
switch(config)# class-map type network-qos class-roce  
switch(config)# match qos-group 3
```
5. Configure QoS policy maps as follows:

```
switch(config)# policy-map type qos roce  
switch(config)# class type qos class-roce  
switch(config)# set qos-group 3
```
6. Configure queuing policy maps to assign network bandwidth. In this example, use a value of 50 percent:

```
switch(config)# policy-map type queuing roce  
switch(config)# class type queuing class-roce  
switch(config)# bandwidth percent 50
```
7. Configure network QoS policy maps to set priority flow control for no-drop traffic class as follows:

```
switch(config)# policy-map type network-qos roce  
switch(config)# class type network-qos class-roce  
switch(config)# pause no-drop
```
8. Apply the new policy at the system level as follows:

```
switch(config)# system qos  
switch(config)# service-policy type qos input roce
```

```
switch(config)# service-policy type queuing output roce
switch(config)# service-policy type queuing input roce
switch(config)# service-policy type network-qos roce
```

9. Assign a vLAN ID to the switch port to match the vLAN ID assigned to the adapter (5).

```
switch(config)# interface ethernet x/x
switch(config)# switchport mode trunk
switch(config)# switchport trunk allowed vlan 1,5
```

Configuring the Dell Z9100 Ethernet Switch for RoCE

Configuring the Dell Z9100 Ethernet Switch for RoCE comprises configuring a DCB map for RoCE, configuring priority-based flow control (PFC) and enhanced transmission selection (ETS), verifying the DCB map, applying the DCB map to the port, verifying PFC and ETS on the port, specifying the DCB protocol, and assigning a VLAN ID to the switch port.

NOTE

For instructions on configuring a Dell Z91000 switch port to connect to the 41000 Series Adapter at 25Gbps, see [“Dell Z9100 Switch Configuration” on page 318](#).

To configure the Dell switch:

1. Create a DCB map.

```
Dell# configure
Dell(conf)# dcb-map roce
Dell(conf-dcbmap-roce)#
```

2. Configure two ETS traffic classes in the DCB map with 50 percent bandwidth assigned for RoCE (group 1).

```
Dell(conf-dcbmap-roce)# priority-group 0 bandwidth 50 pfc off
Dell(conf-dcbmap-roce)# priority-group 1 bandwidth 50 pfc on
```

3. Configure the PFC priority to match the adapter traffic class priority number (5).

```
Dell(conf-dcbmap-roce)# priority-pgid 0 0 0 0 0 1 0 0
```

4. Verify the DCB map configuration priority group.

```
Dell(conf-dcbmap-roce)# do show qos dcb-map roce
-----
State      :Complete
PfcMode    :ON
```


7-RoCE Configuration

Preparing the Ethernet Switch

```
-----  
PG:0 TSA:ETS BW:40 PFC:OFF  
Priorities:0 1 2 3 4 6 7
```

```
PG:1 TSA:ETS BW:60 PFC:ON  
Priorities:5
```

5. Apply the DCB map to the port.

```
Dell(conf)# interface twentyFiveGigE 1/8/1  
Dell(conf-if-tf-1/8/1)# dcb-map roce
```

6. Verify the ETS and PFC configuration on the port. The following examples show summarized interface information for ETS and detailed interface information for PFC.

```
Dell(conf-if-tf-1/8/1)# do show interfaces twentyFiveGigE 1/8/1 ets summary
```

```
Interface twentyFiveGigE 1/8/1
```

```
Max Supported TC is 4
```

```
Number of Traffic Classes is 8
```

```
Admin mode is on
```

```
Admin Parameters :
```

```
-----  
Admin is enabled
```

PG-grp	Priority#	BW-%	BW-COMMITTED	BW-PEAK	TSA
	%	Rate (Mbps)	Burst (KB)	Rate (Mbps)	Burst (KB)

0	0,1,2,3,4,6,7	40	-	-	ETS
1	5	60	-	-	ETS
2		-	-	-	-
3		-	-	-	-

```
Dell(Conf)# do show interfaces twentyFiveGigE 1/8/1 pfc detail
```

```
Interface twentyFiveGigE 1/8/1
```

```
Admin mode is on
```

```
Admin is enabled, Priority list is 5
```

```
Remote is enabled, Priority list is 5
```

```
Remote Willing Status is enabled
```

```
Local is enabled, Priority list is 5
```

```
Oper status is init
```

```
PFC DCBX Oper status is Up
```

```
State Machine Type is Feature
```

```
TLV Tx Status is enabled
PFC Link Delay 65535 pause quntams
Application Priority TLV Parameters :
-----
FCOE TLV Tx Status is disabled
ISCSI TLV Tx Status is enabled
Local FCOE PriorityMap is 0x0
Local ISCSI PriorityMap is 0x20
Remote ISCSI PriorityMap is 0x200
```

```
66 Input TLV pkts, 99 Output TLV pkts, 0 Error pkts, 0 Pause Tx pkts, 0 Pause
Rx pkts
```

```
66 Input Appln Priority TLV pkts, 99 Output Appln Priority TLV pkts, 0 Error
Appln Priority TLV Pkts
```

7. Configure the DCBX protocol (CEE in this example).

```
Dell(conf)# interface twentyFiveGigE 1/8/1
Dell(conf-if-tf-1/8/1)# protocol lldp
Dell(conf-if-tf-1/8/1-lldp)# dcbx version cee
```

8. Assign a VLAN ID to the switch port to match the VLAN ID assigned to the adapter (5).

```
Dell(conf)# interface vlan 5
Dell(conf-if-vl-5)# tagged twentyFiveGigE 1/8/1
```

NOTE

The VLAN ID need not be the same as the RoCE traffic class priority number. However, using the same number makes configurations easier to understand.

Configuring RoCE on the Adapter for Windows Server

Configuring RoCE on the adapter for Windows Server host comprises enabling RoCE on the adapter and verifying the Network Direct MTU size.

To configure RoCE on a Windows Server host:

1. Enable RoCE on the adapter.
 - a. Open the Windows Device Manager, and then open the 41000 Series Adapters NDIS Miniport Properties.
 - b. On the QLogic FastLinQ Adapter Properties, click the **Advanced** tab.

- c. On the Advanced page, configure the properties listed in [Table 7-2](#) by selecting each item under **Property** and choosing an appropriate **Value** for that item. Then click **OK**.

Table 7-2. Advanced Properties for RoCE

Property	Value or Description
NetworkDirect Functionality	Enabled
Network Direct Mtu Size	<p>The network direct MTU size must be less than the jumbo packet size.</p> <p>For optimal performance, set this (per physical function) RoCE MTU size to 4,096 when the (per physical function) L2 Ethernet MTU size is set to a value larger than 4,096. Additionally, set the network and target ports to an equivalent MTU size (to prevent the packets from fragmenting or dropping).</p> <p>On Windows, this RoCE MTU value is not automatically changed when the L2 Ethernet MTU value is changed.</p>
Quality of Service	<p>For RoCE v1/v2, always select Enabled to allow Windows DCB-QoS service to control and monitor DCB. For more information, see “Configuring QoS by Disabling DCBX on the Adapter” on page 276 and “Configuring QoS by Enabling DCBX on the Adapter” on page 281.</p>
NetworkDirect Technology	RoCE or RoCE v2.
VLAN ID	Assign any vLAN ID to the interface. The value must be the same as is assigned on the switch.

Figure 7-1 shows an example of configuring a property value.

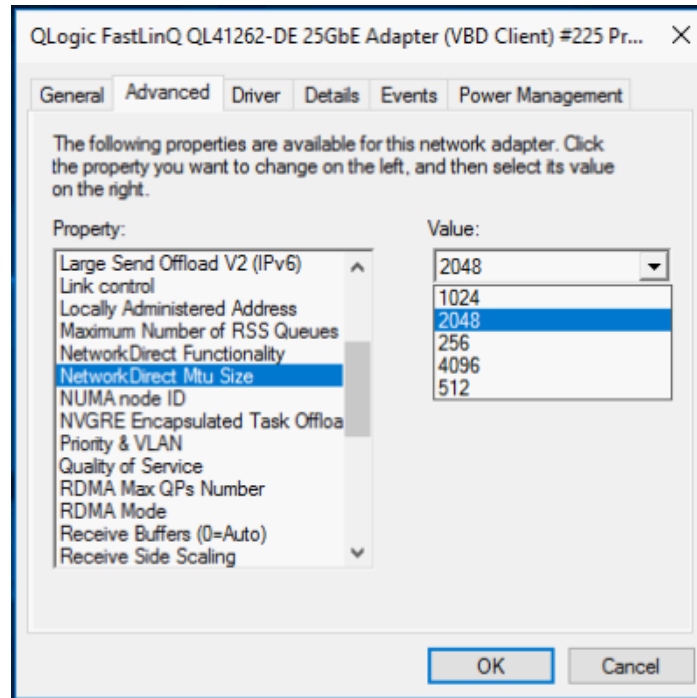


Figure 7-1. Configuring RoCE Properties

- Using Windows PowerShell, verify that RDMA is enabled on the adapter. The `Get-NetAdapterRdma` command lists the adapters that support RDMA—both ports are enabled.

NOTE

If you are configuring RoCE over Hyper-V, do not assign a vLAN ID to the physical interface.

```
PS C:\Users\Administrator> Get-NetAdapterRdma
Name                               InterfaceDescription           Enabled
----                               -
SLOT 4 3 Port 1                    QLogic FastLinQ QL41262...     True
SLOT 4 3 Port 2                    QLogic FastLinQ QL41262...     True
```

- Using Windows PowerShell, verify that `NetworkDirect` is enabled on the host operating system. The `Get-NetOffloadGlobalSetting` command shows `NetworkDirect` is enabled.

```
PS C:\Users\Administrators> Get-NetOffloadGlobalSetting
ReceiveSideScaling                  : Enabled
```

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

4. Connect a server message block (SMB) drive (see “[Mapping the SMB Drive](#)” on page 271), run RoCE traffic, and verify the results.

To set up and connect to an SMB drive, view the information available online from Microsoft:

[https://technet.microsoft.com/en-us/library/hh831795\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/hh831795(v=ws.11).aspx)

5. By default, Microsoft's SMB Direct establishes two RDMA connections per port, which provides good performance, including line rate at a higher block size (for example, 64KB). To optimize performance, you can change the quantity of RDMA connections per RDMA interface to four (or greater).

To increase the quantity of RDMA connections to four (or more), issue the following command in Windows PowerShell:

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

Viewing RDMA Counters

The following procedure also applies to iWARP.

To view RDMA counters for RoCE:

1. Launch Performance Monitor.
2. Open the Add Counters dialog box. [Figure 7-2](#) shows an example.

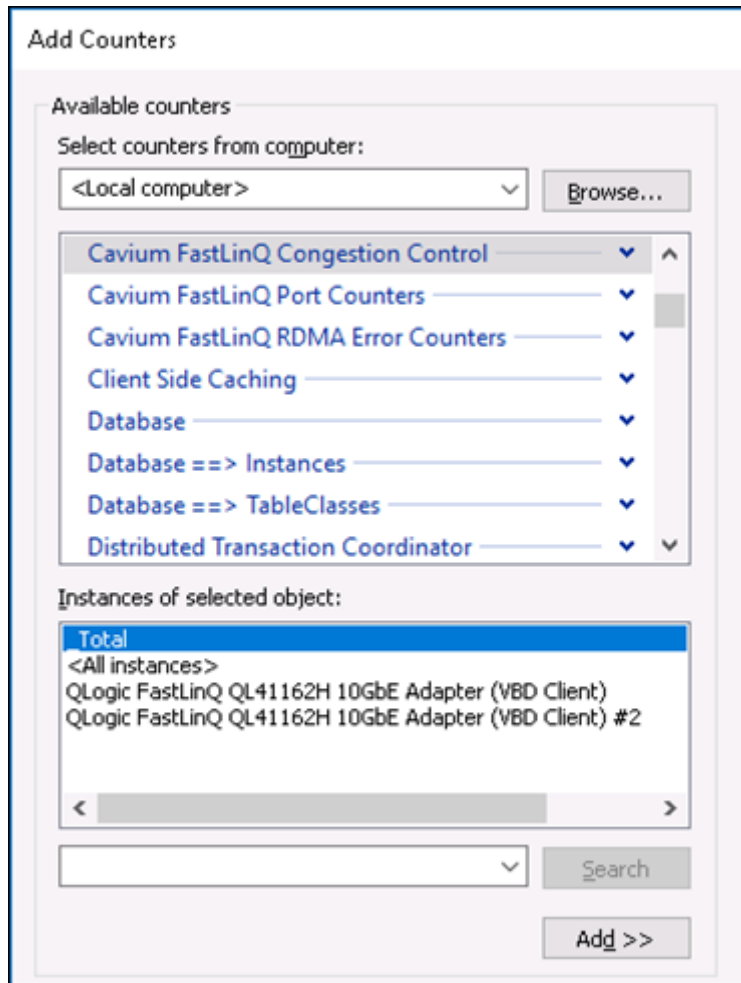


Figure 7-2. Add Counters Dialog Box

NOTE

If Marvell RDMA counters are not listed in the Performance Monitor Add Counters dialog box, manually add them by issuing the following command from the driver location:

```
Lodctr /M:qend.man
```

3. Select one of the following counter types:
 - Cavium FastLinQ Congestion Control:**
 - Increment when there is congestion in the network and ECN is enabled on the switch.
 - Describe RoCE v2 ECN Marked Packets and Congestion Notification Packets (CNPs) sent and received successfully.
 - Apply only to RoCE v2.
 - Cavium FastLinQ Port Counters:**
 - Increment when there is congestion in the network.
 - Pause counters increment when flow control or global pause is configured and there is a congestion in the network.
 - PFC counters increment when priority flow control is configured and there is a congestion in the network.
 - Cavium FastLinQ RDMA Error Counters:**
 - Increment if any error occurs in transport operations.
 - For details, see [Table 7-3](#).
4. Under **Instances of selected object**, select **Total**, and then click **Add**.

Figure 7-3 shows three examples of the counter monitoring output.

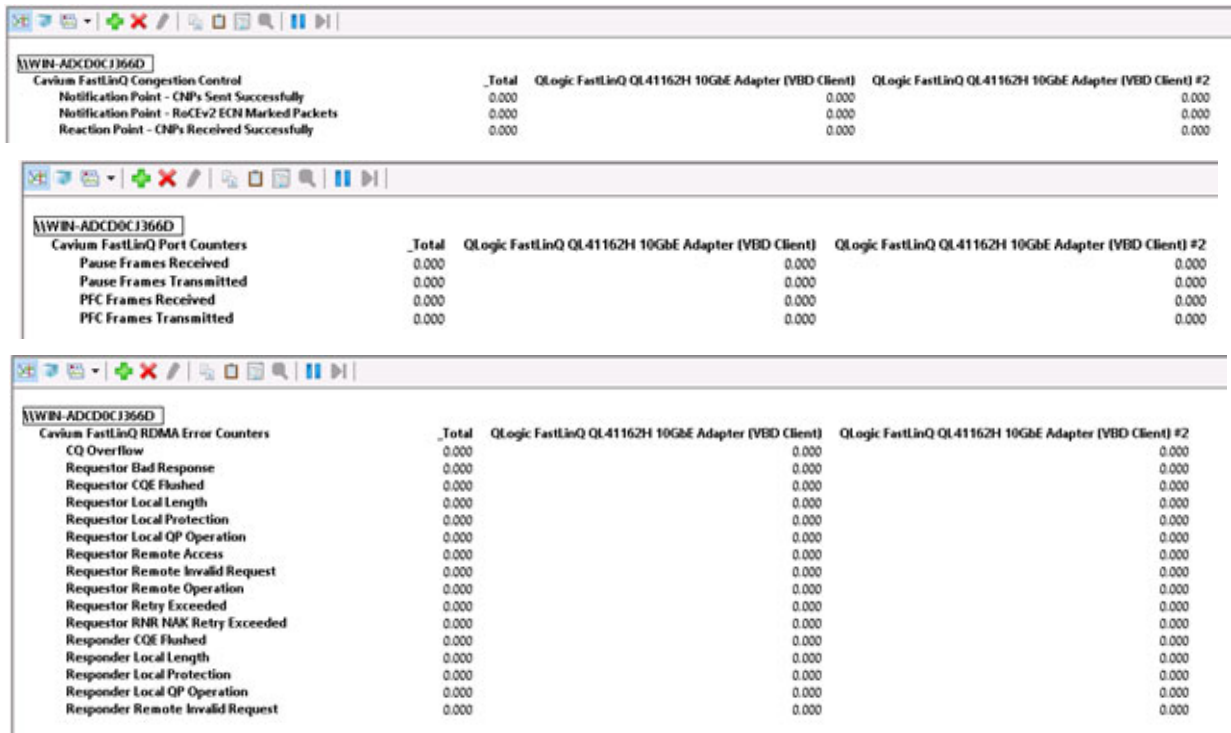


Figure 7-3. Performance Monitor: 41000 Series Adapters’ Counters

Table 7-3 provides details about error counters.

Table 7-3. Marvell FastLinQ RDMA Error Counters

RDMA Error Counter	Description	Applies to RoCE?	Applies to iWARP?	Troubleshooting
CQ overflow	A completion queue on which an RDMA work request is posted. This counter specifies the quantity of instances where there was a completion for a work request on the send or receive queue, but no space on the associated completion queue.	Yes	Yes	Indicates a software design issue causing an insufficient completion queue size.
Requestor Bad response	A malformed response was returned by the responder.	Yes	Yes	—

Table 7-3. Marvell FastLinQ RDMA Error Counters (Continued)

RDMA Error Counter	Description	Applies to RoCE?	Applies to iWARP?	Troubleshooting
Requestor CQEs flushed with error	Posted work requests may be flushed by sending completions with a flush status to the completion queue (CQ) (without completing the actual execution of the work request) if the queue pair (QP) moves to an error state for any reason and pending work requests exist. If a work request completed with error status, all other pending work requests for that QP are flushed.	Yes	Yes	Occurs when the RDMA connection is down.
Requestor Local length	The RDMA Read response message contained too much or too little payload data.	Yes	Yes	Usually indicates an issue with the host software components.
Requestor local protection	The locally posted work request's data segment does not reference a memory region that is valid for the requested operation.	Yes	Yes	Usually indicates an issue with the host software components.
Requestor local QP operation	An internal QP consistency error was detected while processing this work request.	Yes	Yes	—
Requestor Remote access	A protection error occurred on a remote data buffer to be read by an RDMA Read, written by an RDMA Write, or accessed by an atomic operation.	Yes	Yes	—
Requestor Remote Invalid request	The remote side received an invalid message on the channel. The invalid request may have been a Send message or an RDMA request.	Yes	Yes	Possible causes include the operation is not supported by this receive queue, insufficient buffering to receive a new RDMA or atomic operation request, or the length specified in an RDMA request is greater than 231 bytes.

Table 7-3. Marvell FastLinQ RDMA Error Counters (Continued)

RDMA Error Counter	Description	Applies to RoCE?	Applies to iWARP?	Troubleshooting
Requestor remote operation	Remote side could not complete the operation requested due to a local issue.	Yes	Yes	A software issue at the remote side (for example, one that caused a QP error or a malformed work queue element (WQE) on the receive queue (RQ) prevented operation completion.
Requestor retry exceeded	Transport retries have exceeded the maximum limit.	Yes	Yes	The remote peer may have stopped responding, or a network issue is preventing messages acknowledgment.
Requestor RNR Retries exceeded	Retry due to reset not require (RNR) no acknowledgment (NAK) received have been tried the maximum number of times without success.	Yes	No	The remote peer may have stopped responding, or a network issue is preventing messages acknowledgment.
Responder CQE flushed	Posted work requests (receive buffers on RQ) may be flushed by sending completions with a flush status to the CQ if the QP moves to an error state for any reason, and pending receive buffers exist on the RQ. If a work request completed with an error status, all other pending work requests for that QP are flushed.	Yes	Yes	This counter may increment at the end of an SMB direct tear-down session, when the RDMA receiver needs to post unused receive WQEs. Under some high bandwidth conditions, more receive WQEs than actually needed could be created by the application. Then during the application tear-down phase, any unused receive WQEs will be reported as completion with an error, to indicate that the hardware does not plan to use them. This error, by itself, is not an error indication.

Table 7-3. Marvell FastLinQ RDMA Error Counters (Continued)

RDMA Error Counter	Description	Applies to RoCE?	Applies to iWARP?	Troubleshooting
Responder local length	Invalid length in inbound messages.	Yes	Yes	Misbehaving remote peer. For example, the inbound send messages have lengths greater than the receive buffer size.
Responder local protection	The locally posted work request's data segment does not reference a memory region that is valid for the requested operation.	Yes	Yes	Indicates a software issue with memory management.
Responder Local QP Operation error	An internal QP consistency error was detected while processing this work request.	Yes	Yes	Indicates a software issue.
Responder remote invalid request	The responder detected an invalid inbound message on the channel.	Yes	Yes	Indicates possible misbehavior by a remote peer. Possible causes include: the operation is not supported by this receive queue, insufficient buffering to receive a new RDMA request, or the length specified in an RDMA request is greater than 2 ³¹ bytes.

Configuring RoCE for SR-IOV VF Devices (VF RDMA)

The following sections describe how to configure RoCE for SR-IOV VF devices (also referred to as *VF RDMA*). Associated information and limitations are also provided.

Configuration Instructions

To configure VF RDMA:

1. Install the VF RDMA capable components (drivers, firmware, multiboot image (MBI)).
2. Configure QoS for VF RDMA.

QoS configuration is needed to configure priority flow control (PFC) for RDMA. Configure QoS in the host as documented in [“Configuring QoS for RoCE” on page 276](#). (QoS configuration must be done in the host, not in the VM).

3. Configure Windows Hyper-V for VF RDMA:
 - a. Enable SR-IOV in HII and on the Advanced tab in Windows Device Manager.
 - b. Open the **Windows Hyper-V Manager** on the host.
 - c. Open the **Virtual Switch Manager** from the right pane.
 - d. Select **New Virtual Network switch** with type **External**.

[Figure 7-4](#) shows an example.

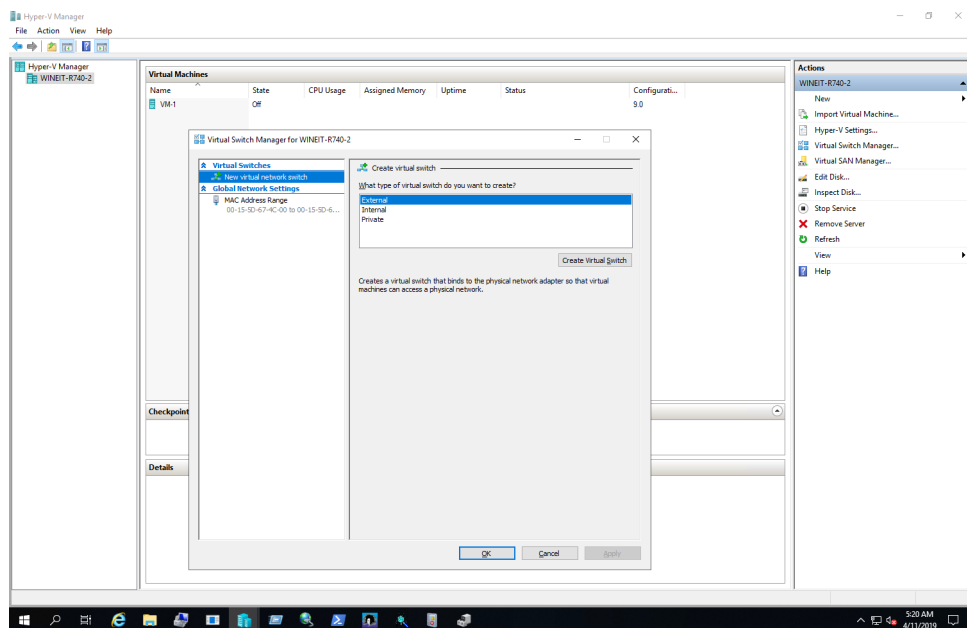


Figure 7-4. Setting an External New Virtual Network Switch

- e. Click the **External network** button, and then select the appropriate adapter. Click **Enable single-root I/O virtualization (SR-IOV)**.

Figure 7-5 shows an example.

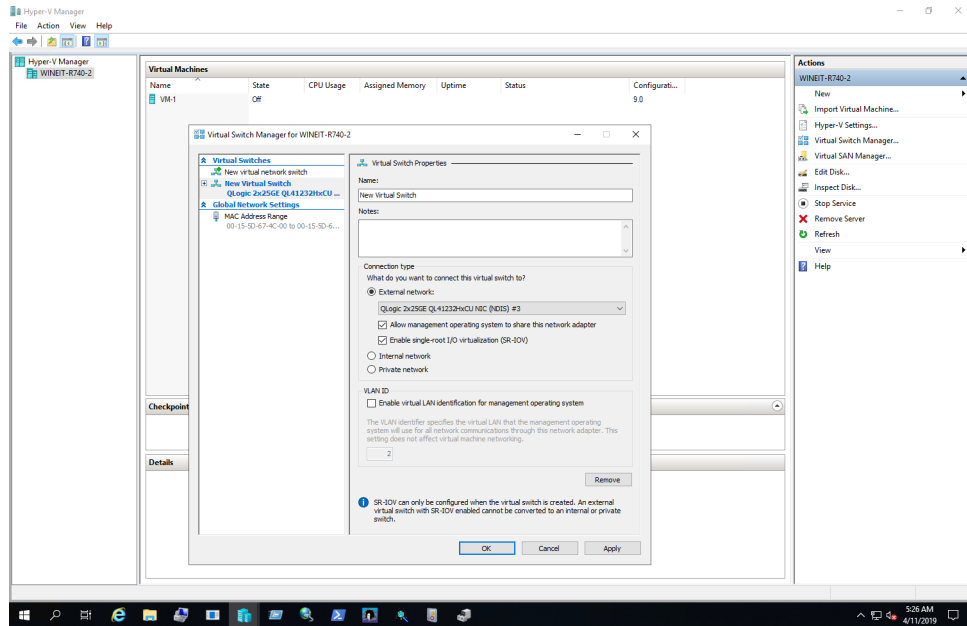


Figure 7-5. Setting SR-IOV for New Virtual Switch

- f. Create a VM and open the VM settings.
Figure 7-6 shows an example.

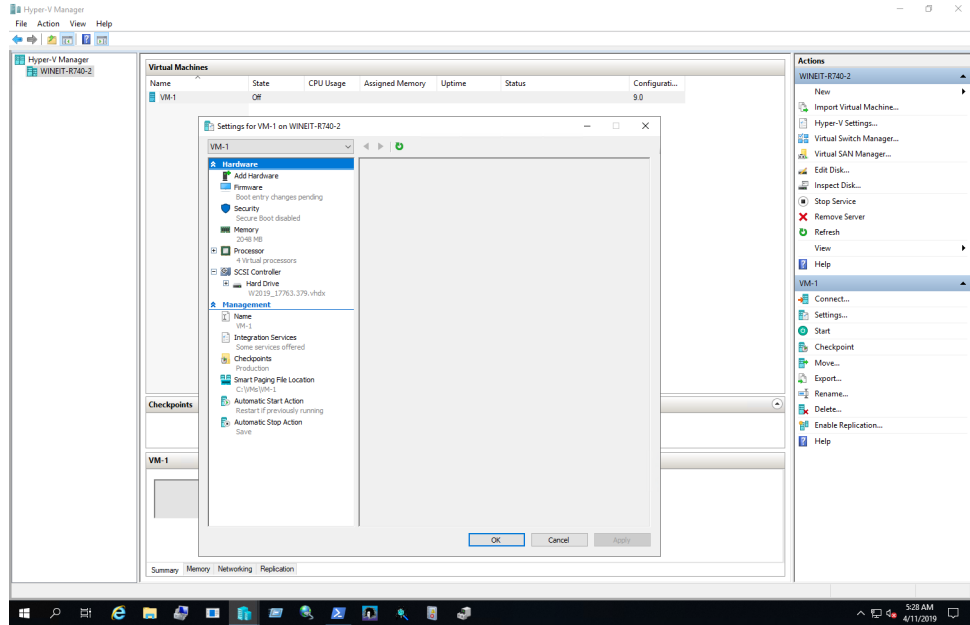


Figure 7-6. VM Settings

- g. Select **Add Hardware**, and then select **Network Adapter** to assign the virtual network adapters (VMNICs) to the VM.
- h. Select the newly created virtual switch.

- i. Enable VLAN to the network adapter.
Figure 7-7 shows an example.

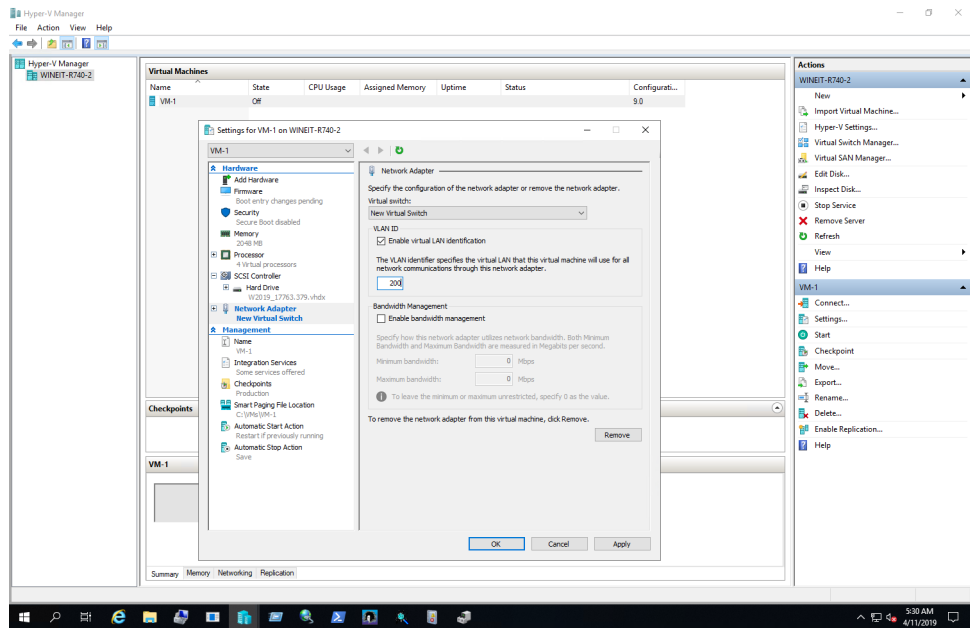


Figure 7-7. Enabling VLAN to the Network Adapter

- j. Expand the network adapter settings. Under Single-root I/O virtualization, select **Enable SR-IOV** to enable SR-IOV capabilities for the VMNIC.

Figure 7-8 shows an example.

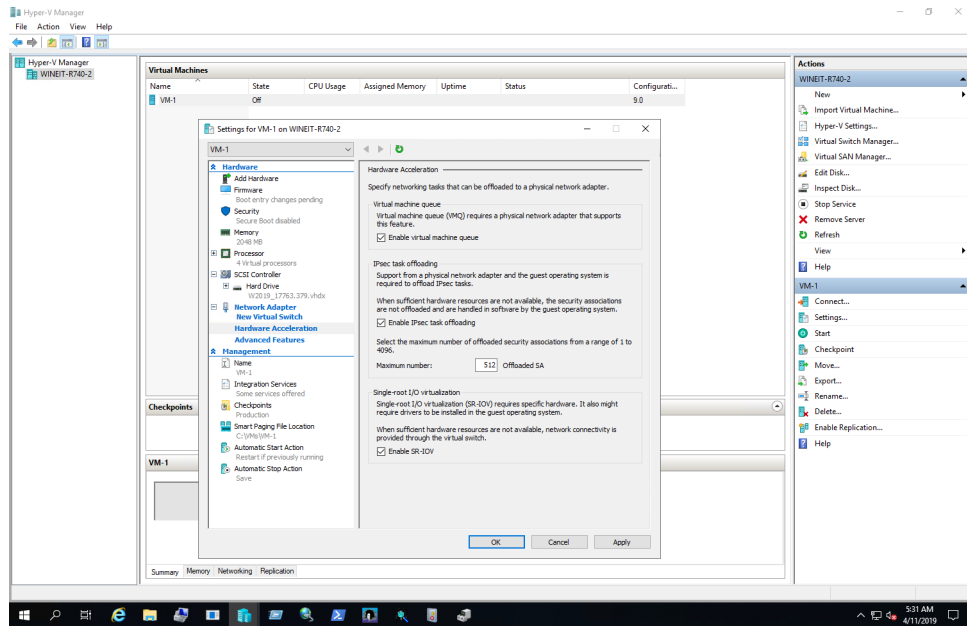


Figure 7-8. Enabling SR-IOV for the Network Adapter

4. Issue the following PowerShell command on the host to enable RDMA capabilities for the VMNIC (SR-IOV VF).

```
Set-VMNetworkAdapterRdma -VMName <VM_NAME>  
-VMNetworkAdapterName <VM_NIC_NAME> -RdmaWeight 100
```

NOTE

The VM must be powered off before issuing the PowerShell command.

5. Upgrade the Marvell drivers in the VM by booting the VM and installing the latest drivers using the Windows Super Installer on the Marvell CD.

Figure 7-9 shows an example.

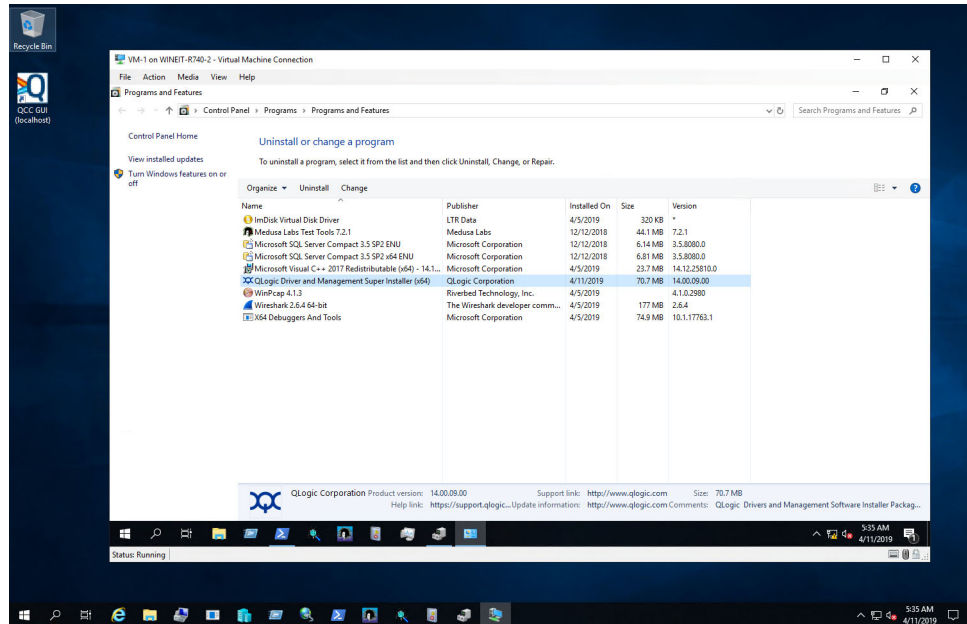


Figure 7-9. Upgrading Drivers in VM

6. Enable RMDA on the Microsoft network device associated with the VF inside the VM.

Figure 7-10 shows an example.

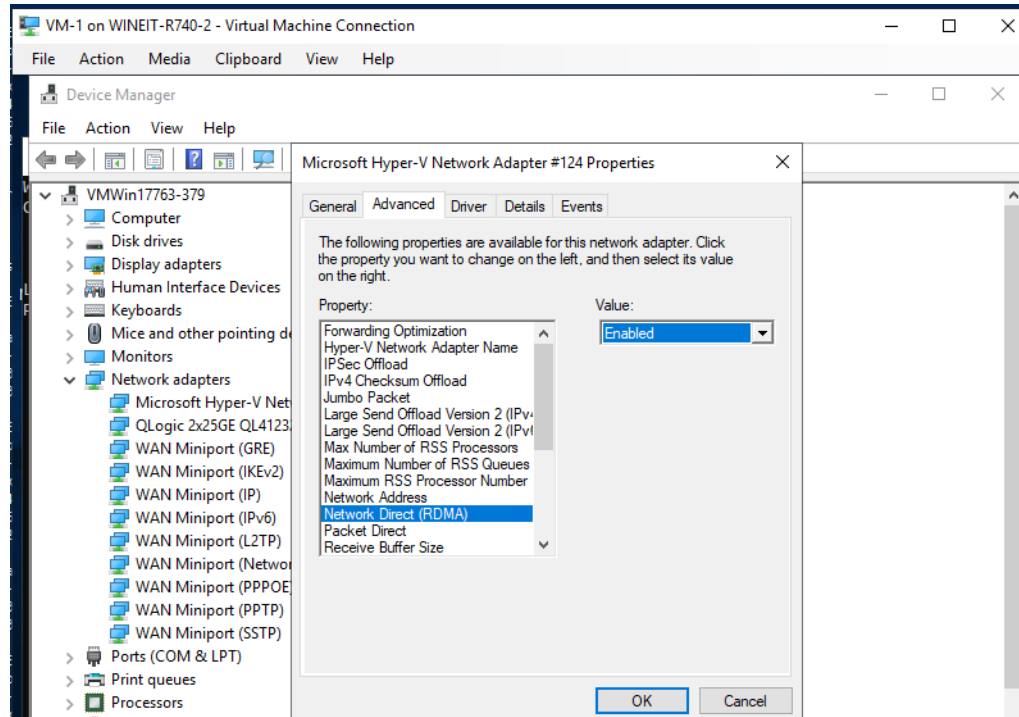


Figure 7-10. Enabling RDMA on the VMNIC

7. Start the VM RMDA traffic:
 - a. Connect a server message block (SMB) drive, run RoCE traffic, and verify the results.
 - b. Open the Performance monitor in the VM, and then add **RDMA Activity counter**.

- c. Verify that RDMA traffic is running.

Figure 7-11 provides an example.

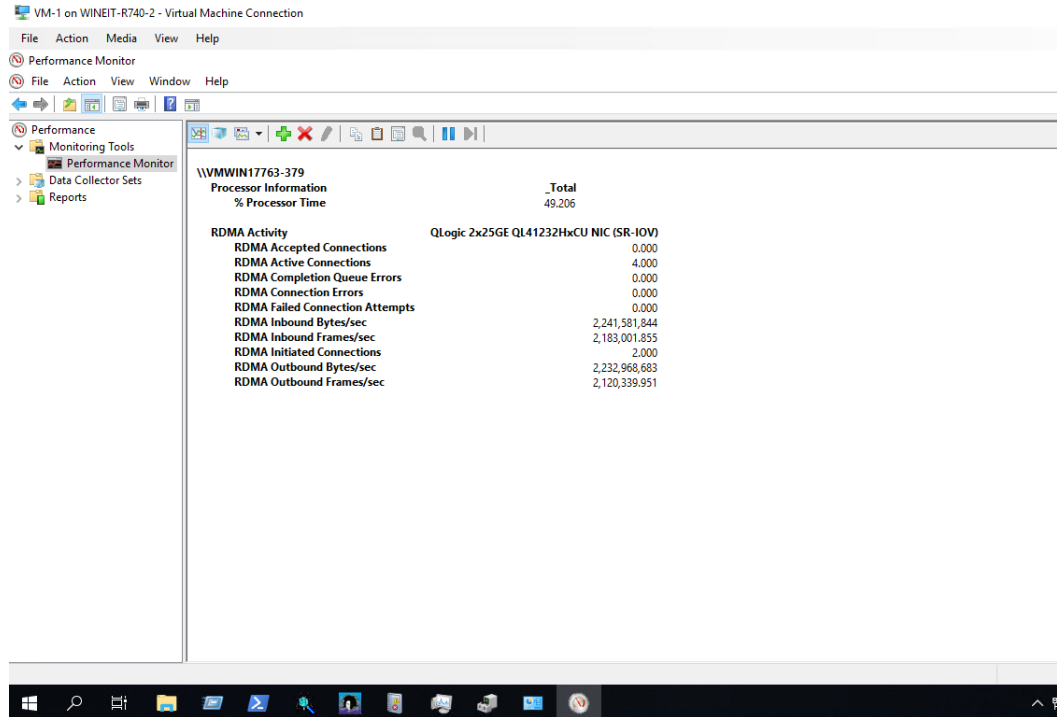


Figure 7-11. RDMA Traffic

Limitations

VF RDMA has the following limitations:

- VF RDMA is supported only for 41000-based devices.
- At the time of publication, only RoCEv2 is supported for VF RDMA. The same network direct technology must be configured in physical functions on both the host and SR-IOV VFs in the VM.
- A maximum of 16 VFs per PF can support VF RDMA. For quad-port adapters, the maximum is 8 VFs per PF.
- VF RDMA is supported only in Windows Server 2019/Azure Stack HCI (for both the host and VM OSs).
- VF RDMA is not supported for Linux VMs on Windows Hypervisor.
- VF RDMA is not supported in NPar mode.
- A maximum of 128 queue pairs (QPs)/connections are supported per VF.
- RDMA traffic between PF and its VFs, and among VFs of same PF, is supported. This traffic pattern is referred to as *loopback traffic*.

- On some older server platforms, VF devices may not be enumerated for one of the NIC PCI functions (PF). This limitation is because of the increased PCI base address register (BAR) requirements to support VF RDMA, meaning that the OS/BIOS cannot assign the required BAR for each VF.
- To support the maximum number of QPs in a VM, approximately 8GB of RAM must be available, assuming that only one VF is assigned to the VM. If less than 8GB of RAM is assigned to VM, there can be a sudden drop in the number of active connections due to insufficient memory and memory allocation failures.

Configuring RoCE on the Adapter for Linux

This section describes the RoCE configuration procedure for RHEL and SLES. It also describes how to verify the RoCE configuration and provides some guidance about using group IDs (GIDs) with vLAN interfaces.

- [RoCE Configuration for RHEL](#)
- [RoCE Configuration for SLES](#)
- [Verifying the RoCE Configuration on Linux](#)
- [vLAN Interfaces and GID Index Values](#)
- [RoCE v2 Configuration for Linux](#)
- [Configuring RoCE for SR-IOV VF Devices \(VF RDMA\)](#)

RoCE Configuration for RHEL

To configure RoCE on the adapter, the Open Fabrics Enterprise Distribution (OFED) must be installed and configured on the RHEL host.

To prepare inbox OFED for RHEL:

1. While installing or upgrading the operating system, select the InfiniBand and OFED support packages.
2. Install the following RPMs from the RHEL ISO image:

```
libibverbs-devel-x.x.x.x86_64.rpm  
(required for libqedr library)  
perftest-x.x.x.x86_64.rpm  
(required for InfiniBand bandwidth and latency applications)
```

or, using Yum, install the inbox OFED:

```
yum groupinstall "Infiniband Support"  
yum install perftest  
yum install tcl tcl-devel tk zlib-devel libibverbs  
libibverbs-devel
```

NOTE

During installation, if you already selected the previously mentioned packages, you need not reinstall them. The inbox OFED and support packages may vary depending on the operating system version.

3. Install the new Linux drivers as described in [“Installing the Linux Drivers with RDMA”](#) on page 15.

RoCE Configuration for SLES

To configure RoCE on the adapter for a SLES host, OFED must be installed and configured on the SLES host.

To install inbox OFED for SLES:

1. While installing or upgrading the operating system, select the InfiniBand support packages.
2. (SLES 12.x) Install the following RPMs from the corresponding SLES SDK kit image.

```
libibverbs-devel-x.x.x.x86_64.rpm  
(required for libqedr installation)
```

```
perftest-x.x.x.x86_64.rpm  
(required for bandwidth and latency applications)
```

3. (SLES 15/15 SP1) Install the following RPMs.

After installation, the `rdma-core*`, `libibverbs*`, `libibumad*`, `libibmad*`, `librdmacm*`, and `perftest` RPMs may be missing (all are required for RDMA). Install these packages using one of the following methods:

- Load the Package DVD and install the missing RPMs.
- Use the `zypper` command to install the missing RPMs. For example:

```
#zypper install rdma*  
#zypper install libib*  
#zypper install librdma*  
#zypper install perftest
```

4. Install the Linux drivers, as described in [“Installing the Linux Drivers with RDMA”](#) on page 15.

Verifying the RoCE Configuration on Linux

After installing OFED, installing the Linux driver, and loading the RoCE drivers, verify that the RoCE devices were detected on all Linux operating systems.

To verify RoCE configuration on Linux:

1. Stop firewall tables using `service/systemctl` commands.
2. For RHEL only: If the RDMA service is installed (`yum install rdma`), verify that the RDMA service has started.

NOTE

For RHEL 7.x and SLES 12 SPx and later, RDMA service starts itself after reboot.

On RHEL or CentOS: Use the `service rdma status` command to start service:

- If RDMA has not started, issue the following command:

```
# service rdma start
```

- If RDMA does not start, issue either of the following alternative commands:

```
# /etc/init.d/rdma start
```

or

```
# systemctl start rdma.service
```

3. Verify that the RoCE devices were detected by examining the `dmesg` logs:

```
# dmesg|grep qedr
```

```
[87910.988411] qedr: discovered and registered 2 RoCE funcs
```

4. Verify that all of the modules have been loaded. For example:

```
# lsmod|grep qedr
```

```
qedr                89871  0
qede                96670  1 qedr
qed                 2075255  2 qede,qedr
ib_core            88311  16 qedr, rdma_cm, ib_cm,
                  ib_sa, iw_cm, xprtrdma, ib_mad, ib_srp,
                  ib_ucm, ib_iser, ib_srpt, ib_umad,
                  ib_uverbs, rdma_ucm, ib_ipoib, ibisert
```

5. Configure the IP address and enable the port using a configuration method such as `ifconfig`. For example:

```
# ifconfig ethX 192.168.10.10/24 up
```

6. Issue the `ibv_devinfo` command. For each PCI function, you should see a separate `hca_id`, as shown in the following example:

```
root@captain:~# ibv_devinfo
hca_id: qedr0
      transport:                InfiniBand (0)
      fw_ver:                    8.3.9.0
      node_guid:                 020e:1eff:fe50:c7c0
      sys_image_guid:           020e:1eff:fe50:c7c0
      vendor_id:                 0x1077
      vendor_part_id:           5684
      hw_ver:                    0x0
      phys_port_cnt:            1
      port: 1
      state:                     PORT_ACTIVE (1)
      max_mtu:                   4096 (5)
      active_mtu:                1024 (3)
      sm_lid:                    0
      port_lid:                 0
      port_lmc:                 0x00
      link_layer:                Ethernet
```

7. Verify the L2 and RoCE connectivity between all servers: one server acts as a server, another acts as a client.

- Verify the L2 connection using a simple `ping` command.
- Verify the RoCE connection by performing an RDMA ping on the server or client:

On the server, issue the following command:

```
ibv_rc_pingpong -d <ib-dev> -g 0
```

On the client, issue the following command:

```
ibv_rc_pingpong -d <ib-dev> -g 0 <server L2 IP address>
```

The following are examples of successful ping pong tests on the server and the client.

Server Ping:

```
root@captain:~# ibv_rc_pingpong -d qedr0 -g 0
local address: LID 0x0000, QPN 0xff0000, PSN 0xb3e07e, GID
fe80::20e:1eff:fe50:c7c0
remote address: LID 0x0000, QPN 0xff0000, PSN 0x934d28, GID
fe80::20e:1eff:fe50:c570
8192000 bytes in 0.05 seconds = 1436.97 Mbit/sec
1000 iters in 0.05 seconds = 45.61 usec/iter
```

Client Ping:

```
root@lambodar:~# ibv_rc_pingpong -d qedr0 -g 0 192.168.10.165
local address: LID 0x0000, QPN 0xff0000, PSN 0x934d28, GID
fe80::20e:1eff:fe50:c570
remote address: LID 0x0000, QPN 0xff0000, PSN 0xb3e07e, GID
fe80::20e:1eff:fe50:c7c0
8192000 bytes in 0.02 seconds = 4211.28 Mbit/sec
1000 iters in 0.02 seconds = 15.56 usec/iter
```

- To display RoCE statistics, issue the following commands, where **x** is the device number:

```
> mount -t debugfs nodev /sys/kernel/debug
> cat /sys/kernel/debug/qedr/qedrX/stats
```

vLAN Interfaces and GID Index Values

If you are using vLAN interfaces on both the server and the client, you must also configure the same vLAN ID on the switch. If you are running traffic through a switch, the InfiniBand applications must use the correct GID value, which is based on the vLAN ID and vLAN IP address.

Based on the following results, the GID value (-x 4 / -x 5) should be used for any perfest applications.

```
# ibv_devinfo -d qedr0 -v|grep GID
GID[ 0]: fe80:0000:0000:0000:020e:1eff:fe50:c5b0
GID[ 1]: 0000:0000:0000:0000:0000:ffff:c0a8:0103
GID[ 2]: 2001:0db1:0000:0000:020e:1eff:fe50:c5b0
GID[ 3]: 2001:0db2:0000:0000:020e:1eff:fe50:c5b0
GID[ 4]: 0000:0000:0000:0000:0000:ffff:c0a8:0b03 IP address for vLAN interface
GID[ 5]: fe80:0000:0000:0000:020e:1e00:0350:c5b0 vLAN ID 3
```


NOTE

The default GID value is zero (0) for back-to-back or pause settings. For server and switch configurations, you must identify the proper GID value. If you are using a switch, refer to the corresponding switch configuration documents for the correct settings.

RoCE v2 Configuration for Linux

To verify RoCE v2 functionality, you must use RoCE v2 supported kernels.

To configure RoCE v2 for Linux:

1. Ensure that you are using one of the following supported kernels:
 - SLES 15 SP1, SP2, and later
 - RHEL 7.8, 7.9, 8.2, 8.3, and later
2. Configure RoCE v2 as follows:
 - a. Identify the GID index for RoCE v2.
 - b. Configure the routing address for the server and client.
 - c. Enable L3 routing on the switch.

NOTE

You can configure RoCE v1 and RoCE v2 by using RoCE v2-supported kernels. These kernels allow you to run RoCE traffic over the same subnet, as well as over different subnets such as RoCE v2 and any routable environment. Only a few settings are required for RoCE v2, and all other switch and adapter settings are common for RoCE v1 and RoCE v2.

Identifying the RoCE v2 GID Index or Address

To find RoCE v1- and RoCE v2-specific GIDs, use either sys or class parameters, or run RoCE scripts from the 41000 FastLinQ source package. To check the default **RoCE GID Index** and address, issue the `ibv_devinfo` command and compare it with the sys or class parameters. For example:

```
#ibv_devinfo -d qedr0 -v|grep GID
GID[ 0]:          fe80:0000:0000:0000:020e:1eff:fec4:1b20
GID[ 1]:          fe80:0000:0000:0000:020e:1eff:fec4:1b20
GID[ 2]:          0000:0000:0000:0000:0000:ffff:1e01:010a
GID[ 3]:          0000:0000:0000:0000:0000:ffff:1e01:010a
GID[ 4]:          3ffe:ffff:0000:0f21:0000:0000:0000:0004
GID[ 5]:          3ffe:ffff:0000:0f21:0000:0000:0000:0004
```

```
GID[ 6]:          0000:0000:0000:0000:0000:ffff:c0a8:6403
GID[ 7]:          0000:0000:0000:0000:0000:ffff:c0a8:6403
```

Verifying the RoCE v1 or RoCE v2 GID Index and Address from sys and class Parameters

Use one of the following options to verify the RoCE v1 or RoCE v2 GID Index and address from the sys and class parameters:

■ Option 1:

```
# cat /sys/class/infiniband/qedr0/ports/1/gid_attrs/types/0
IB/RoCE v1
# cat /sys/class/infiniband/qedr0/ports/1/gid_attrs/types/1
RoCE v2

# cat /sys/class/infiniband/qedr0/ports/1/gids/0
fe80:0000:0000:0000:020e:1eff:fec4:1b20
# cat /sys/class/infiniband/qedr0/ports/1/gids/1
fe80:0000:0000:0000:020e:1eff:fec4:1b20
```

■ Option 2:

Use the scripts from the FastLinQ source package.

```
#!/../fastlinq-8.x.x.x/add-ons/roce/show_gids.sh
DEV  PORT  INDEX  GID                                     IPv4          VER    DEV
---  ----  -
qedr0  1    0    fe80:0000:0000:0000:020e:1eff:fec4:1b20          v1    p4p1
qedr0  1    1    fe80:0000:0000:0000:020e:1eff:fec4:1b20          v2    p4p1
qedr0  1    2    0000:0000:0000:0000:0000:ffff:1e01:010a    30.1.1.10    v1    p4p1
qedr0  1    3    0000:0000:0000:0000:0000:ffff:1e01:010a    30.1.1.10    v2    p4p1
qedr0  1    4    3ffe:ffff:0000:0f21:0000:0000:0000:0004          v1    p4p1
qedr0  1    5    3ffe:ffff:0000:0f21:0000:0000:0000:0004          v2    p4p1
qedr0  1    6    0000:0000:0000:0000:0000:ffff:c0a8:6403    192.168.100.3    v1    p4p1.100
qedr0  1    7    0000:0000:0000:0000:0000:ffff:c0a8:6403    192.168.100.3    v2    p4p1.100
qedr1  1    0    fe80:0000:0000:0000:020e:1eff:fec4:1b21          v1    p4p2
qedr1  1    1    fe80:0000:0000:0000:020e:1eff:fec4:1b21          v2    p4p2
```

NOTE

You must specify the GID index values for RoCE v1- or RoCE v2-based server or switch configuration (Pause/PFC). Use the GID index for the link local IPv6 address, IPv4 address, or IPv6 address. To use vLAN tagged frames for RoCE traffic, you must specify GID index values that are derived from the vLAN IPv4 or IPv6 address.

Verifying the RoCE v1 or RoCE v2 Function Through perfest Applications

This section shows how to verify the RoCE v1 or RoCE v2 function through perfest applications. In this example, the following server IP and client IP are used:

- Server IP: 192.168.100.3
- Client IP: 192.168.100.4

Verifying RoCE v1

Run over the same subnet and use the RoCE v1 GID Index.

```
Server# ib_send_bw -d qedr0 -F -x 0
Client# ib_send_bw -d qedr0 -F -x 0 192.168.100.3
```

Verifying RoCE v2

Run over the same subnet and use the RoCE v2 GID Index.

```
Server# ib_send_bw -d qedr0 -F -x 1
Client# ib_send_bw -d qedr0 -F -x 1 192.168.100.3
```

NOTE

If you are running through a switch PFC configuration, use vLAN GIDs for RoCE v1 or v2 through the same subnet.

Verifying RoCE v2 Through Different Subnets

NOTE

You must first configure the route settings for the switch and servers. On the adapter, set the RoCE priority and DCBX mode using either the HII, UEFI user interface, or one of the Marvell management utilities.

To verify RoCE v2 through different subnets:

1. Set the route configuration for the server and client using the DCBX-PFC configuration.

System Settings:

Server VLAN IP : 192.168.100.3 and **Gateway** :192.168.100.1

Client VLAN IP : 192.168.101.3 and **Gateway** :192.168.101.1

Server Configuration:

```
#!/sbin/ip link add link p4p1 name p4p1.100 type vlan id 100
#ifconfig p4p1.100 192.168.100.3/24 up
#ip route add 192.168.101.0/24 via 192.168.100.1 dev p4p1.100
```

Client Configuration:

```
#!/sbin/ip link add link p4p1 name p4p1.101 type vlan id 101
#ifconfig p4p1.101 192.168.101.3/24 up
#ip route add 192.168.100.0/24 via 192.168.101.1 dev p4p1.101
```

2. Set the switch settings using the following procedure.

- Use any flow control method (Pause, DCBX-CEE, or DCBX-IEEE), and enable IP routing for RoCE v2. See [“Preparing the Ethernet Switch” on page 136](#) for RoCE v2 configuration, or refer to the vendor switch documents.
- If you are using PFC configuration and L3 routing, run RoCE v2 traffic over the vLAN using a different subnet, and use the RoCE v2 vLAN GID index.

```
Server# ib_send_bw -d qedr0 -F -x 5
```

```
Client# ib_send_bw -d qedr0 -F -x 5 192.168.100.3
```

Server Switch Settings:

```
[root@RoCE-Auto-2 /]# ib_send_bw -d qedr0 -F -x 5 -q 2 --report_gbits
*****
* Waiting for client to connect... *
*****
-----
                Send BW Test
Dual-port      : OFF          Device       : qedr0
Number of qps  : 2           Transport type : IB
Connection type : RC         Using SRQ     : OFF
RX depth       : 512
CQ Moderation  : 100
MTU            : 1024[B]
Link type      : Ethernet
Gid index      : 5
Max inline data : 0[B]
rdma_cm QPs    : OFF
Data_ex. method : Ethernet
-----
local address: LID 0000 QPN 0xff0000 PSN 0xf0b2c3
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:100:03
local address: LID 0000 QPN 0xff0002 PSN 0xa2b8f1
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:100:03
remote address: LID 0000 QPN 0xff0000 PSN 0x40473a
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:101:03
remote address: LID 0000 QPN 0xff0002 PSN 0x124cd3
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:101:03
-----
#bytes      #iterations  BW peak[Gb/sec]  BW average[Gb/sec]  MsgRate[Mpps]
65536       1000          0.00             23.07                0.043995
-----
```

Figure 7-12. Switch Settings, Server

Client Switch Settings:

```
[root@roce-auto-1 ~]# ib_send_bw -d qedr0 -F -x 5 192.168.100.3 -q 2 --report_gbits
-----
                Send BW Test
Dual-port      : OFF          Device       : qedr0
Number of qps  : 2           Transport type : IB
Connection type : RC         Using SRQ     : OFF
TX depth       : 128
CQ Moderation  : 100
MTU            : 1024[B]
Link type      : Ethernet
Gid index      : 5
Max inline data : 0[B]
rdma_cm QPs    : OFF
Data_ex. method : Ethernet
-----
local address: LID 0000 QPN 0xff0000 PSN 0x40473a
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:101:03
local address: LID 0000 QPN 0xff0002 PSN 0x124cd3
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:101:03
remote address: LID 0000 QPN 0xff0000 PSN 0xf0b2c3
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:100:03
remote address: LID 0000 QPN 0xff0002 PSN 0xa2b8f1
GID: 00:00:00:00:00:00:00:00:00:00:255:255:192:168:100:03
-----
#bytes      #iterations  BW peak[Gb/sec]  BW average[Gb/sec]  MsgRate[Mpps]
65536       1000          23.04            23.04                0.043936
-----
```

Figure 7-13. Switch Settings, Client

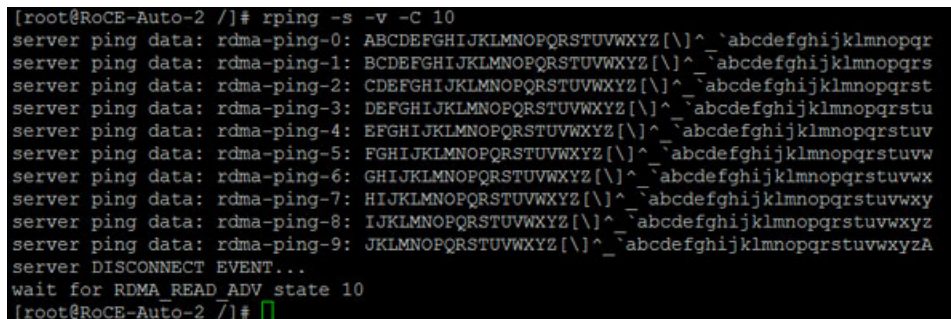
Configuring RoCE v1 or RoCE v2 Settings for RDMA_CM Applications

To configure RoCE, use the following scripts from the FastLinQ source package:

```
# ./show_rdma_cm_roce_ver.sh
qedr0 is configured to IB/RoCE v1
qedr1 is configured to IB/RoCE v1

# ./config_rdma_cm_roce_ver.sh v2
configured rdma_cm for qedr0 to RoCE v2
configured rdma_cm for qedr1 to RoCE v2
```

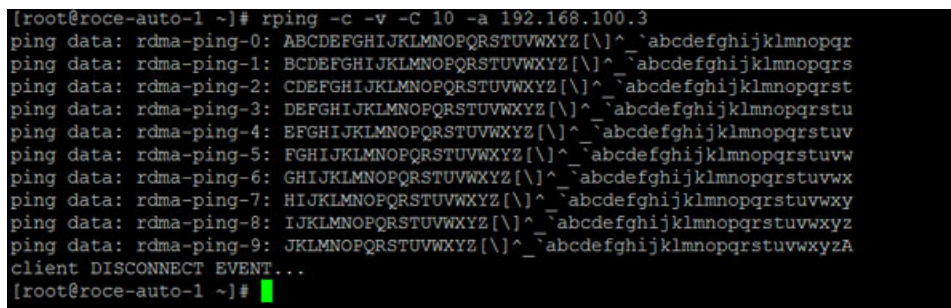
Server Settings:



```
[root@RoCE-Auto-2 /]# rping -s -v -C 10
server ping data: rdma-ping-0: ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-1: BCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-2: CDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-3: DEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-4: EFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-5: FGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-6: GHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-7: HIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-8: IJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server ping data: rdma-ping-9: JKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
server DISCONNECT EVENT...
wait for RDMA_READ_ADV state 10
[root@RoCE-Auto-2 /]#
```

Figure 7-14. Configuring RDMA_CM Applications: Server

Client Settings:



```
[root@roce-auto-1 ~]# rping -c -v -C 10 -a 192.168.100.3
ping data: rdma-ping-0: ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-1: BCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-2: CDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-3: DEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-4: EFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-5: FGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-6: GHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-7: HIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-8: IJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-9: JKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
client DISCONNECT EVENT...
[root@roce-auto-1 ~]#
```

Figure 7-15. Configuring RDMA_CM Applications: Client

Configuring RoCE for SR-IOV VF Devices (VF RDMA)

The following sections describe how to configure RoCE for SR-IOV VF devices (also referred to as VFRMDA) on Linux. Associated information and limitations are also provided.

Table 7-4 lists the supported Linux OS combinations.

Table 7-4. Supported Linux OSs for VF RDMA

Hypervisor	Guest OS					
	RHEL 7.8	RHEL 7.9	RHEL 8.2	RHEL 8.3	SLES15 SP1	SLES15 SP2
	Yes	Yes	Yes	Yes	Yes	Yes
RHEL 7.8	Yes	Yes	Yes	Yes	Yes	Yes
RHEL 7.9	Yes	Yes	Yes	Yes	Yes	Yes
RHEL 8.2	Yes	Yes	Yes	Yes	Yes	Yes
RHEL 8.3	Yes	Yes	Yes	Yes	Yes	Yes
SLES15 SP1	Yes	Yes	Yes	Yes	Yes	Yes
SLES15 SP2	Yes	Yes	Yes	Yes	Yes	Yes

If you are using the inbox OFED, use the same OFED distribution between the hypervisor host OS and the guest (VM) OS. Check the out-of-box OFED distribution release notes for their specific supported host OS-to-VM OS distribution matrix.

Enumerating VFs for L2 and RDMA

There are two ways to enumerate the VFs:

- [User Defined VF MAC Allocation](#)
- [Dynamic or Random VF MAC Allocation](#)

User Defined VF MAC Allocation

When defining the VF MAC allocation, there are no changes in the default VF enumeration method. After creating the number of VFs, assign the static MAC address.

To create a user defined VF MAC allocation:

1. Enumerate the default VF.

```
# modprobe -v qede
# echo 2 > /sys/class/net/p6p1/device/sriov_numvfs
# ip link show
14: p6p1: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN
mode DEFAULT group default qlen 1000
    link/ether 14:02:ec:ce:d0:e4 brd ff:ff:ff:ff:ff:ff
```

```
vf 0 MAC 00:00:00:00:00:00, spoof checking off, link-state auto
vf 1 MAC 00:00:00:00:00:00, spoof checking off, link-state auto
```

2. Assign the static MAC address:

```
# ip link set dev p6p1 vf 0 mac 3c:33:44:55:66:77
# ip link set dev p6p1 vf 1 mac 3c:33:44:55:66:89
#ip link show
14: p6p1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default
qlen 1000
    link/ether 14:02:ec:ce:d0:e4 brd ff:ff:ff:ff:ff:ff
        vf 0 MAC 3c:33:44:55:66:77, tx rate 25000 (Mbps), max_tx_rate 25000Mbps, spoof checking off,
link-state auto
        vf 1 MAC 3c:33:44:55:66:89, tx rate 25000 (Mbps), max_tx_rate 25000Mbps, spoof checking off,
link-state auto
```

3. To reflect for RDMA, load/unload the `qedr` driver if it is already loaded.

```
#rmmod qedr
#modprobe          qedr
#ibv_devices
    device          node GUID
    -----
    qedr0           1602ecffffeced0e4
    qedr1           1602ecffffeced0e5
    qedr_vf0        3e3344ffffe556677
    qedr_vf1        3e3344ffffe556689
```

Dynamic or Random VF MAC Allocation

To dynamically allocate a VF MAC:

```
# modprobe -r qedr
# modprobe -v qed vf_mac_origin=3 [Use this module parameter for dynamic
MAC allocation]
# modprobe -v qede
# echo 2 > /sys/class/net/p6p1/device/sriov_numvfs
# modprobe qedr (This is an optional, mostly qedr driver loads
itself)
# ip link show|grep vf
    vf 0 MAC ba:1a:ad:08:89:00, tx rate 25000 (Mbps), max_tx_rate
25000Mbps, spoof checking off, link-state auto
    vf 1 MAC 96:40:61:49:cd:68, tx rate 25000 (Mbps), max_tx_rate
25000Mbps, spoof checking off, link-state auto
# lsmod |grep qedr
```



```
# ibv_devices
device                node GUID
-----                -
qedr0                 1602ecfffececfa0
qedr1                 1602ecfffececfa1
qedr_vf0              b81aadfffe088900
qedr_vf1              944061fffe49cd68
```

Number of VFs Supported for RDMA

For the 41000 Series Adapters, the number of VFs for L2 and RDMA are shared based on resources availability.

Dual Port Adapters

Each PF supports a maximum of 40 VFs for RDMA. If the number of VFs exceeds 56, it will be subtracted by the total number of VFs (96).

In the following example, PF0 is

```
/sys/class/net/<PF-interface>/device/sriov_numvfs
```

```
Echo 40 > PF0 (VFs for L2+RDMA=40+40 (40 VFs can use for both L2 and RDMA))
```

```
Echo 56 > PF0 (VFs for L2+RDMA=56+40)
```

After crossing 56 VFs, this number is subtracted by the total number of VFs. For example:

```
echo 57 > PF0 then 96-57=39 VFs for RDMA (57 VFs for L2 + 39VFs for RDMA)
```

```
echo 96 > PF0 then 96-96=0 VFs for RDMA (all 96 VFs can use only for L2)
```

To view the available VFs for L2 and RDMA:

```
L2          : # ip link show
```

```
RDMA: # ibv_devices
```

Quad Port Adapters

Each PF supports a maximum of 20 VFs for RDMA; until 48 VFs, there are 20 VFs for RDMA. When exceeding 28 VFs, that number is subtracted by the total VFs (48).

For example, in a 4x10G:

```
Echo 20 > PF0 (VFs for L2+RDMA=20+20)
```

```
Echo 28 > PF0 (VFs for L2+RDMA=28+20)
```

When exceeding 28 VFs, this number is subtracted by the total number of VFs.
For example:

```
echo 29 > PF0 (48-29=19VFs for RDMA; 29 VFs for L2 + 19 VFs for RDMA)
echo 48 > PF0 (48-48=0 VFs for RDMA; all 48 VFs can use only for L2)
```

Limitations

VF RMDA has the following limitations:

- No iWARP support
- No NPar support
- Cross OS is not supported on components earlier than 8.5x.x.x; for example, a Linux hypervisor cannot use a Windows guest OS (VM).
- Perf test latency test on VF interfaces can be run only with the inline size zero `-I 0` option. Neither the default nor more than one inline size works.

This limitation is expected behavior using the inbox OFED/rdma-core available in current distribution releases. To use the default/different inline size, use the upstream rdma-core from GitHub®, found here:

<https://github.com/linux-rdma/rdma-core/releases>

Compile and export user library path by issuing the following command:

```
export LD_LIBRARY_PATH=<rdma-core-path>/build/lib
```

- To allow RDMA_CM applications to run on different MTU sizes (512–9000) other than the default (1500), follow these steps:
 1. Unload the `qedr` driver:

```
#rmmod qedr
```
 2. Set MTU on the VF interface:

```
#ifconfig <VF interface> mtu 9000
```
 3. Load the `qedr` driver (the RoCE MTU size is automatically increased to 4,096 when the driver is reloaded):

```
#modprobe qedr
```
- The `rdma_server/rdma_xserver` does not support VF interfaces. The workaround is to use the upstream rdma-core from GitHub.
- No RDMA bonding support on VFs.

Configuring RoCE on the Adapter for VMware ESX

This section provides the following procedures and information for RoCE configuration:

- [Configuring RDMA Interfaces](#)
- [Configuring MTU](#)
- [RoCE Mode and Statistics](#)
- [Configuring a Paravirtual RDMA Device \(PVRDMA\)](#)

NOTE

Mapping Ethernet speeds to RDMA speeds is not always accurate, because values that can be specified by the RoCE driver are aligned with Infiniband®. For example, if RoCE is configured on an Ethernet interface operating at 1Gbps, the RDMA speed is shown as 2.5Gbps. There are no other suitable values available in the header files provided by ESXi that can be used by the RoCE driver to indicate 1Gbps speed.

Configuring RDMA Interfaces

To configure the RDMA interfaces:

1. Install both Marvell NIC and RoCE drivers.
2. Using the module parameter, enable the RoCE function from the NIC driver by issuing the following command:

```
esxcfg-module -s 'enable_roce=1' qedentv
```

To apply the change, reload the NIC driver or reboot the system.

3. To view a list of the NIC interfaces, issue the `esxcfg-nics -l` command. For example:

```
esxcfg-nics -l
```

Name	PCI	Driver	Link	Speed	Duplex	MAC Address	MTU	Description
Vmnic0	0000:01:00.2	qedentv	Up	25000Mbps	Full	a4:5d:36:2b:6c:92	1500	QLogic Corp.
QLogic FastLinQ	QL41xxx	1/10/25	GbE	Ethernet	Adapter			
Vmnic1	0000:01:00.3	qedentv	Up	25000Mbps	Full	a4:5d:36:2b:6c:93	1500	QLogic Corp.
QLogic FastLinQ	QL41xxx	1/10/25	GbE	Ethernet	Adapter			

4. To view a list of the RDMA devices, issue the `esxcli rdma device list` command. For example:

```
esxcli rdma device list
```

7-RoCE Configuration

Configuring RoCE on the Adapter for VMware ESX

Name	Driver	State	MTU	Speed	Paired Uplink	Description
vmrdma0	qedrntv	Active	1024	25 Gbps	vmnic0	QLogic FastLinQ QL45xxx RDMA Interface
vmrdma1	qedrntv	Active	1024	25 Gbps	vmnic1	QLogic FastLinQ QL45xxx RDMA Interface

5. To create a new virtual switch, issue the following command:

```
esxcli network vswitch standard add -v <new vswitch name>
```

For example:

```
# esxcli network vswitch standard add -v roce_vs
```

This creates a new virtual switch named *roce_vs*.

6. To associate the Marvell NIC port to the vSwitch, issue the following command:

```
# esxcli network vswitch standard uplink add -u <uplink device> -v <roce vswitch>
```

For example:

```
# esxcli network vswitch standard uplink add -u vmnic0 -v roce_vs
```

7. To create a new port group on this vSwitch, issue the following command:

```
# esxcli network vswitch standard portgroup add -p roce_pg -v roce_vs
```

For example:

```
# esxcli network vswitch standard portgroup add -p roce_pg -v roce_vs
```

8. To create a vmknic interface on this port group and configure the IP, issue the following command:

```
# esxcfg-vmknic -a -i <IP address> -n <subnet mask> <roce port group name>
```

For example:

```
# esxcfg-vmknic -a -i 192.168.10.20 -n 255.255.255.0 roce_pg
```

9. To configure the VLAN ID, issue the following command:

```
# esxcfg-vswitch -v <VLAN ID> -p roce_pg
```

To run RoCE traffic with a VLAN ID, configure the VLAN ID on the corresponding VMkernel port group.

Configuring MTU

To modify the MTU for an RoCE interface, change the MTU of the corresponding vSwitch.

For optimal performance, the RoCE MTU size should be 4,096. Therefore, set the vSwitch L2 Ethernet MTU size to be larger than 4,096. In other words, this (per vSwitch instance) RoCE MTU size is automatically set to the largest supported size, which is smaller than the current vSwitch instance's L2 Ethernet MTU size. Additionally, set the network and target ports to an equivalent L2 Ethernet MTU size (to prevent the packets from fragmenting or dropping). On VMware ESXi, this RoCE MTU value is automatically changed when the vSwitch's L2 Ethernet MTU value is changed.

Set the MTU size of the RDMA interface based on the MTU of the vSwitch by issuing the following command:

```
# esxcfg-vswitch -m <new MTU> <RoCE vswitch name>
```

For example:

```
# esxcfg-vswitch -m 4000 roce_vs
# esxcli rdma device list
```

Name	Driver	State	MTU	Speed	Paired Uplink	Description
vmrdma0	qedrntv	Active	2048	25 Gbps	vmnic0	QLogic FastLinQ QL45xxx RDMA Interface
vmrdma1	qedrntv	Active	1024	25 Gbps	vmnic1	QLogic FastLinQ QL45xxx RDMA Interface

RoCE Mode and Statistics

For the RoCE mode, ESXi requires concurrent support of both RoCE v1 and v2. The decision regarding which RoCE mode to use is made during queue pair creation. The ESXi driver advertises both modes during registration and initialization. To view RoCE statistics, issue the following command:

```
# esxcli rdma device stats get -d vmrdma0
  Packets received: 0
  Packets sent: 0
  Bytes received: 0
  Bytes sent: 0
  Error packets received: 0
  Error packets sent: 0
  Error length packets received: 0
  Unicast packets received: 0
  Multicast packets received: 0
  Unicast bytes received: 0
  Multicast bytes received: 0
```

```
Unicast packets sent: 0
Multicast packets sent: 0
Unicast bytes sent: 0
Multicast bytes sent: 0
Queue pairs allocated: 0
Queue pairs in RESET state: 0
Queue pairs in INIT state: 0
Queue pairs in RTR state: 0
Queue pairs in RTS state: 0
Queue pairs in SQD state: 0
Queue pairs in SQE state: 0
Queue pairs in ERR state: 0
Queue pair events: 0
Completion queues allocated: 1
Completion queue events: 0
Shared receive queues allocated: 0
Shared receive queue events: 0
Protection domains allocated: 1
Memory regions allocated: 3
Address handles allocated: 0
Memory windows allocated: 0
```

Configuring a Paravirtual RDMA Device (PVRDMA)

See VMware's documentation (for example, <https://kb.vmware.com/articleview?docid=2147694>) for details on configuring PVRDMA using a vCenter interface. The following instructions are only for reference.

To configure PVRDMA using a vCenter interface:

1. Create and configure a new distributed virtual switch as follows:
 - a. In the VMware vSphere® Web Client, right-click the **RoCE** node in the left pane of the Navigator window.
 - b. On the Actions menu, point to **Distributed Switch**, and then click **New Distributed Switch**.
 - c. Select the current vSphere version being used.
 - d. Under **New Distributed Switch**, click **Edit settings**, and then configure the following:
 - **Number of uplinks**. Select an appropriate value.

- **Network I/O Control.** Select **Disabled**.
- **Default port group.** Select the **Create a default port group** check box.
- **Port group name.** Type a name for the port group.

Figure 7-16 shows an example.

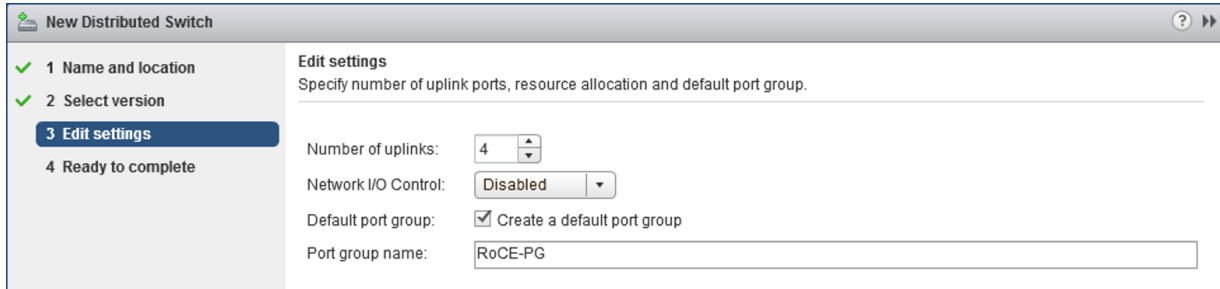


Figure 7-16. Configuring a New Distributed Switch

2. Configure a distributed virtual switch as follows:
 - a. In the VMware vSphere Web Client, expand the **RoCE** node in the left pane of the Navigator window.
 - b. Right-click **RoCE-VDS**, and then click **Add and Manage Hosts**.
 - c. Under **Add and Manage Hosts**, configure the following:
 - **Assign uplinks.** Select from the list of available uplinks.
 - **Manage VMkernel network adapters.** Accept the default, and then click **Next**.
 - **Migrate VM networking.** Assign the port group created in [Step 1](#).
3. Assign a vmknics for PVRDMA to use on ESX hosts:
 - a. Right-click a host, and then click **Settings**.
 - b. On the Settings page, expand the **System** node, and then click **Advanced System Settings**.
 - c. The Advanced System Settings page shows the key-pair value and its summary. Click **Edit**.
 - d. On the Edit Advanced System Settings page, filter on **PVRDMA** to narrow all the settings to just Net.PVRDMAMknics.
 - e. Set the **Net.PVRDMAMknics** value to **vmknics**; for example, **vmk1**.

Figure 7-17 shows an example.

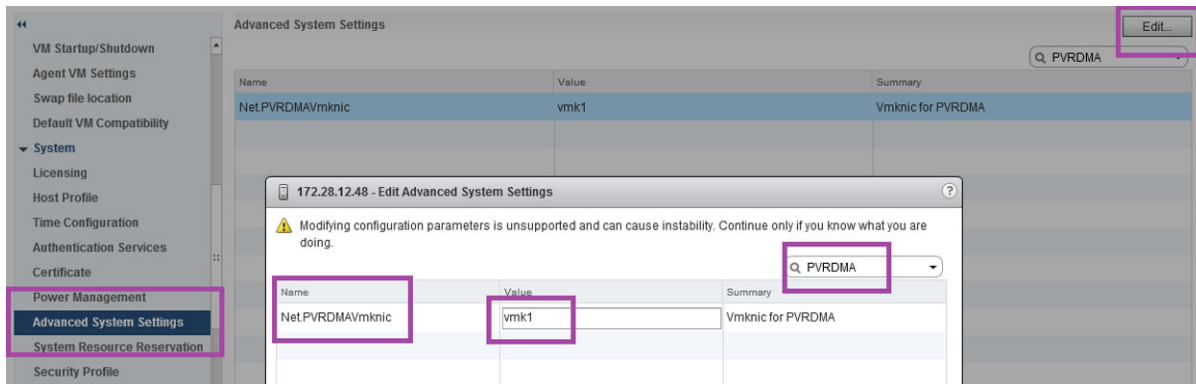


Figure 7-17. Assigning a vmknick for PVRDMA

4. Set the firewall rule for the PVRDMA:
 - a. Right-click a host, and then click **Settings**.
 - b. On the Settings page, expand the **System** node, and then click **Security Profile**.
 - c. On the Firewall Summary page, click **Edit**.
 - d. In the Edit Security Profile dialog box under **Name**, scroll down, select the **pvrDMA** check box, and then select the **Set Firewall** check box.

Figure 7-18 shows an example.

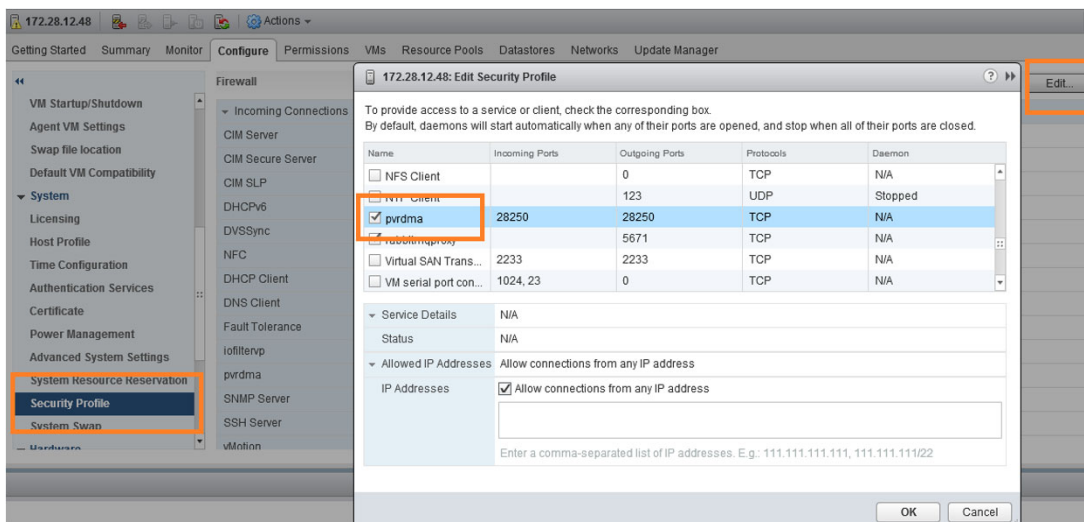


Figure 7-18. Setting the Firewall Rule

5. Set up the VM for PVRDMA as follows:
 - a. Install the supported RHEL guest OS.
 - b. Install OFED4.17-1.
 - c. Compile and install the PVRDMA guest driver and library.
 - d. Add a new PVRDMA network adapter to the VM as follows:
 - Edit the VM settings.
 - Add a new network adapter.
 - Select the newly added DVS port group as **Network**.
 - Select **PVRDMA** as the adapter type.
 - e. After the VM is booted, ensure that the PVRDMA guest driver is loaded.

Configuring the RoCE Namespace

To configure the RoCE namespace:

1. Install latest E4 driver bundle.

In latest driver, RoCE namespace is enable by default. For example:

```
ns_en: bool
Enable PVRDMA namespace support. 1: Enable(Default),0: Disable
```

The `num_ns` parameter indicates the number of namespaces to be enabled; it must be an unsigned integer. Valid values are 2–`VMK_STRINGIFY(QEDRNTV_MAX_NS)`. (See [Step 2 on page 260](#)).

The `num_ns` parameter is valid only when the `ns_en` parameter is 1.

2. Change the VM compatibility hardware version to 17 if needed.
3. Reboot the system.
4. Check for the bootlog and search for the namespace with the `grep` utility. Following is a sample output showing that the namespace has been created.

```
"cpu46:2098584) [qedrntv_dev_associate:738 (R0000:03:00.0)]Namespace
management capability registered."
```

5. Verify the namespace in `vsish` by issuing the following command

```
/> get /vmkModules/vrdma/pvrDMADevices/2112542_0/properties
PVRDMA Device Properties {
    VMM leader ID of VM:2112542
    adapter index:0
    MAC address:00:50:56:a3:96:10
    Physical HCA available:1
    Namespace allocated:1
```

```
        SRQ support enabled:1
        MR Key extension enabled:1
        Phys handles enabled:1
        Prefer RoCE v1 over v2:0
        RoCE version:2
        Active MTU:1024
    }
```

Configuring DCQCN

Data Center Quantized Congestion Notification (DCQCN) is a feature that determines how an RoCE receiver notifies a transmitter that a switch between them has provided an explicit congestion notification (notification point), and how a transmitter reacts to such notification (reaction point).

This section provides the following information about DCQCN configuration:

- [DCQCN Terminology](#)
- [DCQCN Overview](#)
- [DCB-related Parameters](#)
- [Global Settings on RDMA Traffic](#)
- [Configuring DSCP-PFC](#)
- [Enabling DCQCN](#)
- [Configuring CNP](#)
- [DCQCN Algorithm Parameters](#)
- [MAC Statistics](#)
- [Script Example](#)
- [Limitations](#)

DCQCN Terminology

The following terms describe DCQCN configuration:

- **ToS** (type of service) is a single-byte in the IPv4 header field. ToS comprises two ECN least significant bits (LSB) and six Differentiated Services Code Point (DSCP) most significant bits (MSB). For IPv6, traffic class is the equivalent of the IPv4 ToS.
- **ECN** (explicit congestion notification) is a mechanism where a switch adds to outgoing traffic an indication that congestion is imminent.

- **CNP** (congestion notification packet) is a packet used by the notification point to indicate that the ECN arrived from the switch back to the reaction point. CNP is defined in the Supplement to *InfiniBand Architecture Specification Volume 1 Release 1.2.1*, located here:
<https://cw.infinibandta.org/document/dl/7781>
- **VLAN Priority** is a field in the L2 vLAN header. The field is the three MSBs in the vLAN tag.
- **PFC** (priority-based flow control) is a flow control mechanism that applies to traffic carrying a specific vLAN priority.
- **DSCP-PFC** is a feature that allows a receiver to interpret the priority of an incoming packet for PFC purposes, rather than according to the vLAN priority or the DSCP field in the IPv4 header. You may use an indirection table to indicate a specified DSCP value to a vLAN priority value. DSCP-PFC can work across L2 networks because it is an L3 (IPv4) feature.
- **Traffic classes**, also known as priority groups, are groups of vLAN priorities (or DSCP values if DSCP-PFC is used) that can have properties such as being lossy or lossless. Generally, 0 is used for the default common lossy traffic group, 3 is used for the FCoE traffic group, and 4 is used for the iSCSI-TLV traffic group. You may encounter DCB mismatch issues if you attempt to reuse these numbers on networks that also support FCoE or iSCSI-TLV traffic. Marvell recommends that you use numbers 1–2 or 5–7 for RoCE-related traffic groups.
- **ETS** (enhanced transition services) is an allocation of maximum bandwidth per traffic class.

DCQCN Overview

Some networking protocols (RoCE, for example) require droplessness. PFC is a mechanism for achieving droplessness in an L2 network, and DSCP-PFC is a mechanism for achieving it across distinct L2 networks. However, PFC is deficient in the following regards:

- When activated, PFC completely halts the traffic of the specified priority on the port, as opposed to reducing transmission rate.
- All traffic of the specified priority is affected, even if there is a subset of specific connections that are causing the congestion.
- PFC is a single-hop mechanism. That is, if a receiver experiences congestion and indicates the congestion through a PFC packet, only the nearest neighbor will react. When the neighbor experiences congestion (likely because it can no longer transmit), it also generates its own PFC. This generation is known as *pause propagation*. Pause propagation may cause inferior route utilization, because all buffers must congest before the transmitter is made aware of the problem.

DCQCN addresses all of these disadvantages. The ECN delivers congestion indication to the reaction point. The reaction point sends a CNP packet to the transmitter, which reacts by reducing its transmission rate and avoiding the congestion. DCQCN also specifies how the transmitter attempts to increase its transmission rate and use bandwidth effectively after congestion ceases. DCQCN is described in the 2015 SIGCOMM paper, *Congestion Control for Large-Scale RDMA Deployments*, located here:

<http://conferences.sigcomm.org/sigcomm/2015/pdf/papers/p523.pdf>

DCB-related Parameters

Use DCB to map priorities to traffic classes (priority groups). DCB also controls which priority groups are subject to PFC (lossless traffic), and the related bandwidth allocation (ETS).

Global Settings on RDMA Traffic

Global settings on RDMA traffic include configuration of vLAN priority, ECN, and DSCP.

Setting vLAN Priority on RDMA Traffic

Use an application to set the vLAN priority used by a specified RDMA Queue Pair (QP) when creating a QP. For example, the `ib_write_bw` benchmark controls the priority using the `-sl` parameter. When RDMA-CM (RDMA Communication Manager) is present, you may be unable to set the priority.

Another method to control the vLAN priority is to use the `rdma_glob_vlan_pri` node. This method affects QPs that are created after setting the value. For example, to set the vLAN priority number to 5 for subsequently created QPs, issue the following command:

```
./debugfs.sh -n eth0 -t rdma_glob_vlan_pri 5
```

Setting ECN on RDMA Traffic

Use the `rdma_glob_ecn` node to enable ECN for a specified RoCE priority. For example, to enable ECN on RoCE traffic using priority 5, issue the following command:

```
./debugfs.sh -n eth0 -t rdma_glob_ecn 1
```

This command is typically required when DCQCN is enabled.

Setting DSCP on RDMA Traffic

Use the `rdma_glob_dscp` node to control DSCP. For example, to set DSCP on RoCE traffic using priority 5, issue the following command:

```
./debugfs.sh -n eth0 -t rdma_glob_dscp 6
```

This command is typically required when DCQCN is enabled.

Configuring DSCP-PFC

Use `dscp_pfc` nodes to configure the `dscp->priority` association for PFC. You must enable the feature before you can add entries to the map. For example, to map DSCP value 6 to priority 5, issue the following commands:

```
./debugfs.sh -n eth0 -t dscp_pfc_enable 1  
./debugfs.sh -n eth0 -t dscp_pfc_set 6 5
```

Enabling DCQCN

To enable DCQCN for RoCE traffic, probe the qed driver with the `dcqcn_enable` module parameter. DCQCN requires enabled ECN indications (see [“Setting ECN on RDMA Traffic” on page 182](#)).

Configuring CNP

Congestion notification packets (CNPs) can have a separate configuration of vLAN priority and DSCP. Control these packets using the `dcqcn_cnp_dscp` and `dcqcn_cnp_vlan_priority` module parameters. For example:

```
modprobe qed dcqcn_cnp_dscp=10 dcqcn_cnp_vlan_priority=6
```

DCQCN Algorithm Parameters

[Table 7-5](#) lists the algorithm parameters for DCQCN.

Table 7-5. DCQCN Algorithm Parameters

Parameter	Description and Values
<code>dcqcn_cnp_send_timeout</code>	Minimal difference of send time between CNPs. Units are in microseconds. Values range between 50..500000.
<code>dcqcn_cnp_dscp</code>	DSCP value to be used on CNPs. Values range between 0..63.
<code>dcqcn_cnp_vlan_priority</code>	vLAN priority to be used on CNPs. Values range between 0..7. FCoE-Offload uses 3 and iSCSI-Offload-TLV generally uses 4. Marvell recommends that you specify a number from 1–2 or 5–7. Use this same value throughout the entire network.
<code>dcqcn_notification_point</code>	0 – Disable DCQCN notification point. 1 – Enable DCQCN notification point.
<code>dcqcn_reaction_point</code>	0 – Disable DCQCN reaction point. 1 – Enable DCQCN reaction point.

Table 7-5. DCQCN Algorithm Parameters (Continued)

Parameter	Description and Values
dcqcn_rl_bc_rate	Byte counter limit
dcqcn_rl_max_rate	Maximum rate in Mbps
dcqcn_rl_r_ai	Active increase rate in Mbps
dcqcn_rl_r_hai	Hyperactive increase rate in Mbps.
dcqcn_gd	Alpha update gain denominator. Set to 32 for 1/32, and so on.
dcqcn_k_us	Alpha update interval
dcqcn_timeout_us	DCQCN timeout

MAC Statistics

To view MAC statistics, including per-priority PFC statistics, issue the `phy_mac_stats` command. For example, to view statistics on port 1 issue the following command:

```
./debugfs.sh -n eth0 -d phy_mac_stat -P 1
```

Script Example

The following example can be used as a script:

```
# probe the driver with both reaction point and notification point enabled
# with cnp dscp set to 10 and cnp vlan priority set to 6
modprobe qed dcqcn_enable=1 dcqcn_notification_point=1 dcqcn_reaction_point=1
dcqcn_cnp_dscp=10 dcqcn_cnp_vlan_priority=6
modprobe qede

# dscp-pfc configuration (associating dscp values to priorities)
# This example is using two DCBX traffic class priorities to better demonstrate
DCQCN in operation
debugfs.sh -n ens6f0 -t dscp_pfc_enable 1
debugfs.sh -n ens6f0 -t dscp_pfc_set 20 5
debugfs.sh -n ens6f0 -t dscp_pfc_set 22 6

# static DCB configurations. 0x10 is static mode. Mark priorities 5 and 6 as
# subject to pfc
debugfs.sh -n ens6f0 -t dcbx_set_mode 0x10
debugfs.sh -n ens6f0 -t dcbx_set_pfc 5 1
debugfs.sh -n ens6f0 -t dcbx_set_pfc 6 1

# set roce global overrides for qp params. enable exn and open QPs with dscp 20
debugfs.sh -n ens6f0 -t rdma_glob_ecn 1
```

```
debugfs.sh -n ens6f0 -t rdma_glob_dscp 20

# open some QPs (DSCP 20)
ib_write_bw -d qedr0 -q 16 -F -x 1 --run_indefinitely

# change global dscp qp params
debugfs.sh -n ens6f0 -t rdma_glob_dscp 22

# open some more QPs (DSCP 22)
ib_write_bw -d qedr0 -q 16 -F -x 1 -p 8000 --run_indefinitely

# observe PFCs being generated on multiple priorities
debugfs.sh -n ens6f0 -d phy_mac_stat -P 0 | grep "Class Based Flow Control"
```

Limitations

DCQCN has the following limitations:

- DCQCN mode currently supports only up to 64 QPs.
- Marvell adapters can determine vLAN priority for PFC purposes from vLAN priority or from DSCP bits in the ToS field. However, in the presence of both, vLAN takes precedence.

8

iWARP Configuration

Internet wide area RDMA protocol (iWARP) is a computer networking protocol that implements RDMA for efficient data transfer over IP networks. iWARP is designed for multiple environments, including LANs, storage networks, data center networks, and WANs.

This chapter provides instructions for:

- [Preparing the Adapter for iWARP](#)
- [“Configuring iWARP on Windows” on page 187](#)
- [“Configuring iWARP on Linux” on page 191](#)

NOTE

Some iWARP features may not be fully enabled in the current release. For details, refer to [Appendix E Feature Constraints](#).

Preparing the Adapter for iWARP

This section provides instructions for preboot adapter iWARP configuration using the HII. For more information about preboot adapter configuration, see [Chapter 5 Adapter Preboot Configuration](#).

To configure iWARP through HII in Default mode:

1. Access the server BIOS System Setup, and then click **Device Settings**.
2. On the Device Settings page, select a port for the 25G 41000 Series Adapter.
3. On the Main Configuration Page for the selected adapter, click **NIC Configuration**.
4. On the NIC Configuration page:
 - a. Set the **NIC + RDMA Mode** to **Enabled**.
 - b. Set the **RDMA Protocol Support** to **RoCE/iWARP** or **iWARP**.
 - c. Click **Back**.
5. On the Main Configuration Page, click **Finish**.

6. In the Warning - Saving Changes message box, click **Yes** to save the configuration.
7. In the Success - Saving Changes message box, click **OK**.
8. Repeat [Step 2](#) through [Step 7](#) to configure the NIC and iWARP for the other ports.
9. To complete adapter preparation of both ports:
 - a. On the Device Settings page, click **Finish**.
 - b. On the main menu, click **Finish**.
 - c. Exit to reboot the system.

Proceed to [“Configuring iWARP on Windows” on page 187](#) or [“Configuring iWARP on Linux” on page 191](#).

Configuring iWARP on Windows

This section provides procedures for enabling iWARP, verifying RDMA, and verifying iWARP traffic on Windows. For a list of OSs that support iWARP, see [Table 7-1 on page 134](#).

For optimal iWARP performance, set the (per physical function) L2 Ethernet MTU size to be greater than 4,096. Additionally, set the network and target ports to an equivalent MTU size (to prevent the packets from fragmenting or dropping).

To enable iWARP on the Windows host and verify RDMA:

1. Enable iWARP on the Windows host.
 - a. Open the Windows Device Manager, and then open the 41000 Series Adapter NDIS Miniport Properties.
 - b. On the FastLinQ Adapter properties, click the **Advanced** tab.
 - c. On the Advanced page under **Property**, do the following:
 - Select **Network Direct Functionality**, and then select **Enabled** for the **Value**.
 - Select **NetworkDirect Technology**, and then select **iWARP** for the **Value**.
 - d. Click **OK** to save your changes and close the adapter properties.

2. Using Windows PowerShell, verify that RDMA is enabled. The `Get-NetAdapterRdma` command output (Figure 8-1) shows the adapters that support RDMA.

```
[172.28.41.178]: PS C:\Users\Administrator\Documents> Get-NetAdapterRdma
Name                               InterfaceDescription          Enabled
----                               -
SLOT 2 4 Port 2                    QLogic FastLinQ QL41262-DE 25GbE Adap... True
SLOT 2 3 Port 1                    QLogic FastLinQ QL41262-DE 25GbE Adap... True
```

Figure 8-1. Windows PowerShell Command: `Get-NetAdapterRdma`

3. Using Windows PowerShell, verify that `NetworkDirect` is enabled. The `Get-NetOffloadGlobalSetting` command output (Figure 8-2) shows `NetworkDirect` as Enabled.

```
PS C:\Users\Administrator> Get-NetOffloadGlobalSetting
ReceiveSideScaling      : Enabled
ReceiveSegmentCoalescing : Enabled
Chimney                 : Disabled
TaskOffload             : Enabled
NetworkDirect           : Enabled
NetworkDirectAcrossIPSubnets : Blocked
PacketCoalescingFilter  : Disabled
```

Figure 8-2. Windows PowerShell Command: `Get-NetOffloadGlobalSetting`

To verify iWARP traffic:

1. Map SMB drives and run iWARP traffic.
2. Launch Performance Monitor (Perfmon).
3. In the Add Counters dialog box, click **RDMA Activity**, and then select the adapter instances.

Figure 8-3 shows an example.

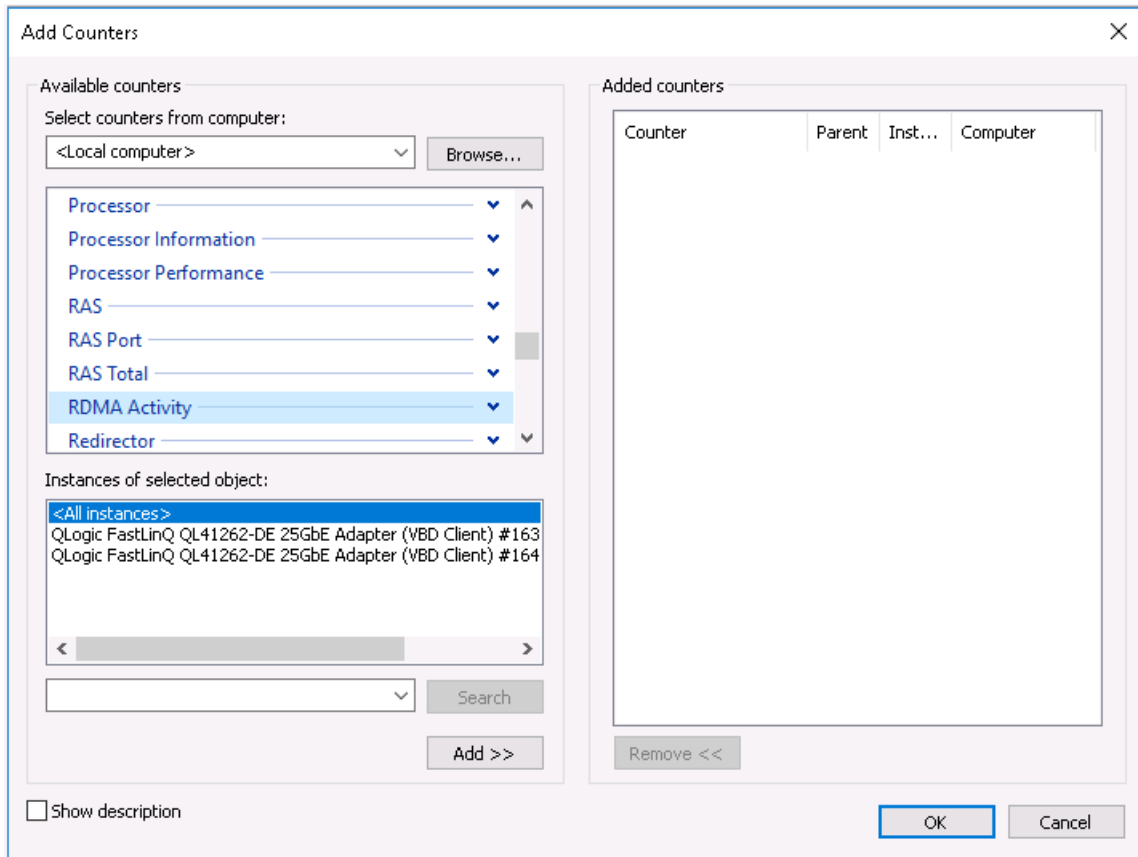


Figure 8-3. Perfmon: Add Counters

If iWARP traffic is running, counters appear as shown in the [Figure 8-4](#) example.

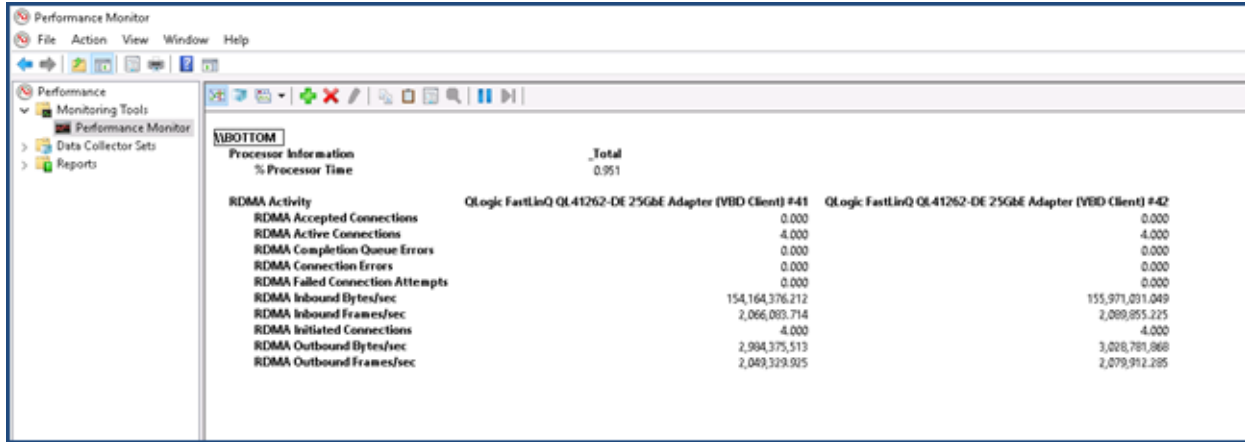


Figure 8-4. Perfmon: Verifying iWARP Traffic

NOTE

For more information on how to view Marvell RDMA counters in Windows, see [“Viewing RDMA Counters”](#) on page 144.

4. To verify the SMB connection:
 - a. At a command prompt, issue the `net use` command as follows:

```
C:\Users\Administrator> net use
New connections will be remembered.
```

```
Status      Local      Remote      Network
-----
OK          F:         \\192.168.10.10\Share1  Microsoft Windows Network
The command completed successfully.
```

- b. Issue the `netstat -xan` command as follows, where `Share1` is mapped as an SMB share:

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

Mode   IfIndex Type           Local Address           Foreign Address         PID
-----
Kernel 56 Connection 192.168.11.20:16159 192.168.11.10:445      0
Kernel 56 Connection 192.168.11.20:15903 192.168.11.10:445      0
Kernel 56 Connection 192.168.11.20:16159 192.168.11.10:445      0
```

```
Kernel      56 Connection 192.168.11.20:15903 192.168.11.10:445 0
Kernel      60 Listener  [fe80::e11d:9ab5:a47d:4f0a%56]:445 NA 0
Kernel      60 Listener  192.168.11.20:445 NA 0
Kernel      60 Listener  [fe80::71ea:bdd2:ae41:b95f%60]:445 NA 0
Kernel      60 Listener  192.168.11.20:16159 192.168.11.10:445 0
```

Configuring iWARP on Linux

Marvell 41000 Series Adapters support iWARP on the Linux Open Fabric Enterprise Distributions (OFEDs) listed in [Table 7-1 on page 134](#).

iWARP configuration on a Linux system includes the following:

- [Installing the Driver](#)
- [Configuring iWARP and RoCE](#)
- [Detecting the Device](#)
- [Supported iWARP Applications](#)
- [Running PerfTest for iWARP](#)
- [Configuring NFS-RDMA](#)

Installing the Driver

Install the RDMA drivers as shown in [Chapter 3 Driver Installation](#).

Configuring iWARP and RoCE

NOTE

This procedure applies only if you previously selected **iWARP+RoCE** as the value for the RDMA Protocol Support parameter during preboot configuration using HII (see [Configuring NIC Parameters, Step 5 on page 63](#)).

To enable iWARP and RoCE:

1. Unload all FastLinQ drivers as follows:

```
# modprobe -r qedr or modprobe -r qede
```
2. If only RoCE or only iWARP was enabled in the pre-boot UEFI HII setup, this step is not needed, since only that RDMA type is enabled. If both RoCE and iWARP are enabled in pre-boot UEFI HII, then RoCE is enabled, if not changed to iWARP by the following `qed` driver module command.

Use the following command syntax to change the RDMA protocol by loading the `qed` driver with a port interface PCI ID (`xx:xx.x`) and an RDMA protocol value (`p`).

```
# modprobe -v qed rdma_protocol_map=<xx:xx.x-p>
```

The RDMA protocol (`p`) values are as follows:

- 0—Accept the default (RoCE)
- 1—No RDMA
- 2—RoCE
- 3—iWARP

For example, to change the interface on the port given by 04:00.0 from RoCE to iWARP, issue the following command:

```
# modprobe -v qed rdma_protocol_map=04:00.0-3
```

3. Load the RDMA driver by issuing the following command:

```
# modprobe -v qedr
```

The following example shows the command entries to change the RDMA protocol to iWARP on multiple NPar interfaces:

```
# modprobe qed rdma_protocol_map=04:00.1-3,04:00.3-3,04:00.5-3,  
04:00.7-3,04:01.1-3,04:01.3-3,04:01.5-3,04:01.7-3  
# modprobe -v qedr  
# ibv_devinfo |grep iWARP  
transport: iWARP (1)  
transport: iWARP (1)  
transport: iWARP (1)  
transport: iWARP (1)  
transport: iWARP (1)  
transport: iWARP (1)  
transport: iWARP (1)  
transport: iWARP (1)
```

Detecting the Device

To detect the device:

1. To verify whether RDMA devices are detected, view the `dmesg` logs:

```
# dmesg |grep qedr  
[10500.191047] qedr 0000:04:00.0: registered qedr0  
[10500.221726] qedr 0000:04:00.1: registered qedr1
```

2. Issue the `ibv_devinfo` command, and then verify the transport type.

If the command is successful, each PCI function will show a separate `hca_id`. For example (if checking the second port of the above dual-port adapter):

```
[root@localhost ~]# ibv_devinfo -d qedr1
hca_id: qedr1
      transport:                iWARP (1)
      fw_ver:                    8.14.7.0
      node_guid:                  020e:1eff:fec4:c06e
      sys_image_guid:            020e:1eff:fec4:c06e
      vendor_id:                  0x1077
      vendor_part_id:            5718
      hw_ver:                     0x0
      phys_port_cnt:              1
      port:      1
      state:                PORT_ACTIVE (4)
      max_mtu:                4096 (5)
      active_mtu:            1024 (3)
      sm_lid:                  0
      port_lid:                0
      port_lmc:                0x00
      link_layer:              Ethernet
```

Supported iWARP Applications

Linux-supported RDMA applications for iWARP include the following:

- `ibv_devinfo`, `ib_devices`
- `ib_send_bw/lat`, `ib_write_bw/lat`, `ib_read_bw/lat`, `ib_atomic_bw/lat`
For iWARP, all applications must use the RDMA communication manager (`rdma_cm`) using the `-R` option.
- `rdma_server`, `rdma_client`
- `rdma_xserver`, `rdma_xclient`
- `rping`
- NFS over RDMA (NFSoverRDMA)
- iSER (for details, see [Chapter 9 iSER Configuration](#))
- NVMe-oF (for details, see [Chapter 13 NVMe-oF Configuration with RDMA](#))

Running Perftest for iWARP

All perftest tools are supported over the iWARP transport type. You must run the tools using the RDMA connection manager (with the `-R` option).

For optimal iWARP performance, set the (per physical function) L2 Ethernet MTU size to be greater than 4,096. Additionally, set the network and target ports to an equivalent MTU size (to prevent the packets from fragmenting or dropping).

Example:

1. On one server, issue the following command (using the second port in this example):

```
# ib_send_bw -d qedr1 -F -R
```
2. On one client, issue the following command (using the second port in this example):

```
[root@localhost ~]# ib_send_bw -d qedr1 -F -R 192.168.11.3
```

```
-----  
                        Send BW Test  
Dual-port      : OFF           Device      : qedr1  
Number of qps  : 1            Transport type : IW  
Connection type : RC           Using SRQ    : OFF  
TX depth       : 128  
CQ Moderation  : 100  
Mtu            : 1024[B]  
Link type      : Ethernet  
GID index      : 0  
Max inline data : 0[B]  
rdma_cm QPs    : ON  
Data ex. method : rdma_cm  
-----  
local address: LID 0000 QPN 0x0192 PSN 0xcde932  
GID: 00:14:30:196:192:110:00:00:00:00:00:00:00:00:00:00  
remote address: LID 0000 QPN 0x0098 PSN 0x46fffc  
GID: 00:14:30:196:195:62:00:00:00:00:00:00:00:00:00:00  
-----  
#bytes  #iterations  BW peak[MB/sec]  BW average[MB/sec]  MsgRate[Mpps]  
65536   1000          2250.38          2250.36              0.036006  
-----
```


NOTE

For latency applications (send/write), if the perftest version is the latest (for example, `perftest-3.0-0.21.g21dc344.x86_64.rpm`), use the supported inline size value: 0-128.

Configuring NFS-RDMA

NFS-RDMA for iWARP includes both server and client configuration steps.

To configure the NFS server:

1. Create an `nfs-server` directory and grant permission by issuing the following commands:

```
# mkdir /tmp/nfs-server
# chmod 777 /tmp/nfs-server
```

2. In the `/etc/exports` file for the directories that you must export using NFS-RDMA on the server, make the following entry:

```
/tmp/nfs-server *(rw,fsid=0,async,insecure,no_root_squash)
```

Ensure that you use a different file system identification (FSID) for each directory that you export.

3. Load the `svcrdma` module as follows:

```
# modprobe svcrdma
```

4. Load the service as follows:

For SLES, enable and start the NFS server alias:

```
# systemctl enable|start|status nfsserver
```

For RHEL, enable and start the NFS server and services:

```
# systemctl enable|start|status nfs
```

5. Include the default RDMA port 20049 into this file as follows:

```
# echo rdma 20049 > /proc/fs/nfsd/portlist
```

6. To make local directories available for NFS clients to mount, issue the `exportfs` command as follows:

```
# exportfs -v
```

To configure the NFS client:

NOTE

This procedure for NFS client configuration also applies to RoCE.

1. Create an `nfs-client` directory and grant permission by issuing the following commands:

```
# mkdir /tmp/nfs-client
# chmod 777 /tmp/nfs-client
```

2. Load the `xprtrdma` module as follows:

```
# modprobe xprtrdma
```

3. Mount the NFS file system as appropriate for your version:

For NFS Version 3:

```
# mount -o rdma,port=20049 192.168.2.4:/tmp/nfs-server
/tmp/nfs-client
```

For NFS Version 4:

```
# mount -t nfs4 -o rdma,port=20049 192.168.2.4:/tmp/nfs-server
/tmp/nfs-client
```

NOTE

The default port for NFSoRDMA is 20049. However, any other port that is aligned with the NFS client will also work.

4. Verify that the file system is mounted by issuing the `mount` command. Ensure that the RDMA port and file system versions are correct.

```
# mount |grep rdma
```

9 iSER Configuration

This chapter provides procedures for configuring iSCSI Extensions for RDMA (iSER) for Linux (RHEL and SLES) and VMware ESXi 6.7/7.0, including:

- [Before You Begin](#)
- [“Configuring iSER for RHEL” on page 198](#)
- [“Configuring iSER for SLES 15 and Later” on page 201](#)
- [“Using iSER with iWARP on RHEL and SLES” on page 202](#)
- [“Optimizing Linux Performance” on page 203](#)
- [“Configuring iSER on ESXi 6.7 and ESXi 7.0” on page 205](#)

Before You Begin

As you prepare to configure iSER, consider the following:

- iSER is supported only in inbox OFED for the following operating systems:
 - RHEL 8.x
 - RHEL 7.8 and later
 - SLES 15 SP1 and later
 - VMware ESXi 6.7 U1
 - VMware ESXi 7.0
- After logging into the targets or while running I/O traffic, unloading the Linux RoCE qedr driver may crash the system.
- While running I/O, performing interface down/up tests or performing cable pull-tests can cause driver or iSER module errors that may crash the system. If this happens, reboot the system.

Configuring iSER for RHEL

To configure iSER for RHEL:

1. Install inbox OFED as described in [“RoCE Configuration for RHEL” on page 158](#).

NOTE

Out-of-box OFEDs are not supported for iSER because the `ib_isert` module is not available in the out-of-box OFED versions. The inbox `ib_isert` module does not work with any out-of-box OFED versions.

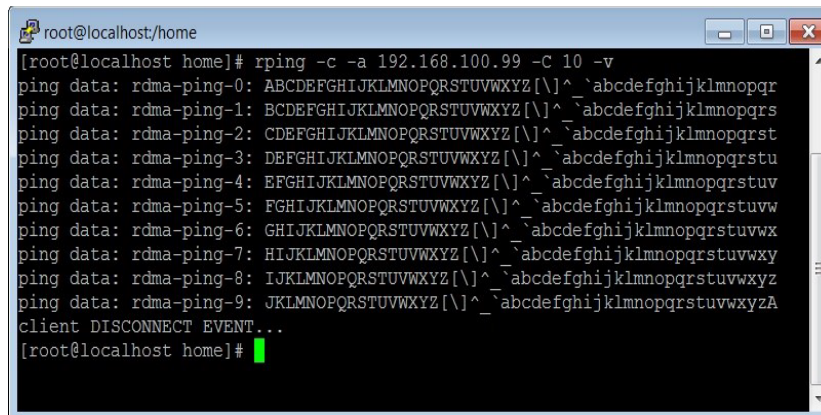
2. Unload any existing FastLinQ drivers as described in [“Removing the Linux Drivers” on page 11](#).
3. Install the latest FastLinQ driver and `libqedr` packages as described in [“Installing the Linux Drivers with RDMA” on page 15](#).
4. Load the RDMA services as follows:

```
systemctl start rdma
modprobe qedr
modprobe ib_iser
modprobe ib_isert
```
5. Verify that all RDMA and iSER modules are loaded on the initiator and target devices by issuing the `lsmod | grep qed` and `lsmod | grep iser` commands.
6. Verify that there are separate `hca_id` instances by issuing the `ibv_devinfo` command, as shown in [Step 6 on page 161](#).
7. Check the RDMA connection on the initiator device and the target device.
 - a. On the initiator device, issue the following command:

```
rping -s -C 10 -v
```
 - b. On the target device, issue the following command:

```
rping -c -a 192.168.100.99 -C 10 -v
```

Figure 9-1 shows an example of a successful RDMA ping.



```
root@localhost/home
[root@localhost home]# rping -c -a 192.168.100.99 -c 10 -v
ping data: rdma-ping-0: ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-1: BCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-2: CDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-3: DEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-4: EFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-5: FGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-6: GHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-7: HIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-8: IJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
ping data: rdma-ping-9: JKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyzA
Client DISCONNECT EVENT...
[root@localhost home]#
```

Figure 9-1. RDMA Ping Successful

8. You can use a Linux TCM-LIO target to test iSER. The setup is the same for any iSCSI target, except that you issue the command `enable_iser Boolean=true` on the applicable portals. The portal instances are identified as `iser` in Figure 9-2.



```
/iscsi/ign.20.../tpg1/portals> cd 192.168.100.99:3260
/iscsi/ign.20...8.100.99:3260> enable_iser boolean=true
iSER enable now: True
/iscsi/ign.20...8.100.99:3260>
/iscsi/ign.20...8.100.99:3260> cd /
/> ls
o- / ..... [..]
o- backstore ..... [..]
| o- block ..... [Storage Objects: 0]
| o- fileio ..... [Storage Objects: 0]
| o- pscsi ..... [Storage Objects: 0]
| o- ramdisk ..... [Storage Objects: 1]
| o- raml ..... [nullio (512.0MiB) activated]
o- iscsi ..... [Targets: 1]
| o- ign.2015-06.test.target1 ..... [TPGs: 1]
| | o- tpg1 ..... [gen-acls, no-auth]
| | | o- acls ..... [ACLS: 0]
| | | o- luns ..... [LUNs: 1]
| | | | o- lun0 ..... [ramdisk/raml]
| | | o- portals ..... [Portals: 1]
| | | | o- 192.168.100.99:3260 ..... [iser]
o- loopback ..... [Targets: 0]
o- srpt ..... [Targets: 0]
/>
```

Figure 9-2. iSER Portal Instances

9. Install Linux iSCSI Initiator Utilities using the `yum install iscsi-initiator-utils` commands.
 - a. To discover the iSER target, issue the `iscsiadm` command. For example:
`iscsiadm -m discovery -t st -p 192.168.100.99:3260`

- b. To change the transport mode to iSER, issue the `iscsiadm` command. For example:

```
iscsiadm -m node -T iqn.2015-06.test.target1 -o update -n iface.transport_name -v iser
```
- c. To connect to or log in to the iSER target, issue the `iscsiadm` command. For example:

```
iscsiadm -m node -l -p 192.168.100.99:3260 -T iqn.2015-06.test.target1
```
- d. Confirm that the `Iface Transport` is `iser` in the target connection, as shown in [Figure 9-3](#). Issue the `iscsiadm` command; for example:

```
iscsiadm -m session -P2
```

```
[root@localhost ~]# iscsiadm -m discovery -t st -p 192.168.100.99:3260
192.168.100.99:3260,1 iqn.2015-06.test.target1
192.168.100.99:3260,1 iqn.2015-06.test.target1
[root@localhost ~]#
[root@localhost ~]# iscsiadm -m node -T iqn.2015-06.test.target1 -o update -n iface.transport_name -v iser
[root@localhost ~]#
[root@localhost ~]#
[root@localhost ~]# iscsiadm -m node -l -p 192.168.100.99:3260 -T iqn.2015-06.test.target1
Logging in to [iface: default, target: iqn.2015-06.test.target1, portal: 192.168.100.99,3260] (multiple)
Login to [iface: default, target: iqn.2015-06.test.target1, portal: 192.168.100.99,3260] successful.
[root@localhost ~]#
[root@localhost ~]# iscsiadm -m session -P2
Target: iqn.2015-06.test.target1 (non-flash)
Current Portal: 192.168.100.99:3260,1
Persistent Portal: 192.168.100.99:3260,1
*****
Interface:
*****
Iface Name: default
Iface Transport: iser
Iface Initiatorname: iqn.1994-05.com.redhat:c672dfb8b08f
Iface IPaddress: <empty>
Iface HWaddress: <empty>
Iface Netdev: <empty>
SID: 33
iSCSI Connection State: LOGGED IN
iSCSI Session State: LOGGED_IN
Internal iscsid Session State: NO CHANGE
*****
Timeouts:
*****
Recovery Timeout: 120
```

Figure 9-3. Iface Transport Confirmed

- e. To check for a new iSCSI device, as shown in [Figure 9-4](#), issue the `lsscsi` command.

```
[root@localhost ~]# lsscsi
[6:0:0:0]   disk      HP          LOGICAL VOLUME  1.18  /dev/sdb
[6:0:0:1]   disk      HP          LOGICAL VOLUME  1.18  /dev/sda
[6:0:0:3]   disk      HP          LOGICAL VOLUME  1.18  /dev/sdc
[6:3:0:0]   storage  HP          P440ar          1.18  -
[39:0:0:0]  disk      LIO-ORG    ram1             4.0   /dev/sdd
[root@localhost ~]#
```

Figure 9-4. Checking for New iSCSI Device

Configuring iSER for SLES 15 and Later

Because the `targetcli` is not in box on SLES 15 and later, you must complete the following procedure.

To configure iSER for SLES 15 and later:

1. Install `targetcli`.
Load the SLES Package DVD and install `targetcli` by issuing the following Zypper command, which installs all the dependency packages:

```
# zypper install python3-targetcli-fb
```
2. Before starting the `targetcli`, load all RoCE device drivers and iSER modules as follows:

```
# modprobe qed
# modprobe qede
# modprobe qedr
# modprobe ib_iser (initiator)
# modprobe ib_isert (target)
```
3. Before configuring iSER targets, configure NIC interfaces and run L2 and RoCE traffic, as described in [Step 7](#) on [page 161](#).
4. Insert the SLES Package DVD and install the `targetcli` utility. This command also installs all the dependency packages.

```
# zypper install python3-targetcli-fb
```
5. Start the `targetcli` utility, and configure your targets on the iSER target system.

NOTE

`targetcli` versions are different in RHEL and SLES. Be sure to use the proper backstores to configure your targets:

- RHEL uses `ramdisk`
 - SLES uses `rd_mcp`
-

To configure an initiator for iWARP:

1. To discover the iSER LIO target using port 3261, issue the `iscsiadm` command as follows:

```
# iscsiadm -m discovery -t st -p 192.168.21.4:3261 -I iser  
192.168.21.4:3261,1 iqn.2017-04.com.org.iserport1.target1
```

2. Change the transport mode to `iser` as follows:

```
# iscsiadm -m node -o update -T iqn.2017-04.com.org.iserport1.target1 -n  
iface.transport_name -v iser
```

3. Log into the target using port 3261:

```
# iscsiadm -m node -l -p 192.168.21.4:3261 -T iqn.2017-04.com.org.iserport1.target1  
Logging in to [iface: iser, target: iqn.2017-04.com.org.iserport1.target1,  
portal: 192.168.21.4,3261] (multiple)  
Login to [iface: iser, target: iqn.2017-04.com.org.iserport1.target1, portal:  
192.168.21.4,3261] successful.
```

4. Ensure that those LUNs are visible by issuing the following command:

```
# ls SCSI  
[1:0:0:0] storage HP P440ar 3.56 -  
[1:1:0:0] disk HP LOGICAL VOLUME 3.56 /dev/sda  
[6:0:0:0] cd/dvd hp DVD-ROM DUD0N UMD0 /dev/sr0  
[7:0:0:0] disk LIO-ORG Ramdisk1-1 4.0 /dev/sdb
```

Optimizing Linux Performance

Consider the following Linux performance configuration enhancements described in this section.

- [Configuring CPUs to Maximum Performance Mode](#)
- [Configuring Kernel sysctl Settings](#)
- [Configuring IRQ Affinity Settings](#)
- [Configuring Block Device Staging](#)

Configuring CPUs to Maximum Performance Mode

Configure the CPU scaling governor to performance by using the following script to set all CPUs to maximum performance mode:

```
for CPUFREQ in  
/sys/devices/system/cpu/cpu*/cpufreq/scaling_governor; do [ -f  
$CPUFREQ ] || continue; echo -n performance > $CPUFREQ; done
```

Verify that all CPU cores are set to maximum performance mode by issuing the following command:

```
cat /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
```

Configuring Kernel sysctl Settings

Set the kernel sysctl settings as follows:

```
sysctl -w net.ipv4.tcp_mem="4194304 4194304 4194304"  
sysctl -w net.ipv4.tcp_wmem="4096 65536 4194304"  
sysctl -w net.ipv4.tcp_rmem="4096 87380 4194304"  
sysctl -w net.core.wmem_max=4194304  
sysctl -w net.core.rmem_max=4194304  
sysctl -w net.core.wmem_default=4194304  
sysctl -w net.core.rmem_default=4194304  
sysctl -w net.core.netdev_max_backlog=250000  
sysctl -w net.ipv4.tcp_timestamps=0  
sysctl -w net.ipv4.tcp_sack=1  
sysctl -w net.ipv4.tcp_low_latency=1  
sysctl -w net.ipv4.tcp_adv_win_scale=1  
echo 0 > /proc/sys/vm/nr_hugepages
```

Configuring IRQ Affinity Settings

The following example sets CPU core 0, 1, 2, and 3 to interrupt request (IRQ) XX, YY, ZZ, and XYZ respectively. Perform these steps for each IRQ assigned to a port (default is eight queues per port).

```
systemctl disable irqbalance  
systemctl stop irqbalance  
cat /proc/interrupts | grep qedr Shows IRQ assigned to each port queue  
echo 1 > /proc/irq/XX/smp_affinity_list  
echo 2 > /proc/irq/YY/smp_affinity_list  
echo 4 > /proc/irq/ZZ/smp_affinity_list  
echo 8 > /proc/irq/XYZ/smp_affinity_list
```

Configuring Block Device Staging

Set the block device staging settings for each iSCSI device or target as follows:

```
echo noop > /sys/block/sdd/queue/scheduler  
echo 2 > /sys/block/sdd/queue/nomerges  
echo 0 > /sys/block/sdd/queue/add_random  
echo 1 > /sys/block/sdd/queue/rq_affinity
```

Configuring iSER on ESXi 6.7 and ESXi 7.0

This section provides information for configuring iSER for VMware ESXi 6.7 and ESXi 7.0.

Before You Begin

Before you configure iSER for ESXi 6.7/7.0, ensure that the following is complete:

- The CNA package with NIC and RoCE drivers is installed on the ESXi 6.7/7.0 system and the devices are listed. To view RDMA devices, issue the following command:

```
esxcli rdma device list
```

Name	Driver	State	MTU	Speed	Paired Uplink	Description
vmrdma0	qedrntv	Active	1024	40 Gbps	vmnic4	QLogic FastLinQ QL45xxx RDMA Interface
vmrdma1	qedrntv	Active	1024	40 Gbps	vmnic5	QLogic FastLinQ QL45xxx RDMA Interface

```
[root@localhost:~] esxcfg-vmknics -l
```

Interface	Port	Group/DVPort/Opaque	Network	IP Family	IP Address	Netmask	Broadcast	MAC Address	MTU	TSO	MSS	Enabled	Type
vmk0		Management Network		IPv4	172.28.12.94	255.255.240.0	172.28.15.255	e0:db:55:0c:5f:94	1500	65535	true		DHCP
vmk0		Management Network		IPv6	fe80::e2db:55ff:fe0c:5f94			e0:db:55:0c:5f:94	1500	65535	true		STATIC, PREFERRED

- The iSER target is configured to communicate with the iSER initiator.

Configuring iSER for ESXi 6.7 and ESXi 7.0

To configure iSER for ESXi 6.7/7.0:

1. Add iSER devices by issuing the following commands:

```
esxcli rdma iser add
esxcli iscsi adapter list
```

Adapter	Driver	State	UID	Description
vmhba64	iser	unbound	iscsi.vmhba64	VMware iSCSI over RDMA (iSER) Adapter
vmhba65	iser	unbound	iscsi.vmhba65	VMware iSCSI over RDMA (iSER) Adapter

2. Disable the firewall as follows.

```
esxcli network firewall set --enabled=false
esxcli network firewall unload
vsish -e set /system/modules/iscsi_trans/loglevels/iscsitrans 0
```

9-iSER Configuration

Configuring iSER on ESXi 6.7 and ESXi 7.0

```
vsish -e set /system/modules/iser/loglevels/debug 4
```

3. Create a standard vSwitch VMkernel port group and assign the IP:

```
esxcli network vswitch standard add -v vSwitch_iser1
```

```
esxcfg-nics -l
```

Name	PCI	Driver	Link	Speed	Duplex	MAC Address	MTU	Description
vmnic0	0000:01:00.0	ntg3	Up	1000Mbps	Full	e0:db:55:0c:5f:94	1500	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic1	0000:01:00.1	ntg3	Down	0Mbps	Half	e0:db:55:0c:5f:95	1500	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic2	0000:02:00.0	ntg3	Down	0Mbps	Half	e0:db:55:0c:5f:96	1500	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic3	0000:02:00.1	ntg3	Down	0Mbps	Half	e0:db:55:0c:5f:97	1500	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic4	0000:42:00.0	qedentv	Up	40000Mbps	Full	00:0e:1e:d5:f6:a2	1500	QLogic Corp. QLogic FastLinQ QL41xxx 10/25/40/50/100 GbE Ethernet Adapter
vmnic5	0000:42:00.1	qedentv	Up	40000Mbps	Full	00:0e:1e:d5:f6:a3	1500	QLogic Corp. QLogic FastLinQ QL41xxx 10/25/40/50/100 GbE Ethernet Adapter

```
esxcli network vswitch standard uplink add -u vmnic5 -v vSwitch_iser1
```

```
esxcli network vswitch standard portgroup add -p "rdma_group1" -v vSwitch_iser1
```

```
esxcli network ip interface add -i vmk1 -p "rdma_group1"
```

```
esxcli network ip interface ipv4 set -i vmk1 -I 192.168.10.100 -N 255.255.255.0 -t static
```

```
esxcfg-vswitch -p "rdma_group1" -v 4095 vSwitch_iser1
```

```
esxcli iscsi networkportal add -A vmhba67 -n vmk1
```

```
esxcli iscsi networkportal list
```

```
esxcli iscsi adapter get -A vmhba65
```

```
vmhba65
```

```
Name: iqn.1998-01.com.vmware:localhost.punelab.qlogic.com qlogic.org qlogic.com
mv.qlogic.com:1846573170:65
```

```
Alias: iser-vmnic5
```

```
Vendor: VMware
```

```
Model: VMware iSCSI over RDMA (iSER) Adapter
```

```
Description: VMware iSCSI over RDMA (iSER) Adapter
```

```
Serial Number: vmnic5
```

```
Hardware Version:
```

```
Asic Version:
```

```
Firmware Version:
```

```
Option Rom Version:
```

```
Driver Name: iser-vmnic5
```

```
Driver Version:
```

```
TCP Protocol Supported: false
```

9-iSER Configuration

Configuring iSER on ESXi 6.7 and ESXi 7.0

```
Bidirectional Transfers Supported: false
Maximum Cdb Length: 64
Can Be NIC: true
Is NIC: true
Is Initiator: true
Is Target: false
Using TCP Offload Engine: true
Using ISCSI Offload Engine: true
```

4. Add the target to the iSER initiator as follows:

```
esxcli iscsi adapter target list
esxcli iscsi adapter discovery sendtarget add -A vmhba65 -a 192.168.10.11
esxcli iscsi adapter target list
```

Adapter	Target	Alias	Discovery Method	Last Error
vmhba65	iqn.2015-06.test.target1		SENDTARGETS	No Error

```
esxcli storage core adapter rescan --adapter vmhba65
```

5. List the attached target as follows:

```
esxcfg-scsidevs -l
mpx.vmhba0:C0:T4:L0
  Device Type: CD-ROM
  Size: 0 MB
  Display Name: Local TSSTcorp CD-ROM (mpx.vmhba0:C0:T4:L0)
  Multipath Plugin: NMP
  Console Device: /vmfs/devices/cdrom/mpx.vmhba0:C0:T4:L0
  Devfs Path: /vmfs/devices/cdrom/mpx.vmhba0:C0:T4:L0
  Vendor: TSSTcorp Model: DVD-ROM SN-108BB Revis: D150
  SCSI Level: 5 Is Pseudo: false Status: on
  Is RDM Capable: false Is Removable: true
  Is Local: true Is SSD: false
  Other Names:
    vml.0005000000766d686261303a343a30
  VAAI Status: unsupported
naa.6001405e81ae36b771c418b89c85dae0
  Device Type: Direct-Access
  Size: 512 MB
  Display Name: LIO-ORG iSCSI Disk (naa.6001405e81ae36b771c418b89c85dae0)
  Multipath Plugin: NMP
  Console Device: /vmfs/devices/disks/naa.6001405e81ae36b771c418b89c85dae0
  Devfs Path: /vmfs/devices/disks/naa.6001405e81ae36b771c418b89c85dae0
  Vendor: LIO-ORG Model: ram1 Revis: 4.0
  SCSI Level: 5 Is Pseudo: false Status: degraded
  Is RDM Capable: true Is Removable: false
  Is Local: false Is SSD: false
  Other Names:
    vml.02000000006001405e81ae36b771c418b89c85dae072616d312020
```

9-iSER Configuration

Configuring iSER on ESXi 6.7 and ESXi 7.0

```
VAAI Status: supported  
naa.690b11c0159d050018255e2d1d59b612
```

10 iSCSI Configuration

This chapter provides the following iSCSI configuration information:

- [iSCSI Boot](#)
- [“iSCSI Offload in Windows Server” on page 210](#)
- [“iSCSI Offload in Linux Environments” on page 218](#)
- [“iSCSI Offload in VMware ESXi” on page 221](#)

NOTE

Some iSCSI features may not be fully enabled in the current release. For details, refer to [Appendix E Feature Constraints](#).

To enable iSCSI-Offload mode, see [“Configuring Partitions” on page 73 \(Figure 5-19\)](#).

HII level target discovery configuration is not required for iSCSI-Offload LUN discovery in the local boot environment.

iSCSI Boot

Marvell 4xxxx Series gigabit Ethernet (GbE) adapters support iSCSI boot to enable network boot of operating systems to diskless systems. iSCSI boot allows a Windows, Linux, or VMware operating system to boot from an iSCSI target machine located remotely over a standard IP network.

Jumbo frames with iSCSI boot are supported only on Windows OSs, when the adapter is used as either an NDIS or HBA offload device.

For iSCSI boot from SAN information, see [Chapter 6 Boot from SAN Configuration](#).

iSCSI Offload in Windows Server

iSCSI offload is a technology that offloads iSCSI protocol processing overhead from host processors to the iSCSI HBA. iSCSI offload increases network performance and throughput while helping to optimize server processor use. This section covers how to configure the Windows iSCSI offload feature for the Marvell 41000 Series Adapters.

With the proper iSCSI offload licensing, you can configure your iSCSI-capable 41000 Series Adapter to offload iSCSI processing from the host processor. The following sections describe how to enable the system to take advantage of Marvell's iSCSI offload feature:

- [Installing Marvell Drivers](#)
- [Installing the Microsoft iSCSI Initiator](#)
- [Configuring Microsoft Initiator to Use Marvell's iSCSI Offload](#)
- [iSCSI Offload FAQs](#)
- [Windows Server 2016 and 2019/Azure Stack HCI iSCSI Boot Installation](#)
- [iSCSI Crash Dump](#)

Installing Marvell Drivers

Install the Windows drivers as described in [“Installing Windows Driver Software” on page 21](#).

Installing the Microsoft iSCSI Initiator

Launch the Microsoft iSCSI initiator applet. At the first launch, the system prompts for an automatic service start. Confirm the selection for the applet to launch.

Configuring Microsoft Initiator to Use Marvell's iSCSI Offload

After the IP address is configured for the iSCSI adapter, you must use Microsoft Initiator to configure and add a connection to the iSCSI target using the Marvell FastLinQ iSCSI adapter. For more details on Microsoft Initiator, see the Microsoft user guide.

To configure Microsoft Initiator:

1. Open Microsoft Initiator.
2. To configure the initiator IQN name according to your setup, follow these steps:
 - a. On the iSCSI Initiator Properties, click the **Configuration** tab.
 - b. On the Configuration page ([Figure 10-1](#)), click **Change** next to **To modify the initiator name**.

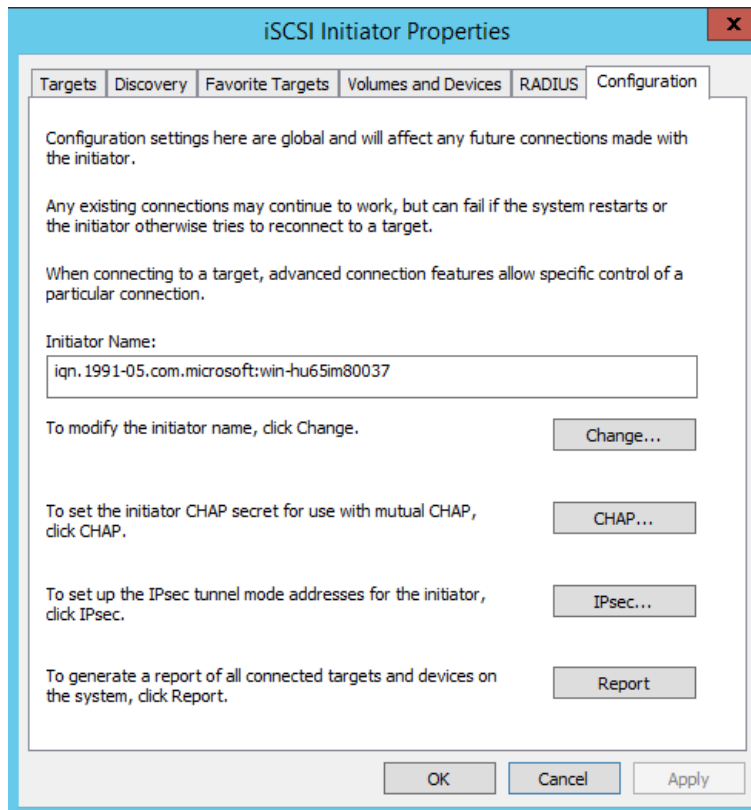


Figure 10-1. iSCSI Initiator Properties, Configuration Page

- c. In the iSCSI Initiator Name dialog box, type the new initiator IQN name, and then click **OK**. (Figure 10-2)

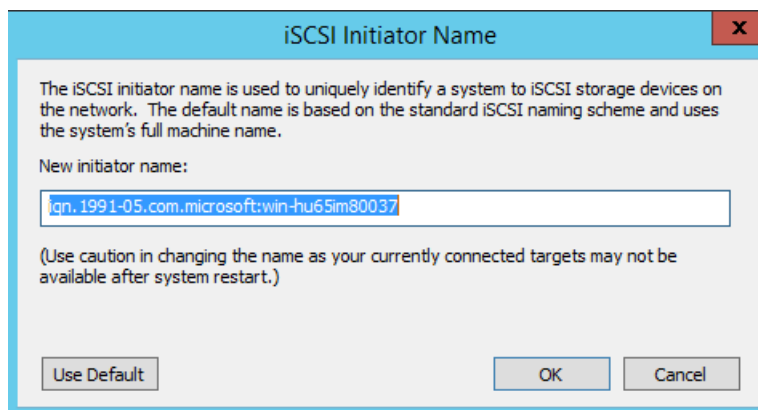


Figure 10-2. iSCSI Initiator Node Name Change

3. On the iSCSI Initiator Properties, click the **Discovery** tab.

4. On the Discovery page (Figure 10-3) under **Target portals**, click **Discover Portal**.

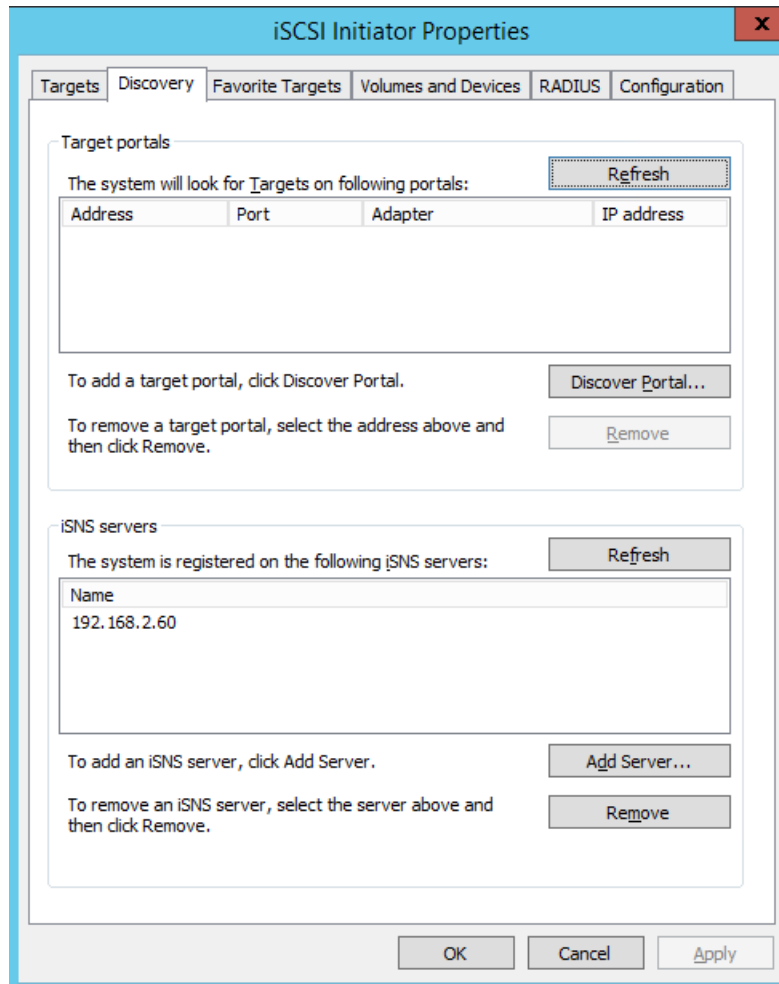


Figure 10-3. iSCSI Initiator—Discover Target Portal

5. In the Discover Target Portal dialog box (Figure 10-4):
 - a. In the **IP address or DNS name** box, type the IP address of the target.
 - b. Click **Advanced**.

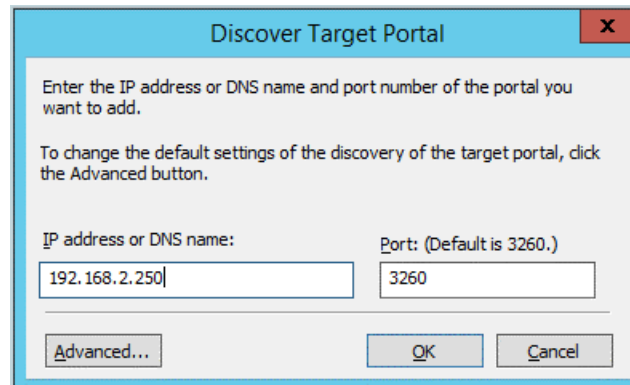


Figure 10-4. Target Portal IP Address

6. In the Advanced Settings dialog box (Figure 10-5), complete the following under **Connect using**:
 - a. For **Local adapter**, select the **QLogic <name or model> Adapter**.
 - b. For **Initiator IP**, select the adapter IP address.
 - c. Click **OK**.

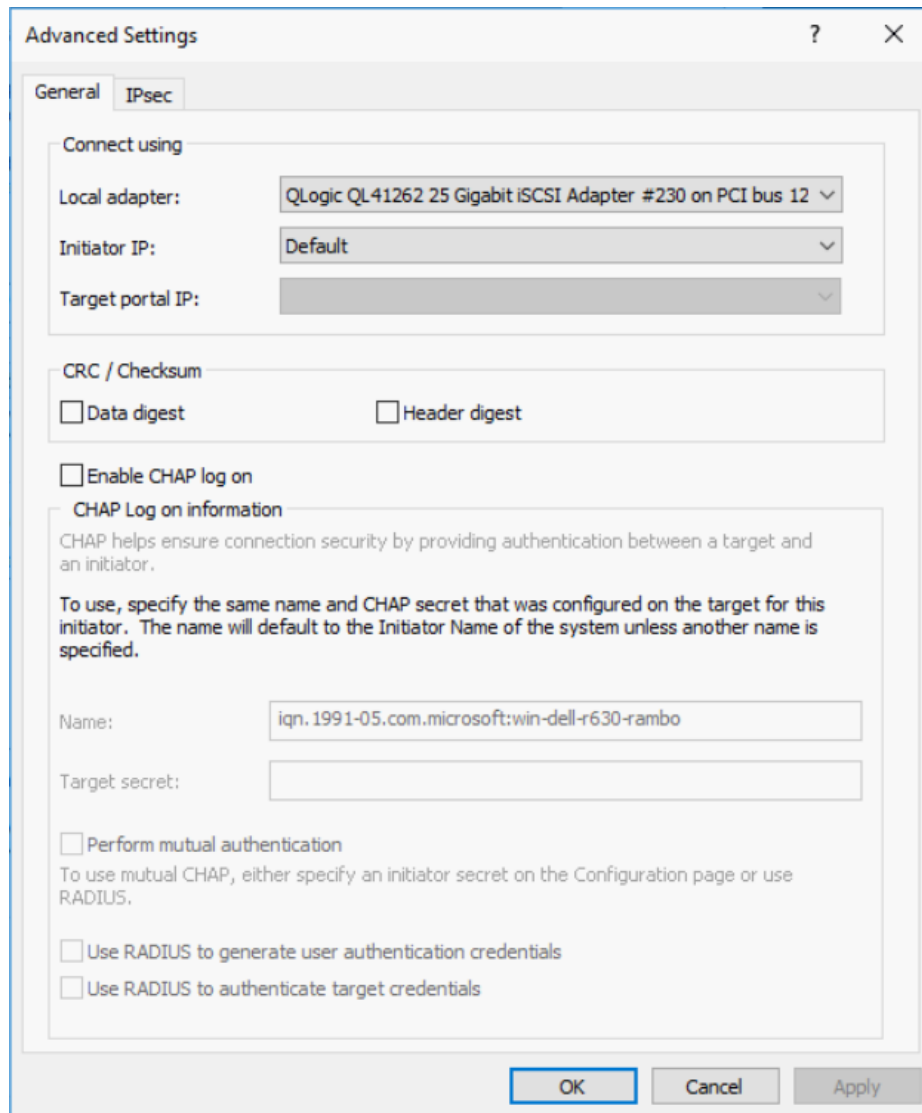


Figure 10-5. Selecting the Initiator IP Address

7. On the iSCSI Initiator Properties, Discovery page, click **OK**.

- Click the **Targets** tab, and then on the Targets page (Figure 10-6), click **Connect**.

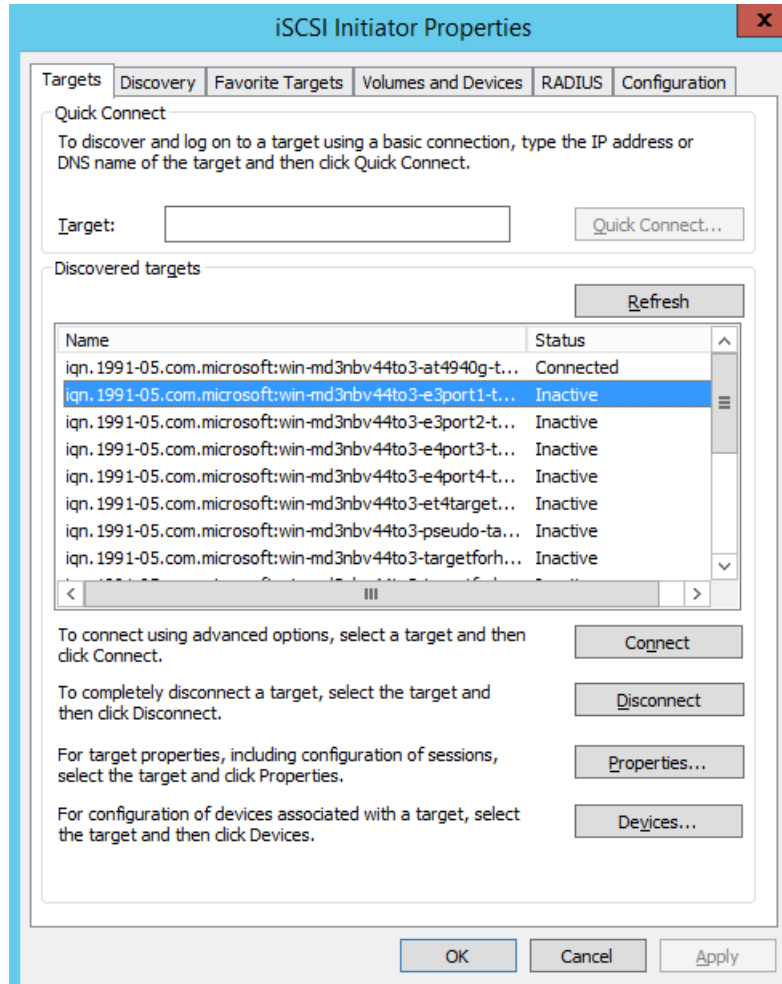


Figure 10-6. Connecting to the iSCSI Target

9. On the Connect To Target dialog box (Figure 10-7), click **Advanced**.

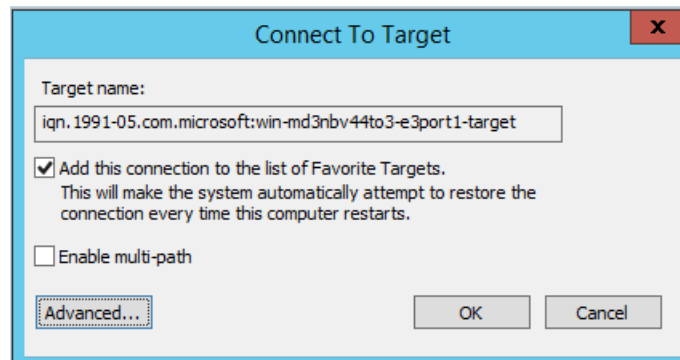


Figure 10-7. Connect To Target Dialog Box

10. In the Local Adapter dialog box, select the **QLogic <name or model> Adapter**, and then click **OK**.
11. Click **OK** again to close Microsoft Initiator.
12. To format the iSCSI partition, use Disk Manager.

NOTE

Some limitations of the teaming functionality include:

- Teaming does not support iSCSI adapters.
- Teaming does not support NDIS adapters that are in the boot path.
- Teaming supports NDIS adapters that are not in the iSCSI boot path, but only for the switch-independent NIC team type.
- Switch dependent teaming (IEEE 802.3ad LACP and Generic/Static Link Aggregation (Trunking)) cannot use a switch independent partitioned virtual adapter. IEEE standards require Switch Dependent Teaming (IEEE 802.3ad LACP and Generic/Static Link Aggregation (Trunking)) mode to work per the entire port instead of just the MAC address (fraction of a port) granularity.
- Microsoft recommends using their in-OS NIC teaming service instead of any adapter vendor-proprietary NIC teaming driver on Windows Server 2012 and later.

iSCSI Offload FAQs

Some of the frequently asked questions about iSCSI offload include:

Question: How do I assign an IP address for iSCSI offload?

Answer: Use the Configurations page in QConvergeConsole GUI.

Question: What tools should I use to create the connection to the target?

Answer: Use Microsoft iSCSI Software Initiator (version 2.08 or later).

Question: How do I know that the connection is offloaded?

Answer: Use Microsoft iSCSI Software Initiator. From a command line, type `oiscsicli sessionlist`. From **Initiator Name**, an iSCSI offloaded connection will display an entry beginning with `B06BDRV`. A non-offloaded connection displays an entry beginning with `Root`.

Question: What configurations should be avoided?

Answer: The IP address should not be the same as the LAN.

Windows Server 2016 and 2019/Azure Stack HCI iSCSI Boot Installation

Windows Server 2016 and Windows Server 2019/Azure Stack HCI support booting and installing in either the offload or non-offload paths. Marvell requires that you use a slipstream DVD with the latest Marvell drivers injected. See [“Injecting \(Slipstreaming\) Adapter Drivers into Windows Image Files” on page 125](#).

The following procedure prepares the image for installation and booting in either the offload or non-offload path.

To set up Windows Server 2016/2019/Azure Stack HCI iSCSI boot:

1. Remove any local hard drives on the system to be booted (remote system).
2. Prepare the Windows OS installation media by following the slipstreaming steps in [“Injecting \(Slipstreaming\) Adapter Drivers into Windows Image Files” on page 125](#).
3. Load the latest Marvell iSCSI boot images into the NVRAM of the adapter.
4. Configure the iSCSI target to allow a connection from the remote device. Ensure that the target has sufficient disk space to hold the new OS installation.
5. Configure the UEFI HII to set the iSCSI boot type (offload or non-offload), correct initiator, and target parameters for iSCSI boot.
6. Save the settings and reboot the system. The remote system should connect to the iSCSI target and then boot from the DVD-ROM device.
7. Boot from DVD and begin installation.
8. Follow the on-screen instructions.

At the window that shows the list of disks available for the installation, the iSCSI target disk should be visible. This target is a disk connected through the iSCSI boot protocol and located in the remote iSCSI target.

9. To proceed with Windows Server 2016/2019/Azure Stack HCI installation, click **Next**, and then follow the on-screen instructions. The server will undergo a reboot multiple times as part of the installation process.
10. After the server boots to the OS, you should run the driver installer to complete the Marvell drivers and application installation.

iSCSI Crash Dump

Crash dump functionality is supported for both non-offload and offload iSCSI boot for the 41000 Series Adapters. No additional configurations are required to configure iSCSI crash dump generation.

iSCSI Offload in Linux Environments

The Marvell FastLinQ 41000 iSCSI software consists of a single kernel module called `qedi.ko` (`qedi`). The `qedi` module is dependent on additional parts of the Linux kernel for specific functionality:

- `qed.ko` is the Linux eCore kernel module used for common Marvell FastLinQ 41000 hardware initialization routines.
- `scsi_transport_iscsi.ko` is the Linux iSCSI transport library used for upcall and downcall for session management.
- `libiscsi.ko` is the Linux iSCSI library function needed for protocol data unit (PDU) and task processing, as well as session memory management.
- `iscsi_boot_sysfs.ko` is the Linux iSCSI sysfs interface that provides helpers to export iSCSI boot information.
- `uio.ko` is the Linux Userspace I/O interface, used for light L2 memory mapping for `iscsiuio`.

These modules must be loaded before `qedi` can be functional. Otherwise, you might encounter an “unresolved symbol” error. If the `qedi` module is installed in the distribution update path, the requisite is automatically loaded by `modprobe`.

This section provides the following information about iSCSI offload in Linux:

- [Differences from `bnx2i`](#)
- [Configuring `qedi.ko`](#)
- [Verifying iSCSI Interfaces in Linux](#)

Differences from bnx2i

Some key differences exist between `qedi`—the driver for the Marvell FastLinQ 41000 Series Adapter (iSCSI)—and the previous Marvell iSCSI offload driver—`bnx2i` for the Marvell 8400 Series Adapters. Some of these differences include:

- `qedi` directly binds to a PCI function exposed by the CNA.
- `qedi` does not sit on top of the `net_device`.
- `qedi` is not dependent on a network driver such as `bnx2x` and `cnic`.
- `qedi` is not dependent on `cnic`, but it has dependency on `qed`.
- `qedi` is responsible for exporting boot information in `sysfs` using `iscsi_boot_sysfs.ko`, whereas `bnx2i` boot from SAN relies on the `iscsi_ibft.ko` module for exporting boot information.

Configuring `qedi.ko`

The `qedi` driver automatically binds to the exposed iSCSI functions of the CNA, and the target discovery and binding is done through the Open-iSCSI tools. This functionality and operation is similar to that of the `bnx2i` driver.

To load the `qedi.ko` kernel module, issue the following commands:

```
# modprobe qed
# modprobe libiscsi
# modprobe uio
# modprobe iscsi_boot_sysfs
# modprobe qedi
```

Verifying iSCSI Interfaces in Linux

After installing and loading the `qedi` kernel module, you must verify that the iSCSI interfaces were detected correctly.

To verify iSCSI interfaces in Linux:

1. To verify that the `qedi` and associated kernel modules are actively loaded, issue the following command:

```
# lsmod | grep qedi
qedi                114578  2
qed                 697989  1 qedi
uio                 19259   4 cnic,qedi
libiscsi            57233   2 qedi,bnx2i
scsi_transport_iscsi 99909   5 qedi,bnx2i,libiscsi
iscsi_boot_sysfs    16000   1 qedi
```

2. To verify that the iSCSI interfaces were detected properly, issue the following command. In this example, two iSCSI CNA devices are detected with SCSI host numbers 4 and 5.

```
# dmesg | grep qedi
[0000:00:00.0]:[qedi_init:3696]: QLogic iSCSI Offload Driver v8.15.6.0.
....
[0000:42:00.4]:[__qedi_probe:3563]:59: QLogic FastLinQ iSCSI Module qedi 8.15.6.0, FW 8.15.3.0
....
[0000:42:00.4]:[qedi_link_update:928]:59: Link Up event.
....
[0000:42:00.5]:[__qedi_probe:3563]:60: QLogic FastLinQ iSCSI Module qedi 8.15.6.0, FW 8.15.3.0
....
[0000:42:00.5]:[qedi_link_update:928]:59: Link Up event
```

3. Use Open-iSCSI tools to verify that the IP is configured properly. Issue the following command:

```
# iscsiadm -m iface | grep qedi
qedi.00:0e:1e:c4:e1:6d
qedi,00:0e:1e:c4:e1:6d,192.168.101.227,<empty>,iqn.1994-05.com.redhat:534ca9b6adf
qedi.00:0e:1e:c4:e1:6c
qedi,00:0e:1e:c4:e1:6c,192.168.25.91,<empty>,iqn.1994-05.com.redhat:534ca9b6adf
```

4. To ensure that the `iscsiuio` service is running, issue the following command:

```
# systemctl status iscsi.service
iscsiuio.service - iSCSI UserSpace I/O driver
Loaded: loaded (/usr/lib/systemd/system/iscsiuio.service; disabled; vendor preset: disabled)
Active: active (running) since Fri 2017-01-27 16:33:58 IST; 6 days ago
Docs: man:iscsiuio(8)
Process: 3745 ExecStart=/usr/sbin/iscsiuio (code=exited, status=0/SUCCESS)
Main PID: 3747 (iscsiuio)
CGroup: /system.slice/iscsiuio.service !--3747 /usr/sbin/iscsiuio
Jan 27 16:33:58 localhost.localdomain systemd[1]: Starting iSCSI
UserSpace I/O driver...
Jan 27 16:33:58 localhost.localdomain systemd[1]: Started iSCSI UserSpace I/O driver.
```

5. To discover the iSCSI target, issue the `iscsiadm` command:

```
#iscsiadm -m discovery -t st -p 192.168.25.100 -I qedi.00:0e:1e:c4:e1:6c
192.168.25.100:3260,1 iqn.2003-
04.com.sanblaze:virtualun.virtualun.target-05000007
192.168.25.100:3260,1 iqn.2003-04.com.sanblaze:virtualun.virtualun.target-05000012
192.168.25.100:3260,1 iqn.2003-04.com.sanblaze:virtualun.virtualun.target-0500000c
```

```
192.168.25.100:3260,1 iqn.2003-04.com.sanblaze:virtualun.virtualun.target-05000001
192.168.25.100:3260,1 iqn.2003-04.com.sanblaze:virtualun.virtualun.target-05000002
```

6. Log into the iSCSI target using the IQN obtained in [Step 5](#). To initiate the login procedure, issue the following command (where the last character in the command is a lowercase letter "L"):

```
#iscsiadm -m node -p 192.168.25.100 -T
iqn.2003-04.com.sanblaze:virtualun.virtualun.target-0000007 -l
Logging in to [iface: qedi.00:0e:1e:c4:e1:6c,
target:iqn.2003-04.com.sanblaze:virtualun.virtualun.target-05000007, portal:192.168.25.100,3260]
(multiple)
Login to [iface: qedi.00:0e:1e:c4:e1:6c, target:iqn.2003-04.com.sanblaze:virtualun.virtualun.target-05000007, portal:192.168.25.100,3260] successful.
```

7. To verify that the iSCSI session was created, issue the following command:

```
# iscsiadm -m session
qedi: [297] 192.168.25.100:3260,1
iqn.2003-04.com.sanblaze:virtualun.virtualun.target-05000007 (non-flash)
```

8. To check for iSCSI devices, issue the `iscsiadm` command:

```
# iscsiadm -m session -P3
...
*****
Attached SCSI devices:
*****
Host Number: 59 State: running
scsi59 Channel 00 Id 0 Lun: 0
Attached scsi disk sdb State: running scsi59 Channel 00 Id 0 Lun: 1
Attached scsi disk sdc State: running scsi59 Channel 00 Id 0 Lun: 2
Attached scsi disk sdd State: running scsi59 Channel 00 Id 0 Lun: 3
Attached scsi disk sde State: running scsi59 Channel 00 Id 0 Lun: 4
Attached scsi disk sdf State: running
```

For advanced target configurations, refer to the Open-iSCSI README at:

<https://github.com/open-iscsi/open-iscsi/blob/master/README>

iSCSI Offload in VMware ESXi

Follow the procedure in this section to allow a VMware operating system to boot from an iSCSI target machine located remotely over a standard IP network.

Follow the procedure in this section to allow a VMware system to access an iSCSI target machine located remotely over a standard IP network.

To set up iSCSI Offload in VMware ESXi:

1. Using the BIOS interface, set iSCSI Offload Mode to **Enabled** (Figure 10-8).

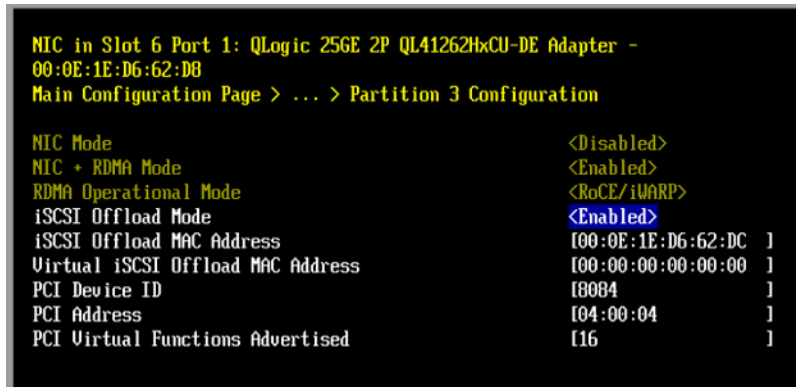


Figure 10-8. Enabling iSCSI Offload Mode in the BIOS

2. Create a new VMkernel adapter (Figure 10-9).

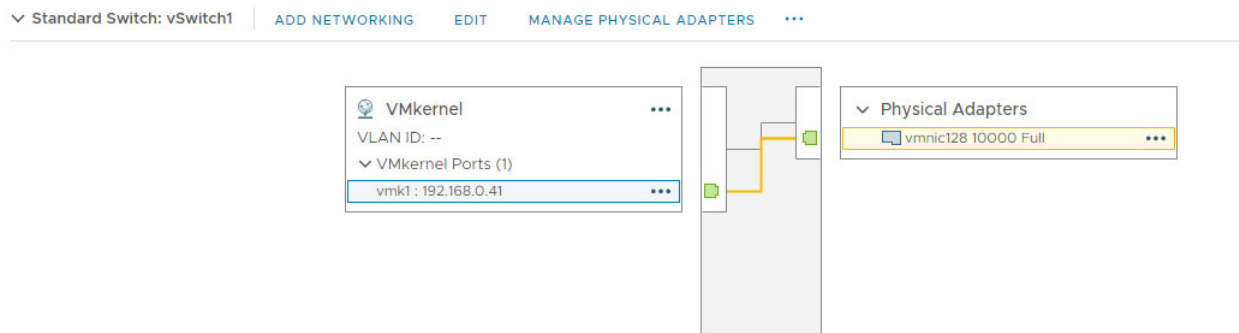


Figure 10-9. Creating a VMkernel Adapter

3. Bind the VMkernel adapter created in [Step 2](#) to the iSCSI partition ([Figure 10-10](#)).

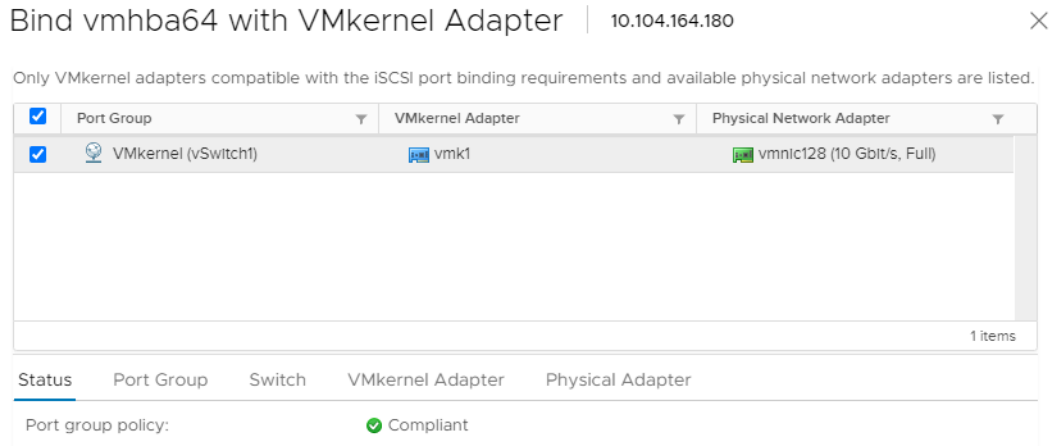


Figure 10-10. Binding the VMkernel Adapter to the iSCSI Partition

4. Add the send target information ([Figure 10-11](#)).

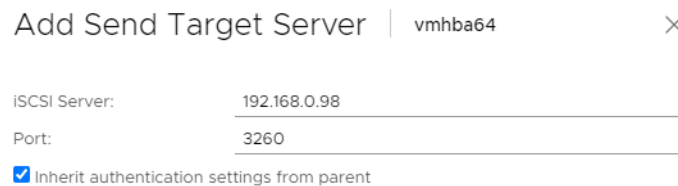


Figure 10-11. Adding Send Target Information

5. Discover the LUNs (Figure 10-12).

Storage Adapters

+ Add Software Adapter Refresh Rescan Storage... Rescan Adapter Remove

Adapter	Type	Status	Identifier	Targets	Devises	Paths
Model: QLogic FastLinQ QL41xxx Series 10/25 GbE Controller (iSCSI)						
vmhba64	iSCSI	Online	qedtl-000efed6899b[qn.1998-01.com.vmware.gad17954.s...	11	11	11
vmhba65	iSCSI	Unbound	qedtl-000efed6899c[qn.1998-01.com.vmware.gad17994.s...	0	0	0
Model: Smart Array P440ar						
vmhba2	SAS	Unknown	--	1	3	3
Model: Wellsburg RAID Controller						
vmhba0	Block SCSI	Unknown	--	1	1	1

Properties Devises Paths Dynamic Discovery Static Discovery Network Port Binding Advanced Options

Refresh Attach Detach Rename...

Name	LUN	Type	Capacity	Datastore	Operational State	Hardware Acceleration	Drive Type	Transport
EQLOGIC iSCSI Disk (naa.6019cbctd119837216b3850aac429ee)	0	disk	40.00 GB	Not Cons...	Attached	Supported	HDD	iSCSI
EQLOGIC iSCSI Disk (naa.6019cbctd11943d531c1a58312e60955)	0	disk	3.00 GB	Not Cons...	Attached	Supported	HDD	iSCSI
EQLOGIC iSCSI Disk (naa.6019cbctd119f36c16b3550aac429c6)	0	disk	40.00 GB	Not Cons...	Attached	Supported	HDD	iSCSI
EQLOGIC iSCSI Disk (naa.6019cbctd119f3d131c1458312e65920)	0	disk	3.00 GB	Not Cons...	Attached	Supported	HDD	iSCSI
EQLOGIC iSCSI Disk (naa.6019cbctd119a3d331c1758312e6e98e)	0	disk	3.00 GB	Not Cons...	Attached	Supported	HDD	iSCSI

Figure 10-12. LUN Discovery

6. Create a new datastore using the New Datastore Wizard.
- With **1 Type** highlighted, in the list of Types, click **VMFS** (Figure 10-13).

New Datastore

1 Type

2 Name and device selection

3 VMFS version

4 Partition configuration

5 Ready to complete

Type

Specify datastore type.

VMFS
Create a VMFS datastore on a disk/LUN.

NFS
Create an NFS datastore on an NFS share over the network.

vVol
Create a Virtual Volumes datastore on a storage container connected to a storage provider.

Figure 10-13. Datastore Type

- b. With **2 Name and device selection** highlighted, select the appropriate LUN for provisioning the datastore (Figure 10-14).

New Datastore

✓ 1 Type
2 Name and device selection
3 VMFS version
4 Partition configuration
5 Ready to complete

Name and device selection
Select a name and a disk/LUN for provisioning the datastore.

Datastore name:

Name	LUN	Capacity	Hardware...	Drive T...	S
EQLOGIC ISCSI Disk (naa....	0	3.00 GB	Supported	HDD	E
EQLOGIC ISCSI Disk (naa....	0	40.00 GB	Supported	HDD	E
EQLOGIC ISCSI Disk (naa....	0	40.00 GB	Supported	HDD	E
EQLOGIC ISCSI Disk (naa....	0	3.00 GB	Supported	HDD	E
EQLOGIC ISCSI Disk (naa....	0	3.00 GB	Supported	HDD	E

Figure 10-14. Selecting a LUN for the Datastore

- c. With **3 VMFS version** highlighted, select the correct VMFS version for the datastore (Figure 10-15).

New Datastore

✓ 1 Type
✓ 2 Name and device selection
3 VMFS version
4 Partition configuration
5 Ready to complete

VMFS version
Specify the VMFS version for the datastore.

VMFS 6
VMFS 6 enables advanced format (512e) and automatic space reclamation support.

VMFS 5
VMFS 5 enables 2+TB LUN support.

Figure 10-15. VMFS Version

- d. With **4 Partition configuration** highlighted, specify the disk layout and partition configuration information (Figure 10-16).

New Datastore

- ✓ 1 Type
- ✓ 2 Name and device selection
- ✓ 3 VMFS version
- 4 Partition configuration**
- 5 Ready to complete

Partition configuration
Review the disk layout and specify partition configuration details.

Partition Configuration	Use all available partitions
Datastore Size	40 GB
Block size	1 MB
Space Reclamation Granularity	1 MB
Space Reclamation Priority	Low: Deleted or unmapped blocks are reclaimed on the LUN at Low priority

Empty: 40.0 GB

Figure 10-16. Partition Configuration Information

- e. With **5 Ready to complete** highlighted, review all of your changes (Figure 10-17).

New Datastore

- ✓ 1 Type
- ✓ 2 Name and device selection
- ✓ 3 VMFS version
- ✓ 4 Partition configuration
- 5 Ready to complete**

Ready to complete
Review your settings selections before finishing the wizard.

General	
Name:	DatastoreiSCSI
Type:	VMFS
Datastore size:	40.00 GB
Device and Formatting	
Disk/LUN:	EQLOGIC iSCSI Disk (naa.6019cbc1d119f36c16b3550aacf429c6)
Partition Format:	GPT
VMFS Version:	VMFS 6
Block Size:	1 MB
Space Reclamation Granularity:	1 MB
Space Reclamation Priority:	Low: Deleted or unmapped blocks are reclaimed on the LUN at low priority

Figure 10-17. Datastore Properties

NOTE

Some limitations of iSCSI Offload in VMware ESXi include:

- There are limitations with vSwitch; distributed vSwitch (DVS) should not be used. For best practice, create a 1:1 mapping between the vmk and the vmnic; that is, create a separate vSwitch per VMkernel adapter and add only the corresponding physical NIC to it.

The Marvell offload solution provides redundancy at the SCSI/vmhba level. Therefore, Marvell does not recommend adding multiple NICs (Teaming) to same standard switch, because it does not add redundancy at the NIC level.

- Due to limitations in the user daemon stack, the current iSCSI connection-over-router solution does not support iSCSI connections to targets that can be accessed over routers.
- For L2 services needed by iSCSI, such as ARP and DHCP, the iSCSI vmhba binds to a vmnic device created by the qedi driver. This binding is only a thin, dummy vmnic implementation, and should not be used for any NIC traffic and/or other functionalities.

This vmnic device is different from the physical vmnic device registered by the adapter.

- When configuring port binding, follow the rules from VMware as indicated in the following article:

<https://docs.vmware.com/en/VMware-vSphere/7.0/com.vmware.vsphere.storage.doc/GUID-0D31125F-DC9D-475B-BC3D-A3E131251642.html#GUID-0D31125F-DC9D-475B-BC3D-A3E131251642>

11 FCoE Configuration

This chapter provides the following Fibre Channel over Ethernet (FCoE) configuration information:

- “Configuring Linux FCoE Offload” on page 228

NOTE

FCoE offload is supported on all 41000 Series Adapters. Some FCoE features may not be fully enabled in the current release. For details, refer to [Appendix E Feature Constraints](#).

To enable iSCSI-Offload mode, see the *Application Note, Enabling Storage Offloads on Dell and Marvell FastLinQ 41000 Series Adapters* at <https://www.marvell.com/documents/5aa5otcbkr0im3ynera3/>.

For FCoE boot from SAN information, see [Chapter 6 Boot from SAN Configuration](#).

Configuring Linux FCoE Offload

The Marvell FastLinQ 41000 Series Adapter FCoE software consists of a single kernel module called `qedf.ko` (`qedf`). The `qedf` module is dependent on additional parts of the Linux kernel for specific functionality:

- `qed.ko` is the Linux eCore kernel module used for common Marvell FastLinQ 41000 hardware initialization routines.
- `libfcoc.ko` is the Linux FCoE kernel library needed to conduct FCoE forwarder (FCF) solicitation and FCoE initialization protocol (FIP) fabric login (FLOGI).
- `libfc.ko` is the Linux FC kernel library needed for several functions, including:
 - Name server login and registration
 - rport session management
- `scsi_transport_fc.ko` is the Linux FC SCSI transport library used for remote port and SCSI target management.

These modules must be loaded before `qedf` can be functional, otherwise errors such as “unresolved symbol” can result. If the `qedf` module is installed in the distribution update path, the requisite modules are automatically loaded by `modprobe`. Marvell FastLinQ 41000 Series Adapters support FCoE offload.

This section provides the following information about FCoE offload in Linux:

- [Differences Between `qedf` and `bnx2fc`](#)
- [Configuring `qedf.ko`](#)
- [Verifying FCoE Devices in Linux](#)

Differences Between `qedf` and `bnx2fc`

Significant differences exist between `qedf`—the driver for the Marvell FastLinQ 41000 10/25GbE Controller (FCoE)—and the previous Marvell FCoE offload driver, `bnx2fc`. Differences include:

- `qedf` directly binds to a PCI function exposed by the CNA.
- `qedf` does not need the open-fcoe user space tools (`fipvlan`, `fcoemon`, `fcoeadm`) to initiate discovery.
- `qedf` issues FIP vLAN requests directly and does not need the `fipvlan` utility.
- `qedf` does not need an FCoE interface created by `fipvlan` for `fcoemon`.
- `qedf` does not sit on top of the `net_device`.
- `qedf` is not dependent on network drivers (such as `bnx2x` and `cnic`).
- `qedf` will automatically initiate FCoE discovery on link up (because it is not dependent on `fipvlan` or `fcoemon` for FCoE interface creation).

NOTE

FCoE interfaces no longer sit on top of the network interface. The `qedf` driver automatically creates FCoE interfaces that are separate from the network interface. Thus, FCoE interfaces do not show up in the FCoE interface dialog box in the installer. Instead, the disks show up automatically as SCSI disks, similar to the way Fibre Channel drivers work.

Configuring `qedf.ko`

No explicit configuration is required for `qedf.ko`. The driver automatically binds to the exposed FCoE functions of the CNA and begins discovery. This functionality is similar to the functionality and operation of the Marvell FC driver, `qla2xx`, as opposed to the older `bnx2fc` driver.

NOTE

For more information on FastLinQ driver installation, see [Chapter 3 Driver Installation](#).

The load `qedf.ko` kernel module performs the following:

```
# modprobe qed
# modprobe libfcoe
# modprobe qedf
```

Verifying FCoE Devices in Linux

Follow these steps to verify that the FCoE devices were detected correctly after installing and loading the `qedf` kernel module.

To verify FCoE devices in Linux:

1. Check `lsmod` to verify that the `qedf` and associated kernel modules were loaded:

```
# lsmod | grep qedf
69632 1 qedf libfc
143360 2 qedf,libfcoe scsi_transport_fc
65536 2 qedf,libfc qed
806912 1 qedf scsi_mod
262144 14 sg,hpsa,qedf,scsi_dh_alua,scsi_dh_rdac,dm_multipath,
scsi_transport_fc,scsi_transport_sas,libfc,scsi_transport_iscsi,scsi_dh_emc,
libata,sd_mod,sr_mod
```

2. Check `dmesg` to verify that the FCoE devices were detected properly. In this example, the two detected FCoE CNA devices are SCSI host numbers 4 and 5.

```
# dmesg | grep qedf
[ 235.321185] [0000:00:00.0]: [qedf_init:3728]: QLogic FCoE Offload Driver
v8.18.8.0.
....
[ 235.322253] [0000:21:00.2]: [__qedf_probe:3142]:4: QLogic FastLinQ FCoE
Module qedf 8.18.8.0, FW 8.18.10.0
[ 235.606443] scsi host4: qedf
....
[ 235.624337] [0000:21:00.3]: [__qedf_probe:3142]:5: QLogic FastLinQ FCoE
Module qedf 8.18.8.0, FW 8.18.10.0
[ 235.886681] scsi host5: qedf
....
```

```
[ 243.991851] [0000:21:00.3]: [qedf_link_update:489]:5: LINK UP (40 GB/s).
```

3. Check for discovered FCoE devices using the `lsscsi` or `lsblk -S` commands. An example of each command follows.

```
# lsscsi
[0:2:0:0]   disk    DELL    PERC H700          2.10  /dev/sda
[2:0:0:0]   cd/dvd  TEAC   DVD-ROM DV-28SW   R.2A  /dev/sr0
[151:0:0:0] disk    HP     P2000G3 FC/iSCSI T252  /dev/sdb
[151:0:0:1] disk    HP     P2000G3 FC/iSCSI T252  /dev/sdc
[151:0:0:2] disk    HP     P2000G3 FC/iSCSI T252  /dev/sdd
[151:0:0:3] disk    HP     P2000G3 FC/iSCSI T252  /dev/sde
[151:0:0:4] disk    HP     P2000G3 FC/iSCSI T252  /dev/sdf
```

```
# lsblk -S
NAME HCTL          TYPE  VENDOR  MODEL          REV  TRAN
sdb  5:0:0:0        disk  SANBlaze VLUN P2T1L0      V7.3 fc
sdc  5:0:0:1        disk  SANBlaze VLUN P2T1L1      V7.3 fc
sdd  5:0:0:2        disk  SANBlaze VLUN P2T1L2      V7.3 fc
sde  5:0:0:3        disk  SANBlaze VLUN P2T1L3      V7.3 fc
sdf  5:0:0:4        disk  SANBlaze VLUN P2T1L4      V7.3 fc
sdg  5:0:0:5        disk  SANBlaze VLUN P2T1L5      V7.3 fc
sdh  5:0:0:6        disk  SANBlaze VLUN P2T1L6      V7.3 fc
sdi  5:0:0:7        disk  SANBlaze VLUN P2T1L7      V7.3 fc
sdj  5:0:0:8        disk  SANBlaze VLUN P2T1L8      V7.3 fc
sdk  5:0:0:9        disk  SANBlaze VLUN P2T1L9      V7.3 fc
```

Configuration information for the host is located in `/sys/class/fc_host/hostX`, where `x` is the number of the SCSI host. In the preceding example, `x` is 4. The `hostX` file contains attributes for the FCoE function, such as worldwide port name and fabric ID.

12 SR-IOV Configuration

Single root input/output virtualization (SR-IOV) is a specification by the PCI SIG that enables a single PCI Express (PCIe) device to appear as multiple, separate physical PCIe devices. SR-IOV permits isolation of PCIe resources for performance, interoperability, and manageability.

NOTE

Some SR-IOV features may not be fully enabled in the current release.

This chapter provides instructions for:

- [Configuring SR-IOV on Windows](#)
- [“Configuring SR-IOV on Linux” on page 239](#)
- [“Configuring SR-IOV on VMware” on page 245](#)

Configuring SR-IOV on Windows

To configure SR-IOV on Windows:

1. Access the server BIOS System Setup, and then click **System BIOS Settings**.
2. On the System BIOS Settings page, click **Integrated Devices**.
3. On the Integrated Devices page ([Figure 12-1](#)):
 - a. Set the **SR-IOV Global Enable** option to **Enabled**.
 - b. Click **Back**.

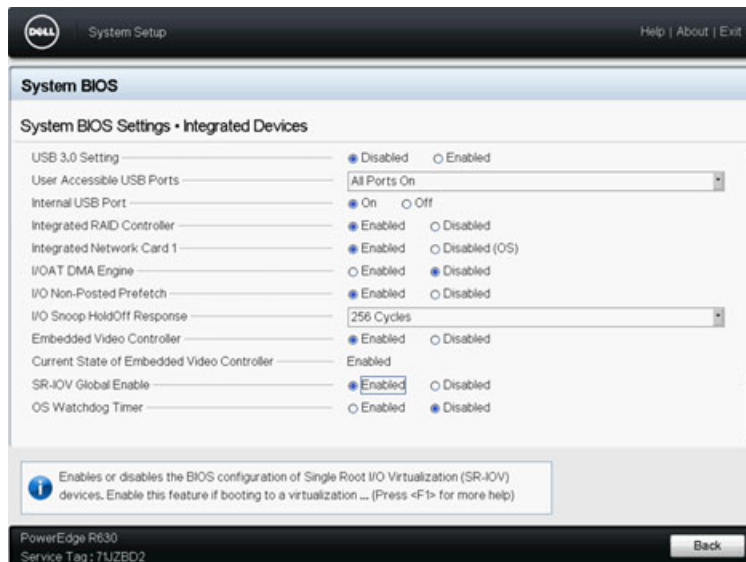


Figure 12-1. System Setup for SR-IOV: Integrated Devices

4. On the Main Configuration Page for the selected adapter, click **Device Level Configuration**.
5. On the Main Configuration Page - Device Level Configuration (Figure 12-2):
 - a. Set the **Virtualization Mode** to **SR-IOV**, or **NPar+SR-IOV** if you are using NPar mode.
 - b. Click **Back**.

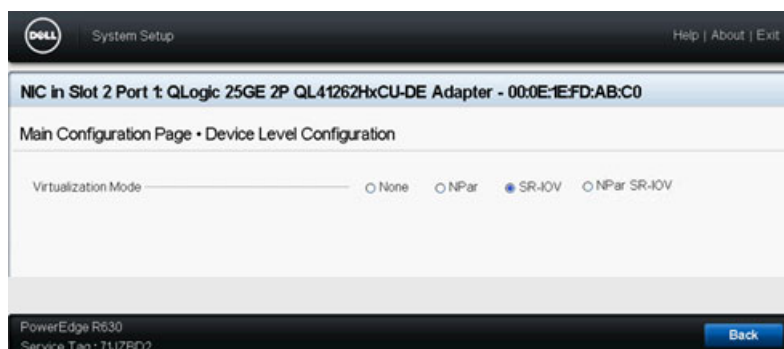


Figure 12-2. System Setup for SR-IOV: Device Level Configuration

6. On the Main Configuration Page, click **Finish**.
7. In the Warning - Saving Changes message box, click **Yes** to save the configuration.
8. In the Success - Saving Changes message box, click **OK**.

9. To enable SR-IOV on the miniport adapter:
 - a. Access Device Manager.
 - b. Open the miniport adapter properties, and then click the **Advanced** tab.
 - c. On the Advanced properties page (Figure 12-3) under **Property**, select **SR-IOV**, and then set the value to **Enabled**.
 - d. Click **OK**.

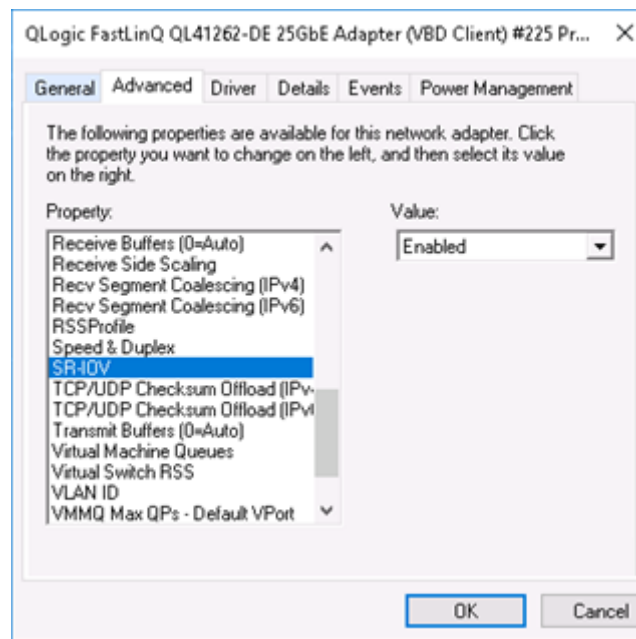


Figure 12-3. Adapter Properties, Advanced: Enabling SR-IOV

10. To create a Virtual Machine Switch (vSwitch) with SR-IOV (Figure 12-4 on page 235):
 - a. Launch the Hyper-V Manager.
 - b. Select **Virtual Switch Manager**.
 - c. In the **Name** box, type a name for the virtual switch.
 - d. Under **Connection type**, select **External network**.
 - e. Select the **Enable single-root I/O virtualization (SR-IOV)** check box, and then click **Apply**.

NOTE

Be sure to enable SR-IOV when you create the vSwitch. This option is unavailable after the vSwitch is created.

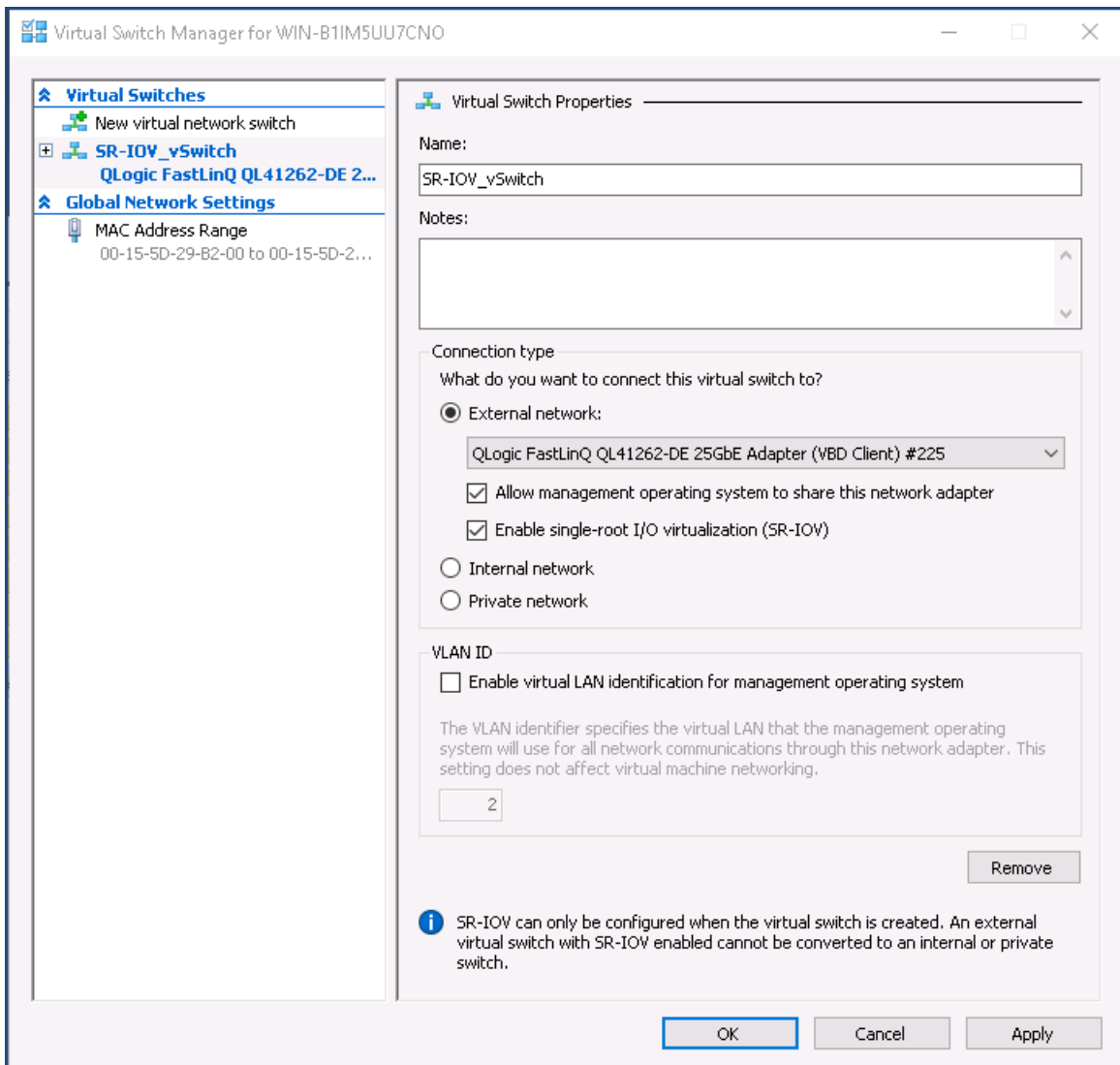


Figure 12-4. Virtual Switch Manager: Enabling SR-IOV

- f. The Apply Networking Changes message box advises you that **Pending changes may disrupt network connectivity**. To save your changes and continue, click **Yes**.

11. To get the virtual machine switch capability, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Get-VMSwitch -Name SR-IOV_vSwitch | fl
```

Output of the `Get-VMSwitch` command includes the following SR-IOV capabilities:

```
IovVirtualFunctionCount           : 80
IovVirtualFunctionsInUse          : 1
```

12. To create a virtual machine (VM) and export the virtual function (VF) in the VM:
 - a. Create a virtual machine.
 - b. Add the VMNetworkadapter to the virtual machine.
 - c. Assign a virtual switch to the VMNetworkadapter.
 - d. In the Settings for VM <VM_Name> dialog box (Figure 12-5), Hardware Acceleration page, under **Single-root I/O virtualization**, select the **Enable SR-IOV** check box, and then click **OK**.

NOTE

After the virtual adapter connection is created, the SR-IOV setting can be enabled or disabled at any time (even while traffic is running).

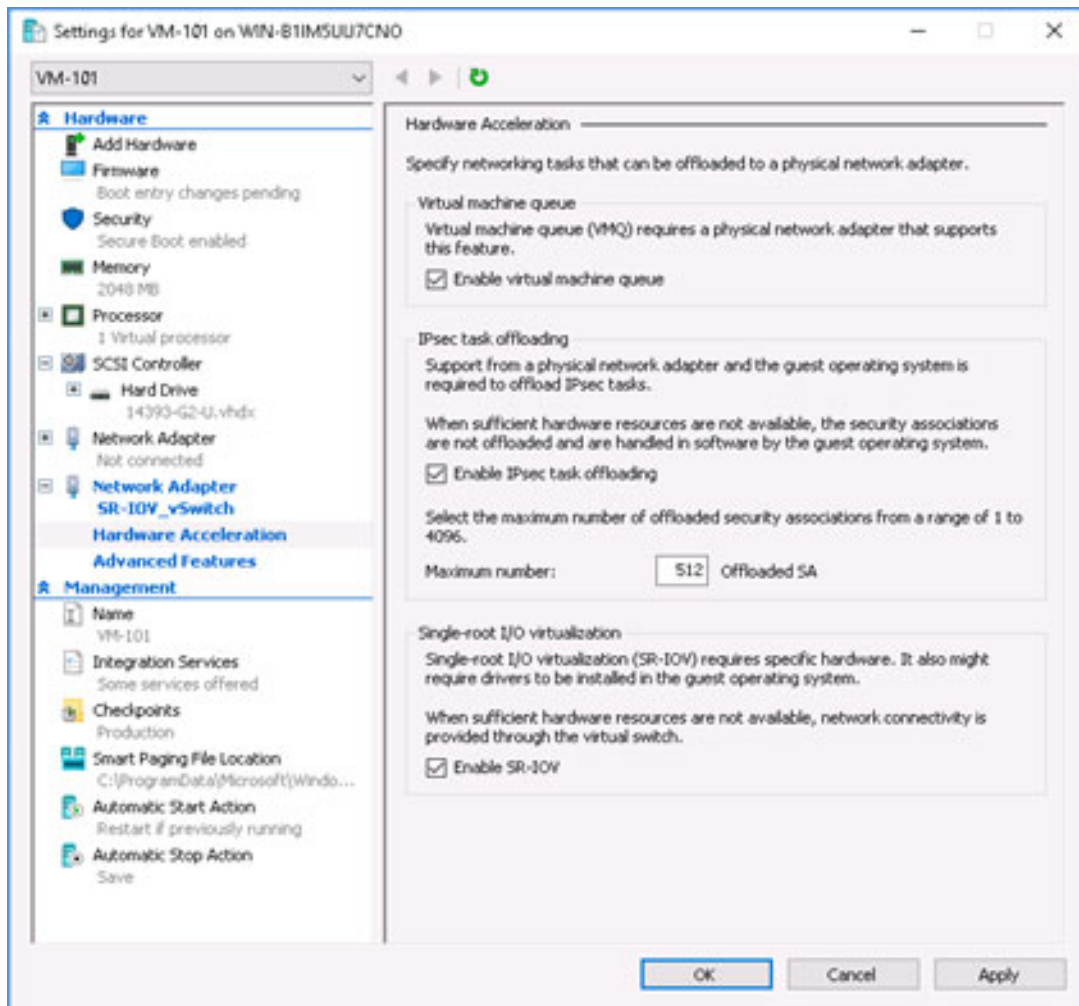


Figure 12-5. Settings for VM: Enabling SR-IOV

13. Install the Marvell drivers for the adapters detected in the VM. Use the latest drivers available from your vendor for your host OS (do not use inbox drivers).

NOTE

Be sure to use the same driver package on both the VM and the host system. For example, use the same qeVBD and qeND driver version on the Windows VM and in the Windows Hyper-V host.

After installing the drivers, the adapter is listed in the VM. [Figure 12-6](#) shows an example.

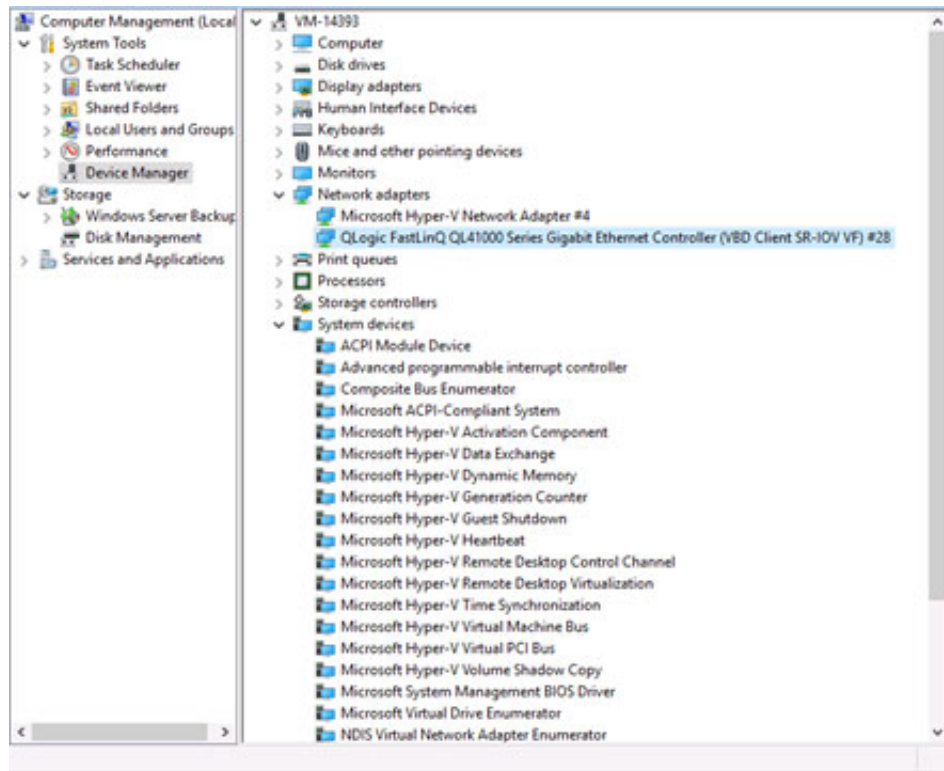


Figure 12-6. Device Manager: VM with QLogic Adapter

14. To view the SR-IOV VF details, issue the following Windows PowerShell command:

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

[Figure 12-7](#) shows example output.

```
PS C:\Users\Administrator>
PS C:\Users\Administrator> Get-NetAdapterSriovVf
Name                FunctionID VPortID MacAddress          VmID                VmFriendlyName
-----
Ethernet 10         0          {2}    00-15-5D-29-B2-01  51F01C52-CDC6-4932-A95E-86D... VM-101
PS C:\Users\Administrator>
```

Figure 12-7. Windows PowerShell Command: Get-NetadapterSriovVf

Configuring SR-IOV on Linux

To configure SR-IOV on Linux:

1. Access the server BIOS System Setup, and then click **System BIOS Settings**.
2. On the System BIOS Settings page, click **Integrated Devices**.
3. On the System Integrated Devices page (see [Figure 12-1 on page 233](#)):
 - a. Set the **SR-IOV Global Enable** option to **Enabled**.
 - b. Click **Back**.
4. On the System BIOS Settings page, click **Processor Settings**.
5. On the Processor Settings ([Figure 12-8](#)) page:
 - a. Set the **Virtualization Technology** option to **Enabled**.
 - b. Click **Back**.

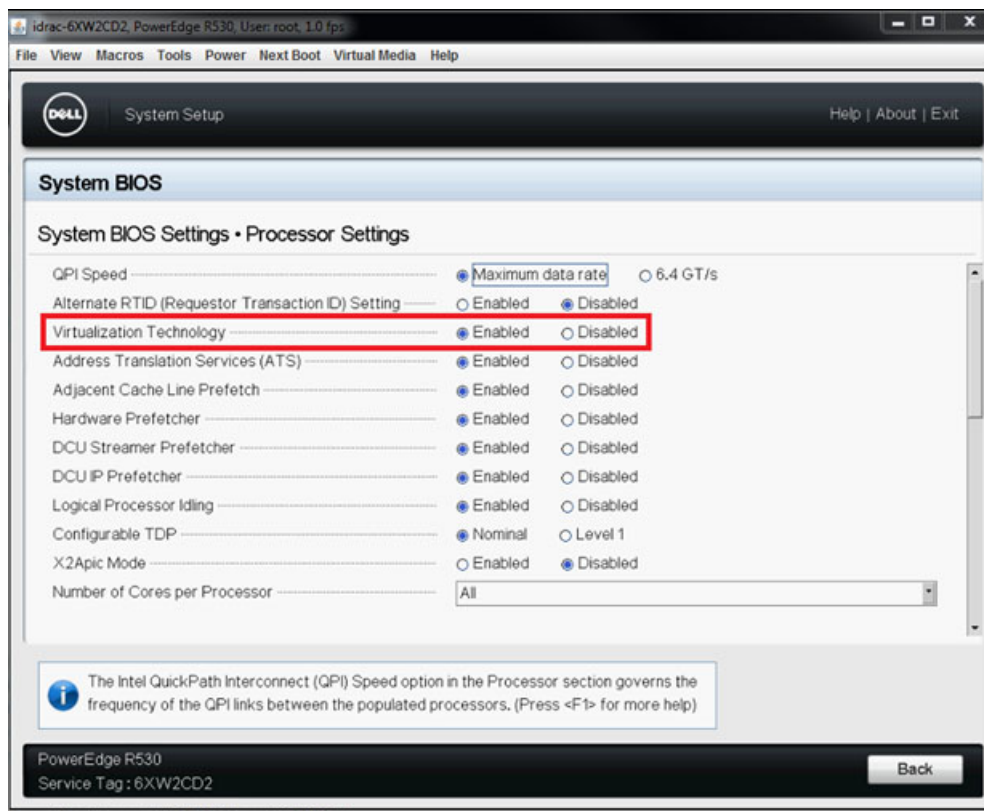


Figure 12-8. System Setup: Processor Settings for SR-IOV

6. On the System Setup page, select **Device Settings**.

7. On the Device Settings page, select **Port 1** for the Marvell adapter.
8. On the Device Level Configuration page ([Figure 12-9](#)):
 - a. Set the **Virtualization Mode** to **SR-IOV**.
 - b. Click **Back**.



Figure 12-9. System Setup for SR-IOV: Integrated Devices

9. On the Main Configuration Page, click **Finish**, save your settings, and then reboot the system.
10. To enable and verify virtualization:
 - a. Open the `grub.conf` file and configure the `iommu` parameter as shown in [Figure 12-10](#). (For details, see “[Enabling IOMMU for SR-IOV in UEFI-based Linux OS Installations](#)” on page 244.)
 - For Intel-based systems, add `intel_iommu=on`.
 - For AMD-based systems, add `amd_iommu=on`.

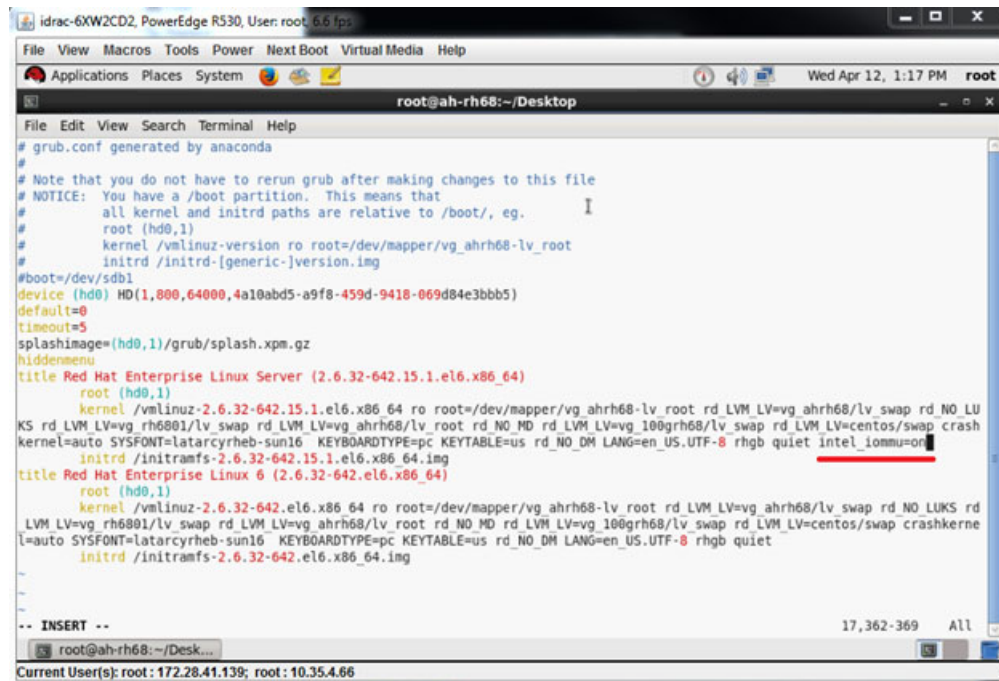


Figure 12-10. Editing the grub.conf File for SR-IOV

- b. Save the `grub.conf` file and then reboot the system.
- c. To verify that the changes are in effect, issue the following command:

```
dmesg | grep -i iommu
```

A successful input–output memory management unit (IOMMU) command output should show, for example:

```
Intel-IOMMU: enabled
```

- d. To view VF details (number of VFs and total VFs), issue the following command:

```
find /sys/|grep -i sriov
```

11. For a specific port, enable a quantity of VFs.

- a. Issue the following command to enable, for example, 8 VFs on PCI instance 04:00.0 (bus 4, device 0, function 0):

```
[root@ah-rh68 ~]# echo 8 >
/sys/devices/pci0000:00/0000:00:02.0/0000:04:00.0/
sriov_numvfs
```


- b. Review the command output (Figure 12-11) to confirm that actual VFs were created on bus 4, device 2 (from the 0000:00:02.0 parameter), functions 0 through 7. Note that the actual device ID is different on the PFs (8070 in this example) versus the VFs (8090 in this example).

```
[root@ah-rh68 Desktop]#  
[root@ah-rh68 Desktop]# echo 8 > /sys/devices/pci0000:00/0000:00:02.0/sriov_numvfs  
[root@ah-rh68 Desktop]#  
[root@ah-rh68 Desktop]# lspci -vv|grep -i QLogic  
04:00.0 Ethernet controller: QLogic Corp. Device 8070 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
Product Name: QLogic 25GE 2P QL41262HxCU-DE Adapter  
[V4] Vendor specific: NMVQLogic  
04:00.1 Ethernet controller: QLogic Corp. Device 8070 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
Product Name: QLogic 25GE 2P QL41262HxCU-DE Adapter  
[V4] Vendor specific: NMVQLogic  
04:02.0 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.1 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.2 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.3 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.4 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.5 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.6 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
04:02.7 Ethernet controller: QLogic Corp. Device 8090 (rev 02)  
Subsystem: QLogic Corp. Device 000b  
[root@ah-rh68 Desktop]#
```

Figure 12-11. Command Output for sriov_numvfs

12. To view a list of all PF and VF interfaces, issue the following command:

```
# ip link show | grep -i vf -b2
```

Figure 12-12 shows example output.

```
[root@localhost ~]# ip link show | grep -i vf -b2  
163-2: em1_1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT group default qlen 1000  
271- link/ether f4:e9:d4:ee:54:c2 brd ff:ff:ff:ff:ff:ff  
326: vf 0 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
439: vf 1 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
552: vf 2 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
665: vf 3 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
778: vf 4 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
891: vf 5 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
1004: vf 6 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto  
1117: vf 7 MAC 00:00:00:00:00:00, tx rate 10000 (Mbps), max_tx_rate 10000Mbps, spoof checking off, link-state auto
```

Figure 12-12. Command Output for ip link show Command

13. Assign and verify MAC addresses:

- a. To assign a MAC address to the VF, issue the following command:

```
ip link set <pf device> vf <vf index> mac <mac address>
```


- b. Ensure that the VF interface is up and running with the assigned MAC address.
14. Power off the VM and attach the VF. (Some OSs support hot-plugging of VFs to the VM.)
- a. In the Virtual Machine dialog box (Figure 12-13), click **Add Hardware**.

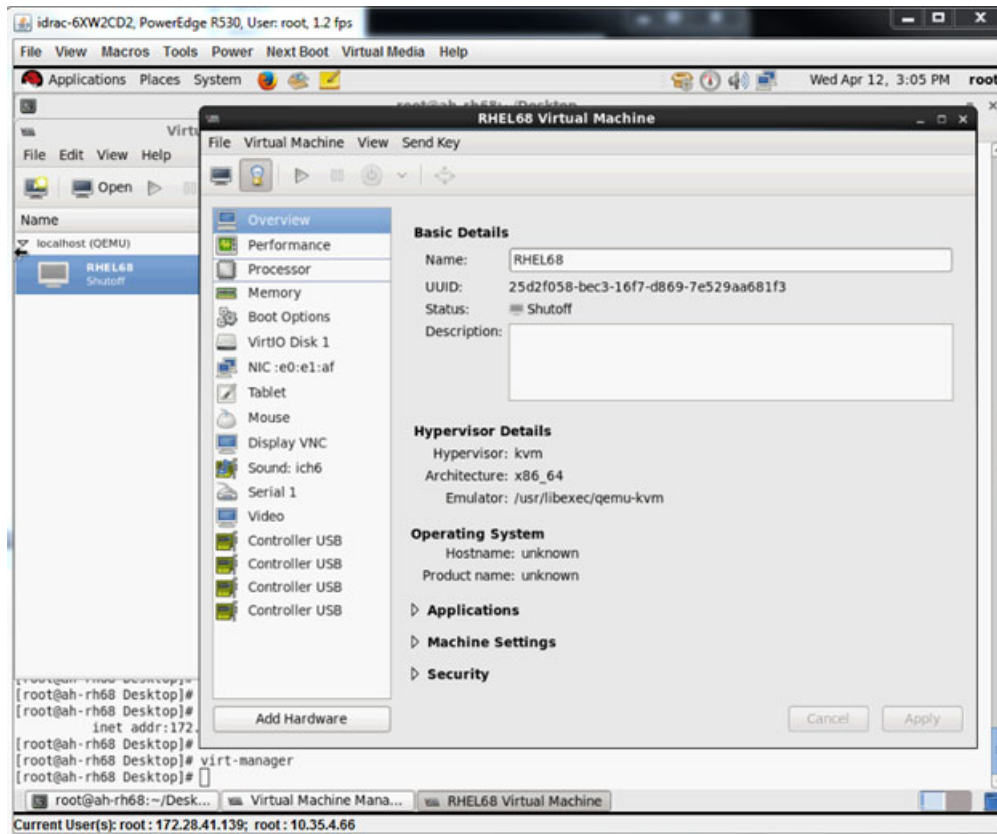


Figure 12-13. RHEL68 Virtual Machine

- b. In the left pane of the Add New Virtual Hardware dialog box (Figure 12-14), click **PCI Host Device**.
- c. In the right pane, select a host device.
- d. Click **Finish**.

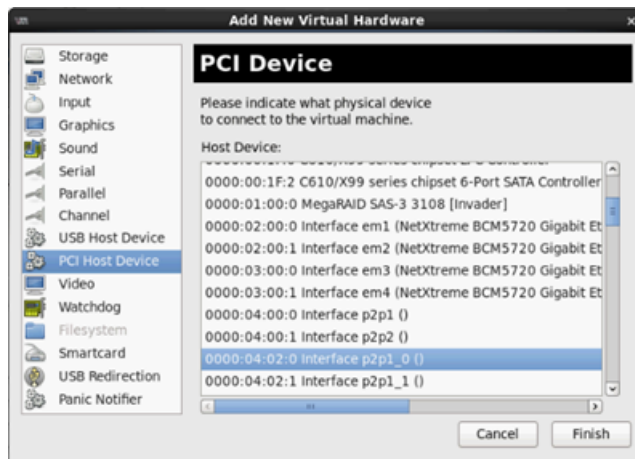


Figure 12-14. Add New Virtual Hardware

15. Power on the VM, and then issue the following command:

```
check lspci -vv|grep -I ether
```
16. Install the drivers for the adapters detected in the VM. Use the latest drivers available from your vendor for your host OS (do not use inbox drivers). The same driver version must be installed on the host and the VM.
17. As needed, add more VFs in the VM.

Enabling IOMMU for SR-IOV in UEFI-based Linux OS Installations

Follow the appropriate procedure for your Linux OS.

NOTE

For AMD systems, replace `intel_iommu=on` with `amd_iommu=on`.

To enable IOMMU for SR-IOV on RHEL 6.x:

- In the `/boot/efi/EFI/redhat/grub.conf` file, locate the kernel line, and then append the `intel_iommu=on` boot parameter.

To enable IOMMU for SR-IOV on RHEL 7.x and later:

1. In the `/etc/default/grub` file, locate `GRUB_CMDLINE_LINUX`, and then append the `intel_iommu=on` boot parameter.
2. To update the grub configuration file, issue the following command:

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

To enable IOMMU for SR-IOV on SLES 12.x:

1. In the `/etc/default/grub` file, locate `GRUB_CMDLINE_LINUX_DEFAULT`, and then append the `intel_iommu=on` boot parameter.
2. To update the grub configuration file, issue the following command:

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

To enable IOMMU for SR-IOV on SLES 15.x and later:

1. In the `/etc/default/grub` file, locate `GRUB_CMDLINE_LINUX_DEFAULT`, and then append the `intel_iommu=on` boot parameter.
2. To update the grub configuration file, issue the following command:

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

Configuring SR-IOV on VMware

NOTE

For instructions on enabling SR-IOV on a host physical adapter, see the VMware documentation.

To configure SR-IOV on VMware:

1. Access the server BIOS System Setup, and then click **System BIOS Settings**.
2. On the System BIOS Settings page, click **Integrated Devices**.
3. On the Integrated Devices page (see [Figure 12-1 on page 233](#)):
 - a. Set the **SR-IOV Global Enable** option to **Enabled**.
 - b. Click **Back**.
4. In the System Setup window, click **Device Settings**.
5. On the Device Settings page, select a port for the 25G 41000 Series Adapter.
6. On the Device Level Configuration page (see [Figure 12-2 on page 233](#)):
 - a. Set the **Virtualization Mode** to **SR-IOV**.
 - b. Click **Back**.
7. On the Main Configuration Page, click **Finish**.
8. Save the configuration settings and reboot the system.

9. To enable the needed quantity of VFs per port (in this example, 16 on each port of a dual-port adapter), issue the following command:

```
"esxcfg-module -s "max_vfs=16,16" qedentv"
```

NOTE

Each Ethernet function of the 41000 Series Adapter must have its own entry.

10. Reboot the host.
11. To verify that the changes are complete at the module level, issue the following command:

```
"esxcfg-module -g qedentv"
```

```
[root@localhost:~] esxcfg-module -g qedentv  
qedentv enabled = 1 options = 'max_vfs=16,16'
```

12. To verify if actual VFs were created, issue the `lspci` command as follows:

```
[root@localhost:~] lspci | grep -i QLogic | grep -i 'ethernet\|network' | more  
0000:05:00.0 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx 10/25  
GbE Ethernet Adapter [vmnic6]  
0000:05:00.1 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx 10/25  
GbE Ethernet Adapter [vmnic7]  
0000:05:02.0 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.0_VF_0]  
0000:05:02.1 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.0_VF_1]  
0000:05:02.2 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.0_VF_2]  
0000:05:02.3 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.0_VF_3]  
.  
.  
.  
0000:05:03.7 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.0_VF_15]  
0000:05:0e.0 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.1_VF_0]  
0000:05:0e.1 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.1_VF_1]  
0000:05:0e.2 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series  
10/25 GbE Controller (SR-IOV VF) [PF_0.5.1_VF_2]
```

```
0000:05:0e.3 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series
10/25 GbE Controller (SR-IOV VF) [PF_0.5.1_VF_3]
.
.
.
0000:05:0f.6 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series
10/25 GbE Controller (SR-IOV VF) [PF_0.5.1_VF_14]
0000:05:0f.7 Network controller: QLogic Corp. QLogic FastLinQ QL41xxx Series
10/25 GbE Controller (SR-IOV VF) [PF_0.5.1_VF_15]
```

13. Attach VFs to the VM as follows:
 - a. Power off the VM and attach the VF. (Some OSs support hot-plugging of VFs to the VM.)
 - b. Add a host to a VMware vCenter Server Virtual Appliance (vCSA).
 - c. Click **Edit Settings** of the VM.
14. Complete the Edit Settings dialog box ([Figure 12-15](#)) as follows:
 - a. In the **New Device** box, select **Network**, and then click **Add**.
 - b. For **Adapter Type**, select **SR-IOV Passthrough**.
 - c. For **Physical Function**, select the Marvell VF.
 - d. To save your configuration changes and close this dialog box, click **OK**.

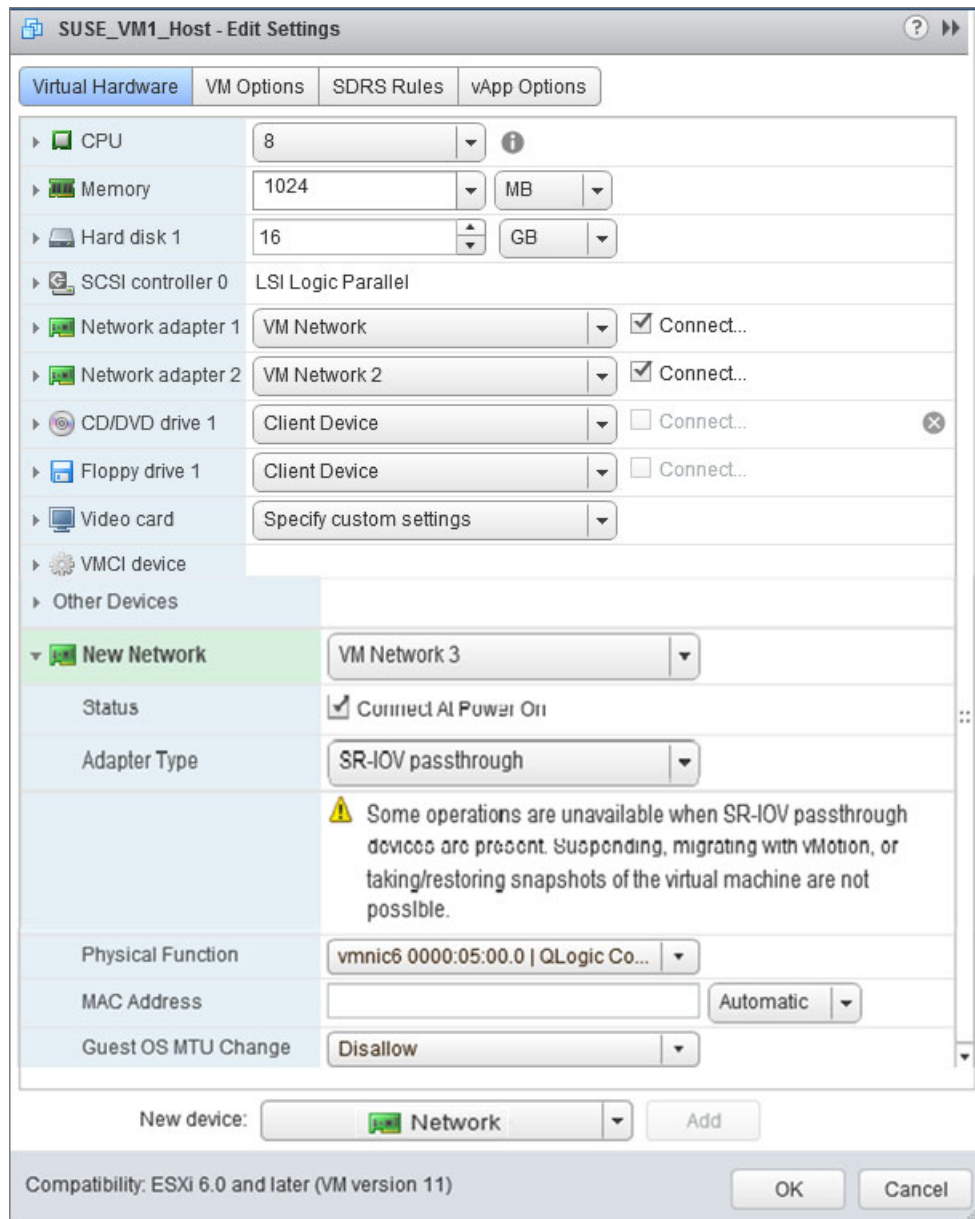


Figure 12-15. VMware Host Edit Settings

- To validate the VFs per port, issue the `esxcli` command as follows:

```
[root@localhost:~] esxcli network sriovnic vf list -n vmnic6
VF ID Active PCI Address Owner World ID
-----
0 true 005:02.0 60591
1 true 005:02.1 60591
```

2	false	005:02.2	-
3	false	005:02.3	-
4	false	005:02.4	-
5	false	005:02.5	-
6	false	005:02.6	-
7	false	005:02.7	-
8	false	005:03.0	-
9	false	005:03.1	-
10	false	005:03.2	-
11	false	005:03.3	-
12	false	005:03.4	-
13	false	005:03.5	-
14	false	005:03.6	-
15	false	005:03.7	-

16. Install the Marvell drivers for the adapters detected in the VM. Use the latest drivers available from your vendor for your host OS (do not use inbox drivers). The same driver version must be installed on the host and the VM.
17. Power on the VM, and then issue the `ifconfig -a` command to verify that the added network interface is listed.
18. As needed, add more VFs in the VM.

13 NVMe-oF Configuration with RDMA

Non-Volatile Memory Express over Fabrics (NVMe-oF) enables the use of alternate transports to PCIe to extend the distance over which an NVMe host device and an NVMe storage drive or subsystem can connect. NVMe-oF defines a common architecture that supports a range of storage networking fabrics for the NVMe block storage protocol over a storage networking fabric. This architecture includes enabling a front-side interface into storage systems, scaling out to large quantities of NVMe devices, and extending the distance within a data center over which NVMe devices and NVMe subsystems can be accessed.

The NVMe-oF configuration procedures and options described in this chapter apply to Ethernet-based RDMA protocols, including RoCE and iWARP. The development of NVMe-oF with RDMA is defined by a technical sub-group of the NVMe organization.

This chapter demonstrates how to configure NVMe-oF for a simple network. The example network comprises the following:

- Two servers: an initiator and a target. The target server is equipped with a PCIe SSD drive.
- Operating system: RHEL 7.6 and later, RHEL 8.x and later, SLES 15.x and later
- Two adapters: One 41000 Series Adapter installed in each server. Each port can be independently configured to use RoCE, RoCEv2, or iWARP as the RDMA protocol over which NVMe-oF runs.
- For RoCE and RoCEv2, an optional switch configured for data center bridging (DCB), relevant quality of service (QoS) policy, and vLANs to carry the NVMe-oF's RoCE/RoCEv2 DCB traffic class priority. The switch is not needed when NVMe-oF is using iWARP.

Figure 13-1 illustrates an example network.

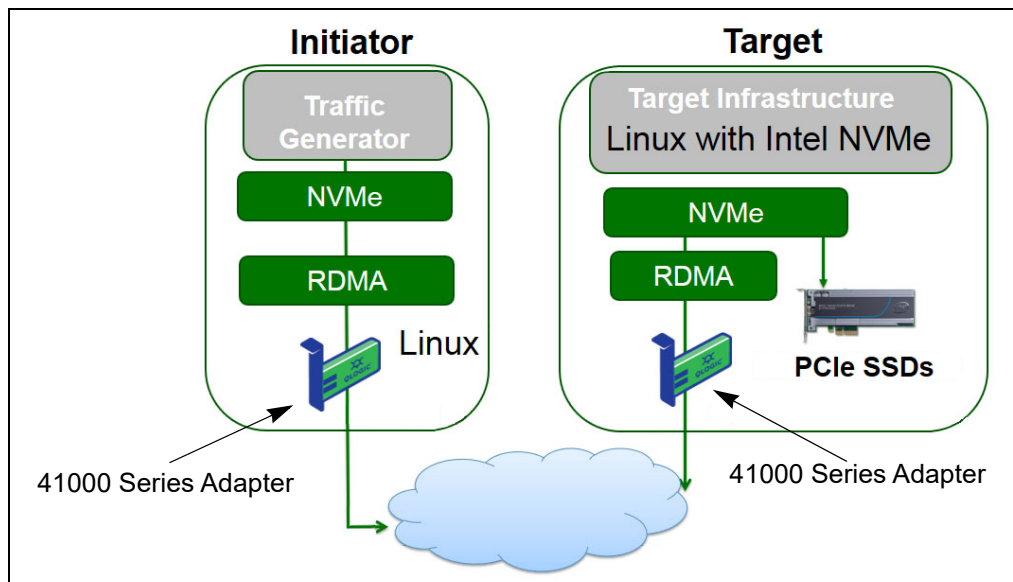


Figure 13-1. NVMe-oF Network

The NVMe-oF configuration process covers the following procedures:

- [Installing Device Drivers on Both Servers](#)
- [Configuring the Target Server](#)
- [Configuring the Initiator Server](#)
- [Preconditioning the Target Server](#)
- [Testing the NVMe-oF Devices](#)
- [Optimizing Performance](#)

Installing Device Drivers on Both Servers

After installing your operating system (SLES 12 SP3), install device drivers on both servers. To upgrade the kernel to the latest Linux upstream kernel, go to:

<https://www.kernel.org/pub/linux/kernel/v4.x/>

1. Install and load the latest FastLinQ drivers (qed, qede, libqedr/qedr) following all installation instructions in the README.
2. (Optional) If you upgraded the OS kernel, you must reinstall and load the latest driver as follows:
 - a. Install the latest FastLinQ firmware following all installation instructions in the README.
 - b. Install the OS RDMA support applications and libraries by issuing the following commands:

```
# yum groupinstall "Infiniband Support"
# yum install tcl-devel libibverbs-devel libnl-devel
glib2-devel libudev-devel lsscsi perftest
# yum install gcc make git ctags ncurses ncurses-devel
openssl* openssl-devel elfutils-libelf-devel*
```
 - c. To ensure that NVMe OFED support is in the selected OS kernel, issue the following command:

```
make menuconfig
```
 - d. Under **Device Drivers**, ensure that the following are enabled (set to **M**):

```
NVM Express block devices
NVM Express over Fabrics RDMA host driver
NVMe Target support
NVMe over Fabrics RDMA target support
```
 - e. (Optional) If the **Device Drivers** options are not already present, rebuild the kernel by issuing the following commands:

```
# make
# make modules
# make modules_install
# make install
```
 - f. If changes were made to the kernel, reboot to that new OS kernel. For instructions on how to set the default boot kernel, go to:

<https://wiki.centos.org/HowTos/Grub2>

3. Enable and start the RDMA service as follows:

```
# systemctl enable rdma.service  
# systemctl start rdma.service
```

Disregard the `RDMA Service Failed` error. All OFED modules required by `qedr` are already loaded.

Configuring the Target Server

Configure the target server after the reboot process. After the server is operating, you cannot change the configuration without rebooting. If you are using a startup script to configure the target server, consider pausing the script (using the `wait` command or something similar) as needed to ensure that each command finishes before executing the next command.

To configure the target service:

1. Load target modules. Issue the following commands after each server reboot:

```
# modprobe qedr  
# modprobe nvmet; modprobe nvmet-rdma  
# lsmod | grep nvme (confirm that the modules are loaded)
```

2. Create the target subsystem NVMe Qualified Name (NQN) with the name indicated by `<nvme-subsystem-name>`. Use the NVMe-oF specifications; for example, `nqn.<YEAR>-<Month>.org.<your-company>`.

```
# mkdir /sys/kernel/config/nvmet/subsystems/<nvme-subsystem-name>  
# cd /sys/kernel/config/nvmet/subsystems/<nvme-subsystem-name>
```

3. Create multiple unique NQNs for additional NVMe devices as needed.
4. Set the target parameters, as listed in [Table 13-1](#).

Table 13-1. Target Parameters

Command	Description
<pre># echo 1 > attr_allow_any_host</pre>	Allows any host to connect.
<pre># mkdir namespaces/1</pre>	Creates a namespace.

Table 13-1. Target Parameters (Continued)

Command	Description
<code># echo -n /dev/nvme0n1 >namespaces/1/device_path</code>	Sets the NVMe device path. The NVMe device path can differ between systems. Check the device path using the <code>lsblk</code> command. This system has two NVMe devices: <code>nvme0n1</code> and <code>nvme1n1</code> . <pre>[root@localhost home]# lsblk NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT nvme1n1 259:0 0 372.6G 0 disk sda 8:0 0 1.1T 0 disk ├─sda2 8:2 0 505G 0 part / ├─sda3 8:3 0 8G 0 part [SWAP] └─sda1 8:1 0 1G 0 part /boot/efi nvme0n1 259:1 0 372.6G 0 disk</pre>
<code># echo 1 > namespaces/1/enable</code>	Enables the namespace.
<code># mkdir /sys/kernel/config/nvmet/ports/1</code> <code># cd /sys/kernel/config/nvmet/ports/1</code>	Creates NVMe port 1.
<code># echo 1.1.1.1 > addr_traddr</code>	Sets the same IP address. For example, 1.1.1.1 is the IP address for the target port of the 41000 Series Adapter.
<code># echo rdma > addr_trtype</code>	Sets the transport type RDMA.
<code># echo 4420 > addr_trsvcid</code>	Sets the RDMA port number. The socket port number for NVMe-oF is typically 4420. However, any port number can be used if it is used consistently throughout the configuration.
<code># echo ipv4 > addr_adrfam</code>	Sets the IP address type.

5. Create a symbolic link (symlink) to the newly created NQN subsystem:

```
# ln -s /sys/kernel/config/nvmet/subsystems/  
nvme-subsystem-name subsystems/nvme-subsystem-name
```

6. Confirm that the NVMe target is listening on the port as follows:

```
# dmesg | grep nvmet_rdma  
[ 8769.470043] nvmet_rdma: enabling port 1 (1.1.1.1:4420)
```

Configuring the Initiator Server

You must configure the initiator server after the reboot process. After the server is operating, you cannot change the configuration without rebooting. If you are using a startup script to configure the initiator server, consider pausing the script (using the `wait` command or something similar) as needed to ensure that each command finishes before executing the next command.

To configure the initiator server:

1. Load the NVMe modules. Issue these commands after each server reboot:

```
# modprobe qedr
# modprobe nvme-rdma
```

2. Download, compile and install the `nvme-cli` initiator utility. Issue these commands at the first configuration—you do not need to issue these commands after each reboot.

```
# git clone https://github.com/linux-nvme/nvme-cli.git
# cd nvme-cli
# make && make install
```

3. Verify the installation version as follows:

```
# nvme version
```

4. Discover the NVMe-oF target as follows:

```
# nvme discover -t rdma -a 1.1.1.1 -s 1023
```

Make note of the subsystem NQN (`subnqn`) of the discovered target (Figure 13-2) for use in Step 5.

```
[root@localhost home]# nvme discover -t rdma -a 1.1.1.1 -s 1023

Discovery Log Number of Records 1, Generation counter 1
====Discovery Log Entry 0====
trtype: rdma
adrfam: ipv4
subtype: nvme subsystem
treq: not specified
portid: 1
trsvcid: 1023

subnqn: nvme-qlogic-tgt1
traddr: 1.1.1.1

rdma_prtype: not specified
rdma_qptype: connected
rdma_cms: rdma-cm
rdma_pkey: 0x0000
```

Figure 13-2. Subsystem NQN

5. Connect to the discovered NVMe-oF target (`nvme-qlogic-tgt1`) using the NQN. Issue the following command after each server reboot. For example:

```
# nvme connect -t rdma -n nvme-qlogic-tgt1 -a 1.1.1.1 -s 1023
```

6. Confirm the NVMe-oF target connection with the NVMe-oF device as follows:

```
# dmesg | grep nvme
# lsblk
# list nvme
```

Figure 13-3 shows an example.

```
[root@localhost home] #dmesg | grep nvme
[ 233.645554] nvme nvme0: new ctrl: NQN "nvme-qlogic-tgt1", addr 1.1.1.1:1023
[root@localhost home] # lsblk
NAME        MAJ:MIN RM   SIZE RO TYPE MOUNTPOINT
sdb         8:0     0    1.1T 0 disk
├─sdb2      8:2     0   493.2G 0 part /
├─sdb3      8:3     0     8G 0 part [SWAP]
└─sdb1      8:1     0     1G 0 part /boot/efi
nvme0n1     259:0   0   372.6G 0 disk
[root@localhost home] # nvme list
Node          SN                      Model      Namespace  Usage          Format          FW Rev
-----
|/dev/nvme0n1 7a591f3ec788a367       Linux      1           1.60 TB / 1.60 TB 512 B + 0 B 4.13.8
```

Figure 13-3. Confirm NVMe-oF Connection

Preconditioning the Target Server

NVMe target servers that are tested out-of-the-box show a higher-than-expected performance. Before running a benchmark, the target server needs to be *prefilled* or *preconditioned*.

To precondition the target server:

1. Secure-erase the target server with vendor-specific tools (similar to formatting). This test example uses an Intel NVMe SSD device, which requires the Intel Data Center Tool that is available at the following link:
<https://downloadcenter.intel.com/download/23931/Intel-Solid-State-Drive-Data-Center-Tool>
2. Precondition the target server (`nvme0n1`) with data, which guarantees that all available memory is filled. This example uses the “DD” disk utility:

```
# dd if=/dev/zero bs=1024k of=/dev/nvme0n1
```

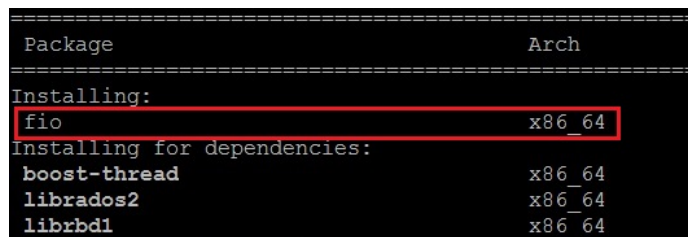
Testing the NVMe-oF Devices

Compare the latency of the local NVMe device on the target server with that of the NVMe-oF device on the initiator server to show the latency that NVMe adds to the system.

To test the NVMe-oF device:

1. Update the Repository (Repo) source and install the Flexible Input/Output (FIO) benchmark utility on both the target and initiator servers by issuing the following commands:

```
# yum install epel-release
# yum install fio
```



```
Package                               Arch
-----                               -
Installing:
fio                                   x86_64
Installing for dependencies:
boost-thread                          x86_64
librados2                              x86_64
librbd1                                x86_64
```

Figure 13-4. FIO Utility Installation

2. Run the FIO utility to measure the latency of the initiator NVMe-oF device. Issue the following command:

```
# fio --filename=/dev/nvme0n1 --direct=1 --time_based
--rw=randread --refill_buffers --norandommap --randrepeat=0
--ioengine=libaio --bs=4k --iodepth=1 --numjobs=1
--runtime=60 --group_reporting --name=temp.out
```

FIO reports two latency types: submission and completion. Submission latency (slat) measures application-to-kernel latency. Completion latency (clat), measures end-to-end kernel latency. The industry-accepted method is to read *clat percentiles* in the 99.00th range.

3. Run FIO to measure the latency of the local NVMe device on the target server. Issue the following command:

```
# fio --filename=/dev/nvme0n1 --direct=1 --time_based
--rw=randread --refill_buffers --norandommap --randrepeat=0
--ioengine=libaio --bs=4k --iodepth=1 --numjobs=1
--runtime=60 --group_reporting --name=temp.out
```

The total latency that results from the use of NVMe-oF is the difference between the initiator device NVMe-oF latency and the target device NVMe-oF latency.

4. Run FIO to measure bandwidth of the local NVMe device on the target server. Issue the following command:

```
fio --verify=crc32 --do_verify=1 --bs=8k --numjobs=1
--iodepth=32 --loops=1 --ioengine=libaio --direct=1
--invalidate=1 --fsync_on_close=1 --randrepeat=1
--norandommap --time_based --runtime=60
--filename=/dev/nvme0n1 --name=Write-BW-to-NVMe-Device
--rw=randwrite
```

Where `--rw` can be `randread` for reads only, `randwrite` for writes only, or `randrw` for reads and writes.

Optimizing Performance

To optimize performance on both initiator and target servers:

1. Configure the following system BIOS settings:
 - Power Profiles = 'Max Performance' or equivalent
 - ALL C-States = Disabled
 - Hyperthreading = Disabled
2. Configure the Linux kernel parameters by editing the `grub` file (`/etc/default/grub`).
 - a. Add parameters to end of line `GRUB_CMDLINE_LINUX`:

```
GRUB_CMDLINE_LINUX="nosoftlockup intel_idle.max_cstate=0
processor.max_cstate=1 mce=ignore_ce idle=poll"
```
 - b. Save the `grub` file.
 - c. Rebuild the `grub` file.
 - To rebuild the `grub` file for a legacy BIOS boot, issue the following command:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg (Legacy BIOS boot)
```
 - To rebuild the `grub` file for an EFI boot, issue the following command:

```
# grub2-mkconfig -o /boot/efi/EFI/<os>/grub.cfg (EFI boot)
```
 - d. Reboot the server to implement the changes.
3. Set the IRQ affinity for all 41000 Series Adapters. The `multi_rss-affin.sh` file is a script file that is listed in [“IRQ Affinity \(multi_rss-affin.sh\)” on page 259](#).

```
# systemctl stop irqbalance
# ./multi_rss-affin.sh eth1
```


NOTE

A different version of this script, `qedr_affin.sh`, is in the 41000 Linux Source Code Package in the `\add-ons\performance\roce` directory. For an explanation of the IRQ affinity settings, refer to the `multiple_irqs.txt` file in that directory.

4. Set the CPU frequency. The `cpufreq.sh` file is a script that is listed in “CPU Frequency (`cpufreq.sh`)” on page 260.

```
# ./cpufreq.sh
```

The following sections list the scripts that are used in [Steps 3](#) and [4](#).

IRQ Affinity (`multi_rss-affin.sh`)

The following script sets the IRQ affinity.

```
#!/bin/bash
#RSS affinity setup script
#input: the device name (ethX)
#OFFSET=0    0/1    0/1/2    0/1/2/3
#FACTOR=1    2      3        4
OFFSET=0
FACTOR=1
LASTCPU='cat /proc/cpuinfo | grep processor | tail -n1 | cut -d":" -f2'
MAXCPUID='echo 2 $LASTCPU ^ p | dc'
OFFSET='echo 2 $OFFSET ^ p | dc'
FACTOR='echo 2 $FACTOR ^ p | dc'
CPUID=1

for eth in $*; do

NUM='grep $eth /proc/interrupts | wc -l'
NUM_FP=$(( ${NUM} ))

INT='grep -m 1 $eth /proc/interrupts | cut -d ":" -f 1'

echo "$eth: ${NUM} (${NUM_FP} fast path) starting irq ${INT}"

CPUID=$(( CPUID*OFFSET ))
for ((A=1; A<=${NUM_FP}; A=${A}+1)) ; do
INT='grep -m $A $eth /proc/interrupts | tail -1 | cut -d ":" -f 1'
SMP='echo $CPUID 16 o p | dc'
echo ${INT} smp affinity set to ${SMP}
```

```
echo ${SMP} > /proc/irq/${INT}/smp_affinity
CPUID=$((CPUID*FACTOR))
if [ ${CPUID} -gt ${MAXCPUID} ]; then
CPUID=1
CPUID=$((CPUID*OFFSET))
fi
done
done
```

CPU Frequency (cpufreq.sh)

The following script sets the CPU frequency.

```
#Usage "./nameofscript.sh"
grep -E '^model name|^cpu MHz' /proc/cpuinfo
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
for CPUFREQ in /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor; do [ -f
$CPUFREQ ] || continue; echo -n performance > $CPUFREQ; done
cat /sys/devices/system/cpu/cpu0/cpufreq/scaling_governor
```

To configure the network or memory settings:

NOTE

The following commands apply only to the initiator server.

```
# echo 0 > /sys/block/nvme0n1/queue/add_random
# echo 2 > /sys/block/nvme0n1/queue/nomerges
```

Configuring NVMe-oF on ESXi 7.0

This section provides information for configuring NVMe-oF for VMware ESXi 7.0.

Before you configure NVMe-oF for ESXi 7.0, perform the following steps:

1. Ensure that the following are installed on the VMware ESXi 7.0 system:
 - Marvell FastLinQ Adapter package
 - NIC driver
 - RoCE driver

In addition, ensure that the devices are listed.

2. Enable RoCE in the NIC (qedentv) module and set the parameter `num_ns` to `32` in the RoCE (qedrntv) module by issuing the following commands:

```
[root@nvme:~] esxcfg-module -g qedentv
```

```
qedrntv enabled = 1 options = 'enable_roce=1'  
[root@nvme: ~] esxcfg-module -g qedrntv  
qedrntv enabled = 1 options = 'num_ns=32'
```

To configure NVME-oF on a ESXi 7.0 system:

1. Discover the NVMe controller:
 - a. Create a vSwitch and add an uplink to that vSwitch.
 - b. Ensure that you have an RDMA device by issuing the following command:

```
# esxcli rdma device list
```
 - c. Enable NVME fabrics by issuing the following command:

```
# esxcli nvme fabrics enable -p RDMA -d vmrdma0
```
2. Discover and connect the target.

NOTE

Ensure that the NVMe-oF hardware target supports *fused* commands.

- a. Discover the target by issuing the following command:

```
# esxcli nvme fabrics discover -a vmhbaXX -i <nvme rdma target ip>
```
 - b. Connect the target by issuing the following command:

```
# esxcli nvme fabrics connect -a vmhbaXX -i <nvme rdma target ip>  
-s <nvme target nqn id>
```
3. Check for NVME devices by issuing the following command:

```
#esxcfg-scsidevs -l
```

14 VXLAN Configuration

This chapter provides instructions for:

- [Configuring VXLAN in Linux](#)
- [“Configuring VXLAN in VMware” on page 264](#)
- [“Configuring VXLAN in Windows Server 2016” on page 265](#)

Configuring VXLAN in Linux

To configure VXLAN in Linux:

1. Download, extract, and configure the openvswitch (OVS) tar ball.
 - a. Download the appropriate openvswitch release from the following location:
<http://www.openvswitch.org/download/>
 - b. Extract the tar ball by navigating to the directory where you downloaded the openvswitch release, and then issue the following command:

```
./configure;make;make install
```

 (compilation)
 - c. Configure openvswitch by issuing the following commands:

```
modprobe -v openvswitch
export PATH=$PATH:/usr/local/share/openvswitch/scripts
ovs-ctl start
ovs-ctl status
```

When running `ovs-ctl status`, the `ovsdb-server` and `ovs-vswitchd` should be running with `pid`. For example:

```
[root@localhost openvswitch-2.11.1]# ovs-ctl status
ovsdb-server is running with pid 8479
ovs-vswitchd is running with pid 8496
```

2. Create the bridge.

- a. To configure Host 1, issue the following commands:

```
ovs-vsctl add-br br0
ovs-vsctl add-br br1
ovs-vsctl add-port br0 eth0
ifconfig eth0 0 && ifconfig br0 192.168.1.10 netmask 255.255.255.0
route add default gw 192.168.1.1 br0
ifconfig br1 10.1.2.10 netmask 255.255.255.0
ovs-vsctl add-port br1 vx1 -- set interface vx1 type=vxlan
options:remote_ip=192.168.1.11 (peer IP address)
```

- b. To configure Host 2, issue the following commands:

```
ovs-vsctl add-br br0
ovs-vsctl add-br br1
ovs-vsctl add-port br0 eth0
ifconfig eth0 0 && ifconfig br0 192.168.1.11 netmask 255.255.255.0
route add default gw 192.168.1.1 br0
ifconfig br1 10.1.2.11 netmask 255.255.255.0
ovs-vsctl add-port br1 vx1 -- set interface vx1 type=vxlan options:
remote_ip=192.168.1.10
```

3. Verify the configuration.

Run traffic between the host and peer using iperf. Ensure that the firewall and iptables stop and clean, respectively.

4. Configure the bridge as a passthrough to the VMs, and then check connectivity from the VM to the Peer.
 - a. Create a VM through virt-manager.
 - b. As there is no option to attach bridge `br1` through virt-manager, change the xml file as follows

Issue the following command:

```
command: virsh edit vm1
```

Add the following code:

```
<interface type='bridge'>  
<source bridge='br1' />  
<virtualport type='openvswitch'>  
<parameters/>  
</virtualport>  
<model type='virtio' />  
</interface>
```

- c. Power up the VM and check `br1` interfaces.

Ensure that `br1` is in the OS. The `br1` interface is named `eth0`, `ens7`; manually configure the static IP through the network device file and assign the same subnet IP to the peer (Host 2 VM).

Run traffic from the peer to the VM.

NOTE

You can use this procedure to test other tunnels, such as Generic Network Virtualization Encapsulation (GENEVE) and generic routing encapsulation (GRE), with OVS.

If you do not want to use OVS, you can continue with the legacy bridge option `brctl`.

Configuring VXLAN in VMware

To configure VXLAN in VMware, follow the instructions in the following locations:

<https://docs.vmware.com/en/VMware-NSX-Data-Center-for-vSphere/6.3/com.vmware.nsx.cross-vcenter-install.doc/GUID-49BAECC2-B800-4670-AD8C-A5292ED6BC19.html>

<https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/products/nsx/vmw-nsx-network-virtualization-design-guide.pdf>

<https://pubs.vmware.com/nsx-63/topic/com.vmware.nsx.troubleshooting.doc/GUID-EA1DB524-DD2E-4157-956E-F36BDD20CDB2.html>

<https://communities.vmware.com/api/core/v3/attachments/124957/data>

Configuring VXLAN in Windows Server 2016

VXLAN configuration in Windows Server 2016 includes:

- [Enabling VXLAN Offload on the Adapter](#)
- [Deploying a Software Defined Network](#)

Enabling VXLAN Offload on the Adapter

To enable VXLAN offload on the adapter:

1. Open the miniport properties, and then click the **Advanced** tab.
2. On the adapter properties' Advanced page ([Figure 14-1](#)) under **Property**, select **VXLAN Encapsulated Task Offload**.

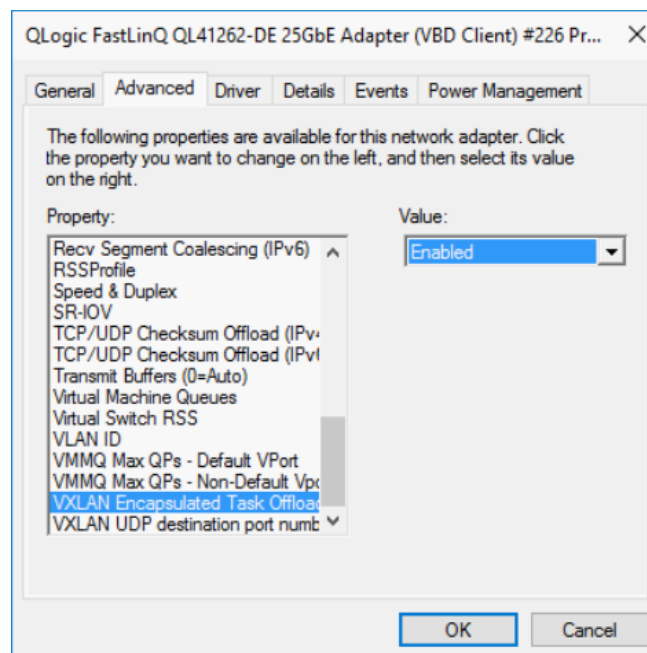


Figure 14-1. Advanced Properties: Enabling VXLAN

3. Set the **Value** to **Enabled**.
4. Click **OK**.

Deploying a Software Defined Network

To take advantage of VXLAN encapsulation task offload on virtual machines, you must deploy a Software Defined Networking (SDN) stack that utilizes a Microsoft Network Controller.

For more details, refer to the following Microsoft TechNet link on Software Defined Networking:

<https://technet.microsoft.com/en-us/windows-server-docs/networking/sdn/software-defined-networking--sdn->

15 Windows Server 2016

This chapter provides the following information for Windows Server 2016:

- [Configuring RoCE Interfaces with Hyper-V](#)
- [“RoCE over Switch Embedded Teaming” on page 274](#)
- [“Configuring QoS for RoCE” on page 276](#)
- [“Configuring VMMQ” on page 284](#)
- [“Configuring Storage Spaces Direct” on page 288](#)

Configuring RoCE Interfaces with Hyper-V

In Windows Server 2016, Hyper-V with Network Direct Kernel Provider Interface (NDKPI) Mode-2, host virtual network adapters (host virtual NICs) support RDMA.

NOTE

DCBX is required for RoCE over Hyper-V. To configure DCBX, either:

- [Configure through the HII \(see “Preparing the Adapter” on page 136\).](#)
 - [Configure using QoS \(see “Configuring QoS for RoCE” on page 276\).](#)
-

RoCE configuration procedures in this section include:

- [Creating a Hyper-V Virtual Switch with an RDMA NIC](#)
- [Adding a vLAN ID to Host Virtual NIC](#)
- [Verifying If RoCE is Enabled](#)
- [Adding Host Virtual NICs \(Virtual Ports\)](#)
- [Mapping the SMB Drive](#)
- [Running RoCE Traffic](#)

Creating a Hyper-V Virtual Switch with an RDMA NIC

Follow the procedures in this section to create a Hyper-V virtual switch and then enable RDMA in the host vNIC.

To create a Hyper-V virtual switch with an RDMA virtual NIC:

1. On all physical interfaces, set the value of the **NetworkDirect Functionality** parameter to **Enabled**.
2. Launch Hyper-V Manager.
3. Click **Virtual Switch Manager** (see [Figure 15-1](#)).

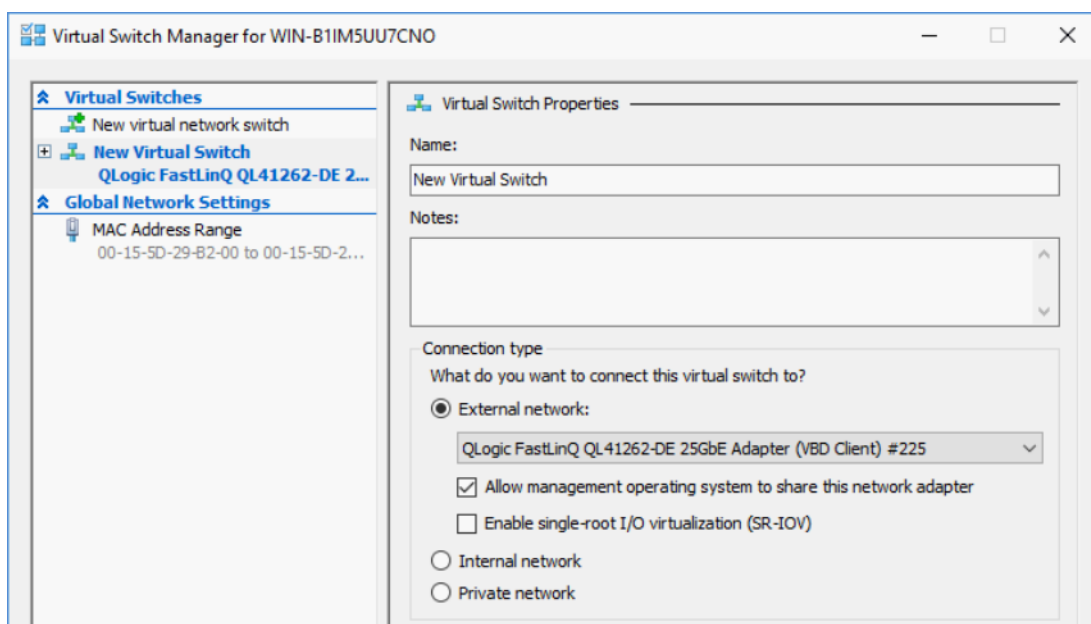


Figure 15-1. Enabling RDMA in Host Virtual NIC

4. Create a virtual switch.
5. Select the **Allow management operating system to share this network adapter** check box.

In Windows Server 2016, a new parameter—Network Direct (RDMA)—is added in the Host virtual NIC.

To enable RDMA in a host virtual NIC:

1. Open the Hyper-V Virtual Ethernet Adapter Properties window.
2. Click the **Advanced** tab.

3. On the Advanced page (Figure 15-2):
 - a. Under **Property**, select **Network Direct (RDMA)**.
 - b. Under **Value**, select **Enabled**.
 - c. Click **OK**.

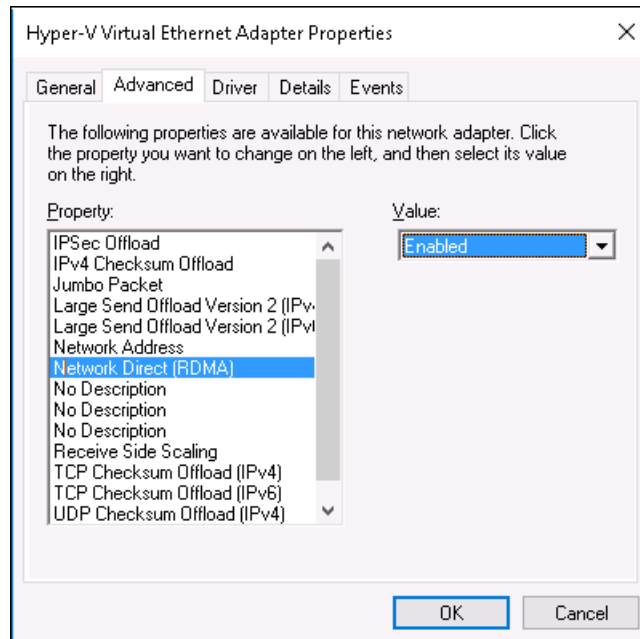


Figure 15-2. Hyper-V Virtual Ethernet Adapter Properties

4. To enable RDMA through PowerShell, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Enable-NetAdapterRdma "vEthernet  
(New Virtual Switch)"  
PS C:\Users\Administrator>
```

Adding a vLAN ID to Host Virtual NIC

To add a vLAN ID to a host virtual NIC:

1. To find the host virtual NIC name, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Get-VMNetworkAdapter -ManagementOS
```

Figure 15-3 shows the command output.

```
PS C:\Users\Administrator> Get-VMNetworkAdapter -ManagementOS
Name                IsManagementOs VMName SwitchName           MacAddress           Status IPAddresses
-----
New Virtual Switch True                New Virtual Switch 000E1EC41F0B {Ok}
```

Figure 15-3. Windows PowerShell Command: Get-VMNetworkAdapter

2. To set the vLAN ID to the host virtual NIC, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Set-VMNetworkAdapterVlan
-VMNetworkAdapterName "New Virtual Switch" -VlanId 5 -Access
-ManagementOS
```

NOTE

Note the following about adding a vLAN ID to a host virtual NIC:

- A vLAN ID must be assigned to a host virtual NIC. The same vLAN ID must be assigned to ports on the switch.
 - Make sure that the vLAN ID is not assigned to the physical interface when using a host virtual NIC for RoCE.
 - If you are creating more than one host virtual NIC, you can assign a different vLAN to each host virtual NIC.
-

Verifying If RoCE is Enabled

To verify if the RoCE is enabled:

- Issue the following Windows PowerShell command:

```
Get-NetAdapterRdma
```

Command output lists the RDMA supported adapters as shown in Figure 15-4.

```
PS C:\Users\Administrator> Get-NetAdapterRdma
Name                InterfaceDescription           Enabled
-----
vEthernet (New Virtual... Hyper-V Virtual Ethernet Adapter True
```

Figure 15-4. Windows PowerShell Command: Get-NetAdapterRdma

Adding Host Virtual NICs (Virtual Ports)

To add host virtual NICs:

1. To add a host virtual NIC, issue the following command:

```
Add-VMNetworkAdapter -SwitchName "New Virtual Switch" -Name SMB - ManagementOS
```
2. Enable RDMA on host virtual NICs as shown in [“To enable RDMA in a host virtual NIC:” on page 268](#).
3. To assign a vLAN ID to the virtual port, issue the following command:

```
Set-VMNetworkAdapterVlan -VMNetworkAdapterName SMB -VlanId 5 -Access -ManagementOS
```

Mapping the SMB Drive

The following sections describe how to map a system management bus (SMB) drive in IPv4 and IPv6 environments.

IPv4 Network Drive Mapping

To use IPv4 addresses for RoCE v2 (or iWARP) on SMB-Direct or Storage Spaces Direct:

1. In the Windows Networking Ethernet Properties, assign Servers A and B IPv4 address as follows:
 - a. Server A, port 1 IPv4 address: **192.168.10.10**
 - b. Server B, port 1 IPv4 address: **192.168.10.20**
2. Ping from Server A to Server B using the IPv4 address in [Step 1](#).
3. Map a network drive on Server A from Server B using the IPv4 address and created shared drive instance, for example:
HYPERLINK "file://192.168.10.20/share1"\\192.168.10.20\share1)
4. Start DiskSpd or another test application and view the RDMA statistics in Windows Perfmon.

IPv6 Network Drive Mapping

To use IPv6 addresses for RoCE v2 (or iWARP) on SMB-Direct or Storage Spaces Direct, use the *host names* instead of the IPv6 address when pinging or mapping the drive (for example, when you are using static IPv6 addresses on both the initiator and target servers (such as when there is no DNS server in your test setup)).

To use IPv6 addresses for RoCE v2 (or iWARP) on SMB-Direct or Storage Spaces Direct:

1. In the Windows Networking Ethernet Properties, assign Servers A and B IPv6 addresses as follows:
 - a. Server A, port 1, IPv6 address: **be10::30**
 - b. Server B, port 1, IPv6 address: **be10::40**
2. On Server A, add the IPv6 and other Server B's name to the entry in the Windows OS *hosts* file:
be10::40 serverb.sc.marvell.com
3. On Server B, add the IPv6 and other Server A's name to the entry in the Windows OS *hosts* file:
be10::30 servera.sc.marvell.com
4. Ping to Server A between Server B using the host name.
When issuing a **ping server** command, note that it uses the configured IPv6 address (be10::40).
5. Create a shared drive on Server B, for example, **share1**.
6. Map a network drive on Server A from Server B using the host name and created shared drive instance, for example:
HYPERLINK "file://serverb/share1"\\serverb\share1)
7. Start DiskSpd or another test application and view the RDMA statistics in Windows Perfmon.

Running RoCE Traffic

To run the RoCE traffic:

1. Launch the Performance Monitor (Perfmon).
2. Complete the Add Counters dialog box (Figure 15-5) as follows:
 - a. Under **Available counters**, select **RDMA Activity**.
 - b. Under **Instances of selected object**, select the adapter.
 - c. Click **Add**.

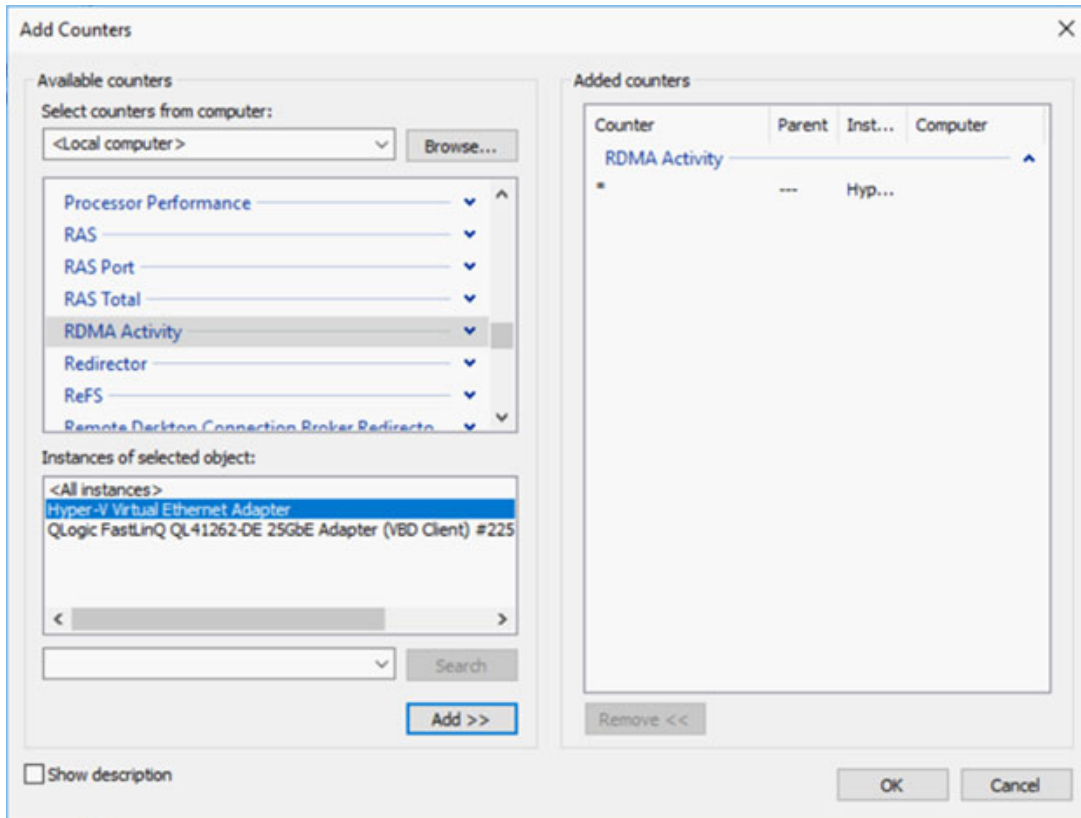


Figure 15-5. Add Counters Dialog Box

If the RoCE traffic is running, counters appear as shown in [Figure 15-6](#).

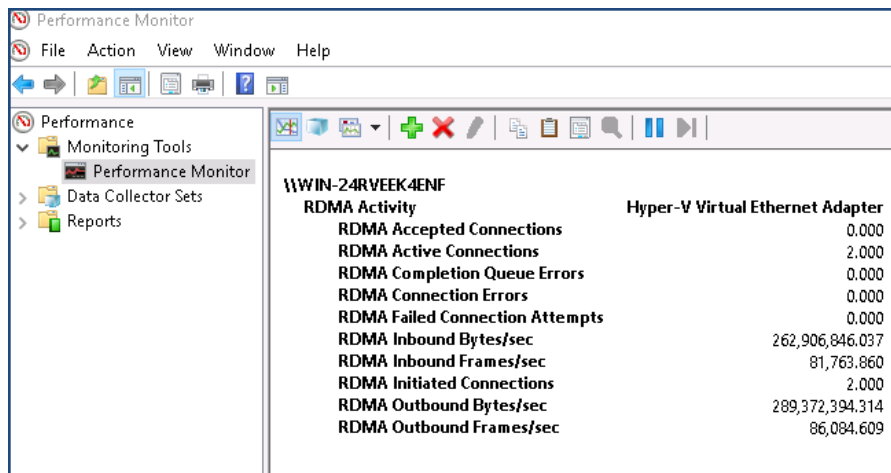


Figure 15-6. Performance Monitor Shows RoCE Traffic

Using IPv6 Addressing for RoCE (and iWARP) on SMB Direct/S2D

To use IPv6 addresses with RoCEv2 (or iWARP) on SMB Direct or Storage Spaces Direct (S2D), use the host names instead of the IPv6 address when pinging or mapping the drive.

For example, when using static IPv6 addresses on both the initiator and target servers (such as when there is no DNS server in your test setup), perform the following steps:

1. Server A, port 1: assign the IPv6 address in Windows Networking Ethernet Properties:
`be10::30`
2. Server B, port 1: assign the IPv6 address in Windows Networking Ethernet Properties:
`be10::40`
3. On Server A, add the IPv6 and other Server B's name to the entry in the Windows OS hosts file:
`be10::40 serverb.sc.marvell.com`
4. On Server B, add the IPv6 and other Server A's name to the entry in the Windows OS hosts file:
`be10::30 servera.sc.marvell.com`
5. Ping from Server A to Server B (and visa versa), using the host name (**ping serverb**).
You should see that it uses the configured IPv6 address for Server B (`be10::40`).
6. Map a network drive on Server A to Server B using the host name and created storage drive instance (such as `\\serverb\ram_disk_1`).
7. Start sqlio (or DiskSpd, IOMeter, or another test application) and observe the RDMA stats in Windows Perfmon.

RoCE over Switch Embedded Teaming

Switch Embedded Teaming (SET) is Microsoft's alternative NIC teaming solution available to use in environments that include Hyper-V and the Software Defined Networking (SDN) stack in Windows Server 2016 Technical Preview. SET integrates limited NIC Teaming functionality into the Hyper-V Virtual Switch.

Use SET to group between one and eight physical Ethernet network adapters into one or more software-based virtual network adapters. These adapters provide fast performance and fault tolerance if a network adapter failure occurs. To be placed on a team, SET member network adapters must all be installed in the same physical Hyper-V host.

RoCE over SET procedures included in this section:

- [Creating a Hyper-V Virtual Switch with SET and RDMA Virtual NICs](#)
- [Enabling RDMA on SET](#)
- [Assigning a vLAN ID on SET](#)
- [Running RDMA Traffic on SET](#)

Creating a Hyper-V Virtual Switch with SET and RDMA Virtual NICs

To create a Hyper-V virtual switch with SET and RDMA virtual NICs:

- To create SET, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> New-VMSwitch -Name SET  
-NetAdapterName "Ethernet 2","Ethernet 3"  
-EnableEmbeddedTeaming $true
```

[Figure 15-7](#) shows command output.

```
PS C:\Users\Administrator> New-VMSwitch -Name SET -NetAdapterName "Ethernet 2","Ethernet 3" -EnableEmbeddedTeaming $true  
Name SwitchType NetAdapterInterfaceDescription  
-----  
SET External Teamed-Interface
```

Figure 15-7. Windows PowerShell Command: New-VMSwitch

Enabling RDMA on SET

To enable RDMA on SET:

1. To view SET on the adapter, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Get-NetAdapter "vEthernet (SET)"
```

[Figure 15-8](#) shows command output.

```
PS C:\Users\Administrator> Get-NetAdapter "vEthernet (SET)"  
Name InterfaceDescription ifIndex Status MacAddress LinkSpeed  
-----  
vEthernet (SET) Hyper-V Virtual Ethernet Adapter 46 Up 00-0E-1E-C4-04-F8 50 Gbps
```

Figure 15-8. Windows PowerShell Command: Get-NetAdapter

2. To enable RDMA on SET, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Enable-NetAdapterRdma "vEthernet  
(SET) "
```

Assigning a vLAN ID on SET

To assign a vLAN ID on SET:

- To assign a vLAN ID on SET, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Set-VMNetworkAdapterVlan  
-VMNetworkAdapterName "SET" -VlanId 5 -Access -ManagementOS
```

NOTE

Note the following when adding a vLAN ID to a host virtual NIC:

- Make sure that the vLAN ID is not assigned to the physical Interface when using a host virtual NIC for RoCE.
 - If you are creating more than one host virtual NIC, a different vLAN can be assigned to each host virtual NIC.
-

Running RDMA Traffic on SET

For information about running RDMA traffic on SET, go to:

<https://technet.microsoft.com/en-us/library/mt403349.aspx>

Configuring QoS for RoCE

The two methods of configuring quality of service (QoS) include:

- [Configuring QoS by Disabling DCBX on the Adapter](#)
- [Configuring QoS by Enabling DCBX on the Adapter](#)

Configuring QoS by Disabling DCBX on the Adapter

All configuration must be completed on all of the systems in use before configuring QoS by disabling DCBX on the adapter. The priority-based flow control (PFC), enhanced transition services (ETS), and traffic classes configuration must be the same on the switch and server.

To configure QoS by disabling DCBX:

1. Using UEFI HII, disable DCBX on the adapter port.
2. Using UEFI HII, set the **RoCE Priority** to 0 on the adapter port.

3. To install the DCB role in the host, issue the following Windows PowerShell command:

```
PS C:\Users\Administrators> Install-WindowsFeature  
Data-Center-Bridging
```

4. To set the **DCBX Willing** mode to **False**, issue the following Windows PowerShell command:

```
PS C:\Users\Administrators> set-NetQosDcbxSetting -Willing 0
```

5. Enable QoS in the miniport as follows:
 - a. Open the miniport Properties, and then click the **Advanced** tab.
 - b. On the adapter properties' Advanced page (Figure 15-9) under **Property**, select **Quality of Service**, and then set the value to **Enabled**.
 - c. Click **OK**.

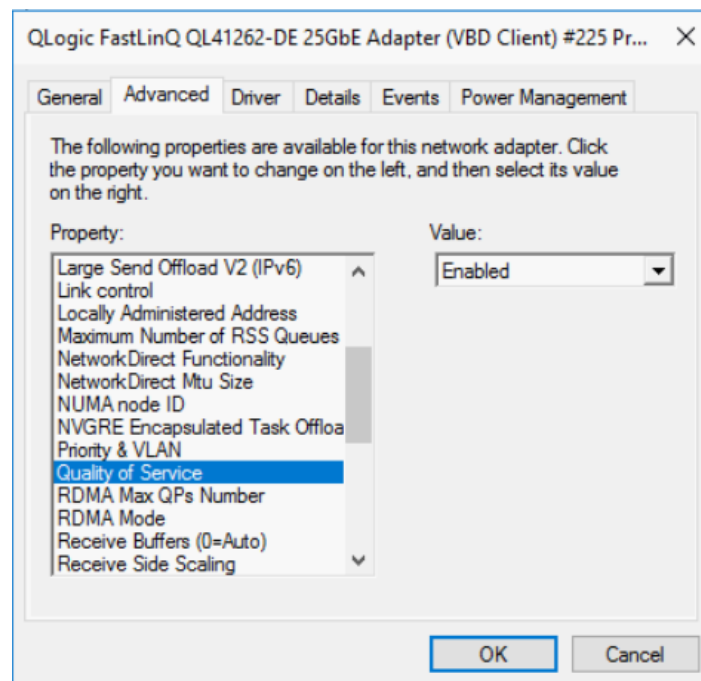


Figure 15-9. Advanced Properties: Enable QoS

6. Assign the VLAN ID to the interface as follows:
 - a. Open the miniport properties, and then click the **Advanced** tab.
 - b. On the adapter properties' Advanced page (Figure 15-10) under **Property**, select **VLAN ID**, and then set the value.

- c. Click **OK**.

NOTE

The preceding step is required for priority flow control (PFC).

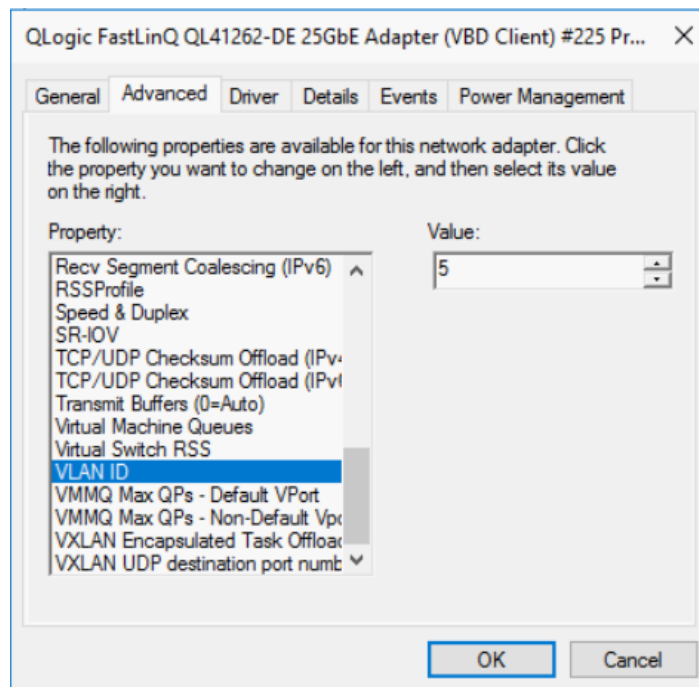


Figure 15-10. Advanced Properties: Setting VLAN ID

7. To enable PFC for RoCE on a specific priority, issue the following command:

```
PS C:\Users\Administrators> Enable-NetQoSFlowControl  
-Priority 5
```

NOTE

If configuring RoCE over Hyper-V, do not assign a vLAN ID to the physical interface.

8. To disable priority flow control on any other priority, issue the following commands:

```
PS C:\Users\Administrator> Disable-NetQoSFlowControl 0,1,2,3,4,6,7
```

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

```
Priority    Enabled    PolicySet    IfIndex IfAlias
```

```
-----
```

0	False	Global
1	False	Global
2	False	Global
3	False	Global
4	False	Global
5	True	Global
6	False	Global
7	False	Global

9. To configure QoS and assign relevant priority to each type of traffic, issue the following commands (where Priority 5 is tagged for RoCE and Priority 0 is tagged for TCP):

```
PS C:\Users\Administrators> New-NetQosPolicy "SMB"  
-NetDirectPortMatchCondition 445 -PriorityValue8021Action 5 -PolicyStore  
ActiveStore
```

```
PS C:\Users\Administrators> New-NetQosPolicy "TCP" -IPProtocolMatchCondition  
TCP -PriorityValue8021Action 0 -Policystore ActiveStore
```

```
PS C:\Users\Administrator> Get-NetQosPolicy -PolicyStore activestore
```

```
Name           : tcp  
Owner          : PowerShell / WMI  
NetworkProfile : All  
Precedence    : 127  
JobObject     :  
IPProtocol    : TCP  
PriorityValue  : 0
```

```
Name           : smb  
Owner          : PowerShell / WMI  
NetworkProfile : All  
Precedence    : 127  
JobObject     :  
NetDirectPort : 445  
PriorityValue  : 5
```

10. To configure ETS for all traffic classes defined in the previous step, issue the following commands:

```
PS C:\Users\Administrators> New-NetQosTrafficClass -name "RDMA class"  
-priority 5 -bandwidthPercentage 50 -Algorithm ETS
```

```
PS C:\Users\Administrators> New-NetQoSTrafficClass -name "TCP class" -priority 0 -bandwidthPercentage 30 -Algorithm ETS
```

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

Name	Algorithm	Bandwidth(%)	Priority	PolicySet	IfIndex	IfAlias
[Default]	ETS	20	1-4,6-7	Global		
RDMA class	ETS	50	5	Global		
TCP class	ETS	30	0	Global		

11. To see the network adapter QoS from the preceding configuration, issue the following Windows PowerShell command:

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

```
Name : SLOT 4 Port 1
Enabled : True
Capabilities :
Hardware :
Current :
MacSecBypass : NotSupported NotSupported
DcbxSupport : None None
NumTCs (Max/ETS/PFC) : 4/4/4 4/4/4

OperationalTrafficClasses : TC TSA Bandwidth Priorities
-- ---
0 ETS 20% 1-4,6-7
1 ETS 50% 5
2 ETS 30% 0

OperationalFlowControl : Priority 5 Enabled
OperationalClassifications : Protocol Port/Type Priority
-----
Default 0
NetDirect 445 5
```

12. Create a startup script to make the settings persistent across the system reboots.
13. Run RDMA traffic and verify as described in [“RoCE Configuration” on page 134](#).

Configuring QoS by Enabling DCBX on the Adapter

All configuration must be completed on all of the systems in use. The PFC, ETS, and traffic classes configuration must be the same on the switch and server.

To configure QoS by enabling DCBX:

1. Using UEFI HII, enable DCBX (IEEE, CEE, or Dynamic) on the adapter port.
2. Using UEFI HII, set the **RoCE Priority** to 0 on the adapter port.

NOTE

If the switch is not a RoCE traffic class type, set the **RoCE Priority** to the RoCE DCBX traffic class priority number used by the switch. Arista® switches can do so, but some other switches cannot.

3. To install the DCB role in the host, issue the following Windows PowerShell command:

```
PS C:\Users\Administrators> Install-WindowsFeature  
Data-Center-Bridging
```

NOTE

For this configuration, set the **DCBX Protocol** to **CEE**.

4. To set the **DCBX Willing** mode to **True**, issue the following command:

```
PS C:\Users\Administrators> set-NetQosDcbxSetting -Willing 1
```

5. Enable QoS in the miniport properties as follows:
 - a. On the adapter properties' Advanced page ([Figure 15-11](#)) under **Property**, select **Quality of Service**, and then set the value to **Enabled**.
 - b. Click **OK**.

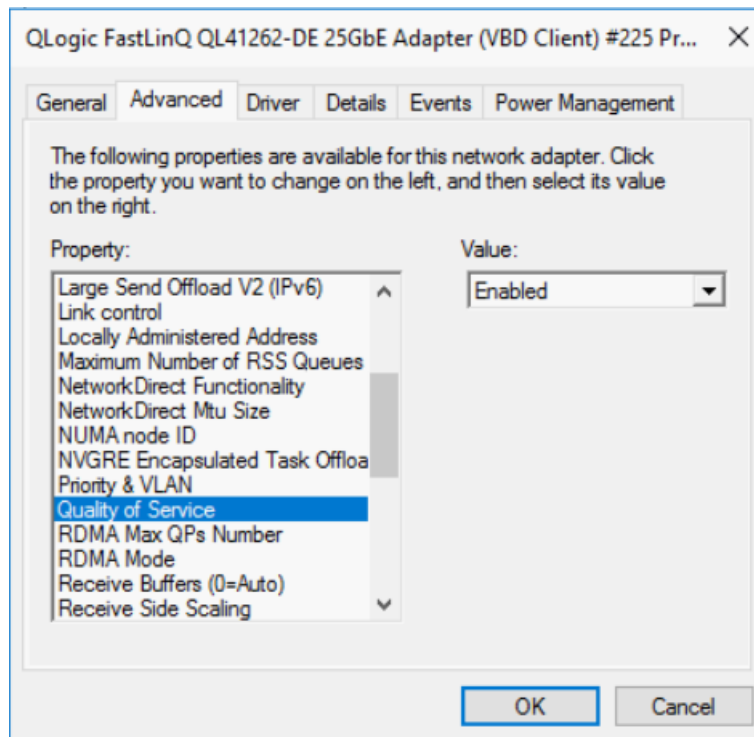


Figure 15-11. Advanced Properties: Enabling QoS

6. Assign the vLAN ID to the interface (required for PFC) as follows:
 - a. Open the miniport properties, and then click the **Advanced** tab.
 - b. On the adapter properties' Advanced page ([Figure 15-12](#)) under **Property**, select **VLAN ID**, and then set the value.
 - c. Click **OK**.

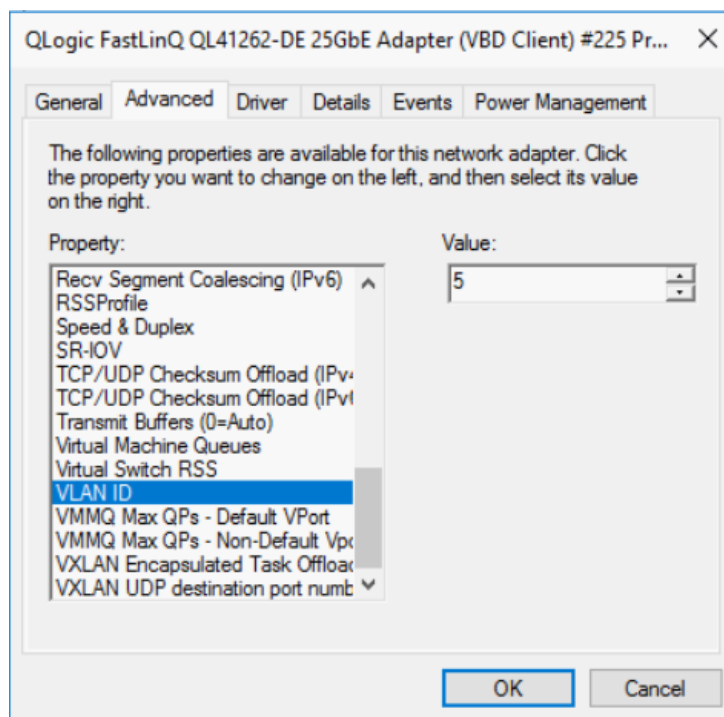


Figure 15-12. Advanced Properties: Setting VLAN ID

- To configure the switch, issue the following Windows PowerShell command:

```
PS C:\Users\Administrators> Get-NetAdapterQoS

Name                : Ethernet 5
Enabled              : True
Capabilities         :
                    Hardware      Current
                    -----      -
                    MacSecBypass  : NotSupported NotSupported
                    DcbxSupport   : CEE           CEE
                    NumTCs (Max/ETS/PFC) : 4/4/4       4/4/4

OperationalTrafficClasses : TC TSA      Bandwidth Priorities
                    --  ---      -
                    0  ETS      5%      0-4, 6-7
                    1  ETS      95%     5

OperationalFlowControl  : Priority 5 Enabled
```

```
OperationalClassifications : Protocol  Port/Type  Priority
                          -----  -
                          NetDirect  445        5

RemoteTrafficClasses      : TC  TSA      Bandwidth  Priorities
                          --  ---      -
                          0  ETS      5%        0-4, 6-7
                          1  ETS      95%       5

RemoteFlowControl         : Priority 5 Enabled
RemoteClassifications     : Protocol  Port/Type  Priority
                          -----  -
                          NetDirect  445        5
```

NOTE

The preceding example is taken when the adapter port is connected to an Arista 7060X switch. In this example, the switch PFC is enabled on Priority 5. RoCE App TLVs are defined. The two traffic classes are defined as TC0 and TC1, where TC1 is defined for RoCE. **DCBX Protocol** mode is set to **CEE**. For Arista switch configuration, refer to [“Preparing the Ethernet Switch” on page 136](#). When the adapter is in **Willing** mode, it accepts Remote Configuration and shows it as **Operational Parameters**.

Configuring VMMQ

Virtual machine multiqueue (VMMQ) configuration information includes:

- [Enabling VMMQ on the Adapter](#)
- [Creating a Virtual Machine Switch with or Without SR-IOV](#)
- [Enabling VMMQ on the Virtual Machine Switch](#)
- [Getting the Virtual Machine Switch Capability](#)
- [Creating a VM and Enabling VMMQ on VMNetworkAdapters in the VM](#)
- [Enabling and Disabling VMMQ on a Management NIC](#)
- [Monitoring Traffic Statistics](#)

Enabling VMMQ on the Adapter

To enable VMMQ on the adapter:

1. Open the miniport properties, and then click the **Advanced** tab.
2. On the adapter properties' Advanced page (Figure 15-13) under **Property**, select **Virtual Switch RSS**, and then set the value to **Enabled**.
3. Click **OK**.

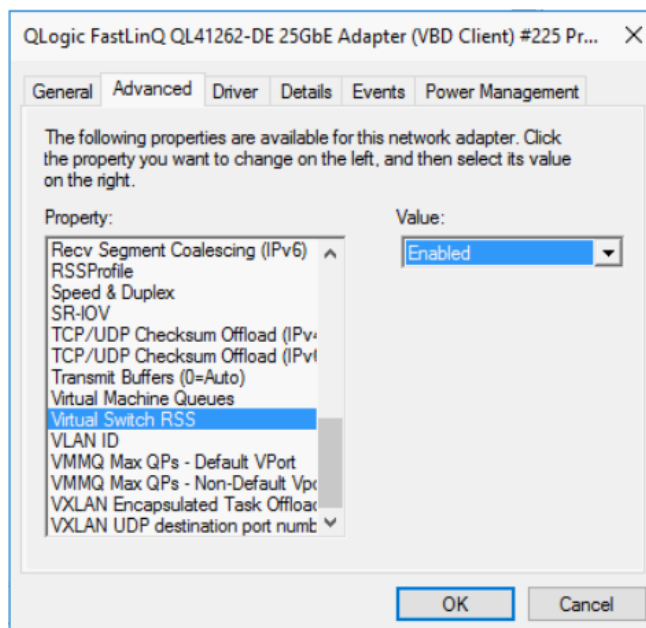


Figure 15-13. Advanced Properties: Enabling Virtual Switch RSS

Creating a Virtual Machine Switch with or Without SR-IOV

To create a virtual machine switch with or without SR-IOV:

1. Launch the Hyper-V Manager.
2. Select **Virtual Switch Manager** (see Figure 15-14).
3. In the **Name** box, type a name for the virtual switch.
4. Under **Connection type**:
 - a. Click **External network**.
 - b. Select the **Allow management operating system to share this network adapter** check box.

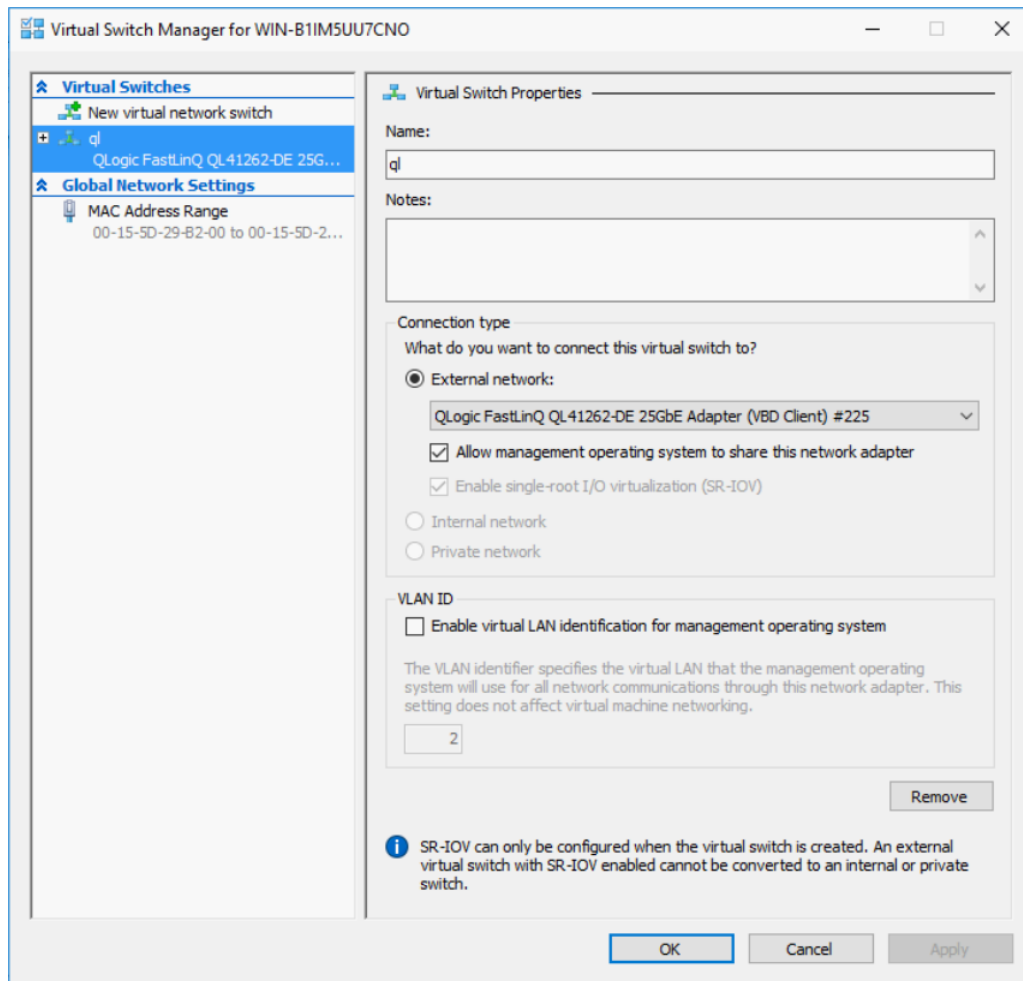


Figure 15-14. Virtual Switch Manager

5. Click **OK**.

Enabling VMMQ on the Virtual Machine Switch

To enable VMMQ on the virtual machine switch:

- Issue the following Windows PowerShell command:

```
PS C:\Users\Administrators> Set-VMSwitch -name ql  
-defaultqueuevmmqenabled $true -defaultqueuevmmqqueuepairs 4
```

Getting the Virtual Machine Switch Capability

To get the virtual machine switch capability:

- Issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Get-VMSwitch -Name ql | fl
```

Figure 15-15 shows example output.

```
PS C:\Users\Administrator> Get-VMSwitch -Name ql | fl

Name                : ql
Id                  : 4dff5da3-f8bc-4146-a809-e1ddc6a04f7a
Notes               :
Extensions          : {Microsoft Windows Filtering Platform, Microsoft Azure VFP Switch Extension,
Microsoft NDIS Capture}
BandwidthReservationMode : None
PacketDirectEnabled  : False
EmbeddedTeamingEnabled : False
IovEnabled           : True
SwitchType           : External
AllowManagementOS    : True
NetAdapterInterfaceDescription : QLogic FastLinQ QL41262-DE 25GbE Adapter (VBD Client) #225
NetAdapterInterfaceDescriptions : {QLogic FastLinQ QL41262-DE 25GbE Adapter (VBD Client) #225}
IovSupport           : True
IovSupportReasons    :
AvailableIPSecSA     : 0
NumberIPSecSAAllocated : 0
AvailableVMQueues    : 103
NumberVmqAllocated   : 1
IovQueuePairCount    : 127
IovQueuePairsInUse   : 2
IovVirtualFunctionCount : 96
IovVirtualFunctionsInUse : 0
PacketDirectInUse    : False
DefaultQueueVrssEnabledRequested : True
DefaultQueueVrssEnabled : True
DefaultQueueVmqEnabledRequested : False
DefaultQueueVmqEnabled : False
DefaultQueueVmqQueuePairsRequested : 16
DefaultQueueVmqQueuePairs : 16
BandwidthPercentage  : 0
DefaultFlowMinimumBandwidthAbsolute : 0
DefaultFlowMinimumBandwidthWeight : 0
CimSession           : CimSession: .
ComputerName         : WIN-B1IM5UU7CNO
IsDeleted            : False
```

Figure 15-15. Windows PowerShell Command: Get-VMSwitch

Creating a VM and Enabling VMMQ on VMNetworkAdapters in the VM

To create a virtual machine (VM) and enable VMMQ on VMNetworksapters in the VM:

1. Create a VM.
2. Add the VMNetworkadapter to the VM.
3. Assign a virtual switch to the VMNetworkadapter.

4. To enable VMMQ on the VM, issue the following Windows PowerShell command:

```
PS C:\Users\Administrators> set-vmnetworkadapter -vmname vm1  
-VMNetworkAdapterName "network adapter" -vmmqenabled $true  
-vmmqqueuepairs 4
```

Enabling and Disabling VMMQ on a Management NIC

To enable or disable VMMQ on a management NIC:

- To enable VMMQ on a management NIC, issue the following command:

```
PS C:\Users\Administrator> Set-VMNetworkAdapter -ManagementOS  
-vmmqEnabled $true
```

- To disable VMMQ on a management NIC, issue the following command:

```
PS C:\Users\Administrator> Set-VMNetworkAdapter -ManagementOS  
-vmmqEnabled $false
```

A VMMQ will also be available for the multicast open shortest path first (MOSPF).

Monitoring Traffic Statistics

To monitor virtual function traffic in a virtual machine, issue the following Windows PowerShell command:

```
PS C:\Users\Administrator> Get-NetAdapterStatistics | fl
```

NOTE

Marvell supports the new parameter added for Windows Server 2016 and Windows Server 2019/Azure Stack HCI to configure the maximum quantity of queue pairs on a virtual port. For details, see [“Max Queue Pairs \(L2\) Per VPort” on page 298](#).

Configuring Storage Spaces Direct

Windows Server 2016 introduces Storage Spaces Direct, which allows you to build highly available and scalable storage systems with local storage. For more information, refer to the following Microsoft TechNet link:

<https://technet.microsoft.com/en-us/windows-server-docs/storage/storage-spaces/storage-spaces-direct-windows-server-2016>

Configuring the Hardware

Figure 15-16 shows an example of hardware configuration on Windows Server 2016.

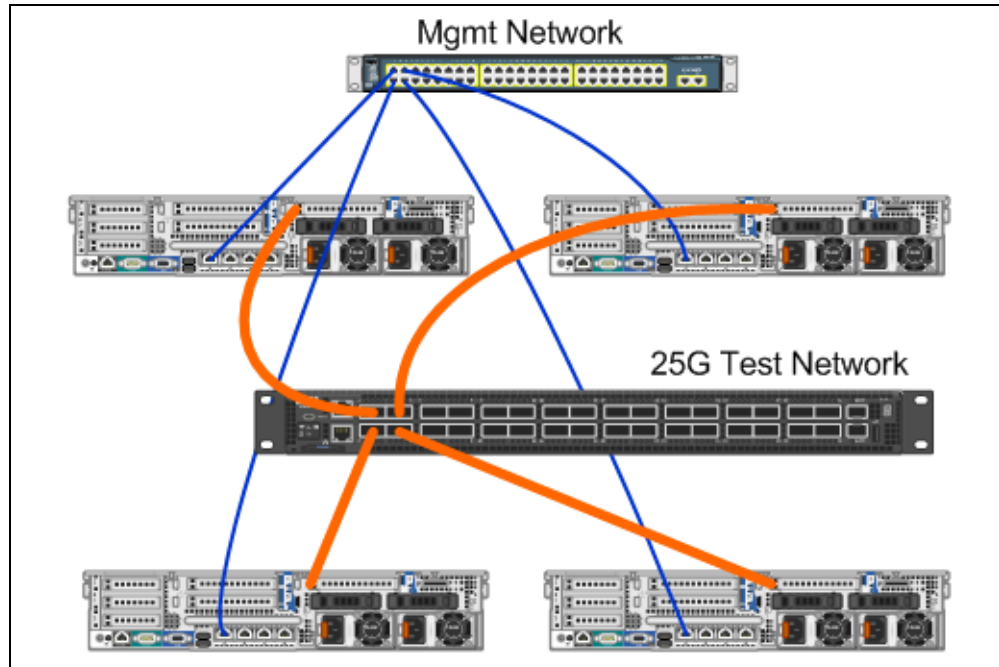


Figure 15-16. Example Hardware Configuration

NOTE

The disks used in this example are 4 × 400G NVMe™, and 12 × 200G SSD disks.

Deploying a Hyper-Converged System

This section includes instructions to install and configure the components of a Hyper-Converged system using the Windows Server 2016. The act of deploying a Hyper-Converged system can be divided into the following three high-level phases:

- [Deploying the Operating System](#)
- [Configuring the Network](#)
- [Configuring Storage Spaces Direct](#)

Deploying the Operating System

To deploy the operating systems:

1. Install the operating system.
2. Install the Windows Server roles (Hyper-V).
3. Install the following features:
 - Failover
 - Cluster
 - Data center bridging (DCB)
4. Connect the nodes to a domain and add domain accounts.

Configuring the Network

To deploy Storage Spaces Direct, the Hyper-V switch must be deployed with RDMA-enabled host virtual NICs.

NOTE

The following procedure assumes that there are four RDMA NIC ports.

To configure the network on each server:

1. Configure the physical network switch as follows:
 - a. Connect all adapter NICs to the switch port.

NOTE

If your test adapter has more than one NIC port, you must connect both ports to the same switch.

- b. Enable the switch port and make sure that:
 - The switch port supports switch-independent teaming mode.
 - The switch port is part of multiple vLAN networks.

Example Dell switch configuration:

```
no ip address
mtu 9416
portmode hybrid
switchport
dcb-map roce_S2D
protocol lldp
dcbx version cee
no shutdown
```


2. Enable **Network Quality of Service**.

NOTE

Network Quality of Service is used to ensure that the Software Defined Storage system has enough bandwidth to communicate between the nodes to ensure resiliency and performance. To configure QoS on the adapter, see [“Configuring QoS for RoCE” on page 276](#).

3. Create a Hyper-V virtual switch with Switch Embedded Teaming (SET) and RDMA virtual NIC as follows:

- a. To identify the network adapters, issue the following command:

```
Get-NetAdapter | FT  
Name, InterfaceDescription, Status, LinkSpeed
```

- b. To create a virtual switch connected to all of the physical network adapters, and to then enable SET, issue the following command:

```
New-VMSwitch -Name SETswitch -NetAdapterName  
"<port1>","<port2>","<port3>","<port4>"  
-EnableEmbeddedTeaming $true
```

- c. To add host virtual NICs to the virtual switch, issue the following commands:

```
Add-VMNetworkAdapter -SwitchName SETswitch -Name SMB_1  
-managementOS  
Add-VMNetworkAdapter -SwitchName SETswitch -Name SMB_2  
-managementOS
```

NOTE

The preceding commands configure the virtual NIC from the virtual switch that you just configured for the management operating system to use.

- d. To configure the host virtual NIC to use a vLAN, issue the following commands:

```
Set-VMNetworkAdapterVlan -VMNetworkAdapterName "SMB_1"  
-VlanId 5 -Access -ManagementOS  
Set-VMNetworkAdapterVlan -VMNetworkAdapterName "SMB_2"  
-VlanId 5 -Access -ManagementOS
```

NOTE

These commands can be on the same or different vLANs.

- e. To verify that the vLAN ID is set, issue the following command:

```
Get-VMNetworkAdapterVlan -ManagementOS
```
- f. To disable and enable each host virtual NIC adapter so that the vLAN is active, issue the following commands:

```
Disable-NetAdapter "vEthernet (SMB_1)"  
Enable-NetAdapter "vEthernet (SMB_1)"  
Disable-NetAdapter "vEthernet (SMB_2)"  
Enable-NetAdapter "vEthernet (SMB_2)"
```
- g. To enable RDMA on the host virtual NIC adapters, issue the following command:

```
Enable-NetAdapterRdma "SMB1", "SMB2"
```
- h. To verify RDMA capabilities, issue the following command:

```
Get-SmbClientNetworkInterface | where RdmaCapable -EQ  
$true
```

Configuring Storage Spaces Direct

Configuring Storage Spaces Direct in Windows Server 2016 includes the following steps:

- [Step 1. Running a Cluster Validation Tool](#)
- [Step 2. Creating a Cluster](#)
- [Step 3. Configuring a Cluster Witness](#)
- [Step 4. Cleaning Disks Used for Storage Spaces Direct](#)
- [Step 5. Enabling Storage Spaces Direct](#)
- [Step 6. Creating Virtual Disks](#)
- [Step 7. Creating or Deploying Virtual Machines](#)

Step 1. Running a Cluster Validation Tool

Run the cluster validation tool to make sure server nodes are configured correctly to create a cluster using Storage Spaces Direct.

To validate a set of servers for use as a Storage Spaces Direct cluster, issue the following Windows PowerShell command:

```
Test-Cluster -Node <MachineName1, MachineName2, MachineName3,  
MachineName4> -Include "Storage Spaces Direct", Inventory,  
Network, "System Configuration"
```

Step 2. Creating a Cluster

Create a cluster with the four nodes (which was validated for cluster creation) in [Step 1. Running a Cluster Validation Tool](#).

To create a cluster, issue the following Windows PowerShell command:

```
New-Cluster -Name <ClusterName> -Node <MachineName1, MachineName2, MachineName3, MachineName4> -NoStorage
```

The `-NoStorage` parameter is required. If it is not included, the disks are automatically added to the cluster, and you must remove them before enabling Storage Spaces Direct. Otherwise, they will not be included in the Storage Spaces Direct storage pool.

Step 3. Configuring a Cluster Witness

You should configure a witness for the cluster, so that this four-node system can withstand two nodes failing or being offline. With these systems, you can configure file share witness or cloud witness.

For more information, go to:

<https://docs.microsoft.com/en-us/windows-server/failover-clustering/manage-cluster-quorum>

Step 4. Cleaning Disks Used for Storage Spaces Direct

The disks intended to be used for Storage Spaces Direct must be empty and without partitions or other data. If a disk has partitions or other data, it will not be included in the Storage Spaces Direct system.

The following Windows PowerShell command can be placed in a Windows PowerShell script (.PS1) file and executed from the management system in an open Windows PowerShell (or Windows PowerShell ISE) console with Administrator privileges.

NOTE

Running this script helps identify the disks on each node that can be used for Storage Spaces Direct. It also removes all data and partitions from those disks.

```
icm (Get-Cluster -Name HCNanoUSClu3 | Get-ClusterNode) {  
Update-StorageProviderCache  
  
Get-StoragePool |? IsPrimordial -eq $false | Set-StoragePool  
-IsReadOnly:$false -ErrorAction SilentlyContinue  
  
Get-StoragePool |? IsPrimordial -eq $false | Get-VirtualDisk |  
Remove-VirtualDisk -Confirm:$false -ErrorAction SilentlyContinue
```

```
Get-StoragePool |? IsPrimordial -eq $false | Remove-StoragePool
-Confirm:$false -ErrorAction SilentlyContinue

Get-PhysicalDisk | Reset-PhysicalDisk -ErrorAction
SilentlyContinue

Get-Disk |? Number -ne $null |? IsBoot -ne $true |? IsSystem -ne
$true |? PartitionStyle -ne RAW |% {
$_ | Set-Disk -isoffline:$false
$_ | Set-Disk -isreadonly:$false
$_ | Clear-Disk -RemoveData -RemoveOEM -Confirm:$false
$_ | Set-Disk -isreadonly:$true
$_ | Set-Disk -isoffline:$true
}

Get-Disk |? Number -ne $null |? IsBoot -ne $true |? IsSystem -ne
$true |? PartitionStyle -eq RAW | Group -NoElement -Property
FriendlyName

} | Sort -Property PsComputerName,Count
```

Step 5. Enabling Storage Spaces Direct

After creating the cluster, issue the `Enable-ClusterS2D` Windows PowerShell cmdlet. The cmdlet places the storage system into the Storage Spaces Direct mode and automatically does the following:

- Creates a single, large pool that has a name such as *S2D on Cluster1*.
- Configures a Storage Spaces Direct cache. If there is more than one media type available for Storage Spaces Direct use, it configures the most efficient type as cache devices (in most cases, read and write).
- Creates two tiers—**Capacity** and **Performance**—as default tiers. The cmdlet analyzes the devices and configures each tier with the mix of device types and resiliency.

Step 6. Creating Virtual Disks

If the Storage Spaces Direct was enabled, it creates a single pool using all of the disks. It also names the pool (for example *S2D on Cluster1*), with the name of the cluster that is specified in the name.

The following Windows PowerShell command creates a virtual disk with both mirror and parity resiliency on the storage pool:

```
New-Volume -StoragePoolFriendlyName "S2D*" -FriendlyName
<VirtualDiskName> -FileSystem CSVFS_ReFS -StorageTierfriendlyNames
Capacity,Performance -StorageTierSizes <Size of capacity tier in
size units, example: 800GB>, <Size of Performance tier in size
units, example: 80GB> -CimSession <ClusterName>
```

Step 7. Creating or Deploying Virtual Machines

You can provision the virtual machines onto the nodes of the hyper-converged S2D cluster. Store the virtual machine's files on the system's Cluster Shared Volume (CSV) namespace (for example, `c:\ClusterStorage\Volume1`), similar to clustered virtual machines on failover clusters.

16 Windows Server 2019/ Azure Stack HCI

This chapter provides the following information for Windows Server 2019/Azure Stack HCI:

- [RSSv2 for Hyper-V](#)
- [“Windows Server 2019/Azure Stack HCI Behaviors” on page 297](#)
- [“New Adapter Properties” on page 298](#)

RSSv2 for Hyper-V

In Windows Server 2019/Azure Stack HCI, Microsoft added support for Receive Side Scaling version 2 (RSSv2) with Hyper-V (RSSv2 per vPort).

RSSv2 Description

Compared to RSSv1, RSSv2 decreases the time between the CPU load measurement and the indirection table update. This feature prevents slowdown during high-traffic situations. RSSv2 can dynamically spread receive queues over multiple processors much more responsively than RSSv1. For more information, visit the following Web page:

<https://docs.microsoft.com/en-us/windows-hardware/drivers/network/receive-side-scaling-version-2-rssv2->

RSSv2 is supported by default in the Windows Server 2019/Azure Stack HCI driver when the **Virtual Switch RSS** option is also enabled. This option is enabled (the default), and the NIC is bound to the Hyper-V or vSwitch.

Known Event Log Errors

Under typical operation, the dynamic algorithm of RSSv2 may initiate an indirection table update that is incompatible with the driver and return an appropriate status code. In such cases, an event log error occurs, even though no functional operation issue exists. [Figure 16-1](#) shows an example.

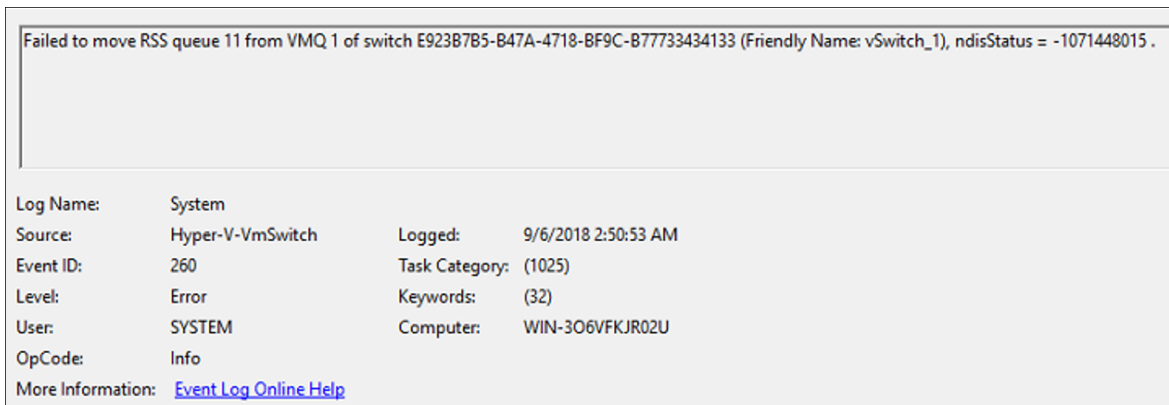


Figure 16-1. RSSv2 Event Log Error

Windows Server 2019/Azure Stack HCI Behaviors

Windows Server 2019/Azure Stack HCI introduced the following new behaviors affecting adapter configuration.

VMMQ Is Enabled by Default

In the inbox driver of Windows Server 2019/Azure Stack HCI, the **Virtual Switch RSS** (VMMQ) option is enabled by default in the NIC properties. In addition, Microsoft changed the default behavior of the **Virtual NICs** option to have VMMQ enabled with the 16 queue pairs. This behavior change impacts the quantity of available resources.

For example, suppose the NIC supports 32 VMQs and 64 queue pairs. In Windows Server 2016, when you add 32 virtual NICs (vNICs), they will have VMQ acceleration. However in Windows Server 2019/Azure Stack HCI, you will get 4 vNICs with VMMQ acceleration, each with 16 queue pairs and 30 vNICs with no acceleration.

Because of this functionality, Marvell introduced a new user property, **Max Queue Pairs (L2) Per VPort**. For more details, see [New Adapter Properties](#).

Inbox Driver Network Direct (RDMA) Is Disabled by Default

In the inbox driver of Windows Server 2019/Azure Stack HCI, the **Network Direct** (RDMA) option is disabled by default in the NIC properties. However, when upgrading the driver to an out-of-box driver, **Network Direct** is enabled by default.

New Adapter Properties

New user-configurable properties available in Windows Server 2019/Azure Stack HCI are described in the following sections:

- [Max Queue Pairs \(L2\) Per VPort](#)
- [Network Direct Technology](#)
- [Virtualization Resources](#)
- [VMQ and VMMQ Default Accelerations](#)
- [Single VPort Pool](#)

Max Queue Pairs (L2) Per VPort

As explained in [VMMQ Is Enabled by Default](#), Windows 2019/Azure Stack HCI (and Windows 2016) introduced a new user-configurable parameter, **Max Queue Pairs (L2) per VPort**. This parameter permits greater control of resource distribution by defining the maximum quantity of queue pairs that can be assigned to the following:

- VPort-Default VPort
- PF Non-Default VPort (VMQ/VMMQ)
- SR-IOV Non-Default VPort (VF)¹

The default value of the **Max Queue Pairs (L2) per VPort** parameter is set to **Auto**, which is one of the following:

- Max Queue Pairs for Default vPort = 8
- Max Queue Pairs for Non-Default vPort = 4

If you select a value less than 8, then:

- Max Queue Pairs for Default vPort = 8
- Max Queue Pairs for Non-Default vPort = value

If you select a value greater than 8, then:

- Max Queue Pairs for Default vPort = value
- Max Queue Pairs for Non-Default vPort = value

¹ This parameter also applies to Windows Server 2016.

Network Direct Technology

Marvell supports the new **Network Direct Technology** parameter that allows you to select the underlying RDMA technology that adheres to the following Microsoft specification:

<https://docs.microsoft.com/en-us/windows-hardware/drivers/network/inf-requirements-for-ndkpi>

This option replaces the **RDMA Mode** parameter.

Virtualization Resources

[Table 16-1](#) lists the maximum quantities of virtualization resources in Windows 2019/Azure Stack HCI for 41000 Series Adapters.

Table 16-1. Windows 2019/Azure Stack HCI Virtualization Resources for 41000 Series Adapters

Two-port NIC-only Single Function Non-CNA	Quantity
Maximum VMQs	102
Maximum VFs	80
Maximum QPs	112
Four-port NIC-only Single Function Non-CNA	Quantity
Maximum VMQs	47
Maximum VFs	32
Maximum QPs	48

VMQ and VMMQ Default Accelerations

Table 16-2 lists the VMQ and VMMQ default and other values for accelerations in Windows Server 2019/Azure Stack HCI for 41000 Series Adapters.

Table 16-2. Windows 2019/Azure Stack HCI VMQ and VMMQ Accelerations

Two-port NIC-only Single Function Non-CNA	Default Value	Other Possible Values				
Maximum Queue Pairs (L2) per VPort ^a	Auto	1	2	4	8	16
Maximum VMQs	26	103	52	26	13	6
Default VPort Queue Pairs	8	8	8	8	8	16
PF Non-default VPort Queue Pairs	4	1	2	4	8	16
Four-port NIC-only Single Function Non-CNA	Default Value	Other Possible Values				
Maximum Queue Pairs (L2) per VPort ^a	Auto	1	2	4	8	16
Maximum VMQs	10	40	20	10	5	2
Default VPort Queue Pairs	8	8	8	8	8	16
PF Non-default VPort Queue Pairs	4	1	2	4	8	16

^a Max Queue Pairs (L2) VPort is configurable parameter of NIC advanced properties.

Single VPort Pool

The 41000 Series Adapter supports the **Single VPort Pool** parameter, which allows the system administrator to assign any available IOVQueuePair to either Default-VPort, PF Non-Default VPort, or VF Non-Default VPort. To assign the value, issue the following Windows PowerShell commands:

- Default-VPort:

```
Set-VMSwitch -Name <vswitch name> -DefaultQueueVmmqEnabled:1  
-DefaultQueueVmmqQueuePairs:<number>
```

NOTE

Marvell does not recommend that you disable VMMQ or decrease the quantity of queue pairs for the Default-VPort, because it may impact system performance.

■ PF Non-Default VPort:

For the host:

```
Set-VMNetworkAdapter -ManagementOS -VmmqEnabled:1  
-VmmqQueuePairs:<number>
```

For the VM:

```
Set-VMNetworkAdapter -VMName <vm name> -VmmqEnabled:1  
-VmmqQueuePairs:<number>
```

■ VF Non-Default VPort:

```
Set-VMNetworkAdapter -VMName <vm name> -IovWeight:100  
-IovQueuePairsRequested:<number>
```

NOTE

The default quantity of QPs assigned for a VF (**IovQueuePairsRequested**) is still 1.

To apply multiple quantities of queue pairs to any vPort:

- The quantity of queue pairs must be less than or equal to the total number of CPU cores on the system.
- The quantity of queue pairs must be less than or equal to the value of **Max Queue Pairs (L2) Per VPort**. For more information, see [Max Queue Pairs \(L2\) Per VPort](#).

17 Troubleshooting

This chapter provides the following troubleshooting information:

- [Troubleshooting Checklist](#)
- [“Verifying that Current Drivers Are Loaded” on page 303](#)
- [“Testing Network Connectivity” on page 304](#)
- [“Atypical FCoE Configurations” on page 305](#)
- [“Linux-specific Issues” on page 308](#)
- [“Miscellaneous Issues” on page 309](#)
- [“Collecting Debug Data” on page 309](#)

Troubleshooting Checklist

CAUTION

Before you open the server cabinet to add or remove the adapter, review the [“Safety Precautions” on page 6](#).

The following checklist provides recommended actions to resolve problems that may arise while installing the 41000 Series Adapter or running it in your system.

- Inspect all cables and connections. Verify that the cable connections at the network adapter and the switch are attached properly.
- Verify the adapter installation by reviewing [“Installing the Adapter” on page 7](#). Ensure that the adapter is properly seated in the slot. Check for specific hardware problems, such as obvious damage to board components or the PCI edge connector.
- Verify the configuration settings and change them if they are in conflict with another device.
- Verify that your server is using the latest BIOS.
- Try inserting the adapter in another slot. If the new position works, the original slot in your system may be defective.

- Replace the failed adapter with one that is known to work properly. If the second adapter works in the slot where the first one failed, the original adapter is probably defective.
- Install the adapter in another functioning system, and then run the tests again. If the adapter passes the tests in the new system, the original system may be defective.
- Remove all other adapters from the system, and then run the tests again. If the adapter passes the tests, the other adapters may be causing contention.

Verifying that Current Drivers Are Loaded

Ensure that the current drivers are loaded for your Windows, Linux, or VMware system.

Verifying Drivers in Windows

See the Device Manager to view vital information about the adapter, link status, and network connectivity.

Verifying Drivers in Linux

To verify that the qed.ko driver is loaded properly, issue the following command:

```
# lsmod | grep -i <module name>
```

If the driver is loaded, the output of this command shows the size of the driver in bytes. The following example shows the drivers loaded for the qed module:

```
# lsmod | grep -i qed
qed                199238  1
qede               1417947  0
```

If you reboot after loading a new driver, you can issue the following command to verify that the currently loaded driver is the correct version:

```
modinfo qede
```

Or, you can issue the following command:

```
[root@test1]# ethtool -i eth2
driver: qede
version: 8.4.7.0
firmware-version: mfw 8.4.7.0 storm 8.4.7.0
bus-info: 0000:04:00.2
```

If you loaded a new driver, but have not yet rebooted, the `modinfo` command will not show the updated driver information. Instead, issue the following `dmesg` command to view the logs. In this example, the last entry identifies the driver that will be active upon reboot.

```
# dmesg | grep -i "QLogic" | grep -i "qede"

[ 10.097526] QLogic FastLinQ 4xxxx Ethernet Driver qede x.x.x.x
[ 23.093526] QLogic FastLinQ 4xxxx Ethernet Driver qede x.x.x.x
[ 34.975396] QLogic FastLinQ 4xxxx Ethernet Driver qede x.x.x.x
[ 34.975896] QLogic FastLinQ 4xxxx Ethernet Driver qede x.x.x.x
[ 3334.975896] QLogic FastLinQ 4xxxx Ethernet Driver qede x.x.x.x
```

Verifying Drivers in VMware

To verify that the VMware ESXi drivers are loaded, issue the following command:

```
# esxcli software vib list
```

Testing Network Connectivity

This section provides procedures for testing network connectivity in Windows and Linux environments.

NOTE

When using forced link speeds, verify that both the adapter and the switch are forced to the same speed.

Testing Network Connectivity for Windows

Test network connectivity using the `ping` command.

To determine if the network connection is working:

1. Click **Start**, and then click **Run**.
2. In the **Open** box, type `cmd`, and then click **OK**.
3. To view the network connection to be tested, issue the following command:

```
ipconfig /all
```

4. Issue the following command, and then press ENTER.

```
ping <ip_address>
```

The displayed ping statistics indicate whether or not the network connection is working.

Testing Network Connectivity for Linux

To verify that the Ethernet interface is up and running:

1. To check the status of the Ethernet interface, issue the `ifconfig` command.
2. To check the statistics on the Ethernet interface, issue the `netstat -i` command.

To verify that the connection has been established:

1. Ping an IP host on the network. From the command line, issue the following command:

```
ping <ip_address>
```

2. Press ENTER.

The displayed ping statistics indicate whether or not the network connection is working.

The adapter link speed can be forced to 10Gbps or 25Gbps using either the operating system GUI tool or the `ethtool` command, `ethtool -s ethX speed SSSS`.

Atypical FCoE Configurations

Marvell FCoE solutions using 41000 Series Adapters are optimized to operate in an environment where a Converged Network Adapter (CNA) port is physically connected to a switch port with the following characteristics:

- DCBX is converged for lossless traffic for FCoE protocol (priority and priority-based flow control (PFC) enabled).
- A single VLAN is configured for FCoE traffic.
- A single FCoE forwarder (FCF) is available for login.
- There is a single FCoE/Fibre Channel fabric.

Most solutions in which Marvell FCoE products are deployed conform to these characteristics. However, there are system configurations that do not.

The following sections describe some of these atypical configurations and how to troubleshoot possible issues.

Atypical Configurations

The following sections describe possible issues on these nonstandard configurations:

- [Multiple FCoE VLANs](#)
- [Fallback FCoE VLAN](#)
- [Multiple FCFs in a Single Fabric](#)
- [Multiple Fabric \(With One or More FCFs\)](#)

Multiple FCoE VLANs

FCoE drivers retrieve the VLAN over which FCoE traffic flows, using the FCoE initialization protocol (FIP) VLAN discovery protocol defined in the Fibre Channel-Backbone (FC-BB) specification. In configurations where multiple VLANs are configured for FCoE, there may be more than one entity responding to the FIP VLAN request from the initiator and reporting multiple and/or different VLANs.

Since the drivers cannot validate the VLAN and the corresponding FCoE fabric for storage connectivity in advance, having multiple VLAN responses can result in unpredictable behaviors.

Fallback FCoE VLAN

FCoE drivers try to find the VLAN over which to operate FCoE by following the FIP VLAN discovery protocol outlined in the FC-BB specifications.

When FIP VLAN discovery does not succeed, the drivers attempt to discover the FCF over a fallback VLAN.

Multiple FCFs in a Single Fabric

FCoE drivers identify available FCFs for login using the FIP discovery protocol defined in FC-BB specification. In configurations where multiple FCFs are configured for FCoE, there may be more than one FCF (with a unique MAC address) that responds to the FIP discovery solicitation. In this case, the driver chooses a single FCF based on the available list of FCFs.

Multiple Fabric (With One or More FCFs)

An FCoE fabric is identified by a unique fabric WWPN, virtual fabric ID, and Fibre Channel map. In a multi-fabric environment, there are multiple FCFs connected to the CNA port, each with a different fabric WWPN/virtual fabric ID/Fibre Channel map.

Having multiple fabrics exposed to a CNA port is considered an invalid configuration, because the driver cannot validate the exact same storage connectivity across the multiple fabrics. In the event of an FCF failure, the driver can choose the alternate fabric that may not have the same storage connectivity as the previous fabric, leading to I/O failures.

Troubleshooting Atypical FCoE Configurations

The following sections contain troubleshooting steps based on OS:

- [Windows](#)
- [Linux](#)
- [VMware ESX \(Driver v2.x.y.z and Later\)](#)
- [VMware ESX \(Driver v1.x.y.z and Later\)](#)

Windows

- Problem:** FCoE devices are not discovered in multiple VLAN or FCF configurations.
- Solution:** Ensure consistent storage connectivity across all VLANs configured for FCoE and/or across all FCFs.
Alternately, redesign the system to have only one VLAN and/or FCF in the configuration.
- Problem:** There is excessive event logging (event ID 26) in multi-fabric configurations.
- Solution:** Verify that there is consistent storage connectivity across multiple FCFs on multiple fabrics.
If the configuration is valid, update to FCoE driver v8.53.1.0 (or later) to limit the event logging.

Linux

- Problem:** FCoE devices are not discovered after the link is disrupted on the initiator port.
- Solution:** Update to FCoE driver v8.42.10.1 or later.
- Problem:** FCoE devices are not discovered in a multiple VLAN configuration.
- Solution:** Multiple VLAN configurations are not supported on the Linux OS. Reconfigure the system to have only one FCoE VLAN.
- Problem:** FCoE devices are not discovered in multiple fabric configurations.
- Solution:** Multiple fabrics are not supported on the Linux OS. Reconfigure the system to have only one FCoE fabric.

VMware ESX (Driver v2.x.y.z and Later)

- Problem:** FCoE devices are not discovered in multiple VLAN configurations or multiple FCF configurations.
- Solution:** Ensure that there is consistent storage connectivity across all VLANs configured for FCoE and/or across all FCFs.
Alternately, redesign the system to have only one VLAN and/or FCF per configuration.

Problem: A server PSOD appears in a multiple VLAN configuration.

Solution: Update the FCoE driver to v2.2.6.0 (or later).

Problem: There is excessive message logging (in `vmkernel.log`) in multiple fabric configurations.

Solution: Verify that there is consistent storage connectivity across multiple FCFs on multiple fabrics.

If the configuration is valid, update to FCoE driver v2.2.6.0 (or later) to limit the message logging.

VMware ESX (Driver v1.x.y.z and Later)

Problem: FCoE devices are not discovered after a long duration (greater than 1 hour) of link disruption on the initiator port.

Solution: Set the driver module parameter `qedf_fipvlan_retries` to a larger value (default=1800) by issuing the following command, where `x` is the requested value (maximum=FFFFFFFFh):

```
esxcfg-module -s 'qedf_fipvlan_retries=X' qedf
```

Reboot the server for the change to take effect.

Problem: FCoE devices are not discovered in a multiple FCF configuration.

Solution: Update to driver v1.3.42.0.

Problem: FCoE devices are not discovered in multiple fabric configurations.

Solution: Multiple fabric configurations are not supported on ESX platforms with driver v1.x.y.z. To limit the use of a single fabric, make one of the following changes:

Reconfigure the system to have only one FCoE fabric.

Set the driver module parameter `qlibfcoe_max_fcf` to 1 (to limit the number of fabrics supported to 1) by issuing the following command:

```
esxcfg-module -s qlibfcoe_max_fcf=1 qedf
```

Reboot the server for the change to take effect.

Linux-specific Issues

Problem: Errors appear when compiling driver source code.

Solution: Some installations of Linux distributions do not install the development tools and kernel sources by default. Before compiling driver source code, ensure that the development tools for the Linux distribution that you are using are installed.

Miscellaneous Issues

- Problem:** The 41000 Series Adapter has shut down, and an error message appears indicating that the fan on fan-equipped adapters has failed.
- Solution:** The 41000 Series Adapter may intentionally shut down to prevent permanent damage. Contact Marvell Technical Support for assistance.
- Problem:** In an ESXi environment, with the iSCSI driver (qedil) installed, sometimes, the VI-client cannot access the host. This is due to the termination of the hostd daemon, which affects connectivity with the VI-client.
- Solution:** Contact VMware technical support.

Collecting Debug Data

Use the commands in [Table 17-1](#) to collect debug data.

Table 17-1. Collecting Debug Data Commands

Debug Data	Description
<code>dmesg -T</code>	Kernel logs
<code>ethtool -d</code>	Register dump
<code>sys_info.sh</code>	System information; available in the driver bundle

A Adapter LEDs

Table A-1 lists the LED indicators for the state of the adapter port link and activity.

Table A-1. Adapter Port Link and Activity LEDs

Port LED	LED Appearance	Network State
Link LED	Off	No link (cable disconnected or port down)
	Continuously illuminated GREEN	Link at highest supported link speed
	Continuously illuminated AMBER	Link at lower supported link speed
Activity LED	Off	No port activity
	Blinking	Port activity

B Cables and Optical Modules

This appendix provides the following information for the supported cables and optical modules:

- [Supported Specifications](#)
- [“Tested Cables and Optical Modules” on page 312](#)
- [“Tested Switches” on page 317](#)

Supported Specifications

The 41000 Series Adapters support a variety of cables and optical modules that comply with SFF8024. Specific form factor compliance is as follows:

- SFPs:
 - ❑ SFF8472 (for memory map)
 - ❑ SFF8419 or SFF8431 (low speed signals and power)
- Optical modules electrical input/output, active copper cables (ACC), and active optical cables (AOC):
 - ❑ 10G—SFF8431 limiting interface
 - ❑ 25G—IEEE 802.3by Annex 109B (25GAUI) (does not support RS-FEC)

Tested Cables and Optical Modules

Marvell does not guarantee that every cable or optical module that satisfies the compliance requirements will operate with the 41000 Series Adapters. Marvell has tested the components listed in [Table B-1](#) and presents this list for your convenience.

Table B-1. Tested Cables and Optical Modules

Speed/Form Factor	Manufacturer	Part Number	Type	Cable Length ^a	Gauge
Cables					
10G DAC ^b	Brocade®	1539W	SFP+10G-to-SFP+10G	1	26
		V239T	SFP+10G-to-SFP+10G	3	26
		48V40	SFP+10G-to-SFP+10G	5	26
	Cisco	H606N	SFP+10G-to-SFP+10G	1	26
		K591N	SFP+10G-to-SFP+10G	3	26
		G849N	SFP+10G-to-SFP+10G	5	26
	Dell	V250M	SFP+10G-to-SFP+10G	1	26
		53HVN	SFP+10G-to-SFP+10G	3	26
		358VV	SFP+10G-to-SFP+10G	5	26
		407-BBBK	SFP+10G-to-SFP+10G	1	30
		407-BBBI	SFP+10G-to-SFP+10G	3	26
407-BBBP		SFP+10G-to-SFP+10G	5	26	
25G DAC	Amphenol®	NDCCGF0001	SFP28-25G-to-SFP28-25G	1	30
		NDCCGF0003	SFP28-25G-to-SFP28-25G	3	30
		NDCCGJ0003	SFP28-25G-to-SFP28-25G	3	26
		NDCCGJ0005	SFP28-25G-to-SFP28-25G	5	26
	Dell	2JVDD	SFP28-25G-to-SFP28-25G	1	26
		D0R73	SFP28-25G-to-SFP28-25G	2	26
		OVXFJY	SFP28-25G-to-SFP28-25G	3	26
		9X8JP	SFP28-25G-to-SFP28-25G	5	26

Table B-1. Tested Cables and Optical Modules (Continued)

Speed/Form Factor	Manufacturer	Part Number	Type	Cable Length ^a	Gauge
40G Copper QSFP Splitter (4 × 10G)	Dell	TCPM2	QSFP+40G-to-4xSFP+10G	1	30
		27GG5	QSFP+40G-to-4xSFP+10G	3	30
		P8T4W	QSFP+40G-to-4xSFP+10G	5	26
1G Copper RJ45 Transceiver	Dell	8T47V	SFP+ to 1G RJ	1G RJ45	N/A
		XK1M7	SFP+ to 1G RJ	1G RJ45	N/A
		XTY28	SFP+ to 1G RJ	1G RJ45	N/A
10G Copper RJ45 Transceiver	Dell	PGYJT	SFP+ to 10G RJ	10G RJ45	N/A
40G DAC Splitter (4 × 10G)	Dell	470-AAVO	QSFP+40G-to-4xSFP+10G	1	26
		470-AAWG	QSFP+40G-to-4xSFP+10G	3	26
		470-AAWH	QSFP+40G-to-4xSFP+10G	5	26
100G DAC Splitter (4 × 25G)	Amphenol	NDAQGJ-0001	QSFP28-100G-to-4xSFP28-25G	1	26
		NDAQGF-0002	QSFP28-100G-to-4xSFP28-25G	2	30
		NDAQGF-0003	QSFP28-100G-to-4xSFP28-25G	3	30
		NDAQGJ-0005	QSFP28-100G-to-4xSFP28-25G	5	26
	Dell	026FN3 Rev A00	QSFP28-100G-to-4XSFP28-25G	1	26
		0YFNDD Rev A00	QSFP28-100G-to-4XSFP28-25G	2	26
		07R9N9 Rev A00	QSFP28-100G-to-4XSFP28-25G	3	26
	FCI	10130795-4050LF	QSFP28-100G-to-4XSFP28-25G	5	26

Table B-1. Tested Cables and Optical Modules (Continued)

Speed/Form Factor	Manufacturer	Part Number	Type	Cable Length ^a	Gauge
Optical Solutions					
10G Optical Transceiver	Avago®	AFBR-703SMZ	SFP+ SR	N/A	N/A
		AFBR-701SDZ	SFP+ LR	N/A	N/A
	Dell	Y3KJN	SFP+ SR	1G/10G	N/A
		WTRD1	SFP+ SR	10G	N/A
		3G84K	SFP+ SR	10G	N/A
		RN84N	SFP+ SR	10G-LR	N/A
	Finisar®	FTLX8571D3BCL-QL	SFP+ SR	N/A	N/A
		FTLX1471D3BCL-QL	SFP+ LR	N/A	N/A
25G Optical Transceiver	Dell	P7D7R	SFP28 Optical Transceiver SR	25G SR	N/A
	Finisar	FTLF8536P4BCL	SFP28 Optical Transceiver SR	N/A	N/A
		FTLF8538P4BCL	SFP28 Optical Transceiver SR no FEC	N/A	N/A
10/25G Dual Rate Transceiver	Dell	M14MK	SFP28	N/A	N/A

Table B-1. Tested Cables and Optical Modules (Continued)

Speed/Form Factor	Manufacturer	Part Number	Type	Cable Length ^a	Gauge
10G AOC ^c	Dell	470-ABLV	SFP+ AOC	2	N/A
		470-ABLZ	SFP+ AOC	3	N/A
		470-ABLT	SFP+ AOC	5	N/A
		470-ABML	SFP+ AOC	7	N/A
		470-ABLU	SFP+ AOC	10	N/A
		470-ABMD	SFP+ AOC	15	N/A
		470-ABMJ	SFP+ AOC	20	N/A
		YJF03	SFP+ AOC	2	N/A
		P9GND	SFP+ AOC	3	N/A
		T1KCN	SFP+ AOC	5	N/A
		1DXKP	SFP+ AOC	7	N/A
		MT7R2	SFP+ AOC	10	N/A
		K0T7R	SFP+ AOC	15	N/A
		W5G04	SFP+ AOC	20	N/A
25G AOC	Dell	X5DH4	SFP28 AOC	20	N/A
	InnoLight®	TF-PY003-N00	SFP28 AOC	3	N/A
		TF-PY020-N00	SFP28 AOC	20	N/A

^a Cable length is indicated in meters.

^b DAC is direct attach cable.

^c AOC is active optical cable.

Known Issue: Using SmartAN Mode with Invalid FEC Configuration for 25G DAC

The following paragraphs describe known issues using a CA-25G-L cable with SmartAN.

Problem: SmartAN does not allow a 25G link on CA-25G-L cable without RS-FEC enabled on the link partner with Marvell firmware versions 15.x and later.

Marvell's SmartAN feature attempts to bring up link with the link partner when standard auto-negotiation is not available or configured. When the installed cable does not support the link partner's configured FEC mode as listed in the IEEE Std 802.3by-2016 specification, SmartAN does not allow the link to come up.

Marvell firmware contained in the Dell Firmware DUP with FFV 14.xx.xx incorrectly allows this link set up in the following two configurations:

CA-25G-L DAC Requires RS-FEC per the IEEE standard, but allows the link to come up with FC-FEC or no FEC.

CA-25G-S DAC Requires RS-FEC or FC-FEC per the IEEE standard, but allows link with no FEC.

This issue has been fixed, but there may be a change in behavior when upgrading the firmware from the previous version.

Solution: Try one of the following if the 25G link fails after updating Dell Firmware DUP with FFV 15.xx.xx and later.

Enable a compatible FEC mode on the link partner.

Replace the cable with a 25G-N cable. If FC-FEC is configured, use a 25G-S cable.

Tested Switches

Table B-2 lists the switches that have been tested for interoperability with the 41000 Series Adapters. This list is based on switches that are available at the time of product release, and is subject to change over time as new switches enter the market or are discontinued.

Table B-2. Switches Tested for Interoperability

Manufacturer	Ethernet Switch Model
Arista	7060X 7160
Cisco	Nexus 3132 Nexus 3232C Nexus 5548 Nexus 5596T Nexus 6000
Dell EMC	S6100 Z9100
HPE	FlexFabric 5950
Mellanox	SN2410 SN2700

C Dell Z9100 Switch Configuration

The 41000 Series Adapters support connections with the Dell Z9100 Ethernet Switch. However, until the auto-negotiation process is standardized, the switch must be explicitly configured to connect to the adapter at 25Gbps.

To configure a Dell Z9100 switch port to connect to the 41000 Series Adapter at 25Gbps:

1. Establish a serial port connection between your management workstation and the switch.
2. Open a command line session, and then log in to the switch as follows:

```
Login: admin  
Password: admin
```

3. Enable configuration of the switch port:

```
Dell> enable  
Password: xxxxxxx  
Dell# config
```

4. Identify the module and port to be configured. The following example uses module 1, port 5:

```
Dell(conf)#stack-unit 1 port 5 ?  
portmode          Set portmode for a module  
Dell(conf)#stack-unit 1 port 5 portmode ?  
dual              Enable dual mode  
quad             Enable quad mode  
single           Enable single mode  
Dell(conf)#stack-unit 1 port 5 portmode quad ?  
speed            Each port speed in quad mode  
Dell(conf)#stack-unit 1 port 5 portmode quad speed ?  
10G              Quad port mode with 10G speed
```

```
25G                               Quad port mode with 25G speed
Dell(conf)#stack-unit 1 port 5 portmode quad speed 25G
```

For information about changing the adapter link speed, see [“Testing Network Connectivity” on page 304](#).

5. Verify that the port is operating at 25Gbps:

```
Dell# Dell#show running-config | grep "port 5"
stack-unit 1 port 5 portmode quad speed 25G
```

6. To disable auto-negotiation on switch port 5, follow these steps:

- a. Identify the switch port interface (module 1, port 5, interface 1) and confirm the auto-negotiation status:

```
Dell(conf)#interface tw 1/5/1

Dell(conf-if-tf-1/5/1)#intf-type cr4 ?
autoneg                               Enable autoneg
```

- b. Disable auto-negotiation:

```
Dell(conf-if-tf-1/5/1)#no intf-type cr4 autoneg
```

- c. Verify that auto-negotiation is disabled.

```
Dell(conf-if-tf-1/5/1)#do show run interface tw 1/5/1
!
interface twentyFiveGigE 1/5/1
no ip address
mtu 9416
switchport
flowcontrol rx on tx on
no shutdown
no intf-type cr4 autoneg
```

For more information about configuring the Dell Z9100 switch, refer to the *Dell Z9100 Switch Configuration Guide* on the Dell Support Web site:

support.dell.com

D VMware ESXi Enhanced Networking Stack Support

This appendix describes how to use the 41000 Series Adapter as a virtual NIC (vNIC) in a VMware hypervisor environment to support an NSX-Transformer (NSX-T) managed Virtual Distribution Switch (N-VDS), which is part of the Enhanced Network Stack (ENS).

NOTE

At the time of publication, this feature is supported only on VMware ESXi 6.7.

Overview

The VMware ESXi hypervisor provides various virtual networking capabilities for the enterprise network. The virtual switch is a key component in VMware infrastructure. There are different types of virtual switches:

- **vSphere Standard Switch**, which is configured at an individual ESXi host
- **Virtual Distributed Switch (VDS)**, which is configured and centrally managed at the vCenter level
- **NSX-T managed Virtual Distribution Switch (N-VDS)**

The 41000 Series Adapter can be programmed to support an N-VDS.

What is Enhanced Network Stack?

Enhanced Networking Stack (ENS) is one of the key features introduced in N-VDS.

The ENS provides an Enhanced Data Path in the ESX networking stack for Data Plane Development Kit (DPDK) applications running on a virtual machine (VM). The DPDK applications have strict latency and packet rate expectations. The ENS stack provides the superior network performances demanded by Network Function Virtualization (NFV) workloads.

The Enhanced Data Path also requires the physical device driver to work in polling mode, which eliminates the overhead of interrupt processing. Therefore, the ENS requires a Poll Mode Driver (PMD) to be implemented on the vNIC, as per the ENS driver development model introduced by VMware. An overview of ENS stack is shown in [Figure D-1](#).

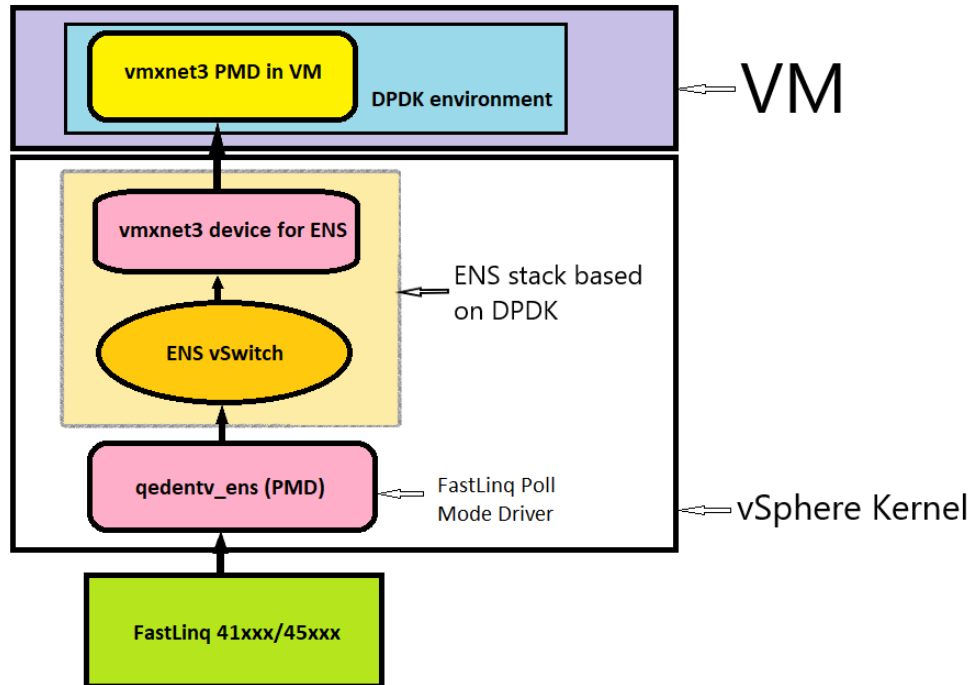


Figure D-1. ENS Stack Block Diagram

Poll Mode Driver

The ENS PMD for the 41000 Series Adapter, provided in the vSphere Installation bundle (VIB), works as an L2 PMD. It implements all the necessary callbacks required by the Enhanced Data Path mode of the N-VDS. A poll mode driver for a NIC is different from a native NIC driver: it performs transmit and receive operations without using interrupts, thereby keeping interrupt overheads to a minimum. Consequently, the PMD results in a low latency and high packet rate.

For instructions on installing the PMD on a 41000 Series Adapter, see [“Installing the Enhanced Network Stack Poll Mode Driver”](#) on page 37.

Capabilities

The PMD has the following capabilities:

- Transmit/receive burst
- Getting private statistics

- Getting NetQueue statistics
- Getting the link state

Features

The PMD has the following features:

- IPv4 and IPv6 stateless offloads
- Transmission control protocol (TCP)/user datagram protocol (UDP) checksum offload
- Transmit send offload (TSO)
- Jumbo MTU
- VLAN tag insertion/stripping
- Multiple queue support and NetQueue support
- Generic Network Virtualization Encapsulation (GENEVE) overlay
- ENS interface control. If you need to switch from ENS to a standard NIC L2 driver, you can make configuration changes in the NSX-T and reboot the system. Both L2 native and ENS drivers can share an adapter and claim different ports. For example, in a two-port adapter, Port1 can be claimed by the L2 native driver and Port2 can be claimed by the ENS driver.
- Teaming and HA. The setting are defined as part of the Uplink Profile create step of the Enhanced Data Path configuration procedure, as documented here:
<https://docs.vmware.com/en/VMware-NSX-T-Data-Center/2.5/installation/GUID-F459E3E4-F5F2-4032-A723-07D4051EFF8D.html>

Limitations

The PMD has the following limitations:

- Large receive offload (LRO)/transparent packet aggregation (TPA) is not supported
- RSS is not supported
- NetDump is not supported (use the regular L2 driver for NetDump)

Installing and Configuring an ENS-capable N-VDS

The following procedure describes, at a high level, how to install and configure an ENS-capable N-VDS that includes the 41000 Series Adapter attached to the N-VDS.

Prerequisites

Before creating an ENS-capable N-VDS, ensure that:

- The ENS driver VIB is installed (is this the PMD for the 41000 Series Adapter). Due to dependencies, the standard NIC driver qedentv VIB must also be installed.
- The ENS devices will be claimed during N-VDS configuration while the Transport Node is created. After the devices are claimed, you can identify them as uplinks claimed by qedentv_ens in `esxcfg-nics -e`
- The NSX-T manager and NSX-T controller (or NSX-T v2.3 and higher) have been installed.

Preparing the Host to Become Part of the NSX-T Fabric

To configure the host to be part of the NSX-T Fabric:

1. Login to the NSX manager and navigate to the home page (Figure D-2).

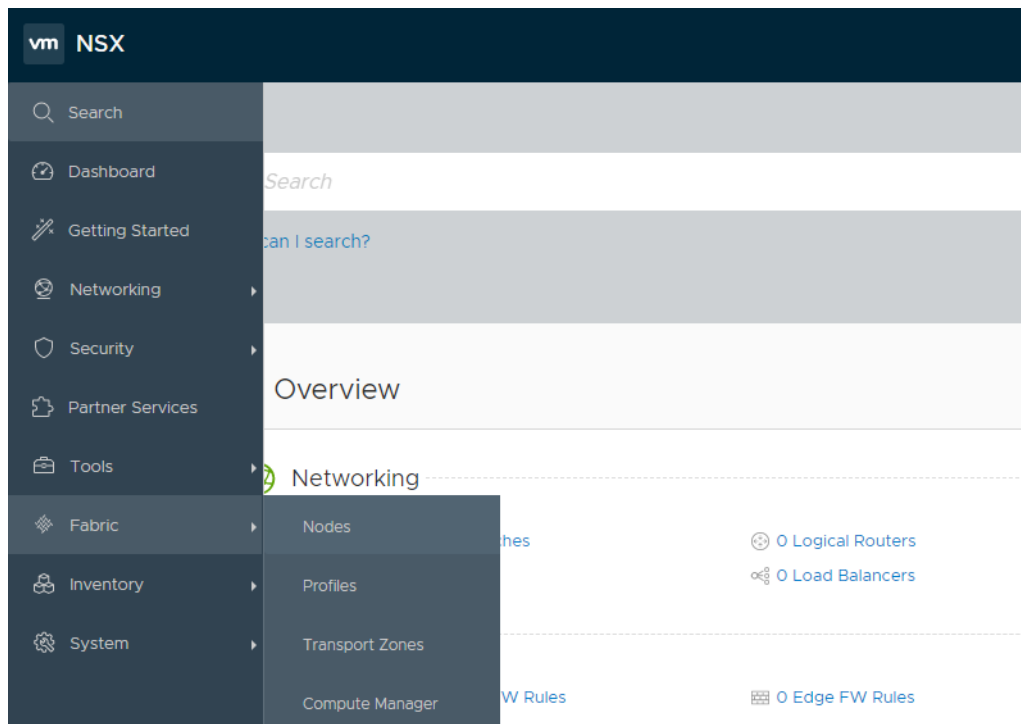


Figure D-2. NSX Home Page

2. Select **Fabric**, and then click **Nodes**.

The Nodes window appears (Figure D-3).

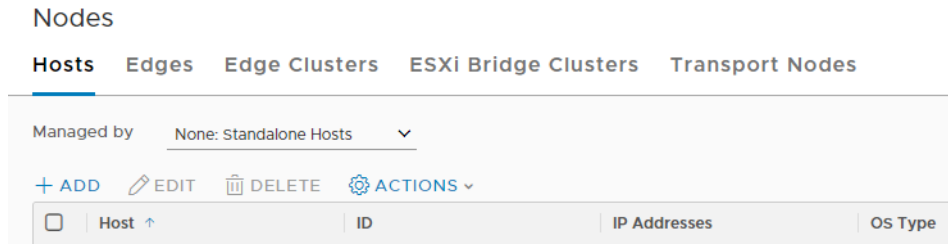


Figure D-3. Nodes Menu

3. Select the **Hosts** tab, and then click **+ Add**.
The Add Host window appears (Figure D-4).

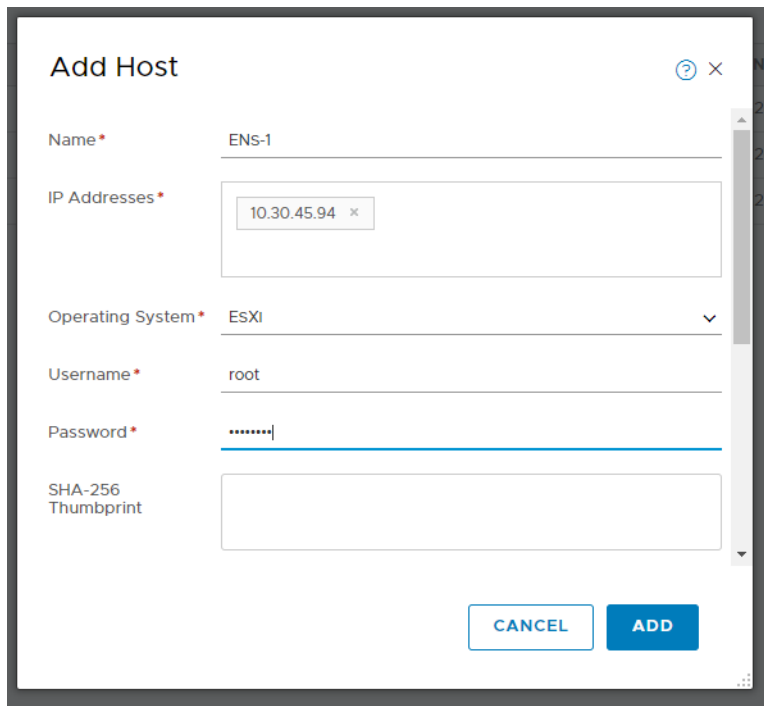
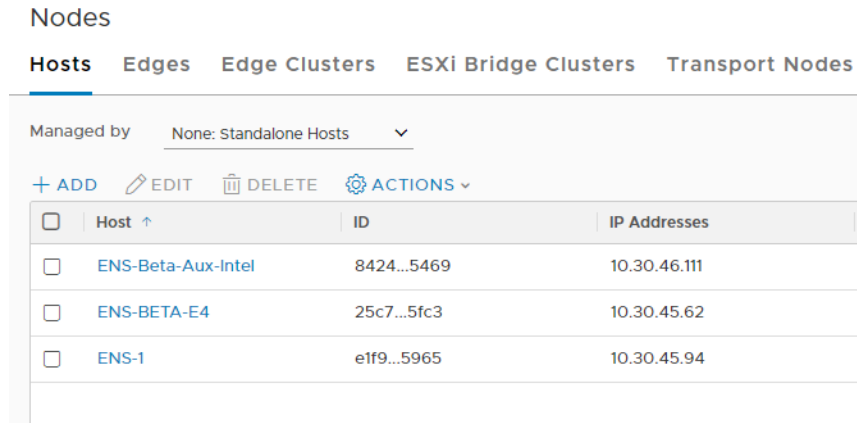


Figure D-4. Add Host Window

4. Add the host details, then click **Add** and accept the SHA-256 thumb print.

The NSX-T manager pushes and installs the necessary VIBs on the Host; this process can take a few minutes. The host is then added to the Hosts lists (Figure D-5).



The screenshot shows the NSX-T manager interface with the 'Hosts' tab selected. The interface includes a navigation bar with 'Hosts', 'Edges', 'Edge Clusters', 'ESXi Bridge Clusters', and 'Transport Nodes'. Below the navigation bar, there is a 'Managed by' dropdown menu set to 'None: Standalone Hosts'. A toolbar contains '+ ADD', 'EDIT', 'DELETE', and 'ACTIONS' buttons. The main content area displays a table with the following data:

<input type="checkbox"/>	Host ↑	ID	IP Addresses
<input type="checkbox"/>	ENS-Beta-Aux-Intel	8424...5469	10.30.46.111
<input type="checkbox"/>	ENS-BETA-E4	25c7...5fc3	10.30.45.62
<input type="checkbox"/>	ENS-1	ef9...5965	10.30.45.94

Figure D-5. NSX Host Added

Creating an Uplink Profile

An uplink profile defines policies for the link from the hypervisor hosts to the NSX-T logical switches or from NSX Edge nodes to top-of-rack switches.

To create an uplink profile:

1. Navigate to the NSX home page (Figure D-2).
2. Select **Fabric**, and then click **Profiles**.

The New Uplink Profile window appears (Figure D-6).

New Uplink Profile

Name *

Description

LAGs

+ ADD

<input type="checkbox"/>	Name *	LACP Mode	LACP Load Balancing *	Uplinks	LACP Time Out
No LAGs found					

Teamings

+ ADD

<input checked="" type="checkbox"/>	Name *	Teaming Policy *	Active Uplinks *	Standby Uplinks
<input checked="" type="checkbox"/>	[Default Teaming]	Failover Order	uplink1	

Transport VLAN

MTU *

Figure D-6. New Uplink Profile Window

3. Enter the uplink profile details:
 - a. **LAGS, LACP Load Balancing.** In this teaming policy, more than one Active uplinks are specified, and host switches can use all the active uplinks. This policy is only supported for ESXi hosts.
 - b. **Teamings, Teaming Policy.** Teaming policy dictates how the host switches use its uplinks to achieve load balancing and redundancy. There are two types of teaming policy that can be specified in an uplink profile:

- **Failover Order.** In this teaming policy, an active uplink is specified with an optional standby uplink. If the active uplink fails, the next available standby uplinks become active. This teaming policy can be used for both ESXi and KVM hypervisors.
 - **Load Balanced Source.** In this teaming policy, more than one active uplinks are specified. The host switches can use all active uplinks. This policy is only supported for ESXi hosts.
4. (Optional) Specify a **Transport VLAN**.
 5. Click **ADD** to create the uplink profile.

Configuring the Transport Zone

A transport zone is a container that defines the potential reach of transport nodes. Transport nodes are hypervisor hosts and NSX Edges that will participate in an NSX-T overlay.

A transport zone is created to isolate and segregate traffic. Virtual machines (VMs) on two separate ESXi hosts can communicate with each other only if they are in the same transport zone.

To add a transport zone:

1. Navigate to the NSX home page ([Figure D-2](#)).
2. Select **Fabric**, and then click **Transport Zones**.
The Transport Zones window appears ([Figure D-7](#)).

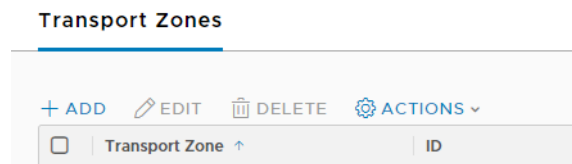


Figure D-7. Transport Zones Menu

3. Click **+ Add**.

The New Transport Zone window appears (Figure D-8).

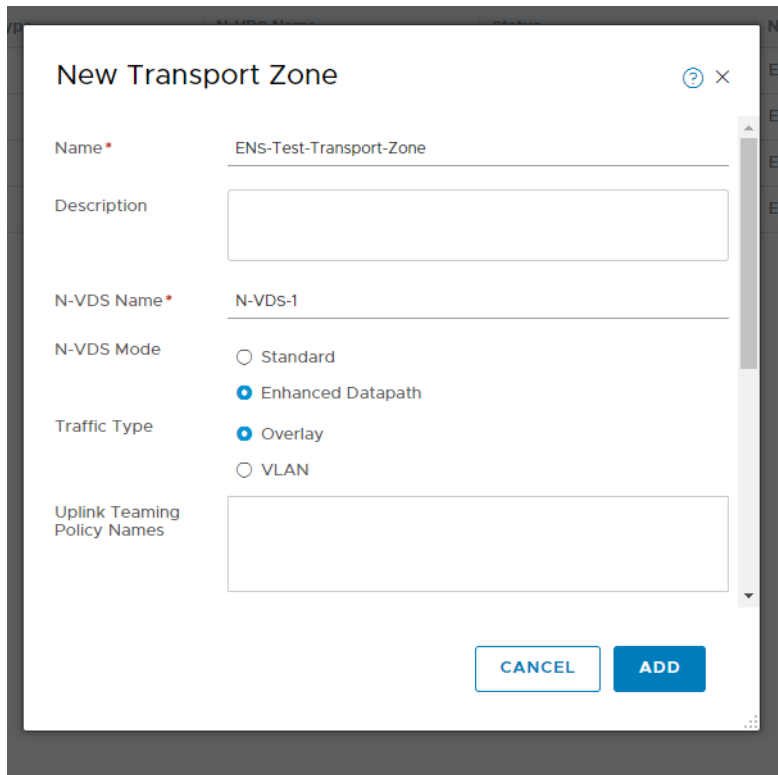


Figure D-8. New Transport Zone

4. Enter the transport zone details:
 - a. **Name.** Add a new name to identify this transport zone.
 - b. **N-VDS Name.** Add a new to identify the N-VDS switch to be created.
 - c. **N-VDS Mode.** Click **Enhanced Datapath** to select the type of stack to be used by the uplink device connected to this N-VDS switch.
 - d. **Traffic Type.** Click **Overlay** or **VLAN**.
5. Click **Add** to create the transport zone.

Creating an IP Pool

Follow these steps to create a range of IP addresses that are allocated to the new transport node.

To create an IP pool:

1. Navigate to the NSX home page (Figure D-2).
2. Select **Inventory**, and then click **Groups**.

The Groups window appears (Figure D-9).

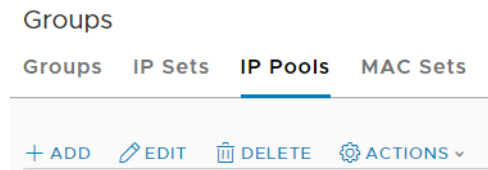


Figure D-9. Groups Menu

3. Select the **IP Pools** tab, and then click **+ Add**.

The Add New IP Pool window appears (Figure D-10).

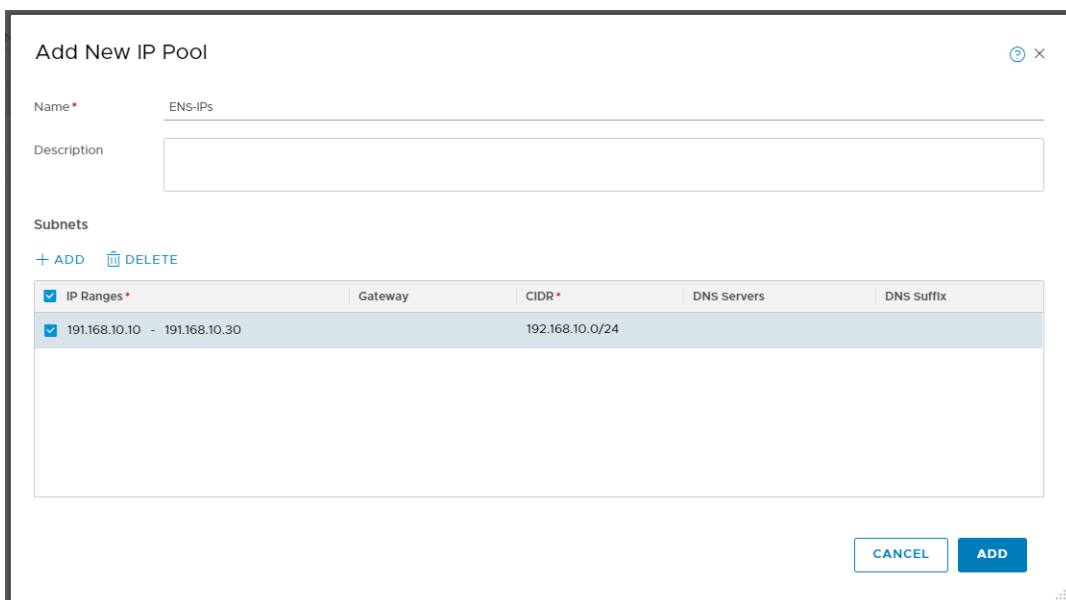


Figure D-10. Add New IP Pool Window

4. Enter the IP Pool details:
 - a. **Name.** IP pool name.
 - b. **IP Ranges.** The range of IP addresses.
 - c. Classless Inter-domain Routing (**CIDR**)
5. Click **Add** to create the IP pool.

Creating the Transport Node

The transport node connects the host to the NSX-T fabric.

To create a transport node:

1. Navigate to the NSX home page (Figure D-2).
2. Select **Fabric**, and then click **Notes**.

The Nodes window appears (Figure D-11).

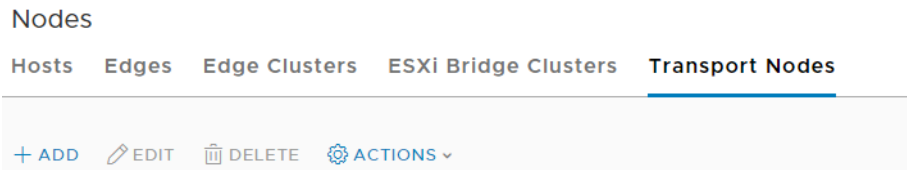


Figure D-11. Nodes Menu

3. Select the **Transport Nodes** tab, and then click **+ Add**.
The Add Transport Node window appears (Figure D-12).

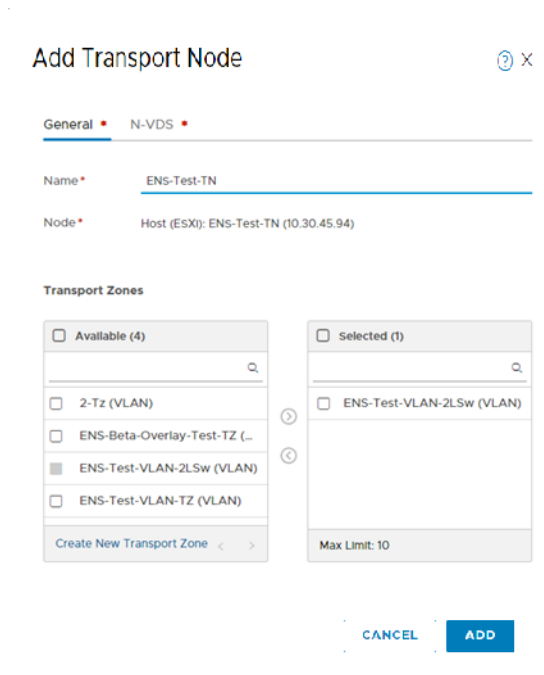


Figure D-12. Add Transport Node Window

4. Enter the transport node details:
 - a. **Name.** The name of the new transport node.
 - b. **Node.** Select the host's IP address from the drop-down list.
The host IP address will be visible in the drop down only after preparing the host for the NSX-T fabric, per [“Preparing the Host to Become Part of the NSX-T Fabric”](#) on page 323.
 - c. **Transport Zones.** Select the transport zone or zones to which the transport node will be attached.
5. Click **Add**.
The N-VDS window appears. ([Figure D-13](#)).

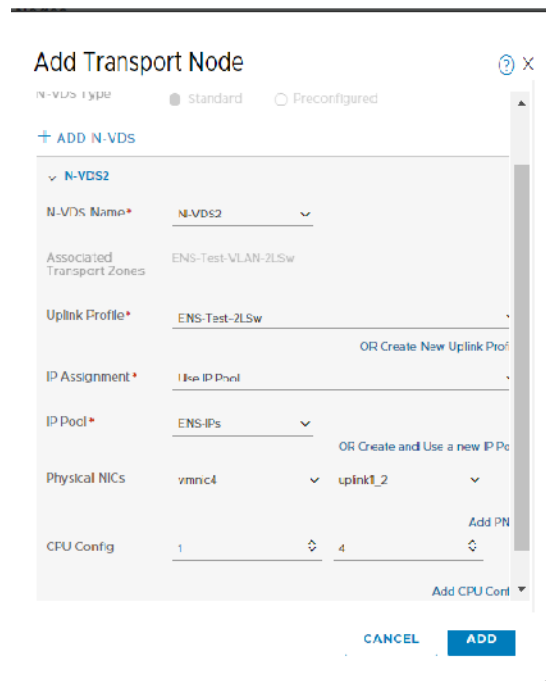


Figure D-13. Add Transport Node–ADD N-VDS Window

6. Enter the N-VDS details:
 - a. **N-VDS Name.** Use the name you created in [“Configuring the Transport Zone”](#) on page 327.
 - b. **Uplink Profile**
 - c. **IP Assignment**
 - d. **IP Pool.** If you selected **Use IP Pool** in the IP Assignment field, provide the IP pool name in the IP Pool field.

- e. **Physical NICs.** Select the uplink to be connected to this N-VDS. Only uplinks are marked as *Enhanced Datapath Enabled* will be connected to this N-VDS.
- f. **CPU Config.** Select the correct CPU configuration. Selecting the correct NUMA node is critical for performance.

7. Click **Add**.

Creating the Logical Switch

The logical switch provides layer 2 connectivity for the VMs to which it is attached. When logical switches are attached to transport zones, the switches connect to the N-VDS for networking. When deployed, a logical switch creates a broadcast domain to allow isolation of the VMs running in the infrastructure.

To create a logical switch:

1. Log into VMware NSX-T and navigate to the **Home** page (Figure D-14).

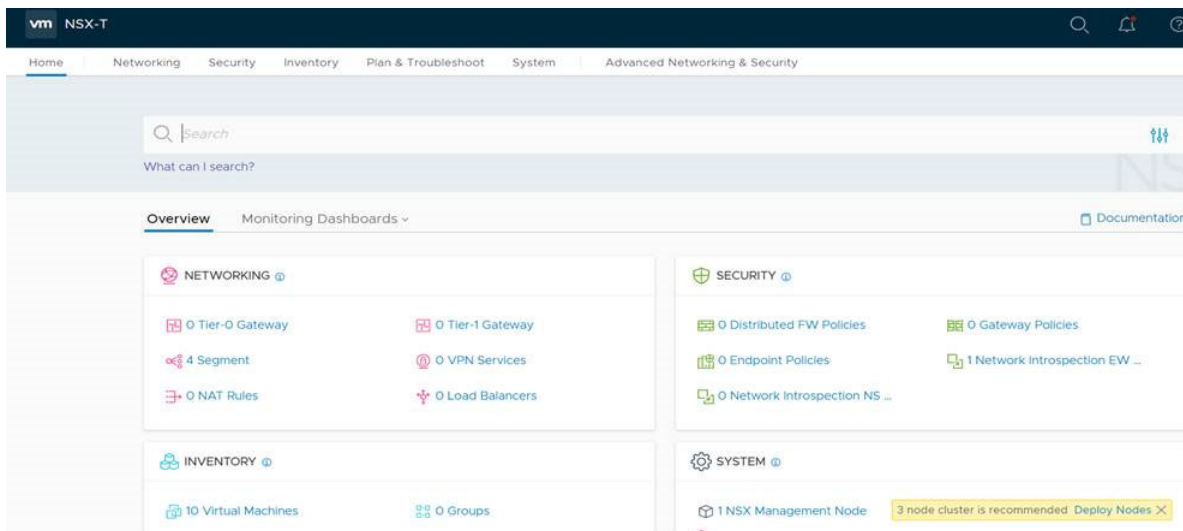


Figure D-14. NSX-T Home Window

- From the Home page, select **Advanced Networking & Security** (Figure D-15).

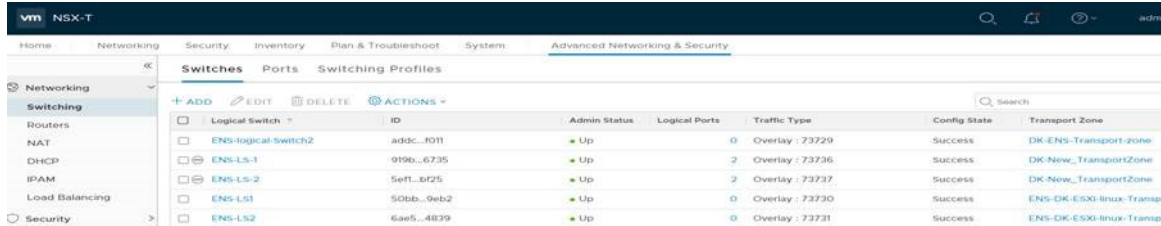


Figure D-15. Advanced Networking & Security Window

- Click **Add**.
- The Add New Logical Switch window appears. (Figure D-14).

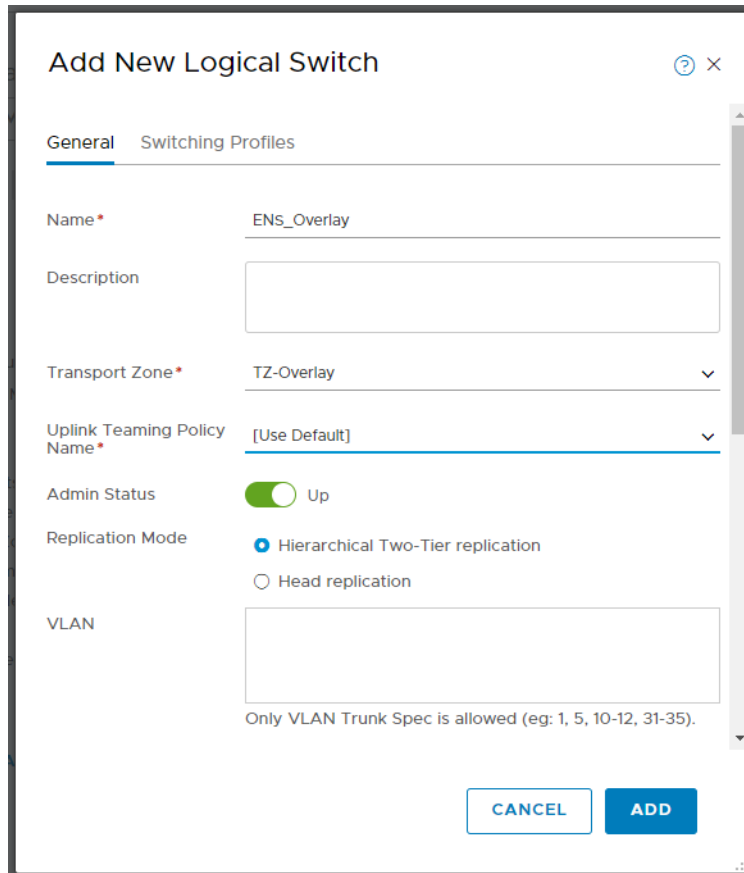


Figure D-16. Add New Logical Switch Window

5. Enter the logical switch details:
 - a. **Name**
 - b. **Transport Zone**
 - c. **Uplink Teaming Policy Name**
6. Click **Add**.
7. Add a vNIC on top of this network, which connects that vNIC to the N-VDS.
This step is done by logging into vSphere and then creating/editing the VM. In the Edit Settings window, add the network adapters ([Figure D-17](#)).

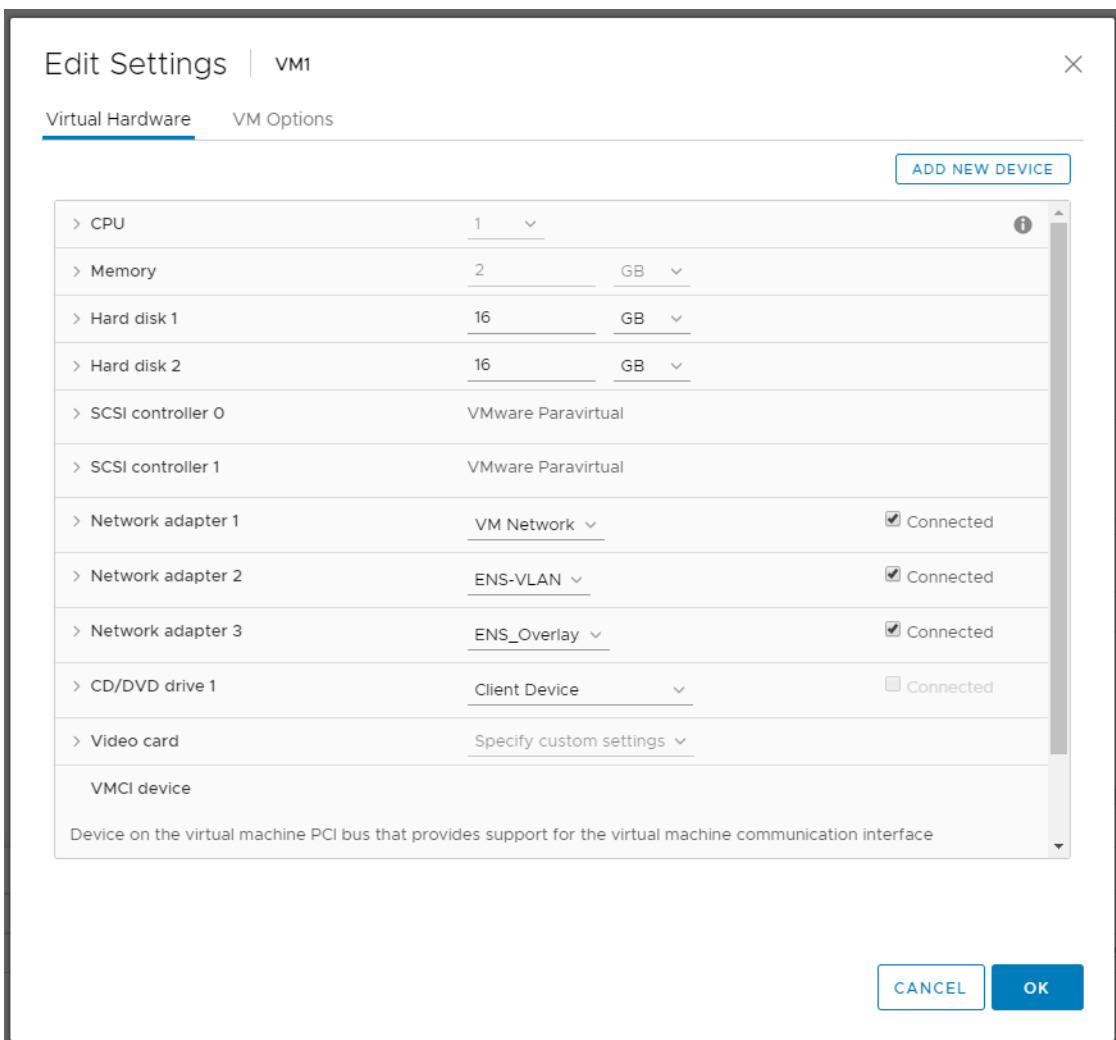


Figure D-17. Edit Settings, Virtual Hardware Window

8. Enter the required information, and then click **OK**.

Unified Enhanced Network Stack (UENS)

Unified Enhanced Network Stack (UENS) combines the enhanced network stack (ENS) poll mode driver (PMD) and the standard L2 driver. You can have either a single driver base for both driver modes or two separate drivers. Marvell has a single driver that supports both ENS and standard L2.

UENS has the following enhancements over ENS:

- Supports a new interrupt controlled data-path
- Provides additional control for control-path operations, which eases some strict limitations posed by ENS
- Adds additional information to the data path routines, which improves transmit completions and makes them easier to handle, as well as reducing packet drops in receive operations.

To have a single driver for both stacks, the driver should must support the three types of ENS uplink modes:

```
typedef enum vmk_EnsUplinkMode {  
    /** ENS is disabled. */  
    VMK_ENS_UPLINK_MODE_DISABLED = 0x1,  
  
    /** ENS is enabled with polling. */  
    VMK_ENS_UPLINK_MODE_POLLING = 0x2,  
  
    /** ENS is enabled with interrupt. */  
    VMK_ENS_UPLINK_MODE_INTERRUPT = 0x4,  
  
} vmk_EnsUplinkMode;
```

`VMK_ENS_UPLINK_MODE_DISABLED` is the standard NIC driver mode. The driver does not act as an ENS driver if the uplink device is in this mode; it continues to use the callbacks that are designed for the regular NIC driver. ENS operations are not gt invoked.

In `VMK_ENS_UPLINK_MODE_POLLING` mode, the driver acts as a PMD (similar to the behavior of the `qedentv_ens` driver for VMware ESXi 6.7). ENS operations are invoked in this mode, and interrupts are disabled.

`VMK_ENS_UPLINK_MODE_INTERRUPT` is a new mode where the ENS is enabled with interrupts. In this case, the data path is based on interrupts. Interrupts must be allocated for fast path, and interrupt service routines (ISRs) must be written to handle these interrupts. The `morePkts` argument of the `ensRx()` callback has a capability similar to NAPI: depending on the traffic, the driver switches in between poll mode and interrupt mode.

The configuration of N-VDS is the same as the configuration for ENS explained in [“Installing and Configuring an ENS-capable N-VDS” on page 322](#). The support for UENS is available from NSX 3.0 and ESXi 7.0.

ENS traffic is run in either poll mode or interrupt mode. An ENS N-VDS automatically controls the switch between the two modes. For test purposes, you can create a distributed virtual switch (dvSwitch) and explicitly define the mode using the commands in the following section.

Configuring ENS Using a Command Line Interface

Use the following commands in a command line interface (CLI) to configure the ENS.

To create an ENS Interrupt mode dvSwitch, issue the following command:

```
esxcfg-vswitch -ay --intr dvs1 --dvswitch --impl-class=vswitch
```

To create an ENS poll mode dvSwitch, issue the following command:

```
esxcfg-vswitch -ay dvs1 --dvswitch --impl-class=vswitch
```

To create an uplink port to attach a vmnic, issue the following command:

```
net-dvs -U 1 dvs1
```

To attach a vmnic, issue the following command:

```
esxcfg-vswitch -P vmnic9 -V uplink0 dvs1
```

To create a port group to create a vmknic, issue the following command:

```
net-dvs -A -p pg1 dvs1
```

To create a vmknic, issue the following command:

```
esxcli network ip interface add --dvs-name dvs1 --dvport-id pg1
```

To assign an IP address to a vmknic, issue the following command:

```
esxcli network ip interface ipv4 set --interface-name=vmk1  
--ipv4=192.168.20.11 --netmask=255.255.255.0 --type=static
```

To removing the vmknic interface, issue the following command:

```
esxcli network ip interface remove --dvs-name dvs1 --dvport-id pg1
```

To delete a vSwitch, issue the following command:

```
esxcfg-vswitch -d --dvswitch dvs1
```

You can use the output of the `esxcli-nics -e` command to verify the current mode of the ENS-capable device. Following is an example.

```
[root@localhost:~] esxcfg-nics -e
```

Name	Driver	ENS Capable	ENS Driven	INTR Capable	INTR Enabled	MAC Address	Description
vmnic0	ntg3	False	False	False	False	20:47:47:95:9c:64	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic1	ntg3	False	False	False	False	20:47:47:95:9c:65	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic2	ntg3	False	False	False	False	20:47:47:95:9c:66	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic3	ntg3	False	False	False	False	20:47:47:95:9c:67	Broadcom Corporation NetXtreme BCM5720 Gigabit Ethernet
vmnic4	qedentv	True	True	True	True	94:f1:28:b4:9c:4e	QLogic Corp. 10/25GbE 2P QL41222HLCU-HP Adapter
vmnic5	qedentv	True	True	True	False	94:f1:28:b4:9c:4f	QLogic Corp. 10/25GbE 2P QL41222HLCU-HP Adapter

As shown in the output, the ENS driven flag is `True` if the uplink device is driven by the driver in ENS mode. In addition, the poll mode and interrupt mode (in the `INTR Enabled` column) with a `True` value (highlighted in red) indicates that the device is in ENS + interrupt mode. A `False` value (highlighted in green) indicates that the device is in ENS + poll mode.

Reference Documents

See the following documents for more information about ENS and its components:

- DDK API reference for ESXi 6.7
- VMware NSX documentation center, <https://docs.vmware.com/en/VMware-NSX-T-Data-Center/index.html>
- DPDK web site, <https://www.dpdk.org/>

E Feature Constraints

This appendix provides information about feature constraints implemented in the current release.

These feature coexistence constraints may be removed in a future release. At that time, you should be able to use the feature combinations without any additional configuration steps beyond what would be usually required to enable the features.

Concurrent FCoE and iSCSI Is Not Supported on the Same Port in NPar Mode

The device does not support configuration of both FCoE-Offload and iSCSI-Offload on the same port when in NPar Mode. FCoE-Offload is supported on the second physical function (PF) and iSCSI-Offload is supported on the third PF in NPar mode. The device does support configuration of both FCoE-Offload and iSCSI-Offload on the same port when in single Ethernet PF DEFAULT Mode. Not all devices support FCoE-Offload and iSCSI-Offload.

After a PF with either an iSCSI or FCoE personality has been configured on a port using either HII or Marvell management tools, configuration of the storage protocol on another PF is disallowed by those management tools.

Because storage personality is disabled by default, only the personality that has been configured using HII or Marvell management tools is written in NVRAM configuration. When this limitation is removed, users can configure additional PFs on the same port for storage in NPar Mode.

Concurrent RoCE and iWARP Is Not Supported on the Same Physical Function

RoCE and iWARP are not supported on the same PF. The UEFI HII and Marvell management tools allow users to configure both concurrently, but the RoCE functionality takes precedence over the iWARP functionality in this case, unless overridden by the in-OS driver settings.

NIC and SAN Boot to Base Is Supported Only on Select PFs

Ethernet (such as software iSCSI remote boot) and PXE boot are currently supported only on the first Ethernet PF of a physical port. In NPar Mode configuration, the first Ethernet PF (that is, not the other Ethernet PFs) supports Ethernet (such as software iSCSI remote boot) and PXE boot. Not all devices support FCoE-Offload and iSCSI-Offload.

- When the **Virtualization** or **Multi-Function Mode** is set to **NPar**, FCoE-Offload boot is supported on the second PF of a physical port, iSCSI-Offload boot is supported on the third PF of a physical port, and Ethernet (such as software iSCSI) and PXE boot are supported on the first PF of a physical port.
- iSCSI and FCoE boot is limited to a single target per boot session.
- Only one boot mode is allowed per physical port.
- iSCSI-Offload and FCoE-Offload boot is only supported in NPar mode.

F Revision History

Document Revision History
Revision A, April 28, 2017
Revision B, August 24, 2017
Revision C, October 1, 2017
Revision D, January 24, 2018
Revision E, March 15, 2018
Revision F, April 19, 2018
Revision G, May 22, 2018
Revision H, August 23, 2018
Revision J, January 23, 2019
Revision K, July 2, 2019
Revision L, July 3, 2019
Revision M, October 16, 2019
Revision N, April 3, 2020
Revision P, May 14, 2020
Revision R, June 24, 2020
Revision T, July 7, 2020
Revision W, Dec xx, 2020
Revision X, January 29, 2020

Changes	Sections Affected
<p>Added support for the following OSs: RHEL 7.9, 8.2, 8.3 SLES 15 SP2 Azure Stack HCI</p> <p>Removed support for the following OSs: Windows 2012 (all versions) RHEL 7.7 Citrix Hypervisor 7.0, 7.1, 8.0, 8.1</p> <p>Removed the NOTE about QCC GUI being the only GUI management tool across adapters.</p> <p>Removed section referencing Marvell Web site as the source for updates and documentation.</p> <p>Added TUV IEC 62368 2nd and 3rd Edition CB</p> <p>In VMwareDirect Path I/O bullet: Added over PCI physical functions in first line. Changed second paragraph, third sentence to “Sharing PCIe physical functions across the hypervisor...”</p> <p>In the NOTE, referred the user to the <i>Read Me</i> and <i>Release Notes</i> for the most up-to-date OS information.</p> <p>In Table 3-5, changed the footnote to indicate that “...the NIC, RoCE, FCoE, and iSCSI drivers have been combined as a single component package. This package can be installed using standard ESXi installation methods and commands.”</p> <p>In the NOTE: Clarified the first sentence to “The iSCSI interface (iSCSI offload)...” Changed the sixth and seventh sentences to “This is not a comprehensive L2 solution. Do not use this implementation to carry regular networking traffic; that is, do not assign this to a VM as a network adapter.” Added a new last sentence referencing the iSCSI offload limitations listed in the iSCSI Offload in VMware EXSi section.</p> <p>Added new section on default NPar/NParEP mode numbering.</p>	<p>All</p> <p>“Supported Products” on page xix</p> <p>was “Downloading Updates and Documentation”</p> <p>“Product Safety Compliance” on page xxvi</p> <p>“Features” on page 1</p> <p>“Software Requirements” on page 6</p> <p>“VMware Drivers and Driver Packages” on page 35</p> <p>“iSCSI Support” on page 43</p> <p>“Default NPar/NParEP Mode Numbering” on page 56</p>

<p>In the NOTE, added a second paragraph indicating that on QL41164Hxxx quad-port adapters, FCoE and iSCSI storage offloads are only enabled on partition 2.</p> <p>Added a bullet stating that the PermitTotalPortShutdown feature cannot be used on ports configured to boot from SAN.</p> <p>Corrected step 4 to say, "..., and then press CTRL+X to start."</p> <p>Added a NOTE stating that the PermitTotalPortShutdown feature cannot be used on ports configured to boot from SAN</p> <p>Added new section on how to use NPar to configure FCoE offloads on the adapter.</p> <p>In Step 5, in the NOTE, added a reference to the <i>Application Note, Enabling Storage Offload on Dell and Marvell FastLinQ 41000 Series Adapters</i> to provide information on enabling the FCoE Mode feature.</p> <p>Added section for configuring FCoE boot from SAN for RHEL.</p> <p>Added a new second paragraph describing optimal performance settings.</p> <p>In Step 1, changed part c to "Select the current vSphere version being used."</p> <p>Added a new second paragraph describing optimal performance settings.</p> <p>Added a NOTE to the end of the section listing iSCSI Offload limitations.</p> <p>Removed the problem "FCoE devices are not discovered after the link is disrupted on the initiator port."</p> <p>In the solution that sets the driver module parameter <code>qedf_fipvlan_retries</code>, added quotes to the command being issued:</p> <pre>esxcfg-module -s 'qedf_fipvlan_retries=X' qedf</pre>	<p>"Configuring FCoE Boot" on page 67, "Configuring iSCSI Boot" on page 68</p> <p>"Before You Begin" on page 102</p> <p>"Configuring iSCSI Boot from SAN for RHEL 7.8 and Later" on page 107</p> <p>"FCoE Boot from SAN" on page 116</p> <p>"Enabling NPar and the FCoE HBA" on page 117</p> <p>"Configuring Adapter UEFI Boot Mode" on page 118</p> <p>"Configuring FCoE Boot from SAN for RHEL 7.4 and Later" on page 126</p> <p>"Configuring MTU" on page 175</p> <p>"Configuring a Paravirtual RDMA Device (PVRDMA)" on page 176</p> <p>"Configuring iWARP on Windows" on page 187</p> <p>"iSCSI Offload in VMware ESXi" on page 221</p> <p>"VMware ESX (Driver v2.x.y.z and Later)" on page 307</p> <p>"VMware ESX (Driver v1.x.y.z and Later)" on page 308</p>
---	--

<p>Added an additional issue: <code>qedf_fipvlan_retries</code> is not supported in qedf driver version 2.2.32.0. Added fix.</p> <p>Added known cable issue and workaround.</p> <p>Added new Steps 1–3 and corresponding figures.</p> <p>Clarified Step 7.</p> <p>Added a section on UENS.</p> <p>In the first bullet, removed the PDF name of the DDK API reference for ESXi 6.7.</p>	<p>“VMware ESX (Driver v2.x.y.z and Later)” on page 307</p> <p>“Known Issue: Using SmartAN Mode with Invalid FEC Configuration for 25G DAC” on page 316</p> <p>“Creating the Logical Switch” on page 332</p> <p>“Unified Enhanced Network Stack (UENS)” on page 335</p> <p>“Reference Documents” on page 337</p>
--	--

Glossary

ACPI

The *Advanced Configuration and Power Interface (ACPI)* specification provides an open standard for unified operating system-centric device configuration and power management. The ACPI defines platform-independent interfaces for hardware discovery, configuration, power management, and monitoring. The specification is central to operating system-directed configuration and Power Management (OSPM), a term used to describe a system implementing ACPI, which therefore removes device management responsibilities from legacy firmware interfaces.

adapter

The board that interfaces between the host system and the target devices. Adapter is synonymous with Host Bus Adapter, host adapter, and board.

adapter port

A port on the adapter board.

Advanced Configuration and Power Interface

See [ACPI](#).

bandwidth

A measure of the volume of data that can be transmitted at a specific transmission rate. A 1Gbps or 2Gbps Fibre Channel port can transmit or receive at nominal rates of 1 or 2Gbps, depending on the device to which it is connected. This corresponds to actual bandwidth values of 106MB and 212MB, respectively.

BAR

Base address register. Used to hold memory addresses used by a device, or offsets for port addresses. Typically, memory address BARs must be located in physical RAM while I/O space BARs can reside at any memory address (even beyond physical memory).

base address register

See [BAR](#).

basic input output system

See [BIOS](#).

BIOS

Basic input output system. Typically in Flash PROM, the program (or utility) that serves as an interface between the hardware and the operating system and allows booting from the adapter at startup.

challenge-handshake authentication protocol

See [CHAP](#).

CHAP

Challenge-handshake authentication protocol (CHAP) is used for remote logon, usually between a client and server or a Web browser and Web server. A challenge/response is a security mechanism for verifying the identity of a person or process without revealing a secret password that is shared by the two entities. Also referred to as a *three-way handshake*.

CNA

See [Converged Network Adapter](#).

Converged Network Adapter

Marvell Converged Network Adapters support both data networking (TCP/IP) and storage networking ([Fibre Channel](#)) traffic on a single I/O adapter using two new technologies: Enhanced Ethernet and Fibre Channel over Ethernet ([FCoE](#)).

data center bridging

See [DCB](#).

data center bridging exchange

See [DCBX](#).

DCB

Data center bridging. Provides enhancements to existing 802.1 bridge specifications to satisfy the requirements of protocols and applications in the data center. Because existing high-performance data centers typically comprise multiple application-specific networks that run on different link layer technologies (Fibre Channel for storage and Ethernet for network management and LAN connectivity), DCB enables 802.1 bridges to be used for the deployment of a converged network where all applications can be run over a single physical infrastructure.

DCBX

Data center bridging exchange. A protocol used by [DCB](#) devices to exchange configuration information with directly connected peers. The protocol may also be used for misconfiguration detection and for configuration of the peer.

device

A [target](#), typically a disk drive. Hardware such as a disk drive, tape drive, printer, or keyboard that is installed in or connected to a system. In Fibre Channel, a *target device*.

DHCP

Dynamic host configuration protocol. Enables computers on an IP network to extract their configuration from servers that have information about the computer only after it is requested.

driver

The software that interfaces between the file system and a physical data storage device or network media.

dynamic host configuration protocol

See [DHCP](#).

eCore

A layer between the OS and the hardware and firmware. It is device-specific and OS-agnostic. When eCore code requires OS services (for example, for memory allocation, PCI configuration space access, and so on) it calls an abstract OS function that is implemented in OS-specific layers. eCore flows may be driven by the hardware (for example, by an interrupt) or by the OS-specific portion of the driver (for example, loading and unloading the load and unload).

EEE

Energy-efficient Ethernet. A set of enhancements to the twisted-pair and backplane Ethernet family of computer networking standards that allows for less power consumption during periods of low data activity. The intention was to reduce power consumption by 50 percent or more, while retaining full compatibility with existing equipment. The Institute of Electrical and Electronics Engineers (IEEE), through the IEEE 802.3az task force, developed the standard.

EFI

Extensible firmware interface. A specification that defines a software interface between an operating system and platform firmware. EFI is a replacement for the older BIOS firmware interface present in all IBM PC-compatible personal computers.

energy-efficient Ethernet

See [EEE](#).

enhanced transmission selection

See [ETS](#).

Ethernet

The most widely used LAN technology that transmits information between computers, typically at speeds of 10 and 100 million bits per second (Mbps).

ETS

Enhanced transmission selection. A standard that specifies the enhancement of transmission selection to support the allocation of bandwidth among traffic classes. When the offered load in a traffic class does not use its allocated bandwidth, enhanced transmission selection allows other traffic classes to use the available bandwidth. The bandwidth-allocation priorities coexist with strict priorities. ETS includes managed objects to support bandwidth allocation. For more information, refer to:

<http://ieee802.org/1/pages/802.1az.html>

extensible firmware interface

See [EFI](#).

FCoE

Fibre Channel over Ethernet. A new technology defined by the T11 standards body that allows traditional Fibre Channel storage networking traffic to travel over an Ethernet link by encapsulating Fibre Channel frames inside Layer 2 Ethernet frames. For more information, visit www.fcoe.com.

Fibre Channel

A high-speed serial interface technology that supports other higher layer protocols such as [SCSI](#) and [IP](#).

Fibre Channel over Ethernet

See [FCoE](#).

file transfer protocol

See [FTP](#).

FTP

File transfer protocol. A standard network protocol used to transfer files from one host to another host over a TCP-based network, such as the Internet. FTP is required for out-of-band firmware uploads that will complete faster than in-band firmware uploads.

HBA

See [Host Bus Adapter](#).

HII

Human interface infrastructure. A specification (part of UEFI 2.1) for managing user input, localized strings, fonts, and forms, that allows OEMs to develop graphical interfaces for preboot configuration.

host

One or more adapters governed by a single memory or CPU complex.

Host Bus Adapter

An adapter that connects a host system (the computer) to other network and storage devices.

human interface infrastructure

See [HII](#).

IEEE

Institute of Electrical and Electronics Engineers. An international nonprofit organization for the advancement of technology related to electricity.

Internet Protocol

See [IP](#).

Internet small computer system interface

See [iSCSI](#).

Internet wide area RDMA protocol

See [iWARP](#).

IP

Internet protocol. A method by which data is sent from one computer to another over the Internet. IP specifies the format of packets, also called *datagrams*, and the addressing scheme.

IQN

iSCSI qualified name. iSCSI node name based on the initiator manufacturer and a unique device name section.

iSCSI

Internet small computer system interface. Protocol that encapsulates data into IP packets to send over Ethernet connections.

iSCSI qualified name

See [IQN](#).

iWARP

Internet wide area [RDMA](#) protocol. A networking protocol that implements RDMA for efficient data transfer over IP networks. iWARP is designed for multiple environments, including LANs, storage networks, data center networks, and WANs.

jumbo frames

Large IP frames used in high-performance networks to increase performance over long distances. Jumbo frames generally means 9,000 bytes for Gigabit [Ethernet](#), but can refer to anything over the IP [MTU](#), which is 1,500 bytes on an Ethernet.

large send offload

See [LSO](#).

Layer 2

Refers to the data link layer of the multilayered communication model, Open Systems Interconnection (OSI). The function of the data link layer is to move data across the physical links in a network, where a switch redirects data messages at the Layer 2 level using the destination MAC address to determine the message destination.

Link Layer Discovery Protocol

See [LLDP](#).

LLDP

A vendor-neutral Layer 2 protocol that allows a network device to advertise its identity and capabilities on the local network. This protocol supersedes proprietary protocols like Cisco Discovery Protocol, Extreme Discovery Protocol, and Nortel Discovery Protocol (also known as SONMP).

Information gathered with LLDP is stored in the device and can be queried using SNMP. The topology of a LLDP-enabled network can be discovered by crawling the hosts and querying this database.

LSO

Large send offload. LSO Ethernet adapter feature that allows the TCP/IP network stack to build a large (up to 64KB) TCP message before sending it to the adapter. The adapter hardware segments the message into smaller data packets (frames) that can be sent over the wire: up to 1,500 bytes for standard Ethernet frames and up to 9,000 bytes for jumbo Ethernet frames. The segmentation process frees up the server CPU from having to segment large TCP messages into smaller packets that will fit inside the supported frame size.

maximum transmission unit

See [MTU](#).

message signaled interrupts

See [MSI](#), [MSI-X](#).

MSI, MSI-X

Message signaled interrupts. One of two PCI-defined extensions to support message signaled interrupts (MSIs), in PCI 2.2 and later and PCI Express. MSIs are an alternative way of generating an interrupt through special messages that allow emulation of a pin assertion or deassertion.

MSI-X (defined in PCI 3.0) allows a device to allocate any number of interrupts between 1 and 2,048 and gives each interrupt separate data and address registers. Optional features in MSI (64-bit addressing and interrupt masking) are mandatory with MSI-X.

MTU

Maximum transmission unit. Refers to the size (in bytes) of the largest packet (IP datagram) that a specified layer of a communications protocol can transfer.

network interface card

See [NIC](#).

NIC

Network interface card. Computer card installed to enable a dedicated network connection.

NIC partitioning

See [NPar](#).

non-volatile random access memory

See [NVRAM](#).

non-volatile memory express

See [NVMe](#).

NPar

NIC partitioning. The division of a single NIC port into multiple physical functions or partitions, each with a user-configurable bandwidth and personality (interface type). Personalities include **NIC**, **FCoE**, and **iSCSI**.

NVRAM

Non-volatile random access memory. A type of memory that retains data (configuration settings) even when power is removed. You can manually configure NVRAM settings or restore them from a file.

NVMe

A storage access method designed for solid-state drives (SSDs).

OFED™

OpenFabrics Enterprise Distribution. An open source software for RDMA and kernel bypass applications.

PCI™

Peripheral component interface. A 32-bit local bus specification introduced by Intel®.

PCI Express (PCIe)

A third-generation I/O standard that allows enhanced Ethernet network performance beyond that of the older peripheral component interconnect (PCI) and PCI extended (PCI-X) desktop and server slots.

QoS

Quality of service. Refers to the methods used to prevent bottlenecks and ensure business continuity when transmitting data over virtual ports by setting priorities and allocating bandwidth.

quality of service

See **QoS**.

PF

Physical function.

RDMA

Remote direct memory access. The ability for one node to write directly to the memory of another (with address and size semantics) over a network. This capability is an important feature of **VI** networks.

reduced instruction set computer

See **RISC**.

remote direct memory access

See **RDMA**.

RISC

Reduced instruction set computer. A computer microprocessor that performs fewer types of computer instructions, thereby operating at higher speeds.

RDMA over Converged Ethernet

See **RoCE**.

RoCE

RDMA over Converged Ethernet. A network protocol that allows remote direct memory access (RDMA) over a converged or a non-converged Ethernet network. RoCE is a link layer protocol that allows communication between any two hosts in the same Ethernet broadcast domain.

SCSI

Small computer system interface. A high-speed interface used to connect devices, such as hard drives, CD drives, printers, and scanners, to a computer. The SCSI can connect many devices using a single controller. Each device is accessed by an individual identification number on the SCSI controller bus.

SerDes

Serializer/deserializer. A pair of functional blocks commonly used in high-speed communications to compensate for limited input/output. These blocks convert data between serial data and parallel interfaces in each direction.

serializer/deserializer

See [SerDes](#).

single root input/output virtualization

See [SR-IOV](#).

small computer system interface

See [SCSI](#).

SR-IOV

Single root input/output virtualization. A specification by the PCI SIG that enables a single PCIe device to appear as multiple, separate physical PCIe devices. SR-IOV permits isolation of PCIe resources for performance, interoperability, and manageability.

target

The storage-device endpoint of a SCSI session. Initiators request data from targets. Targets are typically disk-drives, tape-drives, or other media devices. Typically a SCSI peripheral device is the target but an adapter may, in some cases, be a target. A target can contain many LUNs.

A target is a device that responds to a requested by an initiator (the host system). Peripherals are targets, but for some commands (for example, a SCSI COPY command), the peripheral may act as an initiator.

TCP

Transmission control protocol. A set of rules to send data in packets over the Internet protocol.

TCP/IP

Transmission control protocol/Internet protocol. Basic communication language of the Internet.

TLV

Type-length-value. Optional information that may be encoded as an element inside of the protocol. The type and length fields are fixed in size (typically 1–4 bytes), and the value field is of variable size. These fields are used as follows:

- **Type**—A numeric code that indicates the kind of field that this part of the message represents.
- **Length**—The size of the value field (typically in bytes).
- **Value**—Variable-sized set of bytes that contains data for this part of the message.

transmission control protocol

See [TCP](#).

transmission control protocol/Internet protocol

See [TCP/IP](#).

type-length-value

See [TLV](#).

UDP

User datagram protocol. A connectionless transport protocol without any guarantee of packet sequence or delivery. It functions directly on top of IP.

UEFI

Unified extensible firmware interface. A specification detailing an interface that helps hand off control of the system for the preboot environment (that is, after the system is powered on, but before the operating system starts) to an operating system, such as Windows or Linux. UEFI provides a clean interface between operating systems and platform firmware at boot time, and supports an architecture-independent mechanism for initializing add-in cards.

unified extensible firmware interface

See [UEFI](#).

user datagram protocol

See [UDP](#).

VF

Virtual function.

VI

Virtual interface. An initiative for remote direct memory access across Fibre Channel and other communication protocols. Used in clustering and messaging.

virtual interface

See [VI](#).

virtual logical area network

See [vLAN](#).

virtual machine

See [VM](#).

virtual port

See [vPort](#).

vLAN

Virtual logical area network (LAN). A group of hosts with a common set of requirements that communicate as if they were attached to the same wire, regardless of their physical location. Although a vLAN has the same attributes as a physical LAN, it allows for end stations to be grouped together even if they are not located on the same LAN segment. vLANs enable network reconfiguration through software, instead of physically relocating devices.

VM

Virtual machine. A software implementation of a machine (computer) that executes programs like a real machine.

vPort

Virtual port. Port number or service name associated with one or more virtual servers. A virtual port number should be the same TCP or UDP port number to which client programs expect to connect.

wake on LAN

See [WoL](#).

WoL

Wake on LAN. An Ethernet computer networking standard that allows a computer to be remotely switched on or awakened by a network message sent usually by a simple program executed on another computer on the network.



Marvell first revolutionized the digital storage industry by moving information at speeds never thought possible. Today, that same breakthrough innovation remains at the heart of the company's storage, networking and connectivity solutions. With leading intellectual property and deep system-level knowledge, Marvell semiconductor solutions continue to transform the enterprise, cloud, automotive, industrial, and consumer markets. For more information, visit www.marvell.com.

© 2021 Marvell. All rights reserved. The MARVELL mark and M logo are registered and/or common law trademarks of Marvell and/or its Affiliates in the US and/or other countries. This document may also contain other registered or common law trademarks of Marvell and/or its Affiliates.

Doc. No. AH0054602-00 Rev. X Revised: January 29, 2021