Broadcom Ethernet NIC Controller Configuration Guide

Table of Contents

Introduction

Provides information to quickly install and configure Broadcom Etherenet NIC controllers.

Designed for today's enterprise and cloud-scale environments, Broadcom's Ethernet NIC controllers are the ideal solution for high-performance virtualization, intelligent flow processing, secure data center connectivity, and machine learning.

The introduction consists of the following sections:

- Regulatory and Safety Approvals
- Functional Description
- Network Link and Activity Indication
- Features

Regulatory and Safety Approvals

The following sections detail the regulatory approvals, safety approvals, and warning statements for Broadcom Etherenet NIC controllers. See the individual data sheets for the product classification and referenced standard dates.

Table 1: Regulatory Approvals

Standard/Country	Certification Type	Compliance
CE/EU	EN 55032 EN 55024/EN 55035 EN 61000-3-2 EN 61000-3-3	CE report and CE sDoC
FCC/USA	CFR47, Part 15	FCC sDoC and EMC report referencing FCC part 15 regulations
IC/Canada	ICES-003	Report referencing IC standards
ACA/Australia, New Zealand	AS/NZS CISPR 32	sDoC certificate RCM Mark
BSMI/Taiwan	CNS13438, CNS15663	BSMI certificate
MIC/S. Korea	KN 32 and KN 35	Korea certificate MSIP Mark
VCCI/Japan	V-3/VCCI CISPR 32	Copy of VCCI on-line certificate

Table 2: Safety Approvals

Item	Applicable Standard	Approval/Certificate
CE/European Union	IEC 62368-1	CB report and certificate
UL/USA	IEC 62368-1 CTUVus UL	UL report and certificate
CSA/Canada	CSA 22.2 No. 950	CSA report and certificate

Functional Description

The Broadcom BCM9574XX and BCM9575XX family of Ethernet controllers are highly-integrated, full-featured Ethernet LAN controllers optimized for data center and cloud infrastructures. These controllers support 200G/100G/50G/40G/25G/10G/1G in single, dual, or quad-port configurations. These controllers can support up to sixteen lanes of PCIe Gen 3. The BCM9575XX family additionally supports PCIe Gen4. An extensive set of stateless offloads and virtualization offloads to enhance packet processing efficiency are included to enable low-overhead, high-speed network communications.

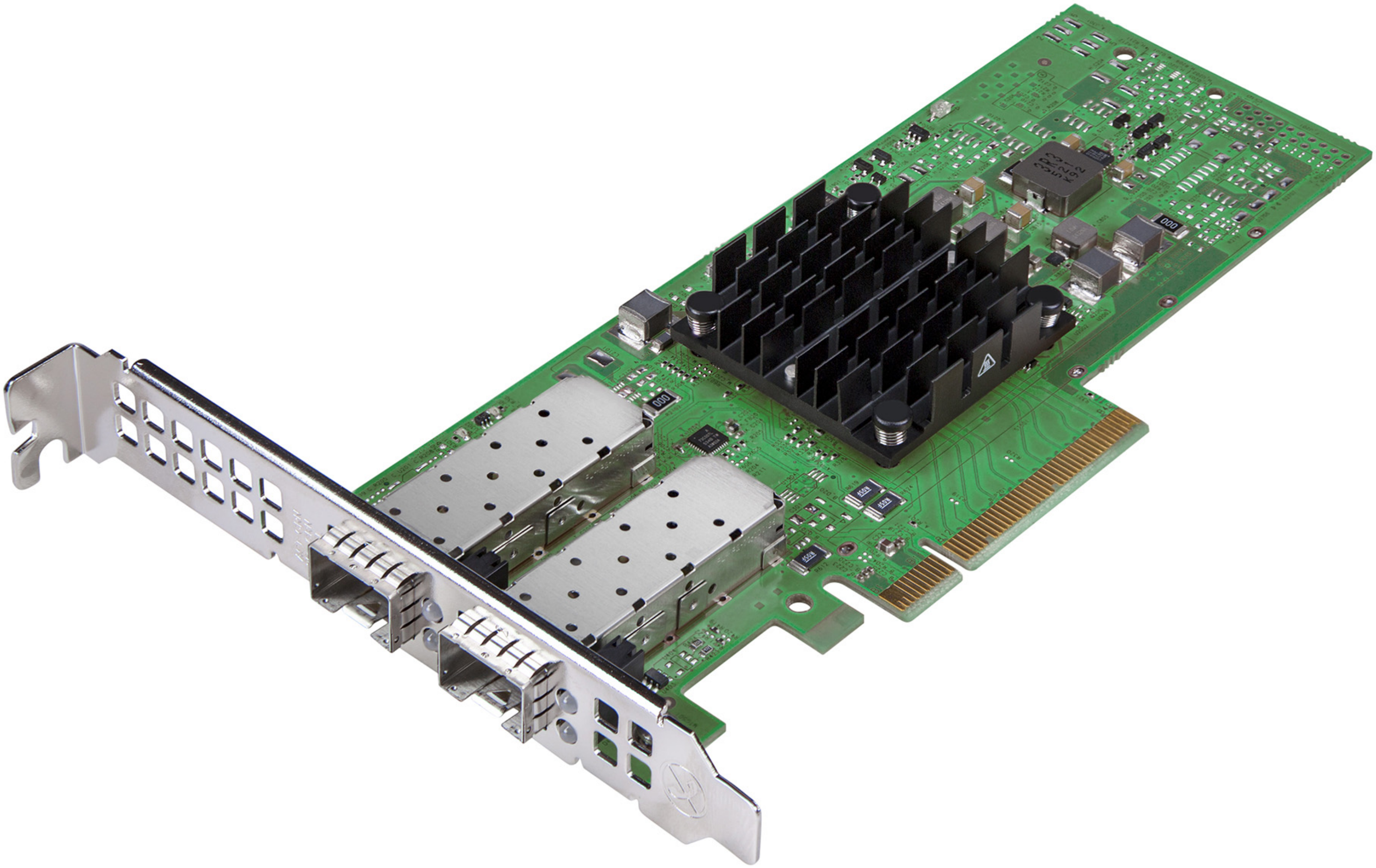
Table 3: Functional Description

Network Interface Card	**Description**
BCM957412A4120AC	
Speed	Dual-Port 10 Gb/s Ethernet
PCIe	Gen 3 x8
	The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.
Interface	SFP+ for 10 Gb/s
Device	Broadcom BCM57412 10 Gb/s MAC controller with integrated dual-channel 10 Gb/s SFI transceiver.
NDIS Name	Broadcom P210p NetXtreme-E Dual-Port 10Gb Ethernet PCIe Adapter
UEFI Name	Broadcom P210p NetXtreme-E Dual-Port 10Gb Ethernet PCIe Adapter
BCM957412N4120C	
Speed	Dual-Port 10 Gb/s Ethernet
PCIe	Gen3 x8
	The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.
Interface	SFP+ for 10 Gb/s
Device	Broadcom BCM57412 10 Gb/s MAC controller with Integrated dual-channel 10 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E Series Dual-Port 10Gb SFP+ Ethernet OCP 3.0 Adapter
UEFI Name	Broadcom NetXtreme-E Dual 10Gb SFP+ OCP 3.0 Ethernet
BCM957414A4142CC	
Speed	Dual-Port 25 Gb/s Ethernet
PCIe	Gen 3 x8
	The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.
Interface	SFP28 for 25 Gb/s
Device	Broadcom BCM57414 25 Gb/s MAC controller with integrated dual-channel 25 Gb/s SFI transceiver.
NDIS Name	Broadcom P225p NetXtreme-E Dual-Port 10Gb/25Gb Ethernet PCIe Adapter
UEFI Name	Broadcom P225p NetXtreme-E Dual-Port 10Gb/25Gb Ethernet PCIe Adapter
BCM957414N4140C	
Speed	Dual-Port 25 Gb/s
PCIe	Gen 3 x8
	The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.
Interface	SFP28 for 25 Gb/s
Device	Broadcom BCM57414 25 Gb/s MAC controller with integrated dual-channel 25 Gb/s SFI transceiver.

NDIS Name	Broadcom NetXtreme-E Series Dual-Port 25Gb SFP28 Ethernet OCP 3.0 Adapter
UEFI Name	Broadcom NetXtreme-E Dual 25Gb SFP28 OCP 3.0 Ethernet
BCM957416A4160C	
Speed	Dual-Port 10GBASE-T Ethernet
PCIe	Gen 3 x8
	The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.
Interface	RJ-45 for 10 Gb/s
Device	Broadcom BCM57416 10 Gb/s MAC controller with integrated dual-channel 10GBASE-T transceiver.
NDIS Name	Broadcom NetXtreme-E Series Dual-Port 10GBASE-T Ethernet PCIe Adapter
UEFI Name	Broadcom Dual 10GBASE-T Ethernet
BCM957416N4160C	
Speed	Dual-Port 10GBASE-T Ethernet
PCIe	Gen 3 x8
	The NIC supports PCIe 3, 2, and 1 speeds, however, PCIe is recommended to achieve nominal throughput when 2 ports of 25G links transmit and receive traffic at the same time.
Interface	RJ-45 for 10 Gb/s
Device	Broadcom BCM57416 10 Gb/s MAC controller with integrated dual-channel 10GBASE-T transceiver.
NDIS Name	Broadcom NetXtreme-E Series Dual-Port 10GBASE-T Ethernet OCP 3.0 Adapter
UEFI Name	Broadcom NetXtreme-E 2Px10GBASE-T OCP 3.0 Ethernet
BCM957504-N425G	
Speed	Quad-Port 25Gb/s Ethernet
PCIe	Gen4 x16
Interface	SFP 28 for 25 Gb/s
Device	Broadcom BCM57504 25G Gb/s MAC controller with integrated quad-channel 25 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E Series Quad-Port 25Gb SFP28 OCP 3.0 Ethernet Adapter
UEFI Name	Broadcom NetXtreme-E Quad 25Gb SFP28 OCP 3.0 Ethernet
BCM957504-N1100G	
Speed	Single-Port 100 Gb/s Ethernet
PCIe	Gen4 x16
Interface	QSFP28 for 100 Gb/s
Device	Broadcom BCM57504 100 Gb/s MAC controller with integrated single-channel 100 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E Series Single Port 100Gb OCP 3.0 Ethernet Adapter
UEFI Name	Broadcom NetXtreme-E Single 100-Gb OCP 3.0 Ethernet
BCM957504-M1100G16	
Speed	Single-Port 100 Gb/s Ethernet
PCIe	Gen4 x16
Interface	QSFP28 for 100 Gb/s

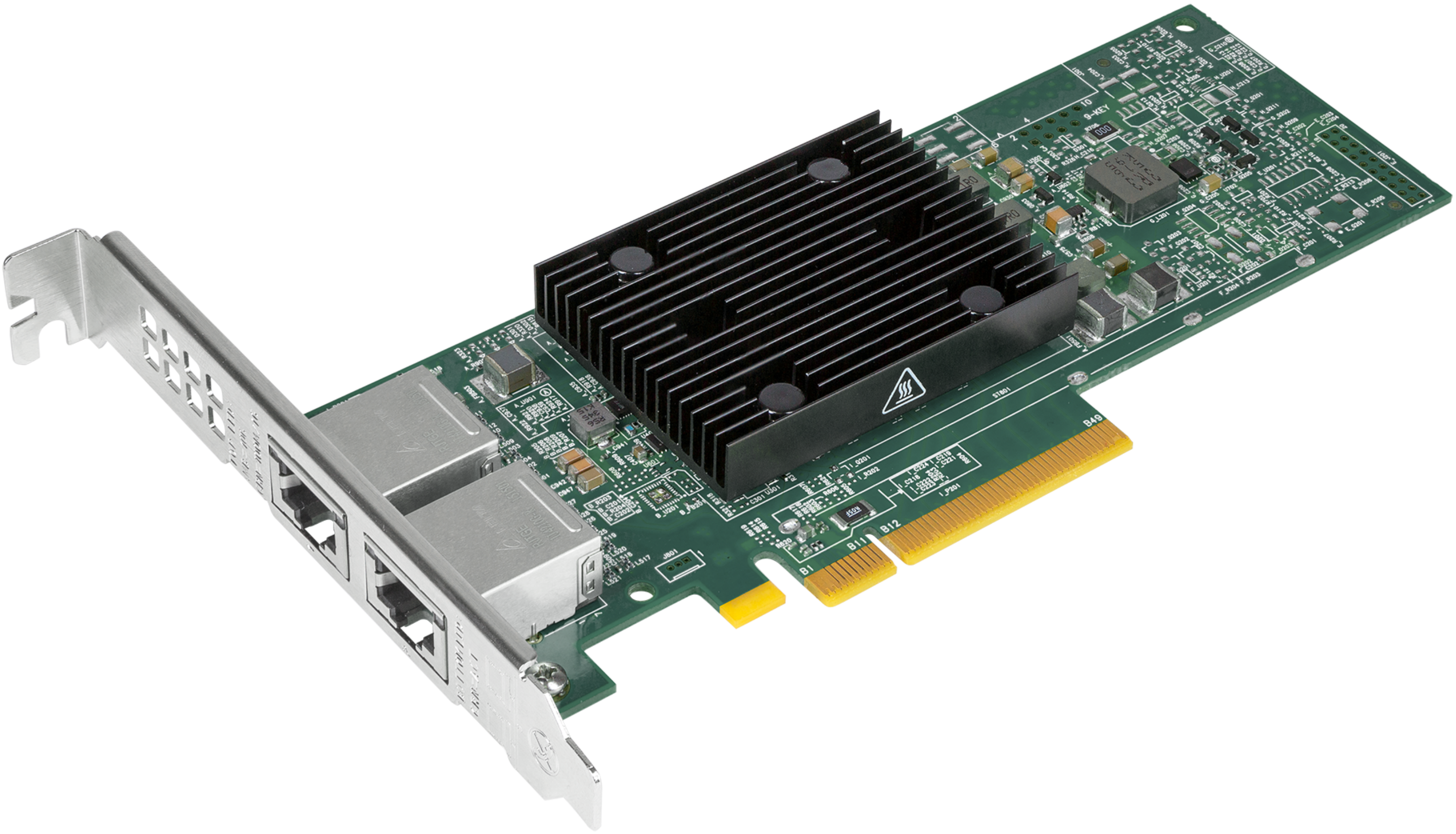
Device	BCM57504 100 Gb/s MAC controller with integrated single-channel 100 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme-E Series Single Port 100Gb OCP Ethernet Adapter
UEFI Name	Broadcom NetXtreme-E Single 100Gb OCP Ethernet
BCM957504-P425G	
Speed	Quad-Port 25 Gb/s Ethernet
PCIe	Gen4 x16
Interface	SFP28 for 25 Gb/s
Device	Broadcom BCM57504 25G Gb/s MAC controller with integrated quad-channel 25 Gb/s SFI transceiver.
NDIS Name	Broadcom NetXtreme E-Series Quad-port 25Gb SFP28 PCIe Ethernet Adapter
UEFI Name	Broadcom NetXtreme-E Quad 25Gb SFP28 PCIe Ethernet
BCM957508-P2100G	
Speed	Dual-Port 100 Gb/s Ethernet
PCIe	Gen4 x16
Interface	QSFP28 for 100 Gb/s
Device	Broadcom BCM57508 100 Gb/s MAC controller with integrated dual-channel 100 Gb/s SFI transceiver.
NDIS Name	Broadcom P2100G NetXtreme-E zDual-Port 100Gb Ethernet PCIe Adapter
UEFI Name	Broadcom P2100G NetXtreme-E Dual 100Gb PCIe Ethernet

Figure 1: BCM957412A4120AC/BCM957414A4142CC Network Interface Card



NOTE
The previous figure shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receive. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

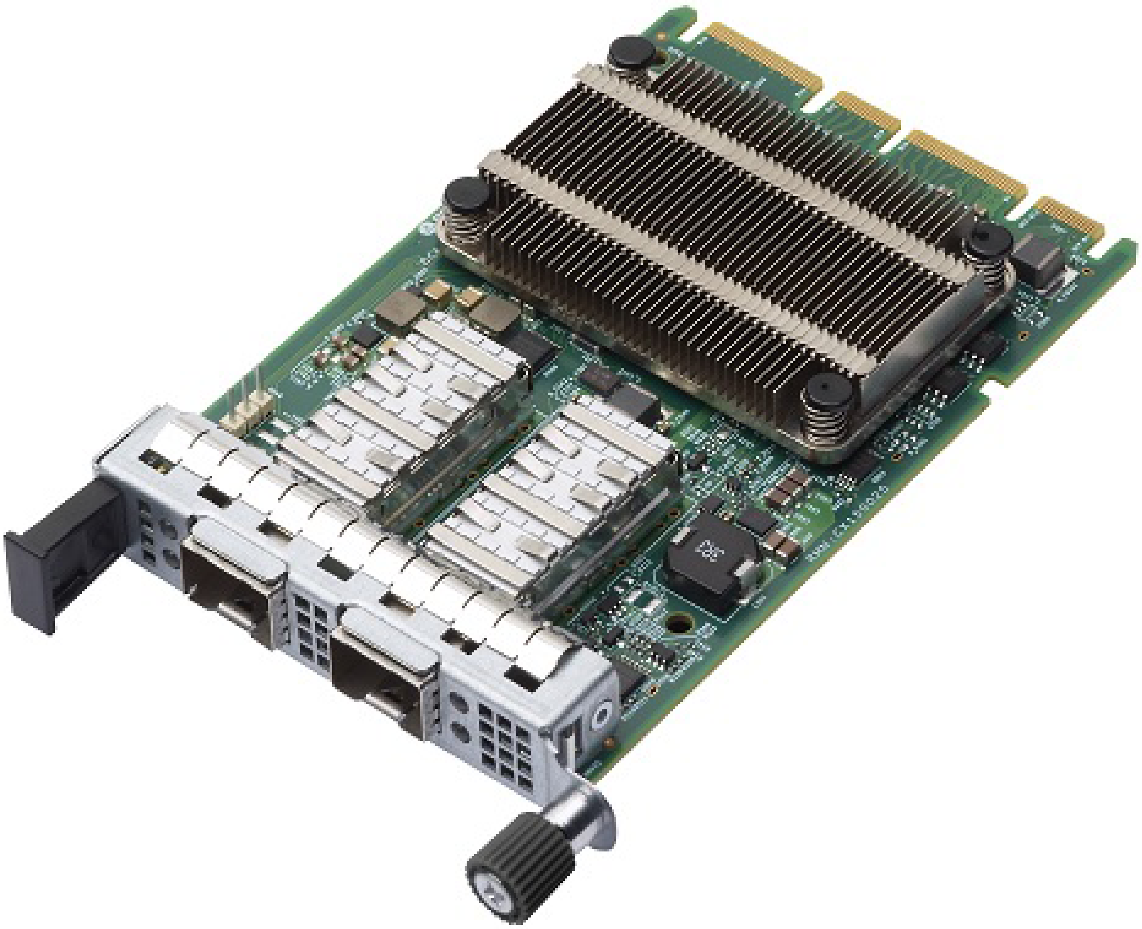
Figure 2: BCM957416A4160C Network Interface Card



NOTE

The previous figure shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

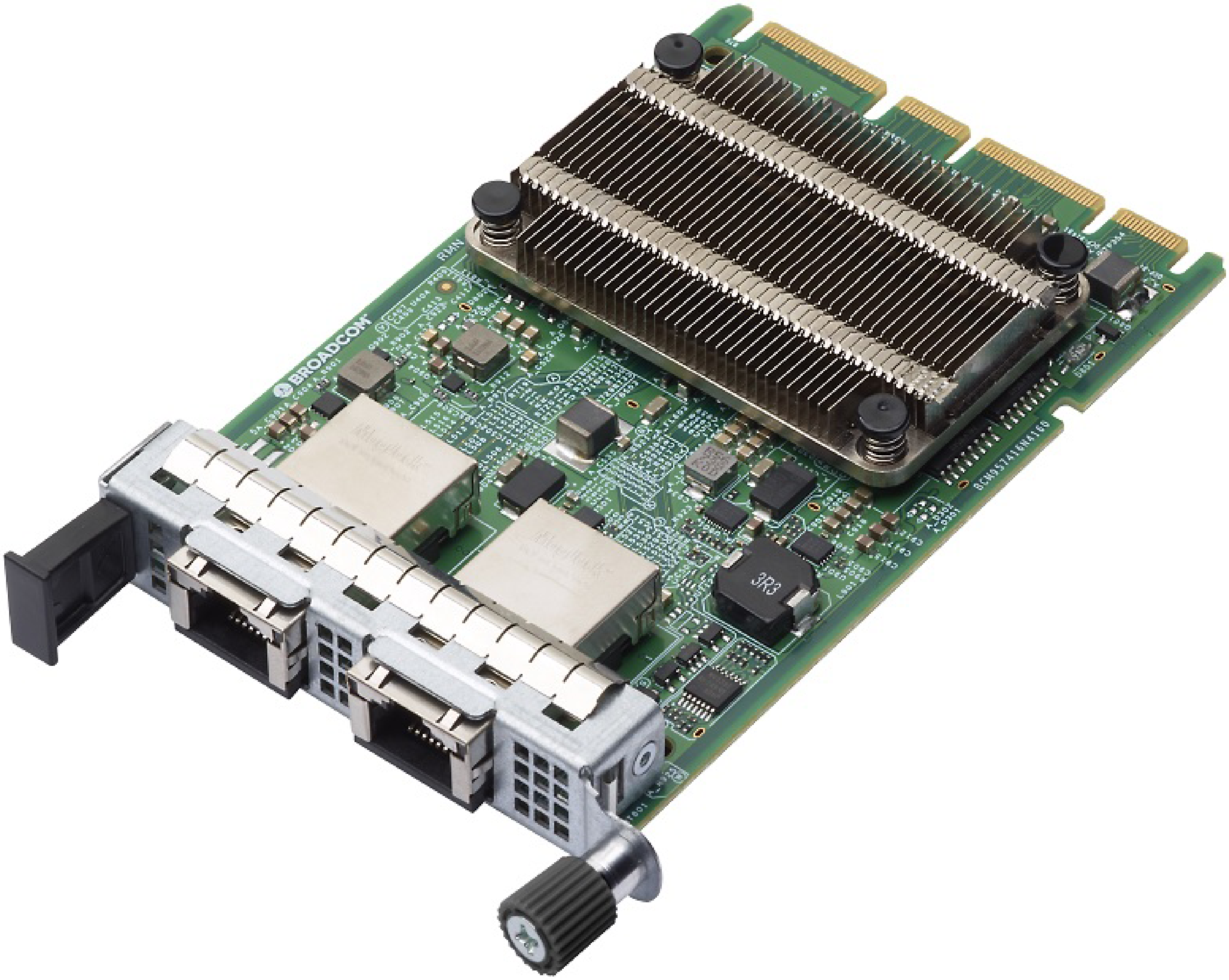
Figure 3: BCM957412N4120/BCM957414N4140C Small-Form-Factor Network Adapter



NOTE

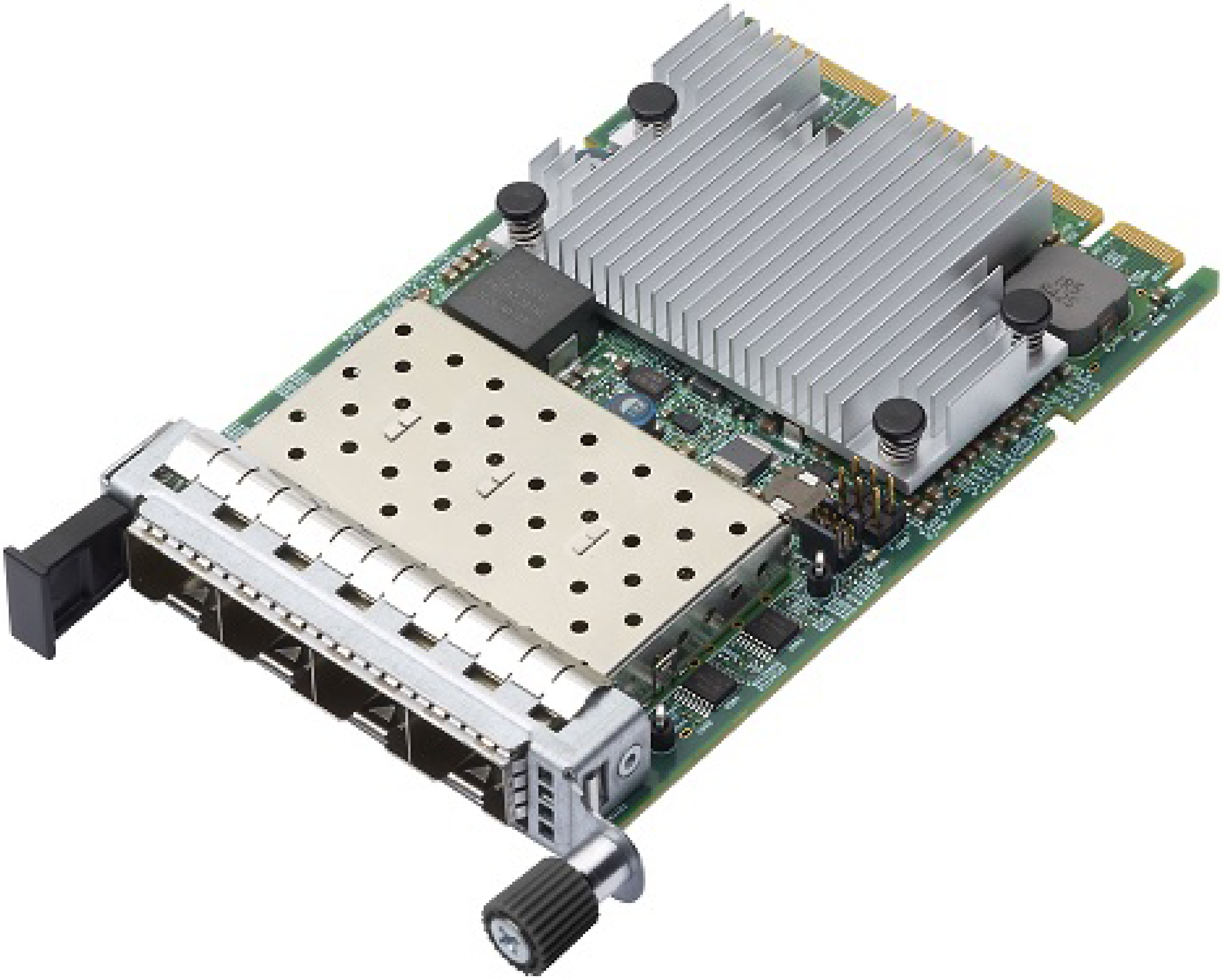
The previous figure shows the pull-tab faceplate installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Figure 4: BCM957416N4160C Small-Form-Factor Network Adapter



NOTE
The previous figure shows the pull-tab faceplate installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Figure 5: BCM957504-N425G OCP 3.0 SFF Card



NOTE
The previous figure shows the pull-tab faceplate installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Figure 6: BCM957504-P425G Network Interface Card

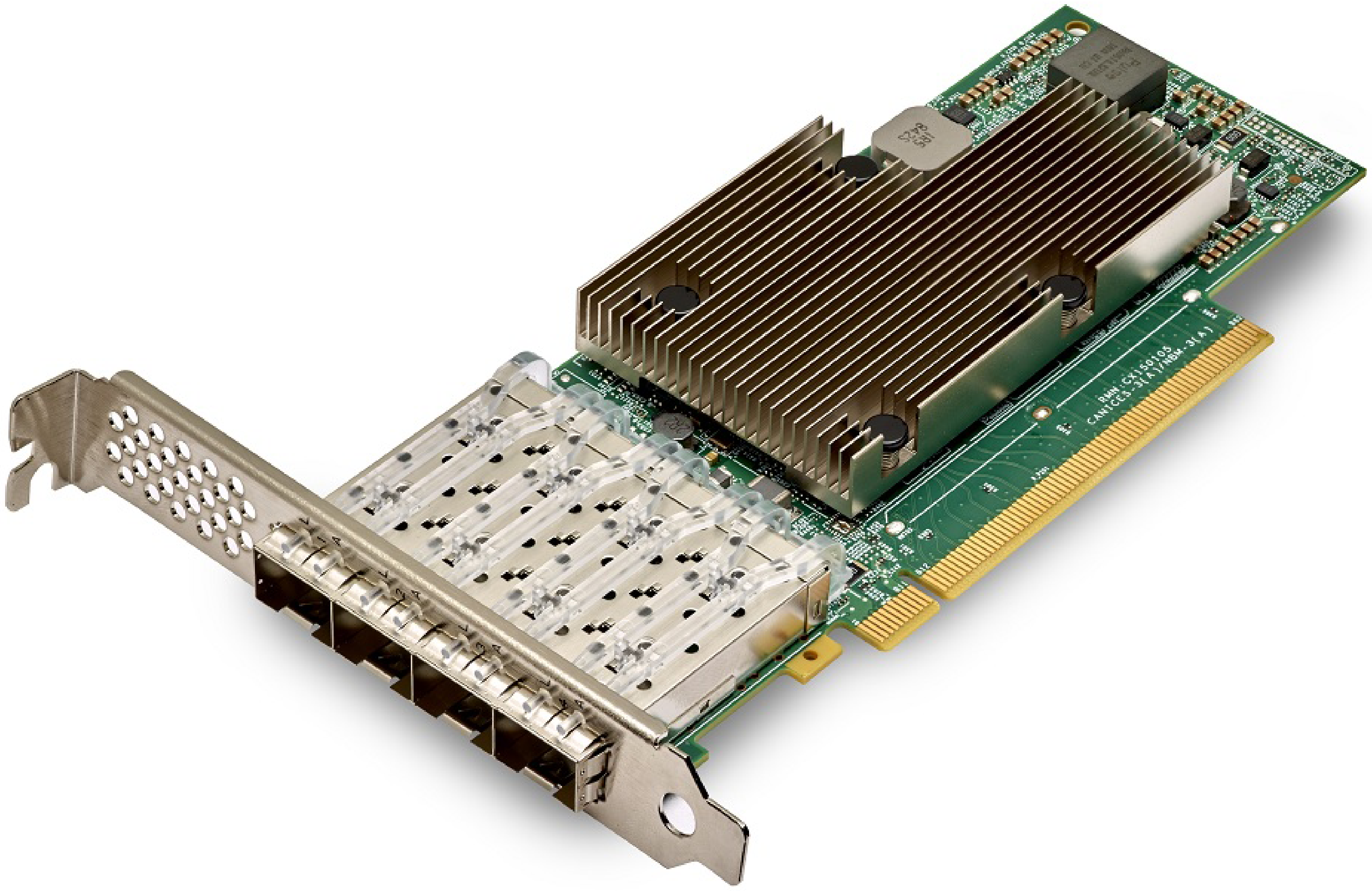


Figure 7: BCM957508-P2100G Network Interface Card

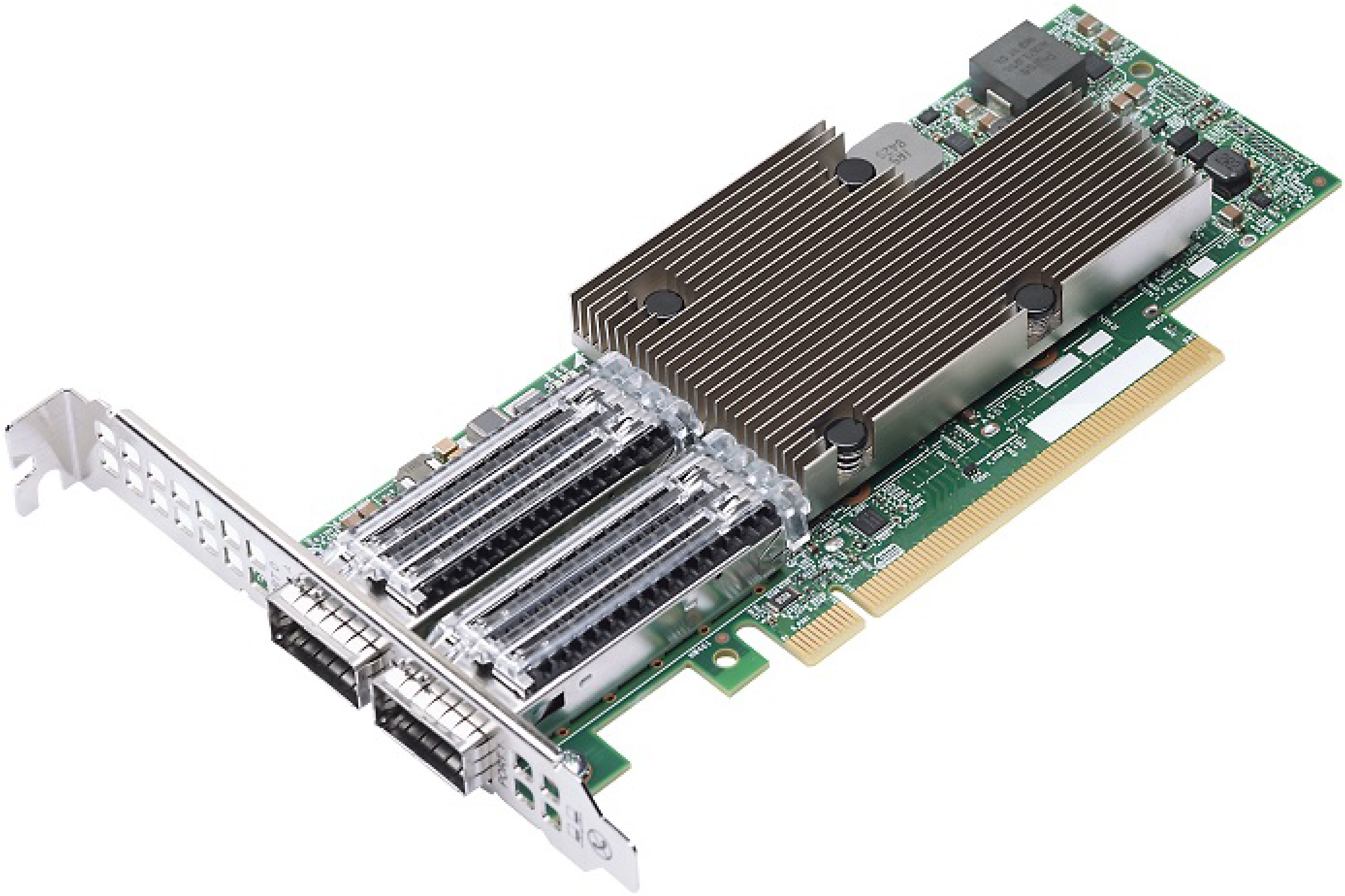
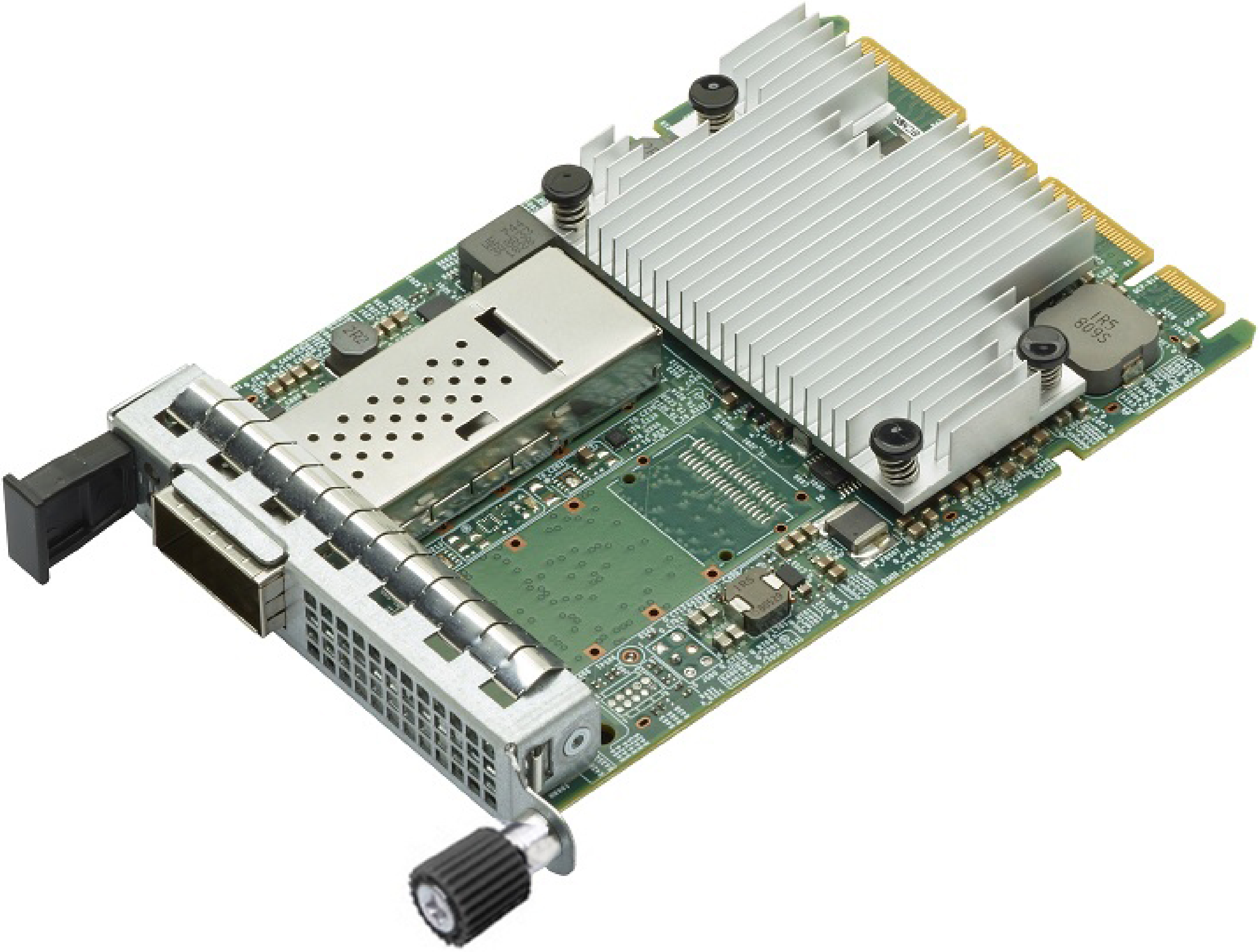
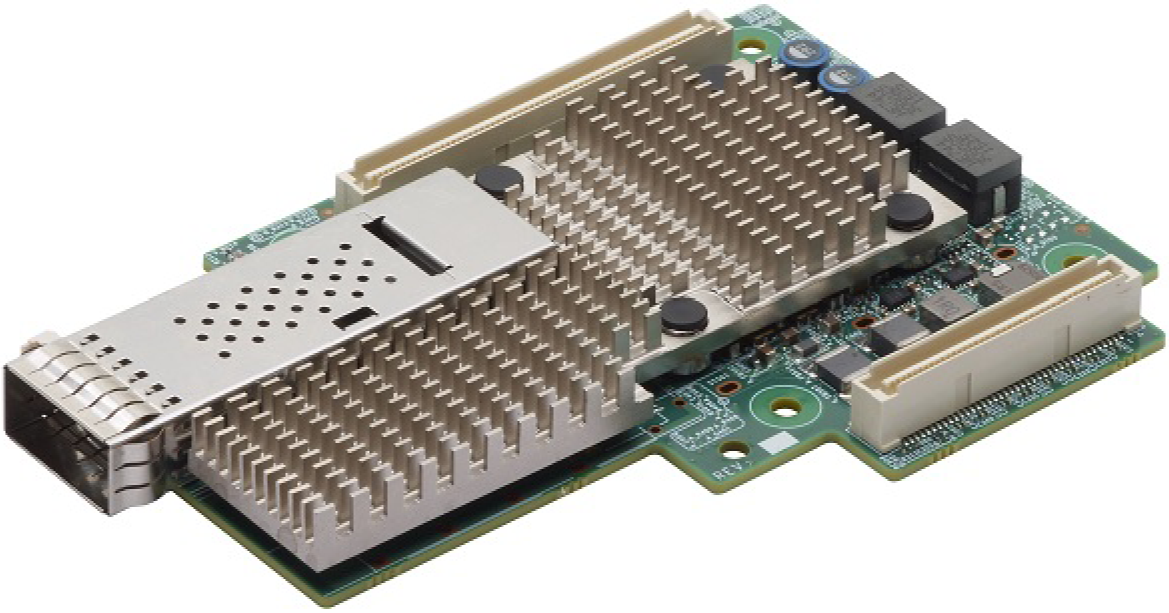


Figure 8: BCM957504-N1100G OCP 3.0 SFF Network Adapter



NOTE
The previous figure shows the pull-tab bracket installed by default. The surface markings of the component may not reflect the product received. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Figure 9: BCM957504-M1100G16 OCP Mezzanine Card



NOTE
The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Network Link and Activity Indication

Ethernet connections, the state of the network link, and activity are indicated by the LEDs on the rear connector. See the following sections for the individual boards:

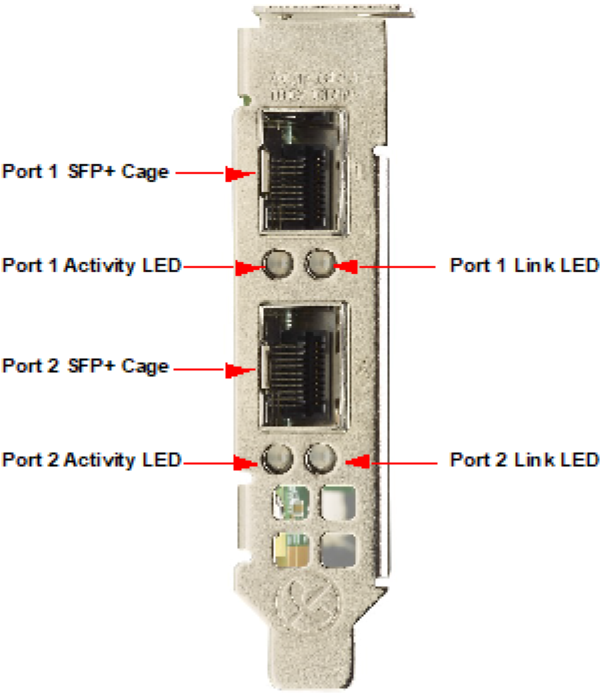
- BCM9574120AC
- BCM957414A4142CC
- BCM957416A4160C
- BCM957412N4120C
- BCM957414N4140C
- BCM957416N4160C
- BCM957504-N425G
- BCM957504-N1100G
- BCM957504-M1100G16
- BCM957504-P425G
- BCM957508-P2100G

NOTE
See the individual board data sheets for specific media design.

BCM957412A4120AC

The SFP+ port has two LEDs to indicate traffic activities and link speed. The LEDs are shown in the following figure and described in the following table.

Figure 10: BCM957412A4120AC Activity and Link LED Locations



NOTE
The previous figure shows the low-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

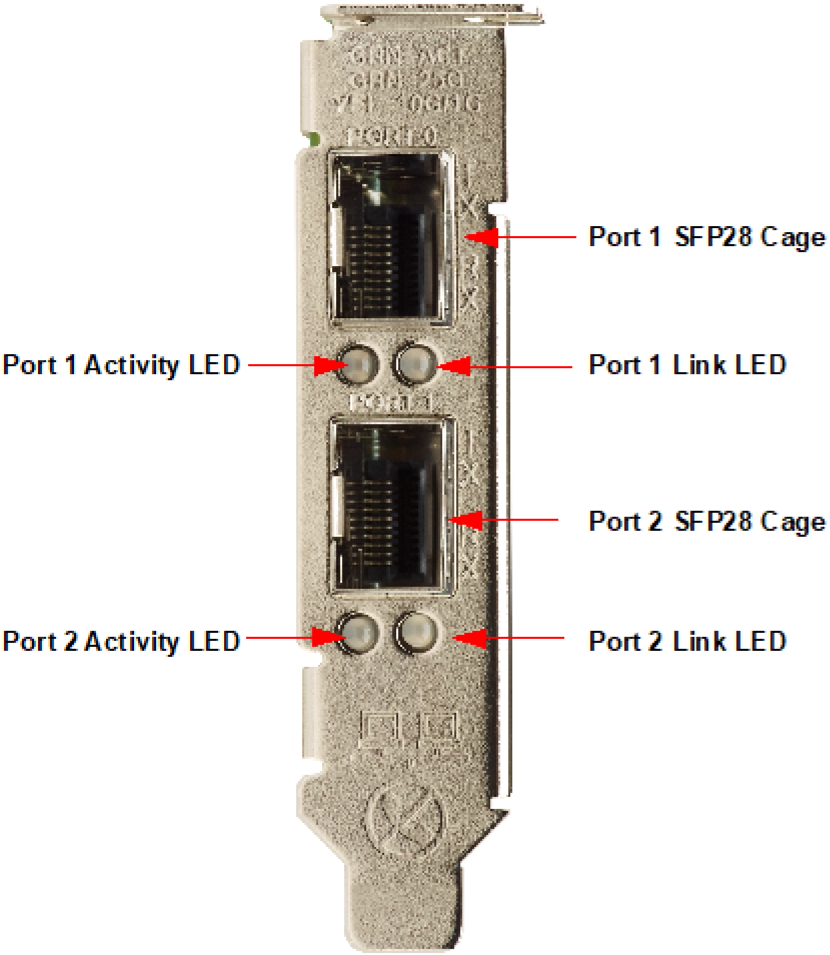
Table 4: BCM957412A4120AC Activity and Link LED Locations

LED Type	Color/Behavior	Note
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Yellow	Linked at 1 Gb/s

BCM957414A4142CC

The SFP28 port has two LEDs to indicate traffic activities and link speed. The LEDs are shown in the following figure and described in the following table.

Figure 11: BCM957414A4142CC Activity and Link LED Locations



NOTE
The previous figure shows the low-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

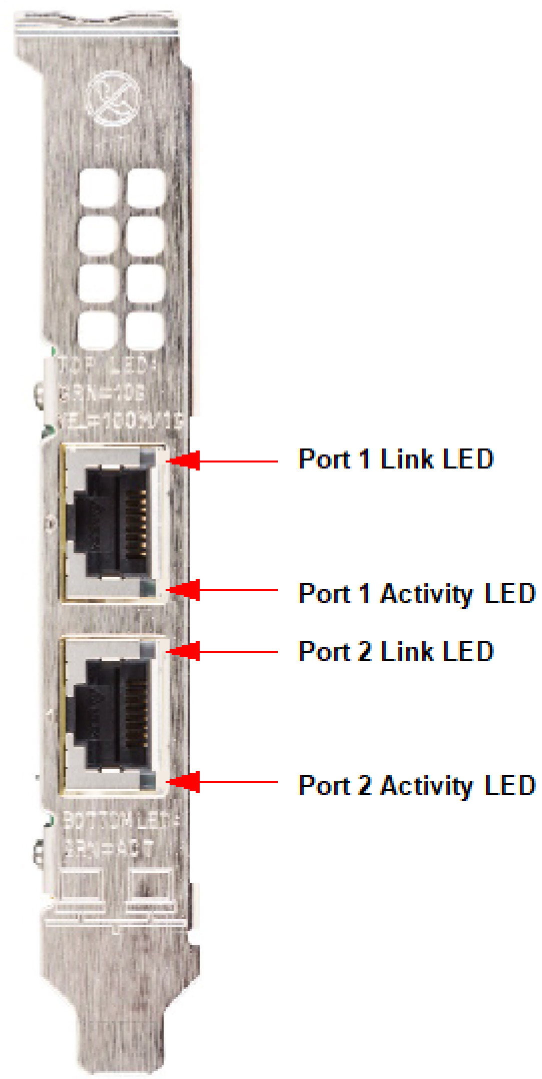
Table 5: BCM957414A4142CC Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Activity
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Yellow	Linked at 10 Gb/s

BCM957416A4160C

Each Ethernet interface has a link LED to indicate Link status and an activity LED to indicate data traffic. The LEDs are shown in the following figure and described in the following table.

Figure 12: BCM957416A4160C Activity and Link LED Locations



NOTE
The previous figure shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Table 6: BCM957416A4160C Activity and Link LED Functions

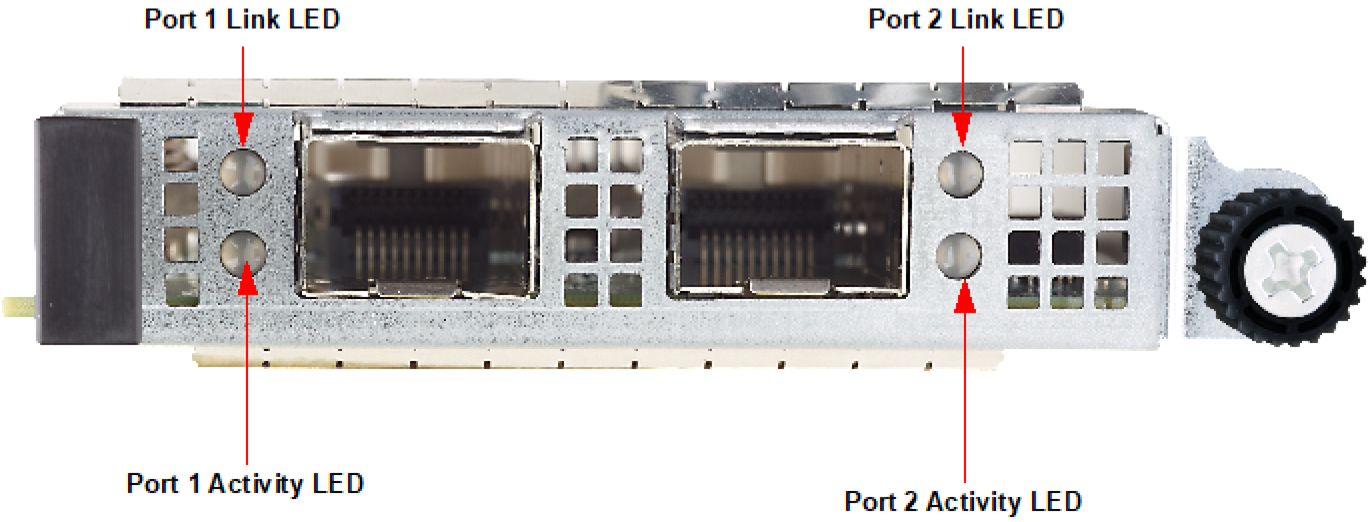
LED Type	Color/Behavior	Notes
Activity	Off	No Activity

LED Type	Color/Behavior	Notes
	Green blinking	Traffic Flowing Activity
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Amber	Linked at 1 Gb/s

BCM957412N4120C

The SFP+ port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in the following figure. Its locations and form factors conform to the OCP 3.0 Design Specification.

Figure 13: BCM957412N4120C Activity and Link LED Locations



NOTE
The previous figure shows the pull-tab bracket installed by default. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

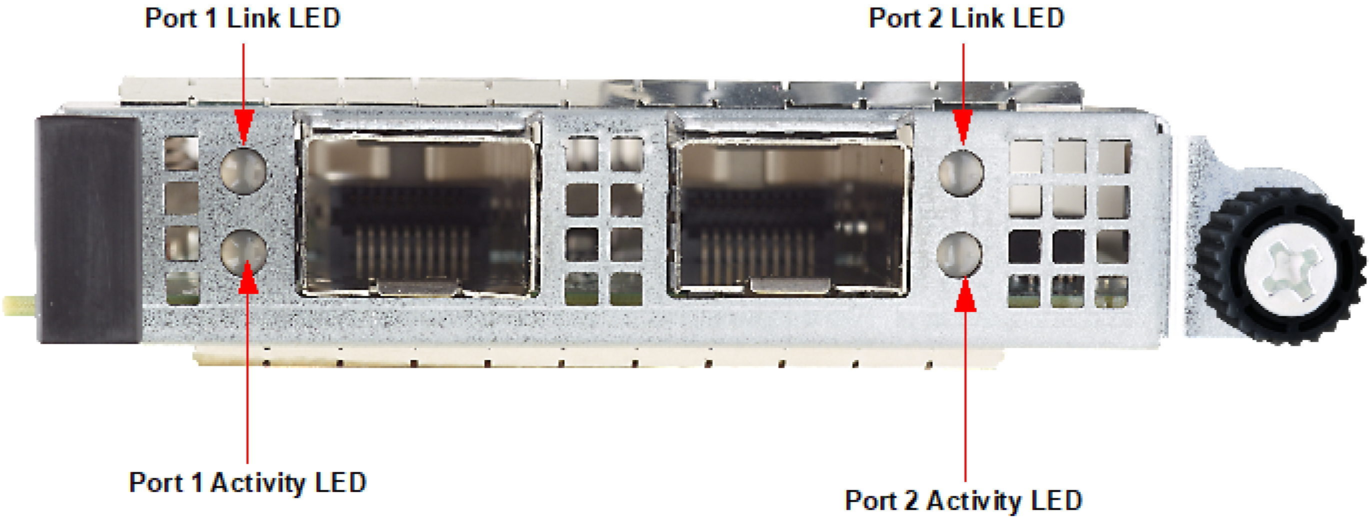
Table 7: BCM957412N4120C Network Adapter Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Amber	Linked at 1 Gb/s

BCM957414N4140C

The SFP28 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in the following figure. Its locations and form factors conform to the OCP 3.0 Design Specification.

Figure 14: BCM957414N4140C Activity and Link LED Locations



NOTE
The previous figure shows the pull-tab bracket installed by default. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

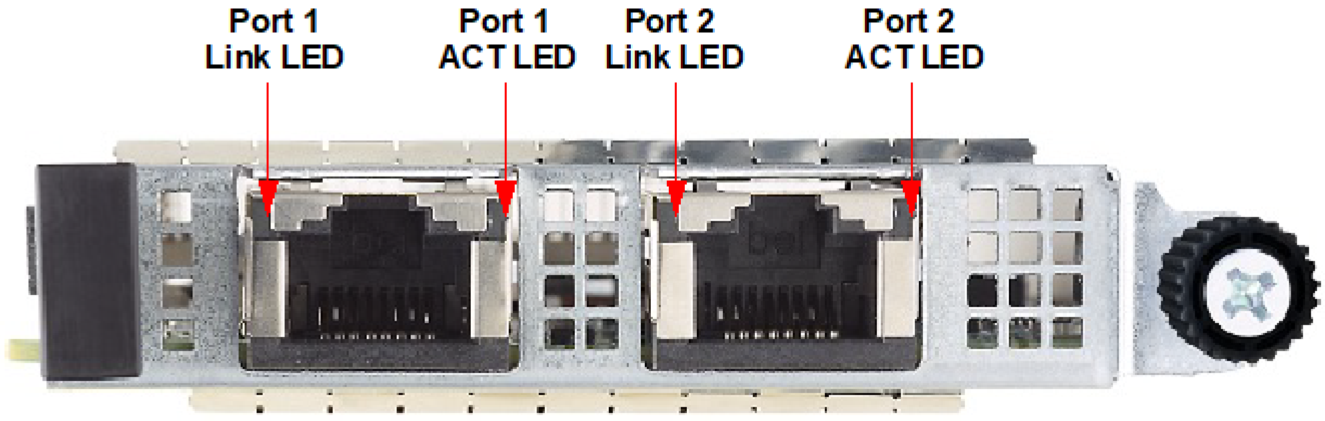
Table 8: BCM957414N4140C Network Adapter Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Amber	Linked at 10 Gb/s or 1 Gb/s

BCM957416N4160C

he RJ-45 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in the following figure. Its locations and form factors conform to the OCP 3.0 Design Specification.

Figure 15: BCM957416N4160C Activity and Link LED Locations



NOTE
The previous figure shows the pull-tab bracket installed by default. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

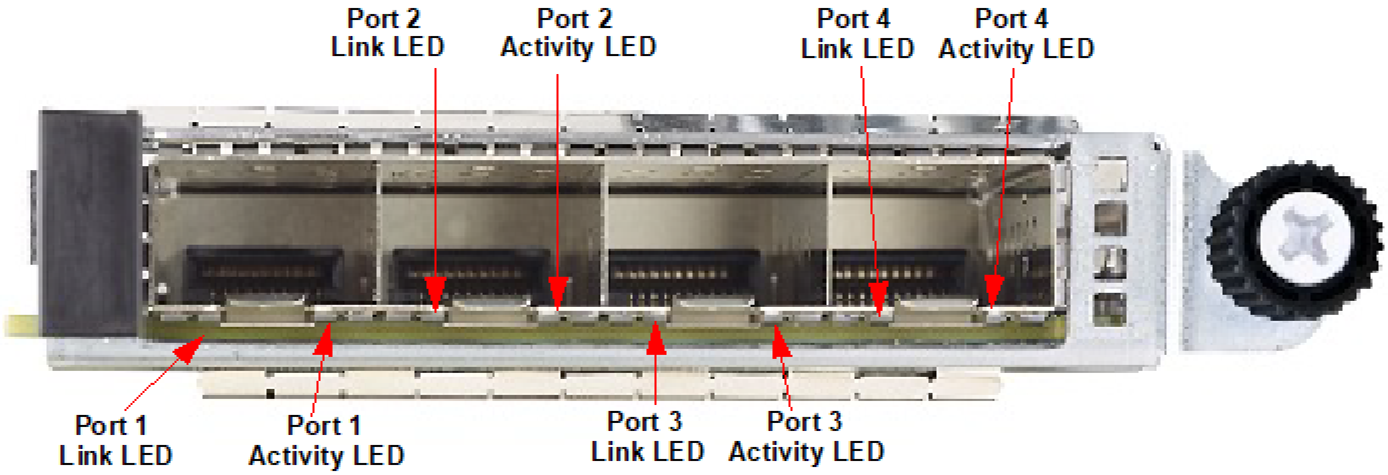
Table 9: BCM957416N4160C Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 10 Gb/s
	Amber	Linked at 1 Gb/s

BCM957504-N425G

he SFP28 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in the following figure. Its locations and form factors conform to the OCP 3.0 Design Specification. The LED functionality is described in the following table.

Figure 16: BCM957504-N425G Activity and Link LED Locations



NOTE
The previous figure shows the pull-tab bracket installed by default. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

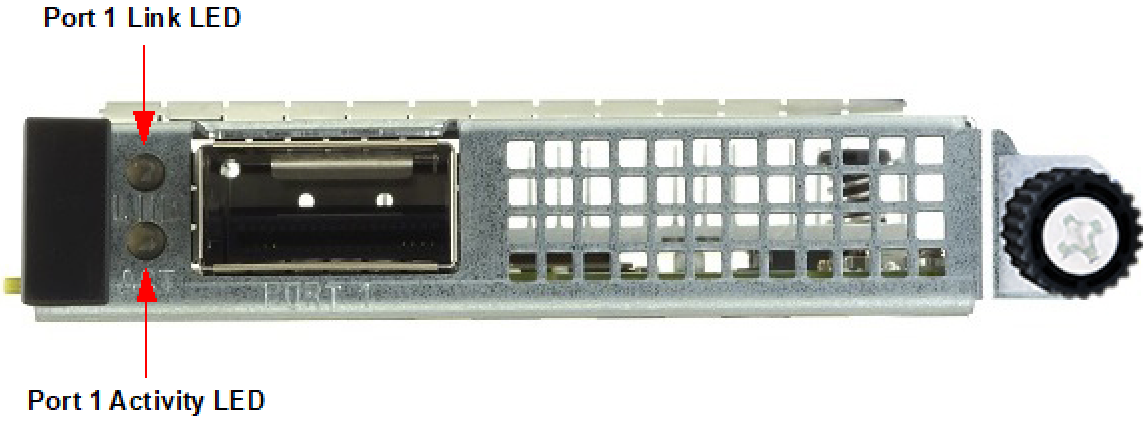
Table 10: BCM957504-N425G Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Amber	Linked at lower speed

BCM957504-N1100G

The QSFP56 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible as shown in the following figure. Its locations and form factors conform to the OCP 3.0 Design Specification. The LED functionality is described in the following table.

Figure 17: BCM957504-N1100G Activity and Link LED Locations



NOTE
The previous shows the pull-tab bracket installed by default. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

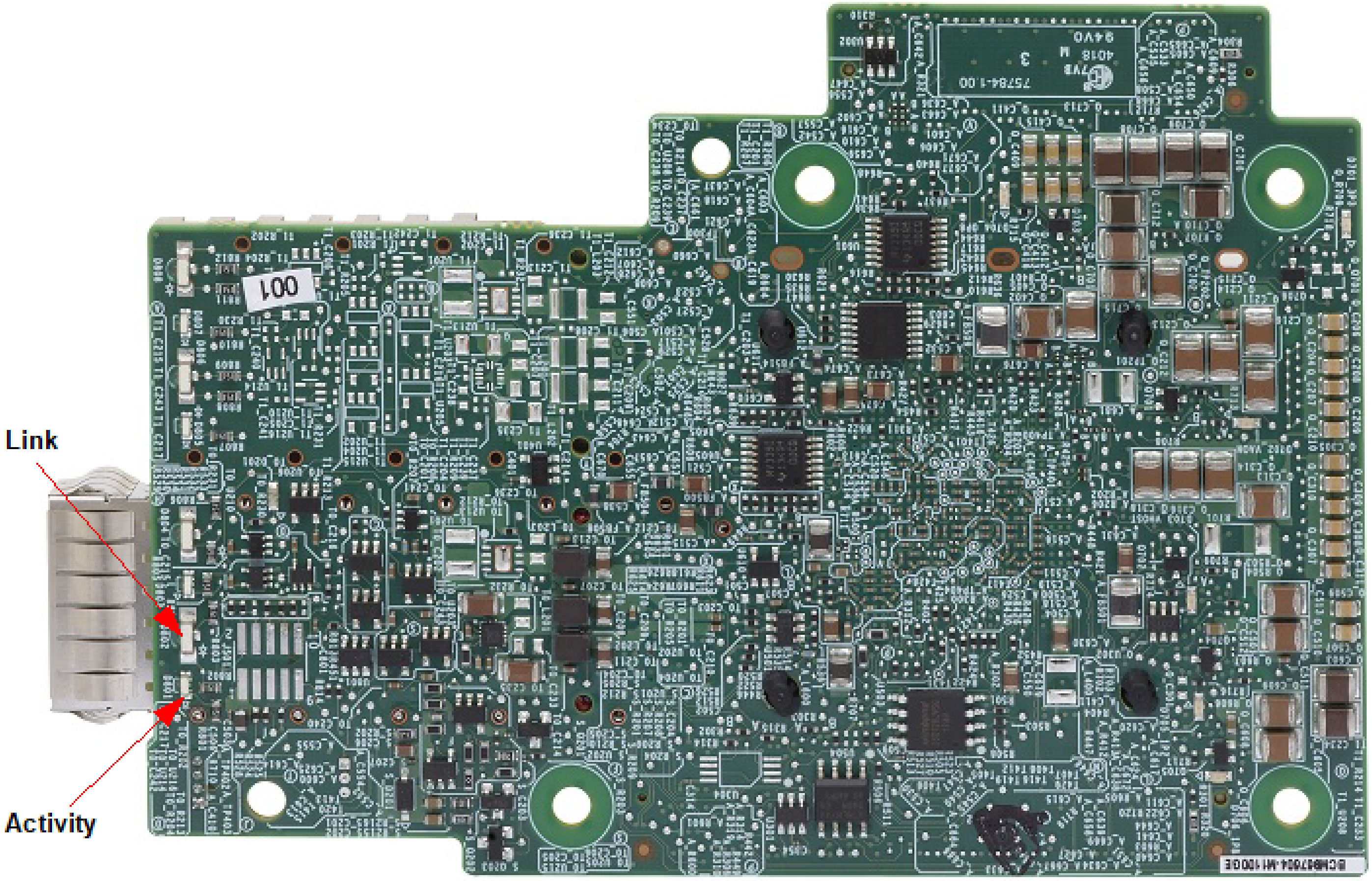
Table 11: BCM957504-N1100G Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 100 Gb/s
	Amber	Linked at lower speed

BCM957504-M1100G16

The QSFP56 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible on the bottom side as shown in the following figure. Their locations and form factors conform to the OCP mezzanine card specification. The LED functionality is described in the following table.

Figure 18: BCM957504-M1100G16 Activity and Link LED Locations



NOTE
The surface markings of the components shown in the previous figure may not reflect the final product. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

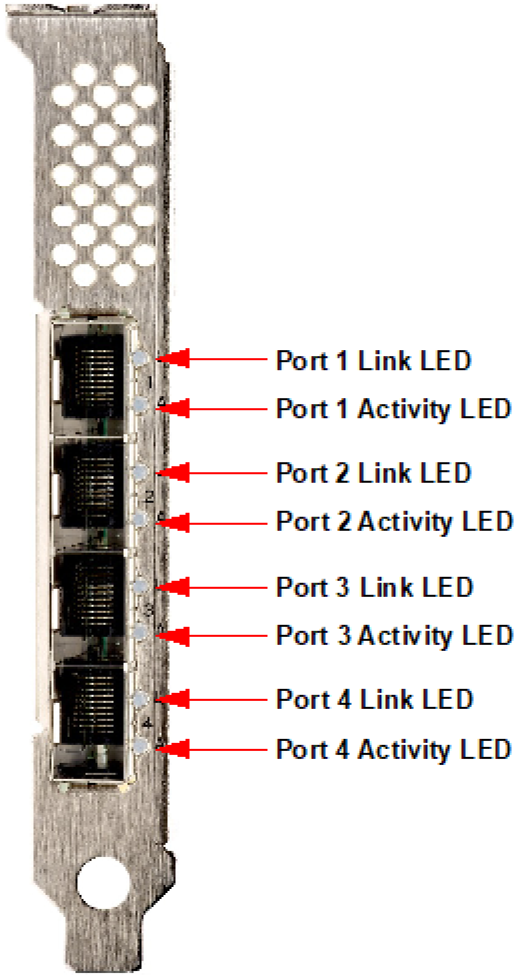
Table 12: BCM957504-M1100G16 Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 100 Gb/s
	Amber	Linked at lower speed

BCM957504-P425G

The SFP28 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible through the cutout on the bracket as shown in the following figure.

Figure 19: BCM957504-P425G Activity and Link LED Locations



NOTE
The previous figure shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

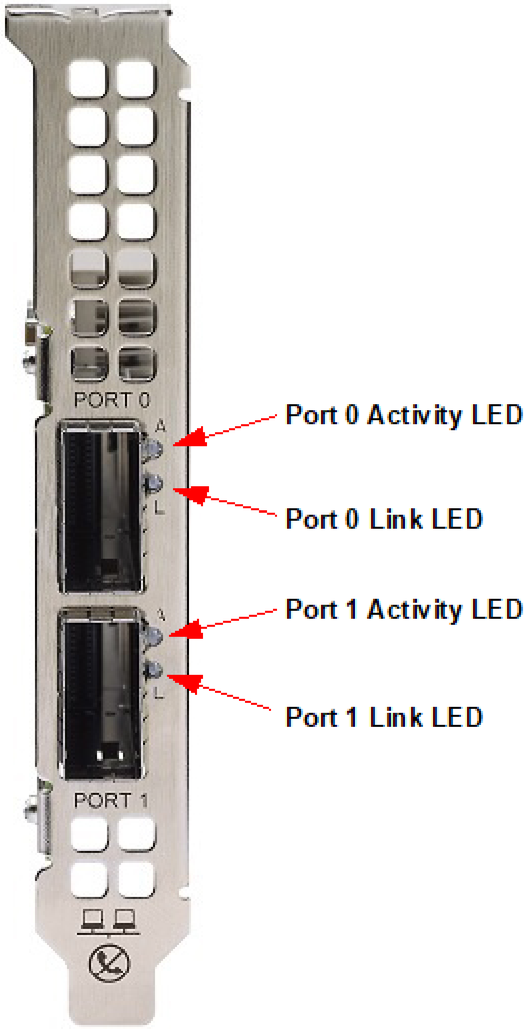
Table 13: BCM957504-P425G Activity and Link LED Functions

LED Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link
	Green	Linked at 25 Gb/s
	Amber	Linked at lower speed

BCM957508-P2100G

The QSFP56 port supports two LEDs to indicate traffic activities and link speed. The LEDs are visible through the cutout on the bracket as shown in the following figure. The LED functionality is described in the following table.

Figure 20: BCM957508-P2100G Activity and Link LED Locations



NOTE
The previous figure shows the standard-profile bracket installed. The surface markings of the component may not reflect the product upon receipt. Broadcom reserves the right to change any component on the printed circuit board with the same functionality.

Table 14: BCM957508-P2100G Activity and Link LED Functions

LET Type	Color/Behavior	Note
Activity	Off	No Link
	Green (blinking)	Link up (traffic flowing)
Link	Off	No Link

LET Type	Color/Behavior	Note
	Green	Linked at 100 Gb/s
	Amber	Linked at lower speed

Features

See the following sections for device features:

- Virtualization Features
- VXLAN
- NVGRE/GRE/IP-in-IP/Geneve
- Stateless Offloads
- Priority Flow Control
- Virtualization Offload
- SR-IOV
- Network Partitioning (NPAR)
- Security (BCM575XX Only)
- Supported Combinations
- Unsupported Combinations
- RDMA over Converged Ethernet – RoCE

Virtualization Features

The following table lists the virtualization features of the NetXtreme-E.

Table 15: Virtualization Features

Feature	Details
Linux KVM Multiqueue	Supported.
VMware NetQueue	Supported.
NDIS Virtual Machine Queue (VMQ)	Supported.
Virtual eXtensible LAN (VXLAN) – Aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, NetQueue, VMQ, RSS, TCP segmentation offload, Large Send Offload, Generic Receive Offload).	Supported.
Generic Routing Encapsulation (GRE) – Aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, VMQ, RSS, TCP segmentation offload, Generic Receive Offload).	Supported.
Network Virtualization using Generic Routing Encapsulation (NVGRE) – Aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, VMQ, RSS, Large Send Offload).	Supported.
Generic Network Virtualization Encapsulation (Geneve) – Aware Stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/ removal, NetQueue, RSS, TCP segmentation offload, Generic Receive Offload).	Supported.
IP-in-IP aware stateless offloads (IP/UDP/TCP checksum offloads, VLAN insertion/removal, NetQueue, RSS, TCP segmentation offload, Generic Receive Offload).	Supported.

SR-IOV v1.0	BCM9574XX – 128 Virtual Functions (VFs) for Guest Operating Systems (GOS) per device. BCM9575XX – 1K Virtual Functions (VFs) for Guest Operating Systems (GOS) per device.
Edge Virtual Bridging (EVB) (IEEE 802.1Qbg)	BCM9575XX – Edge Virtual Bridging (EVB) enables switching of traffic between PFs/VFs, forwarding of outgoing network traffic from PFs/VFs to appropriate network ports, and steering of incoming network traffic to appropriate PFs/VFs. Both VEB (local switching in the NIC) and VEPA (switching in the adjacent switch) EVB modes of operation are supported. The EVB features supported are: • 1K VFs and up to 128 queues per VF (flexible allocation across PFs/VFs). • PCIe AER, TPH, FLR support. • Virtual Ethernet Bridge (VEB)/Virtual Ethernet Port Aggregator (VEPA). • MAC/VLAN filtering and mirroring. • VF isolation, source pruning, anti-spoofing checks. • Stateless and packet steering offloads per VF. • Forwarding of unicast frames based on {Tunnel ID (optional), Destination MAC, VLAN ID (optional)}. • Frame replication for multicast, broadcast, and promiscuous mode. • Source pruning – Provide support for source knockout (prevent sending a multicast or broadcast frame back to the source). • Mirroring of traffic to a specific PF or VF. • Packet editing – VLAN insert/swap/delete. • Anti-spoof checks.
MSI-X vector port	74 per port default value (two port configuration). 16 per VF and is configurable in HII and CCM.

VXLAN

A Virtual eXtensible Local Area Network (VXLAN), defined in IETF RFC 7348, is used to address the need for overlay networks within virtualized data centers accommodating multiple tenants. VXLAN is a Layer 2 overlay or tunneling scheme over a Layer 3 network. Only VMs within the same VXLAN segment can communicate with each other.

NVGRE/GRE/IP-in-IP/Geneve

Network Virtualization using GRE (NVGRE), defined in IETF RFC 7637, is similar to a VXLAN.

NOTE
Checksum offload must be enabled when using NVGRE.

Stateless Offloads

This section contains the following information on stateless offloads:

- IP, TCP, UDP Checksum Offload
- UDP Fragmentation Offload
- TCP Segmentation Offload and Large Send Offload
- Generic Receive Offload (GRO) and Large Receive Offload (LRO)
- Header and Data Split
- VLAN Tag Insertion and Removal
- Packet Steering
- Data Center Bridging

IP, TCP, UDP Checksum Offload

Host software can configure the Ethernet controller to calculate IP, TCP, and UDP checksums as described in RFC 791, RFC 793, and RFC 768 respectively. The first step in checksum calculation is determining the start of an IP and UDP datagram and TCP segment within a frame, which could vary depending on whether the frame is tagged (VLAN) or encapsulated with an LLC/SNAP header. Then the checksum is computed from the start to the end of the datagram and inserted into the appropriate location in the protocol header. The Ethernet controller is designed to support checksum calculation on all frame types and also on IP datagram and TCP segments containing options.

UDP Fragmentation Offload

UDP Fragmentation Offload (UFO) is a feature that enables the software stack to offload fragmentation of large UDP/IP datagrams into multiple UDP/IP packets of size suitable for transmission. Enabling UFO can result in reduced CPU load for UDP applications. Support for this feature is only available in the Linux environment.

TCP Segmentation Offload and Large Send Offload

Large Segment Offload (LSO) is a feature that enables the software stack to offload segmentation of large TCP messages into multiple TCP/IP packets of size suitable for transmission. Enabling LSO can result in reduced CPU load for TCP applications. This is also called TCP Segmentation Offload (TSO).

Generic Receive Offload (GRO) and Large Receive Offload (LRO)

Generic Receive Offload (GRO) and Large Receive Offload (LRO) are hardware acceleration for TCP data reception. Both GRO and LRO modes of TCP receive offload are supported by the Ethernet Controller's Transparent Packet Aggregation (TPA) feature. Enabling GRO and LRO can significantly reduce CPU load and increase throughput for TCP applications by reducing the number of received messages, interrupts, and DMA operations. TPA aggregates TCP streams by managing context entries. Each entry in the TPA context is identified by the 4-tuple: Source IP, destination IP, source TCP port, and destination TCP port. GRO is the preferred TPA mode as packet boundaries are preserved for network routing applications, which may enable LSO for transmission.

Header and Data Split

Header-payload split is a feature that enables the software TCP/IP stack to receive TCP/IP packets with header and payload data split into separate buffers. The support for this feature is available in both Windows and Linux environments. The following are potential benefits of header-payload split:

- The header-payload split enables compact and efficient caching of packet headers into host CPU caches. This can result in a receive side TCP/IP performance improvement.
- Header-payload splitting enables page flipping and zero copy operations by the host TCP/IP stack. This can further improve the performance of the receive path.

VLAN Tag Insertion and Removal

On the TX Path, the Ethernet controller is capable of inserting IEEE 802.1Q-compliant VLAN tags into transmitted frames and extracting the VLAN tags from received frames. On the RX path, receiving VLAN-tagged (IEEE 802.1q-compliant) packets is supported by the Ethernet controller. If a function is configured to strip VLAN tag, then the VLAN tag is stripped from the IEEE 802.1q-compliant packet at reception and placed in a receive completion record.

Packet Steering

Receive Side Scaling (RSS)

RSS is a scalable networking technology that enables receive packet processing to be balanced across multiple processors in the system while maintaining in-order delivery of the data. RSS enables different packets, received by a single network adapter, to be processed on different CPUs/cores in parallel while preserving in-order delivery of TCP connections. Receive Side Scaling (RSS) uses a Toeplitz algorithm which uses 4-tuple match on the received frames and forwards it to a deterministic CPU for frame processing. This allows streamlined frame processing and balances CPU utilization. An indirection table is used to map the stream to a CPU. Symmetric RSS allows the mapping of packets of a given TCP or UDP flow to the same receive queue.

Accelerated Receive Flow Steering

Accelerated RFS (aRFS, or RFS) is an Ethernet controller feature that improves packet reception efficiency by delivering packets to queues based on CPU locality of the application. This reduces memory access latency and improves performance. Accelerated RFS takes precedence over RSS when enabled and configured. If the incoming flow does not match any existing n-tuple filters, it is steered according to the RSS hash.

Data Center Bridging

Data Center Bridging (DCB) is a set of protocols and capabilities (for example, DCBX, LLDP, ETS, and PFC) for use in a data center environment. Broadcom Ethernet NIC controllers support for priority flow control is described in section on Priority Flow Control.

Priority Flow Control

Priority Flow Control (PFC) is a standard-compliant backpressure mechanism implemented in Broadcom Ethernet NIC controllers. The goal of PFC is to backpressure congested priority traffic flow without affecting the traffic flows of uncongested priorities and to ensure that packets are not dropped in burst or transient scenarios. PFC can be used in a network with real-time or time-sensitive traffic because of its capability to provide differential treatment to Traffic Classes. For example, using PFC lower priority Internet traffic can be backpressured leaving the higher priority traffic like VOIP and Streaming Video flowing through the link without flow control.

Virtualization Offload

This section contains the following information on virtualization offload:

- Multiqueue Support
- KVM/Xen Multiqueue
- Virtual Machine Queue
- Tunneling Offload

Multiqueue Support

Broadcom Ethernet NIC controllers support Multiqueue in the hardware

KVM/Xen Multiqueue

KVM/Multiqueue returns the frames to different queues of the host stack by classifying the incoming frame by processing the received packet's destination MAC address and or IEEE 802.1Q VLAN tag. The classification combined with the ability to DMA the frames directly into a virtual machine's memory allows scaling of virtual machines across multiple processors.

Virtual Machine Queue

The NDIS Virtual Machine Queue (VMQ) is a feature that is supported by Microsoft to improve Hyper-V network performance. The VMQ feature supports packet classification based on the destination MAC address to return received packets on different completion queues. This packet classification combined with the ability to DMA packets directly into a virtual machine's memory allows the scaling of virtual machines across multiple processors.

VMware NetQueue

The VMware NetQueue is a feature that is similar to Microsoft's NDIS VMQ feature. The NetQueue feature supports packet classification based on the destination MAC address and VLAN to return received packets on different NetQueues. This packet classification combined with the ability to DMA packets directly into a virtual machine's memory allows the scaling of virtual machines across multiple processors.

Xen Multiqueue

Xen multiqueue enables network device drivers to dedicate each Rx queue to a specific guest operating system. This means the network device drivers should be able to allocate physical memory from the set of memory pages assigned to a specific guest operating system.

Tunneling Offload

Stateless Transport Tunnel Offload (STT) is a tunnel encapsulation that enables overlay networks in virtualized data centers. STT uses IP-based encapsulation with a TCP-like header. There is no TCP connection state associated with the tunnel and that is why STT is stateless. Open Virtual Switch (OVS) uses STT. An STT frame contains the STT frame header and payload. The payload of the STT frame is an untagged Ethernet frame. The STT frame header and encapsulated payload are treated as the TCP payload and TCP-like header. The IP header (IPv4 or IPv6) and Ethernet header are created for each STT segment that is transmitted. The NetXtreme-E family of Ethernet controllers support Network Overlays or Tunneling, specifically VXLAN, variants of GRE, and IP-in-IP. Both VXLAN and NVGRE are defined to support larger scale than basic IEEE 802.1Q VLANs, using a 24-bit label space rather than 12-bit VID. Both are L2-in-L3 tunneling methods with NVGRE using GRE to carry the tunnel label and VXLAN using UDP to identify the tunnel label. Stateless offload for tunneling and encapsulated frames in the NetXtreme-E applies to VXLAN, Geneve, L2GRE, NVGRE, and also IP-in-IP scheme. The section below describes VXLAN as an example to discuss the general support for this feature. The difference between different tunneling/encapsulation schemes is noted when it is applicable. All the offloads described in this section are supported on both physical and virtual functions. (PFs and VFs).

VXLAN

A Virtual eXtensible Local Area Network (VXLAN), defined in IETF RFC 7348, is used to address the need for overlay networks within virtualized data centers accommodating multiple tenants. The VXLAN scheme and related protocols are defined in IETF RFC 7348. >VXLAN is a Layer 2 overlay or tunneling scheme over a Layer 3 network. Each overlay is termed a VXLAN segment. Only VMs within the same VXLAN segment can communicate with each other. Each VXLAN segment is scoped through a 24-bit segment ID, VXLAN Network Identifier (VNI). This allows up to 16M VXLAN segments to coexist within the same administrative domain. The UDP destination port identifies the presence of a VXLAN tunnel. A VXLAN frame includes the fields described as follows:

GRE and NVGRE

The NetXtreme-E family of Ethernet controllers support Generic Routing Encapsulation (GRE) per RFC 2784 and RFC 2890. Network Virtualization using GRE (NVGRE) is a Layer 2 overlay, used to address the need of subnets for overlay networks with larger numbers of VLANs. As with the VXLAN scheme, each NVGRE segment is scoped through a 24-bit identifier, called Virtual Subnet Identifier (VSID), in a GRE header to support up to 16M virtual network segments.

Geneve

The NetXtreme-E family of Ethernet controllers supports Generic Network Virtualization Encapsulation (Geneve, also known as Next Generation Encapsulation). It leverages the same concepts as VXLAN, using UDP destination port to identify the presence of the Geneve tunnel header. A primary goal for Geneve is to enable transport of metadata (system state) from a source endpoint to one or more destination endpoints within the virtual network indicated by the tunnel identifier.

IP-in-IP

IP-in-IP is a Layer 3 overlay or tunneling scheme over a Layer 3 network. It is a method by which an IP datagram may be encapsulated (carried as payload) within another IP datagram for the purpose of altering the routing of the inner IP packets and allowing them to be delivered to an intermediate destination that would otherwise not be selected by the destination Address field in the inner IP header. IP Encapsulation with IP is defined in RFC 2003.

Checksum Offload

The following checksums are computed on transmit path and then computed and verified on the receive path.

- Outer IPv4 checksum if the outer IP datagram is an IPv4 datagram.
- Outer UDP checksum (if non-zero): The current VXLAN IETF draft suggests that the outer UDP checksum should be transmitted as zero. If the outer UDP checksum field in the VXLAN frame received is zero, then the frame is accepted without computing outer UDP checksum. If the outer UDP checksum field in the VXLAN frame received is non-zero, then the outer UDP checksum should be computed. (Note: The current IETF draft allows the receiver to ignore outer UDP checksum when it is set to non-zero.) This item is not available in L2GRE/NVGRE/IP-in-IP -aware offload.
- Inner IPv4 checksum if the inner IP datagram is an IPv4 datagram.
- Inner UDP or TCP checksum.

VLAN Tagging

A VXLAN-frame supports the following tagging options:

- No IEEE 802.1Q tag in inner and outer datagrams.
- IEEE 802.1Q tag in the outer IP datagram only.
- IEEE 802.1Q tag in the inner IP datagram only.
- IEEE 802.1Q tags in both the inner and outer IP datagrams.

> **NOTE**
> The device supports insertion and removal of outer IEEE 801.Q Tag for VXLAN/GRE/IP-in-IP frames. It also supports insertion and removal of inner 8IEEE 01.Q Tag for VXLAN.frames. The device retains inner IEEE 801.Q Tag for NVGRE frames in the inner packet.

VMQ

For VXLAN frames, the NetQueue uses the following fields for the queue selection:

- Inner destination MAC address.
- Outer destination MAC address.
- VXLAN Network Identifier (VNI).

The device supports NetQueue selection based on any combination of the fields above.

> **NOTE**
> For GRE/IP-in-IP frames, the VMQ selection is performed using the Ethernet header of the encapsulated packet (inner packet) that includes inner destination MAC address and inner 802.1Q Tag (optional).

RSS

For VXLAN frames, there are two options for RSS queue selection:

- RSS hash computation based on outer UDP/IP headers: The VXLAN IETF draft recommends that the source port is set based on the hash of inner headers. This allows the RSS hash computation based on outer UDP/IP headers a viable option.
- RSS hash computation based on inner UDP/IP or TCP/IP headers: This option requires 2-tuple or 4-tuple hash computation based on inner headers. The inner header is parsed for RSS hash computation. The RSS hash computation is performed in parallel with other checksum computations. In some exceptional cases, this may lead to inaccurate hash computations where one or more checksum validation fails.

  For GRE/IP-in-IP frames, the RSS queue selection is performed using inner headers. The following are possible combinations:
- GRE/IP-in-IP frame with inner TCP/IP or UDP/IP headers: The RSS is performed using four-tuple (src IP, dst IP, src port, dst port) hash on the inner IP header and TCP or UDP header.
- GRE/IP-in-IP frame without inner TCP or UDP header: The RSS is performed using 2-tuple (src IP, dst IP) hash on the inner IP header.
- Other encapsulated frames (for example, GRE/IP-in-IP frames that cannot be parsed): The RSS is performed using 2-tuple (src IP, dst IP) hash on the outer IP header.

TCP Segmentation Offload

For VXLAN, the TCP Segmentation Offload (TSO) algorithm is performed on the inner TCP segment. The hypervisor provides template TCP/IP headers for the inner TCP segment as well as template VXLAN/UDP/IP headers for the outer UDP datagram. For every inner TCP segment generated by the VXLAN-aware TSO, the outer VXLAN, UDP, IP, and MAC headers are inserted and outer IPv4 checksum, outer UDP checksum (not for GRE frames), inner IP checksum (for inner IPv4 datagram only), and inner TCP checksum is computed and inserted. The device updates the IP ID field for every inner TCP segment.

For GRE/IP-in-IP, the LSO (Large Send offload) algorithm is performed on the inner TCP segment. The hypervisor provides template TCP/IP headers for the inner TCP segment as well as template GRE/IP/Ethernet headers for the outer IP datagram. For every inner TCP segment generated by the GRE/IP-in-IP-aware LSO, the outer GRE (not applicable for IP- in-IP frames), IP, and MAC headers is inserted and outer IPv4 checksum (for outer IPv4 datagrams only), inner IP checksum (for inner IPv4 datagram only), and inner TCP checksum is computed and inserted. The device updates the IP ID field for every inner TCP segment.

Large Receive Offload

Tunneling Offload support for LRO, RSC, TSO, LSO, GSO, and GRO

SR-IOV

The PCI-SIG defines optional support for Single-Root I/O Virtualization (SR-IOV). SR-IOV is designed to allow access of the VM directly to the device using Virtual Functions (VFs). The NIC Physical Function (PF) is divided into multiple virtual functions and each VF is presented as a PF to VMs. SR-IOV uses IOMMU functionality to translate PCIe virtual addresses to physical addresses by using a translation table. The number of Physical Functions (PFs) and Virtual Functions (VFs) are managed through the UEFI HII menu, the CCM, and through NVRAM configurations. SR-IOV can be supported in combination with NPAR mode. The SR-IOV feature requires corresponding SR-IOV support in the BIOS (Intel VT-d or

AMD IOMMU) and Operating System/ Hypervisor as well as the PCIe endpoint device (the NIC in this case). Broadcom Ethernet NIC controllers offer the following offload functionality to the VF that are available to the PF:

- TX and RX IP/TCP/UDP checksum offload
- Large Send offload (LSO) or TCP segmentation offload (TSO, GSO)
- Receive Segmentation offload (RSC) or Large Receive offload (LRO, GRO)
- Receive Side Scaling (RSS) – up to 64 queues per VF
- Multiple COS queues – up to 4 queues per VF
- Network Virtualization Generic Routing Encapsulation, Virtual Extensible LAN – NVGRE/VXLAN

SR-IOV Configuration Support Matrix

The following table provides an SR-IOV support matrix.

Table 16: SR-IOV Support Matrix

Host OS	Guest OS – VF					
Host OS	Win2k16	Win2k19	RH7.3+	RH8.x	SLES12.2+	SLES15.x
Windows 2016	Yes	Yes	Yes	Yes	Yes	Yes
Windows 2019	Yes	Yes	Yes	Yes	Yes	Yes
RH7.8+	Yes	Yes	Yes	Yes	Yes	Yes
RH8.x	Yes	Yes	Yes	Yes	Yes	Yes
SLES15.x	Yes	Yes	Yes	Yes	Yes	Yes
ESX6.7+	Yes	Yes	Yes	Yes	Yes	Yes
ESX7.x	Yes	Yes	Yes	Yes	Yes	Yes

Network Partitioning (NPAR)

The Network Partitioning (NPAR) feature allows a single physical network interface port to appear to the system as multiple network device functions. When NPAR mode is enabled, the Broadcom Ethernet NIC controller is enumerated as multiple PCIe physical functions (PF). Each PF or partition is assigned a separate PCIe function ID on initial power on. Each partition is assigned its own configuration space, BAR address, and MAC address allowing it to operate independently. Partitions support direct assignment to VMs, VLANs, and so on, just as any other physical interface.

The original PCIe definition allowed for eight PFs per device. For Alternative Routing-ID (ARI) capable systems, Broadcom NetXtreme-E adapters support up to 16 PFs per device.

Security (BCM575XX Only)

The BCM575XX TruTrust™ technology is capable of secure boot meaning it only executes boot images authenticated by the secure boot loader (SBL). Secure boot functionality is the cornerstone of a security enabled system since it is the root of trust from which all subsequent applications are run. The secure boot capability provides the following functionality:

- Secure boot Core Root of Trust – The Secure Boot Loader is based in device ROM, and outside the scope of modification. It functions as the Core Root of Trust for software, meaning that the system is in a trusted state from reset to when a secure image has been authenticated.
- Boot Image Authentication – Only Images authenticated by the SBL are executed by the system.
- Boot Image Integrity – The SBL cryptographically validates the integrity of the Secure Boot Image before it is executed to ensure that it has not been tampered with maliciously or errantly.
- Boot Image Confidentiality – The secure processor has the hardware support to execute encrypted images which ensures that device images are never in the clear and protected from reverse engineering or used in device cloning.

Secure devices can be delivered to the customer in a state pending final customization. This customization step is executed by the customer, and once complete, only customer signed images execute on the device. Customization provides the following capabilities:

- Customer takes responsibility for the creation and management of keys used in signing their code. This allows the customer to apply their own security policies in managing their keys and ensures that no code can be signed for their devices by a third party.
- Only code signed by the customer runs on their customized device. This ensures that device code cannot be tampered with in the field and verifies the authenticity of the image.
- Customer signed images do not run on other customers secure devices. This prevents piracy of customer images. However, it does not relieve the customer of the responsibility for protecting unsecured binaries from reverse engineering.
- Device or customer specific encrypted execution images can be generated. This prevents piracy of customer images and cloning of devices.

Supported Combinations

The following sections describe the supported feature combinations for these devices:

The following table shows the supported feature combinations of NPAR, SR-IOV, and RoCE.

Table 17: NPAR, SR-IOV, and RoCE

SW Feature	Notes
NPAR	Up to 8 PFs or 16 PFs
SR-IOV	Up to 128 VFs (total per chip)
RoCE on PFs	Up to 4 PFs for the BCM5741X devices and 16 PFs for BCM575XX devices
RoCE on VFs	BCM575XX supports RoCE SRIOV over up to 128 VFs in the 219.x Release. The 218.x release does not support RoCE SRIOV for BCM575XX. BCM541X does not support RoCE on VFs.
Host OS	Linux, Windows, ESXi (no vRDMA support)
Guest OS	Linux and Windows
DCB	Up to two COS per port with non-shared reserved memory

NOTE
Certain 4 port BCM9575XX adapters support up to 32 VF and 4 NPAR per port. When NPAR and SR-IOV are enabled, certain ESXi OS are not able to configure more than 8 VFs on partitions 3 and above.

Table 18: NPAR, SR-IOV, and DPDK

SW Feature	Notes
NPAR	Up to 8 PFs or 16 PFs
SR-IOV	Up to 128 VFs (total per chip)
DPDK	Supported only as a VF
Host OS	Linux
Guest OS	DPDK (Linux)

Unsupported Combinations

RoCE SRIOV + NPAR is not supported. BCM5741X does not support RoCE SRIOV. BCM575XX does not support RoCE SRIOV in 218.x release. 2.19 release supports RoCE SRIOV for BCM575XX.

RDMA over Converged Ethernet – RoCE

Remote Direct Memory Access (RDMA) over Converged Ethernet (RoCE) is a complete hardware offload feature in the Broadcom Ethernet NIC controller which allows RDMA functionality over an Ethernet network. RoCE functionality is available for both user mode and kernel mode applications. RoCE is supported under Linux, Windows and VMware operating systems.

> **NOTE**
> When RoCE is enabled, the ETS configuration is modified resulting in optimized RDMA performance and less accurate ETS bandwidth allocation.

See the following links for RDMA support for each operating system:

Windows

Microsoft SMB Direct

Linux

Installing the Linux Driver

VMware

VMware Network Requirements for RDMA

Supported Products:

- BCM9575XX
- BCM95741X

Supported RDMA Protocols:

- RoCE v2 (default)
- RoCE v1

Hardware Requirements

Ensure that RDMA is enabled in the boot-up CCM/HII menus.

Ensure the PCIe slot used for your RNIC supports the Ethernet speed required, otherwise, optimal performance is not obtained. This is important for RNICs since RDMA operations typically use all available bandwidth compared to TCP applications which are often limited to 20-30 Gb/s per stream (see the following table).

Table 19: PCIe Slot Throughput

PCIe Generation	Slot Width (Lanes)	Ethernet Maximum Throughput
4 (Example, AMD Rome/Milan)	16	200 Gb/s
4 (Example, AMD Rome/Milan)	8	100 Gb/s
3 (Example, Intel Xeon)	16	100 Gb/s
3 (Example, Intel Xeon)	8	50 Gb/s

The PCIe slot width is often written on the system motherboard or PCIe riser next to the slot as xN where N is the width).

> **NOTE**
> Make note of the maximum network cable speeds. Broadcom RNICs typically use SFP and QSFP ports. The selected cable must fit physically and support the maximum speed required. Use the shortest cable which reaches the endpoints for the best speed and lowest bit-error rate.

Ensure that the RNIC receives adequate cooling air from the system fans. Overheating triggers a shutdown, interrupting service. Active cabling such as AOC ACC and optical transceivers require additional power and produce more heat.

Software Requirements

The Broadcom RoCE implementation uses the RDMA software stack (librdma, libibverbs) included with all major Linux distributions. Any installation of another vendor's proprietary software stack should be removed before proceeding with these instructions.

Hardware Installation

Provides instructions for installing the hardware components of Broadcom Ethernet NIC controllers.

All components are bundled in the zip file available from MyBroadcom.com.

This section provides the following hardware installation information:

- Safety Precautions
- Hardware Requirements
- NIC Installation
- Cabling
- Ethernet Adapter
- UEFI HII Menu
- BIOS Comprehensive Configuration Management

Safety Precautions

CAUTION!Server class system power supplies with higher current may be hazardous when normal operating procedures are not used. Before removing the cover of the 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 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.

Hardware Requirements

See the following list of hardware requirements:

- One open PCIe Gen3 or Gen4 slot in x8, x16, or x32 mode.
  - 574XXA41XX – PCIe Gen3 slot in X8 mode.
  - 574XXM41XX – OCP 2.0 or rNDC slots.
  - 57XXXNXXXX – OCP 3.0 slot
- 16 GB memory or more (32 GB or more is recommended for virtualization applications and nominal network throughput performance).

NIC Installation

The following instructions apply to installing the Broadcom Ethernet NIC controller (add-in NIC) into most servers. See the manuals that are supplied with the server for details about performing these tasks on this particular server.

1. Review the Safety Precautions and Preinstallation Checklist before installing the adapter. Ensure that the system power is OFF and unplugged from the power outlet, and that proper electrical grounding procedures have been followed.
2. Open the system case and select any empty PCIe Gen3 or Gen4 x8 or x16 slot.
3. Remove the blank cover plate from the slot.
4. Align the adapter connector edge with the connector slot in the system.
5. Secure the adapter with the adapter clip or screw.

6. Close the system case and disconnect any personal antistatic devices.

Cabling

Broadcom executes basic interoperability testing with a subset of cables and transceivers from the marketplace. The interoperability testing does not include BER tests, power, or temperature measurements. Broadcom validates interoperability with many different AOCs, optical modules, and DAC cables, although no vendor can test with every model available on the market. Broadcom Ethernet adapters support any IEEE compliant AOC, optical module or DAC cable in the market, even if a particular AOC, optical module or DAC cable may not have been part of a qualification cycle within Broadcom's interoperability lab. Moreover, Broadcom ensures continued compliance with these industry standards by constantly testing with a subset of industry-compliant AOCs, optical modules, and DAC cables.

The firmware code conforms to the following SNIA industry standards:

- SFF-8024 (SFF Module Management Reference Code Tables)
- SFF-8472 (Management Interface for SFP+)
- SFF-8636 (Management Interface for 4-lane Modules and Cables)
- SFF-8665. (QSFP+ 28 Gb/s 4× Pluggable Transceiver Solution (QSFP28))

The cables shown in the following table are tested for compatibility with Broadcom's generic release or later.

Table 20: Tested Cables

Manufacturer	Cable/Transceiver Part no	Speed	Connector	Type	Length
Avago	AFBR-710DMZ	10G	SFP+	Transceiver	N/A
Cisco	SFP-10G-AOC1M	10G	SFP+	AOC	1 m
Cisco	SFP-10G-AOC7M	10G	SFP+	AOC	7 m
Cisco	SFP-H10GB-CU7M	10G	SFP+	DAC	7 m
Dell	C5RNH	10G	SFP+	Transceiver	N/A
Dell	K0T7R	10G	SFP+	AOC	15 m
Dell	WTRD1	10G	SFP+	Transceiver	N/A
HPE	AP818A	10G	SFP+	AOC	1 m
Avago	AFBR-735SMZ	10/25G	SFP28	Transceiver	N/A
Avago	AFBR-8CER02Z	10/25G	SFP28	AOC	2 m
Dell	M14MK	10/25G	SFP28	Transceiver	N/A
Mellanox	MFA2P10-A005	10/25G	SFP28	AOC	5 m
Accelink	RTXM330-005-C20	25G	SFP28	AOC	20 m
Accelink	RTXM330-C05	25G	SFP28	AOC	5 m
Accelink	RTXM330-C07	25G	SFP28	AOC	7 m
Accelink	RTXM330-C10	25G	SFP28	AOC	10 m
Accelink	RTXM330-C20	25G	SFP28	AOC	20 m
Cisco	SFP-25G-AOC7M	25G	SFP28	AOC	7 m
Cisco	SFP-H25G-CU5M	25G	SFP28	DAC	5 m
Cisco	SFP-25G-AOC5M	25G	SFP28	AOC	5 m
Dell	P7D7R	25G	SFP28	Transceiver	N/A

Manufacturer	Cable/Transceiver Part no	Speed	Connector	Type	Length
Dell	3YWG7	25G	SFP28	AOC	7 m
Dell	9X8JP	25G	SFP28	DAC	5 m
Dell	0YR96	25G	SFP28	Transceiver	N/A
Dell	RCVP5	25G	SFP28	AOC	15 m
Finisar	FCCG125SD1C10BTE	25G	SFP28	AOC	10 m
Finisar	FCBG125SD1C05BTE	25G	SFP28	AOC	5 m
Finisar	FCCG125SD1C05	25G	SFP28	AOC	5 m
Finisar	FCCG125SD1C20BTE	25G	SFP28	AOC	20 m
H3C	SFP-25G-D-AOC-10M-DG	25G	SFP28	AOC	10 m
H3C	SFP-25G-D-AOC-20M-DG	25G	SFP28	AOC	20 m
H3C	SFP-25G-D-AOC-5M-DG	25G	SFP28	AOC	5 m
H3C	SFP-25G-D-AOC-7M-DG	25G	SFP28	AOC	7 m
HPE	844477-B21	25G	SFP28	DAC	3 m
Innolight	TF-PY005-NTA	25G	SFP28	AOC	5 m
Innolight	TF-PY020-NTA	25G	SFP28	AOC	20 m
Luxshare	SFP-25G-DAC-5.0M-B-TX	25G	SFP28	DAC	5 m
Luxshare	SFP-25G-ACC-7.0M	25G	SFP28	ACC	7 m
Mellanox	7G17A03537/MMA2P00-AS	25G	SFP28	Transceiver	N/A
Dell	VXFJY	50G	SFP28	DAC	3 m
3M	9QM6-517-23-2.75	100G	QSFP28	DAC	N/A
3M	9QM6-517-23-2.50	100G	QSFP28	DAC	N/A
Accelink	RTXM420-550	100G	QSFP28	Transceiver	N/A
Accelink	RTXM290-806-C14	100G	–	Transceiver	N/A
Amphenol	FOQQD33P00003	100G	QSFP28	AOC	3 m
Avago	AFBR-89CDHZ	100G	QSFP28	Transceiver	N/A
Dell	4WGYD	100G	QSFP28	Transceiver	N/A
Finisar	FCBN425QE1C03	100G	QSFP28	AOC	3 m
Finisar	FTLC9551REPM	100G	QSFP28	Transceiver	N/A
Finisar	FTLC9558REPM	100G	QSFP28	Transceiver	N/A

Manufacturer	Cable/Transceiver Part no	Speed	Connector	Type	Length
H3C	QSFP-100G-SR4-MM850-A	100G	QSFP28	Transceiver	N/A
H3C	QSFP-100G-CWDM4-SM1300-A	100G	QSFP28	Transceiver	N/A
HPE	845406-B21	100G	QSFP28	Cable	3 m
Huawei	QSFP-100G-SR4	100G	QSFP28	Transceiver	N/A
Innolight	TR-FC85S-NTC	100G	QSFP28	Transceiver	N/A
Mellanox	MMA1B00-C100D	100G	QSFP28	Transceiver	N/A
Mellanox	MFA1A00-C005	100G	QSFP28	AOC	5 m
Mellanox	MFA7A20-C003	100G	QSFP28	AOC	3 m
Molex	1002976020	100G	QSFP28	DAC	2 m
Molex	1002976007	100G	QSFP28	DAC	0.75 m
Molex	1002976010	100G	QSFP28	DAC	1 m
Molex	1002976030	100G	QSFP28	DAC	3 m
Molex	1002976035	100G	QSFP28	DAC	3.5 m
Innolight	TR-FC13T-NTC	100G	QSFP28	Transceiver	N/A
Innolight	TR-FC13R-N00	100G	QSFP28	Transceiver	N/A
Finisar	FTLC1154RDPL	100G	QSFP28	Transceiver	N/A
Amphenol	NDAAXJ0003	100G	QSFP56	DAC	3 m
Mellanox	MCP7H50-H002R26	100G	QSFP56	DAC	2 m
Amphenol	NDARHG-F306	100G	QSFP56	DAC	3 m
Mellanox	MMA1T00-VS	100G	QSFP56	Transceiver	N/A
Mellanox	MFS1S00-V005E	100G	QSFP56	AOC	5 m
Optomind	C4R448GA005AZZ	100G	QSFP56	AOC	5 m

For further information contact your local Broadcom field or application engineer.

Ethernet Adapters

Broadcom provides interoperability testing with popular host operating systems on the market (see the following tables). For Linux distributions not shown in these tables, the system should be operational after recompiling the host drivers (for example, bnxt_en and bnxt_re).

Table 21: BNXT 218.0 Release

Operating System/Product	BCM575XX	BCM5745X	BCM5745X
RHEL/CentOS	7.9 8.2 8.3	7.9 8.2 8.3	7.9 8.2 8.3
SLES	12 SP5 15 SP1 15 SP2	12 SP5 15 SP1 15 SP2	12 SP5 15 SP1 15 SP2
Ubuntu	20.04	20.04	20.04
vSphere/ESX	6.7 6.7 u3 7.0 7.0 u1	6.5 6.5 u3 6.7 6.7 u3 7.0 7.0 u1	6.5 6.5 u3 6.7 6.7 u3 7.0 7.0 u1
Windows	–	Windows 10	Windows 10
Windows Server	2016 2019	2016 2019	2016 2019
XenServer	8.2 LTSR	8.2 LTSR	8.2 LTSR

Table 22: BNXT 216.0 Release

Operating System/Product	BCM575XX	BCM5745X	BCM5745X
RHEL/CentOS	7.4 7.5 7.6 7.7 7.8 8.0 8.1 8.2	7.4 7.5 7.6 7.7 7.8 8.0 8.1 8.2	7.4 7.5 7.6 7.7 7.8 8.0 8.1 8.2
SLES	12 SP4 12 SP5 15 SP1 15 SP2	12 SP4 12 SP5 15 SP1 15 SP2	12 SP4 12 SP5 15 SP1 15 SP2
Ubuntu	20.04	20.04	20.04

Operating System/Product	BCM575XX	BCM5745X	BCM5745X
vSphere/ESX	6.7 6.7 u1 6.7 u2 6.7 u3 7.0	6.5 6.5 u1 6.5 u2 6.5 u3 6.7 6.7 u3 7.0	6.5 6.5 u1 6.5 u2 6.5 u3 6.7 6.7 u3 7.0
Windows	–	Windows 10	Windows 10
Windows Server	2016 2019 RS5	2016 2019 RS5	2016 2019 RS5
XenServer	8.2 LTSR	8.0 8.2 LTSR	8.0 8.2 LTSR

UEFI HII Menu

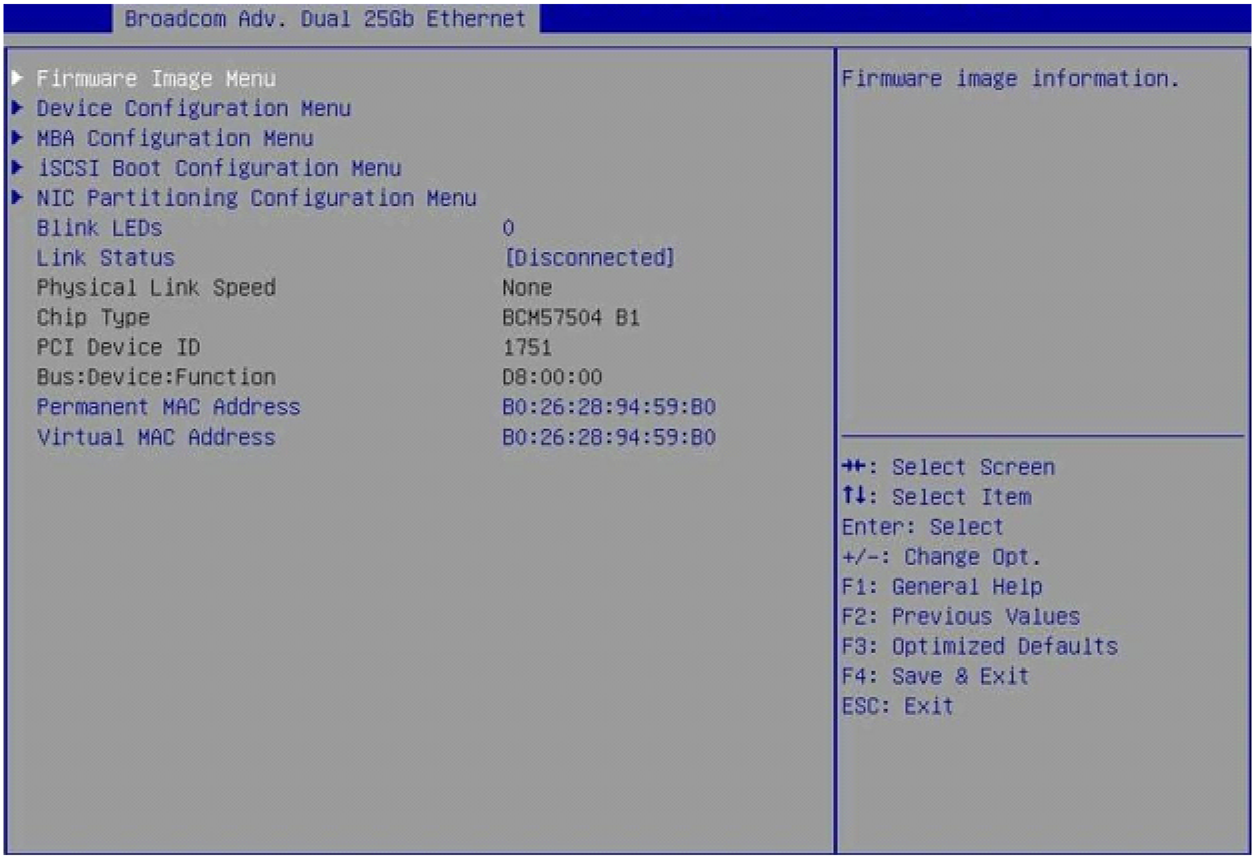
The Broadcom Ethernet NIC controllers can be configured using the HII (Human Interface Infrastructure) menu at boot time. This menu system allows configuration of all persistent settings such as boot protocol (PXE) and virtualization modes (SR-IOV, NPAR), and so on. To enter the HII configuration menu, follow boot-time prompts to BIOS, then device configuration. The layout of the menus of an adapter may not look the same as the others and some settings may not be available or found on the same menu for a different adapter type.

The UEFI HII menu consists of the following menus:

- Main Configuration Page
- Firmware Image Menu
- Device Configuration Menu
- MBA Configuration Menu NIC Configuration
- NIC Partitioning Configuration Menu

Main Configuration Page

Figure 21: Main Configuration Page



```
Broadcom Adv. Dual 25Gb Ethernet

▶ Firmware Image Menu                                  Firmware image information.
▶ Device Configuration Menu
▶ MBA Configuration Menu
▶ iSCSI Boot Configuration Menu
▶ NIC Partitioning Configuration Menu
  Blink LEDs                  0
  Link Status                 [Disconnected]
  Physical Link Speed         None
  Chip Type                   BCM57504 B1
  PCI Device ID               1751
  Bus:Device:Function         D8:00:00
  Permanent MAC Address       B0:26:28:94:59:B0
  Virtual MAC Address         B0:26:28:94:59:B0
                                                       →←: Select Screen
                                                       ↑↓: Select Item
                                                       Enter: Select
                                                       +/-: Change Opt.
                                                       F1: General Help
                                                       F2: Previous Values
                                                       F3: Optimized Defaults
                                                       F4: Save & Exit
                                                       ESC: Exit
```

This page displays the following information:

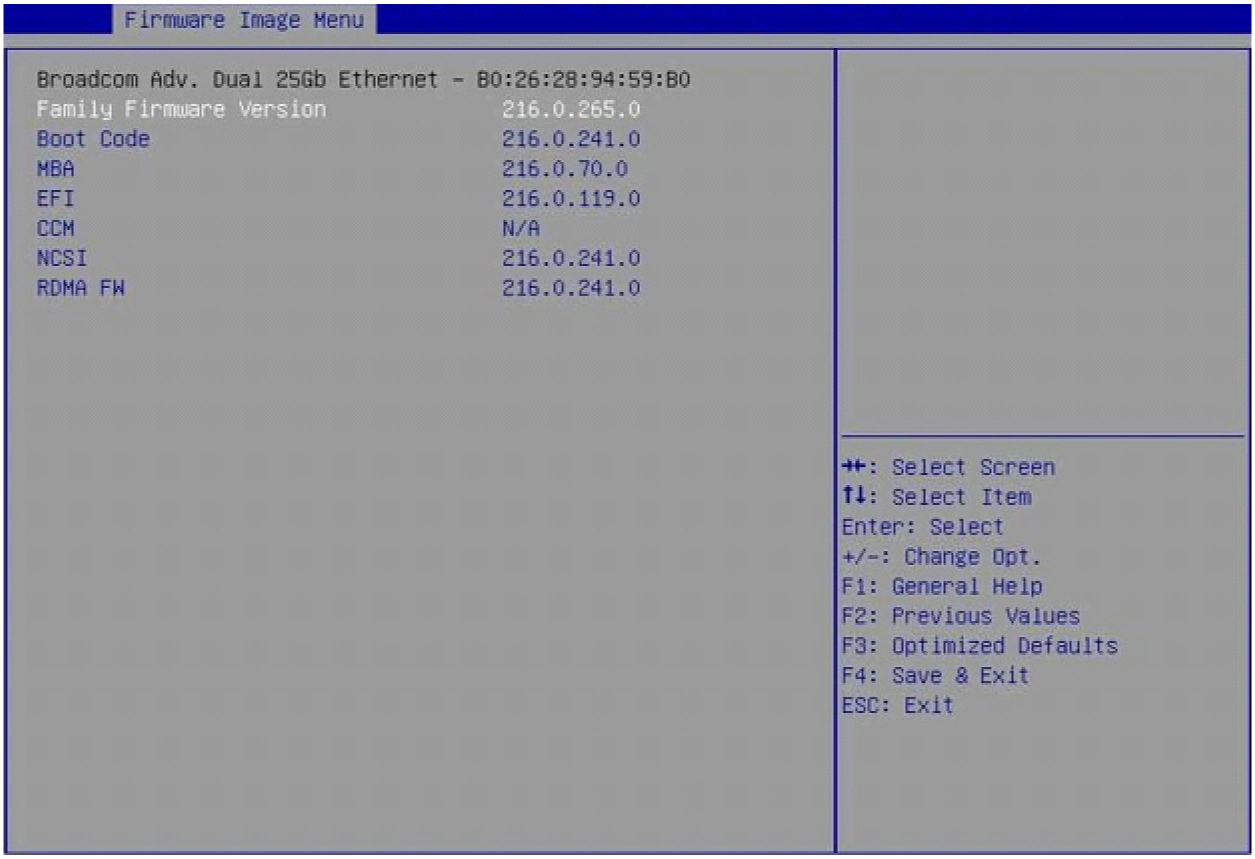
- **Firmware Image Menu** – This menu presents the various component versions present in the current firmware package.
- **Device Configuration Menu** – This menu presents adapter specific parameters for configuration.
- **MBA Configuration Menu NIC** – This menu presents PXE boot related parameters for configuration.
- **NIC Partitioning Configuration Menu**– This menu presents NIC partition related parameters for configuration.
- **Blink LEDs**– This setting allows the user to configure the duration for which the LEDs on the physical network port should blink to assist with port identification. This is a numeric setting. The value must be specified in the range 0 to 15 seconds.
- **Link Status**– This field displays the physical link status of the network port as reported by the controller. This is a read-only field.

  - **Connected**– Link is up
  - **Disconnected**– Link is down
- **Physical Link Speed**– This field displays the current link speed of the network port as reported by the controller. This is a read-only field. Speed is reported in Mb/s/Gb/s.
- **Chip Type**– This field displays the Broadcom specific identifier which denoted the adapter family to which the chip belongs and the revision. This is a read-only field.
- **PCI Devic ID**– This field displays the 16 bit PCI Device ID reported by the controller. This is a vendor defined ID which varies across non-NPAR, NPAR and RDMA mode. Refer to MF mode and Support RDMA sections for more information on these modes. This is a read-only field.
- **Bus Device Function**– This field displays the BIOS assigned PCI Bus:Device:Function identifier of the card. This is a read-only field.
- **Permanent MAC Address**– This field displays the Permanent MAC address assigned during manufacturing. This is a read-only field.
- **Virtual MAC Address**– This field displays the Virtual MAC address assigned to the device. This is a read-only field from the HII menu. The value for this parameter can be configured using remote utilities.

Firmware Image Menu

This menu presents the various component versions present in the current firmware build. Depending on the adapter type, some components may not be available. All fields in this menu are read-only.

Figure 22: Firmware Image Menu



```
Firmware Image Menu

Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:94:59:B0
Family Firmware Version              216.0.265.0
Boot Code                            216.0.241.0
MBA                                  216.0.70.0
EFI                                  216.0.119.0
CCM                                  N/A
NCSI                                 216.0.241.0
RDMA FW                              216.0.241.0
```

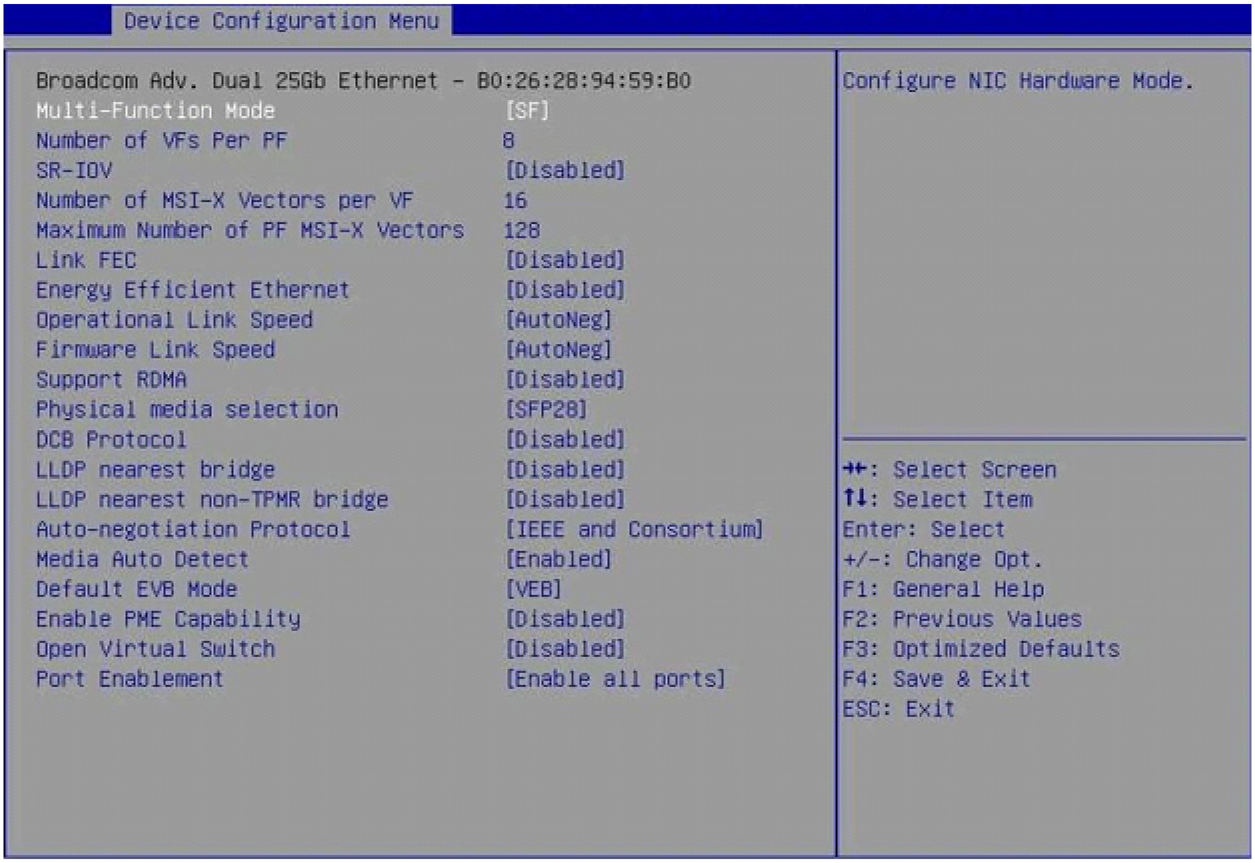
This page displays the following information:

- **Family Firmware Version**– This field displays the family firmware version. This field may be displayed as Firmware Bundle on some adapters.
- **Boot Code**– This field displays the firmware boot code version.
- **MBA** – This field displays the legacy pre-boot driver version.
- **EFI** – This field displays the UEFI pre-boot driver version.
- **CCM**– This field displays the Comprehensive Configuration Menu (CCM) version.
- **NC-SI**– This field displays the NCSI firmware version.
- **RDMA FW**–This field displays the RoCE firmware version.

Device Configuration Menu

This menu presents adapter specific parameters for configuration. Depending on the adapter type, some settings may not be available.

Figure 23: Device Configuration Menu



```
Device Configuration Menu

Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:94:59:B0
Multi-Function Mode                    [SF]
Number of VFs Per PF                   8
SR-IOV                                 [Disabled]
Number of MSI-X Vectors per VF         16
Maximum Number of PF MSI-X Vectors     128
Link FEC                               [Disabled]
Energy Efficient Ethernet              [Disabled]
Operational Link Speed                 [AutoNeg]
Firmware Link Speed                    [AutoNeg]
Support RDMA                           [Disabled]
Physical media selection               [SFP28]
DCB Protocol                           [Disabled]
LLDP nearest bridge                    [Disabled]
LLDP nearest non-TPMR bridge           [Disabled]
Auto-negotiation Protocol              [IEEE and Consortium]
Media Auto Detect                      [Enabled]
Default EVB Mode                       [VEB]
Enable PME Capability                  [Disabled]
Open Virtual Switch                    [Disabled]
Port Enablement                        [Enable all ports]

Configure NIC Hardware Mode.
```

This page allows the user to configure the following items:

- **Multi-Function Mode** – This setting configures the type of virtualization to be used by the controller on all ports. This is available only when NPAR is supported on the adapter.
  - **Single Function Mode (SF)**– In this mode, a single PCIe PF is assigned to each network port.
  - **Network Partitioning Mode (NPAR)**–This mode allows a single physical network port to appear to the system as multiple network device functions. Each PF or partition is assigned a separate PCIe function ID on initial power

on, and a menu is made visible in setup to configure each partition. The 16 configurable partitions are distributed equally across the network ports.

  - **SR-IOV –** This setting configures Single Root - I/O Virtualization (SR-IOV) which allows different virtual machines (VMs) in a virtual environment to share a single PCI Express hardware interface.
    - **Enabled**– Enable support for SR-IOV
    - **Disabled**– Disable support for SR-IOV

      NOTE

      This setting is available only on adapters which support SR-IOV.

- **Number of VFs per PF** – This setting allows the user to configure the number of PCI Virtual Functions Advertised by the port in PCI config space when SR-IOV is enabled in non-NPAR mode. This setting is available for configuration only in non-NPar mode. This is a numeric setting. The value must be specified in multiples of 8.

  NOTE

  The maximum number of VFs supported by software is 128 VFs per device.

- **Number of MSI-X Vectors per VF**

  This setting allows the user to configure the MSI-X Vectors per VF. Message Signaled Interrupts (MSI) are an alternative in-band method of signaling an interrupt, using special in-band messages to replace traditional out-of-band assertion of dedicated interrupt lines. This is a numeric setting. The maximum number of virtual functions supported by the adapter are shared across the number of physical ports on the adapter. Keep the default value of 16 for the BCM574XX and 8 for the BCM5750X to achieve the best resource allocation and maximum number of VFs.
- **Maximum Number of PF MSI-X Vectors** – This setting allows the user to configure the Maximum Number of MSI-X Vectors for a physical function. This is a numeric setting. The minimum value for this setting is 0. The maximum value varies across adapters. Keep the default value of 74 to achieve the best resource allocation and maximum number of VFs.
- **Link FEC**– This setting configures the Forward Error Correction (FEC) mode which is a technique used for controlling errors in data transmission over unreliable or noisy communication channels. This option is useful when longer fiber cables are utilized. This setting is not available on 10GBASE-T controllers. This setting is not available on 10G BaseT controllers. Only a subset of the possible values display on some adapters, based on the configuration. Possible values are:
  - Disabled
  - CL74 – Fire Code
  - CL91 – Reed Solomon
  - RS544 – RS544, using 1 x N RS
  - RS272 – RS272, using 1 x N RS
  - RS544 – RS544, using 2 x N RS
  - RS272 – RS272, using 2 x N RS

  NOTE

  When Media Auto Detect and Auto-negotiation is enabled, FEC is negotiated based on the link partner advertisement.

- **Energy Efficient Ethernet** – This setting configures the Energy Efficient Ethernet (EEE) mode which are a set of enhancements that allow for less power consumption during periods of low-data activity. This setting is available only on 10GBASE-T controllers.
  - **Enabled**– Turn on EEE mode
  - **Disabled**–Turn off EEE mode
- **Operational Link Speed**– This setting configures the default link speed for pre-OS environment in full-power (D0) state. The possible values for this setting depends on the link speeds supported by the adapter.

NOTE
The value for this setting is fixed on some adapters based on the configuration.

- **Firmware Link Speed**– This setting configures the default link speed for pre-OS environment in sleep (D3) state. This setting may be grayed out on some adapters. The possible values for this setting depends on the link speeds supported by the adapter.
- **Support RDMA** – This setting configures Remote Direct Memory Access (RDMA) support on the port. RDMA is a technology that permits computers on a network to exchange data in main memory without the involvement of the processor, cache or operating system of either computer. RDMA allows high throughput and low-latency networking. This setting is available only when RDMA is supported on the adapter. This setting will be displayed on the Device Configuration menu in SF mode and in the NIC Partition Configuration menu in NPar mode.
  - **Enabled**– Turn on RDMA
  - **Disabled**– Turn off RDMA
- **Physical Media Selection**– This setting configures the type of physical media to be used on the network port. This setting is available only on dual media devices.

  – **SFP28**

  –**RJ-45**
- **DCB Protocol**– This setting configures the Data Center Bridging (DCB) settings for the controller. Some of the following options may not be available depending on the adapter in use.
  - **Enabled (IEEE only)**
  - **CEE (only)**
  - **Both (IEEE preferred with fallback to CEE)**
- **LLDP nearest bridge**– This setting configures the Link Layer Discovery Protocol (LLDP) which is a vendor-neutral link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a local area network based on IEEE 802 technology, principally wired Ethernet. An LLDP agent is a mapping of an entity where LLDP runs.
  - **Enabled** – Turn on LLDP nearest bridge
  - **Disabled** – Turn off LLDP nearest bridge
- **LLDP nearest non-TPMR bridge** – This setting configures the Link Layer Discovery Protocol (LLDP) which is a vendor-neutral link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a local area network based on IEEE 802 technology, principally wired Ethernet. An LLDP agent is a mapping of an entity where LLDP runs. This setting enables LLDP on the nearest non-TPMR bridge agent.
  - **Enabled**– Turn on LLDP nearest non-TPMRbridge
  - **Disabled**– Turn off LLDP nearest non-TPMRbridge
- **Auto-negotiation Protocol** – This setting configures the Auto-negotiation protocol for the adapter. Auto-negotiation is a feature that allows a port on a switch, router, server, or other device to communicate with the device on the other end of the link to determine the optimal duplex mode and speed for the connection. This setting is not available on 10GBASE-T controllers.
  - **IEEE and BAM**
  - **IEEE and Consortium**
  - **BAM Only**
  - **Consortium Only**
  - **IEEE 802.3by**
- **Media Auto Detect** – This setting allows the user to configure the Media Auto Detect feature. This setting is not available on 10GBASE-T controllers.
  - **Enabled** – Turn on Media Auto Detect.
  - **Disabled**– Turn off Media Auto Detect.
- **Default EVB Mode**– This setting configures the Edge Virtual Bridging (EVB) mode which is an IEEE standard that involves the interaction between virtual switching environments in a hypervisor and the first layer of the physical switching infrastructure.

  - **VEB**
  - **VEPA**
  - **None**
- **Port Link Training**– This setting configures port link training when using force link speed. Link training should be enabled when using PAM-4. Link training should be disabled when the attached switch does not support link training.
  - **Enabled**– Port Link Training
  - **Disabled**– Port Link Training
- **Live Firmware Upgrade** – This feature allows the device firmware to be upgraded with minimal down time and minimal traffic interruption. This avoids a host reboot, device power cycle, or driver reload. This feature is supported on Linux only. RDMA must be disabled for live firmware update.
  - **Enabled**– Live firmware Upgrade
  - **Disabled**– Live firmware Upgrade
- **Adapter Error Recovery** – This feature enables the recovery of firmware from fatal errors without manual intervention, host reboot, or power cycle. This feature is supported on Linux only.
  - **Enabled**– Adapter Error Recovery
  - **Disabled**– Adapter Error Recovery
- **PME Capability** – This setting configures PME which is the ability to remotely wake a server using Power Management Event (PME). Devices supporting the native PCI Power Management can generate wakeup signals called PMEs to let the kernel know about external events requiring the device to be active.
  - **Enabled**– Turn on PME.
  - **Disabled**– Turn off PME.
- **Open Virtual Switch** – This setting configures Open Virtual Switch (OVS) which is a multilayer virtual switch, designed to enable massive network automation through programmatic extension, while still supporting standard management interfaces and protocols (for example, NetFlow, sFlow, IPFIX, RSPAN, CLI, LACP, 802.1ag).
  - **Enabled** – Turn on Open Virtual Switch.
  - **Disabled** – Turn off Open Virtual Switch.
- **Port Enablement**– This setting configures the number of active ports (PCI functions) on the adapter. This setting is available only on adapters that support the Port Enablement feature.

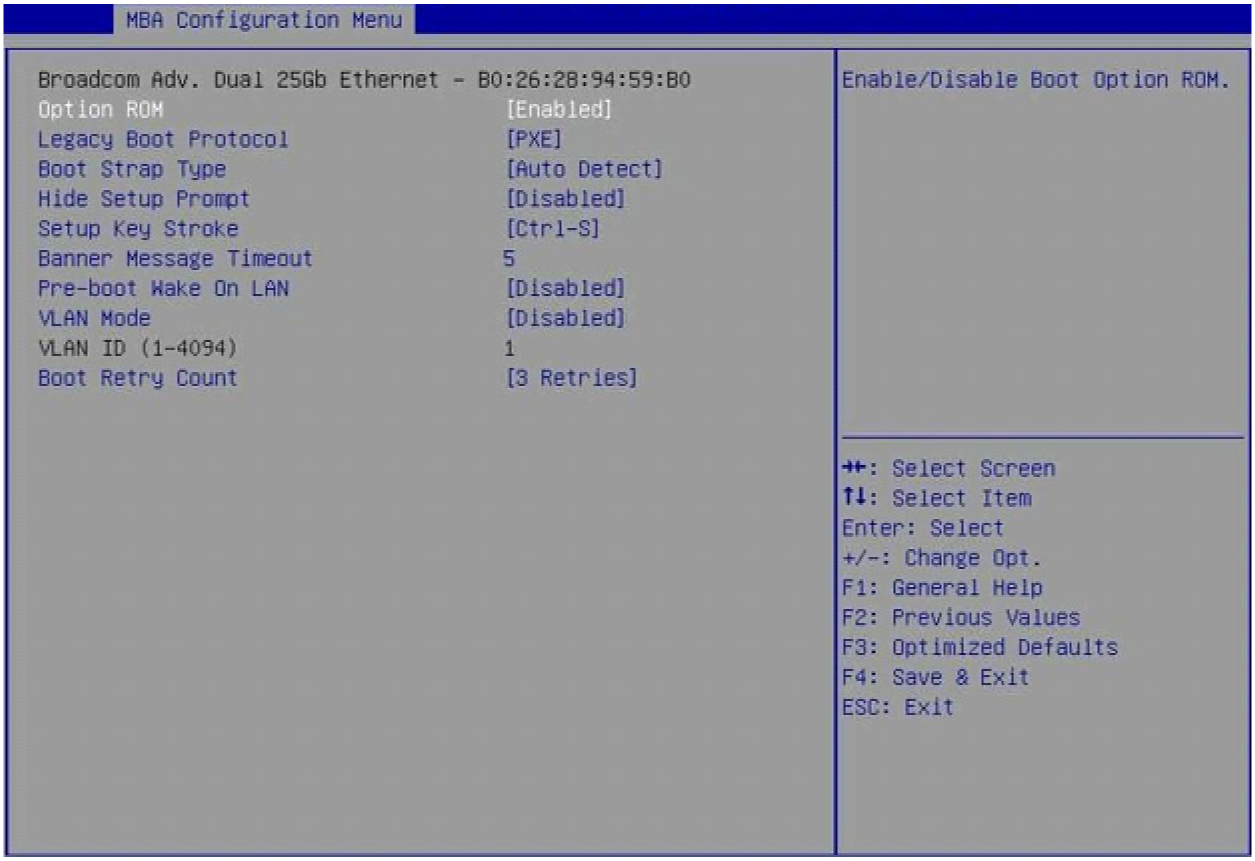
– Enable all ports

- **Disable ports 2, 3, and 4** – Available only on quad port adapters.
- **Disable ports 3 and >4** – Available only on quad port adapters.
- **Disable port 2 or 4** – Displayed as 2 on dual port adapters and as 4 on quad port adapters.

MBA Configuration Menu NIC Configuration

This menu presents legacy boot related parameters for configuration. Depending on the adapter type, some settings may not be available.

Figure 24: MBA Configuration Menu



```
MBA Configuration Menu

Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:94:59:B0     Enable/Disable Boot Option ROM.
Option ROM                            [Enabled]
Legacy Boot Protocol                  [PXE]
Boot Strap Type                       [Auto Detect]
Hide Setup Prompt                     [Disabled]
Setup Key Stroke                      [Ctrl-S]
Banner Message Timeout                5
Pre-boot Wake On LAN                  [Disabled]
VLAN Mode                             [Disabled]
VLAN ID (1-4094)                      1
Boot Retry Count                      [3 Retries]
                                                          →←: Select Screen
                                                          ↑↓: Select Item
                                                          Enter: Select
                                                          +/-: Change Opt.
                                                          F1: General Help
                                                          F2: Previous Values
                                                          F3: Optimized Defaults
                                                          F4: Save & Exit
                                                          ESC: Exit
```

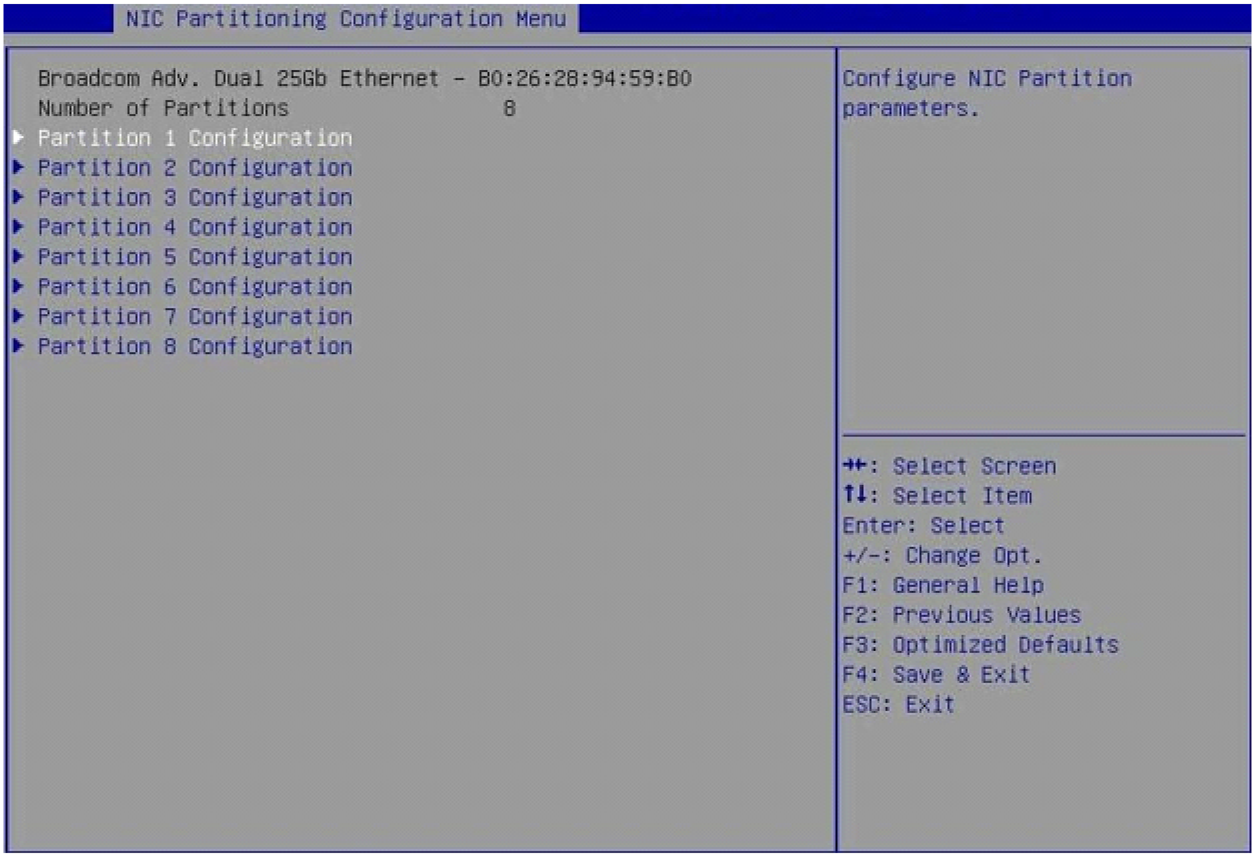
The MBA configurationNIC Configuration menu consists of the following items:

- **Option ROM**– This setting allows the user to control whether the Broadcom legacy option ROM driver should be advertised or not.
  - **Enabled**– Load legacy option ROM driver
  - **Disabled**– Do not load legacy option ROM driver
- **Legacy Boot Protocol**– This setting configures the type of boot to be performed in legacy mode.
  - **PXE**
  - **None**
- **Boot Strap Type**– This setting configures the bootstrap protocol for legacy PXE boot.
  - **Auto Detect**
  - **BBS**
  - **Int 18h**
  - **Int 19h**
- **Hide Setup Prompt**– This setting controls the visibility of the Broadcom legacy configuration setup menus (CCM). This setting is available only if legacy configuration menus are supported on the adapter.
  - **Enabled**– Do not display hotkey prompt to load CCM.
  - **Disabled**– Display hotkey prompt to load CCM.
- **Setup Key Stroke**– This setting allows users to select the hotkey combination for invoking the Broadcom legacy configuration setup menus (CCM). This setting is available only if legacy configuration menus are supported on the adapter.

  - **Ctrl-S**– Pressing the CTRL key along with S key will load CCM.
  - **Ctrl-B**– Pressing the CTRL key along with B key will load CCM.
- **Banner MessageTimeout**– This setting configures the duration for which the Broadcom legacy option ROM banner is to be displayed on the screen during POST. This is a numeric setting. The value must be specified in the range 0 to 15 seconds.
- **Pre-boot Wake on LAN**– This setting configures Wake on LAN (WoL) which is the ability to remotely power on a server or to wake it up from sleep mode. This setting is available only on adapters that support the Wake on LAN feature. This setting is available only on adapters that support the **Wake on LAN** feature.
  - **Enabled**– Turn on WoL
  - **Disabled**– Turn off WoL
- **VLAN Mode**– This setting configures a virtual LAN.
  - **Enabled** – Turn on VLAN mode
  - **Disabled** – Turn off VLAN mode
- **VLAN ID** – This setting configures a VLAN tag when VLAN mode is enabled. This is a numeric setting. The value must be specified in the range 1 to 4094.
- **Boot Retry Count** – This setting configures the number of times legacy boot must be attempted in case of failure.
  - No Retry
  - 1 Retry
  - 2 Retries
  - 3 Retries
  - 4 Retries
  - 5 Retries
  - 6 Retries
  - Indefinite Retries
- **Permit Total Port Shutdown** – This feature is supported on Linux OS only. This setting allows the port to be completely disabled when a port down command is received from the host OS or drive. This feature is not supported when virtualization mode is NPAR or NPAR + SRIOV.
  - **Enabled**– Permit Total Port Shutdown
  - **Disabled**– Permit Total Port Shutdown

NIC Partitioning Configuration Menu

Figure 25: NIC Partitioning Configuration Menu



```
NIC Partitioning Configuration Menu

Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:94:59:B0
Number of Partitions                8
▶ Partition 1 Configuration
▶ Partition 2 Configuration
▶ Partition 3 Configuration
▶ Partition 4 Configuration
▶ Partition 5 Configuration
▶ Partition 6 Configuration
▶ Partition 7 Configuration
▶ Partition 8 Configuration

Configure NIC Partition
parameters.
```

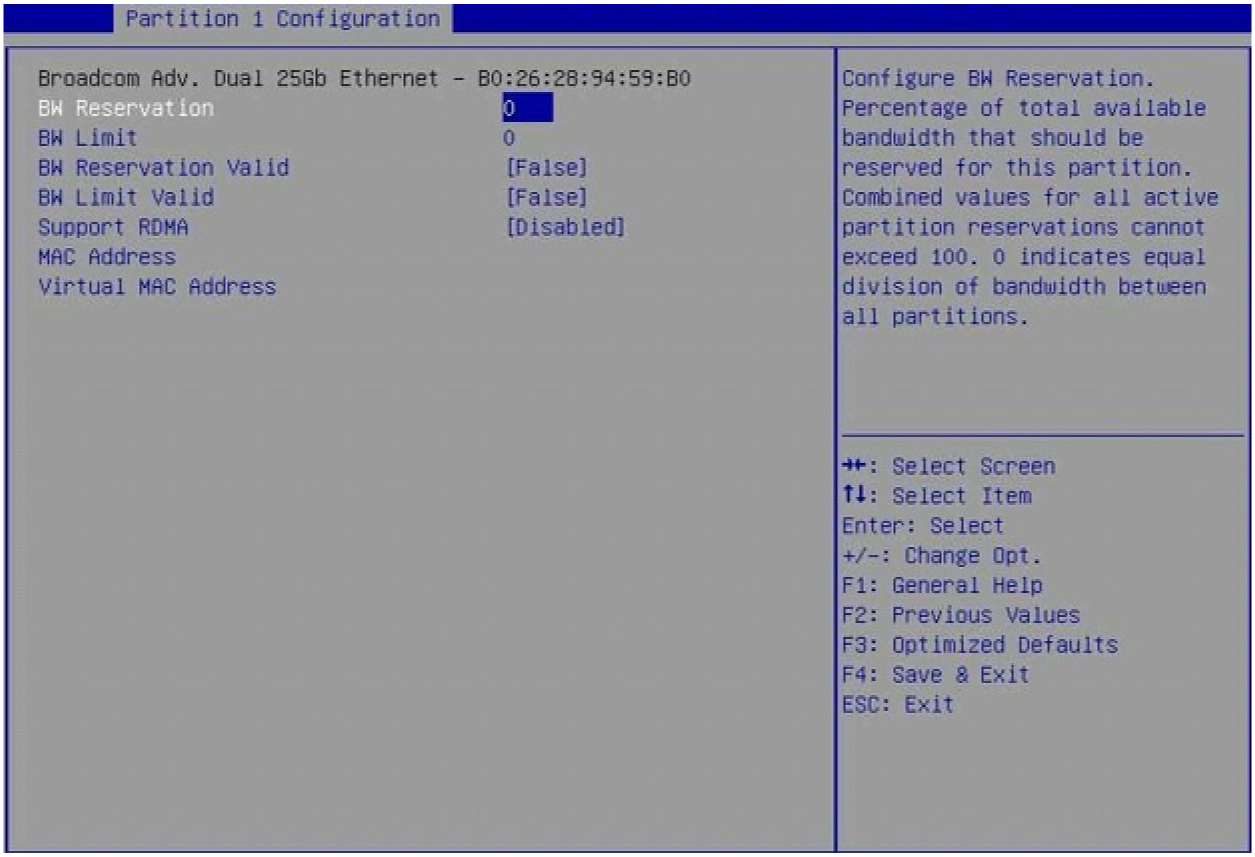
The **NIC Partitioning Configuration** screen has the following sub-menus:

- **Number of Partitions Per Port** – This field displays the number of PCI Physical functions currently enabled on the current network port when MF mode is set to NPAR. This is a read-only field.
- **Partition <n>Configuration** – This menu presents configuration parameters for the ‘n’th partition on the network port in NPar mode. The number of menus presented depends on the value of the **Number of Partitions** field.

Partition n Configuration Menu

This menu presents partition related parameters for configuration. Depending on the adapter type, some settings may not be available.

Figure 26: Partition n Configuration Menu



```
Partition 1 Configuration

Broadcom Adv. Dual 25Gb Ethernet - B0:26:28:94:59:B0
BW Reservation                      0
BW Limit                            0
BW Reservation Valid                [False]
BW Limit Valid                      [False]
Support RDMA                        [Disabled]
MAC Address
Virtual MAC Address
```

Configure BW Reservation. Percentage of total available bandwidth that should be reserved for this partition. Combined values for all active partition reservations cannot exceed 100. 0 indicates equal division of bandwidth between all partitions.

- **BW Reservation** – This setting configures the percentage of total available bandwidth that should be reserved for this partition. The total Bandwidth Reservation assigned for all active partitions cannot exceed 100. A value of 0 on all partitions indicates equal division of bandwidth between all partitions. This setting is available only on adapters that support the Bandwidth Reservation feature. This is a numeric setting. The value must be specified in the range 0 to 100.
- **BW Limit** – This setting configures the maximum percentage of available bandwidth this partition is allowed. This is a numeric setting. The value must be specified in the range 0 to 100.
- **BW Reservation Valid** – This setting configures whether BW Reservation is applicable on the current partition. When this setting is disabled, the BW Reservation value for the current partition will be ignored. This setting is available only on adapters that support the Bandwidth Reservation feature.
  - **Enabled** – Turn on BW Reservation
  - **Disabled** – Turn off BW Reservation
- **BW Limit Valid** – This setting configures whether BW Limit is applicable on the current partition. When this setting is disabled, the BW Limit value for the current partition will be ignored.
  - **Enabled** – Turn on BW Limit
  - **Disabled** – Turn off BW Limit
- **Support RDMA** – This setting configures the RoCE support for the current partition. This setting is available only when RDMA is supported on the current partition. This setting will be displayed on the Device Configuration menu in SF mode and in the **NIC Partition Configuration** menu in NPar mode.

  - **Enabled** – Turn on RDMA
  - **Disabled** – Turn off RDMA
- **MAC Address** – This field displays the Permanent MAC address assigned during manufacturing for the current partition. This is a read-only field.
- **Virtual MAC Address** – This field displays the Virtual MAC address assigned to the current partition. This is a read-only field from the HII menu. The value for this parameter can be configured using remote utilities.

BIOS Comprehensive Configuration Management

For systems with legacy BIOS, preboot configuration can be configured using the Comprehensive Configuration Management (CCM) menu option. During the system BIOS POST, the Broadcom banner message is displayed with an option to change the parameters through the Control-S menu. When Control-S is pressed, a device list is populated with all the Broadcom network adapters found in the system. Select the desired NIC for configuration.

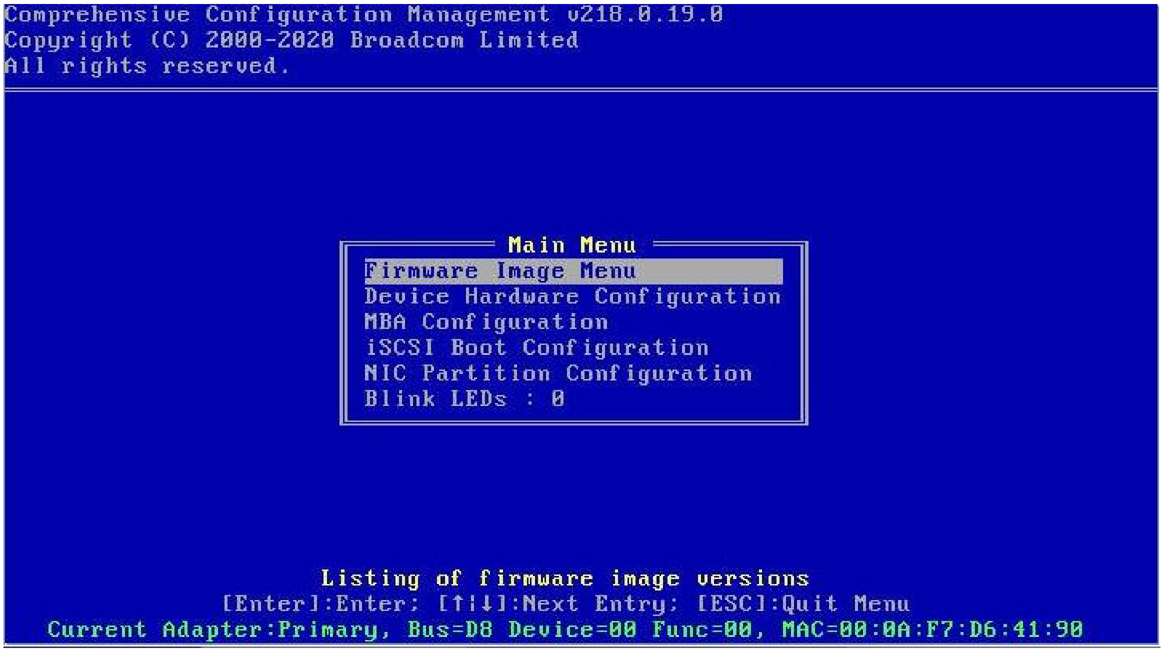
> **NOTE**
> Some adapters may not have CCM firmware in the NVRAM image and must use the HII menu to configure legacy parameters.

The BIOS Comprehensive Configuration Management menu consists of the following menus:

- Main Menu
- Firmware Image Menu
- Device Hardware Configuration Menu
- MBA Configuration Menu
- NIC Partitioning Configuration Menu
- Exiting CCM

Main Menu

Figure 27: CCM Main Menu



```
Comprehensive Configuration Management v218.0.19.0
Copyright (C) 2000-2020 Broadcom Limited
All rights reserved.
```

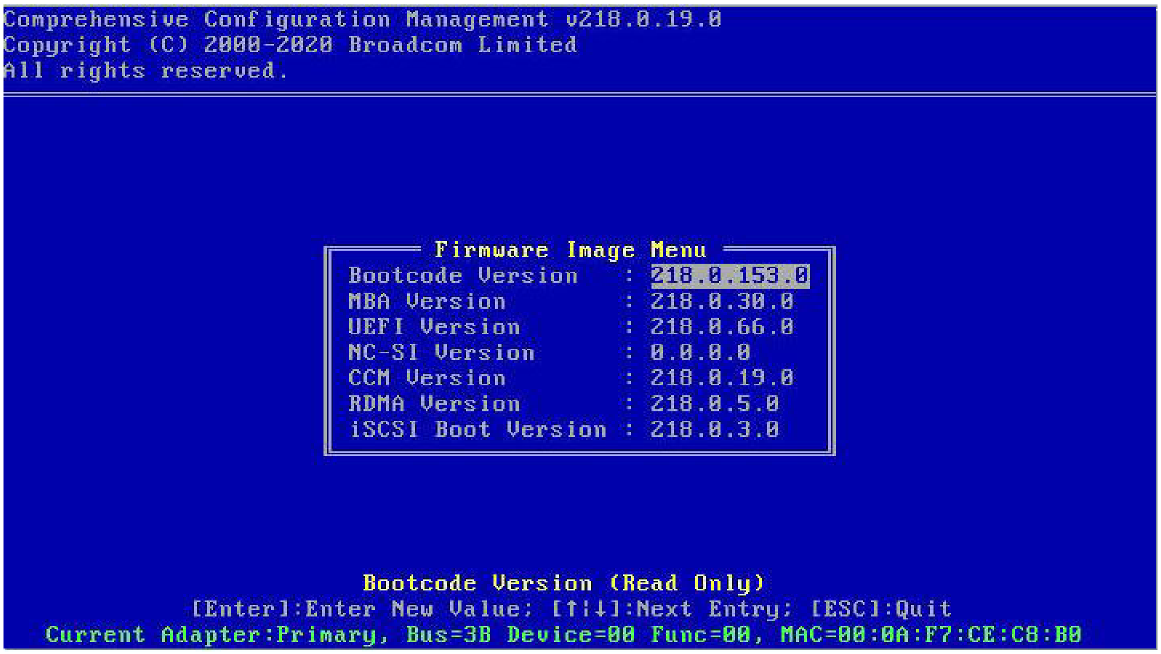
```
Listing of firmware image versions
[Enter]:Enter; [↑|↓]:Next Entry; [ESC]:Quit Menu
Current Adapter:Primary, Bus=D8 Device=00 Func=00, MAC=00:0A:F7:D6:41:90
```

The **Main menu** consists of the following items:

- **Firmware Image Menu** – This menu presents the various component versions present in the current firmware build.
- **Devic Hardware Configuration** – This menu presents adapter specific parameters for configuration.
- **MBA Configuration** – This menu presents PXE boot related parameters for configuration.
- **NICPartitioning Configuration** – This menu presents NIC partition related parameters for configuration.
- **Blink LEDs** – This setting configures the duration for which the LEDs on the physical network port should blink to assist with port identification. This is a numeric setting. The value must be specified in the range 0 to 15 seconds.

Firmware Image Menu

Figure 28: Firmware Image Menu



```
Comprehensive Configuration Management v218.0.19.0
Copyright (C) 2000-2020 Broadcom Limited
All rights reserved.

                  Firmware Image Menu
              Bootcode Version   : 218.0.153.0
              MBA Version        : 218.0.30.0
              UEFI Version       : 218.0.66.0
              NC-SI Version      : 0.0.0.0
              CCM Version        : 218.0.19.0
              RDMA Version       : 218.0.5.0
              iSCSI Boot Version : 218.0.3.0

                 Bootcode Version (Read Only)
         [Enter]:Enter New Value; [↑↓]:Next Entry; [ESC]:Quit
   Current Adapter:Primary, Bus=3B Device=00 Func=00, MAC=00:0A:F7:CE:C8:B0
```

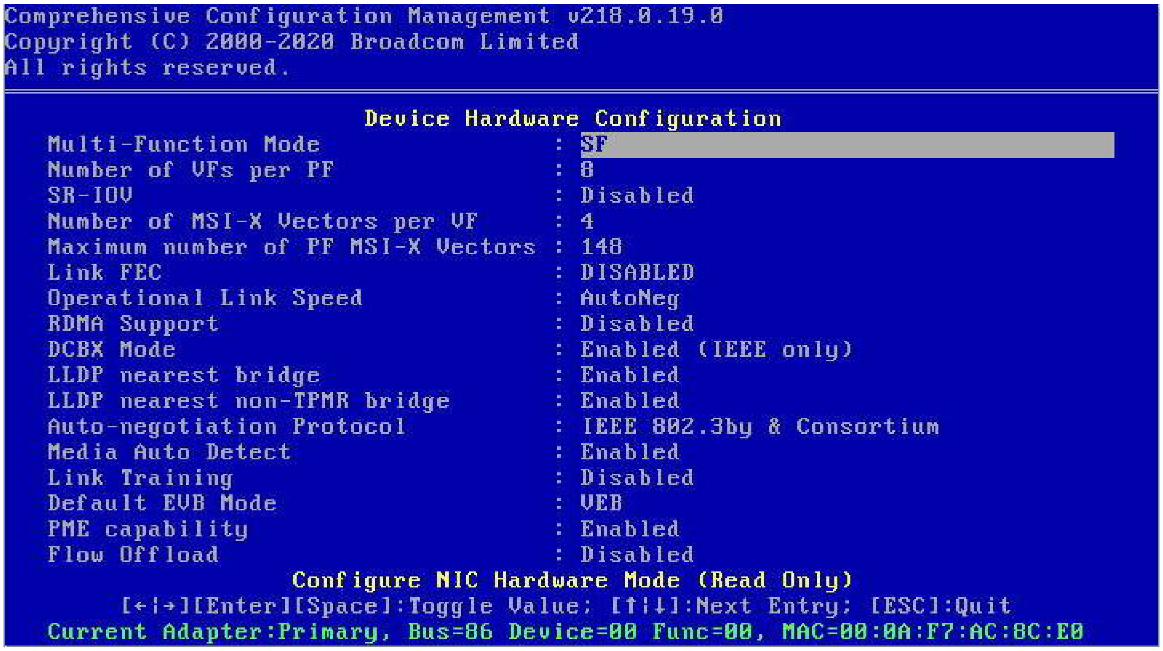
The firmware image menu consists of the following items:

- **Bootcode Version** – This field displays the firmware boot code version.
- **MBA Version** – This field displays the legacy pre-boot driver version.
- **UEFI Version** – This field displays the UEFI pre-boot driver version.
- **NC-SI Version** – This field displays the NCSI version.
- **CCM Version** – This field displays the CCM version.
- **RDMA Version** – This field displays the RoCE firmware version.

Device Hardware Configuration Menu

This menu presents adapter specific parameters for configuration. Depending on the adapter type, some settings are not available.

Figure 29: Device Hardware Configuration Menu



```
Comprehensive Configuration Management v218.0.19.0
Copyright (C) 2000-2020 Broadcom Limited
All rights reserved.

                      Device Hardware Configuration
   Multi-Function Mode                  : SF
   Number of VFs per PF                 : 8
   SR-IOV                               : Disabled
   Number of MSI-X Vectors per VF       : 4
   Maximum number of PF MSI-X Vectors   : 148
   Link FEC                             : DISABLED
   Operational Link Speed               : AutoNeg
   RDMA Support                         : Disabled
   DCBX Mode                            : Enabled (IEEE only)
   LLDP nearest bridge                  : Enabled
   LLDP nearest non-TPMR bridge         : Enabled
   Auto-negotiation Protocol            : IEEE 802.3by & Consortium
   Media Auto Detect                    : Enabled
   Link Training                        : Disabled
   Default EVB Mode                     : VEB
   PME capability                       : Enabled
   Flow Offload                         : Disabled
                    Configure NIC Hardware Mode (Read Only)
        [←|→][Enter][Space]:Toggle Value; [↑|↓]:Next Entry; [ESC]:Quit
   Current Adapter:Primary, Bus=86 Device=00 Func=00, MAC=00:0A:F7:AC:8C:E0
```

The device hardware configuration consists of the following items:

- **Multi-Function Mode** – This setting configures the type of virtualization to be used by the controller on all ports. This is available only when NPAR is supported on the adapter.
  - **Single Function Mode (SF)** – In this mode, a single PCIe PF is assigned to each network port.
  - **Network Partitioning Mode (NPAR)** –This mode allows a single physical network port to appear to the system as multiple network device functions. Each PF or partition is assigned a separate PCIe function ID on initial power on, and a menu is made visible in setup to configure each partition. The 16 configurable partitions are distributed equally across the network ports.
- **Number of VFs per PF** – This setting allows the user to configure the number of PCI Virtual Functions Advertised by the port in PCI config space when SR-IOV is enabled in non-NPAR mode. This setting is available for configuration only in non-NPar mode. The maximum number of virtual functions supported by the adapter are shared across the number of physical ports on the adapter. This value must be specified only in multiples of 8.
- **SR-IOV –** This setting configures Single Root - I/O Virtualization (SR-IOV) which allows different virtual machines (VMs) in a virtual environment to share a single PCI Express hardware interface. This setting is available only on adapters which support SR-IOV.
  - **Enabled** – Enable support for SR-IOV
  - **Disabled** – Disable support for SR-IOV
- **Number of MSI-X Vectors per VF**

  This setting allows the user to configure the MSI-X Vectors per VF. Message Signaled Interrupts (MSI) are an alternative in-band method of signaling an interrupt, using special in-band messages to replace traditional out-of-band assertion of dedicated interrupt lines. This is a numeric setting. The value must be specified in the range 0 to 128.
- **Maximum Number of PF MSI-X Vectors** – This setting allows the user to configure the Maximum Number of MSI-X Vectors for a physical function. This is a numeric setting. The minimum value for this setting is 0. The maximum value varies across adapters.
- **Link FEC** – This setting configures the Forward Error Correction (FEC) mode which is a technique used for controlling errors in data transmission over unreliable or noisy communication channels. This option is useful when longer fiber cables are utilized. This setting is not available on 10G BaseT controllers. Only a subset of the possible values display on some adapters, based on the configuration. Possible values are:

  – Disabled

- **CL74** – Fire Code
- **CL91** – Reed Solomon
- **RS544** – RS544, using 1 x N RS
- **RS272** – RS272, using 1 x N RS
- **RS544** – RS544, using 2 x N RS
- **RS544** – RS544, using 2 x N RS

- **NOTE**
  When Media Auto Detect is enabled and Firmware Link Speed is set to Auto-negotiation, FEC is negotiated based on the link partner advertisement.

Energy Efficient Ethernet – This setting configures the Energy Efficient Ethernet (EEE) mode that allow for less power consumption during periods of low-data activity. This setting is available only on 10GBASE-T controllers.

- **Enabled** – Turn on EEE mode
- **Disabled** – Turn off EEE mode

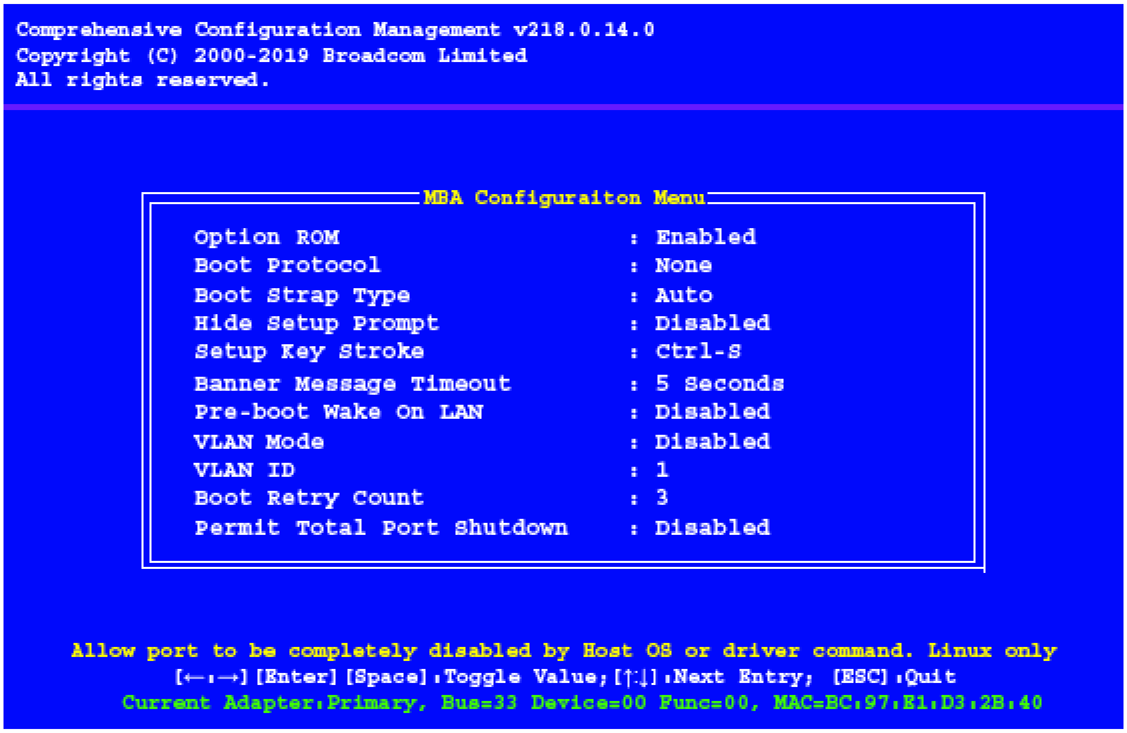
- **Operational Link Speed** – This setting configures the default link speed for pre-OS environment in full-power (D0) state. The possible values for this setting depends on the link speeds supported by the adapter. The value for this setting are fixed on some adapters based on the configuration.
- **Firmware Link Speed** – This setting configures the default link speed for pre-OS environment in sleep (D3) state. This setting may be grayed out on some adapters. The possible values for this setting depends on the link speeds supported by the adapter.
- **Support RDMA** – This setting configures Remote Direct Memory Access (RDMA) support on the port. RDMA is a technology that permits computers on a network to exchange data in main memory without the involvement of the processor, cache or operating system of either computer. RDMA allows high throughput and low-latency networking. This setting is available only when RDMA is supported on the adapter. This setting will be displayed on the Device Configuration menu in SF mode and in the NIC Partition Configuration menu in NPar mode.
  - **Enabled** – Turn on RDMA
  - **Disabled** – Turn off RDMA
- **Physical Media Selection** >– This setting configures the type of physical media to be used on the network port. This setting is available only on dual media devices.
  – SFP28
  – RJ-45
- **DCBX Mode** – This setting configures the Data Center Bridging (DCB) settings for the controller. Some of the following options may not be available depending on the adapter in use.
  - **Enabled (IEEE only)**
  - **CEE (only)**
  - **Both (IEEE preferred with fallback to CEE)**
- **LLDP neares bridge** – This setting configures the Link Layer Discovery Protocol (LLDP) which is a vendor-neutral link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a local area network based on IEEE 802 technology, principally wired Ethernet. An LLDP agent is a mapping of an entity where LLDP runs.
  - **Enabled** – Turn on LLDP nearest bridge
  - **Disabled** – Turn off LLDP nearest bridge
- **LLDP nearest non-TPMR bridge** – This setting configures the Link Layer Discovery Protocol (LLDP) which is a vendor-neutral link layer protocol used by network devices for advertising their identity, capabilities, and neighbors on a local area network based on IEEE 802 technology, principally wired Ethernet. An LLDP agent is a mapping of an entity where LLDP runs. This setting enables LLDP on the nearest non-TPMR bridge agent.
  - **Enabled** – Turn on LLDP nearest non-TPMRbridge
  - **Disabled** – Turn off LLDP nearest non-TPMRbridge
- **Auto-negotiation Protocol** – This setting configures the Auto-negotiation protocol for the adapter. Auto-negotiation is a feature that allows a port on a switch, router, server, or other device to communicate with the device on the other

end of the link to determine the optimal duplex mode and speed for the connection. This setting is not available on 10GBASE-T controllers.

  - **IEEE and BAM**
  - **IEEE and Consortium**
  - **BAM Only**
  - **Consortium Only**
  - **IEEE 802.3by**
- **Media Auto Detect** – This setting allows the user to configure the Media Auto Detect feature. This setting is not available on 10GBASE-T controllers.
  - **Enabled** – Turn on Media Auto Detect.
  - **Disabled** – Turn off Media Auto Detect.
- **Default EVB Mode** – This setting configures the Edge Virtual Bridging (EVB) mode which is an IEEE standard that involves the interaction between virtual switching environments in a hypervisor and the first layer of the physical switching infrastructure.
  - **VEB**
  - **VEPA**
  - **None**
- **Port Link Training** – This setting configures port link training when using force link speed. Link training should be enabled when using PAM-4. Link training should be disabled when the attached switch does not support link training.
  - **Enabled** – Port Link Training
  - **Disabled** – Port Link Training
- **Adapter Error Recovery** – This feature enables the recovery of firmware from fatal errors without manual intervention, host reboot, or power cycle. This feature is supported on Linux only.
  - **Enabled** –– Adapter Error Recovery
  - **Disabled** – Adapter Error Recovery
- **PME Capability** – This setting configures PME which is the ability to remotely wake a server using Power Management Event (PME). Devices supporting the native PCI Power Management can generate wakeup signals called PMEs to let the kernel know about external events requiring the device to be active.
  - **Enabled** – Turn on PME.
  - **Disabled** – Turn off PME.
- **Flow Offload** – This setting configures Open Virtual Switch (OVS) which is a multilayer virtual switch, designed to enable massive network automation through programmatic extension, while still supporting standard management interfaces and protocols (for example, NetFlow, sFlow, IPFIX, RSPAN, CLI, LACP, IEEE 802.1ag).
  - **Enabled** – Turn on Open Virtual Switch offload.
  - **Disabled** – Turn off Open Virtual Switch offload.
- **Port Enablement** – This setting configures the number of active ports (PCI functions) on the adapter. This setting is available only on adapters that support the Port Enablement feature.
  - Enable all ports
  - **Disable ports 2, 3, and 4 – Available only on quad port adapters.**
  - **Disable ports 3 and 4 – Available only on quad port adapters.**
  - **Disable port 2 or 4 – Displayed as 2 on dual port adapters and as 4 on quad port adapters.**

MBA Configuration Menu

Figure 30: Device Hardware Configuration Menu



```
Comprehensive Configuration Management v218.0.14.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                        MBA Configuraiton Menu

      Option ROM                              : Enabled
      Boot Protocol                           : None
      Boot Strap Type                         : Auto
      Hide Setup Prompt                       : Disabled
      Setup Key Stroke                        : Ctrl-S
      Banner Message Timeout                  : 5 Seconds
      Pre-boot Wake On LAN                    : Disabled
      VLAN Mode                               : Disabled
      VLAN ID                                 : 1
      Boot Retry Count                        : 3
      Permit Total Port Shutdown              : Disabled

   Allow port to be completely disabled by Host OS or driver command. Linux only
          [←:→][Enter][Space]:Toggle Value;[↑:↓]:Next Entry;  [ESC]:Quit
       Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

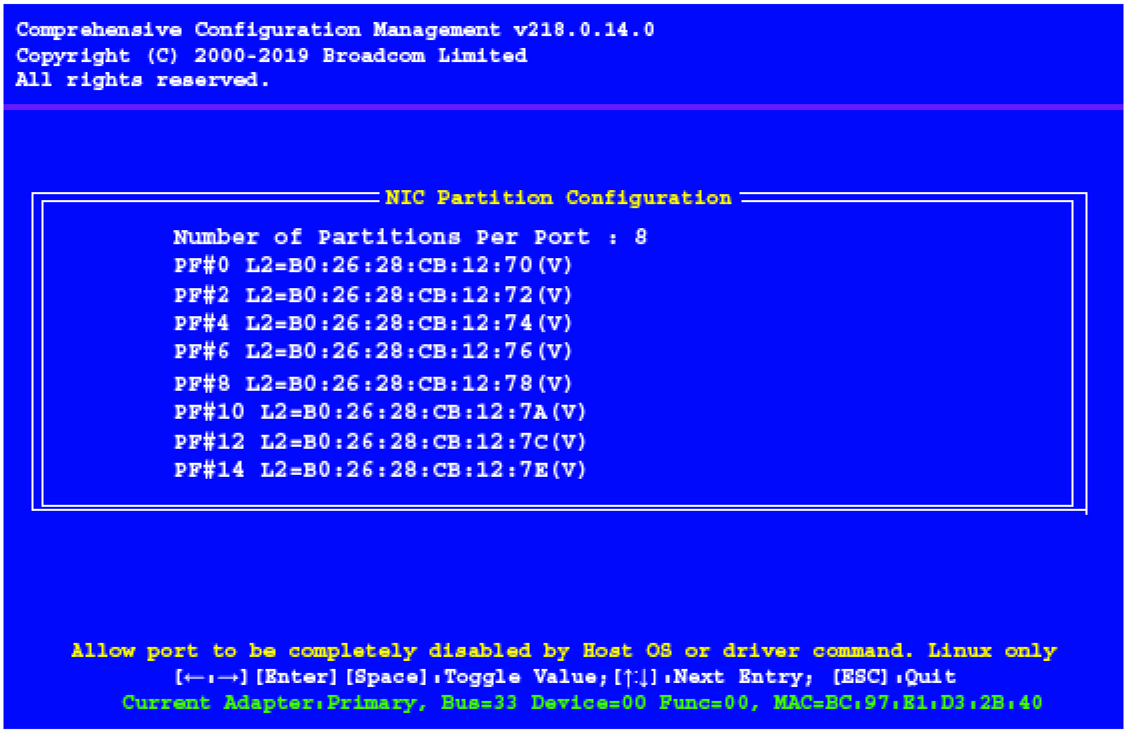
The MBA configuration menu consists of the following items:

- **Option ROM** – This setting allows the user to control whether the Broadcom legacy option ROM driver should be advertised or not.
  - **Enabled** – Load legacy option ROM driver
  - **Disabled** – Do not load legacy option ROM driver
- **Boot Protocol** – This setting configures the type of boot to be performed in legacy mode.
  - **PXE**
  - **None**
- **Boot Strap Type** – This setting configures the bootstrap protocol for legacy PXE boot.
  - **Auto Detect**
  - **BBS**
  - **Int 18h**
  - **Int 19h**
- **Hide Setup Prompt** – This setting controls the visibility of the Broadcom legacy configuration setup menus (CCM). This setting is available only if legacy configuration menus are supported on the adapter.
  - **Enabled** – Do not display hotkey prompt to load CCM.
  - **Disabled** – Display hotkey prompt to load CCM.
- **Setup Key Stroke** – This setting allows users to select the hotkey combination for invoking the Broadcom legacy configuration setup menus (CCM). This setting is available only if legacy configuration menus are supported on the adapter.

  - **Ctrl-S** – Press the CTRL key with the S key to load CCM.
  - **Ctrl-B** – Press the CTRL key with the B key to load CCM.
- **Banner Message Timeout** – This setting configures the duration for which the Broadcom legacy option ROM banner is to be displayed on the screen during POST. This is a numeric setting. The value must be specified in the range 0 to 15 seconds.
- **Wake on LAN** – This setting configures Wake on LAN (WoL) which is the ability to remotely power on a server or to wake it up from sleep mode. This setting is available only on adapters that support the Wake on LAN feature.
  - **Enabled** – Turn on WoL
  - **Disabled** – Turn off WoL
- **VLAN Mode** – This setting configures a virtual LAN.
  - **Enabled** – Turn on VLAN mode
  - **Disabled** – Turn off VLAN mode
- **VLAN ID** – This setting configures a VLAN tag when VLAN mode is enabled. This is a numeric setting. The value must be specified in the range 1 to 4094.
- **Boot Retry Count** – This setting configures the number of times legacy boot must be attempted in case of failure.
  - **No Retry**
  - **1 Retry**
  - **2 Retries**
  - **3 Retries**
  - **4 Retries**
  - **5 Retries**
  - **6 Retries**
  - **Indefinite Retries**
- **Permit** >**Total Port Shutdown** – This feature is supported on Linux OS only. This setting allows the port to be completely disabled when a port down command is received from the host OS or drive. This feature is not supported when virtualization mode is NPAR or NPAR + SRIOV.
  - **Enabled** – Permit Total Port Shutdown
  - **Disabled** – Permit Total Port Shutdown

NIC Partitioning Configuration Menu

Figure 31: NIC Partitioning Configuration Menu



```
Comprehensive Configuration Management v218.0.14.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                         NIC Partition Configuration
        Number of Partitions Per Port : 8
        PF#0  L2=B0:26:28:CB:12:70(V)
        PF#2  L2=B0:26:28:CB:12:72(V)
        PF#4  L2=B0:26:28:CB:12:74(V)
        PF#6  L2=B0:26:28:CB:12:76(V)
        PF#8  L2=B0:26:28:CB:12:78(V)
        PF#10 L2=B0:26:28:CB:12:7A(V)
        PF#12 L2=B0:26:28:CB:12:7C(V)
        PF#14 L2=B0:26:28:CB:12:7E(V)

   Allow port to be completely disabled by Host OS or driver command. Linux only
          [←,→][Enter][Space]:Toggle Value;[↑:↓]:Next Entry; [ESC]:Quit
      Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

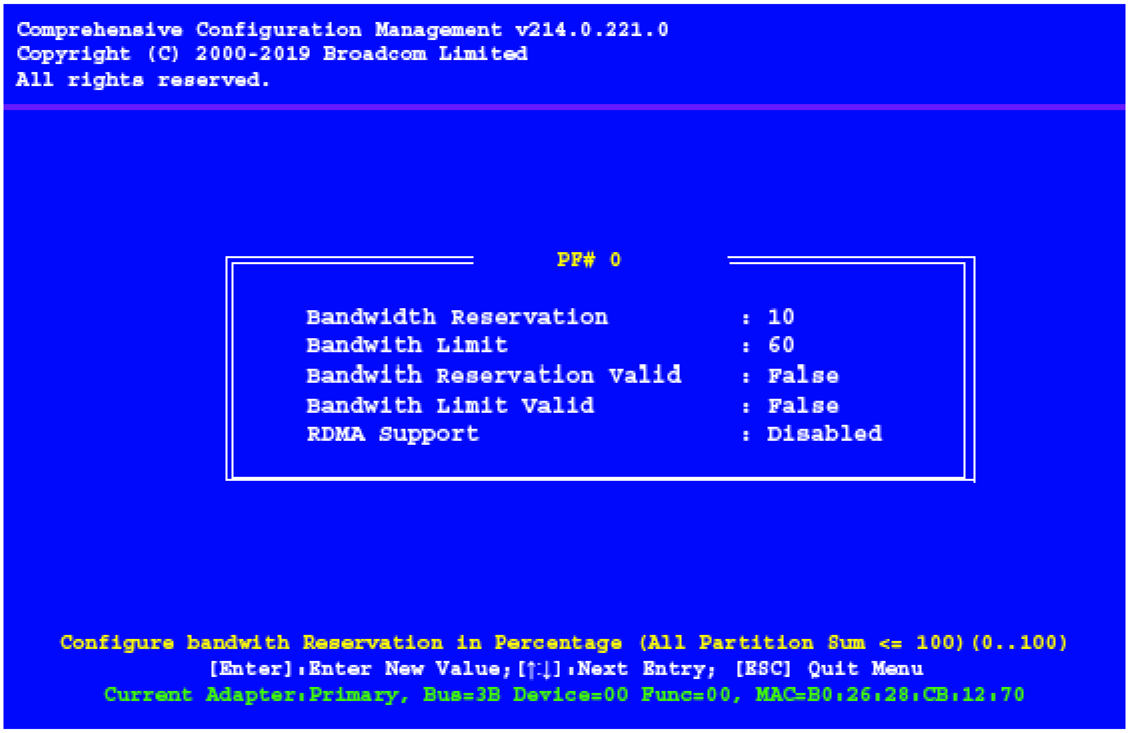
The **NIC Partitioning Configuration** screen has the following sub-menus:

- **Number of Partitions Per Port** – This field displays the number of PCI Physical functions currently enabled on the current network port when MF mode is set to NPAR. This is a read-only field.
- **Partition <n> Configuration** – This menu presents configuration parameters for the *nth* partition on the network port in NPar mode. The number of menus presented depends on the value of the **Number of Partitions** field.

Partition n Configuration

This menu presents partition related parameters for configuration. Depending on the adapter type, some settings may not be available.

Figure 32: Partition n Configuration Menu



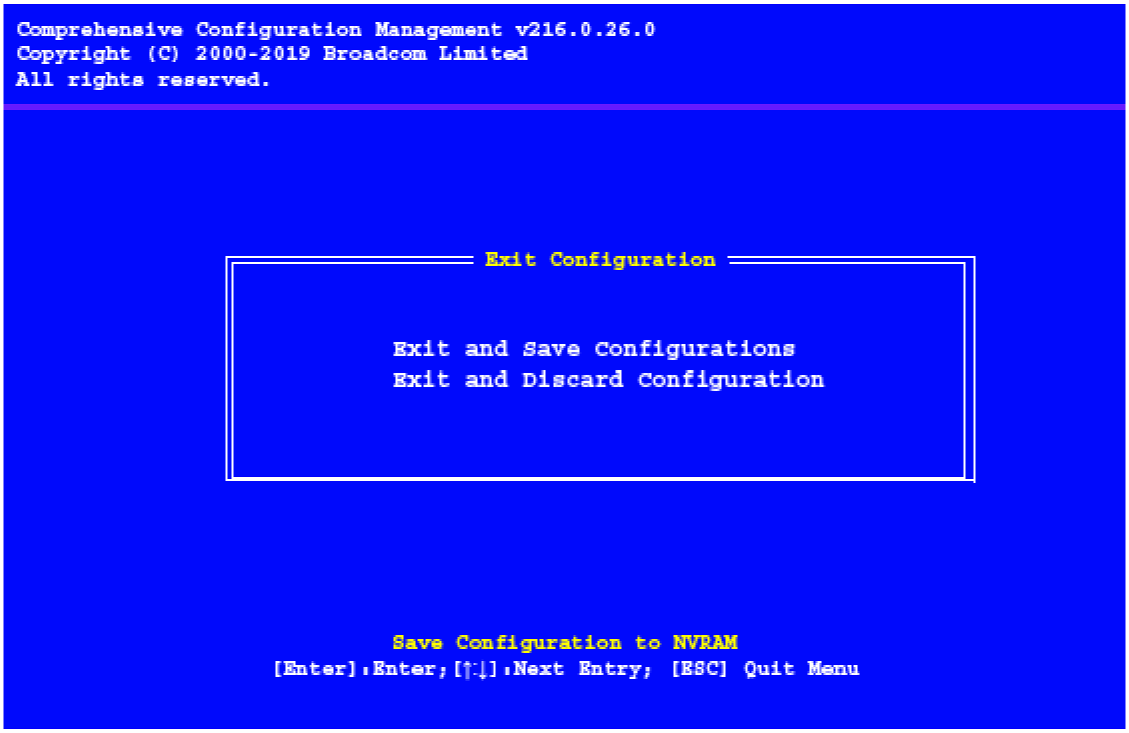
```
Comprehensive Configuration Management v214.0.221.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                                PF# 0

          Bandwidth Reservation            : 10
          Bandwith Limit                   : 60
          Bandwith Reservation Valid       : False
          Bandwith Limit Valid             : False
          RDMA Support                     : Disabled

  Configure bandwith Reservation in Percentage (All Partition Sum <= 100)(0..100)
            [Enter]:Enter New Value;[↑↓]:Next Entry; [ESC] Quit Menu
     Current Adapter:Primary, Bus=3B Device=00 Func=00, MAC=B0:26:28:CB:12:70
```

- **Bandwidth Reservation** – This setting configures the percentage of total available bandwidth that should be reserved for this partition. The total Bandwidth Reservation assigned for all active partitions cannot exceed 100. A value of 0 on all partitions indicates equal division of bandwidth between all partitions. This setting is available only on adapters that support the Bandwidth Reservation feature. This is a numeric setting. The value must be specified in the range 0 to 100.
- **Bandwidth Limit** – This setting configures the maximum percentage of available bandwidth this partition is allowed. This is a numeric setting. The value must be specified in the range 0 to 100.
- **Bandwidth Reservation Valid** – This setting configures whether BW Reservation is applicable on the current partition. When this setting is disabled, the BW Reservation value for the current partition will be ignored. This setting is available only on adapters that support the Bandwidth Reservation feature.
  - **Enabled** – Turn on BW Reservation
  - **Disabled** – Turn off BW Reservation
- **Bandwidth Limit Valid** – This setting configures whether BW Limit is applicable on the current partition. When this setting is disabled, the BW Limit value for the current partition will be ignored.
  - **Enabled** – Turn on BW Limit
  - **Disabled** – Turn off BW Limit
- **RDMA Support** – This setting configures the RoCE support for the current partition. This setting is available only when RDMA is supported on the current partition. This setting will be displayed on the Device Configuration menu in SF mode and in the **NIC Partition Configuration** menu in NPar mode.
  - **Enabled** – Turn on RDMA
  - **Disabled** – Turn off RDMA

Exiting CCM

To exit **CCM**, press **ESC**. Upon an attempt to exit CCM, if changes to any parameter values are detected, **Save** is prompted.

Figure 33: Exiting CCM



```
Comprehensive Configuration Management v216.0.26.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                                   Exit Configuration

                              Exit and Save Configurations
                              Exit and Discard Configuration

                              Save Configuration to NVRAM
                     [Enter]:Enter;[↑:↓]:Next Entry; [ESC] Quit Menu
```

User can choose to Save the configuration and exit CCM or Discard any changes and exit CCM. If Save is selected, the configuration changes, attempt to Exit CCM prompts a choice to continue the current boot or to reboot the system and apply any pending changes.

Software Installation

Provides instructions for installing software for Broadcom Ethernet NIC controllers under Linux, VMware, and Windows.

Software is comprised of the driver, firmware, library, and utility components. All components are bundled in files available from the Broadcom Ethernet network adapters site under the **Downloads** > **Driver** section located on each device page. This section is based on a ZIP file (Broadcom Ethernet NIC controller Linux Driver Installer) which is extracted to a directory referenced as follows as $REL_DIR.

The Software Installation section contains the following sections:

- Supported Operating Systems
- Installing the Linux Driver
- Installing the VMware Driver
- Installing the Windows Driver

Supported Operating Systems

The following table provides a list of supported operating systems.

Table 23: Supported Operating Systems

Operating System	Distribution	Binary Packages	Source Packages
RHEL/CentOS	7.9, 8.2, 8.2	x86 ARM	Yes
SuSE SLES	12 SP5, 15 SP1, 15 SP3	x86	Yes
Ubuntu	20.04.1	x86	Yes
vSphere/ESX	6.7, 6.7u3, 7.0, 7.0u1	x86	N/A
Windows	Windows 10	None	N/A
Windows Server	2016, 2019	x86	N/A
Citrix Hypervisor	8.2 LTSR	None	N/A

Installing the Linux Driver

This section contains the following information on installing the Linux driver:

- Installing the L2 and RoCE Drivers Using the Automated Installer
- Manual Software Installation and Configuration
- Linux Ethtool Commands

Installing the L2 and RoCE Drivers Using the Automated Installer

Included with the installation package is an automation tool for software installation and system configuration. This is the preferred installation method.

This section is based on a ZIP file (Broadcom Ethernet NIC controller Linux Installer) which is extracted to a directory referenced as follows as $REL_DIR.

The installer provides the following:

1. Installs the included drivers, firmware, RDMA library, and utility tools.

2. Configures interface priorities for IP and RoCE.
3. Provides switch configuration examples

Additional system packages are automatically installed as needed:

- ansible, libibversbs-utils, rdmacm-utils, perftest
- gcc, make, rpmbuild, kernel headers: if necessary to compile drivers or library

To start the automated installation with default values and all above components for an interface that is already up and has an IP address:

cd $REL_DIR/Linux/Linux_Installer

sudo bash install.sh -i <IFACE>

NOTE
<IFACE> must be replaced with the interface name or the device's PCIe address: For example, p1p1 or 41:00.0

To install only the L2/Ethernet driver without installing/configuring RoCE:

```
cd $REL_DIR/Linux/Linux_Installer
sudo bash install.sh -i <IFACE> -2
```

To start the automated installation with IP address and MTU specified with verbose output:

```
cd $REL_DIR/Linux/Linux_Installer
sudo bash install.sh -v -i <IFACE> -a <IP> -n <NETMASK> -m <MTU>
```

For a complete explanation of the automated installer including all options and modes of use, see the README file in $REL_DIR/Linux/Linux_Installer.

NOTE
The automated installer installs/updates both the L2 and RoCE drivers. In order to utilize the RoCE feature, see RoCE Configuration for additional configuration details.

Manual Software Installation and Configuration

The requirements to set up Linux RoCE and Linux RoCE congestion control can be divided into the following major steps:

- Installing Prerequisite Packages
- Building and Installing IP and RoCE Drivers
- Updating Initramfs
- IP Configuration
- Bring Up Interface
- User Space RDMA Library
- Configuring User Memory Limits
- Installing the bnxtqos Utility
- bnxtqos Utility
- LLDP Agent
- Installing the bnxtnvm Utility
- Upgrading the Firmware with bnxtnvm
- Upgrading Firmware with ethtool
- Updating Initramfs

Installing Prerequisite Packages

Use the Linux distribution's package manager to install prerequisites for compiling and using RDMA devices.

NOTE
It is recommended to use the latest version of the Linux distribution.

Debian/Ubuntu

```
sudo apt install -y automake autoconf libtool libibverbs-dev ibverbs-utils infiniband-diags perftest ethtool
```

Red Hat/Fedora/CentOS

```
sudo yum install -y libibverbs-devel qperf perftest infiniband-diags make gcc kernel kernel-devel autoconf
 aclocal libtool libibverbs-utils rdma-core-devel ibutils
```

Building and Installing IP and RoCE Drivers

Broadcom's Ethernet device driver (bnxt_en) provides an Ethernet interface and Broadcom's RDMA driver (`bnxt_re`) provides the RoCE interfaces. It is necessary to load both `bnxt_en` and `bnxt_re` to utilize the RDMA capability of Broadcom RNICs.

The `bnxt_en` and `bnxt_re` modules must come from the same package. Extract, compile, and install both drivers using the commands in the following sections.

Enable RDMA

Some Broadcom RNICs have RDMA disabled in the default factory configuration in order to provide the maximum resources possible to IP and DPDK applications. Ensure that the NIC has RDMA enabled using the following commands:

```
chmod +x $REL_DIR/bnxtnvm/Linux/bnxtnvm
sudo $REL_DIR/bnxtnvm/Linux/bnxtnvm -dev=<IFACE> setoption=support_rdma:<FUNCTION>
sudo reboot
```

Compile and install from Source RPM

```
cd $REL_DIR/Linux/KMP-L2-RoCE/KMP/<Distro>/<Distro Version>
sudo rpmbuild --rebuild bnxt_en-x.y.z.src.rpm
sudo rpm -i /root/rpmbuild/RPMS/x86_64/kmod-bnxt_en-x.y.z.rpm
sudo depmod -a
sudo modprobe bnxt_en
sudo modprobe bnxt_re
```

Compile and install directly from source:

```
tar xf netxtreme-bnxt_en-x.y.z.tar.gz
cd netxtreme-bnxt_en-x.y.z
sudo make
sudo make install
sudo depmod -a
sudo modprobe bnxt_en
sudo modprobe bnxt_re
```

Updating Initramfs

Most Linux distributions use a ramdisk image to store drivers for boot-up. These kernel modules will take precedence, so the initramfs must be updated after installing the new `bnxt_en` /`bnxt_re modules` :

Debian/Ubuntu

```
sudo update-initramfs -u
```

Red Hat/CentOS/Fedora

```
sudo dracut -f
```

IP Configuration

To configure the NIC, first determine the interface name using the following command:

```
dmesg | grep "$(lspci |grep BCM575 | cut -d' ' -f1)" |grep NIC
```

This command identifies the NIC interface name, connection speed, and PCIe bus:device:function.

Creating a Boot-Time Configuration File

Debian/Ubuntu

Add to /etc/network/interfaces:

```
auto <iface>
iface <iface> inet static
address w.x.y.z
netmask a.b.c.d
gateway e.f.g.h
```

NOTE
<iface> must be in the form of <iface>.<vlan id> to use VLANs.

Red Hat/Fedora/CentOS

```
Edit /etc/sysconfig/network-scripts/ifcfg-<iface>:
DEVICE=<iface>
ONBOOT=yes
BOOTPROTO=static
NM_CONTROLLED=no
NETMASK=a.b.c.d
IPADDR=w.x.y.z
# Uncomment below for VLAN
#VLAN=yes
#VLAN_EGRESS_PRIORITY_MAP=0:0,1:1,2:1,3:1,4:1,5:1,6:1,7:1
```

NOTE
<iface> must be in the form of <iface>.<vlan id> to use VLANs.

Bring Up Interface

After the interface is defined as above, bring UP the interface as:

```
sudo ifup <IFACE>
```

User Space RDMA Library

The `libbnxt_re library` provides the RDMA verbs interface to applications.

Removing the Conflicting Library File from the Linux Distribution

```
sudo (find /usr/lib64 -name libbnxt_re-rdmav\*.so; find /usr/lib -name libbnxt_re-rdmav\*.so) | xargs
-i{} mv {} {}.bak
```

Installing from Binary RPM

Installing from RPM is the preferred method.

```
rpm -i $REL_DIR/Linux/KMP-RoCE-Lib/KMP/<distro>/<distro version>/libbnxt_re-x.y.z.rpm
```

Compiling and Installing from Source RPM

If you would like to install from a source RPM, you may do so as below.

```
rpmbuild --rebuild $REL_DIR/Linux/KMP-RoCE-Lib/KMP/<distro>/<distro version>/libbnxt_rex.
y.z.src.rpm
sudo rpm -i /root/rpmbuild/RPMS/x86_64/libbnxt_re-x.y.z.rpm
```

Compiling and Installing from Source

The `libbnxt_re` library may also be installed directly from source.

```
cd $REL_DIR/Linux//KMP-RoCE-Lib
tar xf libbnxt_re-x.y.z.tar.gz
cd libbnxt_re-x.y.z
sh autogen.sh
./configure --sysconfdir=/etc
make
make install all
sudo sh -c "echo /usr/local/lib >> /etc/ld.so.conf"
sudo ldconfig
cp bnxt_re.driver /etc/libibverbs.d/
```

Configuring User Memory Limits

Non-root users typically need access to larger than usual amounts of locked memory, so the system defaults must be updated. Add the following lines to limits.conf:

/etc/security/limits.conf:

- soft memlock unlimited
- hard memlock unlimited

Installing the bnxtqos Utility

The `bnxtqos` utility provides RoCE configuration functions. Either `bnxtqos` or the LLDP agent must be installed. `bnxtqos` is the preferred tool.

Install the utility using the commands in the following sections:

RHEL/CentOS

```
sudo rpm -i bnxtqos/bnxtqos-<version>.rpm
```

Debian/Ubuntu

```
sudo dpkg -i bnxtqos/bnxtqos-<version>.deb
```

When using`bnxtqos` , ensure that the FIRMWARE-based DCBx is disabled with:

```
sudo bnxtnvm -dev=<IFACE> setoption=dcbx_mode:0:0
```

bnxtqos Utility

The Broadcom QoS configuration utility (`bnxtqos`) is a Broadcom utility that is used to set QoS mappings, priority flow control, and to configure ETS. `bnxtqos` only configures settings on the NIC and requires users to apply a symmetrical configuration on network switches.

The bnxtqos `utility` is used to set APPTLV's, and configure PFC and ETS:

```
bnxtqos -dev=<interface name> <command> [-options [...]]
```

Table 24: BnxtQoS Commands

Command	Description
`version`	Displays program version details.
`set_ets`	This command is used to configure the priority to TC and bandwidths. This command is also used to configure only TSA and Bandwidths.
	Syntax: `bnxtqos -dev=<interface name> set_ets tsa=<tc[0-7]:tc_type[ets/strict]> [priority2tc=<priority[0-7]:tc>] tcbw=<bandwidths>` with comma seperated and in units of percentage (%).
	Example: `bnxtqos -dev=p7p1 set_ets tsa=0:ets,1:ets,2:strict,3:strict,4:strict,5:strict,6:strict,7:strict priority2tc=0:0,1:0,2:0,3:0,4:0,5:1,6:0,7:0 tcbw=70,30`
`set_pfc`	This command enables PFC on given priority. Valid values are from 0 to 7.
	Syntax: `bnxtqos -dev=<interface name> set_pfc enabled=<0-7>` the values should be with comma seperated
	Example: `bnxtqos -dev=p7p1 set_pfc enabled=5,6`
`set_apptlv`	This parameter used to configure the priority of the APPTLV.
	Syntax: `bnxtqos -dev=<interfac name> set_apptlv app=<priority,selector,protocol>` **Syntax**: `bnxtqos -dev=<interfac name> set_apptlv -d app=<priority,selector,protocol>`
	Example: `bnxtqos -dev=p7p1 set_apptlv app=3,1,35093` **Example**: `bnxtqos -dev=p7p1 set_apptlv -d app=3,1,35093`
`get_qos`	This command is used to get the configured priorities and bandwidth parameters.
`ratelimit`	This command is used to set the rate limit for each TC in units of percentage (%).
	Example: `bnxtqos -dev=p7p1 ratelimit 80,60,70`
`dump`	This command is used to dump the supported dump strings. Supported dump string are pri2cos.
	Syntax: `bnxtqos -dev=<device name> dump <dump string>`
	Example: `bnxtqos -dev=p7p1 dump pri2cos`

Table 25: BnxtQoS Options

Option	Description
-v[v][v]	Enable extra console output (increase verbosity).
-V	Enable maximum verbosity of console output.
-d	Removes the application TLV.

bnxtqos Example

`bnxtqos` is used to adjust any RoCE QoS parameters. Use the LLDP agent to use DCBx on your network. For example, the ETS ratio may be changed to 50/50 and PFC disabled as:

```
sudo bnxtqos -dev=<IFACE> set_pfc enabled=none
sudo bnxtqos -dev=<IFACE> set_ets
tsa=0:ets,1:ets,2:strict,3:strict,4:strict,5:strict,6:strict,7:strict
priority2tc=0:0,1:0,2:0,3:0,4:0,5:1,6:0,7:0 tcbw=50,50
```

LLDP Agent

As an alternative to the Broadcom `bnxtqos` tool, the Linux LLDP agent is used to set QoS mappings and configure Enhanced Transmission Selection (ETS) to set bandwidth allocation ratios. The LLDP agent also allows the use of DCBx to import remote QoS settings. Either `bnxtqos` or LLDP Agent must be installed.

When using lldpagent, ensure that the FIRMWARE-based DCBx is disabled with:

```
sudo bnxtnvm -dev=<IFACE> setoption=dcbx_mode:0:0
```

Installing the LLDP Agent

Install the LLDP agent and lldptool utility using the commands in the following sections:

Debian/Ubuntu

```
sudo apt install libconfig9 libnl-3-200
wget http://ftp.us.debian.org/debian/pool/main/l/lldpad/lldpad_0.9.46-3.1_amd64.deb
sudo dpkg -i lldpad_0.9.46-3.1_amd64.deb
sudo systemctl enable lldpad
sudo systemctl start lldpad
```

Red Hat/Fedora/CentOS

```
sudo yum install lldpad
sudo systemctl enable lldpad
sudo systemctl start lldpad
```

When using LLDP agent, ensure that the FIRMWARE-based DCBx is disabled with:

```
sudo bnxtnvm -dev=<IFACE> setoption=dcbx_mode:0:0
```

bnxtnvm Utility

The Broadcom bnxtnvm utility allows setting non-volatile configuration elements of the RNIC, such as enabling/disabling RoCE, SR-IOV, and other options. bnxtnvm is the preferred tool for firmware upgrades.

Installing the bnxtnvm Utility

The bnxtnvm utility provides firmware and configuration functions. Install the utility using the following commands:

RHEL/CentOS

```
sudo rpm -i bnxtnvm/Linux/bnxtnvm-<version>.rpm
```

Debian/Ubuntu

```
sudo dpkg -i bnxtnvm/Linux/bnxtnvm-<version>.deb
```

Upgrading the Firmware with bnxtnvm

Upgrading firmware requires loading `bnxt_en` and installing bnxtnvm.

1. Upgrade the NIC firmware using the following commands:

   ```
   sudo ifup <interface>
   sudo bnxtnvm install $REL_DIR/NVRAM_Images/*.pkg
   ```

2. Reboot the system to allow the firmware upgrade to complete using the following command:

   ```
   sudo reboot
   ```

Upgrading Firmware with ethtool

The Linux ethtool utility may be used to upgrade Broadcom NIC firmware as follows:

1. Upgrade the NIC firmware using the following commands:

   ```
   sudo cp $REL_DIR/NVRAM_Images/<package>.pkg /lib/firmware
   sudo ethtool -f <interface> <package>.pkg
   ```

   NOTE
   Replace <package> with the appropriate file for your device. The filenames start with the product ID of the device, then board configuration. For example, the BCM957508-P2100G.pkg is for the 957508 (Thor) device in PCIe-card form factor with (2) 100 Gb/s ports.

2. Reboot the system to allow the firmware upgrade to complete using the following command:

   ```
   sudo reboot
   ```

Updating Initramfs

Most Linux distributions use a ramdisk image to store drivers for boot-up. These kernel modules will take precedence, so the initramfs must be updated after installing the new `bnxt_en` /`bnxt_re modules` :

Debian/Ubuntu

```
sudo update-initramfs -u
```

Red Hat/CentOS/Fedora

```
sudo dracut -f
```

Linux Ethtool Commands

In the following table, ethX should be replaced with the actual interface name.

Table 26: Linux Ethtool Commands

Command	Description
ethtool -s ethX speed 25000 autoneg off	Force the speed to 25G. If the link is up on one port, the driver does not allow the other port to be set to a different speed.
ethtool -i ethX	Output includes driver, firmware, and package version.
ethtool -k ethX	Show offload features.
ethtool -K ethX tso off	Turn off TSO.
ethtool -K ethX gro off lro off	Turn off GRO/LRO.
ethtool -g ethX	Show ring sizes.
ethtool -G ethX rx N	Set Ring sizes.
ethtool -S ethX	Get statistics.
ethtool -l ethX	Show number of rings.
ethtool -L ethX rx 0 tx 0 combined M	Set number of rings.
ethtool -C ethX rx-frames N	Set interrupt coalescing. Other parameters supported are: rx-usecs, rx-frames, rx-usecs-irq, rx-frames-irq, tx-usecs, tx-frames, tx-usecs- irq, tx-frames-irq.
ethtool -x ethX	Show RSS flow hash indirection table and RSS key.
ethtool -s ethX autoneg on speed 10000 duplex full	Enable Autoneg
ethtool --show-eee ethX	Show EEE state.
ethtool --set-eee ethX eee off	Disable EEE.
ethtool --set-eee ethX eee on tx-lpi off	Enable EEE, but disable LPI.
ethtool -L ethX combined 1 rx 0 tx 0	Disable RSS. Set the combined channels to 1.
ethtool -K ethX ntuple off	Disable Accelerated RFS by disabling ntuple filters.
ethtool -K ethX ntuple on	Enable Accelerated RFS.
ethtool -t ethX	Performs various diagnostic self-tests.
echo 32768 > /proc/sys/net/core/ rps_sock_flow_entries echo 2048 > /sys/class/net/ethX/queues/rx-X/ rps_flow_cnt	Enable RFS for Ring X.
sysctl -w net.core.busy_read=50	This sets the time to read the device's receive ring to 50 µsecs. For socket applications waiting for data to arrive, using this method can decrease latency by 2 or 3 usecs typically at the expense of higher CPU utilization.
echo 4 > /sys/class/net/<NAME>/device/sriov_numvfs	Enable SR-IOV with four VFs on named interface.
ip link set ethX vf 0 mac 00:12:34:56:78:9a	Set VF MAC address.
ip link set ethX vf 0 state enable	Set VF link state for VF 0.
ip link set ethX vf 0 vlan 100	Set VF 0 modprobe 8021q; ip link add link <NAME> name <VLAN Interface Name> type vlan id <VLAN ID> **Example:** modprobe 8021q; ip link add link ens3 name ens3.2 type vlan id 2

Installing the VMware Driver

The ESX drivers are provided in VMware standard VIB format and can be downloaded from VMware.com.

1. To install the Ethernet and RDMA driver, issue the following commands:

```
$ esxcli software vib install -v <bnxtnet>-<driver version>.vib
$ esxcli software vib install -v <bnxtroce>-<driver version>.vib
```

2. A system reboot is required for the new driver to take effect.

Other useful VMware commands are shown in the following table.

> **NOTE**
> In the following table, replace vmnicX with the actual interface name.

> **NOTE**
> $ kill -HUP $(cat /var/run/vmware/vmkdevmgr.pid)

This command is required after vmkload_mod bnxtnet for successful module bring up.

> **NOTE**
> NPAR + SR-IOV and NPAR + MultiRSS are currently not supported due to resource constraints.

Table 27: VMware Commands

Command	Description
esxcli software vib list |grep bnx	List the VIBs installed to see whether the bnxt driver installed successfully.
esxcfg-module –I bnxtnet	Print module info on to screen.
esxcli network get –n vmnicX	Get vmnicX properties.
esxcfg-module –g bnxtnet	Print module parameters.
esxcfg-module –s ‘multi_rx_filters=2 disable_tap=0 max_vfs=0,0 RSS=0’	Set the module parameters.
vmkload_mod –u bnxtnet	Unload bnxtnet module.
vmkload_mod bnxtnet	Load bnxtnet module.
esxcli network nic set –n vmnicX –D full –S 25000	Set the speed and duplex of vmnicX.
esxcli network nic down –n vmnicX	Disable vmnicX.
esxcli network nic up –n vmnic6	Enable vmnicX.
bnxtnetcli –s –n vmnic6 –S “25000”	Set the link speed. Bnxtnetcli is needed for older ESX versions to support the 25G speed setting.

Installing the Windows Driver

To install the Windows drivers:

1. Download the Windows driver installation package and unzip it.
2. Click **Server Manager** > **Tools** > **Computer Management** > **Device Manager**
3. Right-click on the Broadcom devices under **Network Adapters**.
4. Select **Update Driver**.
5. Select **Browse My Computer for driver Device Manager software** and select **Have Disk** to navigate to the folder where the driver files are located.
6. Click **Next** to update the driver automatically.
7. Reboot the system to ensure that the driver is running.

Driver Advanced Properties

The Windows driver advanced properties are shown in the following table.

Table 28: Windows Driver Advanced Properties

Driver Key	Parameters	Description
Encapsulated Task offload	Enable or Disable	Used for configuring NVGRE encapsulated task offload.
Flow control	TX or RX or TX/RX enable	Configure flow control on RX or TX or both sides.
Encapsulation Overhead	0 to 256	The maximum NVGRE header size from the beginning of the packet to the beginning of the inner packet payload.
Interrupt Moderation	Enable or Disable	Default Enabled. Allows frames to be batch processed by saving CPU time.
Jumbo packet	1514, 4088, 9014, or 9336	Jumbo packet size.
Large Send offload V2 (IPv4)	Enable or Disable	LSO for IPv4.
Large Send offload V2 (IPv6)	Enable or Disable	LSO for IPv6.
Locally Administered Address	User entered MAC address.	Override default hardware MAC address after OS boot.
Maximum Number of RSS Processors	Value 16	The maximum number of RSS processors.
Max number of RSS Queues	Value 1-8	Default is 8. Allows user to configure Receive Side Scaling queues.
Maximum RSS Processor Number	Value 16	The maximum processor number of the RSS interface.
NVGRE Encapsulatied Task Offload	Enable or Disable	Enable or disable support for Network Virualization using Generic Routing Encapsulation (NVGRE) Task Offload.
Packet Direct	Enable or Disable	Enable or disable support for Packet Direct
Preferred NUMA node	Value, Not Present	–
Priority and VLAN	Priority and VLAN Disable, Priority enabled, VLAN enabled, Priority and VLAN enabled.	Default Enabled. Used for configuring IEEE 802.1Q and IEEE 802.1P.
PTP Hardware Timestamp	Enable or Disable	Enable or disable Precision Timing Protocol hardware timestamping defined in IEEE 1588.
Quality of Service	Enable or Disable	Enable or disable support for Quality of Service (QoS) for IEEE 802.1 Data Center Bridging (DCB) for specifying the policies and settings of traffic classes that the network adapter uses for transmit, or egress, packet delivery.
Receive Buffer (0=Auto)	Increments of 500.	Default is Auto.
Receive Side Scaling	Enable or Disable.	Default Enabled.
Receive Segment Coalescing (IPv4)	Enable or Disable.	Default Enabled.
Receive Segment Coalescing (IPv6)	Enable or Disable.	Default Enabled.

RSS Base Processor Group	Value, Not Present	The base processor group for the processor number that is specified in RSS Base Processor Number.
RSS Base Processor Number	Value, Not Present	The number of the base RSS processor in the processor group.
RSS load balancing profile	NUMA scaling static, Closest processor, Closest processor static, conservative scaling, NUMA scaling.	Default NUMA scaling static.
RSS Max Processor Group	Value, Not Present	The maximum processor group.
Software Timestamp	Enable or Disable	Enable or disable Precision Timing Protocol software time stamping defined in IEEE 1588.
SR-IOV	Enable or Disable.	Default Enabled. This parameter works in conjunction with HW configured SR-IOV and BIOS configured SR-IOV setting.
TCP/UDP checksum offload IPv4	TX/RX enabled, TX enabled or RX Enabled or offload disabled.	Default RX and TX enabled.
TCP/UDP checksum offload IPv6	TX/RX enabled, TX enabled or RX Enabled or offload disabled.	Default RX and TX enabled.
Transmit Buffers (0=Auto)	Increment of 50.	Default Auto.
VF Spoofing Protection	Enable or Disable.	Enable or disable VF spoofing filter.
Virtual Machine Queue	Enable or Disable.	Default Enabled.
Virtual Switch RSS	Enable or Disable.	Enable or disable Receive Side Scaling (RSS) for a Virtual Switch.
VLAN ID	User configurable number.	Default 0.
VXLAN Encapsulatioed Task Offload	Enable or Disable.	Enable or disable Virtual Extensible LAN (VXLAN) offload.
Wake on Magic Packet	Enable or Disable.	enable or disable waking the system when a magic packet is received.
Wake on Pattern Match	Enable or Disable.	Enable or disable waking the system when a packet matching an OS defined pattern is received.

Event Log Messages

The event log messages are shown in the following table.

Table 29: Event Log Messages

Message ID	Comment
0x0001	Failed Memory allocation.
0x0002	Link Down Detected.
0x0003	Link up detected.
0x0009	Link 1000 Full.

0x000A	Link 2500 Full.
0x000b	Initialization successful.
0x000c	Miniport Reset.
0x000d	Failed Initialization.
0x000E	Link 10GbE successful.
0x000F	Failed Driver Layer Binding.
0x0011	Failed to set Attributes.
0x0012	Failed scatter gather DMA.
0x0013	Failed default Queue initialization.
0x0014	Incompatible firmware version.
0x0015	Single interrupt.
0x0016	Firmware failed to respond within allocated time.
0x0017	Firmware returned failure status.
0x0018	Firmware is in unknown state.
0x0019	Optics Module is not supported.
0x001A	Incompatible speed selection between Port 1 and Port 2. Reported link speeds are correct and might not match Speed and Duplex setting.
0x001B	Incompatible speed selection between Port 1 and Port 2. Link configuration became illegal.
0x001C	Network controller configured for 25 Gb full-duplex link.
0x001D	Network controller configured for 40 Gb full-duplex link.
0x001E	Network controller configured for 50 Gb full-duplex link.
0x001F	Network controller configured for 100 Gb full-duplex link.
0x0020	RDMA support initialization failed.
0x0021	Device's RDMA firmware is incompatible with this driver.
0x0022	Doorbell BAR size is too small for RDMA.
0x0023	RDMA restart upon device reset failed.
0x0024	RDMA restart upon system power up failed.
0x0025	RDMA startup failed. Not enough resources.
0x0026	RDMA not enabled in firmware.
0x0027	Start failed, a MAC address is not set.
0x0028	Transmit stall detected. TX flow control is disabled from now on.
0x0029	Querying admin or operational parameters is returning a length of zero in the SDH Header. This is a FW error but miniport is logging it.
0x002A	Is listening on address %3
0x002B	Stopped listening on address %3
0x002C	Connected to address %3
0x002D	Initiated connect to address %3
0x002E	Requested disconnect from address %3
0x002F	Received connect request from address %3
0x0030	Accepted connection from address %3
0x0031	Failed to connect to address %3

0x0032	Failed to accept connection from address %3
0x0033	Failed to resolve peer %3 MAC address
0x0034	Received disconnect request from address %3
0x0035	RDMA error detected on requester, qp:sp_state/err=%3
0x0036	RDMA error detected on responder, qp:sp_state/err=%3
0x0037	Maximum MTU size is %3, but user selected Jumbo Frame %4 and Encapsulation Overhead %5 for a total size of %6 bytes. Encapsulation overhead has been disabled and Jumbo Frame has been reduced to 1514.
0x0038	Maximum MTU size is %3, but user selected Jumbo Frame %4 and Encapsulation Overhead %5 for a total size of %6 bytes. Encapsulation overhead has been disabled.
0x0039	Maximum mtu size is %3. Driver requires maximum mtu of at least 1514 bytes.
0x003a	Firmware reset notification received.
0x003b	Firmware failed to execute de-register.
0x003c	Firmware accepted deregister successfully.
0x003d	Took too long to deregister from firmware.
0x003e	Firmware failed to respond to VER_GET after reset.
0x003f	Firmware became responsive.
0x0040	Firmware reset sequence completed.
0x0041	Firmware reset operation failed.
0x0042	All dependent VF(if any) deregistered.
0x0043	Some VF(s) attached to this PF is still registered with FW, giving up waiting. Some VF(s) may not come up after firmware reset.
0x0044	RR worker started processing reset sequence.
0x0045	Firmware fatal event notification received, recovery sequence will start.
0x0046	Adapter failed to restart from a firmware error or live upgrade. A system reboot may be required.

Updating the Firmware

Provides instructions for updating the firmware on Broadcom Ethernet NIC controllers using automated tools or manually.

The steps shown in this section use Linux to show how to update the adapter firmware, however, the steps used are the same for Windows and ESX.The bnxtnvm utility is used to update the adapter firmware on Windows, Linux, and ESX. On all operating systems, the bnxtnvm utility requires the network driver to be loaded and running in order to communicate with the adapter being upgraded. If the adapter being upgraded is not recognized by the bnxtnvm utility, the network driver running on the system is too old and does not support that adapter. In this case, the network driver must be updated using the driver provided on the Broadcom website prior to using bnxtnvm to update the adapter firmware.

NOTE
When upgrading the firmware on Linux, you must be logged in as root or use sudo when using bnxtnvm.There are multiple ways to update the firmware:

- Updating the Firmware Online (Windows/ESX/Linux) – The tool automatically downloads the firmware from the Broadcom website.
- Upgrading the Firmware with Automated Installer for Linux – The installer script updates the firmware along with the drivers.
- Manually Updating the Adapter Firmware on Linux/ESX – Manually update using the bnxtnvm tool.
- Updating the Adapter Firmware on Windows – Manually update using the bnxtnvm tool on Windows.

Updating the Firmware is divided into the following sections:

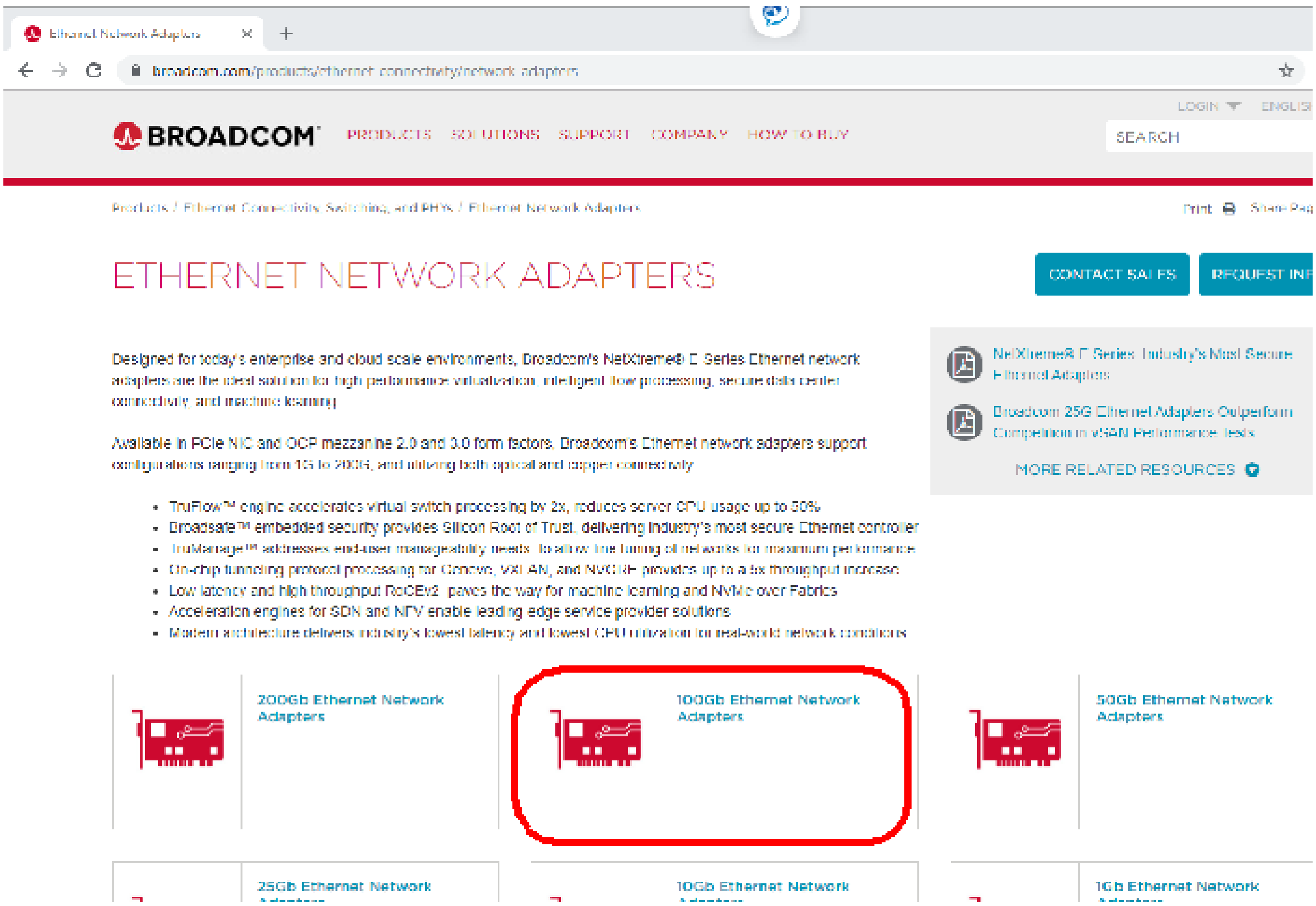
- Updating the Firmware Online (Windows/ESX/Linux)
- Upgrading the Firmware with Automated Installer for Linux
- Manually Updating the Adapter Firmware on Linux/ESX
- Updating the Adapter Firmware on Windows
- Verifying the Adapter Firmware

Updating the Firmware Online (Windows/ESX/Linux)

To update the firmware online, download the upgrade tool only from the Broadcom public website. Then the tool, when run on a server with internet access, downloads the appropriate firmware package from the Internet and installs it.

To update the firmware online:

1. Using a browser, access the following website: https://www.broadcom.com/products/ethernet-connectivity/network-adapters



Ethernet Network Adapters

broadcom.com/products/ethernet-connectivity/network-adapters

ETHERNET NETWORK ADAPTERS

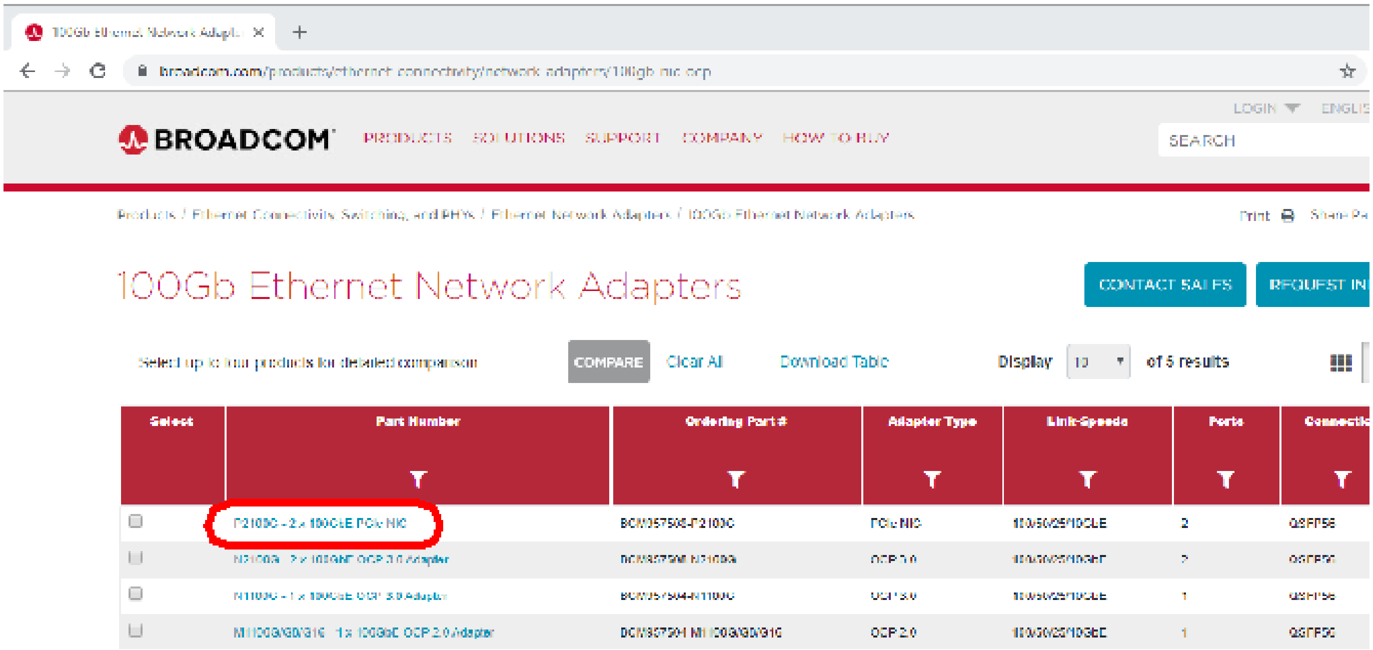
Designed for today's enterprise and cloud scale environments, Broadcom's NetXtreme® E Series Ethernet network adapters are the ideal solution for high performance virtualization, intelligent flow processing, secure data center connectivity, and machine learning.

Available in PCIe NIC and OCP mezzanine 2.0 and 3.0 form factors, Broadcom's Ethernet network adapters support configurations ranging from 1G to 200G, and utilizing both optical and copper connectivity.

- TruFlow™ engine accelerates virtual switch processing by 2x, reduces server CPU usage up to 50%
- Broadsafe™ embedded security provides Silicon Root of Trust, delivering industry's most secure Ethernet controller
- TruManage™ addresses end-user manageability needs to allow fine tuning of networks for maximum performance
- On-chip tunneling protocol processing for Geneve, VXLAN, and NVGRE provides up to a 5x throughput increase
- Low latency and high throughput RoCEv2 paves the way for machine learning and NVMe over Fabrics
- Acceleration engines for SDN and NFV enable leading edge service provider solutions
- Modern architecture delivers industry's lowest latency and lowest CPU utilization for real-world network conditions

NetXtreme® E Series: Industry's Most Secure Ethernet Adapters
Broadcom 25G Ethernet Adapters Outperform Competition in vSAN Performance Tests
MORE RELATED RESOURCES
200Gb Ethernet Network Adapters
100Gb Ethernet Network Adapters
50Gb Ethernet Network Adapters
25Gb Ethernet Network Adapters
10Gb Ethernet Network Adapters
1Gb Ethernet Network Adapters

2. Select the adapter according to your port speed. For example, 100GbE Network Adapter is selected.



Products / Ethernet Connectivity, Switching, and PHYs / Ethernet Network Adapters / 100Gb Ethernet Network Adapters

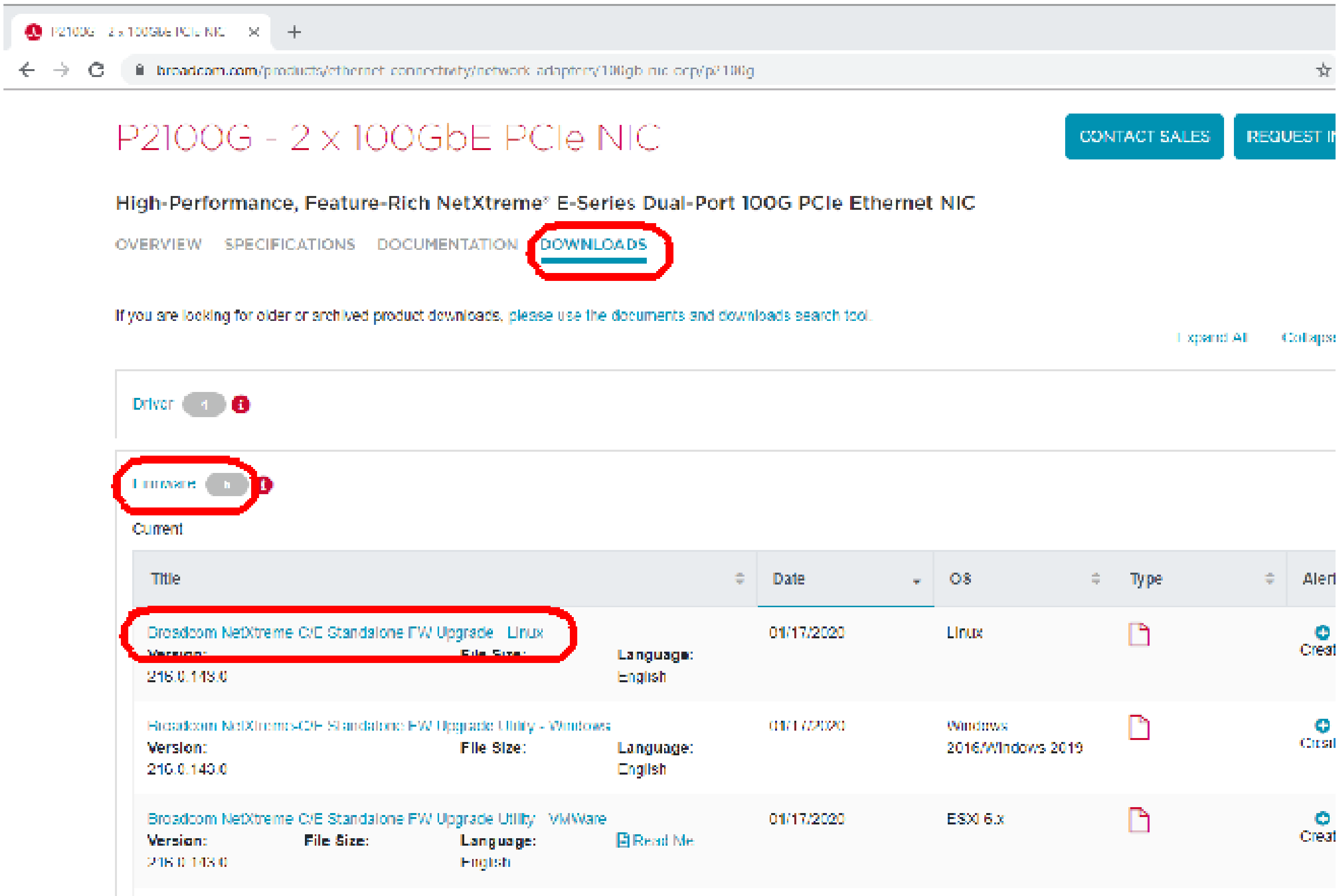
100Gb Ethernet Network Adapters

Select up to four products for detailed comparison

of 5 results

Select	Part Number	Ordering Part #	Adapter Type	Link-Speeds	Ports	Connectio
□	P2100G - 2 x 100GbE PCIe NIC	BCM957508-P2100G	PCIe NIC	100/50/25/10GbE	2	QSFP56
□	N2100G - 2 x 100GbE OCP 3.0 Adapter	BCM957508-N2100G	OCP 3.0	100/50/25/10GbE	2	QSFP56
□	N1100G - 1 x 100GbE OCP 3.0 Adapter	BCM957504-N1100G	OCP 3.0	100/50/25/10GbE	1	QSFP56
□	M1100G/G0/G1G - 1 x 100GbE OCP 2.0 Adapter	BCM957504-M1100G/G0/G1G	OCP 2.0	100/50/25/10GbE	1	QSFP56

3. Select your specific adapter. The P2100G is selected here as an example.
4. Click **Download** followed by **Firmware**.
5. Select **Broadcom NetXtreme-C/E Standalone FW Upgrade - Linux** and download the upgrade tool.


P2100G - 2 x 100GbE PCIe NIC
broadcom.com/products/ethernet-connectivity/network-adapters/100gb-nic-ocp/p2100g
CONTACT SALES
REQUEST IN

P2100G - 2 x 100GbE PCIe NIC

High-Performance, Feature-Rich NetXtreme® E-Series Dual-Port 100G PCIe Ethernet NIC

OVERVIEW SPECIFICATIONS DOCUMENTATION DOWNLOADS

If you are looking for older or archived product downloads, please use the documents and downloads search tool.

Expand All Collapse

Driver 4

Firmware 6

Current

Title	Date	OS	Type	Alert
Broadcom NetXtreme C/E Standalone FW Upgrade Linux Version: 216.0.143.0 File Size: Language: English	01/17/2020	Linux		Creat
Broadcom NetXtreme-C/E Standalone FW Upgrade Utility - Windows Version: 216.0.143.0 File Size: Language: English	01/17/2020	Windows 2016/Windows 2019		Creat
Broadcom NetXtreme C/E Standalone FW Upgrade Utility VMWare Version: 216.0.143.0 File Size: Language: English Read Me	01/17/2020	ESXi 6.x		Creat

6. Install the RPM file to install the tool and then run the tool with bnxtnvm install -online. Follow the instructions.

```
[root@dhcp-10-67-92-181 Downloads]# rpm -ihv bnxtnvm-216.0.143.0-Linux.rpm
 Preparing...################################# [100%]
Updating / installing...
1:bnxtnvm-216.0.143.0-1################################# [100%]
[root@dhcp-10-67-92-181 Downloads]# bnxtnvm install -online
Broadcom NetXtreme-C/E/S firmware update and configuration utility version v216.0.143.0 NetXtreme-E Controller
 #1 at PCI Domain:0000 Bus:01 Dev:00
```

Upgrading the Firmware with Automated Installer for Linux

This section is based on the Broadcom Ethernet NIC controller Linux Installer available from the Broadcom Ethernet Network Adapters site under the Downloads/Driver section located on each device page. Download and extract to a directory referenced as $REL_DIR in the following instructions.

To start the automated installation, see Installing the L2 and RoCE Drivers Using the Automated Installer.

NOTE
The automated installer automatically upgrades the firmware. To upgrade only the firmware, see Manual Software Installation and Configuration.

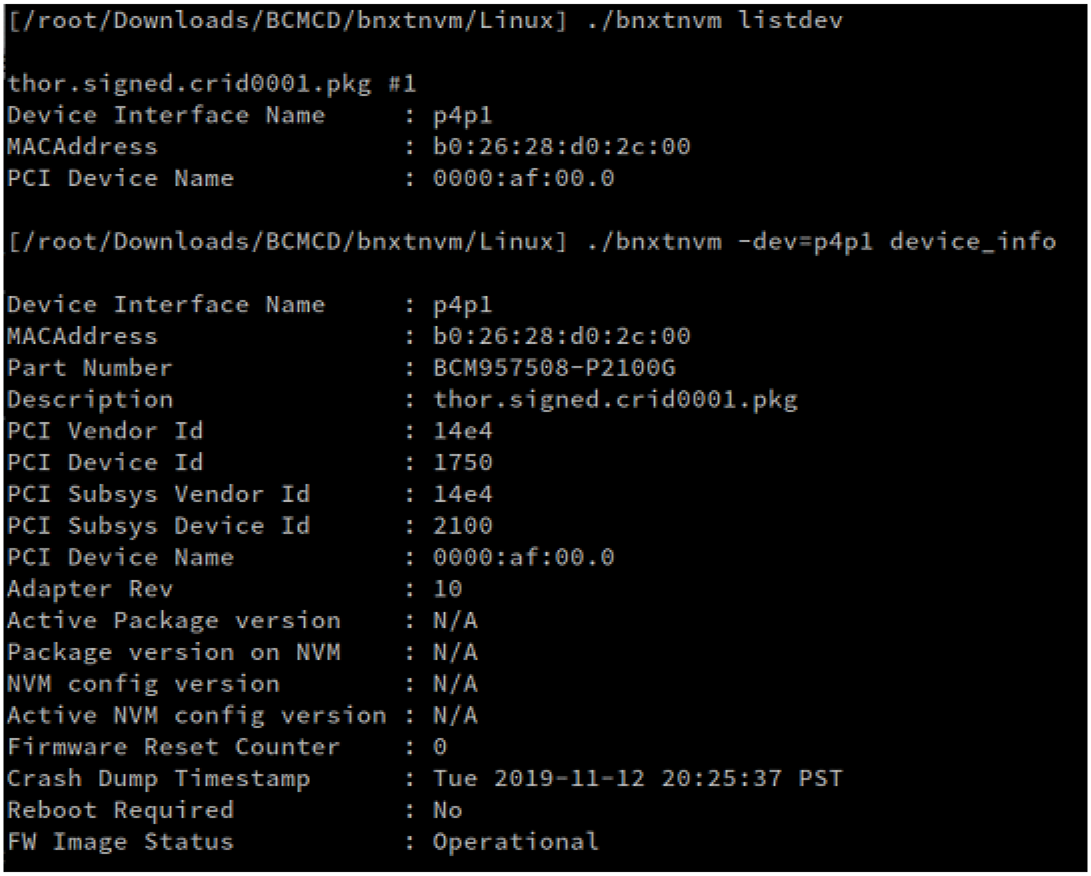
Manually Updating the Adapter Firmware on Linux/ESX

To update the adapter firmware on Linux/ESX:

1. List the adapters in the system using the following bnxtnvm listdev command:

   ```
   ./bnxtnvm listdev
   ```

   This command shows the device interface name that is used to identify the adapter in subsequent commands.

   ```
   [/root/Downloads/BCMCD/bnxtnvm/Linux] ./bnxtnvm listdev

   thor.signed.crid0001.pkg #1
   Device Interface Name     : p4p1
   MACAddress                : b0:26:28:d0:2c:00
   PCI Device Name           : 0000:af:00.0

   [/root/Downloads/BCMCD/bnxtnvm/Linux] ./bnxtnvm -dev=p4p1 device_info

   Device Interface Name     : p4p1
   MACAddress                : b0:26:28:d0:2c:00
   Part Number               : BCM957508-P2100G
   Description               : thor.signed.crid0001.pkg
   PCI Vendor Id             : 14e4
   PCI Device Id             : 1750
   PCI Subsys Vendor Id      : 14e4
   PCI Subsys Device Id      : 2100
   PCI Device Name           : 0000:af:00.0
   Adapter Rev               : 10
   Active Package version    : N/A
   Package version on NVM    : N/A
   NVM config version        : N/A
   Active NVM config version : N/A
   Firmware Reset Counter    : 0
   Crash Dump Timestamp      : Tue 2019-11-12 20:25:37 PST
   Reboot Required           : No
   FW Image Status           : Operational
   ```

2. After selecting the device interface name, the adapter part number can be viewed using the following bnxtnvm device_info command:

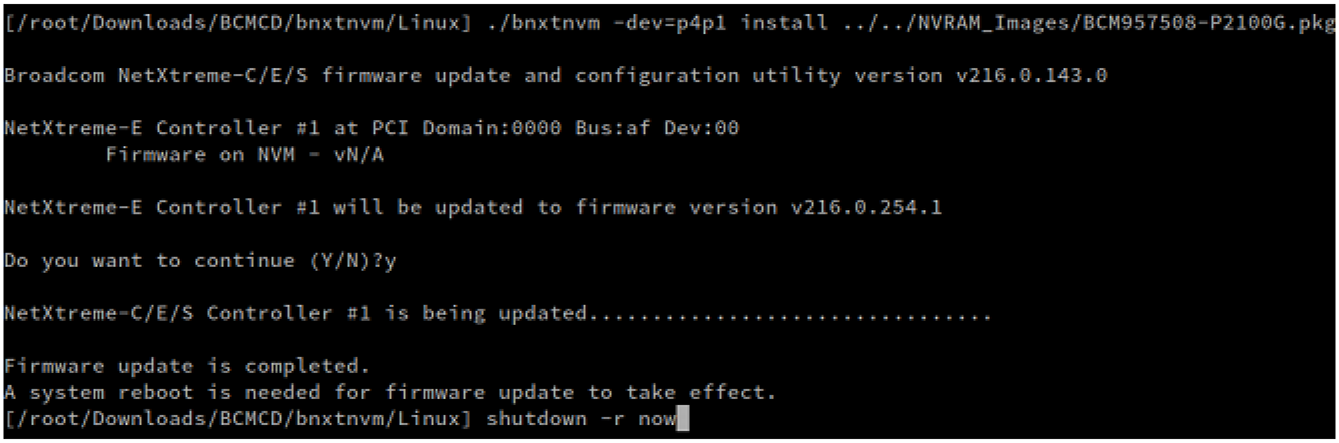
   ```
   ./bnxtnvm -dev=p4p1 device_info
   ```

   This command shows the part number used to select the firmware package to use in the upgrade.

3. The firmware is updated using the following bnxtnvm install command:

   ```
   ./bnxtnvm dev=p4p1 install ../../NVRAM_Images/BCM957508-P2100G.pkg
   ```

   The name of the firmware package used to update the adapter firmware is the part number displayed in the device_info command with a .pkg extension.



```
[/root/Downloads/BCMCD/bnxtnvm/Linux] ./bnxtnvm -dev=p4p1 install ../../NVRAM_Images/BCM957508-P2100G.pkg

Broadcom NetXtreme-C/E/S firmware update and configuration utility version v216.0.143.0

NetXtreme-E Controller #1 at PCI Domain:0000 Bus:af Dev:00
        Firmware on NVM - vN/A

NetXtreme-E Controller #1 will be updated to firmware version v216.0.254.1

Do you want to continue (Y/N)?y

NetXtreme-C/E/S Controller #1 is being updated................................

Firmware update is completed.
A system reboot is needed for firmware update to take effect.
[/root/Downloads/BCMCD/bnxtnvm/Linux] shutdown -r now
```

4. Reboot the system after the adapter has been upgraded with the new firmware package.

Updating the Adapter Firmware on Windows

NOTE
During a hot firmware upgrade or error recovery, there may be a period of time that the firmware can not respond to a command sent from the drivers while the firmware is performing a reset. The driver logs this as an error in the event log. During normal operation, the firmware continues the reset and returns to normal operation. As long as the firmware hot upgrade/error recovery is completed successfully, this error can be ignored. This is verified by observing the *Firmware became responsive* and *Firmware reset sequence completed* events in the event log.

To update the adapter firmware on Windows:

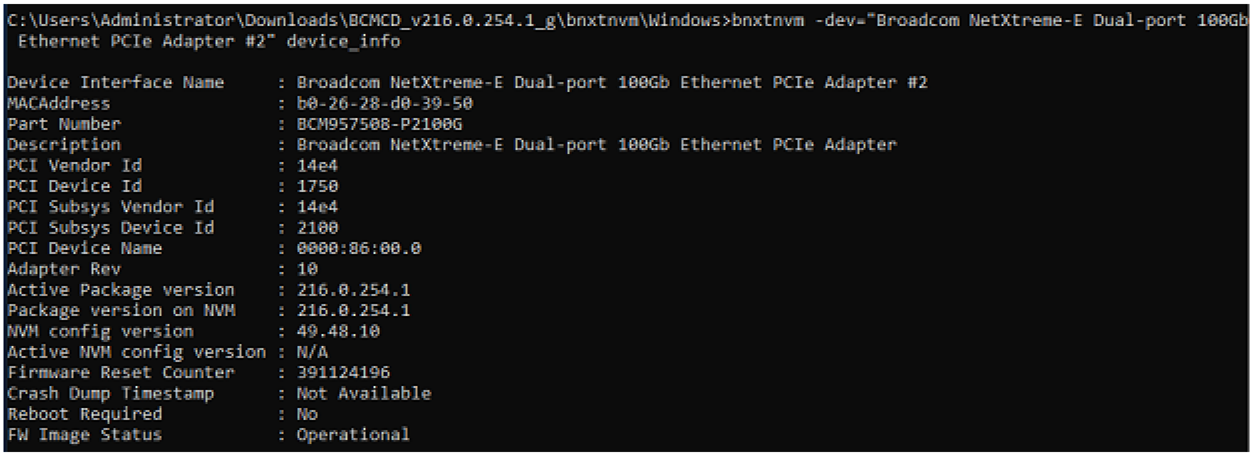
1. List the adapters in the system using the following bnxtnvm listdev command:

   ```
   bnxtnvm listdev
   ```

   This command shows the device interface name that is used to identify the adapter in subsequent commands.
2. After selecting the device interface name, the adapter part number can be viewed using the following bnxtnvm device_info command:

   ```
   bnxtnvm -dev="Broadcom NetXtreme-E Dual-Port 100Gb Ethernet PCIe Adapter #2" verify -v
   ```

   This command shows the part number used to select the firmware package to use in the upgrade.

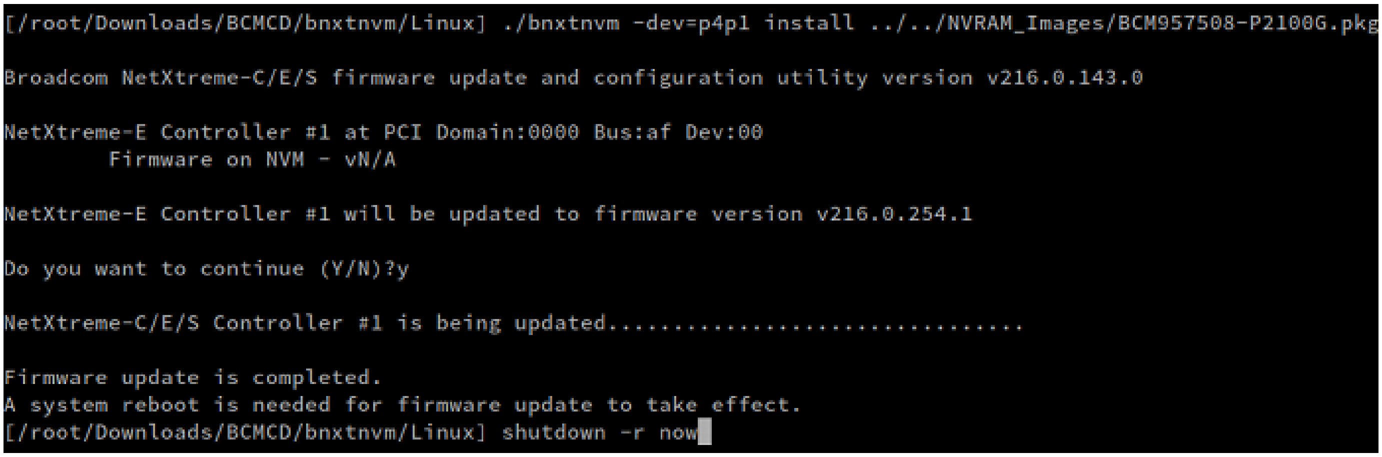


```
C:\Users\Administrator\Downloads\BCMCD_v216.0.254.1_g\bnxtnvm\Windows>bnxtnvm -dev="Broadcom NetXtreme-E Dual-port 100Gb
 Ethernet PCIe Adapter #2" device_info

Device Interface Name     : Broadcom NetXtreme-E Dual-port 100Gb Ethernet PCIe Adapter #2
MACAddress                : b0-26-28-d0-39-50
Part Number               : BCM957508-P2100G
Description               : Broadcom NetXtreme-E Dual-port 100Gb Ethernet PCIe Adapter
PCI Vendor Id             : 14e4
PCI Device Id             : 1750
PCI Subsys Vendor Id      : 14e4
PCI Subsys Device Id      : 2100
PCI Device Name           : 0000:86:00.0
Adapter Rev               : 10
Active Package version    : 216.0.254.1
Package version on NVM    : 216.0.254.1
NVM config version        : 49.48.10
Active NVM config version : N/A
Firmware Reset Counter    : 391124196
Crash Dump Timestamp      : Not Available
Reboot Required           : No
FW Image Status           : Operational
```

3. The firmware is updated using the following bnxtnvm install command:

   ```
   bnxtnvm dev=p4p1 install ../../NVRAM_Images/BCM957508-P2100G.pkg
   ```

The name of the firmware package used to update the adapter firmware is the part number displayed in the device_info command with a .pkg extension.



```
[/root/Downloads/BCMCD/bnxtnvm/Linux] ./bnxtnvm -dev=p4p1 install ../../NVRAM_Images/BCM957508-P2100G.pkg

Broadcom NetXtreme-C/E/S firmware update and configuration utility version v216.0.143.0

NetXtreme-E Controller #1 at PCI Domain:0000 Bus:af Dev:00
        Firmware on NVM - vN/A

NetXtreme-E Controller #1 will be updated to firmware version v216.0.254.1

Do you want to continue (Y/N)?y

NetXtreme-C/E/S Controller #1 is being updated................................

Firmware update is completed.
A system reboot is needed for firmware update to take effect.
[/root/Downloads/BCMCD/bnxtnvm/Linux] shutdown -r now
```

4. Reboot the system after the adapter has been upgraded with the new firmware package.

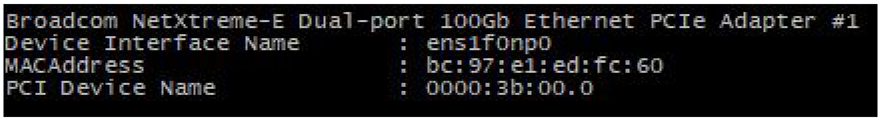
Verifying the Adapter Firmware

To verify the adapter firmware:

After upgrading the firmware and rebooting, the installed firmware version can be verified using any of the following commands:

Example 1

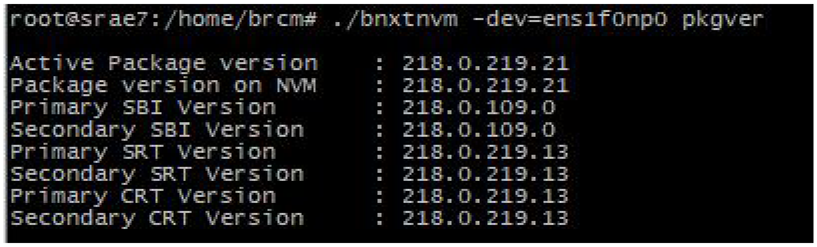
```
[/root/downloads/BCMCD/bnxtnvm/Linux] ./bnxtnvm listdev
```



```
Broadcom NetXtreme-E Dual-port 100Gb Ethernet PCIe Adapter #1
Device Interface Name        : ens1f0np0
MACAddress                   : bc:97:e1:ed:fc:60
PCI Device Name              : 0000:3b:00.0
```

Example 2

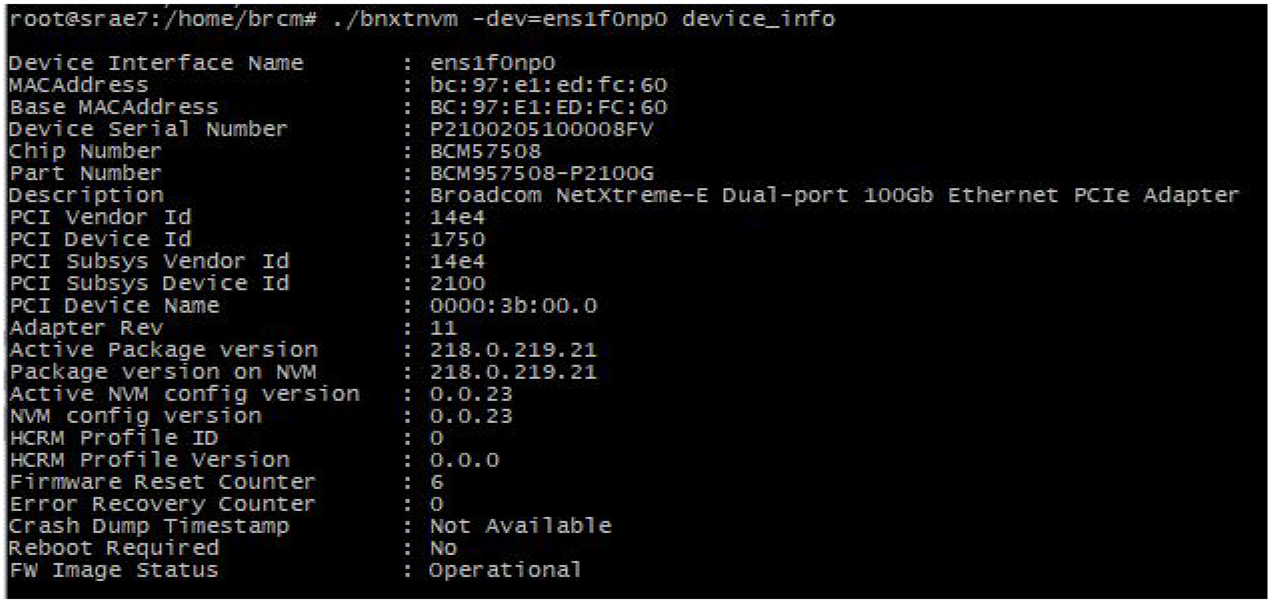
```
[/root/downloads/BCMCD/bnxtnvm/Linux] ./bnxtnvm -dev=ens1f0np0 pkgver
```



```
root@srae7:/home/brcm# ./bnxtnvm -dev=ens1f0np0 pkgver

Active Package version     : 218.0.219.21
Package version on NVM     : 218.0.219.21
Primary SBI Version        : 218.0.109.0
Secondary SBI Version      : 218.0.109.0
Primary SRT Version        : 218.0.219.13
Secondary SRT Version      : 218.0.219.13
Primary CRT Version        : 218.0.219.13
Secondary CRT Version      : 218.0.219.13
```

Example 3

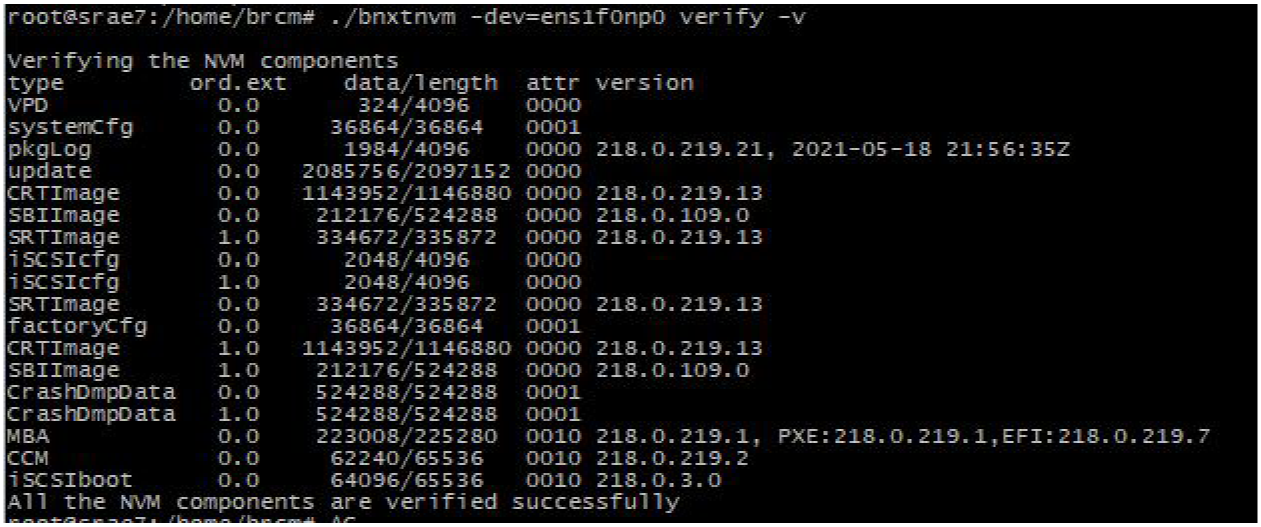
```
[/root/Downloads/BCMCD/bnxtvm/Linux]./bnxtnvm -dev=ens1f0np0 device_info
```



```
root@srae7:/home/brcm# ./bnxtnvm -dev=ens1f0np0 device_info

Device Interface Name        : ens1f0np0
MACAddress                   : bc:97:e1:ed:fc:60
Base MACAddress              : BC:97:E1:ED:FC:60
Device Serial Number         : P2100205100008FV
Chip Number                  : BCM57508
Part Number                  : BCM957508-P2100G
Description                  : Broadcom NetXtreme-E Dual-port 100Gb Ethernet PCIe Adapter
PCI Vendor Id                : 14e4
PCI Device Id                : 1750
PCI Subsys Vendor Id         : 14e4
PCI Subsys Device Id         : 2100
PCI Device Name              : 0000:3b:00.0
Adapter Rev                  : 11
Active Package version       : 218.0.219.21
Package version on NVM       : 218.0.219.21
Active NVM config version    : 0.0.23
NVM config version           : 0.0.23
HCRM Profile ID              : 0
HCRM Profile Version         : 0.0.0
Firmware Reset Counter       : 6
Error Recovery Counter       : 0
Crash Dump Timestamp         : Not Available
Reboot Required              : No
FW Image Status              : Operational
```

Example 4

```
[/root/Downloads/BCMCD/bnxtvm/Linux]./bnxtnvm -dev=ens1f0np0 verify -v
```



```
root@srae7:/home/brcm# ./bnxtnvm -dev=ens1f0np0 verify -v

Verifying the NVM components
type          ord.ext    data/length  attr version
VPD             0.0        324/4096    0000
systemCfg       0.0      36864/36864   0001
pkgLog          0.0       1984/4096    0000 218.0.219.21, 2021-05-18 21:56:35Z
update          0.0    2085756/2097152 0000
CRTImage        0.0    1143952/1146880 0000 218.0.219.13
SBIImage        0.0     212176/524288  0000 218.0.109.0
SRTImage        1.0     334672/335872  0000 218.0.219.13
iSCSIcfg        0.0       2048/4096    0000
iSCSIcfg        1.0       2048/4096    0000
SRTImage        0.0     334672/335872  0000 218.0.219.13
factoryCfg      0.0      36864/36864   0001
CRTImage        1.0    1143952/1146880 0000 218.0.219.13
SBIImage        1.0     212176/524288  0000 218.0.109.0
CrashDmpData    0.0     524288/524288  0001
CrashDmpData    1.0     524288/524288  0001
MBA             0.0     223008/225280  0010 218.0.219.1, PXE:218.0.219.1,EFI:218.0.219.7
CCM             0.0      62240/65536   0010 218.0.219.2
iSCSIboot       0.0      64096/65536   0010 218.0.3.0
All the NVM components are verified successfully
root@srae7:/home/brcm# ^C
```

Configuration

Provides configuring information for various functions on Broadcom Ethernet NIC controllers ensuring optimum performance.

This section provides the following information on configuring the Ethernet NIC controllers.

- Tunneling Configuration Examples
- Link Aggregation
- Auto-Negotiation Configuration
- PXE Boot
- SR-IOV – Configuration and Use Case Examples
- NPAR – Configuration and Use Case Example
- DCBX – Data Center Bridging
- Validating RoCE Network
- Advanced Network Configuration
- Switch Configuration

Tunneling Configuration Examples

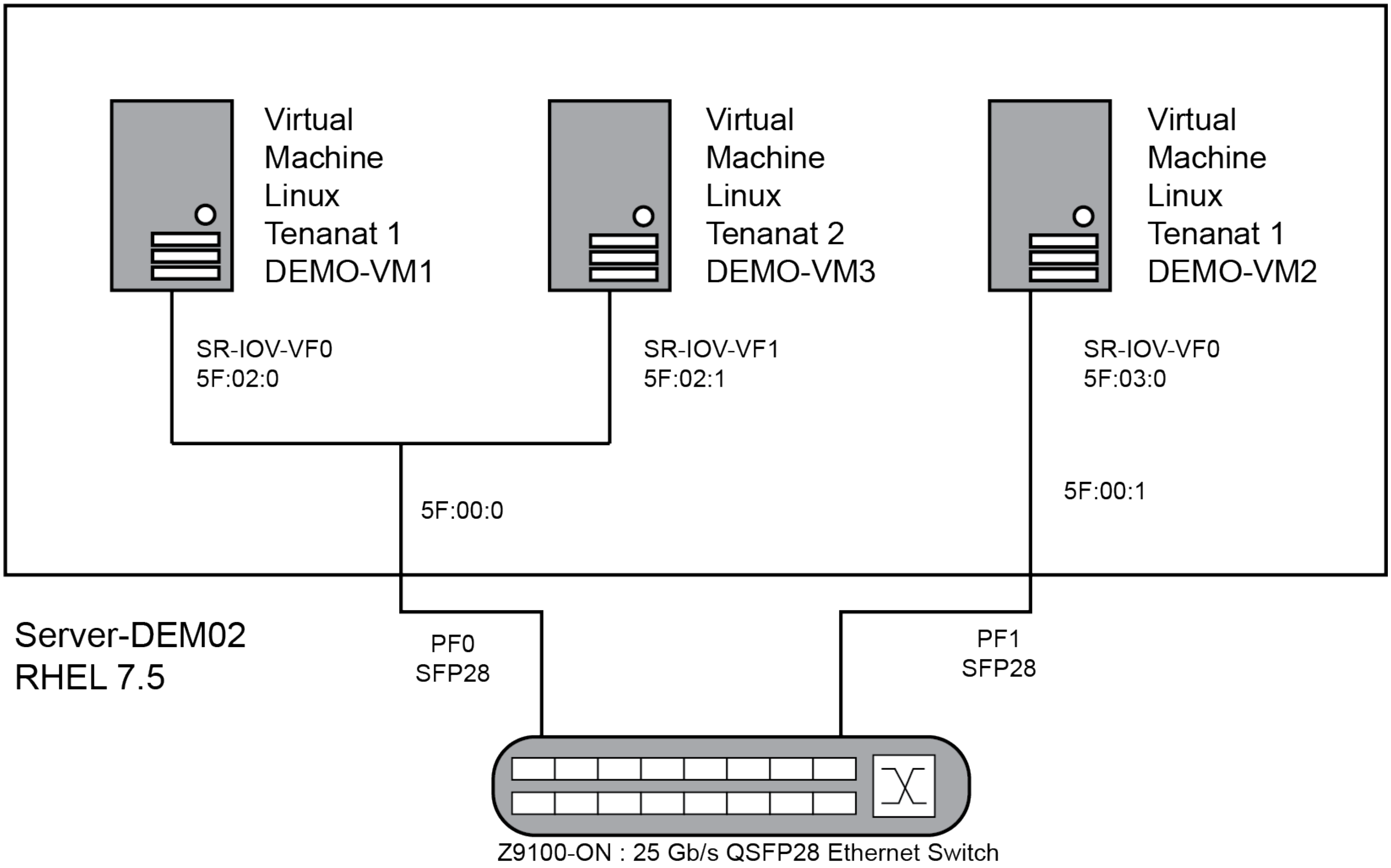
Broadcom BCM5741X and BCM5751X devices support VXLAN, GRE, and IP-in-IP tunneling offloads. This section provides the following tunnelling configuration examples:

- Network Diagram
- VEB and VEPA Modes
- VLAN Double Tagging
- GRE Tunnelling
- IP-in-IP Tunnelling
- VXLAN – Configuration and Use Case Examples

Network Diagram

The test network shown in the following figure uses one Linux server with one two port Ethernet adapter. SR-IOV is enabled in the adapter and two VFs are instantiated on the first port. A VF from the second port is exposed to the third VM (see SR-IOV – Configuration and Use Case Examples for information on SR-IOV bring up).

Figure 34: Network Diagram



VEB and VEPA Modes

VEB (Virtual Ethernet Bridging) mode generates an internal bridge within the NIC for VM-to-VM communication. The Ethernet frames are traverses through the internal bridge. VEPA (Virtual Ethernet Port Aggregator) mode transports the frames to the external switch. The switch handles the frame transport between the ports. VEB and VEPA can be configured through UEFI HII.

VLAN Double Tagging

A VLAN can be configured at the PF level and another VLAN can be configured at the VF level inside the VM. Once the VF is exposed inside the VM, the Linux L2 driver can be installed to activate the interface. The following commands can be used to enable the VLAN inside the VM:

```
modprobe 802.1q;
ip link add link <IntName> name <Vlan.2> type vlan id <vlan num> ip link set <IntName> up;
ip addr add <IPAddr>/mask broadcast <Gateway Addr> dev <IntName>
```

GRE Tunnelling

An IP GRE is an IP inside an IP tunnel which can carry private network traffic between two heterogeneous networks. On the VM, use the following commands:

```
modprobe ip_gre;
ip tunnel add gre45 mode gre local <public IP> remote <Private IP>
ip link set dev gre45 up;
ip addr add <private IP>/mask broadcast <broadcast ip> dev gre45
```

In this example, gre45 is the interface name.

IP-in-IP Tunnelling

Similar to GRE, IP-in-IP is another encapsulation that carries a private IP onto a public IP. Use the following commands:

```
modprobe ipip
ip tunnel add ipip45 mode ipip remote <Peer IP> local <private ip> ttl 255 dev <VM Interface Name>
ip addr add dev ipip45 <IP addr> peer <Peer IP> /mask
ip link set dev ipip45 up
```

NOTE
In this example, ipip45 is the name of the newly created tunnel device.

VXLAN – Configuration and Use Case Examples

VXLAN encapsulation permits multiple virtual machines or containers residing on one server to be isolated from each other in virtual tunnels by encapsulating traffic with VXLAN headers. Broadcom Ethernet NIC controllers accelerate this encapsulation and de-encapsulation in hardware.

This example discusses basic VXLAN connectivity between two Linux servers. Each server has one physical NIC enabled with outer IP address set to 1.1.1.4 and 1.1.1.2, respectively.

A VXLAN interface with ID 10 is created with multicast group 239.0.0.10 and is associated with physical network port eth1 on each server.

An IP address for the host is created on each server and associated to that VXLAN interface. Once the VXLAN interface is brought up, the VM present in system 1 can communicate with the VM present in system 2 using the VXLAN interfaces. The VLXAN format is shown in the following table.

Table 30: VXLAN Frame Format

MAC header	Outer IP header with proto = UDP	UDP header with Destination port= VXLAN	VXLAN header (Flags, VNI)	Original L2 Frame	FCS

The following table provides VXLAN command and configuration examples.

Table 31: VXLAN Command and Configuration Examples

System 1	System 2
ifconfig eth1 1.1.1.4/24	ifconfig eth1 1.1.1.2/24
ip link add vxlan10 type vxlan id 10 group 239.0.0.10 dev eth1 dstport 4789	ip link add vxlan10 type vxlan id 10 group 239.0.0.10 dev eth1 dstport 4789
ifconfig vxlan10 192.168.1.5 mtu 1450	ifconfig vxlan10 192.168.1.10 mtu 1450
ip --d link show vxlan10	–
ping 192.168.1.10	–

Link Aggregation

The following sections provide information on link aggregation:

Windows

Broadcom Ethernet NIC controllers can aggregate network links using the Microsoft teaming feature. For more information on the NIC teaming functionality, see the Microsoft public documentation on Microsoft.com.

Microsoft LBFO is a native teaming driver that can be used in the Windows OS. The teaming driver also provides VLAN tagging capabilities.

Linux

The Linux bonding module is used for link aggregation under Linux. For additional documentation on Linux bonding, see https://www.kernel.org/doc/Documentation/networking/bonding.txt.

Use the following steps as an example to create a bond interface that is ephemeral:

1. Load the bonding module using the following command:
   ```
   sudo modprobe bonding
   ```
2. Create a bond interface named bond0 and mode balance-alb using the following command:
   ```
   sudo ip link add bond0 type bond
   ```
3. Bring down the first interface that will be added to the bond interface using the following command:
   ```
   sudo ip link set enp9s0f0np0 down
   ```
4. Add the first interface to the bond interface using the following command:
   ```
   sudo ip link set enp9s0f0np0 master bond0
   ```
5. Bring down the second interface that will be added to the bond interface using the following command:
   ```
   sudo ip link set enp9s0f1np1 down
   ```
6. Add the second interface to the bond interface using the following command:
   ```
   sudo ip link set enp9s0f1np1 master bond0
   ```
7. Assign an IP address to the bond interface using the following command:
   ```
   sudo ip addr add 192.168.2.35/16 dev bond0
   ```
8. Bring up the bond interface using the following command:
   ```
   sudo ip link set bond0 up
   ```

Use the following steps to setup basic Linux bonding:

1. Load the bonding module using the following command:
   ```
   modprobe bonding mode="balance-alb"
   ```
2. Add physical network interfaces to the bond interface using the following commands:
   ```
   ifenslave bond0 ethX ifenslave bond0 ethY
   ```
3. Assign an IP address to bond the interface. IPV4Address and NetMask are an IPv4 address and the associated network mask.

```
ifconfig bond0 <IP address> netmask <netmask>
```

Auto-Negotiation Configuration

NOTE
In NPAR (NIC partitioning) devices where one port is shared by multiple PCI functions, the port speed is preconfigured and cannot be changed by the driver.

The Broadcom Ethernet NIC controller supports the following auto-negotiation features:

- Link speed auto-negotiation
- FEC auto-negotiation
- Pause/Flow Control auto-negotiation

NOTE
When using SFP+, SFP28 connectors, use DAC or multimode optical transceivers capable of supporting auto- negotiation. Ensure that the link partner port has been set to the matching auto-negotiation protocol. For example, if the local Broadcom port is set to IEEE 802.3by AN protocol, the link partner must support auto-negotiation and must be set to IEEE 802.3by auto-negotiation protocol. When Media Auto Detect and IEEE 802.by + Consortium are enabled, the controller auto detects the Ethernet cables and transceivers to start auto-negotiation with the link partner. FEC is negotiated during auto-negotiation and forced FEC does not take effect in this mode. The default setting enables IEEE 802.by + Consortium and Media Auto Detect.

The supported combination of link speed settings for two port Ethernet NIC controllers are shown in the following table.

Table 32: Suppofted Link Speeds for the BCM95741X and BCM95750X

Dual-Port	BCM95741X	BCM9575XX
1G, 1G	Yes	No
1G, 10G	Yes	No
1G, 25G	Yes	No
10G, 10G	Yes	Yes
10G, 25G	No	Yes
25G, 25G	Yes	Yes
50G, 50G	No	Yes
100G, 100G	No	Yes
Quad-Port		
10G, 10G, 10G, 10G	N/A	Yes
10G, 10G, 10G, 25G	N/A	Yes
10G, 10G, 25G, 25G	N/A	Yes
10G, 25G, 25G, 25G	N/A	Yes
25G, 25G, 25G, 25G	N/A	Yes

NOTE
1 Gb/s link speed for SFP+/SFP28 is currently not supported in this release.

- P1 – Port 1 setting.
- P2 – Port 2 setting.
- AN – Auto-negotiation.
- No AN – Forced speed.
- {link speed} – Expected link speed.
- The BCM57414 does not support independent link speed. All ports must operate at the same speed. For example, the ports cannot be set for port1 = 10G and port2 = 25G. Only the BCM5750X supports independ port speeds.
- **BCM9575XX Supported Combination of Link Speed Settings (25/50/100G-NRZ)**
- All port link speeds are independent of each other.
- Each port can be configured as forced or auto-negotiate for any speed supported by the device.
- AN {link speeds} – Advertised supported auto-negotiation link speeds.
- **BCM957508 Supported Combinations of Link Speed Setting (40/100G-NRZ and 50/100G-PAM-4)**
- All port link speeds are independent of each other.
- Each port can be configured as forced or auto-negotiate (NRZ Only) for any speed supported by the device.
- AN {link speeds} – Advertised supported auto-negotiation link speeds.
- PAM-4 support is enabled at a feature preview level with the following notes/restrictions:
- PAM-4 operation on both ports is supported. NRZ operation on both ports is supported.
- PAM-4 operation on one port with NRZ operation on the other is not supported. PAM-4 supports fixed speed mode only and must be configured through UEFI/HII.
- PAM-4 support in this release is limited to DAC/Twinax cables.
- PAM-4 operation on DAC/Twinax cables requires link training to be enabled. This requires a switch that supports link training an PAM-4.
- PAM-4 requires FEC mode as RS544_1xN for 50G and 100G operation.
- PAM-4 has been tested with the following DAC: DAC Cable: Vendor: Amphenol QSFP56, Part num: NDAAXJ-0003 AOC Cable: Vendor: Optomind Inc, Part num: C4R448GA005AZZ
- PAM-4 on Linux requires that 50/100G PAM-4 is enabled via UEFI menus only. Currently the Linux kernel and associated tools have infrastructure to report link mode. From the reported link mode one can derive the encoding or lanes that may be used for that link. In a case where this is auto-negotiated, this works well. A device reporting a link mode of 100000baseCR2/Full uses a different number of lanes (and therefore encoding) than a device reporting 100000baseCR4/Full. Unfortunately when auto-negotiation is not used there is not ample infrastructure to specify speed and encoding and differentiate between a card using NRZ vs PAM-4 encoding. Currently ethtool supports setting speed, but does not have support for setting encoding or lanes used: ethtool -s|--change DEVNAME Change generic options
- [speed %d]
- [duplex half|full]
- [port tp|aui|bnc|mii|fibre]
- [mdix auto|on|off]
- [autoneg on|off]
- [advertise %x]
- [phyad %d]
- [xcvr internal|external]
- [wol p|u|m|b|a|g|s|d...]
- [sopass %x:%x:%x:%x:%x:%x]
- [msglvl %d | msglvl type on|off ...]

Recent discussions have been occuring on the upstream Linux networking mailing list to address this problem. PAM-4 operation on VMware requires that 50/100G PAM-4 is configured via HII menu and also configured via ESXCLI commands. A reboot is required to ensure proper persistence of these parameters. The command is: esxcli network nic set -n <interface> -S <speed> -D full esxcli network nic set -n vmnic4 -S 200000 -D full PAM-4 operation with 50/100G on Windows Server requires registry key modification. Windows does not contain a PAM4 signaling configuration parameter in adapter device advanced properties. For speeds supported by both NRZ and PAM4, Windows will default to NRZ.

The following work around is required for PAM-4:

1. Examine the NXE device in Windows Device Manager, Network Adapters, Broadcom xxx.
2. Right mouse click and select properties, select Details tab, select Driver Key from the Property drop down, and record the last 4 digits after the backslash.
3. Enter the registry editor (regedit.exe) and navigate to:

- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\xxxx" where xxx is the number recorded previously.
  Add a DWORD type value named "PreferredSignalingMode", and set the value to 1.
  Save and exit the registry editor.
  Disable and Enable device or Reboot server.

The expected link speeds based on the local and link partner settings are shown in the following tables:

Table 33: Expected Link Speeds (Forced)

Local Speed Settings	**Link Partner Speed Settings**		
	Forced 1G	**Forced 10G**	**Forced 25G**
Forced 1G	**1G**	No link	No link
Forced 10G	No link	**10G**	No link
Forced 25G	No link	No link	**25G**
Forced 50G	No link	No link	No link
Forced 100G	No link	No link	No link
Forced 200G	No link	No link	No link

Table 34: Expected Link Speeds (Auto-Negotiate) (Forced)

Link Speed Settings	**Link Partner Speed Settings**								
	AN Enabled 1 G	AN Enabled 10 G	AN Enabled 25G	AN Enabled1/10G	AN Enabled 1/25G	AN Enabled 10/25G	AN Enabled 1/10/25G	AN Enabled 10/25/50G	AN Enabled 10/25/50/100G
AN 1G	1G	No link	No link	1G	1G	No link	1G	No link	No link
AN 10G	No link	10G	No link	10G	No link	10G	10G	10G	10G
AN 25G	No link	No link	25G	No link	25G	25G	25G	25G	25G

Link Speed Settings	Link Partner Speed Settings								
AN 1/10G	1G	10G	No link	10G	1G	10G	10G	10G	10G
AN 1/25G	1G	No link	25G	1G	25G	25G	25G	25G	25G
AN 10/25G	No link	10G	25G	10G	25G	25G	25G	25G	25G
AN 1/10/25G	1G	10G	25G	10G	25G	25G	25G	25G	25G
AN 10/25/50G	1G	10G	25G	10G	25G	25G	25G	50G	50G
AN 10/25/50/ 100G	No link	10G	25G	10G	25G	25G	25G	50G	100G

To enable link speed auto-negotiation, the following options can be enabled in system BIOS HII menu or in CCM:

PXE Boot

To serve PXE requests from PXE clients, a PXE server must be configured. The PXE server can be configured to run regular PXE or iPXE. Regular PXE is a network boot program that downloads config files over TFTP from the PXE server. iPXE is an enhanced implementation of the PXE client firmware and a network boot program which uses iPXE scripts rather than config files and can download scripts and images with HTTP.

> **NOTE**
> UEFI mode PXE boot is supported on physical functions as well as NIC partitions, whereas legacy mode PXE boot is supported only on physical functions.

The procedure for configuring the PXE server is discussed in PXE Server Configuration.

The following sections provide information on PXE boot:

- UEFI Mode
- Legacy BIOS Mode
- PXE Boot with VLAN
- PXE Server Configuration

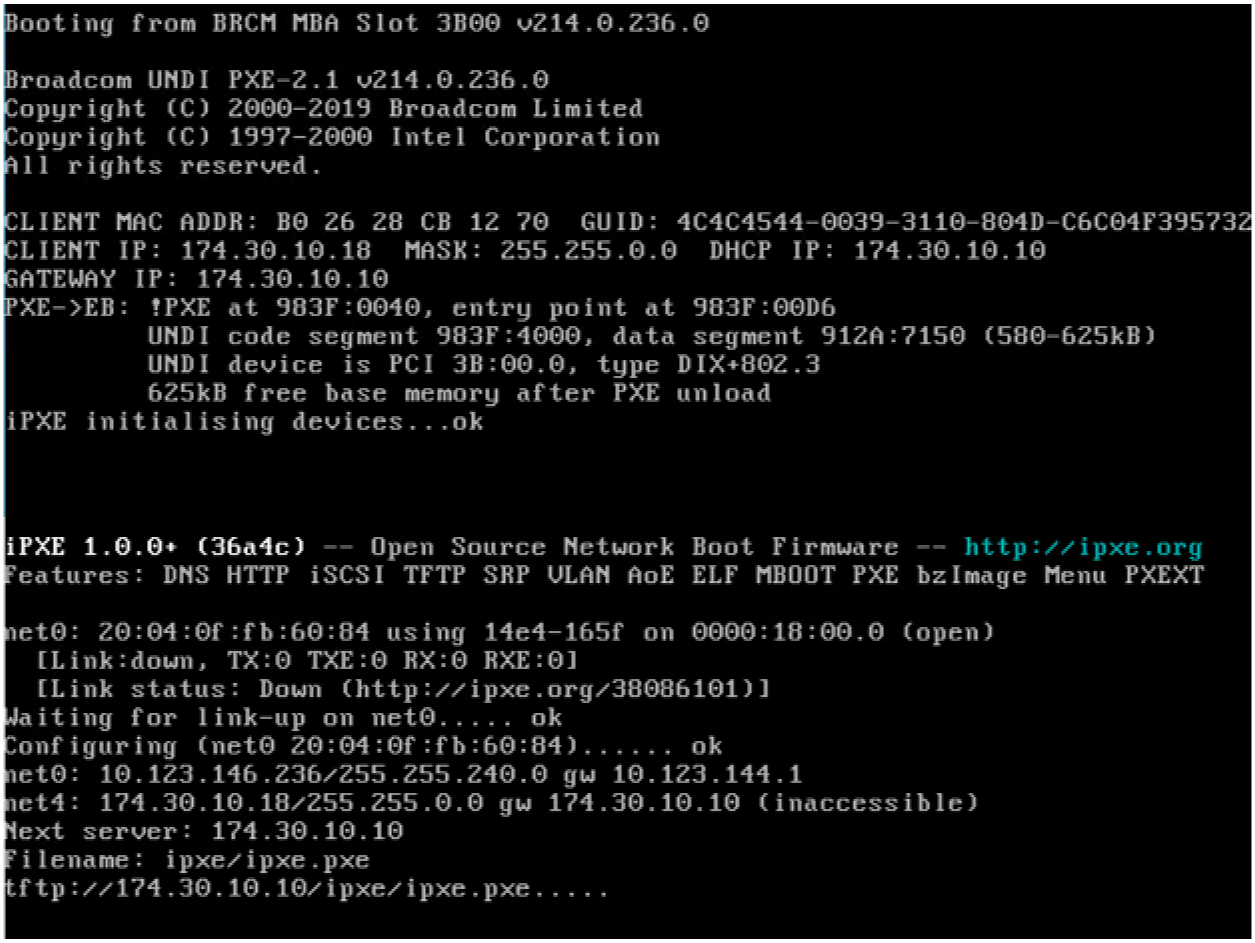
UEFI Mode

Enable PXE for the Broadcom adapter interface in **Network Configuration** under **System Setup**. Refer to the documentation from the server manufacturer on how to enable PXE on the particular server model.

The following sections provide information on UEFI Mode:

iPXE

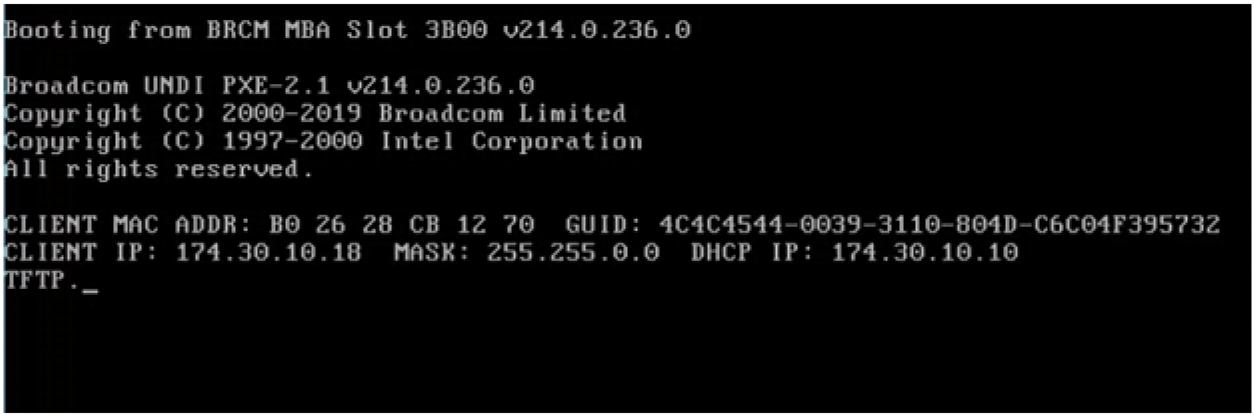
Download ipxe.pxe the boot loader file using TFTP from the PXE server.



```
Booting from BRCM MBA Slot 3B00 v214.0.236.0

Broadcom UNDI PXE-2.1 v214.0.236.0
Copyright (C) 2000-2019 Broadcom Limited
Copyright (C) 1997-2000 Intel Corporation
All rights reserved.

CLIENT MAC ADDR: B0 26 28 CB 12 70  GUID: 4C4C4544-0039-3110-804D-C6C04F395732
CLIENT IP: 174.30.10.18  MASK: 255.255.0.0  DHCP IP: 174.30.10.10
GATEWAY IP: 174.30.10.10
PXE->EB: !PXE at 983F:0040, entry point at 983F:00D6
         UNDI code segment 983F:4000, data segment 912A:7150 (580-625kB)
         UNDI device is PCI 3B:00.0, type DIX+802.3
         625kB free base memory after PXE unload
iPXE initialising devices...ok



iPXE 1.0.0+ (36a4c) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP iSCSI TFTP SRP VLAN AoE ELF MBOOT PXE bzImage Menu PXEXT

net0: 20:04:0f:fb:60:84 using 14e4-165f on 0000:18:00.0 (open)
  [Link:down, TX:0 TXE:0 RX:0 RXE:0]
  [Link status: Down (http://ipxe.org/38086101)]
Waiting for link-up on net0..... ok
Configuring (net0 20:04:0f:fb:60:84)...... ok
net0: 10.123.146.236/255.255.240.0 gw 10.123.144.1
net4: 174.30.10.18/255.255.0.0 gw 174.30.10.10 (inaccessible)
Next server: 174.30.10.10
Filename: ipxe/ipxe.pxe
tftp://174.30.10.10/ipxe/ipxe.pxe.....
```

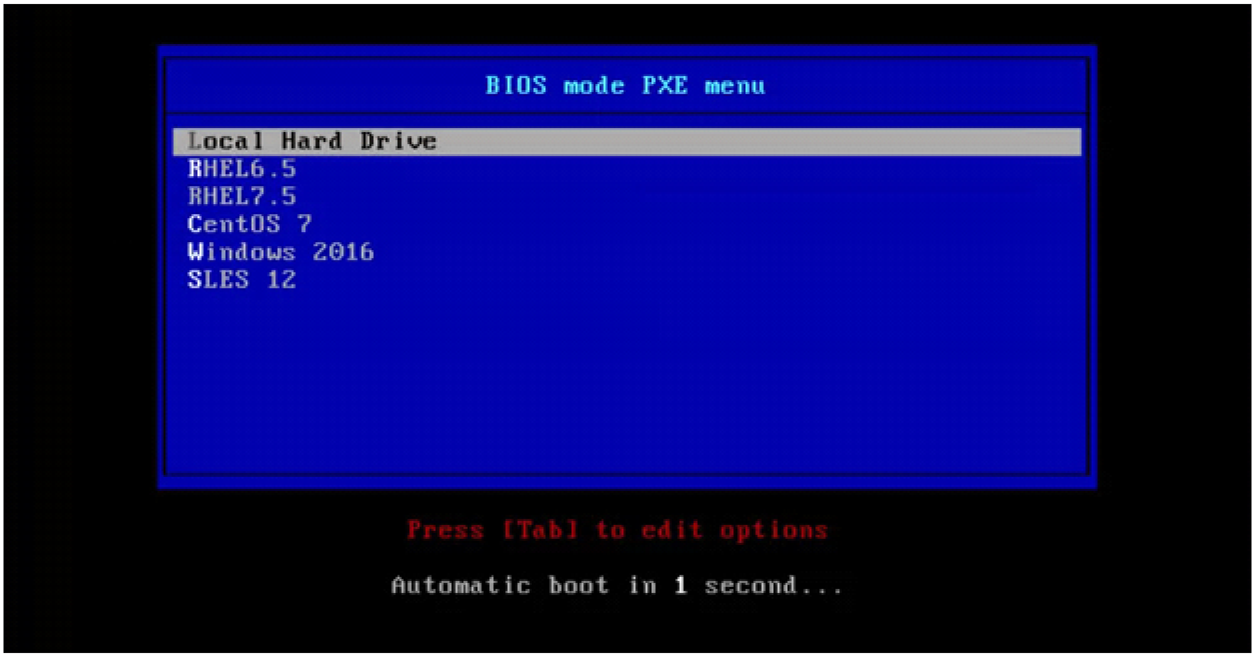
PXE

The boot loader file is downloaded using TFTP from the PXE server.



```
Booting from BRCM MBA Slot 3B00 v214.0.236.0

Broadcom UNDI PXE-2.1 v214.0.236.0
Copyright (C) 2000-2019 Broadcom Limited
Copyright (C) 1997-2000 Intel Corporation
All rights reserved.

CLIENT MAC ADDR: B0 26 28 CB 12 70  GUID: 4C4C4544-0039-3110-804D-C6C04F395732
CLIENT IP: 174.30.10.18  MASK: 255.255.0.0  DHCP IP: 174.30.10.10
TFTP._
```

The menu items from the pxelinux.cfg/default file are listed.



BIOS mode PXE menu

Local Hard Drive
RHEL6.5
RHEL7.5
CentOS 7
Windows 2016
SLES 12

Press [Tab] to edit options

Automatic boot in 1 second...

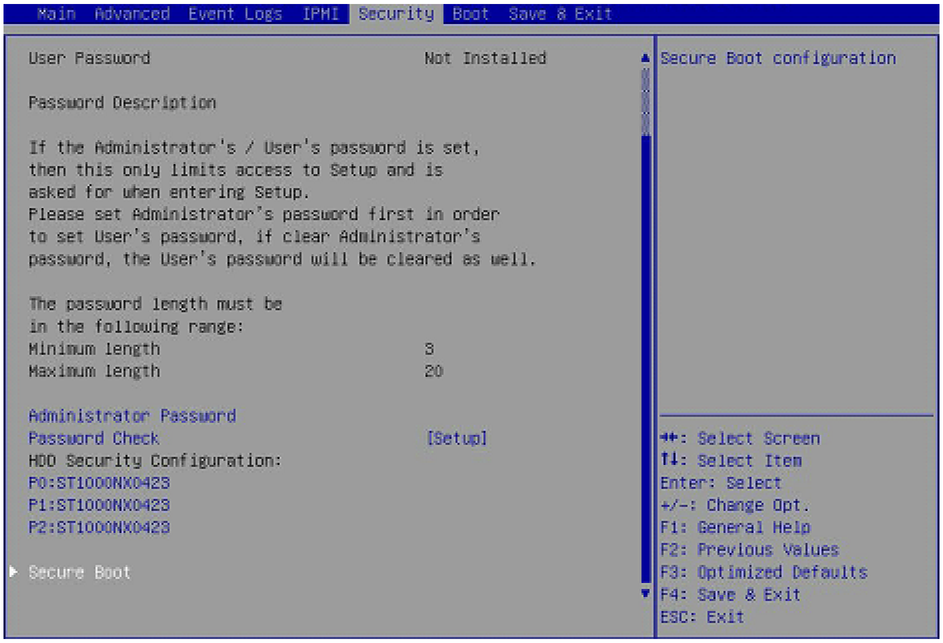
Additional files are downloaded from the PXE server over TFTP based on the menu selection and the PXE boot continues.

Secure Boot

Secure boot is a UEFI based feature developed by UEFI forum to increase security in the pre-boot environment. Secure boot minimizes the threat of potential attacks between firmware initiation and operating system loading. This feature halts the execution of unsigned code before the operating system boots. Any unsigned firmware running on the adapter is not allowed to run. Only adapters running digitally signed firmware images are listed in preboot configuration menus and in network boot options.

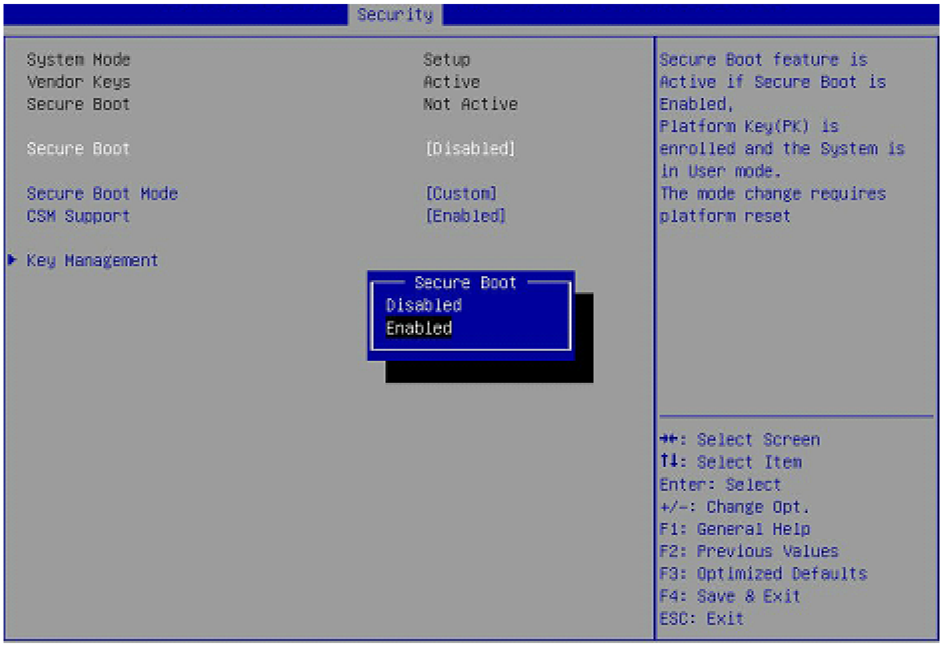
In order to enable this security feature, select **Enable Secure Boot** in the **System Setup Menu**. This option is only available on server platforms that support secure boot. The menu under which the secure boot option is listed varies between server vendors.

Figure 35: Security Menu Example



```
  Main  Advanced  Event Logs  IPMI  Security  Boot  Save & Exit

  User Password                           Not Installed       Secure Boot configuration

  Password Description

  If the Administrator's / User's password is set,
  then this only limits access to Setup and is
  asked for when entering Setup.
  Please set Administrator's password first in order
  to set User's password, if clear Administrator's
  password, the User's password will be cleared as well.

  The password length must be
  in the following range:
  Minimum length                          3
  Maximum length                          20

  Administrator Password
  Password Check                          [Setup]             →←: Select Screen
  HDD Security Configuration:                                 ↑↓: Select Item
  P0:ST1000NX0423                                             Enter: Select
  P1:ST1000NX0423                                             +/-: Change Opt.
  P2:ST1000NX0423                                             F1: General Help
                                                              F2: Previous Values
▶ Secure Boot                                                 F3: Optimized Defaults
                                                              F4: Save & Exit
                                                              ESC: Exit
```

Figure 36: Enable Secure Boot Example



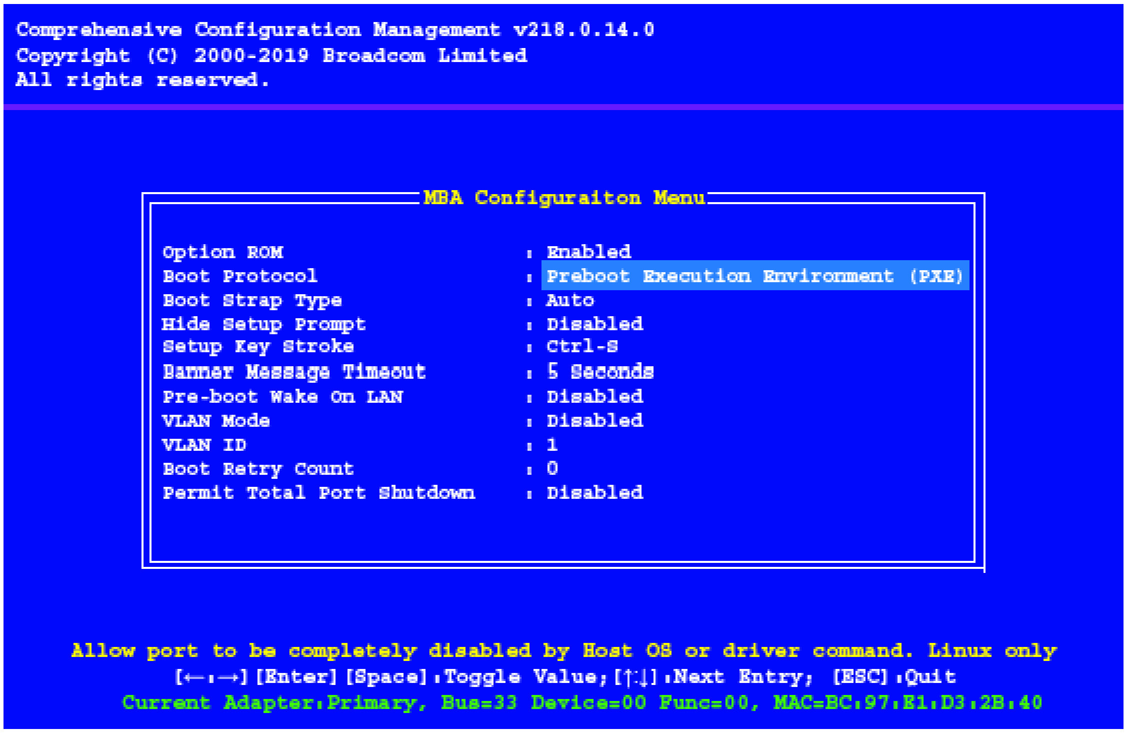
```
                                  Security

  System Mode                             Setup               Secure Boot feature is
  Vendor Keys                             Active              Active if Secure Boot is
  Secure Boot                             Not Active          Enabled,
                                                              Platform Key(PK) is
  Secure Boot                             [Disabled]          enrolled and the System is
                                                              in User mode.
  Secure Boot Mode                        [Custom]            The mode change requires
  CSM Support                             [Enabled]           platform reset

▶ Key Management
                                    ── Secure Boot ──
                                    Disabled
                                    Enabled

                                                              →←: Select Screen
                                                              ↑↓: Select Item
                                                              Enter: Select
                                                              +/-: Change Opt.
                                                              F1: General Help
                                                              F2: Previous Values
                                                              F3: Optimized Defaults
                                                              F4: Save & Exit
                                                              ESC: Exit
```

Legacy BIOS Mode

To configure PXE for legacy BIOS mode:

> **NOTE**
> For Legacy BIOS boot mode, only TFTP-based PXE over IPv4 is supported.

1. Set the **Boot Protocol** to **PXE** in the **MBA Configuration** menu.



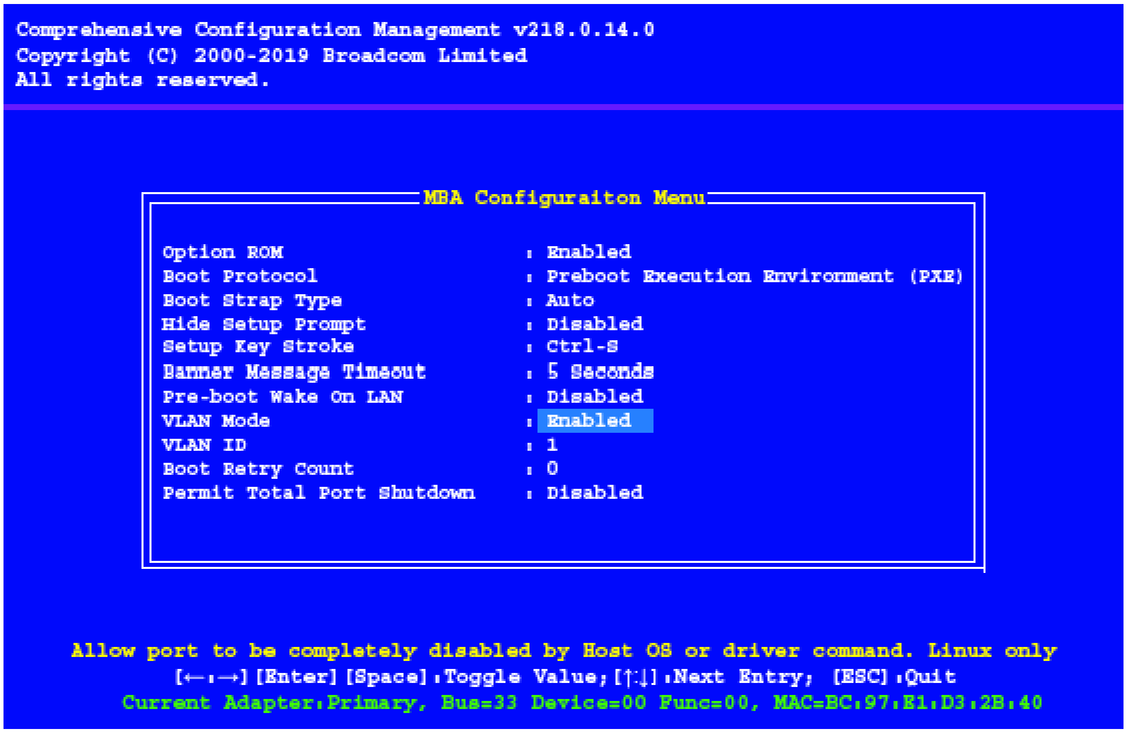
```
Comprehensive Configuration Management v218.0.14.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                 MBA Configuraiton Menu

 Option ROM                    : Enabled
 Boot Protocol                 : Preboot Execution Environment (PXE)
 Boot Strap Type               : Auto
 Hide Setup Prompt             : Disabled
 Setup Key Stroke              : Ctrl-S
 Banner Message Timeout        : 5 Seconds
 Pre-boot Wake On LAN          : Disabled
 VLAN Mode                     : Disabled
 VLAN ID                       : 1
 Boot Retry Count              : 0
 Permit Total Port Shutdown    : Disabled

   Allow port to be completely disabled by Host OS or driver command. Linux only
       [←,→] [Enter] [Space]:Toggle Value; [↑,↓]:Next Entry; [ESC]:Quit
     Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

Save and reboot the server.

PXE Boot with VLAN

To configure PXE boot with VLAN:

1. Set **Virtual LAN Mode** to **Enabled** and assign a valid Virtual LAN ID in the **MBA Configuration** menu.



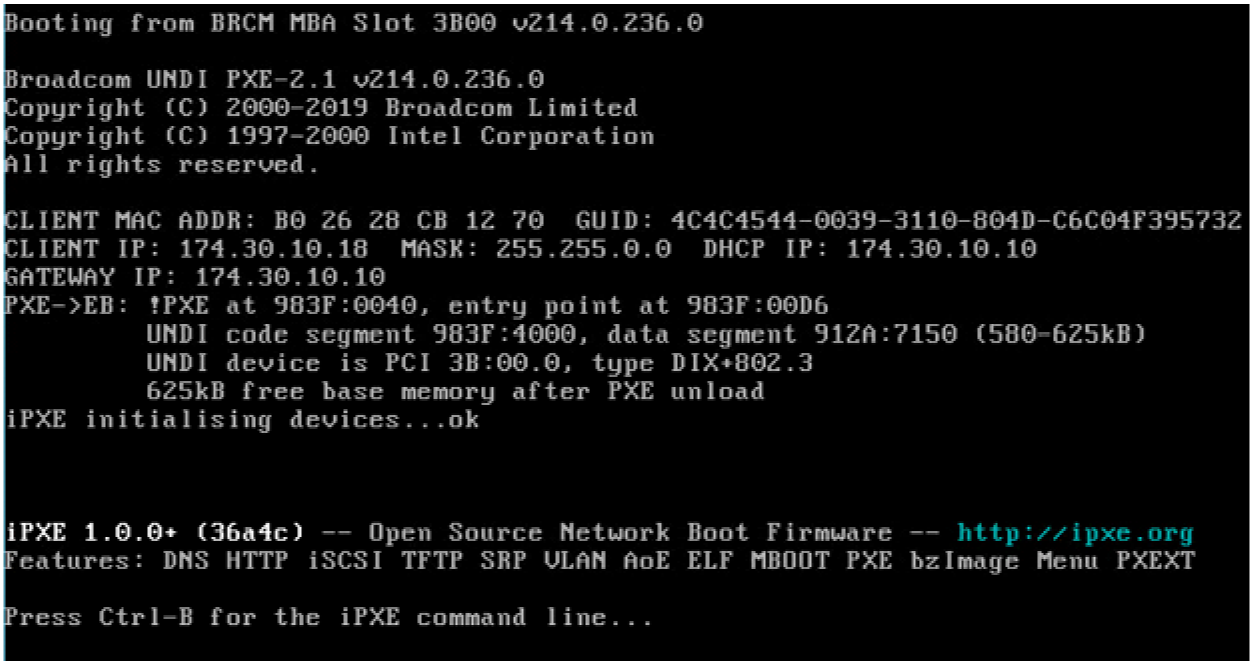
```
Comprehensive Configuration Management v218.0.14.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                    MBA Configuraiton Menu

        Option ROM                    : Enabled
        Boot Protocol                 : Preboot Execution Environment (PXE)
        Boot Strap Type               : Auto
        Hide Setup Prompt             : Disabled
        Setup Key Stroke              : Ctrl-S
        Banner Message Timeout        : 5 Seconds
        Pre-boot Wake On LAN          : Disabled
        VLAN Mode                     : Enabled
        VLAN ID                       : 1
        Boot Retry Count              : 0
        Permit Total Port Shutdown    : Disabled

    Allow port to be completely disabled by Host OS or driver command. Linux only
           [←:→] [Enter] [Space]:Toggle Value;[↑:↓]:Next Entry;  [ESC]:Quit
       Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

2. Configure the PXE server interface with a matching VLAN ID.

   NOTE
   The **Virtual LAN Mode** and **Virtual LAN ID** parameters on the **NIC Configuration** >menu is applicable to PXE boot in UEFI mode as well. If **Virtual LAN** configuration is done from both the **System Setup** menus as well as the adapter **NIC Configuration** menu, the **Virtual LAN ID** configured through **System Setup** takes precedence.

3. Trigger PXE on the adapter interface.

   Once the boot loader is downloaded, press **Ctrl+B** to enter the **iPXE** menu



```
Booting from BRCM MBA Slot 3B00 v214.0.236.0

Broadcom UNDI PXE-2.1 v214.0.236.0
Copyright (C) 2000-2019 Broadcom Limited
Copyright (C) 1997-2000 Intel Corporation
All rights reserved.

CLIENT MAC ADDR: B0 26 28 CB 12 70  GUID: 4C4C4544-0039-3110-804D-C6C04F395732
CLIENT IP: 174.30.10.18  MASK: 255.255.0.0  DHCP IP: 174.30.10.10
GATEWAY IP: 174.30.10.10
PXE->EB: !PXE at 983F:0040, entry point at 983F:00D6
         UNDI code segment 983F:4000, data segment 912A:7150 (580-625kB)
         UNDI device is PCI 3B:00.0, type DIX+802.3
         625kB free base memory after PXE unload
iPXE initialising devices...ok



iPXE 1.0.0+ (36a4c) -- Open Source Network Boot Firmware -- http://ipxe.org
Features: DNS HTTP iSCSI TFTP SRP VLAN AoE ELF MBOOT PXE bzImage Menu PXEXT

Press Ctrl-B for the iPXE command line...
```

Once in the iPXE shell, run the following commands to add a VLAN tag on the adapter interface.

- Ifstat [to identify the adapter interface by its MAC address]
- vcreate --tag <VLANID> <interface>
- autoboot <interface>

This starts the PXE boot on the test interface with VLAN.

In order to destroy the VLAN tag on the adapter interface, the command vdestroy <VLAN_interface> can be run in the iPXE shell.

NOTE
For iPXE to support the the vcreate command, iPXE must be built with vlan_cmd enabled in buildcfg. For more details, refer https://ipxe.org/buildcfg/vlan_cmd.

PXE Server Configuration

The PXE server is operating system independent. A PXE server is any host that can lease out DHCP IPs to any requesting PXE Client, point to a boot loader file, and transfer further configuration and boot files to the PXE client over any application level protocol.

This section explains how to setup a PXE server on a RHEL/CentOS-based OS. For Windows deployment services, refer to documentation provided on the Microsoft Technet website. For setting up PXE servers on any other operating system, refer to the documentation provided by the respective operating system.

The following sections provide information on PXE Server Configuration:

- DHCP Configuration for PXE/iPXE
- TFTP Configuration
- HTTP Configuration

DHCP Configuration for PXE/iPXE

To add the DHCP feature on the PXE server, use the following command:

```
yum install dhcp
```

NOTE
All IP addresses mentioned in this section are for illustration purposes only. They can be modified to match the subnet that the PXE server is configured in.

IPv4 DHCP Configuration

To configure an interface with static network settings using ifcfg files, for an interface with the name p1p1, create a file with name ifcfg-p1p1 in the /etc/sysconfig/network-scripts/ directory as follows:

```
DEVICE=p1p1
TYPE=Ethernet BOOTPROTO=none ONBOOT=yes IPADDR=174.30.10.10 NETMASK=255.255.0.0
```

1. iPXE

   To run iPXE, in the /etc/dhcp/dhcpd.conf file, make the following changes:

   NOTE
   The IP addresses used in this section are for examples only.

```
option ipxe.no-pxedhcp 1;
subnet 174.30.10.0 netmask 255.255.0.0 {
option routers 174.30.10.10;
range 174.30.10.11 174.30.10.50;
next-server 174.30.10.10;
option subnet-mask 255.255.0.0;
default-lease-time 3600;
max-lease-time 4800; class "pxeclients" {
match if substring (option vendor-class-identifier, 0, 9) = "PXEClient"; if not exists ipxe.bus-id {
next-server 174.30.10.10; if option arch = 00:06 {
filename "ipxe/ipxe-x86.efi";
} elsif option arch = 00:07 {
filename "ipxe/ipxe.efi"; # iPXE.efi built with support forBroadcom adapters
} elsif option arch = 00:00 {
filename "ipxe/ipxe.pxe"; # iPXE.pxe built with support for Broadcom adapters
}
} else {
next-server 174.30.10.10;
filename "ipxe/menu.ipxe"; # iPXE boot menu
}
}
}
```

2. PXE

To run PXE, in the /etc/dhcp/dhcpd.conf file, make the following changes:

```
ddns-update-style none; default-lease-time 600;
option space PXE;
option PXE.mtftp-ipcode 1 = ip-address;
option PXE.mtftp-cport code 2 = unsigned integer 16; option PXE.mtftp-sport code 3 = unsigned integer 16;
 option PXE.mtftp-tmout code 4 = unsigned integer 8; option PXE.mtftp-delay code 5 = unsigned integer 8;
 option arch code 93 = unsigned integer 16;
allow booting; allow bootp;
allow unknown-clients;
subnet 174.30.0.0 netmask 255.255.0.0 {
default-lease-time 600;
max-lease-time 6000;
range 174.30.10.11 174.30.10.50; #
class "pxeclients" {
match if substring (option vendor-class-identifier, 0, 9) = "PXEClient"; next-server 174.30.10.10;
if option arch = 00:06 {
filename "bootia32.efi";
} else if option arch = 00:07 { filename "BOOTX64.EFI";
} else {
filename "pxelinux/pxelinux.0";
}
}
}
```

In order to bind a certain hardware MAC address to an IP in the DHCP range, add the following to the dhcpd.conf file.

```
host server1-adapter1-port1 {
hardware ethernet 00:0A:F7:94:F7:A4; fixed-address 174.30.10.15;
}
```

To restart the network service, run the following command:

```
service network restart
```

To restart the DHCP service, run the following command:

```
service dhcpd restart
```

IPv6 DHCP Configuration

In /etc/sysconfig/network, make the following changes:

```
NETWORKING_IPV6=yes IPV6FORWARDING=no IPV6_AUTOCONF=no IPV6_AUTOTUNNEL=no
```

In /etc/sysconfig/network-scripts/ifcfg-<interface_name> make the following changes:

```
IPV6_AUTOCONF=no IPV6INIT=yes
 IPV6ADDR=2015:9:19:ffff::10/64 # Replace with your static address
```

1. iPXE
   In /etc/dhcpd/dhcpd6.conf file, make the following changes:

```
default-lease-time 2592000;
preferred-lifetime 604800;
option dhcp-renewal-time 3600;
option dhcp-rebinding-time 7200; allow leasequery;
option dhcp6.info-refresh-time 21600;
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";
option dhcp6.user-class code 15 = string; option dhcp6.bootfile-url code 59 = string;
option dhcp6.client-arch-type code 61 = array of unsigned integer 16; option dhcp6.name-servers
 2015:9:19:ffff::10;
subnet6 2015:9:19:ffff::/64 {
range6 2015:9:19:ffff::11 2015:9:19:ffff::500;
if exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:07 { option dhcp6.bootfile-url
 "tftp://[2015:9:19:ffff::10]/ipxe/ipxe.efi";
} elsif exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:00 { option dhcp6.bootfile-
url "tftp://[ 2015:9:19:ffff::10]/pxelinux/pxelinux.0";
} elsif exists dhcp6.user-class and substring(option dhcp6.user-class, 2, 4) = "iPXE" { option
 dhcp6.bootfile-url "tftp://[2015:9:19:ffff::10]/ipxe/menu.ipxe";
}
```

```
     }
```

2. PXE

In /etc/dhcpd/dhcpd6.conf file, make the following changes:

```
default-lease-time 2592000;
preferred-lifetime 604800;
option dhcp-renewal-time 3600;
option dhcp-rebinding-time 7200; allow leasequery;
option dhcp6.info-refresh-time 21600;
dhcpv6-lease-file-name "/var/lib/dhcpd/dhcpd6.leases";
option dhcp6.user-class code 15 = string; option dhcp6.bootfile-url code 59 = string;
option dhcp6.client-arch-type code 61 = array of unsigned integer 16; option dhcp6.name-servers
 2015:9:19:ffff::10;
subnet6 2015:9:19:ffff::/64 {
range6 2015:9:19:ffff::11 2015:9:19:ffff::500;
if exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:07 { option dhcp6.bootfile-url
 "tftp://[2015:9:19:ffff::10]/BOOTX64.EFI ";
} elsif exists dhcp6.client-arch-type and option dhcp6.client-arch-type = 00:00 { option dhcp6.bootfile-url
 "tftp://[ 2015:9:19:ffff::10]/pxelinux/pxelinux.0";
        }
```

To restart the network service, run the following command:

```
service network restart
```

To restart the DHCPv6 service, run the following command:

```
service dhcpd6 restart
```

DHCP with VLAN

In /etc/sysconfig/network-scripts, make a copy of the interface file for the interface over which VLAN must be configured.

Example: ifcfg-p1p1 and ifcfg-p1p1.5

NOTE
5 indicates that the interface will use VLAN ID 5.

In the new VLAN interface file, add the parameter VLAN=yes.

UUID can be generated for this new interface using the command uuidgen p1p1.5.

In the file /etc/dhcp/dhcpd, add a scope for the VLAN interface so that the DHCP server leases DHCP IPs on the VLAN interface.

TFTP Configuration

To configure TFTP:

1. Add the TFTP, XINET packages.

```
yum install xinetd tftp-server
```

2. In the /etc/xinetd.d/tftp file, make the following changes.

```
service tftp
    {
    socket_type= dgram protocol= udp
    wait= yes
   user= root
    server= /usr/sbin/in.tftpd server_args= -s /var/lib/tftpboot -6 disable= no
    per_source= 11
    cps= 100 2
   flags= IPv6
    }
```

The base path for the TFTP server is /var/lib/tftpboot/.

iPXE

Under the TFTP root directory, create a new folder called ipxe and add the ipxe.efi and ipxe.pxe files to it. Also add the menu.ipxe file to it, which can chain more iPXE files to list the menu based on the boot mode.

```
#!ipxe
      iseq ${platform} efi && goto uefibios || goto legacybios
      :uefibios
      echo Loading the UEFI Menu
      chain --replace --autofree ${menu-url}efimenu.ipxe
      :legacybios
      echo Loading the LEGACY Menu
      chain --replace --autofree ${menu-url}biosmenu.ipxe
All the chained files are to be present in the <TFTP_root>/ipxe directory.
```

PXE

Create a new directory pxelinux in the TFTP root. Extract the Syslinux package and move the contents to the

```
<TFTP_root>/pxelinux directory.
```

The TFTP root must contain the boot loader file from the target OS. The boot loader file can be copied from the /boot/ efi/ EFI/<OS flavor> directory on the target OS. It should be renamed to BOOTX64.EFI.

Create grub.cfg file in the TFTP root and add UEFI mode PXE boot menu items to the same.

```
default=0 timeout=10
      title RedHat 7u5
      root (nd)
      kernel /images/rhel75/vmlinuz inst.driver=vesa nomodeset method=http:// 174.30.10.10/images/ RHEL75linux
 dd
      initrd /images/RHEL7.2/initrd.img ip=dhcp
      title SLES12SP4
      root (nd)
      kernel images/sles12sp4/vmlinuz inst.driver=vesa nomodeset inst.repo=http:// 174.30.10.10/ images/
SLES12SP4
      initrd /images/sles12sp4/initrd.img ip=dhcp
```

Create pxelinux.cfg/default file and add BIOS mode PXE boot menu items to it.

```
DEFAULT menu.c32 PROMPT 0
      TIMEOUT 10
      MENU TITLE BIOS mode PXE menu
      LABEL localdisk
      MENU LABEL ^Local Hard Drive LOCALBOOT 0
      LABEL RHEL75
      MENU LABEL ^RedHat 7u5 KERNEL images/rhel75/vmlinuz
```

```
     APPEND initrd=images/rhel75/initrd.img ramdisk_size=200000 ip=dhcp inst.xdriver=vesa nomodeset
inst.repo=http://174.30.10.10/images/RHEL75
     LABEL SLES12SP4
     MENU LABEL ^SLES 12 SP 4
     KERNEL images/sles12sp4/vmlinuz
       APPEND initrd=images/sles12sp4/initrd.img ramdisk_size=200000 ip=dhcp inst.xdriver=vesa nomodeset
inst.repo=http://174.30.10.10/images/SLES12SP4
```

Create a new directory images in the TFTP root. Sub-directories can be created to match the kernel image path specified in grub.cfg and pxelinux.cfg/default files. Copy the vmlinuz and initrd.img files from the /boot directory of the target OS's to these sub-directories.

To restart the TFTP service, run the following command:

```
service xinetd restart
```

HTTP Configuration

To configure HTTP:

1. Add the HTTP feature using the following command:

   ```
   yum install httpd
           /var/www/html/ is the base path for the HTTP server. The conf file is present in /etc/httpd/conf/
   httpd.conf.
   ```

2. To restart the HTTP service, run the following command:

   ```
   service httpd restart
   ```

   To specifically make HTTP listen on certain interfaces, in the /etc/httpd/conf\httpd.conf file, add the LISTEN directive for all the required interfaces.

   ```
   Listen 192.0.2.1:80
   Listen 192.0.2.5:8000
   Listen 174.30.10.10:80
   ```

   If this is not specified, the server listens on all interfaces.

3. Create a new directory images in the HTTP root. Subdirectories can be created to match the installation repository path specified in grub.cfg and pxelinux.cfg/default files. Extract the contents of the installation media of the target OS's to these sub-directories.

SR-IOV – Configuration and Use Case Examples

This section provides the following SR-IOV configuration and use case examples:

- Enable SR-IOV in BIOS/UEFI and Device
- Linux Use Case Example: SR-IOV Pass-Through to libvirt Virtual Machine
- VMware SR-IOV Use Case Example

Enable SR-IOV in BIOS/UEFI and Device

To enable SR-IVO in BIOS/UEFI and Device:

1. Enable SR-IOV in the NIC cards:
   a. SR-IOV in the NIC card can be enabled using the HII menu. During system boot, access the system **BIOS→ NetXtreme-E NIC→Device Level Configuration** menu.
   b. Set the **Virtualization** mode to **SR-IOV**.
   c. Set the number of virtual functions per physical function.

   d. Set the number of MSI-X vectors per the VF and Max number of physical function MSI-X vectors. If the VF is running out of resources, balance the number of MSI-X vectors per VM using CCM.
2. Enable virtualization in the BIOS:
   a. During system boot, enter the system **BIOS** →**Processor settings**→**Virtualization Technologies** and set it to Enabled.
   b. During system boot, enter the system **BIOS** →**SR-IOV Global** and set it to **Enabled**.

Linux Use Case Example: SR-IOV Pass-Through to libvirt Virtual Machine

1. Install the desired Linux version with Virtualization enabled (libvirt and Qemu).
2. Enable the IOMMU kernel parameter.
   a. The IOMMU kernel parameter is set by appending intel_iommu=on to the kernel command line

   vi /etc/default/grub (add "intel_iommu=on" to GRUB_CMDLINE_LINUX

   grub2-mkconfig -o /boot/grub2/grub.cfg (or /boot/efi/<system type>/grub.cfg.
3. Use in-box driver, or install the driver as shown in Installing the Linux Driver.
4. Enable Virtual Functions through kernel parameters:
   a. Once the driver is installed, lspci displays the Broadcom Ethernet NIC controller physical interfaces present in the system.
   b. To activate Virtual functions, use the following command:

   echo X > /sys/class/net/<ifname>/device/sriov_numvfs

   NOTE
   Ensure that the physical interface (<interface>) is up. VFs are only created if PFs are up. X is the number of VFs that are exported to the OS.

   Example: echo 4 > /sys/class/net/eth1/device/sriov_numvfs
5. Check the PCIe virtual functions exist with lspci.
6. Create new virtual machine from installation ISO file.
7. Select Customize configuration before install.
8. Select **Add Hardware** → **PCI Host Device** → **Virtual Function**.
9. Finish installation. The VM detects the Broadcom Etherent NIC controller. Use either the in-box driver from the installed operating system or the install driver as shown in Installing the Linux Driver.

VMware SR-IOV Use Case Example

1. On ESXi, install the driver as shown in Installing the VMware Driver.
2. Enable SR-IOV VFs:

Only the physical functions (PFs) are automatically enabled. If a PF supports SR-IOV, the PF(vmknicX) is part of the output of the command shown below.

```
esxcli network sriovnic list
```

To enable one or more virtual functions (VFs), the driver uses the module parameter max_vfs to enable the desired number of VFs per PF. For example, to enable four VFs on PF1:

```
esxcfg-module -s 'max_vfs=4' bnxtnet (reboot required)
```

To enable VFs on a set of PFs, use the command format shown below. For example, to enable four VFs on PF 0 and 2 VFs on PF 2:

```
esxcfg-module -s 'max_vfs=4,2' bnxtnet (reboot required)
```

The required VFs of each supported PF are enabled in order during the PF bring up. See the VMware documentation for information on how to map a VF to a VM.

NOTE
When using NPAR + SR-IOV, every NPAR function (PF) is assigned a maximum of eight VFs.

NOTE
For a VF to be in promiscuous mode, one of the following conditions must be true:

- The VF should be associated with a default VLAN.
- The VF should be a trusted VF.

If none of the above conditions are true, the VF will not be in promiscuous mode and, therefore, will not see packets received by the PF.

NPAR – Configuration and Use Case Example

This section provides the following information on NPAR configuration and use case examples:

- Features and Requirements
- Limitations
- Configuration
- Reducing NIC Memory Consumption with NPAR

Features and Requirements

- OS/BIOS Agnostic – The partitions are presented to the operating system as real network interfaces so no special BIOS or OS support is required like SR-IOV.
- Additional NIC functions without requiring additional switch ports, cabling, PCIe expansion slots.
- Traffic Shaping – The allocation of bandwidth per partition can be controlled so as to limit or reserve as needed.
- Can be used in a Switch Independent manner – The switch does not need any special configuration or knowledge of the NPAR enablement.
- Can be used in conjunction with RoCE and SR-IOV.
- Supports stateless offloads such as LSO, TPA, RSS/TSS, and RoCE (two PFs per port only).
- Alternative Routing-ID support for greater than eight functions per physical device.

NOTE
In the **UEFI HII Menu** page, the Broadcom Etherenet NIC controllers support up to 16 PFs per device on an ARI capable system. For a 2-port device, this means up to 8 PFs for each port.

Limitations

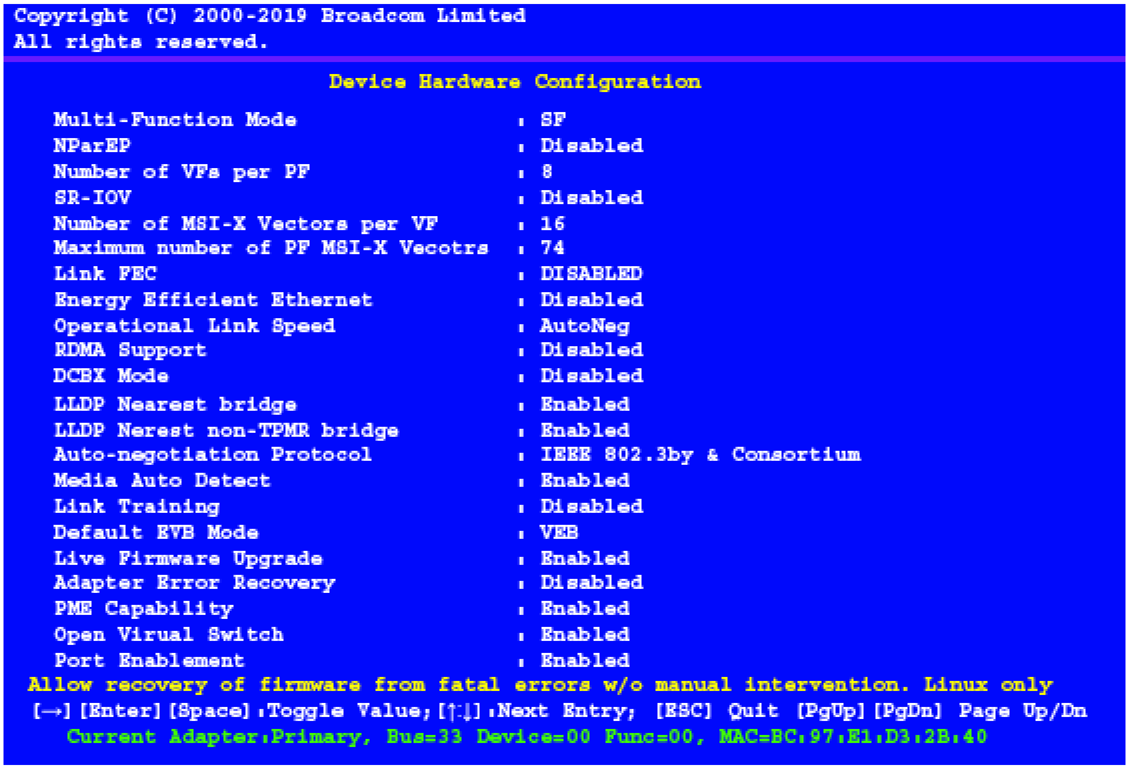
- Shared settings must be suppressed to avoid contention. For example: Speed, Duplex, Flow Control, and similar physical settings are hidden by the device driver to avoid contention.
- Non-ARI systems enable only eight partitions per physical device.
- RoCE + SRIOV + NPAR" combination is not supported.
- RoCE for BCM5741X adapters, is only supported on the first two partitions of each physical port, or a total of four partitions per physical device. BCM9575XX adapters can support RoCE on all partitions.
- NPAR + SR-IOV/MultiRSS is not supported with ESXi 7.0 for the BCM5741X.

Configuration

NPAR can be configured using BIOS configuration HII menus or by using the Broadcom CCM utility on legacy boot systems. Some vendors also expose the configuration via additional proprietary interfaces.

To enable NPAR:

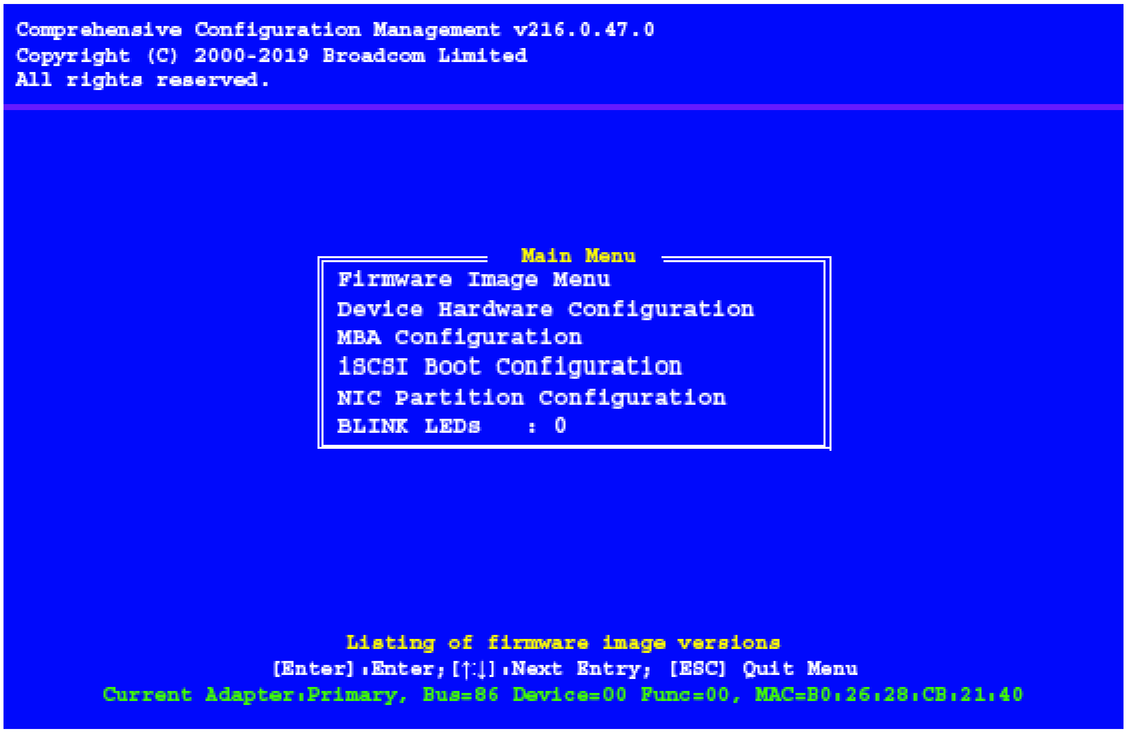
1. Select the target NIC from the BIOS HII Menu or CCM interface and set the Multi-Function Mode or Virtualization Mode option. The choice of options affects the whole NIC instead of the individual port.



```
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                         Device Hardware Configuration

   Multi-Function Mode                   : SF
   NParEP                                : Disabled
   Number of VFs per PF                  : 8
   SR-IOV                                : Disabled
   Number of MSI-X Vectors per VF        : 16
   Maximum number of PF MSI-X Vecotrs    : 74
   Link FEC                              : DISABLED
   Energy Efficient Ethernet             : Disabled
   Operational Link Speed                : AutoNeg
   RDMA Support                          : Disabled
   DCBX Mode                             : Disabled
   LLDP Nearest bridge                   : Enabled
   LLDP Nerest non-TPMR bridge           : Enabled
   Auto-negotiation Protocol             : IEEE 802.3by & Consortium
   Media Auto Detect                     : Enabled
   Link Training                         : Disabled
   Default EVB Mode                      : VEB
   Live Firmware Upgrade                 : Enabled
   Adapter Error Recovery                : Disabled
   PME Capability                        : Enabled
   Open Virual Switch                    : Enabled
   Port Enablement                       : Enabled
 Allow recovery of firmware from fatal errors w/o manual intervention. Linux only
 [→][Enter][Space]:Toggle Value;[↑↓]:Next Entry; [ESC] Quit [PgUp][PgDn] Page Up/Dn
    Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

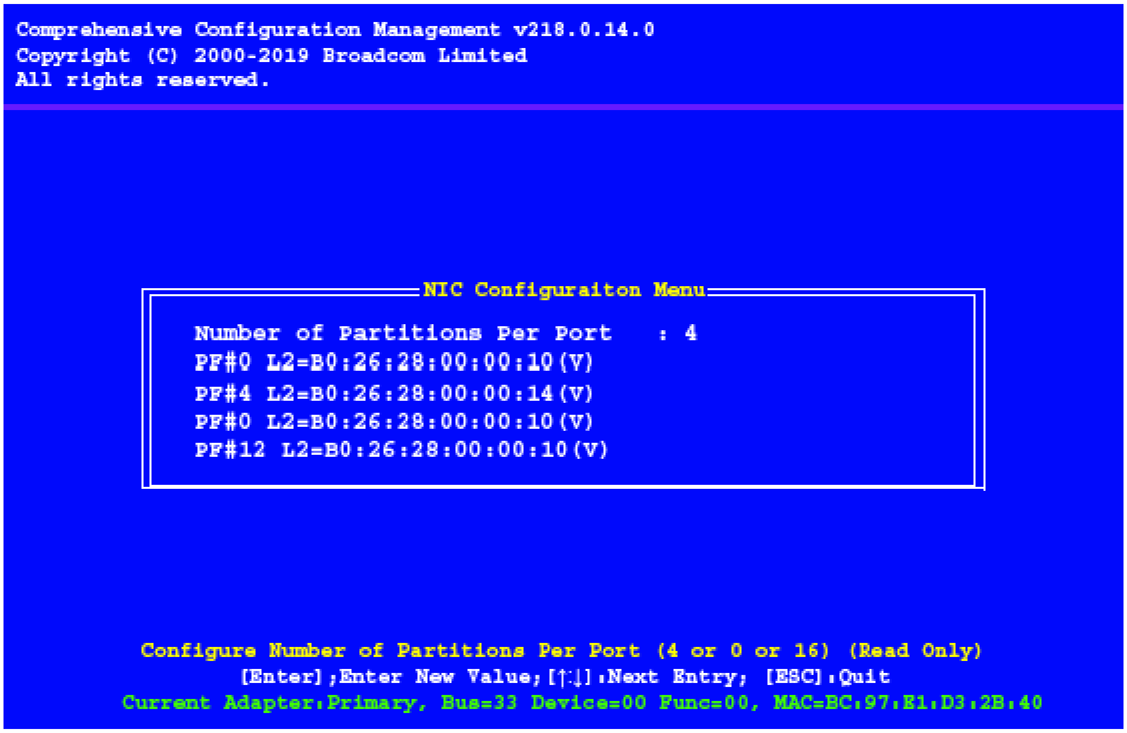
NOTE
For some ARI capable OEM systems, the **NParEP** button is available to explicitly allow the NetXtreme-E to support up to 16 partitions. Switching from single function mode to multifunction mode, the device needs to be re- enumerated, therefore changes do not take effect until a system reboot occurs.

2. Once NPAR is enabled, the **NIC Partitioning Main Configuration** menu option is available from the main NIC Configuration Menu associated with each physical port.



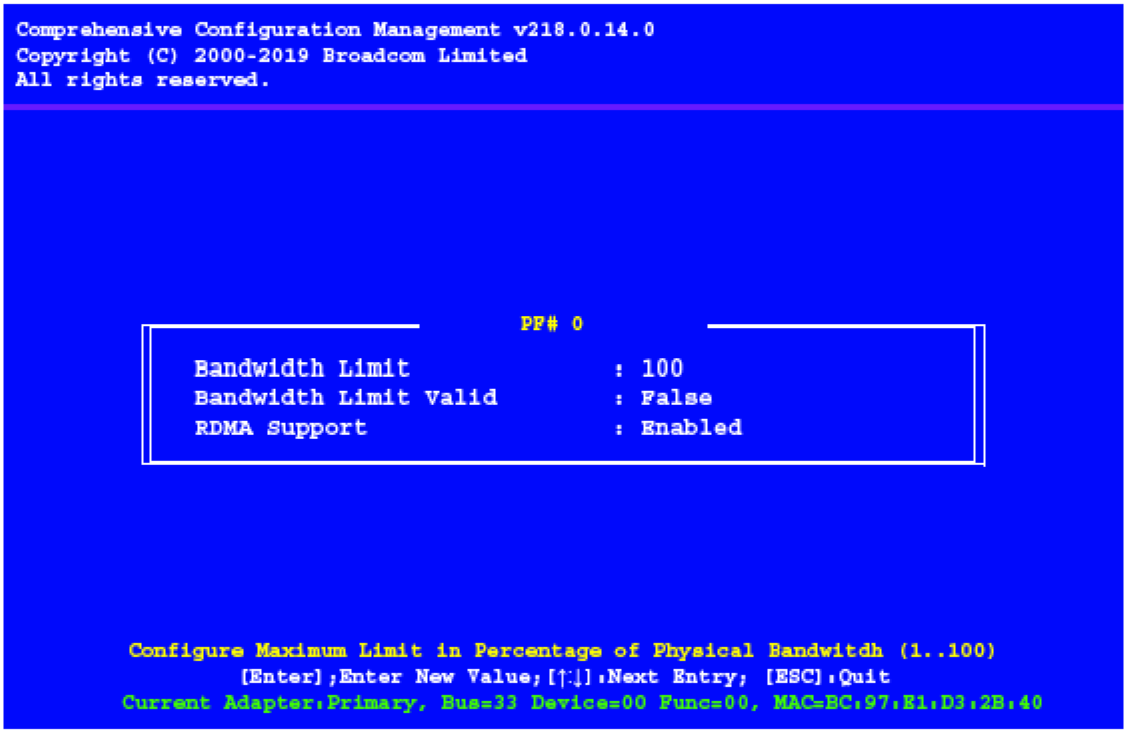
```
Comprehensive Configuration Management v216.0.47.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                         Main Menu
              Firmware Image Menu
              Device Hardware Configuration
              MBA Configuration
              iSCSI Boot Configuration
              NIC Partition Configuration
              BLINK LEDs    : 0

                  Listing of firmware image versions
            [Enter]:Enter;[↑:↓]:Next Entry; [ESC] Quit Menu
    Current Adapter:Primary, Bus=86 Device=00 Func=00, MAC=B0:26:28:CB:21:40
```

3. The NIC Partition Configuration Menu (shown below) allows the user to choose the number of partitions that should be allocated from the selected physical port. Each NetXtreme-E NIC can support a maximum of 16 partitions on an ARI capable server. By default, dual-port adapters are configured for eight partitions per physical port. Configuration options for each partition are also accessible from this menu. For some OEM systems, the HII menu also includes a Global Bandwidth Allocation page where the minimum (reserved) and maximum (limit) TX bandwidth for all partitions can be configured.



```
Comprehensive Configuration Management v218.0.14.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                        NIC Configuraiton Menu

      Number of Partitions Per Port    : 4
      PF#0 L2=B0:26:28:00:00:10(V)
      PF#4 L2=B0:26:28:00:00:14(V)
      PF#0 L2=B0:26:28:00:00:10(V)
      PF#12 L2=B0:26:28:00:00:10(V)

    Configure Number of Partitions Per Port (4 or 0 or 16) (Read Only)
           [Enter];Enter New Value;[↑:↓]:Next Entry; [ESC]:Quit
   Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

4. Set the NIC Partition Configuration parameters.



```
Comprehensive Configuration Management v218.0.14.0
Copyright (C) 2000-2019 Broadcom Limited
All rights reserved.

                              PF# 0

      Bandwidth Limit                 : 100
      Bandwidth Limit Valid           : False
      RDMA Support                    : Enabled

   Configure Maximum Limit in Percentage of Physical Bandwitdh (1..100)
           [Enter];Enter New Value;[↑:↓]:Next Entry; [ESC]:Quit
   Current Adapter:Primary, Bus=33 Device=00 Func=00, MAC=BC:97:E1:D3:2B:40
```

Table 35: NPAR Paremeters

Parameter	Description	Valid Options
BW Limit	Maximum percentage of available bandwidth this partition is allowed.	Value 0 to 100
BW Limit Valid	Functions as an on/off switch for the BW Limit setting.	True/False
RDMA Support	Functions as an on/off switch for RDMA support on this partition. **NOTE:** Only two partitions per physical port can support RDMA. For a dual-port device, up to 4 NPAR partitions can support RDMA.	Enabled/Disabled

NOTE

The NPAR minimum bandwidth parameter cannot be changed on the BCM575XX. BCM5741X devices are permitted to change the NPAR minimum value.

Reducing NIC Memory Consumption with NPAR

The default value of receive buffers was selected to work well for typical configurations. If you have many NICs in a system, have enabled NPAR on multiple NICs, or if you have only a small amount of RAM, you may see a Code 12 yellow bang in the Device Manager for some of the NICs. Code 12 means that the driver failed to load because there were not enough resources. In this case, the resource is a specific type of kernel memory called Non-Paged Pool (NPP) memory.

If you are getting a Code 12, or for other reasons wish to reduce the amount of NPP memory consumed by the NIC:

- Reduce the number of RSS queues from the default of 8 to 4 or 2. Each RSS queue has its own set of receive buffers allocated, so reducing the number of RSS queues reduces the allocated NPP memory. There can be performance implications from reducing the number of RSS queues, as fewer cores participate in processing receive packets from that NIC. Per processor CPU utilization should be monitored to ensure that there are no “hot” processors after this change.
- Reduce memory allocation by reducing the number of receive buffers allocated. The default value of 0 means the driver should automatically determine the number of receive buffers. For typical configurations, a setting of 0 (=auto) maps to XXXX receive buffers per queue. You can choose a smaller value such as 1500, 1000, or 500. (The value needs to be in multiples of 500 and between the range of 500 and 15000.) As mentioned above, a smaller number of receive buffers increases the risk of packet drop and a corresponding impact to packet retransmissions and decreased throughput.

The parameters “Maximum Number of RSS Queues” and “Receive Buffers (0=Auto)” can be modified using the **Advanced** properties tab for each NIC in the **Device Manager**. If you want to modify multiple NICs at the same time, it is faster to use the *Set-NetAdapterAdvancedProperty PowerShell cmdlet*. For example, to assign two RSS queues for all NICs in a system whose NIC name starts with “Sl”, run the following command:

```
Set-NetAdapterAdvancedProperty Sl* -RegistryKeyword *NumRSSQueues -RegistryValue 2
```

Similarly, to set the number of Receive buffers to 1500, run the following command:

```
Set-NetAdapterAdvancedProperty Sl* -RegistryKeyword *ReceiveBuffers -RegistryValue 1500
```

For an overview of how to use PowerShell to modify NIC properties, see Microsoft.com.

DPDK Tunings

Broadcom publishes the DPDK performance report to dpdk.org, this report contains the achieved performance numbers and configuration details. The latest version of the report can be accessed here.

This section provides the following information on DPDK tunings:

- BIOS Tuning
- Kernel Tuning
- PCIe Configuration
- DPDK Configuration
- DPDK Results

BIOS Tuning

See BIOS Tuning and set the following BIOS options:

- Local APIC Mode – x2 APIC
- NUMA nodes per socket – NPS1
- L3 Cache as NUMA – Disabled
- Memory Clock Speed – 1467
- PCIe Ten Bit Tag – Enabled
- Preferred I/O – Manual
- Preferred I/O BUS – (Provide BUS ID)
- Enhanced Preferred I/O – Auto
- Determinism Control – Manual
- Determinism Slider – Performance
- xGMI Link Width Control – Manual
- xGMI Force Link Width – 2
- xGMI Force Link Width Control – Force
- xGMI Max Link Width Control – Auto
- APBDIS 1 and PState=P0
- SMT Control – Enabled

Kernel Tuning

Add the following entries to the kernel command line:

```
amd_iommu=on iommu=pt nohz=off rcu_nocbs=32-47 isolcpus=32-47 selinux=0 numa_balancing=disable
processor.max_cstate=0 nosoftlockup default_hugepagesz=1G hugepagesz=1G hugepages=64
```

PCIe Configuration

Reduce MRRS to 1024B using the following command (Use default setting. For example, 4K for a 4 x 25 adapter):

```
setpci -s 41:00.0 b4.w=3d57
```

NOTE
See PCIe MRRS (Maximum Read Request Size) as incorrect usage may cause a system crash.

Enable **Relaxed** ordering in adapter.

Example: bnxtnvm -dev=[interface] setoption=pcie_relaxed_ordering enable

DPDK Configuration

This section provides information on DPDK configuration. Driver: vfio-pci

Testpmd Command Line

```
testpmd -l 32,33,34,35,36,37,38,39,63 -n 4 --socket-
mem=4096 --master-lcore 63 -- --txq=8 --rxq=8 --rxd=4096 --
```

```
txd=4096 --nb-cores=8 -i
```

DPDK Results

- http://core.dpdk.org/perf-reports/
- BCM5741X (25G) Results:
  - Forwarding Rate is 30 Mp/s using 64B frame
  - Line-Rate from 128B onwards
- BCM575XX (200G) Results:
  - Forwarding Rate is 102 Mp/s using 64B frame
  - Line-Rate with 1518B

DCBX – Data Center Bridging

Broadcom Ethernet NIC controllers support IEEE 802.1Qaz DCBX as well as the older CEE DCBX specification. DCB configuration is obtained by exchanging the locally configured settings with the link peer. Since the two ends of a link may be configured differently, DCBX uses a concept of *willing* to indicate which end of the link is ready to accept parameters from the other end. This is indicated in the DCBX protocol using a single bit in the ETS Configuration and PFC TLV, this bit is not used with ETS Recommendation and Application Priority TLV. By Default, the is in *willing* mode while the link partner network switch is in *non-willing* mode. This ensures the same DCBX setting on the Switch propagates to the entire network.

Users can manually set the Broadcom Ethernet NIC controller to *non-willing* mode and perform various PFC, Strict Priority, ETS, and APP configurations from the host side. See the driver readme.txt for additional details on available configurations. This section provides an example of how such a setting can be done in Windows with Windows PowerShell. Additional information on DCBX, QoS, and associated use cases are described in additional details in a separate white paper, beyond the scope of this user manual.

The following settings in the UEFI HII menu are required to enable DCBX support:

1. Click **System** > **Setup** > **Device** > **Settings** > **NetXtreme-E** > **NIC** > **Device** > **Configuration**

This section provides the following information on DCBX:

- QoS Profile – Default QoS Queue Profile
- DCBX Mode – Enable (IEEE only)
- DCBX Willing Bit

QoS Profile – Default QoS Queue Profile

Quality of Server (QoS) resources configuration is necessary to support various PFC and ETS requirements where finer tuning beyond bandwidth allocation is needed. The Broadcom Ethernet NIC controller allows the administrator to select between devoting NIC hardware resources to support Jumbo Frames and/or combinations of lossy and lossless Class of Service queues (CoS queues). Many combinations of configuration are possible and therefore can be complicated to compute. This option allows a user to select from a list of precomputed QoS Queue Profiles. These precomputed profiles are design to optimize support for PFC and ETS requirements in typical customer deployments.

The following is a summary description for each QoS Profile.

Table 36: QoS Profiles

Profile No.	Jumbo Frame Support	No. of Lossy CoS Queues/Port	No. of Lossless CoS Queues/Port	Support for 2-Port SKU
Profile #1	Yes	0	1 (PFC Supported)	Yes (25 Gb/s)

Profile #2	Yes	4	2 (PFC Supported)	No
Profile #3	No. (MTU <= 2 KB)	6	2 (PFC Supported)	Yes (25 Gb/s)
Profile #4	Yes	1	2 (PFC Supported)	Yes (25 Gb/s)
Profile #5	Yes	1	0 (No PFC Support)	Yes (25 Gb/s)
Profile #6	Yes	8	0 (No PFC Support)	Yes (25 Gb/s)
Profile #7	This configuration maximizes packet-buffer allocations to two lossless CoS Queues to maximize RoCE performance while trading off flexibility.			
	Yes	0	2	Yes (25 Gb/s)
Default	Yes	Same as Profile #4		Yes

DCBX Mode – Enable (IEEE only)

This option allows a user to enable/disable DCBX with the indicated specification. IEEE only indicates that IEEE 802.1Qaz DCBX is selected.

Windows Driver setting:

After enabling the indicated options in the UEFI HII menu to set firmware level settings, perform the follow selection in the Windows driver advanced properties.

Open Windows Manager → Broadcom Ethernet NIC controller → Advanced Properties → Advanced tab

Quality of Service = Enabled

Priority & VLAN = Priority& VLAN enabled VLAN = <ID>

Set desired VLAN id

To exercise the DCB related command in Windows PowerShell, install the appropriate DCB Windows feature.

1. In the **Task Bar**, right-click the Windows PowerShell icon and then click **Run as Administrator**. Windows PowerShell opens in elevated mode.
2. In the Windows PowerShell console, type:

```
Install-WindowsFeature "data-center-bridging"
```

DCBX Willing Bit

The DCBX willing bit is specified in the DCB specification. If the willing bit on a device is true, the device is willing to accept configurations from a remote device through DCBX. If the willing bit on a device is false, the device rejects any configuration from a remote device and enforces only the local configurations.

Use the following to set the willing bit to True or False. 1 for enabled, 0 for disabled. Example set-netQoSdcbxSetting -Willing 1

Use the following to create a Traffic Class.

```
C:\> New-NetQosTrafficClass -name "SMB class" -priority 4 -bandwidthPercentage 30 -Algorithm ETS
```

> **NOTE**
> By default, all IEEE 802.1p values are mapped to a default traffic class, which has 100% of the bandwidth of the physical link. The command shown above creates a new traffic class to which any packet tagged with eight IEEE 802.1p value 4 is mapped, and its Transmission Selection Algorithm (TSA) is ETS and has 30% of the bandwidth. It is possible to create up to seven new traffic classes. In addition to the default traffic class, there is at most eight traffic classes in the system.

Use the following in displaying the created Traffic Class:

```
C:\> Get-NetQoSTrafficClass

NameAlgorithm Bandwidth(%)Priority

---------------------------
```

[Default]	ETS	70	0-3,5-7
SMB class	ETS	30	4

Use the following in modifying the Traffic Class:

```
PS C:\> Set-NetQoSTrafficClass -Name "SMB class" -BandwidthPercentage 40 PS C:\> get-NetQosTrafficClass

Name Algorithm Bandwidth(%) Priority

------------------------------------------------------------------ [Default] ETS60 0-3,5-7

SMB class ETS404
```

Use the following to remove the Traffic Class:

```
PS C:\> Remove-NetQosTrafficClass -Name "SMB class" PS C:\> Get-NetQosTrafficClass

Name Algorithm Bandwidth(%) Priority

------------------------------------------------------------------

[Default] ETS1000-7
```

Use the following to create Traffic Class (Strict Priority):

```
C:\> New-NetQosTrafficClass -name "SMB class" -priority 4 -bandwidthPercentage 30-Algorithm Strict
```

Enabling PFC:

```
PS C:\> Enable-NetQosFlowControl -priority 4 PS C:\> Get-NetQosFlowControl -priority 4 Priority Enabled

------------------------------------------------------------------

4 True

PS C:\> Get-NetQosFlowControl
```

Disabling PFC:

```
PS C:\> disable-NetQosflowControl -priority 4 PS C:\> get-NetQosFlowControl -priority 4 Priority Enabled

------------------------------------------------------------------

4 False
```

Use the following to create QoS Policy:

```
PS C:\> New-NetQosPolicy -Name "SMB policy" -SMB -PriorityValue8021Action 4 Name : SMB policy

Owner : Group Policy (Machine) NetworkProfile : All Precedence : 127
```

NOTE
The previous command creates a new policy for SMB. SMB is an inbox filter that matches TCP port 445 (reserved for SMB). If a packet is sent to TCP port 445 it is tagged by the operating system with IEEE 802.1p value of 4 before the packet is passed to a network miniport driver. In addition to SMB, other default filters include iSCSI (matching TCP port 3260), NFS (matching TCP port 2049), LiveMigration (matching TCP port 6600), FCOE (matching EtherType 0x8906) and NetworkDirect. NetworkDirect is an abstract layer created on top of any RDMA implementation on a network adapter. NetworkDirect must be followed by a Network Direct

port. In addition to the default filters, a user can classify traffic by application's executable name (as in the first example below), or by IP address, port, or protocol.

Use the following to create QoS Policy based on the Source/Destination Address:

PS C:\> New-NetQosPolicy "Network Management" -IPDstPrefixMatchCondition 10.240.1.0/24 -IPProtocolMatchCondition both -NetworkProfile all -PriorityValue8021Action 7

Name : Network Management Owner : Group Policy (Machine) Network Profile : All Precedence : 127

IPProtocol : Both IPDstPrefix : 10.240.1.0/24 PriorityValue : 7

Use the following to display QoS Policy:

PS C:\> Get-NetQosPolicy Name : Network Management

Owner : (382ACFAD-1E73-46BD-A0A-6-4EE0E587B95)

NetworkProfile : All Precedence : 127 IPProtocol : Both

IPDstPrefix : 10.240.1.0/24 PriorityValue : 7

Name : SMB policy

Owner : (382AFAD-1E73-46BD-A0A-6-4EE0E587B95)

NetworkProfile : All Precedence : 127 Template : SMB PriorityValue : 4

Use the following to modify the QoS Policy:

PS C:\> Set-NetqosPolicy -Name "Network Management" -IPSrcPrefixMatchCondition 10.235.2.0/24 -IPProtocolMatchCondition both -PriorityValue802.1Action 7

PS C:\> Get-NetQosPolicy -name "network management" Name : Network Management

Owner : {382ACFD-1E73-46BD-A0A0-4EE0E587B95}

NetworkProfile : All Precedence : 127 IPProtocol : Both

IPSrcPrefix : 10.235.2.0/24 IPDstPrefix : 10.240.1.0/24 PriorityValue : 7

Use the following to remove QoS Policy:

PS C:\> Remove-NetQosPolicy -Name "Network Management"

RoCE

This section contains the following information on configuring RoCE:

- Validating RoCE Network
- Manually Reconfiguring Network Parameters
- Advanced Network Configuration
- Reconfigure Network Parameters with Automated Installer
- Switch Configuration

Validating RoCE Network

With the switches, servers, and NICs configured, RDMA applications can be deployed to run over the RoCE network. Any libverbs-linked application can be used, with appropriate changes for using IP addresses rather than GUIDs for end-point addressing.

Ensure that the RoCEv2 GID.

```
# cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/1 RoCEv2
```

A typical application of exercising the RoCE interface would be to use rping or ib_write_bw/etc from perftest. Perftest can be obtained from GitHub or via most Linux package repositories, such as:

```
Fedora/RHEL/CentOS: yum install perftest
Ubuntu: apt install perftest
```

For example, the perftest tools can be run as:

```
# ib_write_bw -d bnxt_re0 -x 3 -F --report_gbits -p 1800 -s <message_size> -q <num of qps> -D <duration>
 192.168.30.3
```

```
# ib_write_lat -d bnxt_re0 -x 3 -F --report_gbits -p 1800 -s 2 192.168.30.3
```

For a complete understanding of the parameters used by `ib_write_bw/lat` and other IB utilities, refer to the Linux man pages.

Manually Reconfiguring Network Parameters

The Quality of Service parameters of Broadcom RNICs can be changed from the default values to match the network. The available parameters are:

- Congestion Control: Enable/Disable
- Priority Flow Control: Enable/Disable
- RoCE and general IP traffic priority
- Bandwidth allocation ratio for RoCE and IP using ETS
- ESCP priority for RoCE and CNP traffic
- VLAN priority for RoCE and CNP traffic
- Congestion control internal tunings

Most configuration is done using either the `bnxtqos` tool or LLDP Agent. Either may be used, but `bnxtqos` is preferred. The `bnxt_setupcc.sh` script is also provided which simplifies the process for configuration of network parameters, and it uses bnxtqos or lldptool.

Persistent Configuration Parameters

The following table provides the configuration parameters stored in non-volatile memory (NVRAM) on the NIC for firmware-based DCBx that must be set to support the MP12 QoS profile for Windows and VMware operating systems. For Linux, enable the LLDP service using systemd and disable `lldp_nearest_bridge` , `lldp_nearest_non_tpmr_bridge` , and `dcbx_mode` in the NIC NVRAM.

Table 37: NVRAM Firmware Parameters

NVM Parameter Name	NVM Parameter Description	NVM Parameter Supported Values	Configured Value
`support_rdma`	This option indicates if a PF supports RDMA	Disabled(0) Enabled(1)	0
`dcbx_mode`	This option specifies the DCBX mode configuration per port	Disabled(0) Enabled (IEEE only)(1) CEE (only)(2) Both (IEEE preferred with fallback to CEE)(3)	0
`lldp_nearest_bridge`	This option allows enabling Link Layer Data Protocol (LLDP) on the nearest bridge, per port	Disabled(0) Enabled (IEEE only)(1) CEE (only)(2) Both (IEEE preferred with fallback to CEE)(3)	0

NVM Parameter Name	NVM Parameter Description	NVM Parameter Supported Values	Configured Value
`lldp_nearest_non_tpmr_bridge`	This option allows enabling Link Layer Data Protocol (LLDP) on non-TPMR (two port MAC Relay) bridge, per port	Disabled(0) Enabled(1)	0

1. Enable RDMA support for PF0 using the following command:

   ```
   sudo rmmod bnxt_re
   sudo bnxtnvm -dev=p1p1 setoption=support_rdma:0#1
   sudo bnxtnvm -dev=p1p1 reset
   ```

   NOTE
   The notification to reboot can be ignored if this is the only option being changed.
2. Disable in-firmware DCBx with the following command:

   ```
   sudo bnxtnvm -dev=p1p1 setoption=dcbx_mode:0:0
   ```

   NOTE
   This procedure must run after the IP driver has loaded.
3. Reboot the system to apply the changes.

DSCP and VLAN

Ethernet and Broadcom RNICs support two methods for marking traffic with a priority: VLAN and DSCP. If your network uses VLAN tagging for traffic, VLAN priority may be used to separate RoCE traffic from other traffic for congestion control. If your network does not use VLAN tags, DSCP must be used. In either case, the corresponding switch must be configured correctly to ensure the priority (VLAN or DSCP) map to a specific traffic class priority on the switch that has ECN and/or PFC enabled and configured.

For RoCE deployment, specific VLAN PRI or DSCP values must be configured in the network and the host for RoCE, IP, and, RoCE CNP traffic. No industry standard defines specific VLAN PRI and DSCP values for RoCE traffic. In order to simplify RoCE deployment, the following recommended values of VLAN PRI and DSCP should be used (see the following table).

> **NOTE**
> When VLAN priority is used, take care when the frames are transmitted over trunk network. It is possible that the VLAN priority can be dropped when multiple trunks are not configured for correct priority.

Table 38: VLAN PRI and DSCP Recommended Values

Traffic	VLAN PRI	DSCP
RoCE (lossless)	3 (Flash)	26 (Flash)
IP (lossy)	0 (Best Effort)	0 (Best Effort)
RoCE CNP (lossy SP)	7 (Network Control)	48 (Network Control)

Bandwidth Allocation

Enhanced Transmission Selection (ETS) is used to provide minimum bandwidth allocation values for both RoCE and IP traffic. ETS only comes into effect when both IP and RoCE traffic is scheduled for transmission, causing congestion in the server NIC. By default, 90% of traffic is allocated to RoCE, and 10% for IP.

Even if your use case is entirely RoCE, some bandwidth must remain allocated for IP at least for control such as SSH and the RDMA connection manager.

The ETS ratio is set using either bnxtqos or lldptool.

bnxtnvm Utility

The Broadcom bnxtnvm utility allows setting non-volatile configuration elements of the RNIC, such as enabling/disabling RoCE, SR-IOV, and other options. bnxtnvm is the preferred tool for firmware upgrades.

Enable RDMA and Disable DCBx

NVRAM Firmware Parameters provide the NVRAM configuration parameters for firmware-based DCBx that must be set to support the MP12 QoS profile for Windows and VMware operating systems. For Linux, enable the LLDP service using systemd, and disable `lldp_nearest_bridge` , `lldp_nearest_non_tpmr_bridge` , and `dcbx_mode` in the NIC NVRAM.

Table 39: NVRAM Firmware Parameters

NVM Parameter Name	NVM Parameter Description	NVM Parameter Supported Values	Configured Value
`support_rdma`	This option indicates if a PF supports RDMA.	{Disabled(0) Enabled(1)}	1
`dcbx_mode`	This option specifies the DCBX mode configuration per port.	{Disabled(0) Enabled (IEEE only)(1) CEE (only)(2) Both (IEEE preferred with fallback to CEE)(3)}	Disabled
`lldp_nearest_bridge`	This option allows enabling Link Layer Data Protocol (LLDP) on the nearest bridge, per port.	{Disabled(0) Enabled(1)}	0
`lldp_nearest_non_tpmr_bridge`	This option allows enabling Link Layer Data Protocol (LLDP) on non-TPMR (two port MAC Relay) bridge, per port.	{Disabled(0) Enabled(1)}	0

1. Enable RDMA support for PF0 using the following command:
   ```
   sudo bnxtnvm -dev=p1p1 setoption=support_rdma:0#1
   ```
2. Disable in-firmware DCBx with the following command:
   ```
   sudo bnxtnvm -dev=p1p1 setoption=dcbx_mode:0:0
   ```
   NOTE
   This procedure must run after the IP driver has loaded.
3. Reboot the system to apply the changes.

Configuring Congestion Control with bnxt_setupcc.sh

The `bnxt_setupcc.sh` tool uses either bnxtqos or lldptool to configure network parameters with a single command.

Table 40: bnxt_setupcc.sh

Option	Description
`-d`	RoCE Device Name (for example, `bnxt_re0` , `bnxt_re_bond`).
`-i`	Ethernet Interface Name (for example, p1p1 or for bond, specify slave interfaces like -i p6p1 -i p6p2).
`-r [0-7]`	RoCE packet priority.

Option	Description
`-s VALUE`	RoCE packet DSCP value.
`-c [0-7]`	RoCE CNP packet priority.
`-p VALUE`	RoCE CNP packet DSCP value.
`-v VALUE`	Use specified VLAN ID instead of DSCP
`-b VALUE`	RoCE Bandwidth percentage for ETS configuration - Default is 80%
`-h`	Display help.
`-m [1-3]`	1 – PFC only. 2 – CC only. 3 – PFC + CC mode.

Example: Default Values

This example configures the specified interface to the same values used by default.

```
cd $REL_DIR/Linux/netxtreme-bnxt_en-x.y.z/bnxt_re
./bnxt_setupcc.sh -d bnxt_re0 -i ens4f0np0 -m 3 -s 26 -p 48 -r 3 -c 7
```

Example: VLAN 5, PFC Only

This example configures the specified interface to the same values used by default.

```
cd $REL_DIR/Linux/netxtreme-bnxt_en-x.y.z/bnxt_re
./bnxt_setupcc.sh -d bnxt_re0 -i ens4f0np0 -m 1 -v 5 -r 3 -c 7
```

Advanced Network Configuration

The default RoCE settings are usually close to optimal for most users and use-cases. However, some users may wish to use different configurations to match existing network settings or optimize for specific applications. This section provides background into the internal Congestion Control and QoS implementation of Broadcom RNICs, and steps for changing each of the relevant settings.

The RoCE configuration defaults for the Linux drivers are:

- RoCEv2 (RDMA over IPv4) enabled
- Congestion Control and PFC enabled
- RoCE traffic tagged with DSCP value 26 on priority 3
- RoCE CNP traffic tagged with DSCP value 48 on priority 7
- 1500-Byte MTU

Table 41: Common Changes from the Default Configuration

Control Type	Behavior	Example Applications
Disable PFC (Congestion Control Only)	Removes risk of traffic blocking and congestion spreading. Incompatible with “unreliable” RDMA connection types. (UD, UC)	Specify “-o ECN” to installer
Disable ECN (PFC Only)	Improves throughput for very bursty traffic. Incompatible with multi-hop switch fabrics.	Specify “-o PFC” to installer
Use VLAN tags instead of DSCP	Appends VLAN header to RoCE traffic. Uses VLAN priority field to classify packets instead of IP header DS field	Specify “-q <VID>” to installer

Control Type	Behavior	Example Applications
Use Jumbo Frames	Increases maximum throughput. All devices on network must use the same MTU. Typical large value is 9000.	Specify “-m <MTU>” to the installer

Ensure that the RNIC configuration matches the switch fabric configuration.

Congestion Control

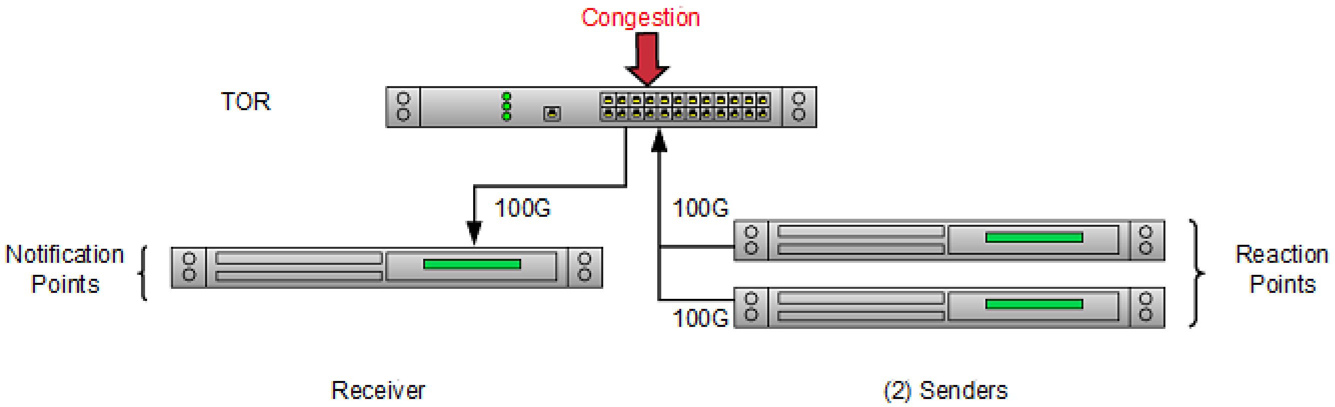
RoCE Congestion Control (CC) relies on ECN marking on the switch. When the switch accumulates excess incoming traffic in its buffers due to congestion, the switch output queue level rises. When the output queue level rises above a configured minimum threshold, the switch starts marking outgoing packets with ECN to indicate congestion. When the receiving NIC receives an ECN marked packet it transmits a Congestion Notification Packet (CNP) to the sender and specifically to the flow (QPN – QP number) the packet belongs to. The sender is the NIC that transmitted the original marked packet. Upon receiving a CNP, the sending NIC reduces the transmission rate for the particular flow (QP).

It is important to note that the sender only reduces rate on a particular flow (QP), the one to which the CNP was destined. This behavior is critical to maintaining good network utilization by only penalizing (reducing rate) flows that contribute to congestion.

The three network components involved in avoiding congestion are:

- The sending NIC – Reaction point as it reacts to congestion indication.
- The switch – Congestion point, where congestion is detected.
- The receiving NIC – Notification point which simply transmits CNP when receiving ECN parked data packet.

Figure 37: Congestion Control Topology



The two main components that determine the congestion control efficiency are the marking policy at the switch and the sending NIC behavior. The set of rules for the behavior of the sending NIC when there is congestion and when there is not, is the congestion control algorithm.

The marking policy and the CC algorithm significantly affect the following:

- Network utilization.
- How likely switch buffers are to fill up.
- End-to-end latency through the network.

Each flow should react correctly and quickly to congestion, and quickly captures the available link BW when there is no congestion, while keeping enough traffic flowing through the switch to utilize the link toward the receiving NIC, but with little queuing delay through the switch. Queuing delay through the switch is the result of the steady-state queue level, or how many packets are buffered in the switch. Minimizing the queue level during congestion directly reduces the end-to-end latency.

Deterministic Marking

In deterministic marking, once the queue level rises above a configured level, all packets are marked. Therefore, all flows receive immediate congestion indication and reduce their rate. The BCM9575XX/BCM95741X adopts a CC algorithm designed for deterministic marking policy and very-low queuing latency called DCTCP. With this algorithm, the marking threshold is very low (few tens of KB) and therefore, flows react quickly to congestion while keeping the number of packets buffered in the queue very small. The queuing latency through congested switches is very low. This may come with a cost of slightly lower egress link utilization.

The BCM9575XX/BCM95741X provides flexibility in CC settings to control how aggressively flows start transmission vs. how likely the switch buffers may fill up in the event of a large incast. When a large number of flows on the network start transmission toward the same destination at the same time, the switch will need to absorb the initial, or transient, excess traffic until the flows receive congestion indication and reduce rate. The more aggressive (higher) initial rate, the more buffering is needed for a given number of starting flows. If switch buffering is exhausted, then the switch asserts flow control if it is configured to do so, or it drops packets. Depending on the expected burstiness of traffic on the network, the optimal setting can be selected.

Quality of Service

Broadcom RNICs support multiple Class of Service Queues (CoSQ) per port. QoS policy is based on traffic classes (TCs) that are mapped 1:1 to CoS queues. VLAN priority to CoSQ mapping is supported as well as DSCP to CoSQ mapping. At the NIC driver level, each transmit ring or RoCE Queue Pair (QP) is associated with a particular CoSQ. This configuration is performed by the NIC driver. On each port, the NIC schedules traffic across multiple CoSQs.

A finite amount of on-chip buffering is available for transmitting and receiving packets. These buffers are referred to as TX and RX MBUFs. On the transmit side, the TX MBUF is used to stage packets that are being transmitted on the network. The host software initiates the transmission of these packets. The RNIC processes, stages, and builds packets in the TX MBUF before they are being transmitted on the network. On the receive side, the RX MBUF is used to stage packets received from the network before the packets or packet payloads are transferred into the host memory.

A CoS Profile (also known as MBUF profile) is a CoSQ and MBUF configuration to enforce a specific QoS policy on a port. An MBUF configuration profile has the following attributes:

- The number of CoSQs.
- The specific CoSQ enablement.
- The type of service such as lossy or lossless per CoSQ.
- The buffer configuration for each CoSQ.
- Pause/PFC the threshold for each CoSQ.

Three transmit and receive class-of-service queues (CoSq) are allocated for each Ethernet port: 0, 4, and 5. By default, all CoSq's are configured for weighted-fair-queueing (WFQ), with priority 0 traffic mapped to CoSq 4. Configure priority-to-CoSq mappings to differentiate traffic types among the allocated class of service queues. See Configuring Congestion Control with bnxt_setupcc.sh for additional information on configuring priorities.

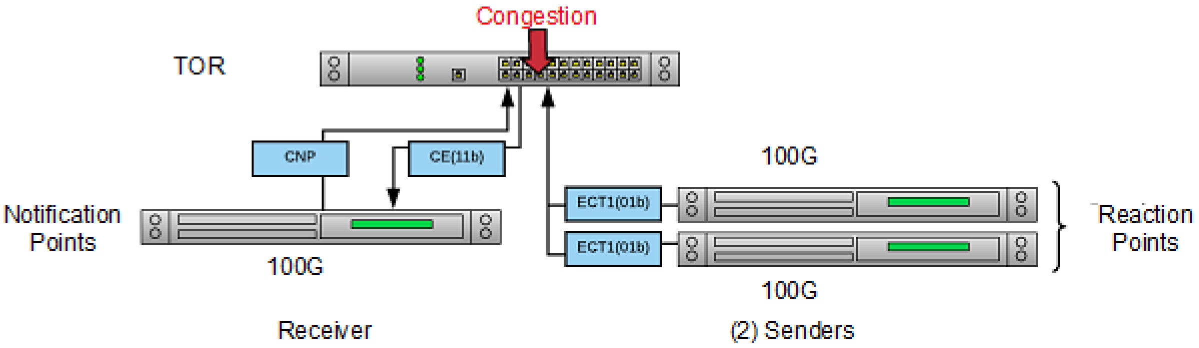
When the RoCE bnxt_re driver is loaded, CoSq 0 is configured for lossless traffic, CoSq 5 is changed from WFQ to strict priority (SP) for CNP processing.

RoCE and CNP traffic may be tagged with different DSCP values, or use VLAN tags instead.

Traffic Control Synopsis

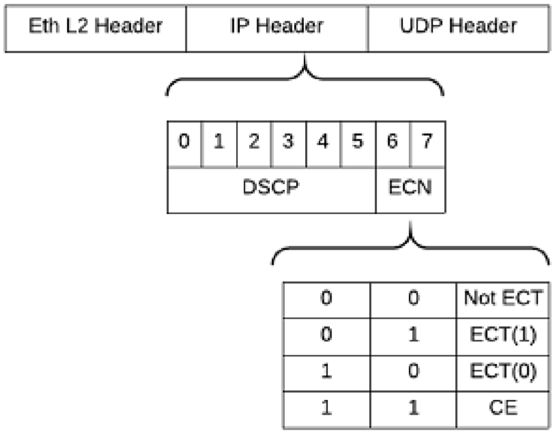
In this section, an example scenario of congestion demonstrates the behavior and function of congestion control.

Figure 38: Congestion Control Example Network



1. Three servers are connected to a switch, and configured for RoCE with the default settings (congestion control and PFC are enabled). The switch is configured to enable ECN and PFC on the RoCE priority.
2. Network congestion is caused by two servers transmitting data to the third server simultaneously, at a speed above the link rate of the receiver. This causes packets to be buffered for a time in the switch before being transmitted. If the buffer fills up, it will drop packets on ingress, which will cause a cycle of retransmits and additional congestion.
3. If ECN is enabled (default), then the initial QP transmit rate will begin at a fraction of line-rate, and quickly ramp up until congestion is encountered or line-rate is reached. If ECN is not enabled, RDMA transmissions are sent at line rate.
4. RoCE packets from the transmitting hosts are tagged as ECN enabled by setting the ECN field in the IP header to ECT1(01b) or ECT0(10b).

Figure 39: ECN Congestion Management



Eth L2 Header	IP Header	UDP Header

0	1	2	3	4	5	6	7
DSCP						ECN	

0	0	Not ECT
0	1	ECT(1)
1	0	ECT(0)
1	1	CE

5. As the egress buffer of the switch reaches the ECN congestion marking threshold as defined by the user, the switch starts marking the ECN bits to CE(11b) to inform the notification point server (receiver) that congestion has been detected for that port.
6. If the buffer of the switch reaches the PFC threshold, PFC frames will be sent to the transmitting servers, instructing them to stop transmitting. This prevents packets from being dropped by the switch but can lead to increased latency and latency jitter. It is possible that switch ingress ports that are not contributing significantly to congestion also transmit PFC frames to their link partners. This is because the switch cannot predict the destination of packets that it may receive, so it needs to make sure that arriving packets have a place to land in switch memory independent of destination. This behavior can result in head of line blocking, since packets destined to non-congested destinations may be held at the link partner.
7. If the network is multi-switch then congestion spreading can occur, due to switch links being throttled because of congestion on only one switch egress port.

8. Upon the reception of a RoCE packet with the ECN CE flag set, the notification point (receiving server) transmits CNP packet(s) to the reaction points (transmitting servers) to signify congestion.
9. The reaction point correlates received CNP packets to a specific QP, and adjusts the transmit rate of that QP in order to reduce congestion based on the algorithm described in Deterministic Marking.
10. QP transmit rate frequently re-assessed to allow increased speed during times of reduced congestion.

Congestion Control Tuning Parameters

Setting Congestion Control Parameters outlines the congestion control parameters that are currently configurable. Updating the parameters is currently done through configfs files. It is not normally necessary to adjust these values.

Table 42: Congestion Control Tuning Parameters

CC Parameter	Description
`cc_mode`	0 for **DCTCP** algorithm (with deterministic marking), 1 for **interop with DCQCN** (probabilistic marking). Default 0.
`ecn_enable`	Enable congestion control.
`g`	Running average weight in computing cp (Congestion Probability). The weight is 2^g, g between 1 and 9, default 8.
`init_cr`	Current rate after QP creation and after QP has not transmitted for more than `inctivity_th` . Value between 0 and 1023 as fraction of full link BW. The driver configures the value when the PF is enabled.
`init_tr`	Target rate after on QP creation and after QP has not transmitted for more than `inctivity_th` . Value between 0 and 1023 as fraction of full link BW.
`rtt`	Time period [us] over which cnp and transmitted packets counts accumulate. At the end of Rtt ratio between CNPs and TxPkts is computed and the CP is updated, the default value is 40 µs.
`inact_cp`	Inactivity time after which the CC parameters of a QP are re-initialized, default 10000 µs.
`cnp_dscp`	DSCP value for RoCE congestion notification packets.
`roce_dscp`	DSCP value for RoCE Packets.
`cnp_prio`	Priority for RoCE congestion notification packets.
`roce_prio`	Priority for RoCE packets.
`apply`	Applies the settings.

Setting Congestion Control Parameters

To manually set congestion control parameters, write to the device's configfs files. For example, these commands set the default values:

```
mkdir -p /sys/kernel/config/rdma_cm/bnxt_re0
echo RoCE v2 > /sys/kernel/config/rdma_cm/bnxt_re0/ports/1/default_roce_mode

mkdir -p /sys/kernel/config/bnxt_re/bnxt_re0
cd /sys/kernel/config/bnxt_re/bnxt_re0/ports/1/cc/

echo -n 0x1a > roce_dscp
echo -n 0x3 > roce_prio
echo -n 0x1 > disable_prio_vlan_tx
echo -n 0x1 > ecn_enable
echo -n 0x30 > cnp_dscp
echo -n 0x7 > cnp_prio

#After setting all the above parameters, apply the values to HW
```

```
echo -n 0x1 > apply
```

Reconfigure Network Parameters with Automated Installer

As described in Installing the L2 and RoCE Drivers Using the Automated Installer, the Linux Installer accepts options for most tuning parameters. The installer can be safely re-run with new parameters, and these changes will be made persistent by installing network interface hook scripts, which call the bnxt_setupcc.sh script (see Configuring Congestion Control with bnxt_setupcc.sh).

For any deployment which uses non-default settings, the recommended process for making these changes is to re-run the Installing the L2 and RoCE Drivers Using the Automated Installer tool.

Switch Configuration

The default configuration of Broadcom RNICs enables the use of both PFC and ECN. To support this configuration, these protocols must be enabled on the network switch. Traffic priority maps must also be configured. To reconfigure Broadcom RNICs for different congestion and flow control settings, see Advanced Network Configuration.

ECN Thresholds

ECN ThresholdsCongestion control uses ECN to react to marked packets within the network switch infrastructure during times of congestion. The correct ECN threshold value is specific to each switch port, and is dependent on the link speed of that port.

Table 43: Marking Values

Switch Egress Port Link Speed (Gb/s)	ECN Min/Max Threshold (kilobytes)
10	12/13
25	16/17
50	24/25
100	64/65
200	64/65
400	130/131

Each switch port participating in congestion control must be configured with the above ECN threshold for the class of service associated with RoCEv2 traffic. Different vendors have different naming conventions that specify the minimum and maximum marking threshold as well as the marking percentage.

Switch Configuration Examples

The switch configuration elements required are as follows:

- Map DSCP traffic priorities for RoCE and CNP traffic to traffic classes
- Enable PFC
- Enable ECN
- Enable ETS

Example: Arista 7060CX

```
qos map dscp 26 to traffic-class 3
qos map dscp 48 to traffic-class 7
!
```

```
interface Ethernet1/1
tx-queue 3
random-detect ecn minimum-threshold 64 kbytes maximum-threshold 65 kbytes max-mark-probability 100
priority-flow-control mode on
priority-flow-control priority 3 no-drop
```

Tuning

Provides information on fine tuning Broadcom Ethernet NIC controllers for improved performance.

This section provides the following tuning information for the Broadcom Ethernet NIC controller:

- BIOS Tuning
- TCP Performance
- Performance Optimization – NetPerf Test
- Example RoCE + TCP Network Configuration

BIOS Tuning

This section covers the various BIOS configuration options to tune the system for optimal performance. The BIOS screens in this section are for reference only and have been captured on AMD EPYC reference platform. It is recommended to find the equivalent settings in the target system BIOS console.

This section provides the following BIOS tuning information for the Broadcom Ethernet NIC controller:

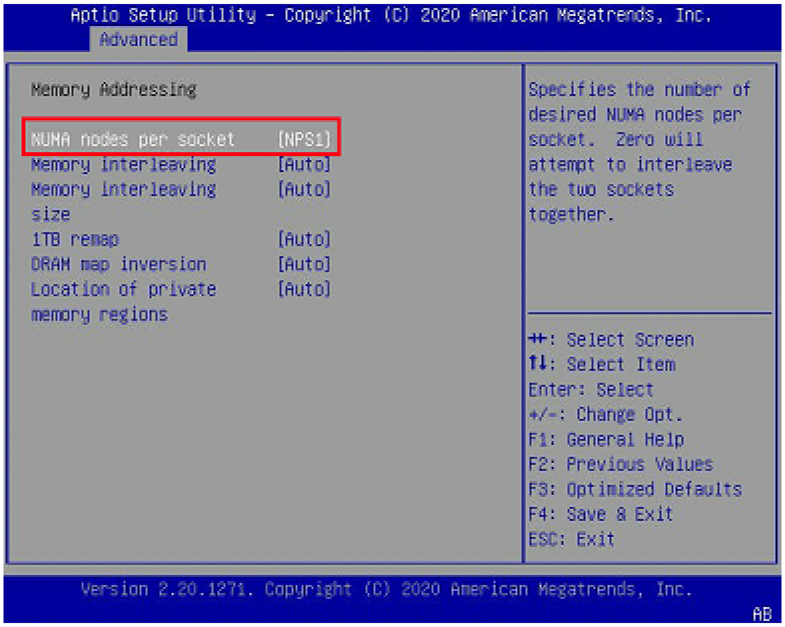
- NPS (NUMA Per Socket)
- X2APIC
- Determinism Control and Determinism Slider
- APBDIS
- Preferred I/O and Enhanced Preferred I/O
- PCIe Ten Bit Tag
- Memory Clock Speed
- L3 LLC (Last Level Cache) as NUMA
- Socket/Inter-Chip Global Memory Interconnect (xGMI)

NPS (NUMA Per Socket)

NOTE
NPS=1 configuration is recommended for 200 Gb/s and above. NPS=4 is recommended for up to 100 Gb/s which provides better CPU and memory locality.

To access the NUMA nodes per socket setting, select **Advanced** > **AMD CBS** > **DF Common Options** > **Memory Addressing** > **NUMA Nodes Per Socket** > **NPS1 Socket** > **NPS1**

Figure 40: NUMA Nodes Per Socket Settings



```
        Aptio Setup Utility – Copyright (C) 2020 American Megatrends, Inc.
              Advanced
  Memory Addressing                         Specifies the number of
                                            desired NUMA nodes per
  NUMA nodes per socket      [NPS1]         socket.  Zero will
  Memory interleaving        [Auto]         attempt to interleave
  Memory interleaving        [Auto]         the two sockets
  size                                      together.
  1TB remap                  [Auto]
  DRAM map inversion         [Auto]
  Location of private        [Auto]
  memory regions
                                            ++: Select Screen
                                            ↑↓: Select Item
                                            Enter: Select
                                            +/-: Change Opt.
                                            F1: General Help
                                            F2: Previous Values
                                            F3: Optimized Defaults
                                            F4: Save & Exit
                                            ESC: Exit
        Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                      AB
```

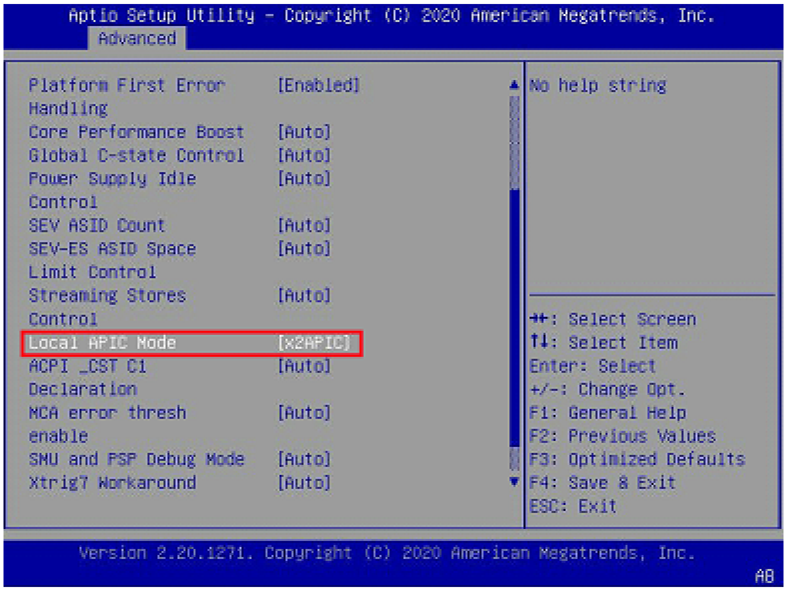
X2APIC

Set **X2APIC** = Enabled to allow the OS to work with 256 threads and improve performance over legacy APIC.

> **NOTE**
> Disable SMT if you are running an OS that does not support X2APIC and has a dual socket 64 core processor.

To access the **Local APIC Mode** setting, select**Advanced** > **AMD CBS** > **CPU Common Options** > **Local APIC Mode** > **X2APIC** > **NPS1 Socket**

Figure 41: Local APIC Mode Settings



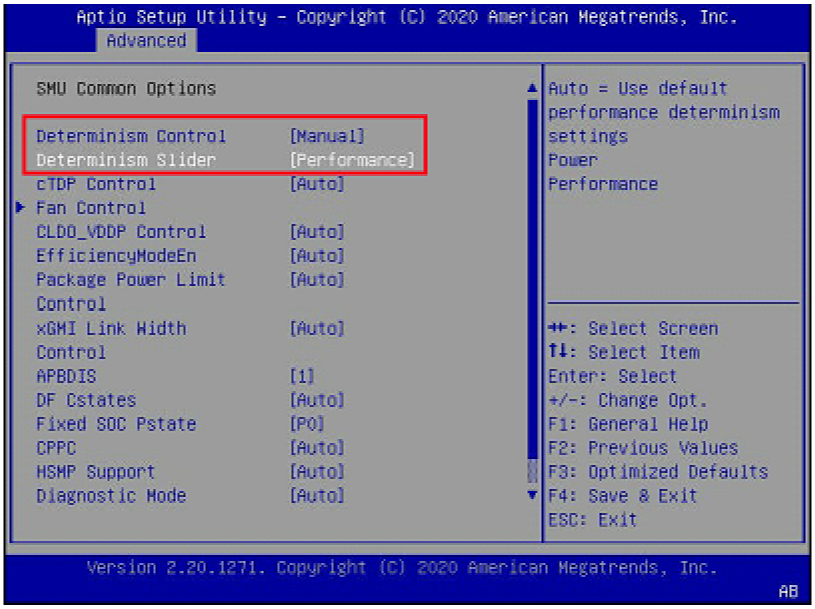
```
        Aptio Setup Utility – Copyright (C) 2020 American Megatrends, Inc.
              Advanced
  Platform First Error       [Enabled]    ▲ No help string
  Handling
  Core Performance Boost     [Auto]
  Global C-state Control     [Auto]
  Power Supply Idle          [Auto]
  Control
  SEV ASID Count             [Auto]
  SEV-ES ASID Space          [Auto]
  Limit Control
  Streaming Stores           [Auto]
  Control                                   ++: Select Screen
  Local APIC Mode            [x2APIC]       ↑↓: Select Item
  ACPI _CST C1               [Auto]         Enter: Select
  Declaration                               +/-: Change Opt.
  MCA error thresh           [Auto]         F1: General Help
  enable                                    F2: Previous Values
  SMU and PSP Debug Mode     [Auto]         F3: Optimized Defaults
  Xtrig7 Workaround          [Auto]       ▼ F4: Save & Exit
                                            ESC: Exit
        Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                      AB
```

Determinism Control and Determinism Slider

Set **Determinism Control**to **Manual** and the **Determinism Slider** to **Performance**(see Figure 7) to ensure consistent performance across a fleet of similarly configured systems.

1. To access the **Determinism Control** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **SMU Common Options** > **Determinism Control** > **Manual**
2. To access the **Determinism Slider** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **Determinism Slider** > **Performance**

Figure 42: Determinism Control/Determinism Slider Settings



```
        Aptio Setup Utility – Copyright (C) 2020 American Megatrends, Inc.
           Advanced

  SMU Common Options                                Auto = Use default
                                                    performance determinism
  Determinism Control       [Manual]                settings
  Determinism Slider        [Performance]           Power
  cTDP Control              [Auto]                  Performance
▶ Fan Control
  CLDO_VDDP Control         [Auto]
  EfficiencyModeEn          [Auto]
  Package Power Limit       [Auto]
  Control
  xGMI Link Width           [Auto]                  →←: Select Screen
  Control                                           ↑↓: Select Item
  APBDIS                    [1]                     Enter: Select
  DF Cstates                [Auto]                  +/-: Change Opt.
  Fixed SOC Pstate          [P0]                    F1: General Help
  CPPC                      [Auto]                  F2: Previous Values
  HSMP Support              [Auto]                  F3: Optimized Defaults
  Diagnostic Mode           [Auto]                  F4: Save & Exit
                                                    ESC: Exit

          Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                        AB
```

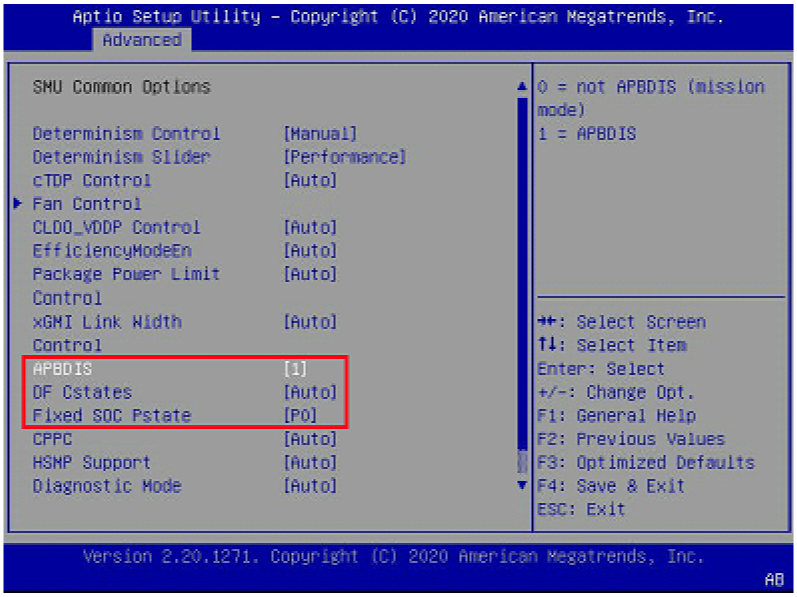
APBDIS

Set **APBDIS**=1 to disable **Algorithmic Performance Boost** which subsequently disables the switching of P- states in infinity fabric (CPU P-states remain unaffected) and forces the system to be in P0 state, which is the highest performing infinity fabric P-state. The **APBDIS** states are as follows:

- 0: Disable APBDIS – Locks the fabric clock to the non-boosted speeds.
- 1: Enable APBDIS – Unlocks the fabric clock to the boosted speeds.
- Auto (Default setting) – Use the default value for APBDIS. The default value is 0.

1. To access the **APBDIS** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **SMU Common Options** > **APBDIS** > **1**
2. To access the **Fixed SOC Pstate** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **SMU Common Options** > **Fixed SOC Pstate** > **P0**

Figure 43: APBDIS Settings



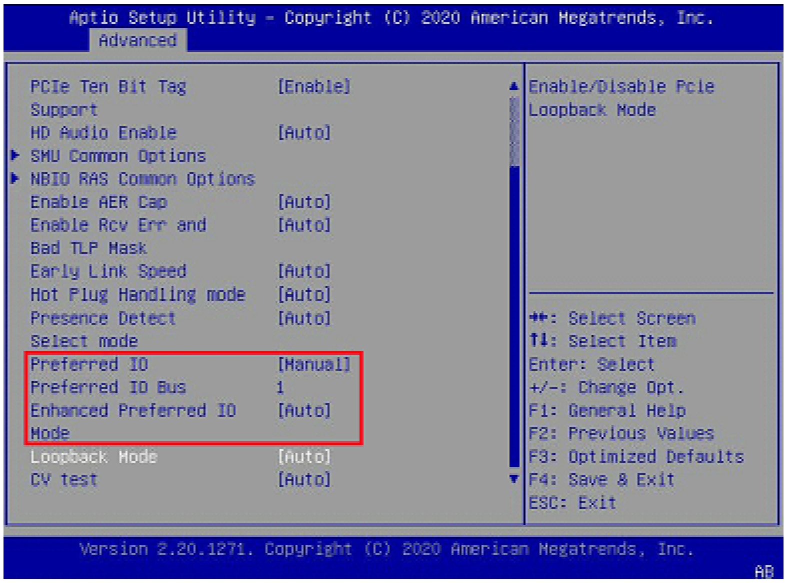
```
        Aptio Setup Utility - Copyright (C) 2020 American Megatrends, Inc.
              Advanced

   SMU Common Options                                 0 = not APBDIS (mission
                                                      mode)
   Determinism Control       [Manual]                 1 = APBDIS
   Determinism Slider        [Performance]
   cTDP Control              [Auto]
 ▶ Fan Control
   CLDO_VDDP Control         [Auto]
   EfficiencyModeEn          [Auto]
   Package Power Limit       [Auto]
   Control
   xGMI Link Width           [Auto]                   →←: Select Screen
   Control                                            ↑↓: Select Item
   APBDIS                    [1]                      Enter: Select
   DF Cstates                [Auto]                   +/-: Change Opt.
   Fixed SOC Pstate          [P0]                     F1: General Help
   CPPC                      [Auto]                   F2: Previous Values
   HSMP Support              [Auto]                   F3: Optimized Defaults
   Diagnostic Mode           [Auto]                   F4: Save & Exit
                                                      ESC: Exit

          Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                          AB
```

Preferred I/O and Enhanced Preferred I/O

Preferred I/O is a new capability in the EPYC 7002 series BIOS that prioritizes the traffic from the selected I/ O device and facilitates the ordering of PCIe packets which reduces the overhead and results in better adapter performance.

Enhanced Preferred I/O, further ensures that the same configured I/O device remains at the highest performance by keeping its clocks at the maximum frequency.

1. To access the **Preferred I/O** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **Preferred I/O** > **Manual**
2. To access the **Preferred I/O Bus** setting select **Advanced** > **AMD CBS** > **NBIO Common Options** > **Preferred I/O Bus** > **[PCIe Bus Number]**
3. To access the **Enhanced Preferred I/O** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **Enhanced Preferred I/O Mode** > **Auto/Enable P0**



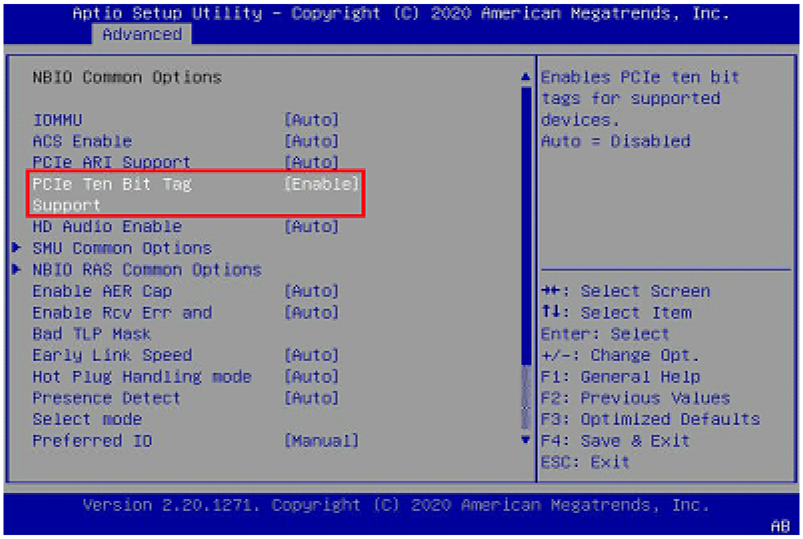
```
       Aptio Setup Utility - Copyright (C) 2020 American Megatrends, Inc.
          Advanced

   PCIe Ten Bit Tag          [Enable]          Enable/Disable Pcie
   Support                                     Loopback Mode
   HD Audio Enable           [Auto]
 ▶ SMU Common Options
 ▶ NBIO RAS Common Options
   Enable AER Cap            [Auto]
   Enable Rcv Err and        [Auto]
   Bad TLP Mask
   Early Link Speed          [Auto]
   Hot Plug Handling mode    [Auto]
   Presence Detect           [Auto]            →←: Select Screen
   Select mode                                 ↑↓: Select Item
   Preferred IO              [Manual]          Enter: Select
   Preferred IO Bus          1                 +/-: Change Opt.
   Enhanced Preferred IO     [Auto]            F1: General Help
   Mode                                        F2: Previous Values
   Loopback Mode             [Auto]            F3: Optimized Defaults
   CV test                   [Auto]            F4: Save & Exit
                                               ESC: Exit

        Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                      AB
```

PCIe Ten Bit Tag

Enable the **PCIe Ten Bit Tag** to increase the number of non posted requests from 256 to 768 for better performance. As latency increases, the increase in unique tags are required to maintain the peak performance at 16 GT/s.

To access the **PCIe Ten Bit Tag** setting, select **Advanced** > **AMD CBS** > **NBIO Common Options** > **PCIe Ten Bit Tag** > **Enable**

Figure 44: PCIe Ten Bit Tag Settings



```
       Aptio Setup Utility - Copyright (C) 2020 American Megatrends, Inc.
          Advanced

   NBIO Common Options                         Enables PCIe ten bit
                                               tags for supported
   IOMMU                     [Auto]            devices.
   ACS Enable                [Auto]            Auto = Disabled
   PCIe ARI Support          [Auto]
   PCIe Ten Bit Tag          [Enable]
   Support
   HD Audio Enable           [Auto]
 ▶ SMU Common Options
 ▶ NBIO RAS Common Options
   Enable AER Cap            [Auto]            →←: Select Screen
   Enable Rcv Err and        [Auto]            ↑↓: Select Item
   Bad TLP Mask                                Enter: Select
   Early Link Speed          [Auto]            +/-: Change Opt.
   Hot Plug Handling mode    [Auto]            F1: General Help
   Presence Detect           [Auto]            F2: Previous Values
   Select mode                                 F3: Optimized Defaults
   Preferred IO              [Manual]          F4: Save & Exit
                                               ESC: Exit

        Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                      AB
```

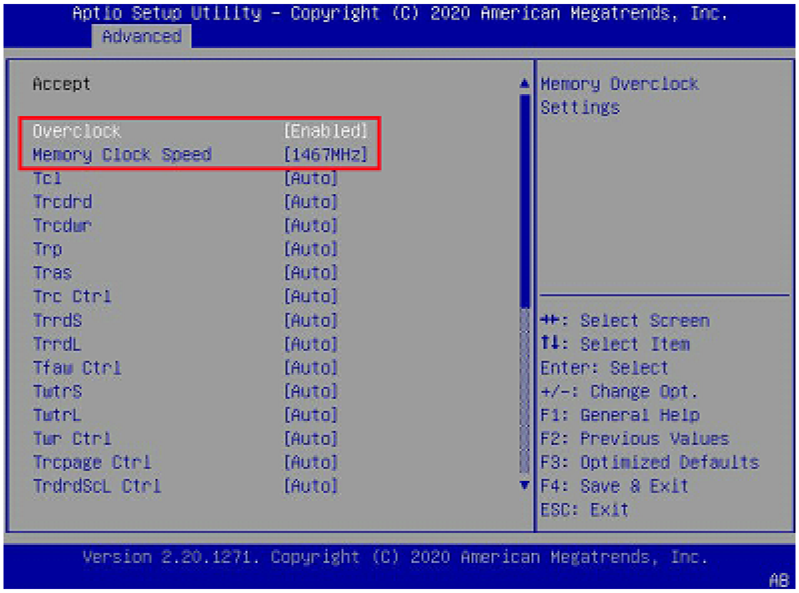
Memory Clock Speed

Set the **Memory Clock Speed** to match the maximum fabric clock speed supported by installed EPYC 7002 series server, which is either 1467 MHz or 1333 MHz (the double data rate is 2x this clock – for example, MCLK = 1467 means 2933 MTS data rate).

NOTE
A platform may be capable of supporting higher speed memory (for example, 1600 MHz memory clock) and while this may increase the overall platform memory bandwidth, the average memory latency is higher.

1. To access the **Overclock**setting, select **Advanced** > **AMD CBS** > **UMC Common Options** > **DDR4 Common Options** > **DRAM Timing Configuration** > **Accept** > **Overclock** > **Enabled**
2. To access the **Memory Clock** setting, select **Advanced** > **AMD CBS** > **UMC Common Options** > **DDR4 Common Options** > **DRAM Timing Configuration** > **Accept** > **Memory Clock Speed** > **1467MHz**

Figure 45: Memory Clock Speed Settings



```
Aptio Setup Utility - Copyright (C) 2020 American Megatrends, Inc.
Advanced

Accept                                      Memory Overclock
                                            Settings
Overclock                 [Enabled]
Memory Clock Speed        [1467MHz]
Tcl                       [Auto]
Trcdrd                    [Auto]
Trcdwr                    [Auto]
Trp                       [Auto]
Tras                      [Auto]
Trc Ctrl                  [Auto]
TrrdS                     [Auto]            ++: Select Screen
TrrdL                     [Auto]            ↑↓: Select Item
Tfaw Ctrl                 [Auto]            Enter: Select
TwtrS                     [Auto]            +/-: Change Opt.
TwtrL                     [Auto]            F1: General Help
Twr Ctrl                  [Auto]            F2: Previous Values
Trcpage Ctrl              [Auto]            F3: Optimized Defaults
TrdrdScL Ctrl             [Auto]            F4: Save & Exit
                                            ESC: Exit

Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                AB
```

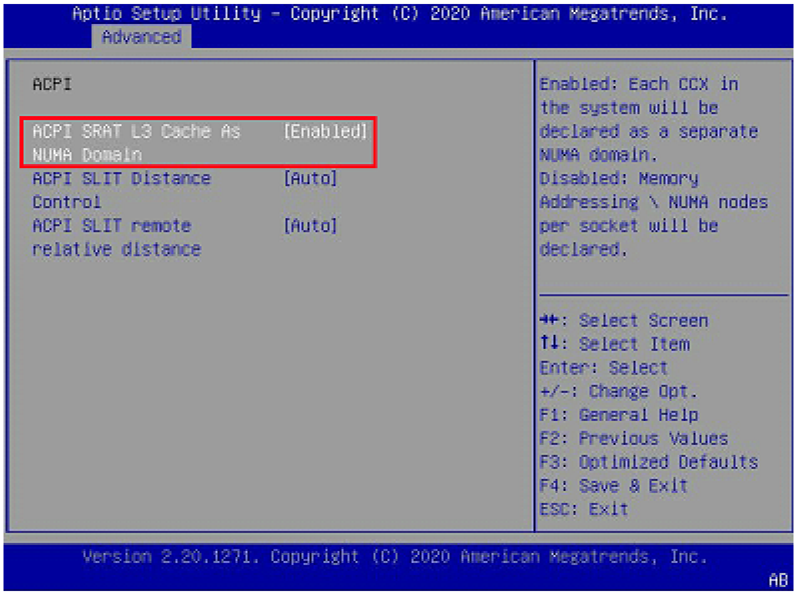
L3 LLC (Last Level Cache) as NUMA

Enable L3 as NUMA to create NUMA nodes equal to the number of L3 Caches (CCX). This helps the operating system schedulers maintain locality to the LLC without causing unnecessary cache-to-cache transactions and improves the performance.

NOTE
Currently this is a benchmarking feature meant for isolating L3 caches and is not recommended for production deployments.

To access the **ACPI** settings, select **Advanced** > **AMD CBS** > **DF Common Options** > **ACPI** > **ACPI SRAT L3 cache As NUMA Domain** > **Enabled**

Figure 46: ACPI SRAT L3 cache As NUMA Domain Setting



```
Aptio Setup Utility – Copyright (C) 2020 American Megatrends, Inc.
    Advanced

ACPI                                     Enabled: Each CCX in
                                         the system will be
ACPI SRAT L3 Cache As     [Enabled]      declared as a separate
NUMA Domain                              NUMA domain.
ACPI SLIT Distance        [Auto]         Disabled: Memory
Control                                  Addressing \ NUMA nodes
ACPI SLIT remote          [Auto]         per socket will be
relative distance                        declared.

                                         →←: Select Screen
                                         ↑↓: Select Item
                                         Enter: Select
                                         +/-: Change Opt.
                                         F1: General Help
                                         F2: Previous Values
                                         F3: Optimized Defaults
                                         F4: Save & Exit
                                         ESC: Exit

Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                              AB
```

Socket/Inter-Chip Global Memory Interconnect (xGMI)

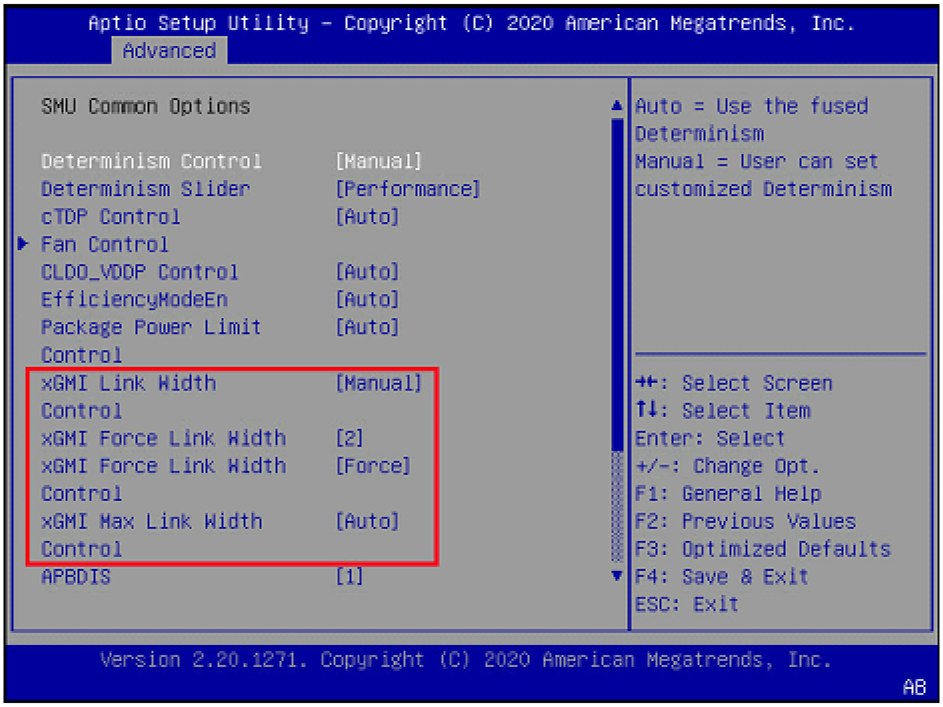
xGMI Dynamic Link Width Management saves power during periods of low socket-to-socket data traffic by reducing the number of active xGMI lanes per link from 16 to 8. However, under certain scenarios, involving low bandwidth, but latency-sensitive traffic, the transition from low power to full power xGMI can adversely impact latency.

Setting **xGMI Link Width Control** to **Manual** and specifying a **Max Link Width** of 16 forces the xGMI interface into full power mode, eliminating any latency jitter.

> **NOTE**
> Socket/Inter-Chip Global Memory Interconnect option only applies to a 2P system.

1. To access the **xGMI Link Width Control** setting, select **Advanced** > **AMD CBS** > **SMU Common Options** > **xGMI Link Width Control** > **Manual**
2. To access the **xGMI Force Link Width** setting, select **Advanced** > **AMD CBS** > **SMU Common Options** > **xGMI Force Link Width** > **2**
3. To access the **xGMI Force Link Width Control** setting, select **Advanced** > **AMD CBS** > **SMU Common Options** > **xGMI Force Link Width Control** > **Force**
4. To access the **xGMI Max Link Width Control** setting, select **Advanced** > **AMD CBS** > **SMU Common Options** > **xGMI Force Link Width Control** > **Auto**

Figure 47: Socket/Inter-Chip Global Memory Interconnect (xGMI) Settings



```
Aptio Setup Utility - Copyright (C) 2020 American Megatrends, Inc.
Advanced

SMU Common Options                              Auto = Use the fused
                                                Determinism
Determinism Control          [Manual]           Manual = User can set
Determinism Slider           [Performance]      customized Determinism
cTDP Control                 [Auto]
▶ Fan Control
CLDO_VDDP Control            [Auto]
EfficiencyModeEn             [Auto]
Package Power Limit          [Auto]
Control
xGMI Link Width              [Manual]           →←: Select Screen
Control                                         ↑↓: Select Item
xGMI Force Link Width        [2]                Enter: Select
xGMI Force Link Width        [Force]            +/-: Change Opt.
Control                                         F1: General Help
xGMI Max Link Width          [Auto]             F2: Previous Values
Control                                         F3: Optimized Defaults
APBDIS                       [1]                F4: Save & Exit
                                                ESC: Exit

Version 2.20.1271. Copyright (C) 2020 American Megatrends, Inc.
                                                                 AB
```

Applications that are known to be insensitive to both socket-to-socket bandwidth and latency can set a fixed link width of eight to save power, which can divert more power to the cores for boost.

TCP Performance Tuning

This section provides the following TCP performance tuning information for the Broadcom Ethernet NIC controller:

- OS Tuning (Linux)
- DPDK Tunings
- Configure IRQ and Application Affinity
- Configure RSS for Performance in Linux
- Configure RSS for Performance in Windows
- Performance Tuning
- IP Forwarding Tunings

OS Tuning (Linux)

This section contains OS tuning information for Linux.

IOMMU

IOMMU

IOMMUIt is recommended to use IOMMU in pass through (pt) mode. This disables the DMAR (DMA Remapping) to the memory and improves the host performance. It can be enabled by adding *iommu=pt* in the grub file as shown in the following commands:

```
vi /etc/default/grub
GRUB_CMDLINE_LINUX="nofb splash=quiet console=tty0 ... iommu=pt"
grub2-mkconfig -o /boot/grub2/grub.cfg
```

Reboot the system and ensure that *iommu=pt* is in /proc/cmdline using the following command:

```
cat /proc/cmdline | grep -i iommu=pt
```

Performance Governor

The CPU frequency performance governor sets the CPU statically to the highest frequency within the borders of scaling_min_freq and scaling_max_freq for highest performance.

```
echo performance | sudo tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
```

Check that the CPUs are running at highest frequency using the following command:

```
cat /proc/cpuinfo | grep -i mhz
```

TCP Memory Configuration

Increase the memory buffer for TCP sockets. This can improve performance for long RTT connections by allowing more data in flight at a time where smaller buffers may not cover the BDP (Bandwidth-delay product) resulting in gaps in transmission. The following three values in the echo statement represent the minimum, default and maximum buffer value for each TCP socket.

```
echo "4096 131072 268435456" > /proc/sys/net/ipv4/tcp_rmem
echo "4096 131072 63108864" > /proc/sys/net/ipv4/tcp_wmem
```

The TCP rmem_max and wmem_max are maximum receive and send buffer sizes for the socket memory. These buffers are used to hold/send the data until it is read by the application.

```
echo 268435456 > /proc/sys/net/core/rmem_max
echo 63108864 > /proc/sys/net/core/wmem_max
```

nohz=off

This is a boot time kernel parameter that disables the dyntick idle mode. In this mode, the kernel sends the timer tick periodically to all CPUs irrespective of the state and prevents the CPU to remain idle for a long time which results in increased power consumption.

> **NOTE**
> This configuration must be tested extensively as results may vary depending upon the workload and applications.

Refer to kernel documentation for more detail.

```
vi /etc/default/grub
GRUB_CMDLINE_LINUX="nofb splash=quiet console=tty0 ... nohz=off"
grub2-mkconfig -o /boot/grub2/grub.cfg
```

TCP Example with the BCM957508-P2100G

This section provides the BCM957508-P2100G configuration for running bi-directional dual port test with netperf for 2 queues and 2 sessions test.

NOTE: The commands below are for reference and may not be a complete set of commands due to brevity.

BIOS Settings:

```
Configure NPS=1 Enable X2APIC
Performance Determinism Slider Configure APBDIS = 1
Configure Preferred IO and Enhanced Preferred IO Enable PCIe Ten Bit Tag
Configure Memory Clock Speed Enable L3 as NUMA
Configure xGMI
```

Adapter Settings:

ethtool	-L	enp65s0f0	combined 4	
ethtool	-C	enp65s0f0	adaptive-rx on rx-usecs 50 rx-frames	50
ethtool	-C	enp65s0f0	adaptive-tx on tx-usecs 50 tx-frames	50
ethtool	-G	enp65s0f0	rx 2047 tx 2047	
ethtool	-K	enp65s0f0	ntuple on	
ethtool	-K	enp65s0f0	tx-nocache-copy on	
ethtool	-K	enp65s0f0	rx-gro-hw on lro off gro on	
ethtool	-L	enp65s0f1	combined 4	
ethtool	-C	enp65s0f1	adaptive-rx on rx-usecs 50 rx-frames	50
ethtool	-C	enp65s0f1	adaptive-tx on tx-usecs 50 tx-frames	50
ethtool	-G	enp65s0f1	rx 2047 tx 2047	
ethtool	-K	enp65s0f1	ntuple on	
ethtool	-K	enp65s0f1	tx-nocache-copy on	
ethtool	-K	enp65s0f1	rx-gro-hw on lro off gro on	

bnxtnvm -dev=enp65s0f0 setoption=pcie_relaxed_ordering enable bnxtnvm -dev=enp65s0f1 setoption=pcie_relaxed_ordering enable

OS Settings:

```
echo 268435456 > /proc/sys/net/core/rmem_max echo 67108864 > /proc/sys/net/core/wmem_max
echo "4096 131072 268435456" > /proc/sys/net/ipv4/tcp_rmem
        echo "4096 131072 63108864" > /proc/sys/net/ipv4/tcp_wmemv
```

Configure Affinities:

```
IRQ list for Port0: '160', '161'
IRQ list for Port1: 163, 164
```

CPU Cores to be Assigned – Port 0: 0, 1 Port 1: 16,17:

```
echo 0 > /proc/160/smp_affinity_list
echo 1 > /proc/161/smp_affinity_list
echo 16 > /proc/163/smp_affinity_list
echo 17 > /proc/164/smp_affinity_list
```

XRFS and XPS Configure:

echo	32768	> /proc/sys/net/core/rps_sock_flow_entries
echo	16384	> /sys/class/net/p1p1/queues/rx-0/rps_flow_cnt
echo	16384	> /sys/class/net/p1p1/queues/rx-1/rps_flow_cnt

XPS setting on Port0:

```
echo 1 > /sys/class/net/p1p1/queues/tx-0/xps_cpus echo 2 > /sys/class/net/p1p1/queues/tx-1/xps_cpus
```

XPS setting on Port1:

```
echo 10000 > /sys/class/net/p1p1/queues/tx-0/xps_cpus
echo 20000 > /sys/class/net/p1p1/queues/tx-1/xps_cpus
```

Netserver Commands:

```
netserver -L 25.0.0.1 -p 12800
netserver -L 25.0.0.1 -p 12801
netserver -L 35.0.0.1 -p 13051
netserver -L 35.0.0.1 -p 13052
```

Netperf Commands:

netperf	-H 25.0.0.2	-t TCP_SENDFILE -T 0,0 -l 60 -L 25.0.0.1 -P 0 -p 12800,32100 -- -S '512K' -s
'512K'	-m '64k' -P	'12900,32200' &
netperf	-H 25.0.0.2	-t TCP_SENDFILE -T 1,1 -l 60 -L 25.0.0.1 -P 0 -p 12801,32101 -- -S '512K' -s
'512K'	-m '64k' -P	'12903,32211' &

netperf	-H 35.0.0.1	-t TCP_SENDFILE -T	16,16	-l	60	-L	35.0.0.2	-P	0	-p	13051,32351	--	-S	'512K'	-s
'512K'	-m '64k' -P	'13151,32451' &													
netperf	-H 35.0.0.1	-t TCP_SENDFILE -T	17,17	-l	60	-L	35.0.0.2	-P	0	-p	13052,32352	--	-S	'512K'	-s
'512K'	-m '64k' -P	'13154,32462' &													

Table 44: Server Configuration

Item	Description
Test	Linux Bi-Directional TCP, 2x 100 Gb/s
Server	AMD EPYC 7002 Series Reference Platform
CPU	AMD EPYC 7742
RAM	128 GB, 16 GB x 8 DIMMs at 3200 MHz
BIOS	REX1006G
NIC	BCM957508-P2100G
Operating System	Linux RHEL 7.6
Kernel	5.3.4
Kernel command line	BOOT_IMAGE=/vmlinuz-5.3.4 root=/dev/mapper/rhel-root ro crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap rhgb quiet iommu=pt nohz=off

Table 45: Throughput Numbers

Speed	MTU	Bidi Throughput
2x100G	1500	339 Gb/s

DPDK Tunings

Broadcom publishes the DPDK performance report to dpdk.org, this report contains the achieved performance numbers and configuration details. The latest version of the report can be accessed here.

This section provides the following information on DPDK tunings:

- BIOS Tuning
- Kernel Tuning
- PCIe Configuration
- DPDK Configuration
- DPDK Results

BIOS Tuning

See BIOS Tuning and set the following BIOS options:

- Local APIC Mode – x2 APIC
- NUMA nodes per socket – NPS1
- L3 Cache as NUMA – Disabled
- Memory Clock Speed – 1467
- PCIe Ten Bit Tag – Enabled
- Preferred I/O – Manual
- Preferred I/O BUS – (Provide BUS ID)
- Enhanced Preferred I/O – Auto
- Determinism Control – Manual
- Determinism Slider – Performance
- xGMI Link Width Control – Manual
- xGMI Force Link Width – 2
- xGMI Force Link Width Control – Force
- xGMI Max Link Width Control – Auto
- APBDIS 1 and PState=P0
- SMT Control – Enabled

Kernel Tuning

Add the following entries to the kernel command line:

```
amd_iommu=on iommu=pt nohz=off rcu_nocbs=32-47 isolcpus=32-47 selinux=0 numa_balancing=disable
processor.max_cstate=0 nosoftlockup default_hugepagesz=1G hugepagesz=1G hugepages=64
```

PCIe Configuration

Reduce MRRS to 1024B using the following command (Use default setting. For example, 4K for a 4 x 25 adapter):

```
setpci -s 41:00.0 b4.w=3d57
```

NOTE
See PCIe MRRS (Maximum Read Request Size) as incorrect usage may cause a system crash.

Enable **Relaxed** ordering in adapter.

Example: bnxtnvm -dev=[interface] setoption=pcie_relaxed_ordering enable

DPDK Configuration

This section provides information on DPDK configuration. Driver: vfio-pci

Testpmd Command Line

```
testpmd -l 32,33,34,35,36,37,38,39,63 -n 4 --socket-
mem=4096 --master-lcore 63 -- --txq=8 --rxq=8 --rxd=4096 --
txd=4096 --nb-cores=8 -i
```

DPDK Results

- http://core.dpdk.org/perf-reports/
- BCM5741X (25G) Results:
  - Forwarding Rate is 30 Mp/s using 64B frame
  - Line-Rate from 128B onwards
- BCM575XX (200G) Results:
  - Forwarding Rate is 102 Mp/s using 64B frame
  - Line-Rate with 1518B

BIOS Tuning

See BIOS Tuning and set the following BIOS options:

- Local APIC Mode – x2 APIC
- NUMA nodes per socket – NPS1
- L3 Cache as NUMA – Disabled
- Memory Clock Speed – 1467
- PCIe Ten Bit Tag – Enabled
- Preferred I/O – Manual
- Preferred I/O BUS – (Provide BUS ID)
- Enhanced Preferred I/O – Auto
- Determinism Control – Manual
- Determinism Slider – Performance
- xGMI Link Width Control – Manual
- xGMI Force Link Width – 2
- xGMI Force Link Width Control – Force
- xGMI Max Link Width Control – Auto
- APBDIS 1 and PState=P0
- SMT Control – Enabled

Kernel Tuning

Add the following entries to the kernel command line:

```
amd_iommu=on iommu=pt nohz=off rcu_nocbs=32-47 isolcpus=32-47 selinux=0 numa_balancing=disable
processor.max_cstate=0 nosoftlockup default_hugepagesz=1G hugepagesz=1G hugepages=64
```

PCIe Configuration

Reduce MRRS to 1024B using the following command (Use default setting. For example, 4K for a 4 x 25 adapter):

```
setpci -s 41:00.0 b4.w=3d57
```

NOTE
See PCIe MRRS (Maximum Read Request Size) as incorrect usage may cause a system crash.

Enable **Relaxed** ordering in adapter.

Example: bnxtnvm -dev=[interface] setoption=pcie_relaxed_ordering enable

DPDK Configuration

This section provides information on DPDK configuration. Driver: vfio-pci

Testpmd Command Line

```
testpmd -l 32,33,34,35,36,37,38,39,63 -n 4 --socket-
mem=4096 --master-lcore 63 -- --txq=8 --rxq=8 --rxd=4096 --
txd=4096 --nb-cores=8 -i
```

DPDK Results

- http://core.dpdk.org/perf-reports/
- BCM5741X (25G) Results:
  - Forwarding Rate is 30 Mp/s using 64B frame
  - Line-Rate from 128B onwards
- BCM575XX (200G) Results:
  - Forwarding Rate is 102 Mp/s using 64B frame
  - Line-Rate with 1518B

Configure IRQ and Application Affinity

IRQ affinity refers to the binding of interrupts from a specific device to one or multiple logical processors. The distribution of the IRQs across different local logical cores results in improved performance due to better CPU utilization.

Use the following steps for IRQ affinity configuration:

1. Disable irqbalance (to prevent the service from dynamically moving your IRQ) using the following commands:
   ```
   service irqbalance stop
   service irqbalance disable (to keep it persistent through reboot)
   ```
2. Identify local CPUs using the following command:
   ```
   cat /sys/class/net/[interface]/device/local_cpulist
   ```
3. Identify IRQ numbers using the following command:
   ```
   cat /proc/interrupts | grep [interface] | awk -F ":" '{print $1}'
   ```
4. Pin each of the interrupts to a different local NUMA CPUs using the following command:

```
echo [cpu_core] > /proc/irq/[interface number]/smp_affinity_list
```

> **NOTE**
> It is preferred to use the same CPUs for application affinity which also allows cache locality between interrupts and application threads and reduces the processing overhead. *taskset* and *numactl* tools or application-specific options (for example, netperf with -T) can be used for configuring application locality:

```
taskset -c [cpu_core list] application
```

or

```
numactl -C [cpu_cores list] application
```

or

application-specific options, for example:

If using netperf there is a -T option to handle both server and client application affinity.

Configure RSS for Performance in Linux

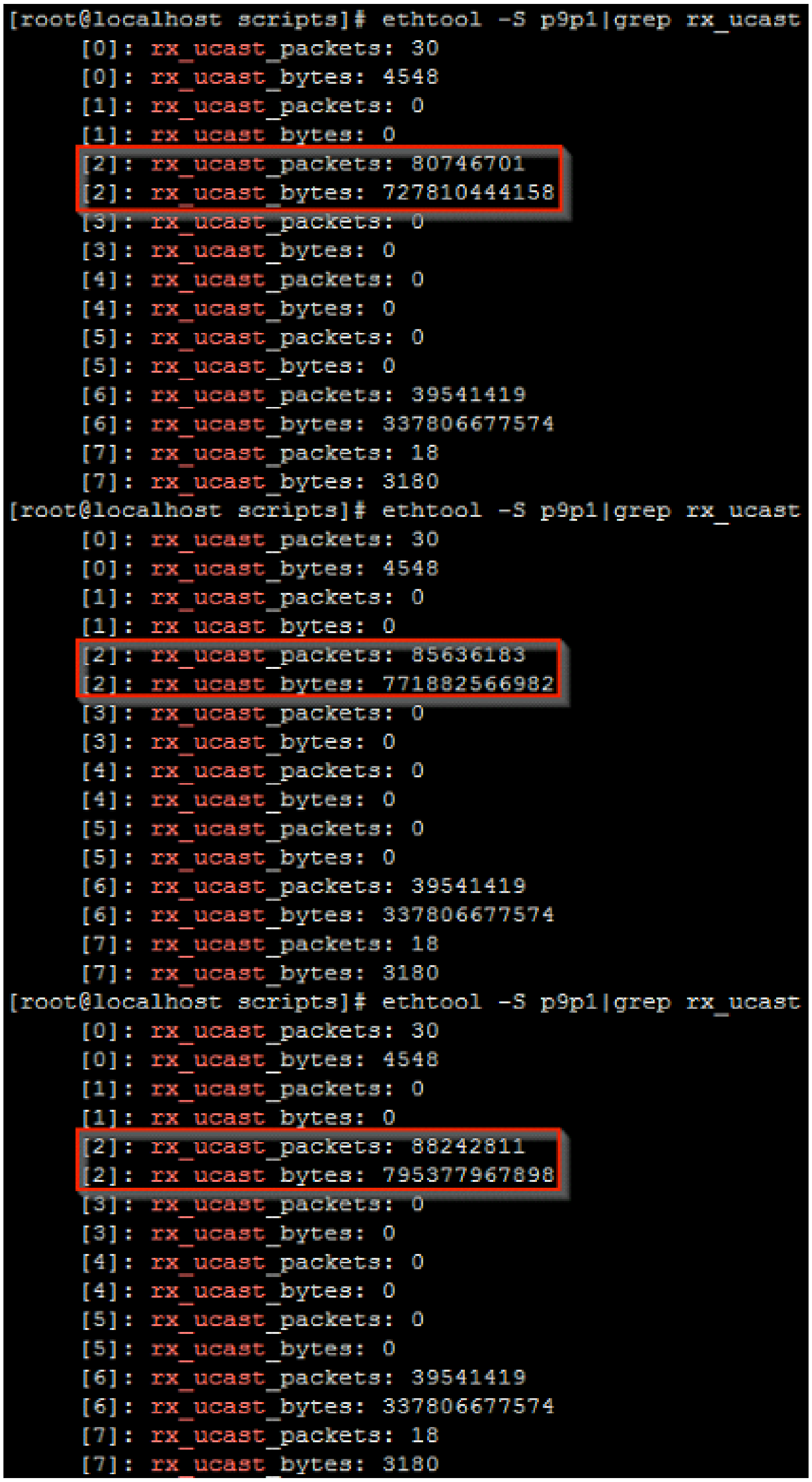
This section provides the steps required to configure NetPerf sessions to fully utilize the receive queues in Linux. The following command is used as an example for the rest of this section:

```
./usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,35 -p 12800,32100 -P 0 -- -m 64K
-s 512K -S 512K -P 12900,32200
```

The test-specific, after --, parameter -P in the NetPerf command controls the source and destination ports of the data traffic. To properly steer the traffic, modify these values using the following procedure:

1. Run the NetPerf command. If 120 seconds creates too much overhead to wait for completion, modify the -l option since sessions are only steered and not recording data in this step.
   ```
   ./usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,35 -p 12800,32100 -P 0 -- -m 64K -s
    512K -S 512K -P 12900,32200
   ```
2. While the command is running, monitor the receive statistics on the server side using ethtool -S <interface>.

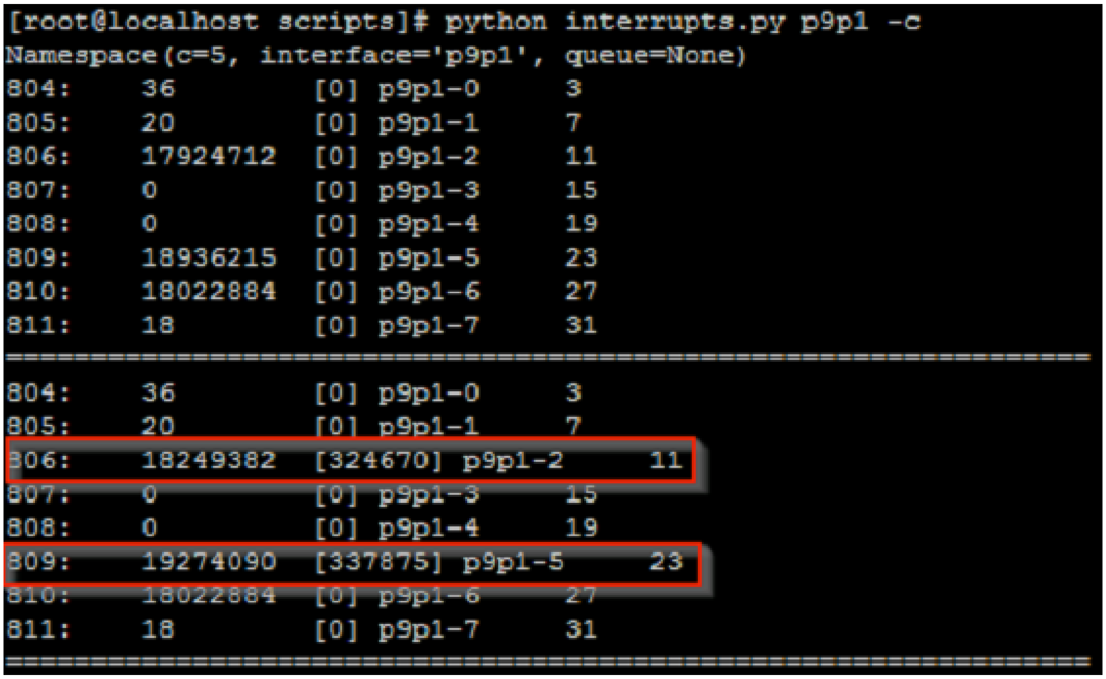
   The statistics do not reset until there is a driver reload. This means that many queues may have received packets, however, only one is changing since only a single session of traffic is running.



```
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 30
     [0]: rx_ucast_bytes: 4548
     [1]: rx_ucast_packets: 0
     [1]: rx_ucast_bytes: 0
     [2]: rx_ucast_packets: 80746701
     [2]: rx_ucast_bytes: 727810444158
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 0
     [4]: rx_ucast_bytes: 0
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 18
     [7]: rx_ucast_bytes: 3180
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 30
     [0]: rx_ucast_bytes: 4548
     [1]: rx_ucast_packets: 0
     [1]: rx_ucast_bytes: 0
     [2]: rx_ucast_packets: 85636183
     [2]: rx_ucast_bytes: 771882566982
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 0
     [4]: rx_ucast_bytes: 0
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 18
     [7]: rx_ucast_bytes: 3180
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 30
     [0]: rx_ucast_bytes: 4548
     [1]: rx_ucast_packets: 0
     [1]: rx_ucast_bytes: 0
     [2]: rx_ucast_packets: 88242811
     [2]: rx_ucast_bytes: 795377967898
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 0
     [4]: rx_ucast_bytes: 0
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 18
     [7]: rx_ucast_bytes: 3180
```

Ethtool displays the session created using the NetPerf command in the previous step goes to queue 2.

3. Confirm this result by monitoring the interrupts using interrupts.py.



```
[root@localhost scripts]# python interrupts.py p9p1 -c
Namespace(c=5, interface='p9p1', queue=None)
804:     36          [0] p9p1-0     3
805:     20          [0] p9p1-1     7
806:     17924712    [0] p9p1-2     11
807:     0           [0] p9p1-3     15
808:     0           [0] p9p1-4     19
809:     18936215    [0] p9p1-5     23
810:     18022884    [0] p9p1-6     27
811:     18          [0] p9p1-7     31
================================================================
804:     36          [0] p9p1-0     3
805:     20          [0] p9p1-1     7
806:     18249382    [324670] p9p1-2     11
807:     0           [0] p9p1-3     15
808:     0           [0] p9p1-4     19
809:     19274090    [337875] p9p1-5     23
810:     18022884    [0] p9p1-6     27
811:     18          [0] p9p1-7     31
================================================================
```

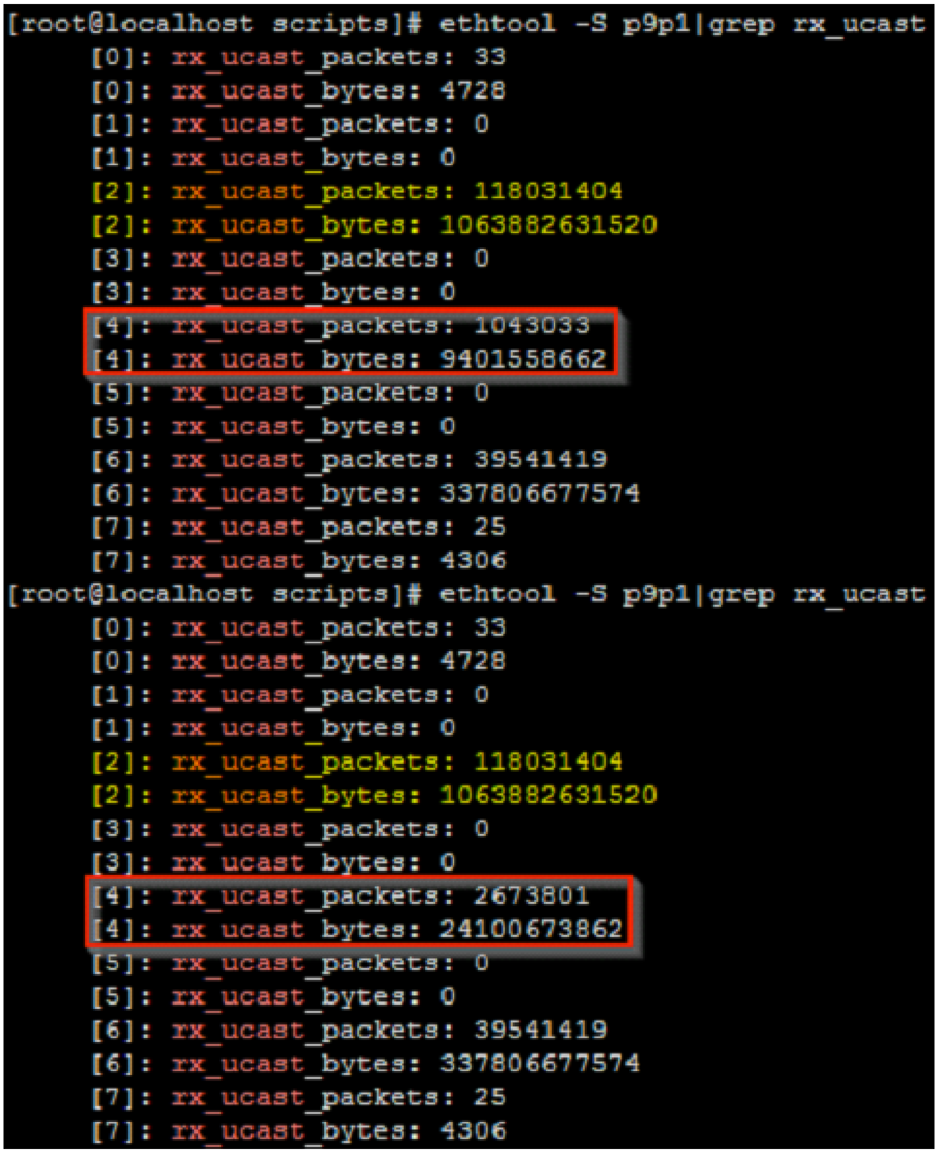
Two queues are receiving IRQs. In the Ethtool, only one queue is receiving packets, therefore queue 5 must be transmitting the acks. Which queue is used for transmits cannot be controlled.

4. Record the queue that is being utilized for receive.
5. Change the values in the -P option (for example, increase each port by one) with the following command:

```
usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,35 -p 12800,32100 -P 0 -- -m 64K
-s 512K -S 512K -P 12901,32201
```

6. While it is running, monitor the receive statistics on the server side using ethtool -S <interface>.

   The statistics do not reset until there is a driver reload. This means that many queues may have received packets, however, only one is changing since only a single session of traffic is running.

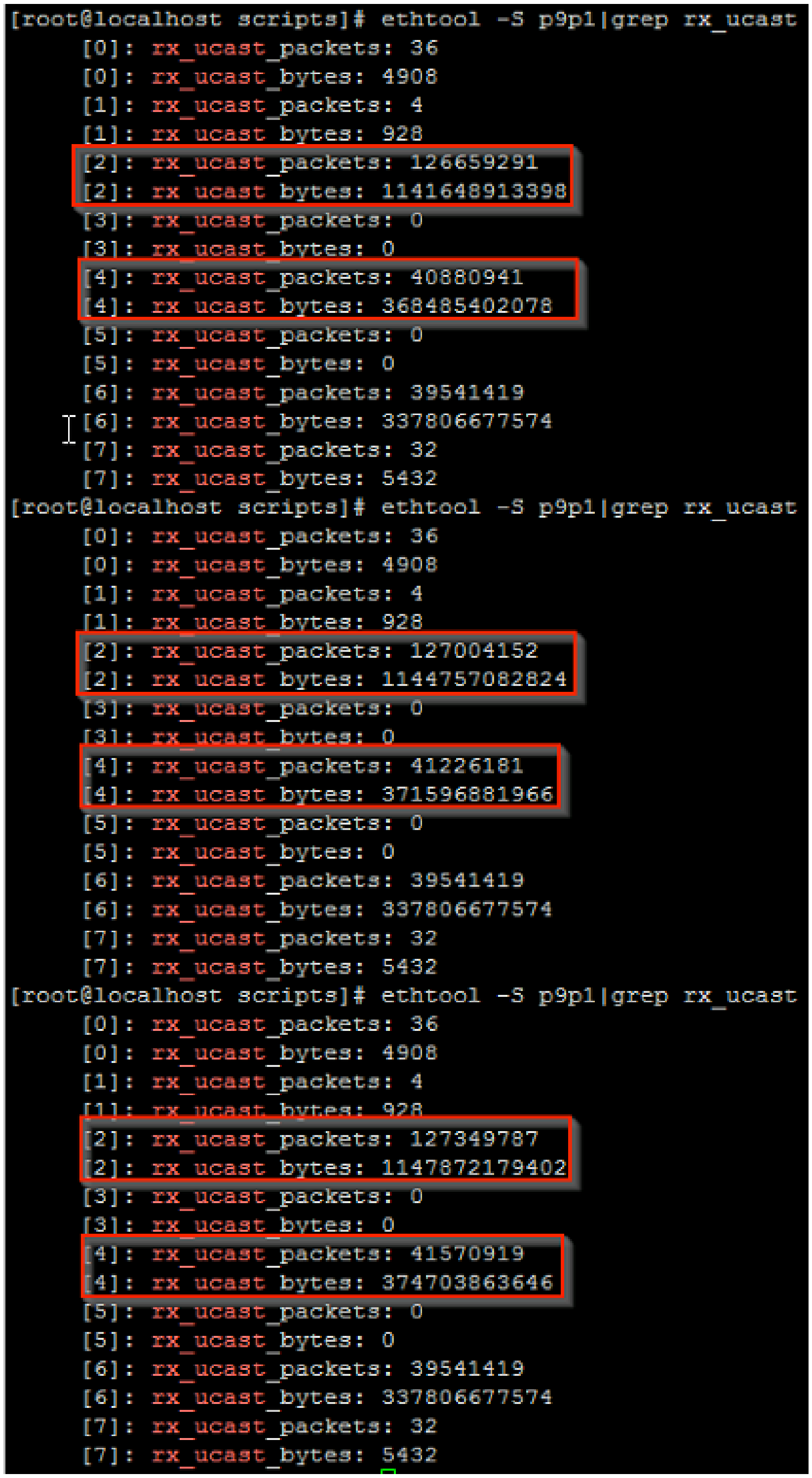


```
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 33
     [0]: rx_ucast_bytes: 4728
     [1]: rx_ucast_packets: 0
     [1]: rx_ucast_bytes: 0
     [2]: rx_ucast_packets: 118031404
     [2]: rx_ucast_bytes: 1063882631520
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 1043033
     [4]: rx_ucast_bytes: 9401558662
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 25
     [7]: rx_ucast_bytes: 4306
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 33
     [0]: rx_ucast_bytes: 4728
     [1]: rx_ucast_packets: 0
     [1]: rx_ucast_bytes: 0
     [2]: rx_ucast_packets: 118031404
     [2]: rx_ucast_bytes: 1063882631520
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 2673801
     [4]: rx_ucast_bytes: 24100673862
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 25
     [7]: rx_ucast_bytes: 4306
```

Queue 4 is now being used and queue 2 is no longer being used.

7. If the queue being used is the same as a previous session, this session is not good and must be tried again. If it is using a different queue, then record this value.
8. Repeat these steps until the required number of sessions is achieved.

If both sessions are run at the same time, see below.



```
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 36
     [0]: rx_ucast_bytes: 4908
     [1]: rx_ucast_packets: 4
     [1]: rx_ucast_bytes: 928
     [2]: rx_ucast_packets: 126659291
     [2]: rx_ucast_bytes: 1141648913398
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 40880941
     [4]: rx_ucast_bytes: 368485402078
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 32
     [7]: rx_ucast_bytes: 5432
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 36
     [0]: rx_ucast_bytes: 4908
     [1]: rx_ucast_packets: 4
     [1]: rx_ucast_bytes: 928
     [2]: rx_ucast_packets: 127004152
     [2]: rx_ucast_bytes: 1144757082824
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 41226181
     [4]: rx_ucast_bytes: 371596881966
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 32
     [7]: rx_ucast_bytes: 5432
[root@localhost scripts]# ethtool -S p9p1|grep rx_ucast
     [0]: rx_ucast_packets: 36
     [0]: rx_ucast_bytes: 4908
     [1]: rx_ucast_packets: 4
     [1]: rx_ucast_bytes: 928
     [2]: rx_ucast_packets: 127349787
     [2]: rx_ucast_bytes: 1147872179402
     [3]: rx_ucast_packets: 0
     [3]: rx_ucast_bytes: 0
     [4]: rx_ucast_packets: 41570919
     [4]: rx_ucast_bytes: 374703863646
     [5]: rx_ucast_packets: 0
     [5]: rx_ucast_bytes: 0
     [6]: rx_ucast_packets: 39541419
     [6]: rx_ucast_bytes: 337806677574
     [7]: rx_ucast_packets: 32
     [7]: rx_ucast_bytes: 5432
```

Configure RSS for Performance in Windows

When running single-ended traffic ensure that all cores are being utilized equally, resulting in the best performance. One way to achieve this is to ensure all traffic is spread across all available queues using RSS.

For an introduction to RSS, see Microsoft's site: https://msdn.microsoft.com/en-us/windows/hardware/drivers/network/introduction-to-receive-side-scaling

This section describes how to monitor the receive queues using Windows PowerShell and how to ensure that all queues are utilized. PowerShell is used so that this methodology is easily automated. If a graphical interface is preferred, Performance Monitor displays the same results. For this example, NetPerf is used.

This section provides the following information on RSS for Performance in Windows:

- Setting Up RSS
- Monitoring Receive Queues/Cores
- Expanding to New Counters

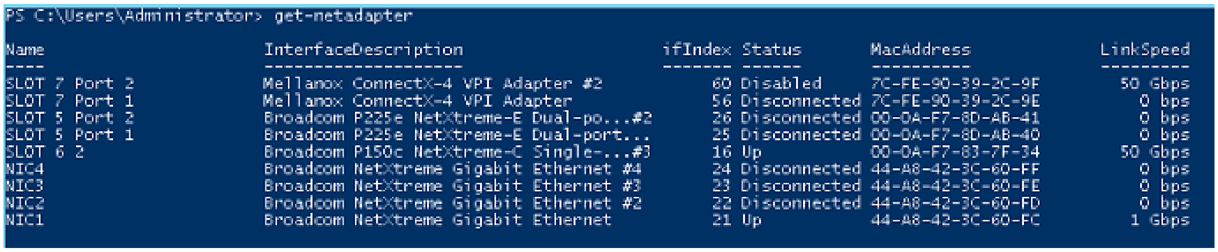
Setting Up RSS

Set up the adapter so that the proper cores are utilized using the following steps:

> **NOTE**
> This process is dependent on the target test environment. The purpose of this section is to provide steps for manipulating the RSS settings. The actual value settings used in the examples are not necessarily the ideal settings for achieving optimal performance.

1. To show the available adapters and their names used in subsequent steps, use the following command:

```
Get-Netadapter
```

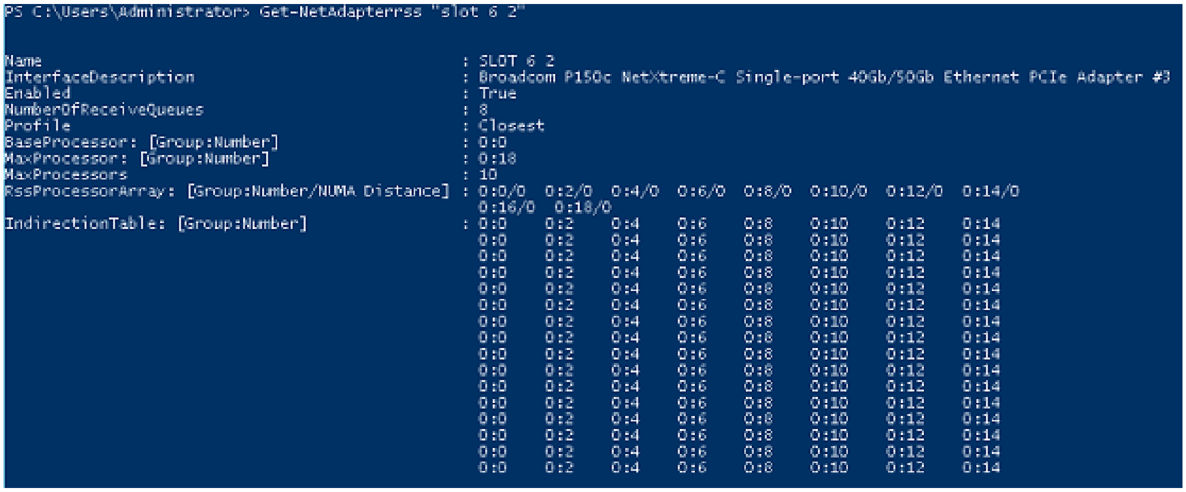


```
PS C:\Users\Administrator> get-netadapter

Name                      InterfaceDescription                    ifIndex Status       MacAddress             LinkSpeed
----                      --------------------                    ------- ------       ----------             ---------
SLOT 7 Port 2             Mellanox ConnectX-4 VPI Adapter #2           60 Disabled     7C-FE-90-39-2C-9F         50 Gbps
SLOT 7 Port 1             Mellanox ConnectX-4 VPI Adapter              56 Disconnected 7C-FE-90-39-2C-9E          0 bps
SLOT 5 Port 2             Broadcom P225e NetXtreme-E Dual-po...#2      26 Disconnected 00-0A-F7-8D-AB-41          0 bps
SLOT 5 Port 1             Broadcom P225e NetXtreme-E Dual-port...      25 Disconnected 00-0A-F7-8D-AB-40          0 bps
SLOT 6 2                  Broadcom P150c NetXtreme-C Single-...#3      16 Up           00-0A-F7-83-7F-34         50 Gbps
NIC4                      Broadcom NetXtreme Gigabit Ethernet #4       24 Disconnected 44-A8-42-3C-60-FF          0 bps
NIC3                      Broadcom NetXtreme Gigabit Ethernet #3       23 Disconnected 44-A8-42-3C-60-FE          0 bps
NIC2                      Broadcom NetXtreme Gigabit Ethernet #2       22 Disconnected 44-A8-42-3C-60-FD          0 bps
NIC1                      Broadcom NetXtreme Gigabit Ethernet          21 Up           44-A8-42-3C-60-FC          1 Gbps
```

For this example, Slot 6 2 is used, which indicates Broadcom single-port 50 Gb/s adapter.

2. To return all available RSS properties, use the following command:

```
Get-NetAdapterRss slot 6 2.
```



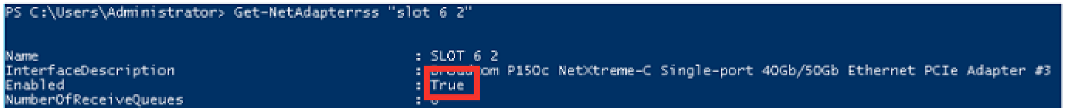
```
PS C:\Users\Administrator> Get-NetAdapterrss "slot 6 2"


Name                                            : SLOT 6 2
InterfaceDescription                            : Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
Enabled                                         : True
NumberOfReceiveQueues                           : 8
Profile                                         : Closest
BaseProcessor: [Group:Number]                   : 0:0
MaxProcessor: [Group:Number]                    : 0:18
MaxProcessors                                   : 10
RssProcessorArray: [Group:Number/NUMA Distance] : 0:0/0  0:2/0  0:4/0  0:6/0  0:8/0  0:10/0  0:12/0  0:14/0
                                                  0:16/0  0:18/0
IndirectionTable: [Group:Number]                : 0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
                                                  0:0    0:2    0:4    0:6    0:8    0:10     0:12     0:14
```

The indirection table is what must be manipulated. The indirection table indicates where the interrupt is processed for each hash value. If the hash returns 0, then processor 0 is used. If the hash returns 1, then processor 2 is used, and so forth.

> **NOTE**
> Windows displays even-numbered processors. This is because hyperthreading is enabled. Windows does not use both cores in a hyperthread pair. Therefore, only physical cores are displayed. On systems with hyperthreading off, the physical cores are sequentially numbered (for example, 0, 1, 2, 3).

3. Ensure that the Virtual Receive-Side scaling is enabled with the following command:

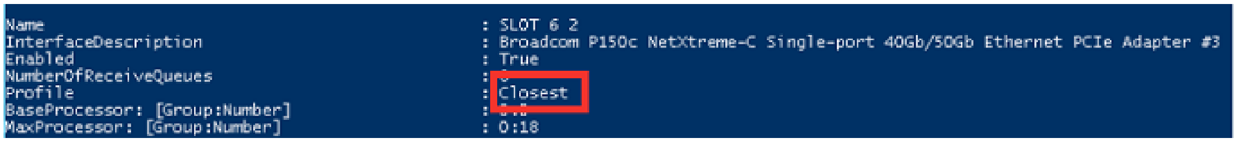
```
Set-NetadapterRss "slot 6 2" -Enabled $False
```



```
PS C:\Users\Administrator> Get-NetAdapterrss "slot 6 2"


Name                                            : SLOT 6 2
InterfaceDescription                            : Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
Enabled                                         : True
NumberOfReceiveQueues                           : [illegible]
```

4. Ensure that the Profile is set to **Closest** with the following command:

```
Set-NetadapterRss "slot 6 2" -Profile "Closest"
```



```
Name                                            : SLOT 6 2
InterfaceDescription                            : Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
Enabled                                         : True
NumberOfReceiveQueues                           : [illegible]
Profile                                         : Closest
BaseProcessor: [Group:Number]                   : [illegible]
MaxProcessor: [Group:Number]                    : 0:18
```

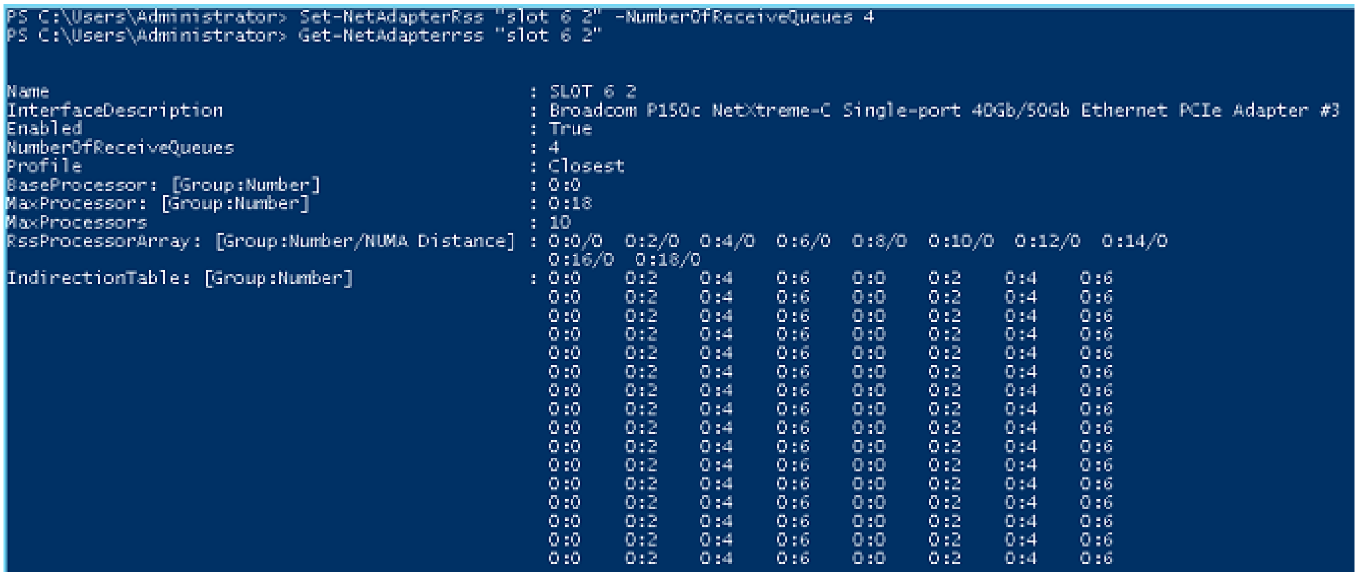
Profiles

RSS profiles determine how CPUs are assigned to the NIC. There are five profiles to choose from and it affects how the indirection table is populated. The following are all currently available RSS profiles. For some profiles, the OS can dynamically load balance. If the OS is load balancing, ensure that the RssProcessorArray only shows core that is intended for RSS use. To change this list, use BaseProcessor, MaxProcessor, MaxProcessors to limit the list.

- Closest – Logical processor numbers that are near the network adapter's base RSS processor are preferred. With this profile, the operating system rebalances logical processors dynamically based on load.
- ClosestStatic – Logical processor numbers near the network adapter's base RSS processor are preferred. With this profile, the operating system does not rebalance logical processors dynamically based on load.
- NUMA – Logical processor numbers are generally selected on different NUMA nodes to distribute the load. With this profile, the operating system rebalances logical processors dynamically based on load.
- NUMAStatic – This is the default profile. Logical processor numbers are generally selected on different NUMA nodes to distribute the load. With this profile, the operating system does not rebalance logical processors dynamically based on load.
- Conservative – RSS uses as few processors as possible to sustain the load. This option helps reduce the number of interrupts.

1. To set the number of receive queues, use the following command:

```
Set-NetAdapterRss "slot 6 2" -NumberOfReceiveQueues 4
```



```
PS C:\Users\Administrator> Set-NetAdapterRss "slot 6 2" -NumberOfReceiveQueues 4
PS C:\Users\Administrator> Get-NetAdapterrss "slot 6 2"


Name                                            : SLOT 6 2
InterfaceDescription                            : Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
Enabled                                         : True
NumberOfReceiveQueues                           : 4
Profile                                         : Closest
BaseProcessor: [Group:Number]                   : 0:0
MaxProcessor: [Group:Number]                    : 0:18
MaxProcessors                                   : 10
RssProcessorArray: [Group:Number/NUMA Distance] : 0:0/0  0:2/0  0:4/0  0:6/0  0:8/0  0:10/0  0:12/0  0:14/0
                                                  0:16/0  0:18/0
IndirectionTable: [Group:Number]                : 0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
                                                  0:0    0:2    0:4    0:6    0:0    0:2     0:4     0:6
```

NOTE

The indirection table has changed. There are only four queues instead of eight, therefore, only four cores are being used.

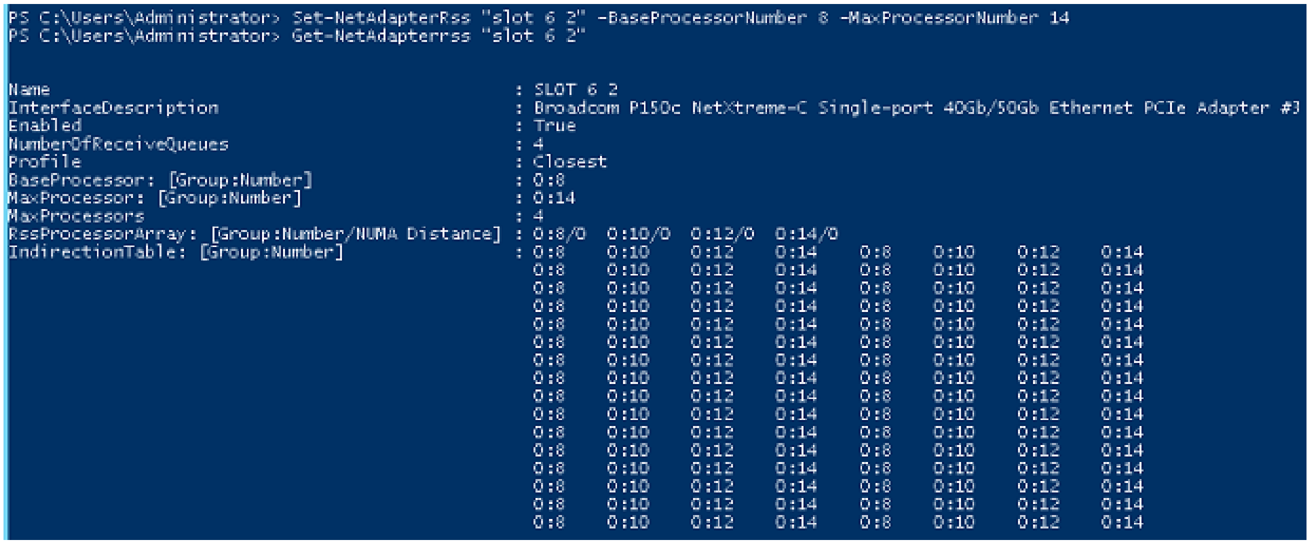
2. To set the base processor and max processor, use the following command:

```
Set-NetAdapterRss "slot 6 2" -BaseProcessorNumber 8 -MaxProcessorNumber 14
```

This effectively changes the adapters RSS processor group, RssProcessorArray.

NOTE

The current assumption is that the processors are in group 0. To change to group 1 use BaseProcessorGroup and MaxProcessorGroup.



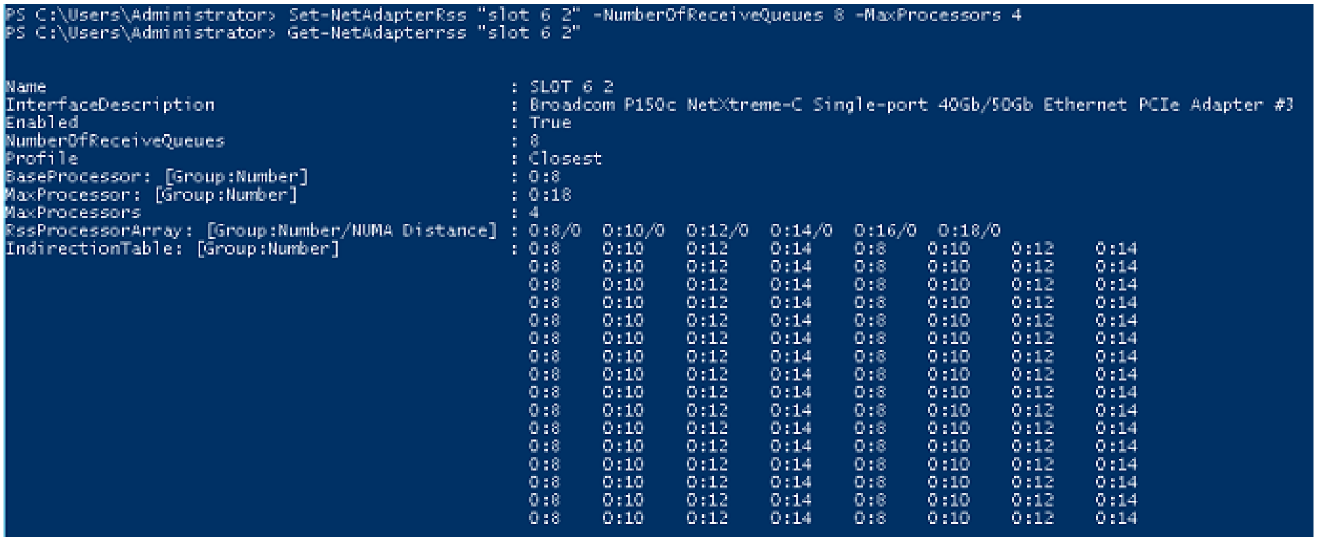
```
PS C:\Users\Administrator> Set-NetAdapterRss "slot 6 2" -BaseProcessorNumber 8 -MaxProcessorNumber 14
PS C:\Users\Administrator> Get-NetAdapterrss "slot 6 2"


Name                                            : SLOT 6 2
InterfaceDescription                            : Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
Enabled                                         : True
NumberOfReceiveQueues                           : 4
Profile                                         : Closest
BaseProcessor: [Group:Number]                   : 0:8
MaxProcessor: [Group:Number]                    : 0:14
MaxProcessors                                   : 4
RssProcessorArray: [Group:Number/NUMA Distance] : 0:8/0  0:10/0  0:12/0  0:14/0
IndirectionTable: [Group:Number]                : 0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                                  0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
```

NOTE

In the RssProcessorArray, the OS is now limited to only using four processors (8 through 14 non-hyperthreaded cores) as the base processor is set to 8 with a max processor of 14.

3. To limit the number of cores in the OS to four, use the following command:

```
Set-NetAdapterRss "slot 6 2" -NumberOfReceiveQueues 8 -MaxProcessors 4.
```



```
PS C:\Users\Administrator> Set-NetAdapterRss "slot 6 2" -NumberOfReceiveQueues 8 -MaxProcessors 4
PS C:\Users\Administrator> Get-NetAdapterrss "slot 6 2"


Name                                        : SLOT 6 2
InterfaceDescription                        : Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
Enabled                                     : True
NumberOfReceiveQueues                       : 8
Profile                                     : Closest
BaseProcessor: [Group:Number]               : 0:8
MaxProcessor: [Group:Number]                : 0:18
MaxProcessors                               : 4
RssProcessorArray: [Group:Number/NUMA Distance] : 0:8/0  0:10/0  0:12/0  0:14/0  0:16/0  0:18/0
IndirectionTable: [Group:Number]            : 0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
                                              0:8    0:10    0:12    0:14    0:8    0:10    0:12    0:14
```

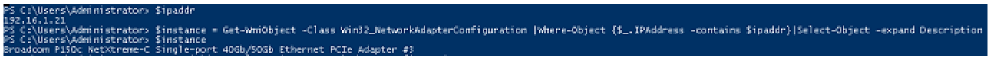
NOTE
In the indirection table, there are eight queues to service and there are six cores in the ProcessorArray, however, MaxProcessors is set to four. The indirection table is only populated with four processors, limiting it to four queues.

Monitoring Receive Queues/Cores

The counters are used to monitor where each flow is being received. A single stream is started with static MAC/IP/Port number (for example, the TCP Port number is specified and does not allow the OS to choose it). The Received Packets/sec on each core is monitored. The expectation is that only one core receives the packets per stream which indicates where the stream is being hashed. As long as the driver is not reloaded, this stream should always be hashed the same (for example, be very predictable).

1. Get-Counter -counter "\Per processor network interface card activity(*, {Adapter Description})\Received Packets/sec".
   Example: Get-Counter -counter "\Per processor network interface card activity(*, Broadcom P150c Single-port 40Gb-50Gb Ethernet PCIe Adapter #3)\Received Packets/sec" This command monitors the Received Packets/sec for the cores associated with the 50G Broadcom NIC. To get the
   specific count of an individual processor, change the * to the processor number (for example, 8 or 10 or 12 or 14).
2. The command in Step 1 executes a single time and returns the results. To execute the command multiple times, utilize the following flags in the command:
   – SampleInterval
   – MaxSamples
   – Continuous
3. If the proper adapter string is not known, find it by using the IP Address:

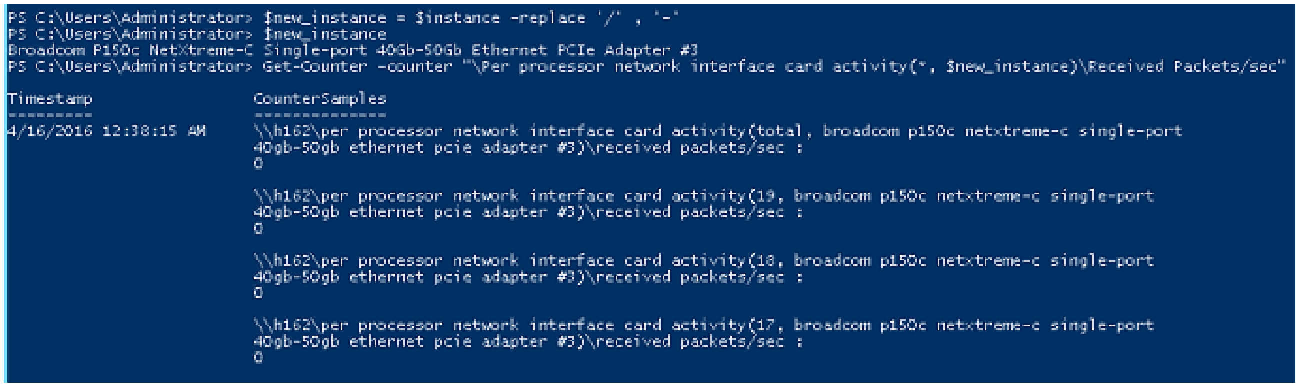
```
$ipaddr = "192.168.1.21"
$instance = Get-WmiObject -Class Win32_NetworkAdapterConfiguration |Where-Object {$_.IPAddress - contains
 $ipaddr}|Select-Object -expand Description
$instance now holds the string of the description of the adapter with the given IP address. This assumes
 that exactly one adapter has the given IP address.
```



```
PS C:\Users\Administrator> $ipaddr
192.16.1.21
PS C:\Users\Administrator> $instance = Get-WmiObject -Class Win32_NetworkAdapterConfiguration |Where-Object {$_.IPAddress -contains $ipaddr}|Select-Object -expand Description
PS C:\Users\Administrator> $instance
Broadcom P150c NetXtreme-C Single-port 40Gb/50Gb Ethernet PCIe Adapter #3
```

NOTE
Between 40Gb and 50Gb there is a '/' and a '-' was expected indicating a problem and should be explored further for other adapters.

4.
```
$new_instance = $instance -replace '/' , '-'
Get-Counter -counter "\Per processor network interface card activity(*, $new_instance)\Received Packets/
sec"
```



```
PS C:\Users\Administrator> $new_instance = $instance -replace '/' , '-'
PS C:\Users\Administrator> $new_instance
Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3
PS C:\Users\Administrator> Get-Counter -counter "\Per processor network interface card activity(*, $new_instance)\Received Packets/sec"

Timestamp                 CounterSamples
---------                 --------------
4/16/2016 12:38:15 AM     \\h162\per processor network interface card activity(total, broadcom p150c netxtreme-c single-port
                          40gb-50gb ethernet pcie adapter #3)\received packets/sec :
                          0

                          \\h162\per processor network interface card activity(19, broadcom p150c netxtreme-c single-port
                          40gb-50gb ethernet pcie adapter #3)\received packets/sec :
                          0

                          \\h162\per processor network interface card activity(18, broadcom p150c netxtreme-c single-port
                          40gb-50gb ethernet pcie adapter #3)\received packets/sec :
                          0

                          \\h162\per processor network interface card activity(17, broadcom p150c netxtreme-c single-port
                          40gb-50gb ethernet pcie adapter #3)\received packets/sec :
                          0
```

a. Another command that reports the same information is interrupts/sec from the per processor network interface card activity object.

```
Get-Counter -counter "\Per processor network interface card activity(*, $new_instance)\Interrupts/ sec"
```

b. The interrupts/sec from the processor object (this does not allow filtering based on a specific network adapter).

```
Get-Counter -counter "\Processor(*)\Interrupts/sec"
```

5. Using the Get-Counter cmdlet, cores can be monitored while transmitting traffic. As each stream is transmitted, monitor the receive packets to understand which queue is being utilized. When a stream that exercises a new unique queue is found, record it. If the stream exercises an already used queue, try a new stream with a different DATA Port.

   NOTE
   In NetPerf, the data ports are the ports that matter in this case, not the control port.

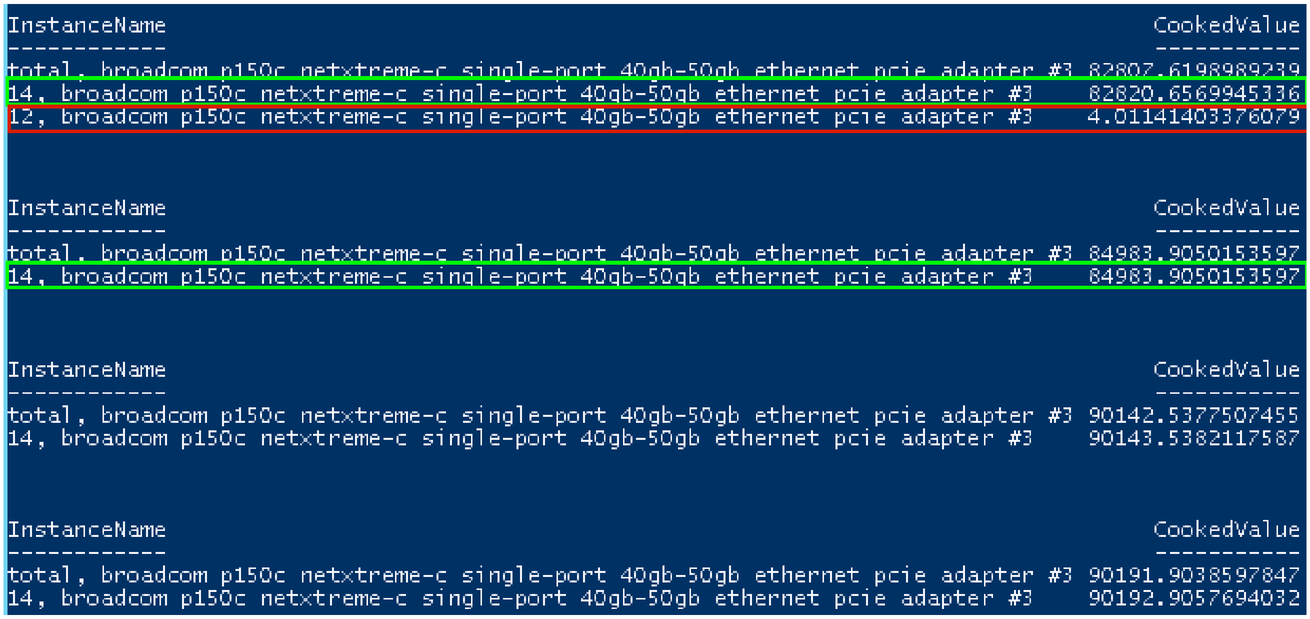
6. To send traffic using NetPerf, use the following command:

```
netperf.exe -l 10 -H 192.16.1.21 -p 12801 -P 0 -- -S 512K-m 64k-s 512K -P 32212,32214
```

   NOTE
   12801 is the Control Port. 32212 is the control port on the transmit side while 32214 is the data port on the remote side.

7. On the receive side, run the following command to monitor the receive packets per core. Only cores that receive packets report. This command is run 20 times:

```
1..20 | % { (Get-Counter -counter "\Per processor network interface card activity(*,
$new_instance)\Received Packets/sec").countersamples|Where-Object cookedvalue -ne 0| Format-Table
 InstanceName, CookedValue -auto}
```

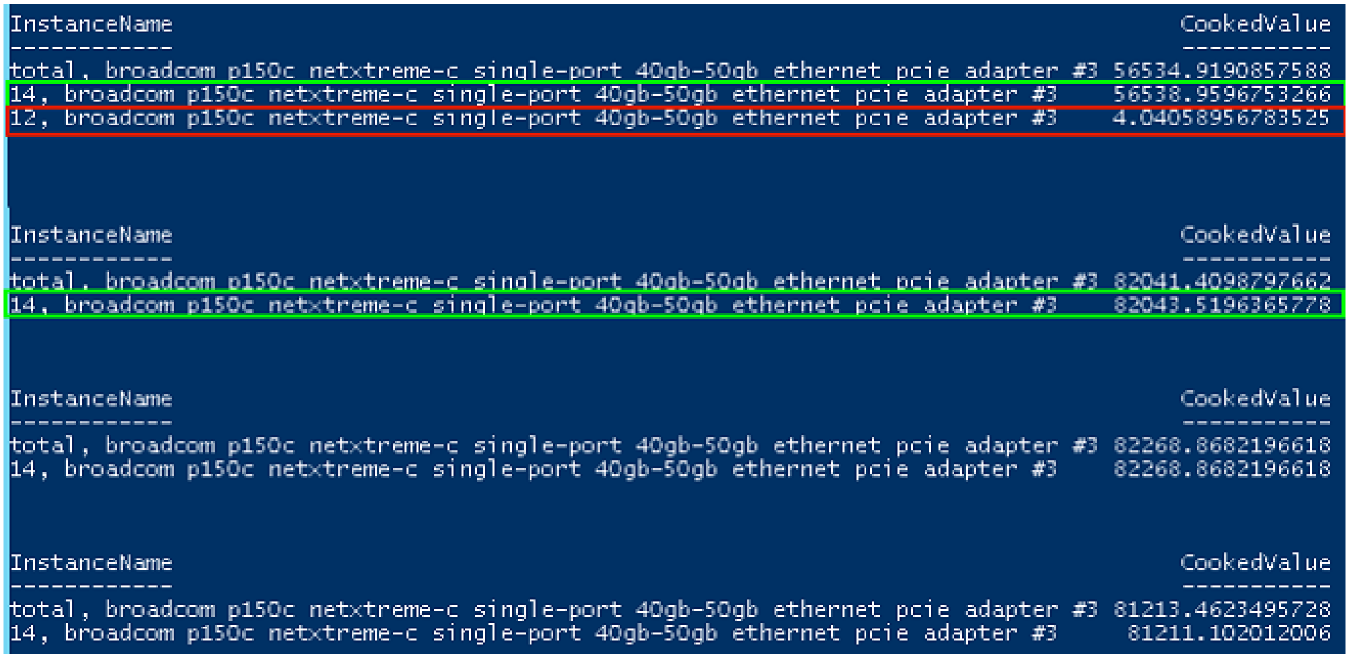


```
InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 82807.6198989239
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    82820.6569945336
12, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    4.01141403376079


InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 84983.9050153597
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    84983.9050153597


InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 90142.5377507455
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    90143.5382117587


InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 90191.9038597847
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    90192.9057694032
```

NOTE
Initially, there are two cores receiving packets. The first in red (12) is the control port setting up the test. The second core (14) is the data port. Additional packets are reported after initiation. Ensure that this transmit command resolves to core 14.

8. To resolve multiple streams, try a new stream with a different data port where it does not resolve to the same core as previously (for example, core 14).

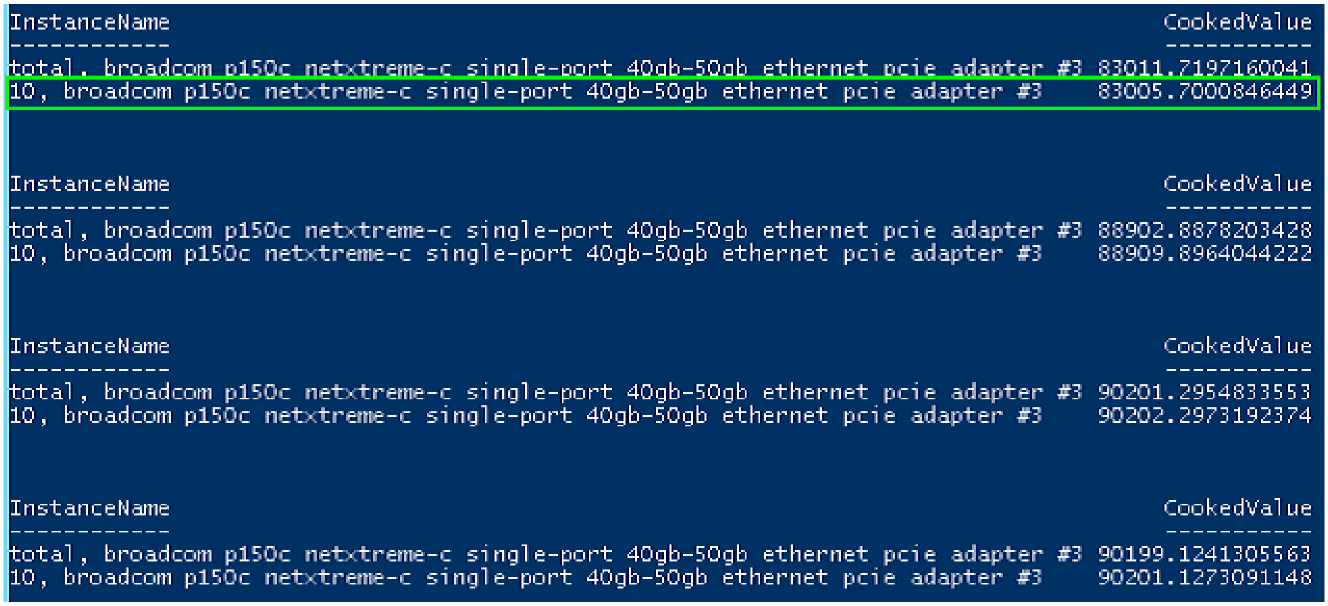
```
netperf.exe -l 10 -H 192.16.1.21 -p 12802 -P 0 -- -S 512K-m 64k-s 512K -P 32223,32225
```



```
InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 56534.9190857588
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    56538.9596753266
12, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    4.04058956783525


InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 82041.4098797662
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    82043.5196365778


InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 82268.8682196618
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    82268.8682196618


InstanceName                                                                   CookedValue
------------                                                                   -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 81213.4623495728
14, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3     81211.102012006
```

Since this stream also resolves to core 14, it cannot be used. A new data port must be tried.

```
netperf.exe -l 10 -H 192.16.1.21 -p 12802 -P 0 -- -S 512K-m 64k-s 512K -P 32245,32247
```



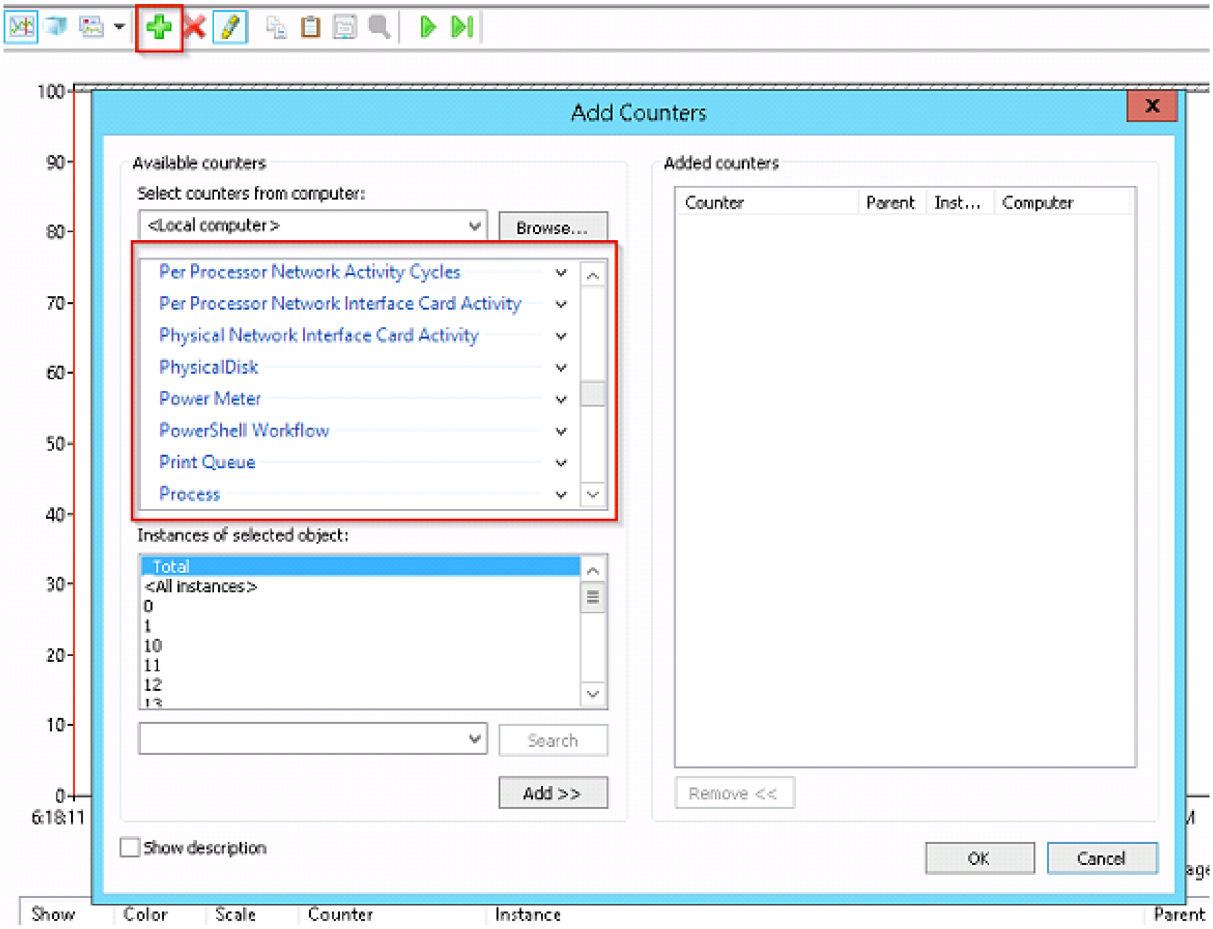
```
InstanceName                                                              CookedValue
------------                                                              -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 83011.7197160041
10, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    83005.7000846449


InstanceName                                                              CookedValue
------------                                                              -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 88902.8878203428
10, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    88909.8964044222


InstanceName                                                              CookedValue
------------                                                              -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 90201.2954833553
10, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    90202.2973192374


InstanceName                                                              CookedValue
------------                                                              -----------
total, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3 90199.1241305563
10, broadcom p150c netxtreme-c single-port 40gb-50gb ethernet pcie adapter #3    90201.1273091148
```

This stream utilizes core 10 and can be kept. Two streams are resolved to different cores. Continue this process until all cores are resolved.

Expanding to New Counters

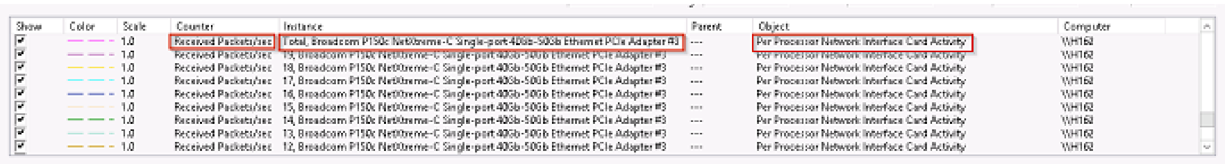
Additional or different counters must be monitored. There are several ways to discover what is available. Query using PowerShell or open Performance Monitor.

1. Once the application is open, click the **+** button. This displays every available counter.



2. For this example, select **Per Processor Network Interface Card Activity** and from the drop-down menu **Received Packets/sec**.
3. Select an instance to monitor from **Instances of selected object**.
4. Select **<All instances>**for this example.
5. Once the **Counter and Instance are highlighted**, click **Add>>**. An item displays in **Added Counters**.
6. Click **OK**.

   A large window on top displays the monitor and a small window displays each counter instance.
7. For debugging, run the test and visually monitor the counters. Select show/hide, scale, highlight counters, and several other items.
8. Monitor the instances window. Three columns must be extracted: Counter, Instance, and Object.



Show	Color	Scale	Counter	Instance	Parent	Object	Computer
☑	— — -	1.0	Received Packets/sec	Total, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	19, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	18, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	17, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	16, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	15, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	14, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	13, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162
☑	— — -	1.0	Received Packets/sec	12, Broadcom P150c NetXtreme-C Single-port 40Gb-50Gb Ethernet PCIe Adapter #3	---	Per Processor Network Interface Card Activity	\\WH162

- Object is the most important column. This is the first item selected above in the top left window. An object contains several counters.
- Counter is the next in importance and displays the specific counter monitored within the object.
- Instance is another level of granularity that is monitored.

For this specific counter, the instance is a tuple. The first value is the processor and the second the adapter. For instance, if only core 0 was monitored for this particular NIC, monitor 0, Broadcom P150c Single-port 40Gb-50Gb Ethernet PCIe Adapter #3 is the result.

Performance Tuning

Congestion control is a comprehensive solution used for congestion management that can help reduce packet losses and congestion spreading as well as improve latency by keeping switch queue levels low.

Congestion control performance is measured by these network traffic performance metrics during periods of congestion:

- Fairness of bandwidth allocation between QPs
- Link utilization
- Latency

Under heavy congestion, congestion control can enforce fairness at a per-QP level, with low variation between streams. Streams not crossing the point of network contention are not affected.

Performance may vary depending on the topology, number of flows, traffic types, and the application used. Careful consideration and understanding should be used when adjusting the different congestion control parameters. As a default, deterministic marking with DCTCP is recommended and can be configured using the setup script provided in Congestion Control Tuning Parameters.

This section provides the following information on performance tuning:

- Priority Flow Control
- CNP Traffic Classes
- Disable CPU Power Saving
- Tuning Applications for High QP Count

Priority Flow Control

Although PFC is not required, it is recommended to enable Priority Flow Control to ensure that packets are not dropped in bursty or transient scenarios. In cases where PFC is not being utilized, adjustments of the CC tunables are required to ensure the ramp-up and ramp-down are appropriate based on the scenario.

CNP Traffic Classes

It is recommended that CNPs are classified onto a different traffic class on both the NIC and the switch for optimal performance to ensure CNPs are not utilizing the same buffers as the RoCE packets.

Disable CPU Power Saving

Intel CPU power saving technology can cause packet RoCE drops with bursty traffic. It is highly recommended to set BIOS power saving mode to maximum performance, and disable c-states and p-states with kernel options by editing the `/etc/ default/grub` file, and appending the following to GRUB_CMDLINE_LINUX_DEFAULT:

```
intel_pstate=disable processor.max_cstate=1 intel_idle.max_cstate=0
```

Rebuild the grub configuration as follows:

```
sudo /usr/sbin/grub-mkconfig* -o /boot/grub/grub.cfg
```

Tuning Applications for High QP Count

At larger scales, when hundreds of QPs are active and Send/Recv protocol is used to exchange data there, is a possibility of observing RNR-NAKs. As per the IB-specification, RNR-NAKs are recoverable errors and the application can be tuned to minimize the occurrence of RNR-NAKs. Parameters that help minimize the RNR-NAKs are as follows:

- Bind the task to the CPU with a matching NUMA node to the network adapter.
- Increase the depth of RX queue using an application specific parameter. For example, `ib_send_bw` has the -r option to increase receive queue depth.
- If the application allows, tune the threshold of completion suppression. This is also known as CQ moderation. For example, `ib_send_bw` has the -Q option.

IP Forwarding Tunings

This section provides the following information on IP forwarding tunings:

- BIOS Tuning
- Kernel Tuning
- NIC Tuning
- IP Forwarding Results

BIOS Tuning

This section provides information on BIOS tuning.

- SVM Mode – Disabled
- SMEE – Disabled
- SR-IOV Support – Disabled
- Custom Pstate0 – Auto
- Custom Pstate1 – Disabled
- Custom Pstate2 – Disabled
- SMT Control – Disabled
- Local APIC Mode – x2APIC
- NUMA nodes per socket – NPS4
- DDR Timing Configuration → Overclock – Enabled
- Memory Clock Speed – 2666 MHz
- DDR Power Options → Power Down Enable – Disabled
- IOMMU – Auto
- Determinism Control – Manual
- Determinism Slider – Performance

Kernel Tuning

This section provides information on kernel tuning.

- Add the following entries to the kernel command line:
  amd_iommu=off iommu=off nohz=off (for 100G Link, set iommu=pt)
- Map Interrupts to CPU's (CPUs that belong to local NUMA, one IRQ per CPU)
- Disable the following services
  - Firewalld

  systemctl stop firewalld
  - Selinux
    echo 0 > /selinux/enforce
  - Set CPU to run at max frequency
    echo performance | sudo tee /sys/devices/system/cpu/cpu*/cpufreq/scaling_governor
  - irqbalance
    systemctl stop irqbalance
  - NetworkManager
  systemctl stop NetworkManager
- Ensure NAT/IP Table modules are unloaded (list of modules that need to be unloaded)
  - iptable_raw
  - iptable_security
  - kvm_amd
  - ip6table_mangle
  - ip6table_security
  - ip6table_raw
  - iptable_mangle
  - iptable_filter
  - ip_tables
  - ip6table_filter
  - ip6_tables
  - ipt_REJECT
  - ebtable_nat
  - ebtable_filter
  - ebtables
  - kvm_intel
  - Kvm

Equation 1:

Open file /etc/modprobe.d/blacklist.conf and turn off auto load using below syntax alias driver-name off

NOTE: Blacklist the dependent modules that do not unload any of the previous modules

NIC Tuning

This section provides information on NIC tuning.

- Increase combined rings to 16 (1 ring per physical core in the local CCD, determine local cores using lscpu commands):
  ethtool -L [interface] combined 16 rx 0 tx 0
- Disable Pause:
  ethtool -A [interface] tx off rx off
- Disable LRO and GRO:
  ethtool -K [interface] rx-gro-hw off lro off gro off
- Turn ON TX no cache copy:
  ethtool -K [interface] tx-nocache-copy on
- Increase TX/RX Ring size to 2047:

  ethtool -G [interface] tx 2047 rx 2047

- Configure ntuple filter (to have even distribution across all rings):

  ethtool -N [interface] flow-type [udp4/tcp4] src-ip [sip] dst-ip [dip] src-port [sport] dst-port [dport] action [Queue to redirect]
- Interrupt Moderation (Not required for 25G):

ethtool -C [interface] rx-usecs 512 tx-usecs 512 rx-frames 512 tx-frames 512

IP Forwarding Results

This section provides information on IP forwarding results.

- IP Forwarding is typically limited by the Linux kernel. Therefore, the results scale with the number of physical cores utilized. It is common to expect roughly 600K – 800 KP/s per physical core utilized.
- BCM5741X (25G) Results:
  - Forwarding Rate is ~18 MP/s using 64B frame
  - Line-Rate from 256B onwards
- BCM575XX (100G) Results:
  - Forwarding Rate is ~18 MP/s using 64B frame

Performance Optimization – NetPerf Test

This section describes the test tool used, configuration parameters, and basic instructions for optimizing performance throughput using NetPerf.

NetPerf is a tool that generates performance metrics over networks. The two metrics that are captured in this document are NetPerf throughput and NetPerf latency. The throughput is reported in Mb/s and the latency is reported as transactions/s. The steps to measure and compute these metrics are shown in the following sections.

This section provides the following information on performance optimization:

- Guidelines
- Basic Instructions
- Throughput Tests
- Collecting Throughput Results
- Latency – Request/Response Test
- Collecting Latency Results
- Running Bi-Directional Traffic
- Running More Than a Single Session
- Helpful Items

Guidelines

There are two types of tests. The first is the NetPerf throughput test. Throughput is defined as the amount of data that is successfully received and/or transmitted. This means that one must take into account the frame overhead to successfully calculate the maximum theoretical throughput. The second test is a latency-like test. This test counts the number of transactions per second. A transaction is a set of requests and responses. There can only be one outstanding request at a time. Once a response is received, a new request is sent immediately. When transactions are known, calculate the round-trip latency by taking its inverse. That is, latency_rt = 1/(transactions/s). This number is on a per-session level.

The following steps allow testers to produce optimal performance for their system on each of these tests. The basic topology for all NetPerf tests is as follows:

For each session:

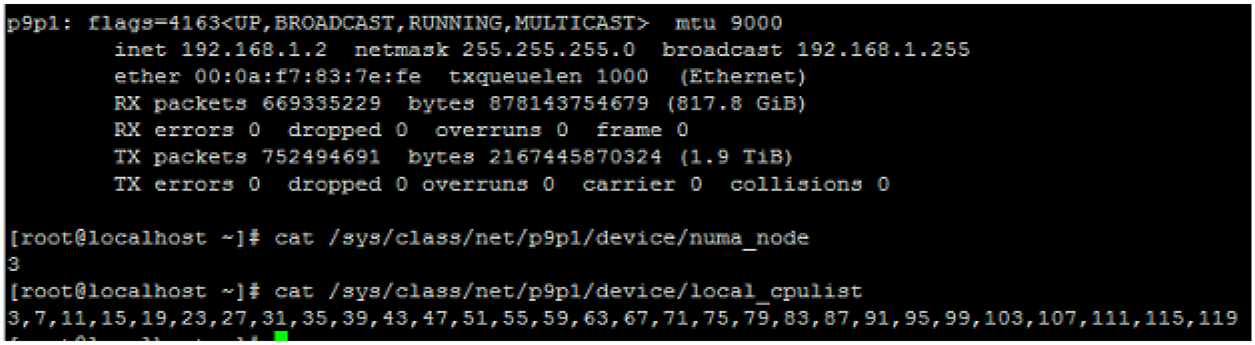
- NetPerf instance on the client side.
- Netserver instance on the server side.
- Transmit queue on the client side to send data.
- Receive queue on the server side to receive data.
- Transmit queue on the server side to send ack.
- Receive queue on the client side to receive ack.

This information is important when trying to understand how to configure the affinities.

Basic Instructions

Use the following instructions for both throughput and latency tests.

1. Connect two systems back-to-back, or via ports on a local network switch. Ensure that the NICs are linked at the desired speed.
   To change link speed:
   ethtool -s <interface> speed <speed>
   Example: ethtool -s p6p1 speed 25000
2. Ensure the systems under test can ping each other. If using 25G/50G, use (Q)SFP/28.
3. Configure the number of queues to be used on each side (default: RX 8 TX 8) with the following command:
   ethtool -l <interface> (show currect config) ethtool -L <interface> combined <num_queues> It is also possible to split queues:
   ethtool -L <interface> rx <num_queues> tx <num_queues> combined 0
   Only combined queues or split queues can be used. They cannot be allocated at the same time. When changing from split to combined or from combined to split, explicitly set the unused queues to 0:
   ethtool -L <interface> combined <num_queues> rx 0 tx 0
4. Determine the NUMA node that the PCIe device is connected to by using the following command:
   cat /sys/class/net/<interface>/device/numa_node
   cat /sys/class/net/<interface>/device/local_cpulist (processors on the same numa node)



```
p9p1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 9000
        inet 192.168.1.2  netmask 255.255.255.0  broadcast 192.168.1.255
        ether 00:0a:f7:83:7e:fe  txqueuelen 1000  (Ethernet)
        RX packets 669335229  bytes 878143754679 (817.8 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 752494691  bytes 2167445870324 (1.9 TiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[root@localhost ~]# cat /sys/class/net/p9p1/device/numa_node
3
[root@localhost ~]# cat /sys/class/net/p9p1/device/local_cpulist
3,7,11,15,19,23,27,31,35,39,43,47,51,55,59,63,67,71,75,79,83,87,91,95,99,103,107,111,115,119
```

5. Disable irqbalance – all affinities are set manually. Irqbalance introduces unnecessary overhead that can affect performance results.

systemctl stop irqbalance systemctl status irqbalance

Throughput Tests

For throughput tests, allow all queues to have their own processor. Do not share these CPUs. Determine the number of queues to use and assign each queue a unique CPU core. Ensure that no other process (NetPerf, NetServer) lands on the same CPU core.

Best practices:

- Use all physical cores before using hyper-threaded cores.
- Once all physical cores have been exhausted, move to hyper-threaded cores.
- Do not move off of the PCIe NUMA node.

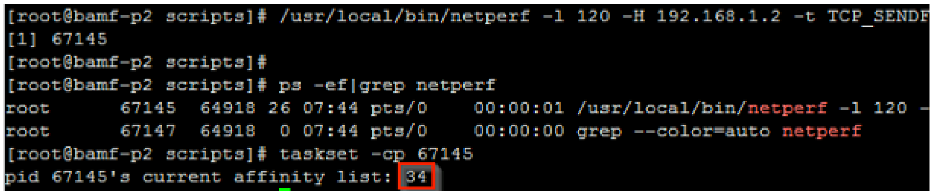
Run iterations of at least 1, 2, 4, 8, 10, 12, and 16 streams. For streams 1, 2, 4, and 8, use a single processor to run the NetPerf instances. For 10, 12, and 16 streams, use two processors to run the NetPerf streams.

> **NOTE**
> If the CPU has a higher core count, the streams can be processed on the unique CPU. On the server side, allow a separate core for each Netserver instance.

Use the following procedure for both the client and the server.

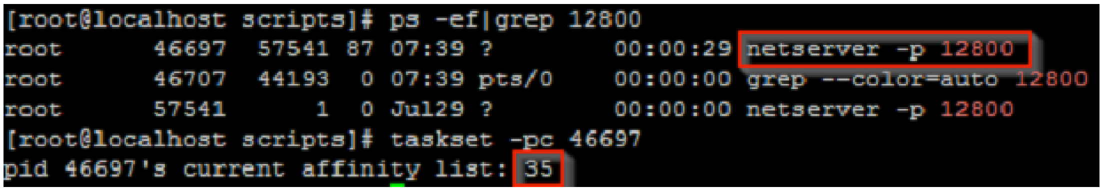
1. Configure IRQ and Application Affinity.
2. The following settings are recommended for the best performance results:
   - TSO on
   - GRO on
   - GSO on
   - Interrupt coalescing default
3. To create a unique control port to listen on, start a NetServer instance on the receive side for each session using the following command.
   - netserver -p <control port>

     For example, for three sessions:
     - netserver -p 12800
     - netserver -p 12801
     - netserver -p 12802
4. Configure RSS for Performance in Linux or Configure RSS for Performance in Windows if running more than a single session.
5. Use the -T option to affinitize the NetPerf and NetServer processes. Identify the process ID and affinity list ID of each netperf and netserv sessions. See the following example:
   a. The first number is the processor on which the local NetPerf runs.



```
[root@bamf-p2 scripts]# /usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDF
[1] 67145
[root@bamf-p2 scripts]#
[root@bamf-p2 scripts]# ps -ef|grep netperf
root      67145  64918 26 07:44 pts/0    00:00:01 /usr/local/bin/netperf -l 120 -
root      67147  64918  0 07:44 pts/0    00:00:00 grep --color=auto netperf
[root@bamf-p2 scripts]# taskset -cp 67145
pid 67145's current affinity list: 34
```

   b. The second number is the processor on which the remote NetServer runs. This creates a child process from the original NetServer process created in step 1.



```
[root@localhost scripts]# ps -ef|grep 12800
root      46697  57541 87 07:39 ?        00:00:29 netserver -p 12800
root      46707  44193  0 07:39 pts/0    00:00:00 grep --color=auto 12800
root      57541      1  0 Jul29 ?        00:00:00 netserver -p 12800
[root@localhost scripts]# taskset -pc 46697
pid 46697's current affinity list: 35
```

6. Run NetPerf by defining the control port on which the NetServer instance is listening, the data port that distributes RSS, and the CPU that affinitizes the process using the following command:

> **NOTE**
> By defining SRC/dest ports from both the control and data flow, the flow is distributed to the same ring each time it is used until the driver is reloaded and the hash changes. However, to reuse this flow, wait for the

TIME_DELAY to timeout. Check ss -tan on Linux to see all open sockets. If it is not required for the flow to consistently go to the same queue, it is not necessary to use -P <port> and a new flow is created each iteration and hash to a new queue. It is not required to wait for the timeout.

./usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,35 -p 12800,32100 -P 0 -- -m 64K -s 512K -S 512K -P 12900,32200

NetPerf arguments:

-l length of test. This will always be at least 60 but generally should be close to 120 to take out some of the connection time overhead.

-H the host that NetServer is listening on

-t the NetPerf test to be run. Use TCP_SENDFILE and TCP_RR (See the manual for available tests and descriptions.)

-T set the CPU affinity for NetPerf locally and nersever remotely. The CPU that NetPerf is running on is the first argument while the second argument is the NetServer CPU.

Defining this parameter spawns a NetServer child, running on the given CPU, from the original NetServer process running to handle the control connection

-p the ports to use for the control connection. The first port is the NetServer port while the second is the NetPerf port to use.

-P toggles the NetPerf banner. Defining the banner enables a more readable output when running many sessions.

-- separates NetPerf commands, control connection commands from test parameters that should be specific to the type of test running (for example, TCP_SENDFILE).

-m message size

-s local socket size

-S remote socket size

-P the ports to use for the control connection. The first port is the NetServer port while the second is the NetPerf port being used.

Collecting Throughput Results

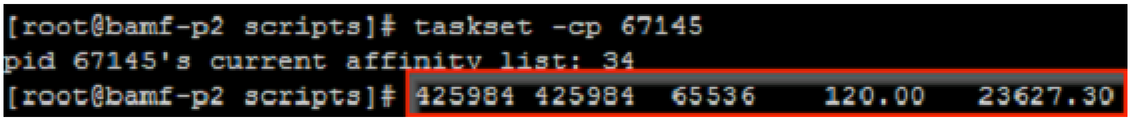
By default, the following numbers are reported:

Recv Send Send

Socket Socket Message Throughput Size Size Size Time

bytes bytes bytes secs. 10^6bits/sec 425984 425984 65536 120.00 23369.86

Since several streams are running, provide the -P 0 option in the test parameter. In this way, the banner does not show up for each stream, just the numbers.



```
[root@bamf-p2 scripts]# taskset -cp 67145
pid 67145's current affinity list: 34
[root@bamf-p2 scripts]# 425984 425984  65536    120.00   23627.30
```

In the previous example, 23627.30 is the recorded throughput. Only the final number is recorded. Override the default data output by using the -O parameter.

Latency – Request/Response Test

Configuration for this test is very similar to the throughput test. This test is only run uni-directionally so there is only NetPerf or NetServer running on each client. Each queue must have a dedicated processor and each instance of NetPerf or NetServer must also have its own dedicated processor. When all physical CPU cores have been assigned, continue using logical (hyperthreads) cores. After hyperthreads cores are consumed, move back to the cores that NetPerf and NetServer are on. For the best performance, disable processor C states, active state power management (ASPM), and set the BIOS settings to performance mode.

1. Configure the number queue as discussed in the Basic Instructions. Do not use the cores dedicated to these queues for anything else.
2. Since this is a latency test, Disable GRO and GSO on both interfaces (client and server) using the following command:
   ```
   ethtool -K <interface> gro off gso off
   ```
3. Run the NetPerf command from the client using the following command:
   ```
   /usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12800,32100 -T 82,83 -P 0 -- -P
   12900,32200
   ```
4. The principles for the control connection ports, the CPU affinity, and data connection ports are still valid. Ensure that all of these items are considered.

Collecting Latency Results

The test reports transactions/sec. For each session, collect the transactions/sec value. The following data is collected from transactions/sec:

1. Transactions/sec for each stream and the inverse of each number gives the latency, sec/transaction, for each stream.
2. Add all of the transactions/sec for all streams, total transactions/sec.
3. Divide total transactions/sec by the number of streams to get average transactions/sec.
4. Take the inverse of the average transactions/sec to get average latency.

NOTE
For measuring application latency of a single stream, it is best to allocate a single queue on both the client and server and assign the same IRQ affinity to both netperf and netserver. Since this is not high-throughput traffic, having all data on a single core helps latency by locality.

Running Bi-Directional Traffic

Both directions of traffic must start close together. To ensure that the processes start at the same time, use Ncat. To find additional details on Ncat, use Linux man pages.

Use the following procedure to create a separate script for each client:

1. Configure Ncat to listen on a specific unique port.
2. List each NetPerf command to be executed from each client to run in the background.
3. Run the script from the client system to be executed on.

NOTE
For bi-direction traffic, two different scripts would be running on two clients.

The script does not proceed until it receives an incoming connection on the given port. An example script is as follows:

```
netperf-clientA.sh #!/bin/sh

nc -l -p 9999

/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,35 -p 12800,32100 -P 0 -- -m 64K
```

-s 512K -S 512K -P 12900,32200 &

/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,39 -p 12801,32101 -P 0 -- -m 64K

-s 512K -S 512K -P 12901,32201 &

/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_SENDFILE -T 34,43 -p 12802,32102 -P 0 -- -m 64K

-s 512K -S 512K -P 12902,32202 &

Two clients are now running a script blocked on the nc command. To push the scripts past the nc command and to execute the NetPerf instances, initiate a connection on each of the clients by running the following commands from a third client:

goClients.sh #!/bin/sh

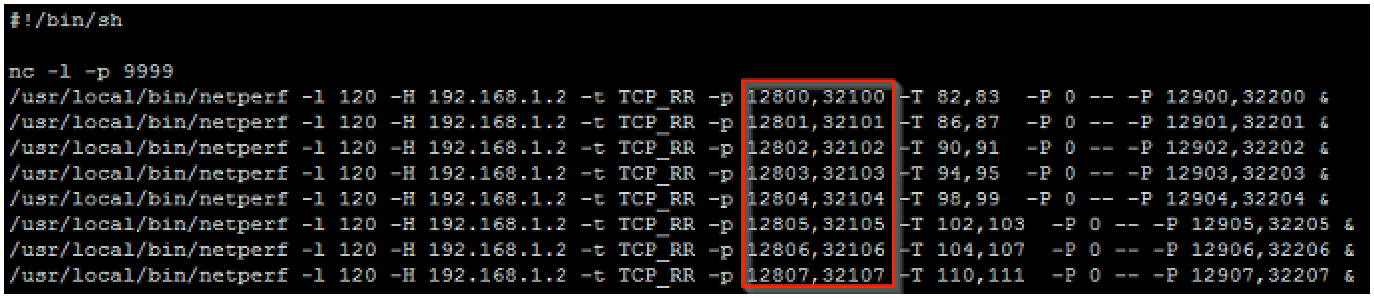
echo 'H' | nc <client1 ip> 9999 & echo 'H' | nc <client2 ip> 9999

Running More Than a Single Session

Running more than a single session is accomplished the same way as running bi-directional traffic with one exception. Instead of having the script running on each of the clients, only execute the script on the client running NetPerf.

When the goTest.sh script is executed, it causes the multiple NetPerf instances to be created on the single client.

1. The control connection ports can be incremented as sessions are added. These ports have no barring on the test results and are used to create a connection to pass information.



```
#!/bin/sh

nc -l -p 9999
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12800,32100 -T 82,83  -P 0 -- -P 12900,32200 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12801,32101 -T 86,87  -P 0 -- -P 12901,32201 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12802,32102 -T 90,91  -P 0 -- -P 12902,32202 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12803,32103 -T 94,95  -P 0 -- -P 12903,32203 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12804,32104 -T 98,99  -P 0 -- -P 12904,32204 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12805,32105 -T 102,103  -P 0 -- -P 12905,32205 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12806,32106 -T 104,107  -P 0 -- -P 12906,32206 &
/usr/local/bin/netperf -l 120 -H 192.168.1.2 -t TCP_RR -p 12807,32107 -T 110,111  -P 0 -- -P 12907,32207 &
```

2. Follow the instructions for configuring RSS so that the data ports are distributed correctly.
3. Configure each session so that the NetPerf and NetServer instances are distributed correctly.

Helpful Items

- For each connection, ensure that the control port numbers and data port numbers are unique. Do not create multiple sockets on the same port.
- When running a new test using the same port numbers from a previous run, make sure the TIME-WAIT delay has expired. The following error occurs otherwise:

  The socket established control could not establish the control connection from 0.0.0.0 port 32100 address family AF_UNSPEC to 192.168.1.1 port 12800 address family AF_INET
- Run the following command to determine if there are sockets in the TIME-WAIT state:

  "ss -tan state TIME-WAIT"
- When analyzing whether processes and queues are affinitized properly, check MPSTAT.

– For each processor, look at the %idle column.
– If the value here is less than 30%, investigate further.
– For the processor under 30% idle, look at the %soft column and the %sys column.
– If the number under %sys is high, it indicates that the process, NetPerf or NetServer, is utilizing the CPU heavily.
– If the number under %soft is high, it indicates that the queue is utilizing the CPU heavily.

Example RoCE + TCP Network Configuration

In this section, we will present an example construction of a set of servers on a 100G network with RoCE enabled and configured for co-existence with IP traffic, using CentOS 8.3, and an Arista switch. This is a typical pattern for a High Performance Computing or Machine Learning cluster. We will then test the cluster with the OSU Benchmark suite.

Downloading Software

Using a web browser, start with the Broadcom Ethernet Network Adapters page, select your product, go to the Downloads tab, Drivers folder, and download the Linux Driver Installer package.

Copy this package to each of the servers.

Installing CentOS

Install CentOS 8.3 from ISO or provisioning system using Minimal profile. Set root password consistently.

Configuring the SSH Keys

On one server, which will be your control node run the following commands:

```
# ssh-keygen
# eval `ssh-agent`
# ssh-copy-id root@<each other host>
```

Updating the Servers

On each server run the following commands as root:

```
# yum update -y
# yum install gcc gcc-c++ make
# reboot
```

Installing the Software

On each server run the following commands as root:

```
# tar xf <Linux Driver Installer Package File>
# cd <DIR>/Linux/Linux_Installer
# lspci | grep Broadcom
# ./install.sh -v -i <device PCIe address, eg: 41:00.0>
```

NOTE
Add -a <IP> -n <NETMASK> if the NIC does not already have a configured IP address.

Configuring the Switch

Configure your switch for PFC and ECN. In the following example, all ports are configured on a 32-port switch.

```
$ ssh admin@<Managemen IP>
$ enable
# configure terminal
# qos map dscp 26 to traffic-class 3
# qos map dscp 48 to traffic-class 7
# interface Ethernet 1/1-32/1
# speed 100g
# tx-queue 3
# random-detect ecn minimum-threshold 64 kbytes maximum-threshold 65 kbytes max-mark-probability 100 priority-
flow-control mode on
# priority-flow-control priority 3 no-drop
```

Installing and Configuring OpenMPI

On each server run the following commands as root:

```
# wget https://download.open-mpi.org/release/open-mpi/v4.1/openmpi-4.1.0-1.src.rpm
# rpmbuild --rebuild openmpi-4.1.0-1.src.rpm
# rpm -i /root/rpmbuild/RPMS/x86_64/openmpi-4.1.0-1.el8.x86_64.rpm
# sed -i 's/0x16f0,0x16f1/0x16f0,0x16f1,0x1750/' /usr/share/openmpi/mca-btl-openib-device-params.ini
```

Compiling OSU Benchmarks

On each server run the following commands as root:

```
On each server run the following commands as root:

# wget http://mvapich.cse.ohio-state.edu/download/mvapich/osu-micro-benchmarks-5.7.tar.gz
# tar xf osu-micro-benchmarks-5.7.tar.gz
# cd osu-micro-benchmarks-5.7
# ./configure CC=/usr/bin/mpicc CXX=/usr/bin/mpicxx
# make
```

Running OSU Benchmarks

On your control node server run the following command:

```
# /usr/bin/mpirun --allow-run-as-root -np 2 -host smc1,smc2 /root/osu-micro-benchmarks-5.7/mpi/pt2pt/
```

Statistics

Provides information on gathering statistics for Broadcom Ethernet NIC controllers verifying system performance

This section describes a the following collection of statistics related to the operation of IP and RoCE:

- Ethernet Statistics
- RoCE Statistics
- Statistics Definitions

Ethernet Statistics

The Linux ethtool utility allows inspection of statistics for all traffic types (Ethernet, IP, RoCE) with its -S option. RoCE traffic is included in ethtool counters, but is not specifically separated. However, RoCE and IP traffic can be differentiated by CoS queue and priority. IP traffic is by default on priority 0 and COS queue 4. RoCE traffic is by default on priority 3 and CoS queue 0. Some example interesting statistics can be seen below for the p1p1 interface:

```
$ ethtool -S p1p1

rx_bytes_cos0: 3172896188
rx_packets_cos0: 51175745
…
rx_bytes_cos4: 1542996
rx_packets_cos4: 21330
…
rx_total_discard_pkts: 0
tx_total_discard_pkts: 0
…
rx_pfc_frames: 0
tx_pfc_frames: 0
…
rx_pause_frames: 0
tx_pause_frames: 0
```

RoCE Statistics

RoCE and congestion control statistics can be viewed from the sysfs files:

```
# cat /sys/kernel/debug/bnxt_re/bnxt_re0/info bnxt_re debug info:
Adapter count: 1
=====[ IBDEV bnxt_re_bond0 ]=============================
link state: UP Max QP: 0xff80 Max SRQ: 0x7fff Max CQ: 0xffff Max MR: 0x3ffff Max MW: 0x8000c Active QP: 2
Active SRQ: 0
Active CQ: 1
Active MR: 0
Active MW: 1
Recoverable Errors: 0 Active QPs P0: 0 Active QPs P1: 2
Rx Pkts: 42907336148
Rx Bytes: 2664310290076
Tx Pkts: 229562600627
Tx Bytes: 248444124528782
CNP Tx Pkts: 0
```

```
 CNP Rx Pkts: 338012327
     ...
```

NOTE
Incrementing CNP (congestion notification packet) counts indicate congestion is being detected within the network switch fabric.

Statistics Definitions

Table 46: Device Resource Limits

Counters	Description
`Max QP`	Max number of QP limit.
`Max SRQ`	Max number of SRQ limit.
`Max CQ`	Max number of CQs limit.
`Max MR`	Max number of memory region limit.
`Max MW`	Max number of memory window limit.
`Max AH`	Max number of Address handle limit.
`Max PD`	Max number of Protection domain limit.

Table 47: Active Resources

Counters	Description
`Active QP`	Number of active QPs.
`Active SRQ`	Number of active SRQs.
`Active CQ`	Number of active CQs.
`Active MR`	Number of active Memory Regions.
`Active MW`	Number of active Memory Windows.
`Active AH`	Number of active Address handles.
`Active PD`	Number of active Protection domains.

Table 48: Receive Counters

Counters	Description
`Total Rx Good Bytes`	Total number of received bytes including Ethernet and RoCE.

`Rx_good_bytes`
Or
`RoCE Only Rx Bytes`
Or
`Rx Bytes`
	Number of good RoCE bytes received by hardware, excluding errors and drops. This includes UD, RC traffic including Acks and Naks for all protocols.

`L2 Only Rx Bytes`	Number of Ethernet only bytes received.

Counters	Description
Rx_good_pkts Or RoCE Only Rx Pkts Or Rx Pkts	Number of good RoCE packets received by hardware, excluding errors and drops. This includes UD, RC traffic including Acks and Naks for all protocols.
L2 Only Rx Pkts	Number of Ethernet only packets received.
rx_atomic_req	Number of atomic-operation-request received.
rx_read_req	Number of read-requests received.
rx_read_resp	Number of read-response received.
rx_write_req	Number of RDMA-write requests received.
rx_send_req	Number of incoming sends.

Table 49: Transmit Counters

Counters	Description
Total Tx Good Bytes	Total number of transmitted bytes including Ethernet and RoCE.
Tx Bytes Or RoCE Only Tx Bytes	Number of good RoCE bytes transmitted by hardware. This includes UD, RC traffic for all protocols.
L2 Only Tx Bytes	Number of Ethernet only bytes transmitted.
Tx Pkts Or RoCE Only Tx Pkts	Number of good RoCE packets transmitted by hardware. This includes UD, RC traffic for all protocols.
L2 Only Tx Pkts	Number of Ethernet only packets transmitted.
tx_atomic_req	Number of transmitted Atomic operation requests.
tx_read_req	Number of transmitted read request.
tx_read_resp	Number of transmitted read responses.
tx_write_req	Number of RDMA-write requests transmitted.
tx_send_req	Number of send requests transmitted.

Table 50: Congestion Notification Counters

Counters	Description
CNP Tx Pkts	Number of RoCE CNP packets received.
CNP Rx Pkts	Number of RoCE CNP packets transmitted.
rx_ecn_marked_pkts	Number of CE CNPs received.

Table 51: Recoverable Errors

Counters	Description
Recoverable Errors	Number of recoverable errors detected. Recoverable errors are detected by the H/W. H/W instructs FW to initiate the recovery process. RC connection does not tear-down as a result of these errors.
to_retransmits	Number of retransmission requests.

Counters	Description
`rnr_naks_rcvd`	Number of RNR (Receiver-Not-Ready) NAKs received.
`dup_req`	Number of duplicated requests detected.
`missing_resp`	Number of responses missing.
`rx_roce_drop_pkts`	Number of incoming packets dropped by hardware.
`rx_roce_discard_pkts`	Number of incoming packets discarded by hardware due to malformed packets.
`res_oob_drop_count`	Number of times the receiver engine depleted of message buffers (MBUFs).
`res_oos_drop_count`	Number of Out of Sequence RoCE packets received.

Table 52: Fatal Errors

Counters	Description
`seq_err_naks_rcvd`	`Number of PSN sequencing error NAKs received.`
`max_retry_exceeded`	Number of retransmission requests exceeded the max.
`unrecoverable_err`	Number of unrecoverable errors detected.
`bad_resp_err`	Number of bad response errors detected.
`local_qp_op_err`	Number of QP local operation errors detected.
`local_protection_err`	Number of local protection errors detected.
`mem_mgmt_op_err`	Number of times H/W detected an error because of illegal bind/fast register/invalidate attempted by the driver.
`remote_invalid_req_err`	Number of invalid requests received from the remote RDMA initiator.
`remote_access_err`	Number of times H/W received a REMOTE ACCESS ERROR NAK from the peer.
`remote_op_err`	Number of times H/W received a REMOTE OPERATIONAL ERROR NAK from the peer.

Table 53: Responder Errors

Counters	Description
`res_exceed_max`	Number of times H/W detected incoming Send, RDMA write or RDMA read messages which exceed the maximum transfer length.
`res_length_mismatch`	Number of times H/W detected incoming RDMA write message payload size does not match write length in the RETH.
`res_exceeds_wqe`	Number of times H/W detected Send payload exceeds RQ/SRQ RQE buffer capacity.
`res_opcode_err`	Number of times H/W detected First, Only, Middle, Last packets for incoming requests are improperly ordered with respect to the previous packet.
`res_rx_invalid_rkey`	Number of times H/W detected an incoming request with an R_KEY that did not reference a valid MR/ MW.
`res_rx_domain_err`	Number of times H/W detected an incoming request with an R_KEY that referenced a MR/MW that was not in the same PD as the QP on which the request arrived.
`res_rx_no_perm`	Number of times H/W detected an incoming RDMA write request with an R_KEY that referenced a MR/ MW which did not have the access permission needed for the operation.
`res_rx_range_err`	Number of times H/W detected an incoming RDMA write request that had a combination of R_KEY, VA and length that was out of bounds of the associated MR/MW.
`res_tx_invalid_rkey`	Number of times H/W detected a R_KEY that did not reference a valid MR/MW while processing incoming read request.

Counters	Description
`res_tx_domain_err`	Number of times H/W detected an incoming request with an R_KEY that referenced a MR/MW that was not in the same PD as the QP on which the RDMA read request is received.
`res_tx_no_perm`	Number of times H/W detected an incoming RDMA read request with an R_KEY that referenced a MR/ MW which did not have the access permission needed for the operation.
`res_tx_range_err`	Number of times H/W detected an incoming RDMA read request that had a combination of R_KEY, VA and length that was out of bounds of the associated MR/MW.
`res_irrq_oflow`	Number of times H/W detected that peer sent us more RDMA read or atomic requests than the negotiated maximum
`res_unsup_opcode`	Number of times H/W detected that peer sent us a request with an opcode for a request type that is not supported on this QP.
`res_unaligned_atomic`	Number of times H/W detected that VA of an atomic request is on a memory boundary that prevents atomic execution.
`res_rem_inv_err`	Number of times H/W detected an incoming send with invalidate request in which the R_KEY to invalidate did not MR/MW which could be invalidated.
`res_mem_error64`	Number of times H/W detected a RQ/SRQ SGE which points to an inaccessible memory.
`res_srq_err`	Number of times H/W detected a QP moving to error state because the associated SRQ is in error.
`res_cmp_err`	Number of times H/W detected that there is no CQE space available on CQ or CQ is not in valid state.
`res_invalid_dup_rkey`	Number of times H/W detected invalid R_KEY while re-sending responses to duplicate read requests.
`res_wqe_format_err`	Number of times H/W detected error in the format of the WQE in the RQ/SRQ.
`res_cq_load_err`	Number of times H/W detected error while attempting to load the CQ context.
`res_srq_load_err`	Number of times H/W detected error while attempting to load the SRQ context.

Table 54: PCI Errors

Counters	Description
`res_tx_pci_err`	Number of PCI errors detected during transmission by responder.
`res_rx_pci_err`	Number of PCI errors detected during reception by responder.

Table 55: Device Control Plane Counters

Counters	Description
`num_irq_started`	Number of times the control plane interrupt service routine had enabled.
`num_irq_stopped`	Number of times the control plane interrupt service routine had disabled.
`poll_in_intr_en`	Number of times a control path command had timed out in interrupt mode and the driver had to fall on to polling mode.
`poll_in_intr_dis`	Number of times control path command has been completed in pure polling mode because the interrupt mode was disabled.

Application Configuration Examples

Provides examples to quickly get your Broadcom Ethernet controllers installed and configured.

This section provides the following installation and configuration information:

- MVAPICH2
- OpenMPI

MVAPICH2

Broadcom's RDMA device supports running HPC applications using the MVAPICH2 MPI distribution without any modification.

This section provides the following information on MVAPICH2:

- Installing MVAPICH2
- Building the Application

Installing MVAPICH2

To build and install MVAPICH2 to be used with Broadcom's RDMA devices:

1. Download the latest MVAPICH2 tarball from the MVAPICH2 website: http://mvapich.cse.ohio-state.edu/download/mvapich/mv2/mvapich2-2.3.5.tar.gz
2. Execute the following commands:

```
tat -zxvf mvapich2-2.3.5.tar.gz
cd mvapich2-2.3.5
$ ./configure --with-device=ch3:mrail --with-rdma=gen2
```

Building the Application

This section provides the step required to build the application with MVAPICH2.

> **NOTE**
> The following command assumes that mpicc and mpicxx have been added to the path $make CC=mpicc CXX=mpicxx.

To run the application, use the following command:

```
$ mpirun_rsh -np 4 n0 n0 n1 n1 ./cpi
```

OpenMPI

If OpenMPI 1.0 is in use, the OpenMPI source and configuration must be patched as follows:

```
$ cd openmpi-x.y.z
$ cat > openmpi.patch << EOF
diff --git a/opal/mca/common/verbs/common_verbs_port.c b/opal/mca/common/verbs/common_verbs_port.c
index 831ba3f..7ebeb30 100644
--- a/opal/mca/common/verbs/common_verbs_port.c
+++ b/opal/mca/common/verbs/common_verbs_port.c
@@ -68,6 +68,10 @@ int opal_common_verbs_port_bw(struct ibv_port_attr *port_attr,
        /* EDR: 25.78125 Gbps * 64/66, in megabits */
        *bandwidth = 25000;
        break;
+    case 64:
+        /* ODR: 50 Gbps * 64/66, in megabits */
+        *bandwidth = 50000;
+        break;
+     case 128:
+       /* ODR: 100 Gbps * 64/66, in megabits */
+       *bandwidth = 100000;
+         break;
    default:
        /* Who knows? */
        return OPAL_ERR_NOT_FOUND;
EOF
$ patch -p1 < openmpi.patch
$ make
$ sudo make install
```

NOTE
This patch is already present in OpenMPI 3.1.x and 3.x.

A device configuration section must also be added as follows:

```
$ sudo cat >> /usr/local/share/openmpi/mca-btl-openib-device-params.ini << EOF
[Broadcom Netxtreme]
vendor_id = 0x14e4
vendor_part_id =
 5637,5638,5652,5824,5825,5838,5839,5846,5847,5848,5849,5855,5858,5859,5861,5867,5869,5871,5872,5873, 5968
use_eager_rdma = 1
mtu = 1024
receive_queues = P,128,256,192,128:S,65536,256,192,128
max_inline_data = 96
EOF
```

Frequently Asked Questions

Provides answers to common questions and issues for Broadcom Ethernet NIC controllers.

This section contains information on the following frequently asked questions.

Why does the -x 3 command parameter fail when used with ib perf tools?

GID[3] is required for -x 3 to function. An ifconfig <index> down; ifconfig <index> up command is required to populate GID[3].

Is there a command to check PFC statistics?

The `ethtool -s <InterfaceIndex> |grep pfc |more` command shows the PFC statistics.

The libibverbs: Warning: Driver bnxt_re does not support the kernel ABI message displays.

This occurs after a kernel upgrade or Linux distribution update. To resolve this issue, rebuild the `bnxt_en` and `bnxt_re` driver files with the newer kernel.

What happens to applications when migrating from an older Broadcom Release using RoCEv1 to a new Broadcom Release which only supports RoCEv2?

In this example, this is the ibv_device output from an older release where RoCEv1 and RoCEv2 are both supported. Note the number of GID's in the example. There are four GID's composed of the following:

1. IPv6 / RoCEv1 (GID index 0)
2. IPv6 / RoCEv2 (GID index 1)
3. IPv4 / RoCEv1 (GID index 2)
4. IPv4 / RoCEv2 (GID index 3)

The IPv4 address is really an IPv4 address mapped into the IPv6 address space. This can be identified by 80 “0” bits, followed by 16 “1” bits (“FFFF” in hexadecimal), followed by the original 32-bit IPv4 address.

```
ibv_devinfo -v -d bnxt_re0 | grep GID
GID[ 0]: fe80:0000:0000:0000:0205:06ff:fe03:0200
GID[ 1]: fe80:0000:0000:0000:0205:06ff:fe03:0200
GID[ 2]: 0000:0000:0000:0000:0000:ffff:c9c9:c90a
GID[ 3]: 0000:0000:0000:0000:0000:ffff:c9c9:c90a

cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/0

IB/RoCE v1
cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/1
RoCE v2
cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/2
IB/RoCE v1
cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/3
RoCE v2
```

In the following example, this is the ibv_device output from a Broadcom release (219.0 and later) where only RoCEv2 is supported. Note the number of GID's. There are 2 GID's:

1. IPv6 / RoCEv2 (GID index 0)

2. IPv4 / RoCEv2 (GID index 1)

```
ibv_devinfo -v -d bnxt_re0 | grep GID
GID[ 0]: fe80:0000:0000:0000:0205:06ff:fe03:0200
GID[ 1]: 0000:0000:0000:0000:0000:ffff:c9c9:c90a
cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/0
RoCE v2
cat /sys/class/infiniband/bnxt_re0/ports/1/gid_attrs/types/1
RoCE v2
```

As part of migrating the application, the correct GID must be used. When using applications that need the GID to be specified as a parameter, ensure that the correct GID is used after any of the following link events:

- When using perftools, the -x is used as the index into the array of GID's supported by the device. In the previous example, the GID index for the IPv4 RoCEv2 for the previous release is 3 while for the current release it is 1.
- GID indexes can change with the NIC configuration and physical port bounces.
- To reset all GIDs, stop any running applications and restart the network interface.