



Intel® Ethernet Adapters and Devices User Guide

September 21, 2023

1. Overview

Welcome to the User Guide for Intel® Ethernet Adapters and devices. This guide covers hardware and software installation, setup procedures, and troubleshooting tips for Intel network adapters, connections, and other devices.

1.1 Intended Audience

This document is intended for information technology professionals with a high level of knowledge, experience, and competency in Ethernet networking technology.

1.2 Supported Operating Systems

The drivers in this release have been tested with the following operating systems (OSs). Additional OSs may function with our drivers but are not tested.

 **NOTE: Not all devices support all operating systems listed.** Refer to "Supported Devices" below for OS limitations for your device.

Microsoft* Windows Server*, Azure Stack HCI

- Microsoft Windows Server 2022
- Microsoft Windows Server 2019
- Microsoft Azure Stack HCI, version 23H2
- Microsoft Azure Stack HCI, version 22H2

VMware* ESXi*

- VMWare ESXi 8.0 U1
- VMware ESXi 7.0 U3

Linux*

- Red Hat* Enterprise Linux* (RHEL) 9.2
- Red Hat Enterprise Linux 8.8
- SUSE* Linux Enterprise Server (SLES) 15 SP5

1.3 Supported Devices

Supported 100 Gigabit Network Devices

Family	Device	OS Limitations
Intel® Ethernet 800 Series	Intel® Ethernet 100G 2P E810-C Adapter	None
	Intel® Ethernet 100G 2P E810-C-st Adapter	Linux and ESXi only
	Intel® Ethernet 100G 2P E810-C-stg Adapter	Linux and ESXi only

 **NOTE:**

- Devices based on the Intel® Ethernet Controller E810-C have an expected total throughput for the entire device of 100 Gb/s in each direction if one 100G cable is connected or if two 100G cables are connected.

Supported 40 Gigabit Network Devices

Family	Device	OS Limitations
Intel® Ethernet 700 Series	Intel® Ethernet 40G 2P XL710 QSFP+ rNDC	None
	Intel® Ethernet Converged Network Adapter XL710-Q2	None

 **NOTE:**

- Devices based on the Intel® Ethernet Controller XL710 (4x10 GbE, 1x40 GbE, and 2x40 GbE) have an expected total throughput for the entire device of 40 Gb/s in each direction.
- The first port of Intel® Ethernet 700 Series adapters will display the correct branding string. All other ports on the same device will display a generic branding string.
- For an Intel® Ethernet 700 Series adapter to reach its full potential, you must install it in a PCIe Gen3 x8 slot. Installing it in a shorter slot, or a Gen2 or Gen1 slot, will limit the throughput of the adapter.

Supported 25 Gigabit Network Devices

Family	Device	OS Limitations
Intel® Ethernet 800 Series	Intel® Ethernet 25G 4P E810-XXV-st Adapter	Linux and ESXi only
	Intel® Ethernet 25G 4P E810-XXV-stg Adapter	Linux and ESXi only
	Intel® Ethernet 25G 2P E810-XXV OCP	None
	Intel® Ethernet 25G 4P E810-XXV OCP	None
	Intel® Ethernet 25G 2P E810-XXV-k Mezz	None
	Intel® Ethernet 25G 2P E810-XXV Adapter	None
	Intel® Ethernet 25G 4P E810-XXV Adapter	None
	Intel® Ethernet Connection 25G 4P E823-C LOM	Supports ESXi, Windows Server, and Azure Stack HCI. Linux support: Supports only RHEL 9.2 and SLES 15 SP4.
Intel® Ethernet 700 Series	Intel® Ethernet 25G 2P XXV710 Adapter	None
	Intel® Ethernet 25G 2P XXV710 Mezz	None

**NOTE:**

- Devices based on the Intel® Ethernet Controller XXV710 (2x25 GbE) have a total hardware throughput limit for the entire device of ~96-97% of dual-port 25 GbE line rate in each direction for IPv4 TCP large packets (>1518 bytes) with an MTU size of 1500 bytes. For example, the total payload throughput is limited to ~45.5 Gb/s in each direction. Thus, while single port 25 GbE throughput is not impacted, total simultaneous dual port 25 GbE throughput is expected to be slightly lower than line rate.
- The first port of Intel® Ethernet 700 Series adapters will display the correct branding string. All other ports on the same device will display a generic branding string.

Supported 10 Gigabit Network Devices

Family	Device	OS Limitations
Intel® Ethernet 700 Series	Intel® Ethernet 10G 2P X710-k bNDC	None
	Intel® Ethernet 10G 4P X710-k bNDC	None
	Intel® Ethernet Converged Network Adapter X710	None
	Intel® Ethernet Converged Network Adapter X710-T	None
	Intel® Ethernet 10G 4P X710/I350 rNDC	None
	Intel® Ethernet 10G 4P X710 SFP+ rNDC	None
	Intel® Ethernet Server Adapter X710-DA2 for OCP	None
	Intel® Ethernet 10G 2P X710 OCP	None
	Intel® Ethernet 10G 4P X710 OCP	None
	Intel® Ethernet 10G 2P X710-T2L-t OCP	None
	Intel® Ethernet 10G 4P X710-T4L-t OCP	None
	Intel® Ethernet 10G 2P X710-T2L-t Adapter	None
	Intel® Ethernet 10G 4P X710-T4L-t Adapter	None
Intel® Ethernet 500 Series	Intel® Ethernet 10G 2P X550-t Adapter	None
	Intel® Ethernet 10G 4P X550 rNDC	None
	Intel® Ethernet 10G 4P X550/I350 rNDC	None



NOTE: The first port of Intel® Ethernet 700 Series adapters will display the correct branding string. All other ports on the same device will display a generic branding string.

Supported Gigabit Network Devices

Family	Device	OS Limitations
Intel® Ethernet 300 Series	Intel® Gigabit 2P I350-t Adapter	None
	Intel® Gigabit 4P I350-t Adapter	None
	Intel® Ethernet 1G 4P I350-t OCP	None
	Intel® Gigabit 4P X550/I350 rNDC	None
	Intel® Gigabit 4P I350-t rNDC	None
	Intel® Gigabit 4P I350-t Mezz	None
	Intel® Gigabit 4P X710/I350 rNDC	None
	Intel® Gigabit 4P I350 bNDC	None
	Intel® Gigabit 2P I350-t LOM	None
	Intel® Gigabit I350-t LOM	None
	Intel® Gigabit 2P I350 LOM	None

1.3.1 Non-tFRU Devices

Some adapters store information about the manufacturer or device in a chip called a field replaceable unit (FRU or tFRU). A few Intel Ethernet cards were redesigned to remove the tFRU and store this information in the vital product data (VPD) section of the NVM.

The following devices no longer have a FRU chip and now store the FRU data in the device's NVM:

- Intel® Ethernet 25G 2P XXV710 Adapter (DPN: N49FM, GY0MM)
- Intel® Ethernet Converged Network Adapter X710 (DPN: DRCGM, 51G03, G48TY)
- Intel® Gigabit 4P I350-t rNDC (DPN: MMW41)
- Intel® Gigabit 4P I350-t Adapter (DPN: Y10X9, NM8TT)



NOTE: There is no impact to the form, fit, or functionality of the device.

Updated firmware for these devices are available starting in the following releases:

- Release 19.5.17
- Release 20.0.20
- Release 20.5.17
- Release 21.5.x and newer

Other releases are not compatible with these redesigned cards.

1.4 Related Documentation

You can find additional resources, configuration guides, and technical documentation for Intel Ethernet products on the [Intel Resource & Documentation Center](#). Some documents may require a login.

1.4.1 User Guides for Specific Devices

Some adapters and devices have user guides with detailed configuration and setup information. You can access public versions of these documents in the [Intel Resource & Documentation Center](#).

Refer to the following documentation for advanced configuration. (Note that this documentation also applies for Dell-branded devices.)

Type of Document + Link	Affected Products
User guide	Intel® Ethernet 25G 4P E810-XXV-st Adapter Intel® Ethernet 25G 4P E810-XXV-stg Adapter (supports GNSS)
User guide	Intel® Ethernet 100G 2P E810-C-st Adapter Intel® Ethernet 100G 2P E810-C-stg Adapter (supports GNSS)

1.5 Customer Support

1.5.1 Web and Internet Sites

<http://support.dell.com/>

1.5.2 Customer Support Technicians

If the troubleshooting procedures in this document do not resolve the problem, please contact Dell, Inc. for technical assistance (refer to the "Getting Help" section in your system documentation).

Before you call...

You need to be at your computer with your software running and the product documentation at hand.

The technician may ask for the following:

- Your address and telephone number
- The name and model number of the product you are calling about
- The serial number and service tag of the product
- The names and version numbers of the software you are using to operate the product
- The name and version number of the operating system you are using
- The computer type (manufacturer and model number)
- Expansion boards or add-in cards in your computer
- The amount of memory in your computer


2. Installation

This chapter covers how to install Intel Ethernet adapters, drivers, and other software.

At a high level, installation involves the following steps, which are covered in more detail later in this chapter.

If you are installing a network adapter, follow this procedure from step 1.

If you are upgrading the driver software, start with step 4.

 **NOTE:** If you update the firmware, you must update the driver software to the same family version.

1. Review [system requirements](#).
2. [Insert the PCI Express Adapter, Mezzanine Card, or Network Daughter Card](#) into your server.
3. Carefully connect the network [copper cable\(s\)](#), [fiber cable\(s\)](#), or [direct attach cables](#)
4. Install the [network drivers and other software](#).
5. [Test the adapter](#).

2.1 Hardware Compatibility


Before installing the adapter, check your system for the following:

- The latest BIOS for your system
- One open PCI Express slot (see the [specifications of your card](#) for slot compatibility)

2.2 Installing the Adapter

2.2.1 Select the Correct Slot

One open PCI-Express slot, x4, x8, or x16, depending on your adapter.

 **NOTE:** Some systems have physical x8 PCI Express slots that actually only support lower speeds. Please check your system manual to identify the slot.

 **NOTE:** For information on identifying PCI Express slots that support your adapters, see your Dell system guide.

2.2.2 Insert the Adapter into the Computer

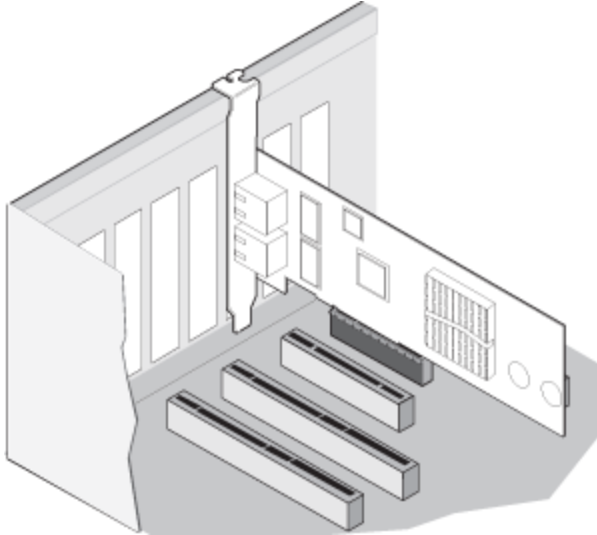
1. If your computer supports PCI Hot Plug, see your computer documentation for special installation instructions.
2. Turn off and unplug your computer. Then remove the cover.



CAUTION: Turn off and unplug the power before removing the computer's cover. Failure to do so could endanger you and may damage the adapter or computer.

3. Remove the cover bracket from an available slot.

4. Insert the adapter, pushing it into the slot until the adapter is firmly seated. You can install a smaller PCI Express adapter in a larger PCI Express slot.



CAUTION: Some PCI Express adapters may have a short connector, making them more fragile than PCI adapters. Excessive force could break the connector. Use caution when pressing the board in the slot.

5. Secure the adapter bracket with a screw, if required.
6. Replace the computer cover and plug in the power cord.
7. Power on the computer.

2.2.3 Install a Mezzanine Card in the Blade Server

See your server documentation for detailed instructions on how to install a Mezzanine card.

1. Turn off the blade server and pull it out of the chassis, then remove its cover.



CAUTION: Failure to turn off the blade server could endanger you and may damage the card or server.

2. Lift the locking lever and insert the card in an available, compatible mezzanine card socket. Push the card into the socket until it is firmly seated.



NOTE: A switch or pass-through module must be present on the same fabric as the card in the chassis to provide a physical connection. For example, if the mezzanine card is inserted in fabric B, a switch must also be present in fabric B of the chassis.

3. Repeat step 2 for each card you want to install.
4. Lower the locking lever until it clicks into place over the card or cards.
5. Replace the blade server cover and put the blade back into the server chassis.
6. Turn the power on.

2.2.4 Install a Network Daughter Card in a Server

See your server documentation for detailed instructions on how to install a bNDC or rNDC.

1. Turn off the server and then remove its cover.



CAUTION: Failure to turn off the server could endanger you and may damage the card or server.

2. Locate the Network Daughter Card connector in your server. See your server's documentation for details.
3. Press the Network Daughter Card into the connector.
4. Tighten the screws on the Network Daughter Card to secure it into place.
5. Replace the server's cover.

2.3 Connecting Network Cables

Connect the appropriate network cable, as described in the following sections.

2.3.1 Supported SFP+, SFP28, QSFP+, and QSFP28 Modules

To view the modules and cables compatible with your Ethernet device, log in to the [Dell Technologies Sales Portal](#) and type "PowerEdge Server Adapter Matrix" in the search box. Download and open the linked file to view the products compatible with your Ethernet device.



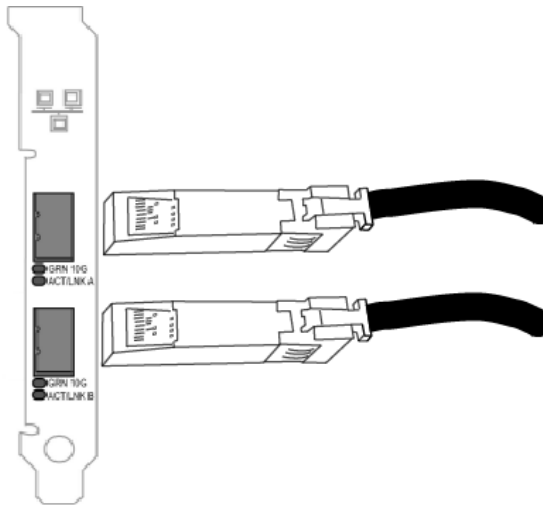
NOTE: Devices based on the Intel® Ethernet 710 Series do not support third-party modules.

Intel Ethernet Server Adapters only support Intel optics and/or all passive and active limiting direct attach cables that comply with SFF-8431 v4.1 and SFF-8472 v10.4 specifications.

Cabling Specification	Laser Wavelength	Connector Type	Cable Type	Max Cable Length
SR transceiver cabling specifications	850 nanometer (not visible)	LC or SC	Multi-mode fiber with 62.5µm core diameter	1 Gbps: 275 meters 10 Gbps (and faster): 33 meters
LR transceiver cabling specifications	1310 nanometer (not visible)	LC	Single-mode fiber with 9.0µm core diameter	10 kilometers

2.3.2 Connect the Direct Attach Cable

Insert the Direct Attach network cable as shown below.



The following table shows the types of direct attached cabling you can use.

Speed	Cable Type	Max Cable Length	Notes
100 Gbps	QSFP28 Direct Attach Cable	5 meters	
40 Gbps	SFP+ Direct Attached Cable (Twinaxial)	7 meters	
25 Gbps	SFP28 Direct Attached Cable (Twinaxial)	5 meters	For optimal performance, must use CA-25G-L with RS-FEC and 25GBASE-C.
10 Gbps	SFP+ Direct Attached Cable (Twinaxial)	7 meters	

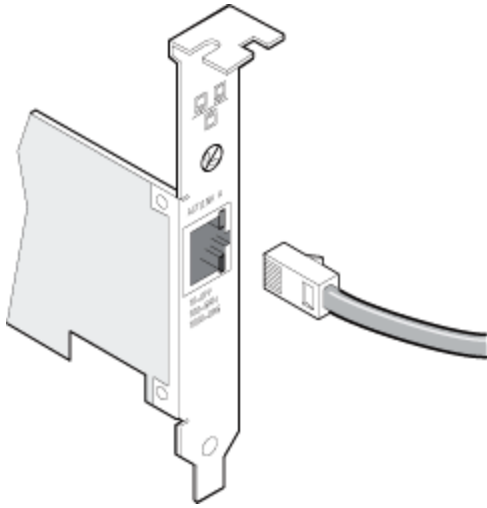
2.3.3 Port Breakout Cables

Some Intel Ethernet 800 Series devices support the use of port breakout cables. The device must have a 100 Gbps port to support this feature. The following devices support port breakout:

- Intel® Ethernet 100G 2P E810-C Adapter
- Intel® Ethernet 100G 2P E810-C-stg Adapter
- Intel® Ethernet 100G 2P E810-C-st Adapter
- Intel® Ethernet Connection 25G 4P E823-C LOM

2.3.4 Connect the RJ-45 Network Cable

Connect the RJ-45 network cable as shown:



The following table shows the maximum lengths for each cable type at a given transmission speed.

Speed	Category 5	Category 6	Category 6a	Category 7
1 Gbps	100m	100m	100m	100m
10 Gbps	NA	55m	100m	100m
25 Gbps	NA	NA	NA	50m
40 Gbps	NA	NA	NA	50m



CAUTION: If using less than 4-pair cabling, you must manually configure the speed and duplex setting of the adapter and the link partner. In addition, with 2- and 3-pair cabling the adapter can only achieve speeds of up to 100Mbps.

In all cases:

- The adapter must be connected to a compatible link partner, preferably set to auto-negotiate speed and duplex for Intel gigabit adapters.
- Intel Gigabit and 10 Gigabit Server Adapters using copper connections automatically accommodate either MDI or MDI-X connections. The auto-MDI-X feature of Intel gigabit copper adapters allows you to directly connect two adapters without using a cross-over cable.

2.4 Install Drivers and Software

2.4.1 On Windows Operating Systems

You must have administrative rights to the operating system to install the drivers and software.

1. Download the latest drivers from the [support website](#) and transfer them to the system.
2. If the Found New Hardware Wizard screen is displayed, click **Cancel**.
3. Double-click the downloaded file.

4. Select **Install** from the Dell Update Package screen.
5. Follow the prompts in the install wizard.

Refer to the following for more detailed information :

- "About Intel PROSet®" on page 13
- "Microsoft* Windows* Driver and Software Installation and Configuration" on page 65

2.4.2 On Linux

Refer to "Building and Installation" on page 73 for more specific information on installing drivers on Linux.

2.4.2.1 Installing Linux Drivers from Source Code

1. Download and expand the driver tar file.
2. Compile the driver module.
3. Install the module using the modprobe command.
4. Assign an IP address using the ifconfig command.

2.4.2.2 Installing Linux Drivers from RPMs

1. Download and expand the driver tar file.
2. Install the driver using the rpm command or another software management tool appropriate for your distribution.

3. About Intel PROSet®

Intel PROSet is a suite of software tools to configure Intel Ethernet devices on Microsoft Windows operating systems. Intel PROSet software includes the following:

Component	Description
Intel® PROSet Adapter Configuration Utility (Intel® PROSet ACU)	A graphical user interface that allows you to configure and manage supported Intel Ethernet adapters.
Intel® PROSet for Windows PowerShell* software	Software that contains several cmdlets that allow you to configure and manage the Intel Ethernet adapters and devices present in your system.

For more information on installing or using Intel PROSet, refer to the following:

- "Compatibility Notes for Intel Ethernet Cmdlets and Intel PROSet" below
- "Configuring Device Features in Microsoft Windows" on page 15
- "Installing Windows Drivers and Software" on page 65



NOTES:

- You must have administrator rights to install or use Intel PROSet.
- Intel PROSet requires the latest driver and software package for your Intel Ethernet devices.

3.1 About Intel® Ethernet Cmdlets

In addition to [Intel PROSet](#), Intel provides Ethernet cmdlets for Windows PowerShell to:

- Display information about Intel Ethernet devices in the system
- Configure device settings
- Configure and gather [firmware logs](#) for debugging supported devices
- Blink the LED on the specified port of an Intel Ethernet device for a defined number of seconds

These Ethernet cmdlets do not require Intel PROSet to be installed on the system. For more information on installing or using Intel Ethernet cmdlets, refer to the following:

- "Compatibility Notes for Intel Ethernet Cmdlets and Intel PROSet" below
- "Installing Intel Ethernet Cmdlets" on the next page
- "Configuring with Windows PowerShell" on page 15
- "Firmware Logging" on page 178

3.2 Compatibility Notes for Intel Ethernet Cmdlets and Intel PROSet

Note the following compatibility requirements for [Intel Ethernet cmdlets](#) and [Intel PROSet](#). The installer will automatically detect and install the components compatible with your operating system.

3.2.1 Device Compatibility

The following devices either do not support Intel Ethernet cmdlets, Intel PROSet, or some of its components.

Device	Ethernet Cmdlets	Intel PROSet	
	No Support	No Support ¹	PowerShell Only ²
Any platform with a System on a Chip (SoC) processor	Varies		X ³
Intel® Ethernet Connection E823-C			X ²
Intel® Ethernet 500 Series and lower	X		
Notes: <ol style="list-style-type: none"> 1. No support for Intel PROSet or any of its components. 2. Supports only Intel PROSet for Windows PowerShell software. Does not support Intel PROSet ACU. 3. This includes a platform with either a server controller (designated by an initial E or X, such as X552 or X722) or both a server and client controller (designated by an initial I, such as I218). 4. This device or family does not support any Microsoft operating systems. As a result, it also does not support Intel Ethernet cmdlets and/or Intel PROSet. 			

3.2.2 Operating System Compatibility

The following table lists compatible operating systems for Intel Ethernet cmdlets, Intel PROSet, or its components. **Not all OS versions listed in the following table are supported in the newest releases of Intel Ethernet software;** refer to "Supported Operating Systems" on page 2 for currently supported versions.

Component	Operating System Compatibility
Intel Ethernet cmdlets	Any supported version of Microsoft Windows Server Microsoft Azure Stack HCI Microsoft Windows PowerShell version 5.1 and later
Intel PROSet for Windows PowerShell software	Any supported version of Microsoft Windows Server Microsoft Azure Stack HCI
Intel PROSet Adapter Configuration Utility	Microsoft Windows Server 2019 and later

3.3 Installing Intel Ethernet Cmdlets

Intel Ethernet cmdlets are automatically installed along with the base drivers. Refer to "Installing Windows Drivers and Software" on page 65 for detailed instructions.

After installation, the cmdlets are installed to `C:\Users\.`

After installation, you can run the cmdlets at the PowerShell prompt without manually importing the module.

Refer to the following for more information:

- The readme.txt file saved to C:\Program Files\Intel\EthernetCmdlets after installation
- The cmdlet help in PowerShell
- In this user guide:
 - "Configuring with Windows PowerShell" below
 - "Firmware Logging" on page 178



NOTE: Intel Ethernet cmdlets will be deleted if you uninstall Intel Ethernet drivers via Add/Remove Programs.

3.4 Installing Intel PROSet

Intel PROSet is automatically installed along with the base drivers. Refer to "Installing Windows Drivers and Software" on page 65 for detailed instructions.

3.5 Configuring Device Features in Microsoft Windows

This section describes how to use Intel Ethernet cmdlets or Intel PROSet to configure device features on supported Windows operating systems.

For an overview or installation information, refer to the following:

- "About Intel PROSet®" on page 13
- "About Intel® Ethernet Cmdlets" on page 13
- "Compatibility Notes for Intel Ethernet Cmdlets and Intel PROSet" on page 13
- "Installing Windows Drivers and Software" on page 65

3.5.1 Configuring with Windows PowerShell

You can configure and manage the Intel Ethernet devices present in your system using the following:

- [Intel Ethernet cmdlets](#)
- Intel PROSet for Windows PowerShell software



NOTES: Refer to "Compatibility Notes for Intel Ethernet Cmdlets and Intel PROSet" on page 13 for information on support limitations.

Module Names

The following table lists the module name in Windows PowerShell for each component.

Component	Module Name	Additional Information
Intel Ethernet cmdlets	IntelEthernetCmdlets	"Installing Intel Ethernet Cmdlets" on the previous page "Firmware Logging" on page 178
Intel PROSet for Windows PowerShell software	IntelNetCmdlets	"Installing Intel PROSet" above

Importing New Cmdlets

After installing Intel PROSet, use the `Import-Module` cmdlet to import the new cmdlets. You may need to restart Windows PowerShell to access the newly installed cmdlets. Refer to "Module Names" on the previous page for the available cmdlet modules.

To use the `Import-Module` cmdlet, you must specify the path. For example:

```
PS c:\> Import-Module -Name "C:\Program Files\Intel\Wired Networking\IntelNetCmdlets"
```



NOTE:

- If you include a trailing backslash ("\") at the end of the `Import-Module` command, the import operation will fail.
- If you encounter issues with Intel Ethernet cmdlets, you may need to manually import the module using the instructions provided above.

See Microsoft TechNet for more information about the `Import-Module` cmdlet.

Changing Intel Ethernet Settings via Microsoft Windows PowerShell

You can use Windows PowerShell software to change most Intel Ethernet settings.

To configure Intel Ethernet device features using Windows PowerShell software, follow these general steps:

1. Install Intel PROSet or Intel Ethernet cmdlets, if you haven't already. See the following for more information:
 - "Installing Intel Ethernet Cmdlets" on page 14
 - "Installing Intel PROSet" on the previous page
2. Open PowerShell.
3. At the PowerShell prompt, run your desired cmdlet.

Help Information for PowerShell Cmdlets

To get help information for both Intel Ethernet cmdlets and Intel PROSet for Windows PowerShell software:

- For a complete list of the cmdlets and their descriptions, type the following at the Windows PowerShell prompt. Refer to "Module Names" on the previous page for the available cmdlet modules.

```
PS C:\> get-help <module name>
```

- For detailed usage information for each cmdlet (including examples), type the following at the Windows PowerShell prompt:


```
PS C:\> get-help <cmdlet_name> -full
```

- To show only examples for a cmdlet, type the following at the Windows PowerShell prompt:

```
PS C:\> get-help <cmdlet_name> -examples
```


- To use the Minihelp property for any cmdlet in the module, append `| Select Minihelp`. For example:

```
PS C:\> Get-IntelNetAdapterSetting -Name "<adapter_name>" -  
RegistryKeyword *RSS | Select Minihelp
```

 **NOTE:** Online help (`get-help -online`) is not supported.

Additional Notes for Intel PROSet for Windows PowerShell Software

- IntelNetCmdlets are digitally signed. Microsoft Windows operating systems check digital signatures online. Depending on your internet connection, this may result in a delay before any cmdlet operation (including `get-help`). If you have not already done so, make sure you use `Import-Module` to import the IntelNetCmdlets.
- The `Get-IntelNetAdapterStatus -Status General` cmdlet may report the status "Link Up - This device is not linked at its maximum capable speed". In that case, if your device is set to auto-negotiate, you can adjust the speed of the device's link partner to the device's maximum speed. If the device is not set to auto-negotiate, you can adjust the device's speed manually, but you must ensure the link partner is set at the same speed.

3.5.2 Configuring with Intel PROSet Adapter Configuration Utility

The Intel PROSet Adapter Configuration Utility (Intel PROSet ACU) is a graphical user interface that allows you to configure and manage supported Intel Ethernet Adapters.

 **NOTE:** Refer to "Compatibility Notes for Intel Ethernet Cmdlets and Intel PROSet" on page 13 for information on support limitations.

To configure Intel Ethernet device features using Intel PROSet ACU, follow these general steps:

1. Select an adapter in the Adapter Selection panel.
2. Select a setting to configure from the Adapter Settings panel.
3. Select or enter the desired value(s) for the selected setting.
4. Click the "Apply Changes" button.

3.5.3 Changing Intel Ethernet Settings Under Windows Server Core

You can use the Intel PROSet for Windows PowerShell software or Intel Ethernet cmdlets to change most Intel Ethernet settings under Windows Server Core. Please refer to their cmdlet help in PowerShell for more information.

For iSCSI Crash Dump configuration, use the Intel PROSet for Windows PowerShell software and refer to the `aboutIntelNetCmdlets.help.txt` help file. iSCSI Crash Dump configuration is not supported in Intel Ethernet cmdlets.

4. Device Features

This chapter describes the features available on Intel Ethernet devices. Major features are organized alphabetically.



NOTE:

- Available settings are dependent on your device and operating system. Not all settings are available on every device/OS combination.
- Some features in this section refer to Intel PROSet, Intel PROSet Adapter Configuration Utility (Intel PROSet ACU), or Intel PROSet for Windows PowerShell* software. Refer to "About Intel PROSet®" on page 13 for more information.

4.1 Adaptive Inter-Frame Spacing

Compensates for excessive Ethernet packet collisions on the network.

The default setting works best for most computers and networks. By enabling this feature, the network adapter dynamically adapts to the network traffic conditions. However, in some rare cases you might obtain better performance by disabling this feature. This setting forces a static gap between packets.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Enabled • Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Adaptive Inter-Frame Spacing" -DisplayValue "Enabled"
```

4.2 Data Center Bridging (DCB)

Data Center Bridging (DCB) is a collection of standards-based extensions to classical Ethernet. It provides a lossless data center transport layer that enables the convergence of LANs and SANs onto a single unified fabric.

Furthermore, DCB is a configuration Quality of Service implementation in hardware. It uses the VLAN priority tag (802.1p) to filter traffic. That means that there are 8 different priorities that traffic can be filtered into. It also enables priority flow control (802.1Qbb) which can limit or eliminate the number of dropped packets during network stress. Bandwidth can be allocated to each of these priorities, which is enforced at the hardware level (802.1Qaz).

DCB includes the following capabilities:

- Priority-based flow control (PFC; IEEE 802.1Qbb)
- Enhanced transmission selection (ETS; IEEE 802.1Qaz)
- Congestion notification (CN)
- Extensions to the Link Layer Discovery Protocol (LLDP) standard (IEEE 802.1AB) that enable Data Center Bridging Capability Exchange Protocol (DCBX)

Adapter firmware implements LLDP and DCBX protocol agents as per 802.1AB and 802.1Qaz respectively.

There are two supported versions of DCBX.

- CEE Version: The specification can be found as a link within the following document: <http://www.ieee802.org/1/files/public/docs2008/dcb-baseline-contributions-1108-v1.01.pdf>
- IEEE Version: The specification can be found as a link within the following document: <https://standards.ieee.org/findstds/standard/802.1Qaz-2011.html>



NOTE: The OS DCBX stack defaults to the CEE version of DCBX, and if a peer is transmitting IEEE TLVs, it will automatically transition to the IEEE version.

For more information on DCB, including the DCB Capability Exchange Protocol Specification, go to <http://www.ieee802.org/1/pages/dcbbridges.html>

Supported Devices

All devices capable of 10GbE or faster support DCB.

4.2.1 DCB for Windows Configuration

To change this setting in Intel PROSet

This setting is found in the Data Center panel in Intel PROSet Adapter Configuration Utility.

You can use Intel PROSet to perform the following tasks:

- **Display Status:**
 - Enhanced Transmission Selection
 - Priority Flow Control

Non-operational status: If the Status indicator shows that DCB is non-operational, there may be a number of possible reasons:

- DCB is not enabled - select the checkbox to enable DCB.
- One or more of the DCB features is in a non-operational state.

A non-operational status is most likely to occur when **Use Switch Settings** is selected or **Using Advanced Settings** is active. This is generally a result of one or more of the DCB features not getting successfully exchanged with the switch. Possible problems include:

- One of the features is not supported by the switch.
 - The switch is not advertising the feature.
 - The switch or host has disabled the feature (this would be an advanced setting for the host).
- Disable/enable DCB
 - Troubleshooting information

 **NOTES:**

- On X710 based devices running Microsoft Windows, DCB is only supported on firmware version 17.0.12 and newer. Older NVM versions must be updated before the adapter is capable of DCB support in Windows.
- On systems running a Microsoft Windows Server operating system, enabling *QoS/priority flow control will disable link level flow control.
- If *QOS/DCB is not available, it may be for one of the following reasons:
 - The Firmware LLDP (FW-LLDP) agent was disabled from a pre-boot environment (typically UEFI).
 - This device is based on the Intel® Ethernet Controller X710 and the current link speed is 2.5 Gbps or 5 Gbps.

4.2.1.1 Hyper-V (DCB and VMQ)

 **NOTE:** Configuring a device in the VMQ + DCB mode reduces the number of VMQs available for guest OSes.

4.2.2 DCB for Linux

Intel Ethernet drivers support firmware-based or software-based DCBX in Linux, depending on the underlying PF device. The following table summarizes DCBX support by driver.

Linux Driver	Firmware-Based DCBX	Software-Based DCBX
ice	Supported	Supported
i40e	Supported	Supported
ixgbe	Not supported	Supported

In **firmware-based** mode, firmware intercepts all LLDP traffic and handles DCBX negotiation transparently for the user. In this mode, the adapter operates in "willing" DCBX mode, receiving DCB settings from the link partner (typically a switch). The local user can only query the negotiated DCB configuration.


In **software-based** mode, LLDP traffic is forwarded to the network stack and user space, where a software agent can handle it. In this mode, the adapter can operate in either "willing" or "nonwilling" DCBX mode and DCB configuration can be both queried and set locally. Software-based mode requires the FW-based LLDP Agent to be disabled, if supported.

 **NOTES:**

- Only one LLDP/DCBX agent can be active on a single interface at a time.
- Software-based and firmware-based DCBX modes are mutually exclusive.
- When the firmware DCBX agent is active, software agents will not be able to receive or transmit LLDP frames. See "Firmware Link Layer Discovery Protocol (FW-LLDP)" on page 23, as well as the Linux driver readme in your installation, for information on enabling or disabling the FW-LLDP agent.
- In software-based DCBX mode, you can configure DCB parameters using software LLDP/DCBX agents that interface with the Linux kernel's DCB Netlink API. We recommend using OpenLLDP as the DCBX agent when running in software mode. For more information, see the OpenLLDP man pages and <https://github.com/intel/openlldp>.
- For information on configuring DCBX parameters on a switch, please consult the switch manufacturer's documentation.

4.2.3 iSCSI Over DCB

Intel® Ethernet adapters support iSCSI software initiators that are native to the underlying operating system. Data Center Bridging is most often configured at the switch. If the switch is not DCB capable, the DCB handshake will fail but the iSCSI connection will not be lost.

 **NOTE:** DCB does not install in a VM. iSCSI over DCB is only supported in the base OS. An iSCSI initiator running in a VM will not benefit from DCB ethernet enhancements.


4.2.3.1 Microsoft Windows Configuration

iSCSI installation includes the installation of the iSCSI DCB Agent (iscsidcb.exe) user mode service. The Microsoft iSCSI Software Initiator enables the connection of a Windows host to an external iSCSI storage array using an Intel Ethernet adapter. Please consult your operating system documentation for configuration details.

To change this setting in Intel PROSet

This setting is found in the Data Center panel in Intel PROSet Adapter Configuration Utility.

This setting provides feedback as to the DCB state, operational or non-operational, as well as providing additional details should it be non-operational.

 **NOTE:** On Microsoft Windows Server operating systems, if you configure Priority using IEEE, the iSCSI policy may not be created automatically. To create the iSCSI policy manually, use Powershell and type:

```
New-NetQosPolicy -Name "UP4" -PriorityValue 8021 Action 4 -iSCSI
```

4.2.3.2 Linux Configuration

In the case of Open Source distributions, virtually all distributions include support for an Open iSCSI Software Initiator and Intel® Ethernet adapters will support them. Please consult your distribution documentation for additional configuration details on their particular Open iSCSI initiator.

Intel® 82599-based adapters support iSCSI within a Data Center Bridging cloud. Used in conjunction with switches and targets that support the iSCSI/DCB application TLV, this solution can provide guaranteed minimum bandwidth for iSCSI traffic between the host and target. This solution enables storage administrators to segment iSCSI traffic from LAN traffic. Previously, iSCSI traffic within a DCB supported environment was treated as LAN traffic by switch vendors. Please consult your switch and target vendors to ensure that they support the iSCSI/DCB application TLV.

4.3 Direct Memory Access (DMA) Coalescing

DMA (Direct Memory Access) allows the network device to move packet data directly to the system's memory, reducing CPU utilization. However, the frequency and random intervals at which packets arrive do not allow the system to enter a lower power state. DMA Coalescing allows the NIC to collect packets before it initiates a DMA event. This may increase network latency but also increases the chances that the system will consume less energy. Adapters and network devices based on the Intel® Ethernet Controller I350 (and later controllers) support DMA Coalescing.

Higher DMA Coalescing values result in more energy saved but may increase your system's network latency. If you enable DMA Coalescing, you should also set the Interrupt Moderation Rate to 'Minimal'. This minimizes the latency impact imposed by DMA Coalescing and results in better peak network throughput performance. You must enable DMA Coalescing on all active ports in the system. You may not gain any energy savings if it is enabled only on some of the ports in your system. There are also several BIOS, platform, and application settings that will affect your potential energy savings. A white paper containing information on how to best configure your platform is available [on the Intel website](#).

To change this setting in Intel PROSet

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "DMA Coalescing" -  
DisplayValue "Enabled"
```

4.4 Dynamic Device Personalization (DDP)

Adapters based on the Intel® Ethernet 800 Series require a Dynamic Device Personalization (DDP) package file to enable advanced features (such as dynamic tunneling, Intel Ethernet Flow Director, RSS, and ADQ). DDP allows you to change the packet processing pipeline of a device by applying a profile package to the device at runtime. Profiles can be used to, for example, add support for new protocols, change existing protocols, or change default settings. DDP profiles can also be rolled back without rebooting the system.

The DDP package loads during device initialization or driver runtime, depending on the operating system. The driver checks to see if the DDP package is present and compatible. If this file exists, the driver will load it into the device. If not, the driver will go into Safe Mode where it will use the configuration contained in the device's NVM.

Safe Mode disables advanced and performance features, and supports only basic traffic and minimal functionality, such as updating the NVM or downloading a new driver or DDP package. For more information, see "Safe Mode" on page 188.

A general-purpose, default DDP package is automatically installed with all supported Intel Ethernet 800 Series drivers on supported operating systems. Additional DDP packages are available to address needs for specific market segments or targeted solutions.

Refer to the [Intel® Ethernet Controller E810 Dynamic Device Personalization \(DDP\) Technology Guide](#) for more information on configuring DDP.

NOTES:

- If you are using DPDK, see the DPDK documentation for installation instructions and more information.
- In ESXi:
 - Support for DDP packages for specific market segments requires the following:
 - Driver: icen 1.9.1.x or higher
 - Tool: intnet 1.8.3.x or higher
 - Use esxcli to load and unload DDP packages for specific market segments during driver runtime.
 - A package update is not persistent between device resets or system reboots.

4.5 Firmware Link Layer Discovery Protocol (FW-LLDP)

Devices based on the Intel® Ethernet 800 and 700 Series use a Link Layer Discovery Protocol (LLDP) agent that runs in the firmware. When it is running, it prevents the operating system and applications from receiving LLDP traffic from the network adapter.

- The FW-LLDP setting is per port and persists across reboots.
- The FW-LLDP Agent is required for DCB to function.

Adapters Based on the Intel® Ethernet 800 Series

FW-LLDP is disabled in NVM by default. To enable/disable the FW-LLDP Agent:

- **Linux:** Use ethtool to persistently set or show the fw-lldp-agent private flag.
- **ESX:** Use the esxcli command to persistently set or get the fw-lldp-agent setting.
- **Microsoft Windows:** The base driver does not persistently change FW-LLDP. Use the LLDP Agent attribute in UEFI HII to persistently change the FW-LLDP setting. If you enable DCB when FW-LLDP is disabled, the base driver temporarily starts the LLDP Agent while DCB functionality is enabled.

Adapters Based on the Intel® Ethernet 700 Series

FW-LLDP is enabled in NVM by default. To enable/disable the FW-LLDP Agent:

- **Linux:** Use ethtool to set or show the disable-fw-lldp private flag.
- **ESX:** Use the esxcfg-module command to set or get the LLDP module parameter.
- **Microsoft Windows:** Use the LLDP Agent attribute in UEFI HII to change the FW-LLDP setting. Note: You must enable the UEFI HII "LLDP AGENT" attribute for the FW-LLDP setting to take effect. If "LLDP AGENT" is set to disabled in UEFI HII, you cannot enable FW-LLDP from the OS.
- You must enable the LLDP Agent from UEFI HII to use DCB.

4.6 Firmware Logs and Advanced Debugging

Intel Ethernet 800 Series devices support the ability to generate firmware logs or other information, to debug issues with Customer Support. Refer to the following for more information:

- "Firmware Logging" on page 178
- "Debug Dump" on page 182
- "Health Status Messages" on page 187

4.7 Forward Error Correction (FEC) Mode

Allows you to set the Forward Error Correction (FEC) mode. FEC improves link stability, but increases latency. Many high quality optics, direct attach cables, and backplane channels provide a stable link without FEC.

The driver allows you to set the following FEC Modes:

- Auto FEC - Sets the FEC Mode based on the capabilities of the attached cable.
- CL108 RS-FEC - Selects only RS-FEC ability and request capabilities.
- CL74 FC-FEC/BASE-R - Selects only BASE-R ability and request capabilities.
- No FEC - Disables FEC.



NOTES:

- For devices to benefit from this feature, link partners must have FEC enabled.
- Intel® Ethernet 800 Series devices only enable Forward Error Correction (FEC) configurations that are supported by the connected media and which are expected to yield healthy Bit Error Rate (BER) connections.
 - If you enable the registry keyword `AllowNoFECModulesInAuto`, Auto FEC negotiation will include 'No FEC' in case your link partner does not have FEC enabled or is not FEC capable.
 - To change this setting in Windows PowerShell, use the `Set-IntelNetAdapterSetting` cmdlet. For example:


```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -RegistryKeyword AllowNoFECModulesInAuto -RegistryValue 1
```
 - To change this setting in Linux, use `ethtool`. For example:


```
# ethtool --set-priv-flags <ethX> allow-no-fec-modules-in-auto on
```
- If you are having link issues (including no link) at link speeds faster than 10 Gbps, check your switch configuration and/or specifications. Many optical connections and direct attach cables require RS-FEC for connection speeds faster than 10 Gbps. One of the following may resolve the issue:
 - Configure your switch to use RS-FEC mode.
 - Specify a 10 Gbps, or slower, link speed connection.
 - If you are attempting to connect at 25 Gbps, try using an SFP28 CA-S or CS-N Direct Attach cable. These cables do not require RS-FEC.
 - If your switch does not support RS-FEC mode, check with your switch vendor for the availability of a SW or FW upgrade.

To change this setting in Intel PROSet

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the `Set-IntelNetAdapterSetting` cmdlet. For example:


```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "FEC Mode" -DisplayValue "Auto FEC"
```

4.8 Flow Control

Enables adapters to more effectively regulate traffic. Adapters generate flow control frames when their receive queues reach a pre-defined limit. Generating flow control frames signals the transmitter to slow transmission. Adapters respond to flow control frames by pausing packet transmission for the time specified in the flow control frame.

By enabling adapters to adjust packet transmission, flow control helps prevent dropped packets. You may improve RDMA performance by enabling flow control on all nodes and on the switch they are connected to.



NOTES:

- For adapters to benefit from this feature, link partners must support flow control frames.
- On systems running a Microsoft Windows Server operating system, enabling *QoS/priority flow control will disable link level flow control.
- Some devices support Auto Negotiation. Selecting this will cause the device to advertise the value stored in its NVM (usually "Disabled").

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Disabled • RX Enabled • TX Enabled • RX & TX Enabled • Auto Negotiation (only available on some adapters)

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Flow Control" -DisplayValue "Rx Enabled"
```

4.9 Gigabit PHY Mode

Determines whether the adapter or link partner is designated as the primary. The other device is designated as the secondary. By default, the IEEE 802.3ab specification defines how conflicts are handled. Multi-port devices such as switches have higher priority over single port devices and are assigned as the primary. If both devices are multi-port devices, the one with higher seed bits becomes the primary. This default setting is called "Hardware Default."



NOTE: In most scenarios, we recommended the default value of this feature.

Setting this to any value other than "Auto Detect" overrides the hardware default.

To change this setting in Intel PROSet

Default	Auto Detect
Range	<ul style="list-style-type: none"> • Force Primary Mode • Force Secondary Mode • Auto Detect



NOTE: When Gigabit PHY Mode is forced to Primary mode on both the Intel adapter and its link partner, the link speed obtained by the Intel adapter may be lower than expected or link may not be established.

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Gigabit PHY Mode" -
DisplayValue "Auto Detect"
```

4.10 Interrupt Moderation Rate

Sets the Interrupt Throttle Rate (ITR). This setting moderates the rate at which Transmit and Receive interrupts are generated.

When an event such as packet receiving occurs, the adapter generates an interrupt. The interrupt interrupts the CPU and any application running at the time, and calls on the driver to handle the packet. At greater link speeds, more interrupts are created, and CPU rates also increase. This results in poor system performance. When you use a higher ITR setting, the interrupt rate is lower and the result is better CPU performance.



NOTE: A higher ITR rate also means that the driver has more latency in handling packets. If the adapter is handling many small packets, it is better to lower the ITR so that the driver can be more responsive to incoming and outgoing packets.

Altering this setting may improve traffic throughput for certain network and system configurations, however the default setting is optimal for common network and system configurations. Do not change this setting without verifying that the desired change will have a positive effect on network performance.

To change this setting in Intel PROSet

Default	Adaptive
Range	<ul style="list-style-type: none"> • Adaptive • Extreme • High • Medium • Low • Minimal • Off

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Interrupt Moderation Rate" -DisplayValue "Adaptive"
```

4.11 Jumbo Frames

Enables or disables Jumbo Packet capability. The standard Ethernet frame size is about 1514 bytes, while Jumbo Packets are larger than this. Jumbo Packets can increase throughput and decrease CPU utilization. However, additional latency may be introduced.

Enable Jumbo Packets only if ALL devices across the network support them and are configured to use the same frame size. When setting up Jumbo Packets on other network devices, be aware that network devices calculate Jumbo Packet sizes differently. Some devices include the frame size in the header information while others do not. Intel adapters do not include frame size in the header information.

Restrictions

- Supported protocols are limited to IP (TCP, UDP).
- Jumbo frames require compatible switch connections that forward Jumbo Frames. Contact your switch vendor for more information.
- When standard-sized Ethernet frames (64 to 1518 bytes) are used, there is no benefit to configuring Jumbo Frames.
- The Jumbo Packets setting on the switch must be set to at least 8 bytes larger than the adapter setting for Microsoft Windows operating systems, and at least 22 bytes larger for all other operating systems.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Disabled (1514 bytes) • 4088 Bytes • 9014 Bytes <p>(Set the switch 4 bytes higher for CRC, plus 4 bytes if using VLANs.)</p>



NOTES:

- End-to-end hardware must support this capability; otherwise, packets will be dropped.
- Intel adapters that support Jumbo Packets have a frame size limit of 9238 bytes, with a corresponding MTU size limit of 9216 bytes.

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Jumbo Packet" -DisplayValue "4088 Bytes"
```

4.12 Link State on Interface Down

Sets if link is enabled or disabled when the interface is brought down. If this is set to **Disabled** and you bring an interface down (using an administrative tool, or in another way), then the port will lose link. This allows an attached switch to detect that the interface is no longer up. However, if Wake on LAN or manageability is enabled on this port, link will remain up.

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> • Enabled • Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Link State on Interface Down" -DisplayValue "Enabled"
```

4.13 Total Port Shutdown

This feature allows a port be completely shut down from the host OS/driver based on a user-configurable setting in the UEFI HII menu. See Dell's iDRAC documentation for more information on this feature.

The following devices support Total Port Shutdown:

- Intel Ethernet 810 and 820 Series
- Some Intel Ethernet 700 Series devices:
 - Intel® Ethernet 10G 2P X710 OCP
 - Intel® Ethernet 10G 4P X710 OCP
 - Intel® Ethernet 10G 4P X710 SFP+ rNDC
 - Intel® Ethernet 10G 2P X710-T2L-t Adapter
 - Intel® Ethernet 10G 4P X710-T4L-t Adapter
 - Intel® Ethernet 10G 2P X710-T2L-t OCP
 - Intel® Ethernet 10G 4P X710-T4L-t OCP
 - Intel® Ethernet 10G 4P X710/I350 rNDC
 - Intel® Ethernet 25G 2P XXV710 Adapter
 - Intel® Ethernet 40G 2P XL710 QSFP+ rNDC
 - Intel® Ethernet Server Adapter X710-DA2 for OCP
 - Intel® Ethernet Converged Network Adapter X710
 - Intel® Ethernet Converged Network Adapter X710-T
 - Intel® Ethernet Converged Network Adapter XL710-Q2



NOTE: Devices based on the Intel Ethernet 500 and 300 Series do not support this feature.

4.14 Locally Administered Address

Overrides the initial MAC address with a user-assigned MAC address. To enter a new network address, type a 12-digit hexadecimal number in this box.

To change this setting in Intel PROSet

Default	None
Range	<p>0000 0000 0001 - FFFF FFFF FFFD</p> <p>Exceptions:</p> <ul style="list-style-type: none"> Do not use a multicast address (Least Significant Bit of the high byte = 1). For example, in the address 0Y123456789A, "Y" cannot be an odd number. (Y must be 0, 2, 4, 6, 8, A, C, or E.) Do not use all zeros or all Fs. <p>If you do not enter an address, the address is the original network address of the adapter.</p> <p>For example,</p> <p style="padding-left: 40px;">Multicast: 0123 4567 8999 Broadcast: FFFF FFFF FFFF Unicast (legal): 0070 4567 8999</p>

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Locally Administered Address" -DisplayValue "<desired address>"
```

4.15 Log Link State Event

This setting is used to enable/disable the logging of link state changes. If enabled, a link up change event or a link down change event generates a message that is displayed in the system event logger. This message contains the link's speed and duplex. Administrators view the event message from the system event log.

The following events are logged.

- The link is up.
- The link is down.
- Mismatch in duplex.
- Spanning Tree Protocol detected.

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> Enabled Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Log Link State Event" -
DisplayValue "Enabled"
```

4.16 Low Latency Interrupts

LLI enables the network device to bypass the configured interrupt moderation scheme based on the type of data being received. It configures which arriving TCP packets trigger an immediate interrupt, enabling the system to handle the packet more quickly. Reduced data latency enables some applications to gain faster access to network data.



NOTE: When LLI is enabled, system CPU utilization may increase.

LLI can be used for data packets containing a TCP PSH flag in the header or for specified TCP ports.

- **Packets with TCP PSH Flag** - Any incoming packet with the TCP PSH flag will trigger an immediate interrupt. The PSH flag is set by the sending device.
- **TCP Ports** - Every packet received on the specified ports will trigger an immediate interrupt. Up to eight ports may be specified.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Disabled • PSH Flag-Based • Port-Based

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Low Latency Interrupts" -
DisplayValue "Port-Based"
```

4.17 Malicious Driver Detection (MDD) for VFs

Some Intel Ethernet devices use Malicious Driver Detection (MDD) to detect malicious traffic from the VF and disable Tx/Rx queues or drop the offending packet until a VF driver reset occurs. You can view MDD messages in the PF's event log.

- If the device supports automatic VF resets and the driver detects an MDD event on the receive path, the PF will automatically reset the VF and reenables queues. If automatic VF resets are disabled, the PF will not automatically reset the VF when it detects MDD events. See the table below for supported MDD features.
- If the PF driver logs MDD events from the VF, confirm that the correct VF driver is installed.
- To restore functionality, you can manually reload the VF or VM or, if supported by the device, enable automatic VF resets.

The following table shows MDD capabilities by device family.

Feature	Intel Ethernet 800 Series	Intel Ethernet 700 Series	Intel Ethernet 500 Series	Intel I350 Gigabit Network Connection
Automatically resets the VF and re-enables queues after MDD events	If enabled	If enabled	Yes	Yes
Can disable automatic VF reset after MDD events	Yes	Yes	No	No

4.17.1 MDD Auto Reset VFs

Automatically resets the virtual machine immediately after the adapter detects a Malicious Driver Detection (MDD) event on the receive path.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Disabled • Enabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "MDD Auto Reset VFs" -
DisplayValue "Enabled"
```

4.18 Max Number of RSS Queues Per Vport

Sets the maximum number of Receive Side Scaling (RSS) queue pairs per VF.

To change this setting in Intel PROSet

Default	4 Queues
Range	<ul style="list-style-type: none"> • 2 Queues • 4 Queues • 8 Queues • 16 Queues

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Max Number of RSS Queues
Per Vport" -DisplayValue "4 Queues"
```

4.19 NC-SI over MCTP for OCP3 Devices

Some Intel OCP3 Ethernet devices support NC-SI communication using Management Component Transport Protocol (MCTP).

The following devices support NC-SI over MCTP:

- Intel® Ethernet 10G 2P X710-T2L-t OCP
- Intel® Ethernet 10G 4P X710-T4L-t OCP

To use this function, make sure you do the following:

- Update iDRAC to enable NC-SI over MCTP
- Ensure your system is configured for this support
- Use Intel firmware and software from Release 22.5.0 or newer

4.20 Offloads

In addition to the offloads included this subsection, see the following pages for related information:

- "Priority & VLAN Tagging" on page 46
- "Virtual Machine Queue Offloading" on page 63

4.20.1 IPv4 Checksum Offload

This allows the adapter to compute the IPv4 checksum of incoming and outgoing packets. This feature enhances IPv4 receive and transmit performance and reduces CPU utilization.

With Offloading off, the operating system verifies the IPv4 checksum.

With Offloading on, the adapter completes the verification (on RX) and computation (on TX) for the operating system.

To change this setting in Intel PROSet

Default	RX & TX Enabled
Range	<ul style="list-style-type: none"> • Disabled • RX Enabled • TX Enabled • RX & TX Enabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "IPv4 Checksum Offload" -
DisplayValue "Tx Enabled"
```

4.20.2 Large Send Offload (IPv4 and IPv6)

Sets the adapter to offload the task of segmenting TCP messages into valid Ethernet frames. The maximum frame size limit for large send offload is set to 64,000 bytes.

Since the adapter hardware is able to complete data segmentation much faster than operating system software, this feature may improve transmission performance. In addition, the adapter uses fewer CPU resources.

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> • Enabled • Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Large Send Offload V2 (IPv4)" -DisplayValue "Enabled"
```

4.20.3 NVGRE Encapsulated Task Offload

Network Virtualization using Generic Routing Encapsulation (NVGRE) increases the efficient routing of network traffic within a virtualized or cloud environment. Some Intel® Ethernet Network devices perform NVGRE processing, offloading it from the operating system. This reduces CPU utilization.

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> • Enabled • Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "NVGRE Encapsulated Task Offload" -DisplayValue "Enabled"
```

4.20.4 QoS Offload

Configures the Quality of Service (QoS) offload setting for the miniport adapter. This feature allows you to set a bandwidth cap and reservation to one or more virtual machines on a physical device, including both software VMs and SR-IOV interfaces.

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> • Disabled • Enabled

This setting is found on the Adapter tab and in the Adapter Settings panel of the Intel® PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "QoS Offload" -
DisplayValue "Enabled"
```

4.20.5 TCP Checksum Offload (IPv4 and IPv6)

Allows the adapter to verify the TCP checksum of incoming packets and compute the TCP checksum of outgoing packets. This feature enhances receive and transmit performance and reduces CPU utilization.

With Offloading off, the operating system verifies the TCP checksum.

With Offloading on, the adapter completes the verification for the operating system.

To change this setting in Intel PROSet

Default	RX & TX Enabled
Range	<ul style="list-style-type: none"> • Disabled • RX Enabled • TX Enabled • RX & TX Enabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "TCP Checksum Offload
(IPv4)" -DisplayValue "Tx Enabled"
```

4.20.6 UDP Checksum Offload (IPv4 and IPv6)

Allows the adapter to verify the UDP checksum of incoming packets and compute the UDP checksum of outgoing packets. This feature enhances receive and transmit performance and reduces CPU utilization.

With Offloading off, the operating system verifies the UDP checksum.

With Offloading on, the adapter completes the verification for the operating system.

To change this setting in Intel PROSet

Default	RX & TX Enabled
Range	<ul style="list-style-type: none"> • Disabled • RX Enabled • TX Enabled • RX & TX Enabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "UDP Checksum Offload (IPv4)" -DisplayValue "Tx Enabled"
```

4.20.7 UDP Segmentation Offload (IPv4 and IPv6)

Allows the adapter to segmenting UDP packets with payloads up to 64K into valid Ethernet frames. Because the adapter hardware is able to complete data segmentation much faster than operating system software, this feature may improve transmission performance. In addition, the adapter may use fewer CPU resources.

With Offloading off, the operating system segments UDP packets into valid Ethernet frames.

With Offloading on, the adapter segments UDP packets for the operating system.



NOTE: UDP Segmentation Offload requires:

- Microsoft* Windows Server* 2019, Version 1903, or later
- Linux* kernel 4.18, or later

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> • Disabled • Enabled

To change this setting in Windows PowerShell, use the Set_IntelNetAdapterSetting cmdlet. For example:

```
Set_IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "UDP Segmentation Offload (IPv4)" -DisplayValue "Enabled"
```

4.20.8 VXLAN Encapsulated Task Offload

Virtual Extensible LAN (VXLAN) allows you to extend an L2 network over an L3 network, which may be useful in a virtualized or cloud environment. Some Intel Ethernet devices perform VXLAN processing, offloading it from the operating system. This reduces CPU utilization.

VXLAN may be useful in multi-tenant environments such as cloud service providers where the number of VLANs exceeds the 4094 limit imposed by the 12-bit VLAN ID used in Ethernet data frames.

To change this setting in Intel PROSet

Default	Enabled
Range	<ul style="list-style-type: none"> • Enabled • Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "VXLAN Encapsulated Task Offload" -DisplayValue "Enabled"
```

4.21 Performance Options

4.21.1 Optimizing Performance

You can configure Intel network adapter advanced settings to help optimize server performance. This section provides tips for:

- "General Optimization" below
- "Optimization for Specific Usage Models" on the next page



NOTES:

- Linux users, see [the Linux section of this guide](#) and the README file in the Linux driver package for Linux-specific performance enhancement details.
- The recommendations below are guidelines and should be treated as such. Additional factors such as installed applications, bus type, network topology, and operating system also affect system performance.
- These adjustments should be performed by a highly skilled network administrator. They are not guaranteed to improve performance. Not all settings shown here may be available through network driver configuration, operating system or system BIOS.
- When using performance test software, refer to the documentation of the application for optimal results.

4.21.1.1 General Optimization

- Install the adapter in an appropriate slot.



NOTE: Some PCIe x8 slots are actually configured as x4 slots. These slots have insufficient bandwidth for full line rate with some dual port devices. The driver can detect this situation and will write the following message in the system log: "PCI-Express bandwidth available for this card is not sufficient for optimal performance. For optimal performance a x8 PCI-Express slot is required." If this error occurs, moving your adapter to a true x8 slot will resolve the issue.

- For an Intel® Ethernet 700 Series adapter to reach its full potential, you must install it in a PCIe Gen3 x8 slot. Installing it in a shorter slot, or a Gen2 or Gen1 slot, will impact the throughput the adapter can attain.
- Use the proper cabling for your device.
- Increase the number of TCP and Socket resources from the default value. For Windows based systems, we have not identified system parameters other than the TCP Window Size which significantly impact performance.
- Increase the allocation size of Driver Resources (transmit/receive buffers). However, most TCP traffic patterns work best with the transmit buffer set to its default value, and the receive buffer set to its minimum value.

Jumbo Frames

Enabling jumbo frames may increase throughput. You must enable jumbo frames on all of your network components to get any benefit.

RSS Queues

If you have multiple 10 Gbps (or faster) ports installed in a system, the RSS queues of each adapter port can be adjusted to use non-overlapping sets of processors within the adapter's local Non-Uniform Memory Access (NUMA) Node/Socket. Change the RSS Base Processor Number for each adapter port so that the combination of the base processor and the max number of RSS processors settings ensure non-overlapping cores. For Microsoft Windows systems, do the following:

1. Identify the adapter ports to be adjusted and inspect their RssProcessorArray using the Get-NetAdapterRSS PowerShell cmdlet.
2. Identify the processors with NUMA distance 0. These are the cores in the adapter's local NUMA Node/Socket and will provide the best performance.
3. Adjust the RSS Base processor on each port to use a non-overlapping set of processors within the local set of processors. You can do this manually or using the following PowerShell command:

```
Set-NetAdapterAdvancedProperty -Name <Adapter Name> -DisplayName "RSS Base Processor Number" -DisplayValue <RSS Base Proc Value>
```
4. Use the Get-NetAdapterAdvancedproperty cmdlet to check that the right values have been set:

```
Get-NetAdapterAdvancedproperty -Name <Adapter Name>
```

For Example: For a 4 port adapter with Local processors 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, and 'Max RSS processor' of 8, set the RSS base processors to 0, 8, 16 and 24.

CPU Affinity

When passing traffic on multiple network ports using an I/O application that runs on most or all of the cores in your system, consider setting the CPU Affinity for that application to fewer cores. This should reduce CPU utilization and in some cases may increase throughput for the device. The cores selected for CPU Affinity must be local to the affected network device's Processor Node/Group. You can use the PowerShell command Get-NetAdapterRSS to list the cores that are local to a device. You may need to increase the number of cores assigned to the application to maximize throughput. Refer to your operating system documentation for more details on setting the CPU Affinity.

4.21.1.2 Optimization for Specific Usage Models

The following table provides guidance for additional server usage models.

Optimize For	Useful For	Optimization Tasks
Quick response and low latency	Video, audio, and High Performance Computing Cluster (HPCC) servers	<ul style="list-style-type: none"> Minimize or disable interrupt moderation rate Disable offload TCP segmentation Disable jumbo packets Increase transmit descriptors Increase receive descriptors Increase RSS queues
Throughput	Data backup/retrieval and file servers	<ul style="list-style-type: none"> Enable jumbo packets Increase transmit descriptors Increase receive descriptors. On systems that support NUMA, set the Preferred NUMA Node on each adapter to achieve better scaling across NUMA nodes
CPU utilization	Application, web, mail, and database servers	<ul style="list-style-type: none"> Maximize interrupt moderation rate Keep the default setting for the number of receive descriptors; avoid setting large numbers of receive descriptors Decrease RSS queues In Hyper-V environments, decrease the max number of RSS CPUs

4.21.2 Tuning Performance with SR-IOV

When SR-IOV is enabled in Hyper-V, the following steps can help to improve performance between VM to VM and VM to Host.

From host OSEs on both Host 1 and Host 2:

1. Enable RSS on the PF and vSwitch.

```
Enable-NetAdapterRss -name "ADAPTER_NAME"
```

2. Enable 4 queues per VF:

```
Set-VMNetworkAdapter -VMName "YOUR_TEST_VM_NAME"
-IovQueuePairsRequested 4
```

```
Get-VmNetworkAdapter -VMName * | where {$_.SwitchName -eq "YOUR_TEST_
SWITCH_NAME"} | Set-VmNetworkAdapter -IovQueuePairsRequested 4
```

3. Ensure the VMs have at least twice as many vCPUs as RSS queues. In this case, set the number of total processors in the VM to 8. To do this:
 - a. Turn off the VM.
 - b. In the VM, click **Settings**.
 - c. Under **Hardware**, select **Processor**.
 - d. Change the value of **Number of virtual processors** to 8.
 - e. Apply the change.

4. For Windows Server 2022, issue the following command while the VM is in the off state:

```
Set-VMProcessor -VMName "YOUR_VM_Name" -HwThreadCountPerCore 1  
-Count 8
```

In both guest OSes VM1 and VM2:

1. Set RSS queues to 4 for all VFs in the guest OSes:

```
Set-NetAdapterRss -InterfaceDescription *adaptive*  
-NumberOfReceiveQueues 4
```

2. Update the number of queues in the guest OS:

```
Set-NetAdapterAdvancedProperty -Name "your_adapter_name_from_guest_os" -  
DisplayName "Maximum Number of RSS Queues" -DisplayValue "8 Queues"
```



NOTE: In the locations where there are settings for the number of queues, that value can be anything from 1 to 16. If you want more total throughput, increase the number of queues. When updating the number of queues, you **must** set `IovQueuePairsRequested` to a value that is equal to or greater than the number of queues you want to use in the VM.

4.21.3 Transmit Balancing

Some Intel® Ethernet 800 Series devices allow you to enable a transmit balancing feature to improve transmit performance under certain conditions. When the feature is enabled, you should experience more consistent transmit performance across queues and/or PFs and VFs.

By default, transmit balancing is disabled in the NVM. To enable this feature, use one of the following to persistently change the setting for the device:

- Use the Ethernet Port Configuration Tool (EPCT) to enable the `tx_balancing` option. Refer to the EPCT readme for more information.
- Enable the Transmit Balancing device setting in UEFI HII.
- Enable transmit balancing via Linux devlink. Refer to the Linux chapter for more information.

When the driver loads, it reads the transmit balancing setting from the NVM and configures the device accordingly.



NOTE:

- The user selection for transmit balancing in EPCT, HII, or Linux devlink is persistent across reboots. You must reboot the system for the selected setting to take effect.
- This setting is device wide.
- The driver, NVM, and DDP package must all support this functionality to enable the feature.

4.21.4 Performance Profile

Performance Profiles are supported on Intel® 10GbE adapters and allow you to quickly optimize the performance of your Intel® Ethernet Adapter. Selecting a performance profile will automatically adjust some Advanced Settings to their optimum setting for the selected application. For example, a standard server has optimal performance with only two RSS (Receive-Side Scaling) queues, but a web server requires more RSS queues for better scalability.

To change this setting in Intel PROSet

You must install Intel PROSet to use Performance profiles.

Profiles	<ul style="list-style-type: none"> • Standard Server – This profile is optimized for typical servers. • Web Server – This profile is optimized for IIS and HTTP-based web servers. • Virtualization Server – This profile is optimized for Microsoft’s Hyper-V virtualization environment. • Storage Server – This profile is optimized for Fibre Channel over Ethernet or for iSCSI over DCB performance. Selecting this profile will disable SR-IOV and VMQ. • Storage + Virtualization – This profile is optimized for a combination of storage and virtualization requirements. • Low Latency – This profile is optimized to minimize network latency.
-----------------	--



NOTES:

- Not all options are available on all adapter/operating system combinations.
- If you have selected the Virtualization Server profile or the Storage + Virtualization profile, and you uninstall the Hyper-V role, you should select a new profile.

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Profile" -DisplayValue "Standard Server"
```

4.22 Power Options

The Adapter Settings panel in Intel PROSet ACU includes several settings that control the device's power consumption. For example, you can set the adapter to reduce its power consumption if the cable is disconnected.

4.22.1 ACPI Power States

Advanced Configuration and Power Interface (ACPI) supports a variety of power states. Each state represents a different level of power, from fully powered up to completely powered down, with partial levels of power in each intermediate state.

The following table describes the ACPI power states.

Power State	Description
S0	On and fully operational
S1	System is in low-power mode (sleep mode). The CPU clock is stopped, but RAM is powered on and being refreshed.
S2	Similar to S1, but power is removed from the CPU.

Power State	Description
S3	Suspend to RAM (standby mode). Most components are shut down. RAM remains operational.
S4	Suspend to disk (hibernate mode). The memory contents are swapped to the disk drive and then reloaded into RAM when the system is awakened.
S5	Power off

Microsoft Windows Server is ACPI-capable. It does not support waking from a power-off (S5) state, only from standby (S3) or hibernate (S4). When shutting down the system, these states shut down ACPI devices, including Intel Ethernet adapters. This disarms the adapter's remote wake-up capability. However, in some ACPI-capable computers, the BIOS may have a setting that allows you to override the operating system and wake from an S5 state anyway. If there is no support for wake from S5 state in your BIOS settings, you are limited to Wake From Standby when using these operating systems in ACPI computers.

4.22.2 Wake on LAN (WoL) Options

The ability to remotely wake computers is an important development in computer management. This feature has evolved from a simple remote power-on capability to a complex system interacting with a variety of device and operating system power states.

The Adapter Settings panel in Intel PROSet ACU includes **Wake on Magic Packet** and **Wake on directed packet** settings. These control the type of packets that wake up the system from standby.

For some adapters, the Adapter Settings panel in Intel PROSet ACU includes a setting called **Wake on Magic Packet from power off state**. Enable this setting to explicitly allow wake-up with a Magic Packet* from shutdown under APM power management mode.



NOTES:

- If **Reduce speed during standby** is enabled, then **Wake on Magic Packet** and/or **Wake on directed packet** must be enabled. If both of these options are disabled, power is removed from the adapter during standby.
- **Wake on Magic Packet from power off state** has no effect on this option.

4.22.2.1 WoL Supported Devices

All devices support Wake on LAN on all ports, on all operating systems, with the exceptions listed in the following table:

Family	Device	Adapter Port(s) supporting WoL
Intel Ethernet 800 Series	Intel® Ethernet 100G 2P E810-C Adapter Intel® Ethernet 100G 2P E810-C-st Adapter Intel® Ethernet 100G 2P E810-C-stg Adapter	Not supported
	Intel® Ethernet 25G 2P E810-XXV Adapter Intel® Ethernet 25G 4P E810-XXV Adapter Intel® Ethernet 25G 4P E810-XXV-st Adapter Intel® Ethernet 25G 4P E810-XXV-stg Adapter	Not supported
	Intel® Ethernet Connection 25G 4P E823-C LOM	WoL supported but disabled by default
	Intel® Ethernet 25G 2P E810-XXV-k Mezz Intel® Ethernet 25G 4P E810-XXV OCP	This device only supports waking from a powered off (S5) state. It does not support waking from sleep/hibernate (S3/S4).
Intel Ethernet 700 Series	All	Waking from S1 through S4 states not supported. As a result, the Allow this device to wake the computer setting in Device Manager is grayed out. Unless listed in other rows in this table, 700 Series devices support waking from S5.
	Intel® Ethernet 25G 2P XXV710 Adapter	Not supported
	Intel® Ethernet 25G 2P XXV710 Mezz	This device only supports waking from a powered off (S5) state. It does not support waking from sleep/hibernate (S3/S4).
	Intel® Ethernet Converged Network Adapter X710-4 Intel® Ethernet Converged Network Adapter X710-2 Intel® Ethernet Converged Network Adapter X710	Port 1 only
	Intel® Ethernet Converged Network Adapter X710-T Intel® Ethernet Converged Network Adapter XL710-Q2	Not supported
	Intel® Ethernet 10G 2P X710-T2L-t Adapter Intel® Ethernet 10G 4P X710-T4L-t Adapter	Not supported
Intel Ethernet 500 Series	Intel® Ethernet 10G 2P X550-t Adapter	Not supported
Intel Ethernet 300 Series	Intel® Gigabit 2P I350-t Adapter Intel® Gigabit 4P I350-t Adapter	Port 1 only NOTE: iDRAC may report that ports 2, 3, or 4 are capable of WoL. Only port 1 supports WoL.

4.22.2.2 Wake on Link Settings

Wakes the computer if the network connection establishes link while the computer is in standby mode. You can enable the feature, disable it, or let the operating system use its default.

NOTES:

- If a copper-based Intel adapter is advertising a speed of one gigabit only, this feature does not work because the adapter cannot identify a gigabit link at a D3 state.
- The network cable must be disconnected when entering into S3/S4 in order to wake the system up by link up event.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Disabled • OS Controlled • Forced

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Wake on Link Settings" -
DisplayValue "Forced"
```


4.22.3 Remote Wake-Up

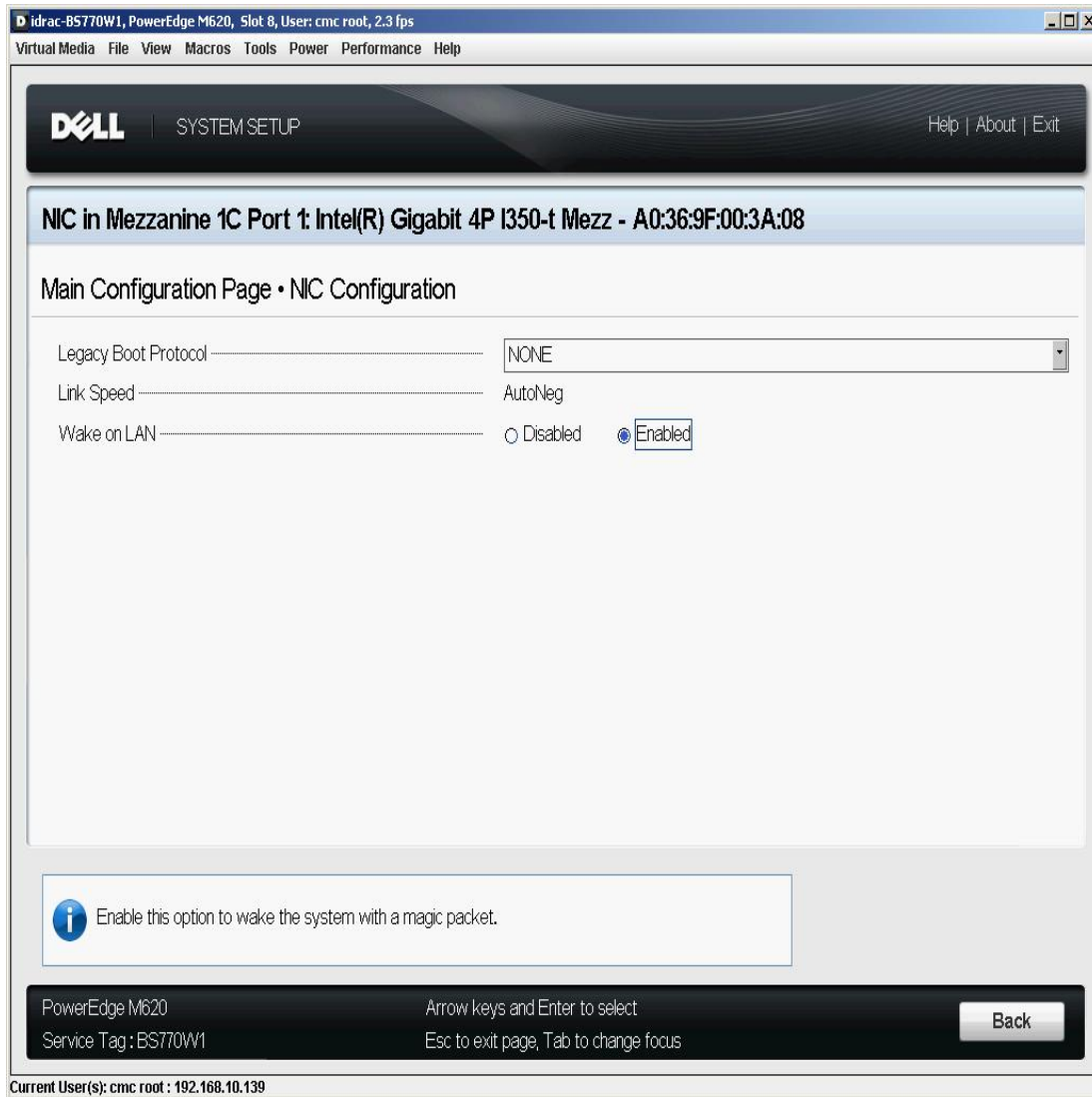
Remote wake-up can wake your server from a low power or powered off state. If Wake On LAN is enabled, when your system is powered down, the network interface draws standby power and listens for specially designed packet. If it receives such a packet it will wake your system.

4.22.3.1 Enabling Wake From Power Off

If you want to wake your system from a power off state, you must enable it from the System Setup.

1. Go to System Setup.
2. Choose a port and go to configuration.
3. Enable Wake on LAN.

 **NOTE:** For Intel OCP adapters that support Wake on LAN from an S5 state, you must also open iDRAC, go to iDRAC Settings > Network, and enable **OCP Slot Power During Host Off (S5 State)**.



4.22.3.2 Wake-Up Address Patterns

Remote wake-up can be initiated by a variety of user selectable packet types and is not limited to the Magic Packet format. For more information about supported packet types, see the [operating system settings](#) section.

The wake-up capability of Intel adapters is based on patterns sent by the OS. You can configure the driver to the following settings using Intel PROSet. For Linux, WoL is provided through the ethtool utility. For more information on ethtool, see the following Web site: <http://sourceforge.net/projects/gkernel>.

- Wake on Directed Packet - accepts only patterns containing the adapter's Ethernet address in the Ethernet header or containing the IP address, assigned to the adapter, in the IP header.
- Wake on Magic Packet - accept only patterns containing 16 consecutive repetitions of the adapter's MAC address.
- Wake on Directed Packet and Wake on Magic Packet - accepts the patterns of both directed packets and magic packets.

Choosing "Wake on directed packet" will also allow the adapter to accept patterns of the Address Resolution Protocol (ARP) querying the IP address assigned to the adapter. If multiple IP addresses are assigned to an adapter, the operating system may request to wake up on ARP patterns querying any of the assigned addresses. However, the adapter will only awaken in response to ARP packets querying the first IP address in the list, usually the first address assigned to the adapter.

4.22.3.3 Physical Installation Issues

The following table describes possible issues you might encounter with the physical device and remote wake-up.

Area	Possible Issue
Slot	Some motherboards will only support remote wake-up (or remote wake-up from S5 state) in a particular slot. See the documentation that came with your system for details on remote wake-up support.
Power	Some Intel adapters are 3.3 volt and some are 12 volt. They are keyed to fit either type of slot. The 3.3 volt standby supply must be capable of supplying at least 0.2 amps for each Intel adapter installed.

4.22.3.4 Operating System Settings

Linux

Remote Wake-Up is supported in [Linux](#).

Microsoft Windows

Microsoft Windows Server is ACPI-capable. It does not support waking from a power-off (S5) state, only from standby (S3) or hibernate (S4). When shutting down the system, these states shut down ACPI devices, including Intel Ethernet adapters. This disarms the adapter's remote wake-up capability. However, in some ACPI-capable computers, the BIOS may have a setting that allows you to override the operating system and wake from an S5 state anyway. If there is no support for wake from S5 state in your BIOS settings, you are limited to Wake From Standby when using these operating systems in ACPI computers.

To change this setting in Intel PROSet

Intel PROSet ACU includes some settings to allow some adapters to wake from power off state. Refer to "Wake on LAN (WoL) Options" on page 41 or the Intel PROSet help for more details.

In ACPI-capable versions of Windows, the Intel PROSet advanced settings include a setting called Wake on Settings. This setting controls the type of packets that wake the system from standby. See Intel PROSet help for more details.

If you do not have Intel PROSet installed, you will need to do the following:

1. Open the Device Manager, then navigate to the **Power Management** tab, and check "**Allow this device to bring the computer out of standby.**"
2. On the **Advanced** tab, set the "**Wake on Magic packet**" option to Enabled.

In order to wake from S5 without Intel PROSet, on the **Advanced tab**, set "**Enable PME**" to Enabled.

4.22.4 Reduce Power if Cable Disconnected & Reduce Link Speed During Standby

Enables the adapter to reduce power consumption when the LAN cable is disconnected from the adapter and there is no link. When the adapter regains a valid link, adapter power usage returns to its normal state (full power usage).

The Hardware Default option is available on some adapters. If this option is selected, the feature is disabled or enabled based on the system hardware.

Default	The default varies with the operating system and adapter.
Range	The range varies with the operating system and adapter.

4.22.5 Energy Efficient Ethernet

The Energy Efficient Ethernet (EEE) feature allows a capable device to enter Low-Power Idle between bursts of network traffic. Both ends of a link must have EEE enabled for any power to be saved. Both ends of the link will resume full power when data needs to be transmitted. This transition may introduce a small amount of network latency.



NOTES:

- Both ends of the EEE link must automatically negotiate link speed.
- EEE is not supported on every adapter.

4.23 Priority & VLAN Tagging

Enables the adapter to offload the insertion and removal of priority and VLAN tags for transmit and receive.

To change this setting in Intel PROSet

Default	Priority & VLAN Enabled
Range	<ul style="list-style-type: none"> • Priority & VLAN Disabled • Priority Enabled • VLAN Enabled • Priority & VLAN Enabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To set this in Windows Powershell, first disable DCB, then set priority and VLAN tagging. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "DCB" -DisplayValue "Disabled"
```

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Packet Priority & VLAN" -DisplayValue "VLAN Enabled"
```

4.24 Quality of Service

Quality of Service (QoS) allows the adapter to send and receive IEEE 802.3ac tagged frames. 802.3ac tagged frames include 802.1p priority-tagged frames and 802.1Q VLAN-tagged frames. In order to implement QoS, the adapter must be connected to a switch that supports and is configured for QoS. Priority-tagged frames allow programs that deal with real-time events to make the most efficient use of network bandwidth. High priority packets are processed before lower priority packets.

To change this setting in Intel PROSet

To implement QoS, the adapter must be connected to a switch that supports and is configured for 802.1p QoS.

QoS Tagging is enabled and disabled in the Adapter Settings panel of Intel PROSet ACU.

To set this in Windows Powershell, first disable DCB, then set QoS using the Priority and VLAN tagging DisplayName in the cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "DCB" -DisplayValue "Disabled"
```

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Packet Priority & VLAN" -DisplayValue "VLAN Enabled"
```

Once QoS is enabled, you can specify levels of priority based on IEEE 802.1p/802.1Q frame tagging.

4.25 Receive Buffers

Defines the number of Receive Buffers, which are data segments. They are allocated in the host memory and used to store the received packets. Each received packet requires at least one Receive Buffer, and each buffer uses 2KB of memory.

You might choose to increase the number of Receive Buffers if you notice a significant decrease in the performance of received traffic. If receive performance is not an issue, use the default setting appropriate to the adapter.

To change this setting in Intel PROSet

Default	512, for all adapters.
Range	128-4096, in intervals of 64, for all adapters.
Recommended Value	Using IPSec and/or multiple features: 352

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Receive Buffers" -DisplayValue "256"
```

4.26 Receive Side Scaling

When Receive Side Scaling (RSS) is enabled, all of the receive data processing for a particular TCP connection is shared across multiple processors or processor cores. Without RSS all of the processing is

performed by a single processor, resulting in less efficient system cache utilization.

4.26.1 LAN RSS

LAN RSS applies to a particular TCP connection.



NOTE: This setting has no effect if your system has only one processing unit.

4.26.1.1 LAN RSS Configuration

If your adapter does not support RSS, or if the SNP or SP2 is not installed, the RSS setting will not be displayed. If RSS is supported in your system environment, the following will be displayed:


- **Port NUMA Node.** This is the NUMA node number of a device.
- **Starting RSS CPU.** This setting allows you to set the preferred starting RSS processor. Change this setting if the current processor is dedicated to other processes. The setting range is from 0 to the number of logical CPUs - 1.
- **Max number of RSS CPU.** This setting allows you to set the maximum number of CPUs assigned to an adapter and is primarily used in a Hyper-V environment. By decreasing this setting in a Hyper-V environment, the total number of interrupts is reduced which lowers CPU utilization. The default is 8 for Gigabit adapters and 16 for 10 Gigabit, or faster, adapters.
- **Preferred NUMA Node.** This setting allows you to choose the preferred NUMA (Non-Uniform Memory Access) node to be used for memory allocations made by the network adapter. In addition, the system will attempt to use the CPUs from the preferred NUMA node first for the purposes of RSS. On NUMA platforms, memory access latency is dependent on the memory location. Allocation of memory from the closest node helps improve performance. The Windows Task Manager shows the NUMA Node ID for each processor.



NOTES:

- This setting only affects NUMA systems. It will have no effect on non-NUMA systems.
 - Choosing a value greater than the number of NUMA nodes present in the system selects the NUMA node closest to the device.
- **Receive Side Scaling Queues.** This setting configures the number of RSS queues, which determine the space to buffer transactions between the network adapter and CPU(s).

To change this setting in Intel PROSet

Default	2 queues for the Intel® 10 Gigabit Server Adapters
Range	<ul style="list-style-type: none"> • 1 queue is used when low CPU utilization is required. • 2 queues are used when good throughput and low CPU utilization are required. • 4 or more queues are used for applications that demand maximum throughput and transactions per second. <p> NOTES:</p> <ul style="list-style-type: none"> • Not all settings are available on all adapters. • 8, or more, queues are only available when Intel PROSet ACU is installed. If Intel PROSet is not installed, only 4 queues are available. • Using 8 or more queues requires the system to reboot.

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Receive Side Scaling" -
DisplayValue "Enabled"
```

4.27 Remote Direct Memory Access (RDMA)

Remote Direct Memory Access, or RDMA, allows a network device to transfer data directly to and from application memory on another system, increasing throughput and lowering latency in certain networking environments.

- Intel® Ethernet 800 Series devices support both iWARP and RoCEv2.

The major difference is that iWARP performs RDMA over TCP, while RoCEv2 uses UDP.

On devices with RDMA capabilities, RDMA is supported on the following operating systems (refer to "Supported Operating Systems" on page 2 for currently supported versions):

- Linux
- ESXi
- Windows Server

To avoid performance degradation from dropped packets, enable link level flow control or priority flow control on all network interfaces and switches.

NOTES:

- On systems running a Microsoft Windows Server operating system, enabling *QoS/priority flow control will disable link level flow control.
- Devices based on the Intel® Ethernet 800 Series do not support RDMA when operating in multiport mode with more than 4 ports.
- On Linux systems, RDMA and link aggregation (LAG, also known as *bonding*) are not compatible on most devices. If RDMA is enabled, bonding will not be functional.
 - On Intel Ethernet 810 Series devices, RDMA and LAG are compatible if all the following are true:
 - RDMA technology is set to RoCEv2.
 - LAG configuration is active-backup.
 - Bonding is between two ports within the same device.
 - The QoS configuration of the two ports matches prior to the bonding of the devices.

4.27.1 RDMA on Linux

For Intel Ethernet devices that support RDMA on Linux, use the drivers shown in the following table.

Device	Linux		Supported Protocols
	Base Driver	RDMA Driver	
Intel® Ethernet 800 Series	ice	irdma	RoCEv2, iWARP

Basic Installation Instructions

At a high level, installing and configuring RDMA on Linux consists of the following steps. See the README file inside the appropriate RDMA driver tarball for full details.

1. Install the base driver.
2. Install the RDMA driver.
3. Install and patch any user-mode RDMA libraries. Exact steps will vary by operating system; refer to the RDMA driver readme for details.
4. Enable flow control on your device. Refer to the base driver README for details and supported modes.
5. If you are using RoCE, enable flow control (PFC or LFC) on the device and endpoint your system is connected to. See your switch documentation and, for Linux, the [Intel® Ethernet 800 Series Linux Flow Control Configuration Guide for RDMA Use Cases](#) for details.

4.27.1.1 RDMA for Virtualized Environments in Linux


Devices based on the Intel Ethernet 800 Series support RDMA in a Linux VF on supported Windows or Linux hosts. Refer to the README file inside the Linux RDMA driver tarball for more information on how to load and configure RDMA in a Linux VF.

4.27.2 RDMA on Microsoft Windows

Refer to "Installing the RDMA Driver" on page 66 for instructions how to install the RDMA driver.

4.27.2.1 RDMA for Network Direct (ND) User-Mode Applications

Network Direct (ND) allows user-mode applications to use RDMA features.

 **NOTE:** User mode applications may have prerequisites such as Microsoft HPC Pack or Intel MPI Library, refer to your application documentation for more details.


4.27.2.2 RDMA Network Direct Kernel (NDK)

RDMA Routing Across IP Subnets

If you want to allow NDK's RDMA functionality across subnets, you will need to select "Enable RDMA routing across IP Subnets" on the RDMA Configuration Options screen during [base driver installation](#).

4.27.2.3 Enabling Priority Flow Control (PFC) on a Microsoft Windows Server Operating System

To avoid performance degradation from dropped packets, enable priority flow control (PFC) or link level flow control on all network interfaces and switches.

 **NOTE:** On systems running a Microsoft Windows Server operating system, enabling *QoS/priority flow control will disable link level flow control.

Use the following PowerShell commands to enable PFC on Microsoft Windows Server:

```
Install-WindowsFeature -Name Data-Center-Bridging -IncludeManagementTools
New-NetQoSPolicy "SMB" -NetDirectPortMatchCondition 445 -PriorityValue8021Action 3
Enable-NetQoSFlowControl -Priority 3
Disable-NetQoSFlowControl -Priority 0,1,2,4,5,6,7
New-NetQoSTrafficClass -Name "SMB" -Priority 3 -BandwidthPercentage 60 -Algorithm ETS
Set-NetQoSDbxSetting -Willing $FALSE
Enable-NetAdapterQos -Name "Slot1 4 2 Port 1"
```

4.27.2.4 Verifying RDMA Operation with Microsoft PowerShell

You can check that RDMA is enabled on the network interfaces using the following Microsoft PowerShell command:

```
Get-NetAdapterRDMA
```

Use the following PowerShell command to check if the network interfaces are RDMA capable and multichannel is enabled:

```
Get-SmbClientNetworkInterface
```

Use the following PowerShell command to check if Network Direct is enabled in the operating system:

```
Get-NetOffloadGlobalSetting | Select NetworkDirect
```

Use netstat to make sure each RDMA-capable network interface has a listener at port 445 (Windows Client OSs that support RDMA may not post listeners). For example:

```
netstat.exe -xan | ? {$_ -match "445"}
```

4.27.2.5 RDMA for Virtualized Environments in Windows

To enable RDMA functionality on virtual adapter(s) connected to a VMSwitch, you must:

- Enable SR-IOV (Single Root IO Virtualization) and VMQ (Virtual Machine Queues) advanced properties on each port.
- Set the number of VFs to enable with RDMA capabilities. You can enable up to 32 VFs with RDMA capabilities.

Under certain circumstances, you may disable these settings by default. You can manually set these options in the Adapter Settings panel of Intel PROSet ACU or with the following PowerShell commands:

```
Set-NetAdapterAdvancedProperty -Name <nic_name> -RegistryKeyword *SRIOV -RegistryValue 1
Set-NetAdapterAdvancedProperty -Name <nic_name> -RegistryKeyword *VMQ -RegistryValue 1
Set-NetAdapterAdvancedProperty -Name <nic_name> -RegistryKeyword RdmaMaxVfsEnabled -
RegistryValue <1-32>
```

4.27.2.6 Configuring RDMA Guest Support (NDK Mode 3)

NDK Mode 3 allows kernel mode Windows components to use RDMA features inside Hyper-V guest partitions. To enable NDK mode 3 on an Intel Ethernet device, do the following:

1. Enable SR-IOV in your system's BIOS or uEFI.
2. Enable the SR-IOV advanced setting on the device.
3. Enable SR-IOV on the VMSwitch bound to the device by performing the following for all physical functions on the same device:


```
New-VMSwitch -Name <switch_name> -NetAdapterName <device_name>
-EnableIov $true
```
4. Configure the number of RDMA virtual functions (VFs) on the device by setting the "RdmaMaxVfsEnabled" advanced setting. All physical functions must be set to the same value. The value is the maximum number of VFs that can be capable of RDMA at one time for the entire device. Enabling more VFs will restrict RDMA resources from physical functions (PFs) and other VFs.


```
Set-NetAdapterAdvancedProperty -Name <device_name> -RegistryKeyword
RdmaMaxVfsEnabled -RegistryValue <Value: 0 - 32>
```
5. Disable all PF adapters on the host and re-enable them. This is required when the registry keyword "RdmaMaxVfsEnabled" is changed or when creating or destroying a VMSwitch.


```
Get-NetAdapterRdma | Disable-NetAdapter
Get-NetAdapterRdma | Enable-NetAdapter
```
6. Create VM Network Adapters for VMs that require RDMA VF support.


```
Add-VMNetworkAdapter -VMName <vm_name> -VMNetworkAdapterName <device_name> -
SwitchName <switch_name>
```
7. If you plan to use Microsoft Windows 10 Creators Update (RS2) or later on a guest partition, set the RDMA weight on the VM Network Adapter by entering the following command on the host:


```
Set-VMNetworkAdapterRdma -VMName <vm_name> -VMNetworkAdapterName <device_name> -
RdmaWeight 100
```
8. Set SR-IOV weight on the VM Network Adapter (Note: SR-IOV weight must be set to 0 before setting the RdmaWeight to 0):


```
Set-VMNetworkAdapter -VMName <vm_name> -VMNetworkAdapterName <device_name> -
IovWeight 100
```
9. Install the VF network adapter with the PROSET Installer in the VM.

10. Enable RDMA on the VF driver and Hyper-V Network Adapter using PowerShell in the VM:

```
Set-NetAdapterAdvancedProperty -Name <device_name> -RegistryKeyword RdmaVfEnabled -
RegistryValue 1
Get-NetAdapterRdma | Enable-NetAdapterRdma
```

4.27.2.7 RDMA for NDK Features such as SMB Direct (Server Message Block)

NDK allows Windows components (such as SMB Direct storage) to use RDMA features.

Testing NDK: Microsoft Windows SMB Direct with DiskSPD

This section outlines the recommended way to test RDMA for Intel Ethernet functionality and performance on Microsoft Windows operating systems.

Note that since SMB Direct is a storage workload, the performance of the benchmark may be limited to the speed of the storage device rather than the network interface being tested. Intel recommends using the fastest storage possible in order to test the true capabilities of the network device(s) under test.

Test instructions:

1. Set up and connect at least two servers running a supported Microsoft Windows Server operating system, with at least one RDMA-capable Intel® Ethernet device per server.
2. On the system designated as the SMB server, set up an SMB share. Note that the performance of the benchmark may be limited to the speed of the storage device rather than the network interface being tested. Storage setup is outside of the scope of this document. You can use the following PowerShell command:

```
New-SmbShare -Name <SMBsharename> -Path <SMBsharefilepath> -FullAccess
<domainname>\Administrator,Everyone
```

For Example:

```
New-SmbShare -Name RAMDISKShare -Path R:\RAMDISK -FullAccess
group\Administrator,Everyone
```

3. Download and install the Diskspd Microsoft utility from here: <https://gallery.technet.microsoft.com/DiskSpd-a-robust-storage-6cd2f223>
4. Using CMD or Powershell, cd to the DiskSpd folder and run tests. (Refer to Diskspd documentation for more details on parameters)

For Example: Set the block size to 4K, run the test for 60 seconds, disable all hardware and software caching, measure and display latency statistics, leverage 16 overlapped IOs and 16 threads per target, random 0% writes and 100% reads and create a 10GB test file at

```
"\\<SMBserverTestIP>\<SMBsharename>\test.dat" :
.\diskspd.exe -b4K -d60 -h -L -o16 -t16 -r -w0 -c10G
\\<SMBserverTestIP>\<SMBsharename>\test.dat
```

5. Verify that RDMA traffic is running using perfmon counters such as "RDMA Activity" and "SMB Direct Connection". Refer to Microsoft documentation for more details.

4.27.2.8 RDMA Windows Performance Monitoring

You can use perfmon, or other performance monitoring tool, to monitor and display RDMA counters and statistics. Refer to Microsoft documentation for more details. Use the Register-IntelEthernetRDMACounterSet cmdlet registers the RDMA statistics counters for the specific device with

perfmom. Refer to "Configuring with Windows PowerShell" on page 15 for more information about how to install and use Intel Ethernet cmdlets. You can use the following PowerShell command to register the RDMA statistics for all supported devices:

```
Register-IntelEthernetRDMACounterSet
```

You can use the following PowerShell cmdlet to unregister the RDMA statistics:

```
Unregister-IntelEthernetRDMACounterSet
```

4.27.3 Accessing Remote NVM Express* Drives Using RDMA

RDMA provides a high throughput, low latency means to directly access NVM Express* (NVMe*) drives on a remote server.

Refer to the following for details on supported operating systems and how to set up and configure your server and client systems:

- *NVM Express over TCP for Intel® Ethernet Products Configuration Guide*
- *NVM Express over Fabrics for Intel® Ethernet Products with RDMA Configuration Guide*

Both guides are available on the [Intel Technical Library](#).

4.28 Setting Speed and Duplex

The Link Speed and Duplex setting lets you choose how the adapter sends and receives data packets over the network.

In the default mode, an Intel network adapter using copper connections will attempt to auto-negotiate with its link partner to determine the best setting. If the adapter cannot establish link with the link partner using auto-negotiation, you may need to manually configure the adapter and link partner to the identical setting to establish link and pass packets. This should only be needed when attempting to link with an older switch that does not support auto-negotiation or one that has been forced to a specific speed or duplex mode.

Auto-negotiation is disabled by selecting a discrete speed and duplex mode in the adapter properties sheet. The settings available when auto-negotiation is disabled are dependent on your device. Not all speeds are available on all devices. Your link partner must match the setting you choose.



NOTES:

- Only experienced network administrators should force speed and duplex manually.
- Fiber-based adapters operate only in full duplex at their native speed. You cannot change the speed or duplex of Intel adapters that use fiber cabling.
- Some devices may list 10 Mbps and 100 Mbps in full or half duplex as options. These settings are not recommended.
- Link speed information in Intel PROSet may display a blue informational icon with a mouse-over message "This device is not linked at its maximum capable speed". In that case, if your device is set to auto-negotiate, you can adjust the speed of the device's link partner to the device's maximum speed. If the device is not set to auto-negotiate, you can adjust the device's speed manually, but you must ensure the link partner is set at the same speed.

4.28.1 Manually Configuring Duplex and Speed Settings



CAUTION: The settings at the switch must always match the adapter settings. Adapter performance may suffer, or your adapter might not operate correctly if you configure the adapter differently from your switch.

Configuration is specific to your operating system driver. To set a specific Link Speed and Duplex mode, refer to the section below that corresponds to your operating system.

4.28.1.1 Windows

The default setting is for auto-negotiation to be enabled. Only change this setting to match your link partner's speed and duplex setting if you are having trouble connecting.

To change this setting in Intel PROSet

In Intel PROSet ACU, link speed is reported on the Adapter Information panel. Change speed and duplex in the Adapter Settings panel.

To change this setting in Windows PowerShell, use the `Set-IntelNetAdapterSetting` cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Speed & Duplex" -  
DisplayValue "Auto Negotiation"
```

4.28.1.2 Linux

Refer to the [Linux section](#) of this guide for information on configuring speed and duplex on Linux systems.

4.29 Thermal Monitoring

Adapters and network controllers based on the Intel® Ethernet Controller I350 (and later controllers) can display temperature data and automatically reduce the link speed if the controller temperature gets too hot.



NOTE: This feature is enabled and configured by the equipment manufacturer. It is not available on all adapters and network controllers. There are no user configurable settings.

4.29.1 Monitoring and Reporting

Temperature information is displayed in the Adapter Information panel in Intel PROSet ACU. There are three possible conditions:

- Temperature: Normal
Indicates normal operation.
- Temperature: Overheated, Link Reduced
Indicates that the device has reduced link speed to lower power consumption and heat.
- Temperature: Overheated, Adapter Stopped
Indicates that the device is too hot and has stopped passing traffic so it is not damaged.

If either of the overheated events occur, the device driver writes a message to the system event log.

4.30 Timestamps

4.30.1 PTP Hardware Timestamp

Allows applications that use PTPv2 (Precision Time Protocol) to use hardware generated timestamps to synchronize clocks throughout your network. If this setting is enabled, it takes precedence over the [Software Timestamp](#) setting.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Enabled • Disabled

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "PTP Hardware Timestamp" -
DisplayValue "Enabled"
```

4.30.2 Software Timestamp

Allows applications that use PTPv2 (Precision Time Protocol) to use software generated timestamps to synchronize clocks throughout your network. If the [PTP Hardware Timestamp](#) setting is enabled, it takes precedence over this setting.

To change this setting in Intel PROSet

Default	Disabled
Range	<ul style="list-style-type: none"> • Disabled • RxAll • TxAll • RxAll & TxAll • TaggedTx • RxAll & TaggedTx

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Software Timestamp" -
DisplayValue "RxAll"
```

4.31 Transmit Buffers

Defines the number of Transmit Buffers, which are data segments that enable the adapter to track transmit packets in the system memory. Depending on the size of the packet, each transmit packet

requires one or more Transmit Buffers.

You might choose to increase the number of Transmit Buffers if you notice a possible problem with transmit performance. Although increasing the number of Transmit Buffers can enhance transmit performance, Transmit Buffers do consume system memory. If transmit performance is not an issue, use the default setting. This default setting varies with the type of adapter.

View the [Adapter Specifications](#) topic for help identifying your adapter.

To change this setting in Intel PROSet

Default	512, depending on the requirements of the adapter
Range	128-16384, in intervals of 64, for 10 Gigabit Server Adapters. 128-4096, in intervals of 64, for all other adapters.

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Transmit Buffers" -
DisplayValue "128"
```

4.32 Virtualization Support

Virtualization makes it possible for one or more operating systems to run simultaneously on the same physical system as virtual machines. This allows you to consolidate several servers onto one system, even if they are running different operating systems. Intel® Network Adapters work with, and within, virtual machines with their standard drivers and software.



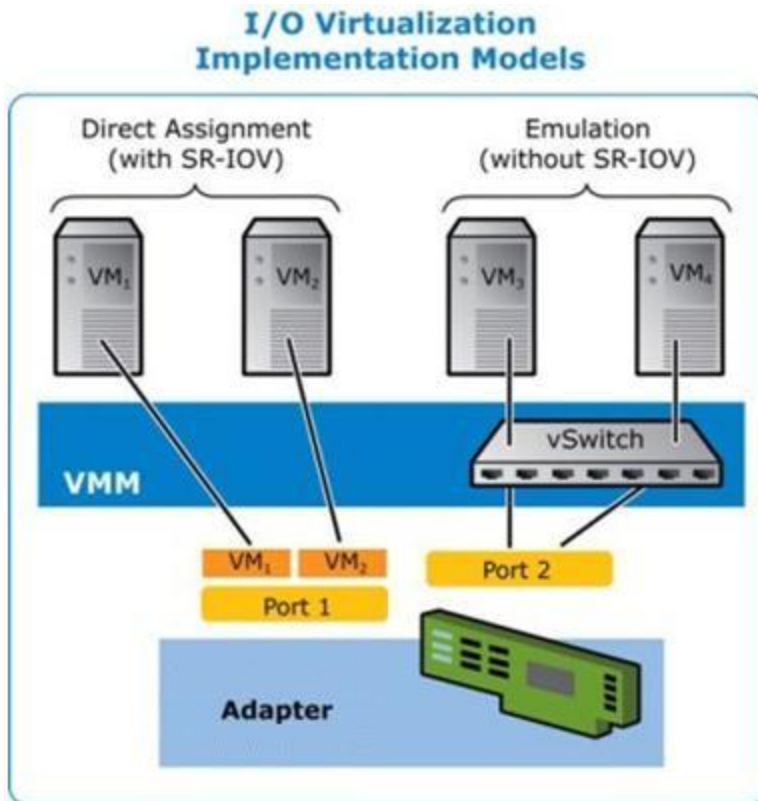
NOTES:

- Some virtualization options are not available on some adapter/operating system combinations.
- The jumbo frame setting inside a virtual machine must be the same, or lower than, the setting on the physical port.
- When you attach a Virtual Machine to a tenant overlay network through the Virtual NIC ports on a Virtual Switch, the encapsulation headers increase the Maximum Transmission Unit (MTU) size on the virtual port. The Encapsulation Overhead feature automatically adjusts the physical port's MTU size to compensate for this increase.

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

4.32.1.1 SR-IOV Overview

Single Root I/O Virtualization (SR-IOV) is a PCI SIG specification allowing PCI Express devices to appear as multiple separate physical PCI Express devices. SR-IOV allows efficient sharing of PCI devices among Virtual Machines (VMs). It manages and transports data without the use of a hypervisor by providing independent memory space, interrupts, and DMA streams for each virtual machine.



SR-IOV architecture includes two functions:

- Physical Function (PF) is a full featured PCI Express function that can be discovered, managed and configured like any other PCI Express device.
- Virtual Function (VF) is similar to PF but cannot be configured and only has the ability to transfer data in and out. The VF is assigned to a Virtual Machine.

4.32.1.2 Configuring SR-IOV

SR-IOV lets a single network port appear to be several virtual functions in a virtualized environment. If you have an SR-IOV capable NIC, each port on that NIC can assign a virtual function to several guest partitions. The virtual functions bypass the Virtual Machine Manager (VMM), allowing packet data to move directly to a guest partition's memory, resulting in higher throughput and lower CPU utilization. SR-IOV also allows you to move packet data directly to a guest partition's memory. See your operating system documentation for system requirements.

For devices that support it, SR-IOV is enabled in the host partition. Some devices may need to have SR-IOV enabled in a preboot environment.

 **NOTES:**

- **Configuring SR-IOV for improved network security:** In a virtualized environment, on Intel® Server Adapters that support SR-IOV or Intel® Scalable I/O Virtualization (Intel® Scalable IOV), the virtual function (VF) may be subject to malicious behavior. Software-generated layer two frames, like IEEE 802.3x (link flow control), IEEE 802.1Qbb (priority based flow-control), and others of this type, are not expected and can throttle traffic between the host and the virtual switch, reducing performance. To resolve this issue, and to ensure isolation from unintended traffic streams, configure all SR-IOV or Intel Scalable IOV enabled ports for VLAN tagging from the administrative interface on the PF. This configuration allows unexpected, and potentially malicious, frames to be dropped.
- SR-IOV must be enabled in the BIOS.
- You must enable VMQ for SR-IOV to function.
- For best performance, on the host use 'Set-VMNetworkAdapter -IovQueuePairsRequested 4' on the VF to allow the virtual network to use 4 queues (maximum supported value) and assign 4 or more virtual CPUs to the connected VM. In the VM, set 'Maximum number of Receive Queues' in the VF's adapter properties to 4.
- Binding more than two virtual functions (VFs) to a virtual machine (VM) is not recommended. Binding more VFs to a VM may cause system instability.
- VMWare ESXi does not support SR-IOV on 1GbE ports.
- Some multiport adapters contain more than one controller. On these adapters, enabling SR-IOV on a port will not enable SR-IOV on all ports. Only ports bound to the same controller will be enabled.
- If SR-IOV is disabled in BIOS or the Boot Manager, enabling SR-IOV from Intel PROSet will require a system reboot.
- Due to chipset limitations, not all systems or slots support SR-IOV. Below is a chart summarizing SR-IOV support on Dell server platforms.

To change this setting in Intel PROSet

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "SR-IOV" -DisplayValue "Enabled"
```

4.32.1.3 Enabling SR-IOV on the Server

You must enable SR-IOV in the system's BIOS and HII.

To enable SR-IOV in BIOS

1. Enter the system BIOS at POST.
2. Enable Global SR-IOV.
3. Enable Virtualization Technology.
4. Save the changes and exit.

To enable SR-IOV in HII

1. During POST, press F2 to enter Device Settings.
2. Navigate to NIC and select "Device Level Settings."
3. Set Virtualization Mode to "SR-IOV."
4. Press the ESC key or click the Back button and then select "NIC Configuration."
5. If "PCI Advertised Functions" is listed, make sure the value is not 0. This number determines the number of virtual functions that are available for this port.
6. Save the changes and exit.

4.32.1.4 SR-IOV Support on Network Devices

NDC, LOM, or Adapter	100Gbe	40Gbe	25Gbe	10Gbe	1Gbe
Intel® Ethernet 100G 2P E810-C Adapter	Yes				
Intel® Ethernet 100G 2P E810-C-st Adapter	Yes				
Intel® Ethernet 100G 2P E810-C-stg Adapter	Yes				
Intel® Ethernet Converged Network Adapter XL710-Q2		Yes			
Intel® Ethernet 40G 2P XL710 QSFP+ rNDC		Yes			
Intel® Ethernet 25G 2P E810-XXV Adapter			Yes		
Intel® Ethernet 25G 2P E810-XXV OCP			Yes		
Intel® Ethernet 25G 2P E810-XXV-k Mezz			Yes		
Intel® Ethernet 25G 4P E810-XXV Adapter			Yes		
Intel® Ethernet 25G 4P E810-XXV-st Adapter			Yes		
Intel® Ethernet 25G 4P E810-XXV-stg Adapter			Yes		
Intel® Ethernet 25G 4P E810-XXV OCP			Yes		
Intel® Ethernet Connection 25G 4P E823-C LOM			Yes		
Intel® Ethernet 25G 2P XXV710 Adapter			Yes		
Intel® Ethernet 25G 2P XXV710 Mezz			Yes		
Intel® Ethernet 10G 2P X710-k bNDC				Yes	
Intel® Ethernet 10G 4P X710-k bNDC				Yes	
Intel® Ethernet 10G 2P X710-T2L-t Adapter				Yes	
Intel® Ethernet 10G 4P X710-T4L-t Adapter				Yes	
Intel® Ethernet Network Adapter X710-TL				Yes	

NDC, LOM, or Adapter	100Gbe	40Gbe	25Gbe	10Gbe	1Gbe
Intel® Ethernet Converged Network Adapter X710				Yes	
Intel® Ethernet Converged Network Adapter X710-T				Yes	
Intel® Ethernet 10G 2P X710-T2L-t OCP				Yes	
Intel® Ethernet 10G 4P X710-T4L-t OCP				Yes	
Intel® Ethernet 10G 2P X710 OCP				Yes	
Intel® Ethernet 10G 4P X710 OCP				Yes	
Intel® Ethernet Server Adapter X710-DA2 for OCP				Yes	
Intel® Ethernet 10G 4P X710/I350 rNDC				Yes	No
Intel® Ethernet 10G 4P X710 SFP+ rNDC				Yes	
Intel® Ethernet 10G 4P X550 rNDC				Yes	
Intel® Ethernet 10G 4P X550/I350 rNDC				Yes	No
Intel® Ethernet 10G 2P X550-t Adapter				Yes	
Intel® Ethernet 1G 4P I350-t OCP					Yes
Intel® Gigabit 4P I350-t rNDC					Yes
Intel® Gigabit 4P I350 bNDC					Yes
Intel® Gigabit 4P I350-t Mezz					Yes
Intel® Gigabit 2P I350-t Adapter					Yes
Intel® Gigabit 4P I350-t Adapter					Yes
PowerEdge C4130 LOMs					No
PowerEdge C6320 LOMs				Yes	
PowerEdge C6420 LOMs					No
PowerEdge T620 LOMs					No
PowerEdge T630 LOMs					No
PowerEdge FC430 LOMs				No	Yes
PowerEdge R530XD LOMs					No

SR-IOV Platform Support

To view the platforms and slots that support SR-IOV, log in to the [Dell Technologies Sales Portal](#). Once you are in the portal, you can view the products compatible with your Ethernet device:

- To view the supported slots, type "slot priority matrix" in the search box. Download and open the file for your platform of interest.
- To view the supported platforms, type "PowerEdge Server Adapter Matrix" in the search box. Download and open the linked file to view the supported servers compatible with your Ethernet device.



NOTE: All Dell 15G, 16G, and newer Intel and AMD platforms support SR-IOV.

4.32.2 Intel® Scalable I/O Virtualization Support

Intel® Scalable I/O Virtualization (Intel® Scalable IOV) allows you to share a physical device across multiple virtual machines and applications. Intel Scalable IOV provides your system the ability to share device resources with different address domains using different abstractions. For example, application processes may access a device using system calls and VMs may access a device through virtual device interfaces.

Intel Scalable IOV and SR-IOV (Single Root I/O Virtualization) are mutually exclusive. If both are enabled on your system, and all of the Intel Scalable IOV requirements are met, the PF driver will use Intel Scalable IOV. If the Intel Scalable IOV requirements are not met, the PF driver will use SR-IOV.

For more information, please refer to the [Intel Scalable I/O Virtualization Technical Specification](#) (login required).

Intel Scalable IOV is not available in the kernel driver. Download and install the current driver to use this feature. Refer to the [Customer Support section](#) for where to download the current driver.

4.32.2.1 Requirements

- Your system platform must support Intel Scalable IOV
- A network device based on an Intel(R) Ethernet 800 Series controller
- The host operating system must be a Linux distro using kernel version 5.12 - 5.15
- The host PF driver must be version 1.9.0, or later
- The guest operating system must be Linux
- The guest iAVF driver must be version 4.5.0, or later

4.32.2.2 Enabling Intel® Scalable IOV

You can use Intel's Ethernet Port Configuration Tool (EPCT) to enable Intel Scalable IOV. If the EPCT tool is not available, you can also enable Intel Scalable IOV through your system's HII interface (if it has one). The recommended method is to use the EPCT tool. To enable or disable Intel Scalable IOV using the EPCT tool, use one of these commands:

```
# epct -nic=1 -set 'siov enable'  
# epct -nic=1 -set 'siov disable'
```


Where `-nic=1` specifies the Intel Ethernet device. See the EPCT tool documentation for instructions on how to determine the NIC number of your device.

If the EPCT tool is not available, and your system has an HII interface, you can use the HII interface to enable/disable Intel Scalable IOV. Find the 'Intel Scalable IOV (Scalable IOV)' setting and select your desired value.

4.32.3 Virtual Machine Queue Offloading

Enabling VMQ offloading increases receive and transmit performance, as the adapter hardware is able to perform these tasks faster than the operating system. Offloading also frees up CPU resources. Filtering is based on MAC and/or VLAN filters.

Each Intel® Ethernet Adapter has a pool of virtual ports that are split between the various features, such as VMQ Offloading, SR-IOV, and Data Center Bridging (DCB). Increasing the number of virtual ports used for one feature decreases the number available for other features. On devices that support it, enabling DCB reduces the total pool available for other features to 32.

 **NOTE:** This does not apply to devices based on the Intel® Ethernet X710 or XL710 controllers.

For devices that support it, VMQ offloading is enabled in the host partition in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

4.32.3.1 Virtual Machine Multiple Queues

Virtual Machine Multiple Queues (VMMQ) enables Receive Side Scaling (RSS) for virtual ports attached to a physical port. This allows RSS to be used with SR-IOV and inside a VMQ virtual machine, and offloads the RSS processing to the network adapter. RSS balances receive traffic across multiple CPUs or CPU cores. This setting has no effect if your system has only one processing unit.

4.32.4 Using Intel® Network Adapters in a Microsoft* Hyper-V* Environment

When a Hyper-V Virtual NIC (vNIC) interface is created in the host OS, the vNIC takes on the MAC address of the underlying physical NIC (PF, or physical function). Since the vNIC uses the MAC address of the underlying interface, any operation that changes the MAC address of the interface (for example, setting LAA on the interface), will cause the vNIC to lose connectivity. In order to prevent this loss of connectivity, Intel PROSet will not allow you to change settings that change the MAC address.

 **NOTES:**

- When sent from inside a virtual machine, LLDP and LACP packets may be a security risk. The Intel® Virtual Function driver blocks the transmission of such packets.
- Prior to configuring the Microsoft* Hyper-V features, the Intel® NIC drivers must be installed by the Dell Update Package.

4.32.4.1 The Virtual Machine Switch

The virtual machine switch is part of the network I/O data path. It sits between the physical NIC and the virtual machine NICs and routes packets to the correct MAC address. Enabling [Virtual Machine Queue \(VMQ\) offloading](#) in Intel PROSet will automatically enable VMQ in the virtual machine switch. For driver-only installations, you must manually enable VMQ in the virtual machine switch.

4.33 Wait for Link

Determines whether the driver waits for auto-negotiation to be successful before reporting the link state. If this feature is off, the driver does not wait for auto-negotiation. If the feature is on, the driver does wait for auto-negotiation.

If this feature is on and the speed is not set to auto-negotiation, the driver will wait for a short time for link to be established before reporting the link state.

If the feature is set to **Auto Detect**, this feature is automatically set to **On** or **Off** depending on speed and adapter type when the driver is installed. The setting is:

- Off for copper Intel gigabit adapters with a speed of "Auto"
- On for copper Intel gigabit adapters with a forced speed and duplex
- On for fiber Intel gigabit adapters with a speed of "Auto"

To change this setting in Intel PROSet

Default	Auto Detect
Range	<ul style="list-style-type: none">• On• Off• Auto Detect

This setting is found in the Adapter Settings panel in Intel PROSet Adapter Configuration Utility.

To change this setting in Windows PowerShell, use the Set-IntelNetAdapterSetting cmdlet. For example:

```
Set-IntelNetAdapterSetting -Name "<adapter_name>" -DisplayName "Wait for Link" -  
DisplayValue "Off"
```


5. Microsoft* Windows* Driver and Software Installation and Configuration

This chapter explains the following in more detail:

- How to [install device drivers](#) on supported versions of Microsoft Windows
- How to [save and restore](#) a device's configuration settings

Refer to "About Intel PROSet®" on page 13 for an overview of this software, how to install it, and how to use it to configure device features in Microsoft Windows.

5.1 Installing Windows Drivers and Software

The Dell Update Package (DUP) is an executable package that will install or update the network drivers on supported Microsoft Windows systems. You can install the DUP using the [Windows install wizard](#) or from the [command line](#).



NOTE: To successfully install or uninstall the drivers or software, you must have administrative privileges on the computer completing installation.

5.1.1 Install the Base Drivers

This section describes how to install the base drivers.



NOTES:

- This will update the drivers for all supported Intel® network adapters in your system.
- The Roll Back Driver feature of Windows Server (available on the Adapter Properties dialog's **Driver** tab) will not work correctly if Intel PROSet is present on the system. Before you use the Roll Back Driver feature, remove any teams. Then remove Intel PROSet using **Programs and Features** from the Control Panel of Windows.
- Using Microsoft Windows Update to upgrade or downgrade your Ethernet network drivers is not supported. Please download the latest driver package from the [support website](#).

Before installing or updating the drivers, insert your adapter(s) in the computer and plug in the network cable. When Windows discovers the new adapter, it attempts to find an acceptable Windows driver already installed with the operating system.

If found, the driver is installed without any user intervention. If Windows cannot find the driver, the Found New Hardware Wizard window is displayed.

Regardless of whether Windows finds the driver, it is recommended that you follow the procedures below to install the driver. Drivers for all Intel adapters supported by this software release are installed.

1. Download the latest drivers from the [support website](#) and transfer them to the system.
2. If the Found New Hardware Wizard screen is displayed, click **Cancel**.
3. Double-click the downloaded file.
4. Select **Install** from the Dell Update Package screen.
5. Follow the prompts in the install wizard.

NOTE:

- During a fresh install, [Intel PROSet](#) is automatically installed along with the base drivers.
- During a driver upgrade or downgrade, features that were previously installed (such as DCB/iSCSI, Intel PROSet, and device drivers) will be upgraded or downgraded.
- The RDMA driver is not automatically installed even if an RDMA-capable device is in the system. Refer to "Installing the RDMA Driver" below for additional instructions.

5.1.2 Installing the RDMA Driver

You must install the [base drivers](#) before you can install the RDMA driver.

1. Install the base drivers as described in "Installing Windows Drivers and Software" on the previous page.
2. After successfully installing the base drivers, open **Add or Remove Programs**.
3. In the list of apps and features, select **Intel Network Connections** and click the **Modify** button.
4. In the wizard, select the RDMA features you want to install or modify.

Refer to "Remote Direct Memory Access (RDMA)" on page 49 for information on configuring RDMA in Windows.

5.1.3 Installing from the Command Line

You can run the Dell Update Package (DUP) from the command line to install or update the network drivers on your system. This section describes the syntax and examples for installing the DUP from the command line in Windows.

- NOTE:** If you are installing a driver in a computer with existing Intel adapters, be sure to update all the adapters and ports with the same driver and Intel PROSet software. This ensures that all adapters will function correctly.

5.1.3.1 Syntax

Use the following syntax with the DUP:

```
Network_Driver_XXXXX_WN64_XX.X.X_A00.exe [/<option1>[=<value1>]] [/<option2>
[=<value2>]]...
```

For a description of the available options, use the `/?`, `/h`, or `/help` switch.

5.1.3.2 Examples

Description	Example
Update the system silently	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s</code>
Fresh install silently	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s /i</code>
Extract the update contents to the folder <code>C:\mydir</code>	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s /e=C:\mydir</code>
Extract the driver components to the folder <code>C:\mydir</code>	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s /drivers=C:\mydir</code>

Description	Example
Only install driver components	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s /driveronly</code>
Change from the default log location to C:\my path with spaces\log.txt	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s /l="C:\my path with spaces\log.txt"</code>
Force update to continue, even on "soft" qualification errors	<code>Network_Driver_XXXXX_WN64_XX.X.X_A00.exe /s /f</code>

5.1.4 Downgrading Drivers

You can use the `/s` and `/f` options to downgrade your drivers. For example, if you have the 21.5.0 drivers loaded and you want to downgrade to 20.5.0, type the following:

```
Network_Driver_XXXXX_WN64_20.5.0_A00.exe /s /f
```

5.2 Saving and Restoring an Adapter's Configuration Settings

The Save and Restore Command Line Tool allows you to copy the current adapter settings into a standalone file (such as on a USB drive) as a backup measure. In the event of a hard drive failure, you can reinstate most of your former settings.

The system on which you restore network configuration settings must have the same configuration as the one on which the save was performed. A saved configuration file can be used to restore adapter settings after an operating system upgrade. However, all adapter configuration settings may not be restored depending on the features supported by the new operating system or adapter configuration software.




NOTES:

- Only adapter settings are saved. The adapter's driver is not saved.
- Restore using the script only once. Restoring multiple times may result in unstable configuration.
- Intel PROSet must be installed for the SaveRestore.ps1 script to run.
- For systems running a 64-bit OS, be sure to run the 64-bit version of Windows PowerShell, not the 32-bit (x86) version, when running the SaveRestore.ps1 script.

5.2.1 Command Line Syntax

```
SaveRestore.ps1 -Action save|restore [-ConfigPath] [-BDF]
```

SaveRestore.ps1 has the following command line options:

Option	Description
-Action	<p>Required. Valid values: save restore.</p> <p>The save option saves adapter settings that have been changed from the default settings. When you restore with the resulting file, any settings not contained in the file are assumed to be the default.</p> <p>The restore option restores the settings.</p>
-ConfigPath	<p>Optional. Specifies the path and filename of the main configuration save file. If not specified, it is the script path and default filename (<code>saved_config.txt</code>).</p>
-BDF	<p>Optional. Default configuration file names are <code>saved_config.txt</code> and <code>Saved_StaticIP.txt</code>.</p> <p>If you specify <code>-BDF</code> during a restore, the script attempts to restore the configuration based on the PCI Bus:Device:Function:Segment values of the saved configuration. If you removed, added, or moved a NIC to a different slot, this may result in the script applying the saved settings to a different device.</p> <p> NOTES:</p> <ul style="list-style-type: none"> • If the restore system is not identical to the saved system, the script may not restore any settings when the <code>-BDF</code> option is specified. • Virtual Function devices do not support the <code>-BDF</code> option.

5.2.2 Examples

Save Example

To save the adapter settings to a file on a removable media device, do the following.

1. Open a Windows PowerShell Prompt.
2. Navigate to the directory where `SaveRestore.ps1` is located (generally `c:\Program Files\Intel\Wired Networking\PROSET`).
3. Type the following:

```
SaveRestore.ps1 -Action Save -ConfigPath e:\settings.txt
```

Restore Example

To restore the adapter settings from a file on removable media, do the following:

1. Open a Windows PowerShell Prompt.
2. Navigate to the directory where `SaveRestore.ps1` is located (generally `c:\Program Files\Intel\Wired Networking\PROSET`).
3. Type the following:

```
SaveRestore.ps1 -Action Restore -ConfigPath e:\settings.txt
```

6. Linux* Driver Installation and Configuration

6.1 Overview

This release includes Linux Base Drivers for Intel® Network Connections. These drivers are only supported as a loadable module. Intel is not supplying patches against the kernel source to allow for static linking of the driver.



NOTE: On systems running Linux, the base driver must be loaded for the Dell FW DUP to function correctly.

6.1.1 Supported Linux Distributions

The drivers in this release support the Linux distributions listed in "Supported Operating Systems" on page 2.

6.1.2 Driver Names

The following tables show the names for Linux drivers. See the [Supported Adapters](#) section below to determine which driver to use for your specific device.

6.1.2.1 Linux Base Drivers

Driver Name	Supported Controllers	For More Information
ice	800 Series (E810 and E823)	"ice Linux Driver for the Intel Ethernet 800 Series" on page 88
i40e	700 Series (X710, XL710, and XXV710)	"i40e Linux Driver for the Intel Ethernet 700 Series" on page 119
ixgbe	500 Series (X550)	"ixgbe Linux Driver for Intel Ethernet 10 Gigabit Server Adapters" on page 148
igb	I350	"igb Linux Driver for Intel Ethernet Gigabit Adapters" on page 161

6.1.2.2 Linux VF Drivers


This release also includes support for SR-IOV drivers. SR-IOV requires the correct platform and OS support. See "Single Root I/O Virtualization (SR-IOV)" on page 57 for more information.

Driver Name	Supported Controllers	For More Information
iavf	800 Series 700 Series	"iavf Linux Driver" on page 143
ixgbev	500 Series	"ixgbev Linux Driver for Intel Ethernet 10 Gigabit Server Adapters" on page 159
igb	I350	Note: SR-IOV support is provided via the inbox driver

6.1.3 Supported Adapters

The following Intel network adapters are compatible with the drivers in this release. Refer to the subsections in this chapter for more information on each driver.

For more information on how to identify your adapter or for the latest network drivers for Linux, see "Customer Support" on page 6.

 **NOTE:** The branding strings below may not match what the driver displays in the OS. The device name may vary depending on the contents of the system's `pci.ids` file. You can find the worldwide PCI ID Repository at <https://pci-ids.ucw.cz/>. Most Linux distributions include the `update-pciids` utility, which you can use to retrieve a current copy of the database and then install it on your system.

6.1.3.1 Linux Base Drivers

Driver Name	Supported Devices
ice	<ul style="list-style-type: none"> • Intel® Ethernet 100G 2P E810-C-st Adapter • Intel® Ethernet 100G 2P E810-C-stg Adapter • Intel® Ethernet 100G 2P E810-C Adapter • Intel® Ethernet 25G 2P E810-XXV OCP • Intel® Ethernet 25G 4P E810-XXV OCP • Intel® Ethernet 25G 2P E810-XXV-k Mezz • Intel® Ethernet 25G 4P E810-XXV-st Adapter • Intel® Ethernet 25G 4P E810-XXV-stg Adapter • Intel® Ethernet 25G 2P E810-XXV Adapter • Intel® Ethernet 25G 4P E810-XXV Adapter • Intel® Ethernet Connection 25G 4P E823-C LOM
i40e	<ul style="list-style-type: none"> • Intel® Ethernet Converged Network Adapter XL710-Q2 • Intel® Ethernet 40G 2P XL710 QSFP+ rNDC • Intel® Ethernet 25G 2P XXV710 Adapter • Intel® Ethernet 25G 2P XXV710 Mezz • Intel® Ethernet 10G 4P X710-k bNDC • Intel® Ethernet 10G 2P X710-k bNDC • Intel® Ethernet Converged Network Adapter X710 • Intel® Ethernet Converged Network Adapter X710-T • Intel® Ethernet 10G 4P X710/I350 rNDC • Intel® Ethernet 10G 4P X710 SFP+ rNDC • Intel® Ethernet Server Adapter X710-DA2 for OCP • Intel® Ethernet 10G 2P X710 OCP • Intel® Ethernet 10G 4P X710 OCP • Intel® Ethernet 10G 2P X710-T2L-t OCP • Intel® Ethernet 10G 4P X710-T4L-t OCP • Intel® Ethernet 10G 2P X710-T2L-t Adapter • Intel® Ethernet 10G 4P X710-T4L-t Adapter
ixgbe	<ul style="list-style-type: none"> • Intel® Ethernet 10G 2P X550-t Adapter • Intel® Ethernet 10G 4P X550 rNDC • Intel® Ethernet 10G 4P X550/I350 rNDC
igb	<ul style="list-style-type: none"> • Intel® Ethernet 1G 4P I350-t OCP • Intel® Gigabit 4P X550/I350 rNDC • Intel® Gigabit 4P I350-t rNDC

Driver Name	Supported Devices
	<ul style="list-style-type: none">• Intel® Gigabit 4P I350-t Mezz• Intel® Gigabit 4P X710/I350 rNDC• Intel® Gigabit 4P I350 bNDC• Intel® Gigabit 2P I350-t Adapter• Intel® Gigabit 4P I350-t Adapter• Intel® Gigabit 2P I350-t LOM• Intel® Gigabit I350-t LOM• Intel® Gigabit 2P I350 LOM

6.1.3.2 Linux VF Drivers

The following drivers support the listed virtual function devices that can only be activated on kernels that support SR-IOV.

Driver Name	Supported Devices
iavf	<ul style="list-style-type: none"> • Intel® Ethernet 100G 2P E810-C-st Adapter • Intel® Ethernet 100G 2P E810-C-stg Adapter • Intel® Ethernet 100G 2P E810-C Adapter • Intel® Ethernet 40G 2P XL710 QSFP+ rNDC • Intel® Ethernet Converged Network Adapter XL710-Q2 • Intel® Ethernet 25G 2P E810-XXV OCP • Intel® Ethernet 25G 4P E810-XXV OCP • Intel® Ethernet 25G 2P E810-XXV-k Mezz • Intel® Ethernet 25G 4P E810-XXV-st Adapter • Intel® Ethernet 25G 4P E810-XXV-stg Adapter • Intel® Ethernet 25G 2P E810-XXV Adapter • Intel® Ethernet 25G 4P E810-XXV Adapter • Intel® Ethernet Connection 25G 4P E823-C LOM • Intel® Ethernet 25G 2P XXV710 Adapter • Intel® Ethernet 25G 2P XXV710 Mezz • Intel® Ethernet 10G 4P X710-k bNDC • Intel® Ethernet 10G 2P X710-k bNDC • Intel® Ethernet Converged Network Adapter X710 • Intel® Ethernet Converged Network Adapter X710-T • Intel® Ethernet 10G 4P X710/I350 rNDC • Intel® Ethernet 10G 4P X710 SFP+ rNDC • Intel® Ethernet Server Adapter X710-DA2 for OCP • Intel® Ethernet 10G 2P X710 OCP • Intel® Ethernet 10G 4P X710 OCP • Intel® Ethernet 10G 2P X710-T2L-t OCP • Intel® Ethernet 10G 4P X710-T4L-t OCP • Intel® Ethernet 10G 2P X710-T2L-t Adapter • Intel® Ethernet 10G 4P X710-T4L-t Adapter
ixgbevf	<ul style="list-style-type: none"> • Intel® Ethernet 10G 2P X550-t Adapter • Intel® Ethernet 10G 4P X550 rNDC • Intel® Ethernet 10G 4P X550/I350 rNDC

6.2 Building and Installation

There are three methods for installing Linux device drivers:

- From source code
- Using a KMOD RPM Package Manager (RPM)
- Using a Kernel Module Package (KMP) RPM

Supported distributions are limited to those listed in "Supported Operating Systems" on page 2. Note the following for each method:

Format	Supported Distributions	Description
Source code	Any supported distribution	<ul style="list-style-type: none"> This format packages the driver as a single archive file [<code>*.tar.gz</code>]. You must unpack, compile, and install it into the file system before you can use the driver.
KMOD RPM	RHEL only	<ul style="list-style-type: none"> This format is a precompiled binary using Red Hat's KMOD RPM format specification. You must match the KMOD RPM to the version of RHEL on the target system. One KMOD RPM file [<code>*.x86_64.rpm</code>] is available for each supported version of RHEL.
KMP RPM	SLES only	<ul style="list-style-type: none"> This format is a precompiled binary targeted solely for SLES systems. You must match the KMP and RPM file set to the version of SLES on the target system. This format uses a set of two RPM files per driver for each supported version of SLES. These RPM files include: <ul style="list-style-type: none"> KMP RPM that contains the actual kernel driver object module [<code>*.ko</code>] file Driver support files, such as the man page, driver README, GPL license files, and other items. The KMP RPM sets are created, built, and digitally signed in a partnership with SUSE as part of the SUSE SolidDriver Program.

6.2.1 Driver Package Contents

To review the contents of archives or RPMs before extracting or installing files to your system, do the following:

- **For source code:** Use the `tar -tf` or `tar -tvf` command to list the contents of the source archive file. For example:

```
# tar -tf iavf-4.0.2.tar.gz
```

- **For RPM files:** Use the following options with the `rpm` command to get more information about an RPM file. The examples below are for SLES KMP RPM files; replace the file names shown below with the name of your desired RPM file.
 - To query and list the contents of an RPM file:

```
# rpm -ql -p intel-ice-kmp-default-1.3.2_k5.3.18_22-1.2.x86_64.rpm
```

- To display information about the RPM file:

```
# rpm -qi -p intel-ice-kmp-default-1.3.2_k5.3.18_22-1.2.x86_64.rpm
```

- To discover the capabilities provided by an RPM file:

```
# rpm -q --provides -p intel-iavf-4.0.2-3.1.x86_64.rpm
```

- To view the prerequisite requirements of an RPM file:

```
# rpm -q --requires -p intel-iaavf-kmp-default-4.0.2_
k5.3.18_22-3.1.x86_64.rpm
```

6.2.2 Installing from Source Code

Building and installing a driver from the original source code is the most flexible method and provides the most freedom for installation. However, it is also the most difficult.

Installing from source code requires development tool and header file support packages be installed on the system. (Consult the documentation provided with your Linux distribution or your support channel for more information on identifying or installing these packages.) **We recommend installing and using the binary RPM driver packages for your system.**



NOTE:

- For the build to work properly, the currently running kernel **must MATCH** the version and configuration of the installed kernel source and header files. If you have just updated or installed a new kernel, reboot the system.
- **You must be logged in as root to install the driver.**
- Use the appropriate filename of the driver (see the driver names listed under "Supported Adapters" on page 70 for more information).

Do the following to install the driver from source code:

1. Download the base driver tar file to the directory of your choice. For example, use `/home/username/ixgbe` OR `/usr/local/src/ixgbe`.
2. Untar/unzip the archive, where `<x.x.x>` is the version number for the driver tar. For example:

```
# tar xf ixgbe-<x.x.x>.tar.gz
```

3. Change to the driver src directory, where `<x.x.x>` is the version number for the driver tar. For example:

```
# cd ixgbe-<x.x.x>/src/
```

4. Compile the driver module:

```
# make install
```

The binary will be installed as:

```
/lib/modules/<KERNEL_
VERSION>/updates/drivers/net/ethernet/intel/ixgbe/ixgbe.ko
```

The install location listed above is the default location. This might differ for various Linux distributions.

NOTE:

- **For ice devices:**

- To build the driver using the schema for unified ethtool statistics defined in <https://sourceforge.net/p/e1000/wiki/Home/>, use the following command (note: this will also apply the `make install` command):

```
# make CFLAGS_EXTRA='-DUNIFIED_STATS' install
```

- To compile the driver with ADQ (Application Device Queues) flags set, use the following command, where `<nproc>` is the number of logical cores (note: this will also apply the `make install` command):

```
# make -j<nproc> CFLAGS_EXTRA='-DADQ_PERF_
COUNTERS' install
```

- **For ice and i40e devices:** You may see warnings from `depmod` related to unknown RDMA symbols during the make of the OOT base driver. These warnings are normal and appear because the in-tree RDMA driver will not work with the OOT base driver. To address the issue, you need to install the latest OOT versions of the base and RDMA drivers.
- **For igb devices:** Some systems have trouble supporting MSI and/or MSI-X interrupts. If your system needs to disable this type of interrupt, the driver can be built and installed with the command:

```
# make CFLAGS_EXTRA=-DDISABLE_PCI_MSI install
```

Normally, the driver generates an interrupt every two seconds. If interrupts are not received in `cat /proc/interrupts` for the `ethX` device, then this workaround may be necessary.

5. Check the status of the newly installed driver. For example:

```
# modinfo ixgbe
```

6. Make the newly installed driver active in the running kernel. You can do this by rebooting the system or using the `rmmmod` or `modprobe` commands. For example:

```
# shutdown -r now
```

or

```
# rmmmod ixgbe; sleep 1; modprobe ixgbe
```

6.2.3 Installing Using KMOD RPM (RHEL only)

RPMs are provided for the RHEL distributions listed in "Supported Operating Systems" on page 2. The following table describes the naming convention for KMOD RPM files.

RPM File Name	Naming Convention
<code>kmod-<driver name>-<driver version>-<rpm rel>.<arch type>.rpm</code>	<p>For example, in <code>kmod-ixgbe-5.9.4-201130.x86_64.rpm</code>:</p> <ul style="list-style-type: none"> • <code>ixgbe</code> is the driver name • <code>5.9.4</code> is the driver version • <code>201130</code> is the RPM release version • <code>x86_64</code> is the architecture type

Do the following to install the KMOD RPM:

 **NOTE: You must be logged in as root to install the driver.**

1. Download the KMOD RPM file to your system.
2. Install the KMOD RPM using the `rpm` command. For example:

```
# rpm -i kmod-ixgbe-5.9.4-201130.x86_64.rpm
```

3. Check the status of the newly installed driver. For example:

```
# modinfo ixgbe
```

4. Make the newly installed driver active in the running kernel. You can do this by rebooting the system or using the `rmmmod` or `modprobe` commands. For example:

```
# shutdown -r now
or
```

```
# rmmmod ixgbe; sleep 1; modprobe ixgbe
```

6.2.4 Installing Using KMP RPM (SLES only)

The `intel-<driver>*.rpm` is a prerequisite for the `intel-<driver>-kmp-default-*.rpm` package. If the normal RPM is not installed, the KMP will not install..

A pair of RPM files is provided for each SLES distribution listed in "Supported Operating Systems" on page 2. The following table describes the naming convention for the KMP RPM files.

RPM File Name	Naming Convention
<code>intel-<driver name>-<driver version>-<rpm rel>.<arch type>.rpm</code>	<p>For example, in <code>intel-ixgbe-5.9.4-2.1.x86_64.rpm</code>:</p> <ul style="list-style-type: none"> • <code>ixgbe</code> is the driver name • <code>5.9.4</code> is the driver version • <code>2.1</code> is the RPM release number • <code>x86_64</code> is the architecture type
<code>intel-<driver name>-kmp-default-<driver version>_k<kernel version>_<rpm rel>.<arch type>.rpm</code>	<p>For example, in <code>intel-ixgbe-kmp-default-5.9.4_k5.3.18_22-2.1.x86_64.rpm</code>:</p> <ul style="list-style-type: none"> • <code>ixgbe</code> is the driver name • <code>5.9.4</code> is the driver version • <code>k5.3.18</code> is the kernel version • <code>22-2.1</code> is the RPM release number • <code>x86_64</code> is the architecture type

Do the following to install the KMP RPMs:

 **NOTE:**

- **You must be logged in as root to install the driver.**
- The `rpm` command will not help you if the dependencies of the package are not met at installation time. It will then refuse to install the package to avoid having the system in an inconsistent state. The `zypper` command will automatically find the required packages and retrieve them.

1. Download the two RPM files to your system.
2. Install both RPM files using the `rpm` or `zypper` command. For example:

```
# rpm -iv intel-ixgbe-5.9.4-2.1.x86_64.rpm \  
intel-ixgbe-kmp-default-5.9.4_k5.3.18_22-2.1.x86_64.rpm
```

or

```
# zypper install intel-ixgbe-5.9.4-2.1.x86_64.rpm \  
intel-ixgbe-kmp-default-5.9.4_k5.3.18_22-2.1.x86_64.rpm
```

3. Check the status of the newly installed driver. For example:

```
# modinfo ixgbe
```

4. Make the newly installed driver active in the running kernel. You can do this by rebooting the system or using the `rmmmod` or `modprobe` commands. For example:

```
# shutdown -r now
```

or

```
# rmmmod ixgbe; modprobe ixgbe
```

6.2.5 Linux Secure Boot Mode

This release includes RPM packages with signed kernel modules that support Linux Secure Boot operation. Packages for Red Hat Enterprise Linux (RHEL) distributions contain modules signed with an Intel private key. Packages for SUSE Linux Enterprise Server (SLES) distributions contain modules signed with SUSE's key.

Public keys necessary to authenticate the signed driver in Secure Boot mode may be included in the packages (RHEL) or available from the OS vendor (SLES). To authenticate the signed driver, you must place the public key in the UEFI Secure Boot key database.

Source RPM (SRPM) with complete source code and SPEC file are provided in `*.src.rpm` packages.

If you decide to recompile the `.ko` module from the provided source files, the new `.ko` module will not be signed with any key. To use this `.ko` module in Secure Boot mode, you must sign it yourself with your own private key and add your public key to the UEFI Secure Boot key database.

The driver kernel module for a specific kernel version can be used with errata kernels within the same minor OS version, unless the errata kernel broke KABI. Whenever you update your kernel with an errata kernel, you must reinstall the driver RPM package.

6.2.5.1 Secure Boot in RHEL

RPM packages built for RHEL distributions include:

- Device driver signed with Intel's private key in precompiled kernel module form
- Complete source code for above driver
- Intel's public key

See the following reference documentation from Red Hat for information on configuring your system for Secure Boot operation:

- Red Hat Enterprise Linux 8 - Managing, monitoring, and updating the kernel (Chapter 4)
- Red Hat Enterprise Linux 9 - Managing, monitoring, and updating the kernel (Chapter 3, Chapter 4)

6.2.5.2 Secure Boot in SLES

RPM packages built for SLES distributions include:

- Device driver signed with SUSE's Package Signing Key in precompiled kernel module form
- Complete source code for above driver

SUSE's public key for signed packages is available from https://drivers.suse.com/doc/Usage/Package_Signing_Key.html.

See the following reference documentation from SUSE for information on configuring your system for Secure Boot operation:

- SUSE Linux Enterprise Server 15 SP4 - Administration Guide (Chapter 17)
- SUSE Linux Enterprise Server 15 SP5 - Administration Guide (Chapter 17)

6.3 Important Notes (All Drivers)

Unless noted otherwise, this section describes important notes that apply to all Intel Ethernet drivers. You can find additional information for specific drivers in "Driver-Specific Information" on page 88.

Configuring SR-IOV for improved network security

In a virtualized environment, on Intel® Server Adapters that support SR-IOV or Intel® Scalable I/O Virtualization (Intel® Scalable IOV), the virtual function (VF) may be subject to malicious behavior. Software-generated layer two frames, like IEEE 802.3x (link flow control), IEEE 802.1Qbb (priority based flow-control), and others of this type, are not expected and can throttle traffic between the host and the virtual switch, reducing performance. To resolve this issue, and to ensure isolation from unintended traffic streams, configure all SR-IOV or Intel Scalable IOV enabled ports for VLAN tagging from the administrative interface on the PF. This configuration allows unexpected, and potentially malicious, frames to be dropped.

Do not unload port driver if VF with active VM is bound to it

Do not unload a port's driver if a Virtual Function (VF) with an active Virtual Machine (VM) is bound to it. Doing so will cause the port to appear to hang. Once the VM shuts down, or otherwise releases the VF, the command will complete.

6.4 Additional Configurations (All Drivers)

Unless noted otherwise, this section describes configuration information that applies to all Intel Ethernet drivers. You can find additional configuration information for specific drivers in "Driver-Specific Information" on page 88.

Use `ethtool`, `lspci`, or the `ip` command to obtain more information about the driver.

6.4.1 ethtool

Intel Ethernet drivers use the `ethtool` interface for driver configuration and diagnostics, as well as displaying statistical information. The latest `ethtool` version is required for this functionality. Download it at <https://kernel.org/pub/software/network/ethtool/>.

6.4.2 Viewing Link Messages

Link messages will not be displayed to the console if the distribution is restricting system messages. In order to see network driver link messages on your console, set `dmesg` to eight by entering the following:

```
# dmesg -n 8
```



NOTE: This setting is not saved across reboots.

6.4.3 Configuring the Driver on Different Distributions

Configuring a network driver to load properly when the system is started is distribution dependent. Typically, the configuration process involves adding an alias line to `/etc/modules.conf` or `/etc/modprobe.conf` as well as editing other system start up scripts and/or configuration files. Many popular Linux distributions ship with tools to make these changes for you. To learn the proper way to configure a network device for your system, refer to your distribution documentation.

6.4.4 NAPI

Some drivers support NAPI (Rx polling mode). For more information on NAPI, see <https://wiki.linuxfoundation.org/networking/napi>.

6.4.5 Wake on LAN (WoL) Support

Some adapters do not support Wake on LAN (WoL). To determine if your adapter supports WoL, run the following command:

```
# ethtool <ethX>
```

WoL is configured through the `ethtool` utility. For instructions on enabling WoL with `ethtool`, refer to <https://kernel.org/pub/software/network/ethtool/>.

WoL will be enabled on the system during the next shutdown or reboot. To enable WoL, the driver must be loaded prior to shutting down or suspending the system.

6.4.6 Configuring VLAN Tagging on SR-IOV Enabled Adapter Ports

To configure VLAN tagging for the ports on an SR-IOV enabled adapter, use the following command. The VLAN configuration should be done before the VF driver is loaded or the VM is booted. The VF is not aware of the VLAN tag being inserted on transmit and removed on received frames (sometimes called "port VLAN" mode).

```
# ip link set dev <ethX> vf <id> vlan <vlan id>
```

For example, the following will configure PF `eth0` and the first VF on VLAN 10:

```
# ip link set dev eth0 vf 0 vlan 10
```

6.4.7 Jumbo Frames

Jumbo Frames support is enabled by changing the MTU to a value larger than the default of 1500 bytes. Use the `ifconfig` command to increase the MTU size. For example, enter the following where `<ethX>` is the interface number:

```
# ifconfig <ethX> mtu 9000 up
```

Alternatively, you can use the `ip` command as follows:

```
# ip link set mtu 9000 dev <ethX>
```



```
# ip link set up dev <ethX>
```

This setting is not saved across reboots. Add 'MTU=9000' to the following file to make the setting change permanent:

- /etc/sysconfig/network-scripts/ifcfg-<ethX> for RHEL
- /etc/sysconfig/network/<config_file> for SLES

NOTES:

- The maximum MTU setting for Jumbo Frames is 9710 bytes. This value coincides with the maximum Jumbo Frames size of 9728 bytes.
- Packet loss may have a greater impact on throughput when you use jumbo frames. If you observe a drop in performance after enabling jumbo frames, enabling flow control may mitigate the issue.

6.4.8 Link-Level Flow Control (LFC)

Ethernet Flow Control (IEEE 802.3x) can be configured with ethtool to enable receiving and transmitting pause frames for this driver. When transmit is enabled, pause frames are generated when the receive packet buffer crosses a predefined threshold. When receive is enabled, the transmit unit will halt for the time delay specified when a pause frame is received.


NOTES:

- You must have a flow control capable link partner.
- The PF driver requires flow control on both the port and link partner. If flow control is disabled on one of the sides, the port may appear to hang on heavy traffic.
- For 82598 backplane cards entering 1 gigabit mode, flow control default behavior is changed to off. Flow control in 1 gigabit mode on these devices can lead to transmit hangs.

Use ethtool to change the flow control settings.


To enable or disable Rx or Tx Flow Control:

```
# ethtool -A <ethX> rx <on|off> tx <on|off>
```

 **NOTE:** This command only enables or disables Flow Control if auto-negotiation is disabled. If auto-negotiation is enabled, this command changes the parameters used for auto-negotiation with the link partner.

To enable or disable auto-negotiation on i40e, ixgbe, or igb devices:

```
# ethtool -s <ethX> autoneg <on|off>
```

 **NOTE:** Flow Control auto-negotiation is part of link auto-negotiation. Depending on your device, you may not be able to change the auto-negotiation setting.

6.4.9 Intel® Ethernet Flow Director

On devices that support the feature, the Intel Ethernet Flow Director performs the following tasks:


- Directs receive packets according to their flows to different queues
- Enables tight control on routing a flow in the platform
- Matches flows and CPU cores for flow affinity
- Supports multiple parameters for flexible flow classification and load balancing (support may vary; in SFP mode only)

An included script (set_irq_affinity) automates setting the IRQ to CPU affinity.

The following table summarizes supported Intel Ethernet Flow Director features across Intel® Ethernet controllers.


Feature	500 Series	700 Series	800 Series
VF Flow Director	Supported	Routing to VF not supported	Not supported
IP Address Range Filter	Supported	Not supported	Field masking
IPv6 Support	Supported	Supported	Supported
Configurable Input Set	Configured per port	Configured globally	Configured per port
ATR	Supported	Supported	Not supported
Flex Byte Filter	Starts at beginning of packet	Starts at beginning of payload	Starts at beginning of packet
Tunneled Packets	Filter matches outer header	Filter matches Inner header	Filter matches inner header

6.4.9.1 Intel Ethernet Flow Director Filters

 **NOTE:** This functionality is supported only on Intel Ethernet 500 Series and newer devices.

Filters are used to direct traffic that matches specified characteristics. They are enabled through ethtool's ntuple interface. To enable or disable the Intel Ethernet Flow Director and these filters:

```
# ethtool -K <ethX> ntuple <off|on>
```

 **NOTE:** When you disable ntuple filters, all the user programmed filters are flushed from the driver cache and hardware. All needed filters must be re-added when ntuple is re-enabled.

To display all of the active filters:

```
# ethtool -u <ethX>
```

To add a new filter:

```
# ethtool -U <ethX> flow-type <type> src-ip <ip> [m <ip_mask>] dst-  
ip <ip> [m <ip_mask>] src-port <port> [m <port_mask>] dst-port  
<port> [m <port_mask>] action <queue>
```

Where:

- <ethX> - the Ethernet device to program
- <type> - can be ip4, tcp4, udp4, sctp4, ip6, tcp6, udp6, sctp6 (depending on the device's capabilities)
- <ip> - the ip address to match on
- <ip_mask> - the IPv4 address to mask on
 - Note: These filters use inverted masks.
 - For i40e devices, address masks can be either all 0 (to ignore a match) or all F (for a full match).
 - For ice devices, an inverted mask with 0 means exactly match while with 0xF means DON'T CARE. Please refer to the examples for more details about inverted masks.
- <port> - the port number to match on
- <port_mask> - the 16-bit integer for masking
 - Note: These filters use inverted masks. For i40e devices, port masks can be either all 0 (to ignore a match) or all F (for a full match).
- <queue> - the queue to direct traffic towards (-1 discards the matched traffic)

To delete a filter:

```
# ethtool -U <ethX> delete <N>
```

Where <N> is the filter ID displayed when printing all the active filters, and may also have been specified using `loc <N>` when adding the filter.



NOTE: For ixgbe devices, Intel Ethernet Flow Director masking works in the opposite manner from subnet masking. For instance, in the following command:

```
# ethtool -U eth11 flow-type ip4 src-ip 172.4.1.2 m 255.0.0.0
dst-ip \
172.21.1.1 m 255.128.0.0 action 31
```

The src-ip value that is written to the filter will be 0.4.1.2, not 172.0.0.0 as might be expected. Similarly, the dst-ip value written to the filter will be 0.21.1.1, not 172.0.0.0.

Examples:

To add a filter that directs packet to queue 2:

```
# ethtool -U <ethX> flow-type tcp4 src-ip 192.168.10.1 dst-ip
192.168.10.2 src-port 2000 dst-port 2001 action 2 [loc 1]
```

To set a filter using only the source and destination IP address:

```
# ethtool -U <ethX> flow-type tcp4 src-ip 192.168.10.1 dst-ip
192.168.10.2 action 2 [loc 1]
```

On i40e and ice devices: To set a filter based on a user-defined pattern and offset:

```
# ethtool -U <ethX> flow-type tcp4 src-ip 192.168.10.1 dst-ip
192.168.10.2 user-def 0x4FFFF action 2 [loc 1]
```


where the value of the user-def field contains the offset (4 bytes) and the pattern (0xffff).

To match TCP traffic sent from 192.168.0.1, port 5300, directed to 192.168.0.5, port 80, and then send it to queue 7:

```
# ethtool -U enp130s0 flow-type tcp4 src-ip 192.168.0.1 dst-ip 192.168.0.5
src-port 5300 dst-port 80 action 7
```

To add a TCPv4 filter with a partial mask for a source IP:

```
# ethtool -U <ethX> flow-type tcp4 src-ip 192.168.0.0 m 0.255.255.255 dst-ip
192.168.5.12 src-port 12600 dst-port 31 action 12
```

 **NOTE:** For ice devices, this is for a source IP subnet. Here the matched `src-ip` is `192.*.*.*` (inverted mask).

For each flow-type, the programmed filters must all have the same matching input set. For example, issuing the following two commands is acceptable:

```
# ethtool -U enp130s0 flow-type ip4 src-ip 192.168.0.1 src-port 5300 action 7
# ethtool -U enp130s0 flow-type ip4 src-ip 192.168.0.5 src-port 55 action 10
```


Issuing the next two commands, however, is not acceptable, since the first specifies `src-ip` and the second specifies `dst-ip`:

```
# ethtool -U enp130s0 flow-type ip4 src-ip 192.168.0.1 src-port 5300 action 7
# ethtool -U enp130s0 flow-type ip4 dst-ip 192.168.0.5 src-port 55 action 10
```

The second command will fail with an error. You may program multiple filters with the same fields, using different values, but, on one device, you may not program two tcp4 filters with different matching fields.

Matching on a subportion of a field is not supported by the driver, thus partial mask fields are not supported.

6.4.9.2 Filters to Direct Traffic to a Specific VF

 **NOTE:** This functionality is supported only on Intel Ethernet 500 Series and 700 Series devices.

It is possible to create filters that direct traffic to a specific Virtual Function. For older versions of `ethtool`, this depends on the "action" parameter. Specify the action as a 64-bit value, where the lower 32 bits represent the queue number, while the next 8 bits represent the VF ID. Note that 0 is the PF, so the VF identifier is offset by 1. For example:

```
# ethtool -U <ethX> flow-type tcp4 src-ip 192.168.10.1 dst-ip
192.168.10.2 src-port 2000 dst-port 2001 action 0x800000002 [loc 1]
```


The action field specifies to direct traffic to Virtual Function 7 (8 minus 1) into queue 2 of that VF.

Newer versions of `ethtool` (version 4.11 and later) use "vf" and "queue" parameters instead of the "action" parameter. Note that using the new `ethtool` "vf" parameter does not require the value to be offset by 1. This command is equivalent to the above example:

```
# ethtool -U <ethX> flow-type tcp4 src-ip 192.168.10.1 dst-ip
192.168.10.2 src-port 2000 dst-port 2001 vf 7 queue 2 [loc 1]
```

Note that these filters will not break internal routing rules, and will not route traffic that otherwise would not have been sent to the specified VF.

6.4.9.3 Flex Byte Intel Ethernet Flow Director Filters

 **NOTE:** This functionality is supported only on Intel Ethernet 500 Series and newer devices.

The driver also supports matching user-defined data within the packet payload. This flexible data is specified using the "user-def" field of the ethtool command in the following way:

31	28	24	20	16	15	12	8	4	0
offset into packet payload					2 bytes of flexible data				

For example,

```
... user-def 0x4FFFF ...
```

tells the filter to look 4 bytes into the payload and match that value against 0xFFFF. The offset is based on the beginning of the payload, and not the beginning of the packet. Thus

```
flow-type tcp4 ... user-def 0x8BEAF ...
```


would match TCP/IPv4 packets which have the value 0xBEAF 8 bytes into the TCP/IPv4 payload.

Note that ICMP headers are parsed as 4 bytes of header and 4 bytes of payload. Thus to match the first byte of the payload, you must actually add 4 bytes to the offset. Also note that ip4 filters match both ICMP frames as well as raw (unknown) ip4 frames, where the payload will be the L3 payload of the IP4 frame.

The maximum offset is 64. The hardware will only read up to 64 bytes of data from the payload. The offset must be even because the flexible data is 2 bytes long and must be aligned to byte 0 of the packet payload.

The user-defined flexible offset is also considered part of the input set and cannot be programmed separately for multiple filters of the same type. However, the flexible data is not part of the input set and multiple filters may use the same offset but match against different data.

6.4.10 Creating Traffic Classes

 **NOTE:** These instructions are not specific to ADQ configuration. Refer to the tc and tc-flower man pages for more information on creating traffic classes (TCs).

To create traffic classes on the interface:

1. Use the tc command to create traffic classes. The maximum number of traffic classes per interface are:

- **ice PF:** 16 TCs per interface
- **i40e PF:** 8 TCs per interface
- **iavf running on an ice device:** 16 TCs from the VF
- **iavf running on an i40e device:** 4 TCs from the VF

```
# tc qdisc add dev <ethX> root mqprio num_tc <tcs> map <priorities> queues
<count1@offset1 ...> hw 1 mode channel shaper bw_rlimit min_rate <min_rate1
...> max_rate <max_rate1 ...>
```

Where:

- `num_tc <tcs>`: The number of TCs to use.
- `map <priorities>`: The map of priorities to TCs. You can map up to 16 priorities to TCs.
- `queues <count1@offset1 ...>`: For each TC, `<num queues>@<offset>`. The max total number of queues for all TCs is the number of cores.
- `hw 1 mode channel`: 'channel' with 'hw' set to 1 is a new hardware offload mode in mqprio that makes full use of the mqprio options, the TCs, the queue configurations, and the QoS parameters.
- `shaper bw_rlimit`: For each TC, sets the minimum and maximum bandwidth rates. The totals must be equal to or less than the port speed. This parameter is optional and is required only to set up the Tx rates.
- `min_rate <min_rate1>`: Sets the minimum bandwidth rate limit for each TC.
- `max_rate <max_rate1 ...>`: Sets the maximum bandwidth rate limit for each TC. You can set a min and max rate together.

**NOTE:**

- If you set `max_rate` to less than 50Mbps, then `max_rate` is rounded up to 50Mbps and a warning is logged in `dmesg`.
- See the `mqprio` man page and the examples below for more information.

2. Verify the bandwidth limit using network monitoring tools such as `ifstat` or `sar -n DEV [interval] [number of samples]`



NOTE: Setting up channels via `ethtool` (`ethtool -L`) is not supported when the TCs are configured using `mqprio`.

3. Enable hardware TC offload on the interface:

```
# ethtool -K <ethX> hw-tc-offload on
```

4. Add `clsact` `qdisc` to enable adding ingress/egress filters for Rx/Tx:

```
# tc qdisc add dev <ethX> clsact
```

5. Verify successful TC creation after `qdisc` is created:

```
# tc qdisc show dev <ethX> ingress
```

Traffic Class Examples



NOTE: See the `tc` and `tc-flower` man pages for more information on traffic control and TC flower filters.

To set up two TCs (`tc0` and `tc1`), with 16 queues each, priorities 0-3 for `tc0` and 4-7 for `tc1`, and max Tx rate set to 1Gbit for `tc0` and 3Gbit for `tc1`:

```
# tc qdisc add dev ens4f0 root mqprio num_tc 2 map 0 0 0 0 1 1 1 1 queues
16@0 16@16 hw 1 mode channel shaper bw_rlimit max_rate 1Gbit 3Gbit
```

Where:

- `map 0 0 0 0 1 1 1 1`: Sets priorities 0-3 to use `tc0` and 4-7 to use `tc1`
- `queues 16@0 16@16`: Assigns 16 queues to `tc0` at offset 0 and 16 queues to `tc1` at offset 16

6.4.11 Creating Traffic Class Filters



NOTE: These instructions are not specific to ADQ configuration.

After creating traffic classes, use the `tc` command to create filters for traffic. Refer to the `tc` and `tc-flower` man pages for more information.

To view all TC filters:

```
# tc filter show dev <ethX> ingress
# tc filter show dev <ethX> egress
```

TC Filter Examples:

To configure TCP TC filters, where:

- `protocol`: Encapsulation protocol (valid options are IP and 802.1Q).
- `prio`: Priority.
- `flower`: Flow-based traffic control filter.
- `dst_ip`: IP address of the device.
- `ip_proto`: IP protocol to use (TCP or UDP).
- `dst_port`: Destination port.
- `src_port`: Source port.
- `skip_sw`: Flag to add the rule only in hardware.
- `hw_tc`: Route incoming traffic flow to this hardware TC. The TC count starts at 0. For example, 'hw_tc 1' indicates that the filter is on the second TC.
- `vlan_id`: VLAN ID.

TCP: Destination IP + L4 Destination Port

To route incoming TCP traffic with a matching destination IP address and destination port to the given TC:

```
# tc filter add dev <ethX> protocol ip ingress prio 1 flower dst_ip
<ip_address> ip_proto tcp dst_port <port_number> skip_sw hw_tc 1
```

TCP: Source IP + L4 Source Port

To route outgoing TCP traffic with a matching source IP address and source port to the given TC associated with the given priority:

```
# tc filter add dev <ethX> protocol ip egress prio 1 flower src_ip
<ip_address> ip_proto tcp src_port <port_number> action skbedit
priority 1
```

TCP: Destination IP + L4 Destination Port + VLAN Protocol

To route incoming TCP traffic with a matching destination IP address and destination port to the given TC using the VLAN protocol (802.1Q):

```
# tc filter add dev <ethX> protocol 802.1Q ingress prio 1 flower
dst_ip <ip address> eth_type ipv4 ip_proto tcp dst_port <port_
number> vlan_id <vlan_id> skip_sw hw_tc 1
```

NOTE: You can add multiple filters to the device, using the same recipe (and requires no additional recipe resources), either on the same interface or on different interfaces. Each filter uses the same fields for matching, but can have different match values.

```
# tc filter add dev <ethX> protocol ip ingress prio 1 flower
ip_proto tcp dst_port <port_number> skip_sw hw_tc 1

# tc filter add dev <ethX> protocol ip egress prio 1 flower
ip_proto tcp src_port <port_number> action skbedit priority 1
```

For example:

```
# tc filter add dev ens4f0 protocol ip ingress prio 1 flower ip_proto
tcp dst_port 5555 skip_sw hw_tc 1

# tc filter add dev ens4f0 protocol ip egress prio 1 flower ip_proto
tcp src_port 5555 action skbedit priority 1
```

6.4.12 IEEE 1588 Precision Time Protocol (PTP) Hardware Clock (PHC)

Precision Time Protocol (PTP) is used to synchronize clocks in a computer network. PTP support varies among Intel devices that support this driver. Use `ethtool -T <ethX>` to get a definitive list of PTP capabilities supported by the device.

6.5 Driver-Specific Information

This section details the command line parameters and configuration information that are specific to a driver.

For information common to multiple drivers, refer to the following:

- "Building and Installation" on page 73
- "Additional Configurations (All Drivers)" on page 79
- "Linux Known Issues" on page 197

6.5.1 ice Linux Driver for the Intel Ethernet 800 Series

The ice Linux driver supports the Intel Ethernet 800 Series family of adapters.

6.5.1.1 Important Notes - ice

SR-IOV, RDMA, and Link Aggregation

Note the following:

- The VF driver will not block teaming/bonding/link aggregation (LAG), but this is not a supported feature. Do not expect failover or load balancing on the VF interface.
- LAG and RDMA are compatible only in certain conditions. See the "RDMA (Remote Direct Memory Access)" section later in this driver's information for more information.

Bridging and MACVLAN are also affected by this. If you wish to use bridging or MACVLAN with RDMA/SR-IOV, you must set up bridging or MACVLAN before enabling RDMA or SR-IOV. If you are using bridging or MACVLAN in conjunction with SR-IOV and/or RDMA, and you want to remove the interface from the bridge or MACVLAN, you must follow these steps:


1. Remove RDMA if it is active
2. Destroy SR-IOV VFs if they exist
3. Remove the interface from the bridge or MACVLAN
4. Reactivate RDMA and recreate SR-IOV VFs as needed

6.5.1.2 Command Line Parameters - ice

The only command line parameter the ice driver supports is the debug parameter that can control the default logging verbosity of the driver. (Note: `dyndbg` also provides dynamic debug information.)

In general, use `ethtool` and other OS-specific commands to configure user-changeable parameters after the driver is loaded.


6.5.1.3 Additional Configurations - ice

 **NOTE:** Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

Dynamic Device Personalization

Dynamic Device Personalization (DDP) allows you to change the packet processing pipeline of a device by applying a profile package to the device at runtime. Profiles can be used to, for example, add support for new protocols, change existing protocols, or change default settings. DDP profiles can also be rolled back without rebooting the system.

The driver automatically installs the default DDP package file during driver installation.

 **NOTE:** It's important to do 'make install' during initial driver installation so that the driver loads the DDP package automatically.

The DDP package loads during device initialization. The driver looks for `intel/ice/ddp/ice.pkg` in your firmware root (typically `/lib/firmware/` or `/lib/firmware/updates/`) and checks that it contains a valid DDP package file.

If the driver is unable to load the DDP package, the device will enter Safe Mode. Safe Mode disables advanced and performance features and supports only basic traffic and minimal functionality, such as updating the NVM or downloading a new driver or DDP package. Safe Mode only applies to the affected physical function and does not impact any other PFs. For more details, see "Dynamic Device Personalization (DDP)" on page 22 and "Safe Mode" on page 188 in this user guide.

 **NOTE:**

- If you encounter issues with the DDP package file, you may need to download an updated driver or DDP package file. See the log messages for more information.
- The ice.pkg file is a symbolic link to the default DDP package file installed by the Linux-firmware software package or the ice out-of-tree driver installation.
- You cannot update the DDP package if any PF drivers are already loaded. To overwrite a package, unload all PFs and then reload the driver with the new package.
- Only the first loaded PF per device can download a package for that device.
- You can install specific DDP package files for different physical devices in the same system. See the Linux driver README in your installation for instructions on how to install a specific DDP package file.

RDMA (Remote Direct Memory Access)

Remote Direct Memory Access, or RDMA, allows a network device to transfer data directly to and from application memory on another system, increasing throughput and lowering latency in certain networking environments.

The ice driver supports the following RDMA protocols:

- iWARP (Internet Wide Area RDMA Protocol)
- RoCEv2 (RDMA over Converged Ethernet)

The major difference is that iWARP performs RDMA over TCP, while RoCEv2 uses UDP.

 **NOTES:**

- RDMA requires auxiliary bus support.
- Devices based on the Intel® Ethernet 800 Series do not support RDMA when operating in multiport mode with more than 4 ports.

For detailed installation and configuration information, see the README file in the RDMA driver tarball.


RDMA in the VF

Devices based on the Intel Ethernet 800 Series support RDMA in a Linux VF, on supported Windows or Linux hosts.

The iavf driver supports the following RDMA protocols in the VF:

- iWARP (Internet Wide Area RDMA Protocol)
- RoCEv2 (RDMA over Converged Ethernet)

Refer to the README inside the irdma driver tarball for details on configuring RDMA in the VF.

 **NOTE:** To support VF RDMA, load the irdma driver on the host before creating VFs. Otherwise VF RDMA support may not be negotiated between the VF and PF driver.

Auxiliary Bus

Inter-Driver Communication (IDC) is the mechanism in which LAN drivers (such as ice) communicate with peer drivers (such as irdma). Starting in kernel 5.11, Intel LAN and RDMA drivers use an auxiliary bus

mechanism for IDC.

RDMA functionality requires use of the auxiliary bus.

If your kernel supports the auxiliary bus, the LAN and RDMA drivers will use the inbox auxiliary bus for IDC. For kernels lower than 5.11, the base driver will automatically install an out-of-tree auxiliary bus module.

NVM Express* (NVMe*) over TCP and Fabrics

RDMA provides a high throughput, low latency means to directly access NVM Express* (NVMe*) drives on a remote server.

Refer to the following for details on supported operating systems and how to set up and configure your server and client systems:

- *NVM Express over TCP for Intel® Ethernet Products Configuration Guide*
- *NVM Express over Fabrics for Intel® Ethernet Products with RDMA Configuration Guide*

Both guides are available on the [Intel Technical Library](#).

Link Aggregation and RDMA

Link aggregation (LAG) and RDMA are compatible only if all the following are true:

- You are using an Intel Ethernet 810 Series device with the latest drivers and NVM installed.
- RDMA technology is set to RoCEv2.
- LAG configuration is active-backup.
- Bonding is between two ports within the same device.
- The QoS configuration of the two ports matches prior to the bonding of the devices.

If the above conditions are not met:

- The PF driver will not enable RDMA.
- RDMA peers will not be able to register with the PF.



NOTE: The first interface added to an aggregate (bond) is assigned as the "primary" interface for RDMA and LAG functionality. If LAN interfaces are assigned to the bond and you remove the primary interface from the bond, RDMA will not function properly over the bonded interface. To address the issue, remove all interfaces from the bond and add them again. Interfaces that are not assigned to the bond will operate normally.

Application Device Queues (ADQ)

Application Device Queues (ADQ) allow you to dedicate one or more queues to a specific application. This can reduce latency for the specified application, and allow Tx traffic to be rate limited per application.

For requirements and configuration information for Intel Ethernet 800 Series devices, refer to the [Intel® Ethernet Controller E810 Application Device Queues \(ADQ\) Configuration Guide](#).

Creating Traffic Classes



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

Additional examples for the ice driver:

To create 8 TCs with 256 queues spread across all the TCs, when ADQ is enabled:

```
# tc qdisc add dev <ethX> root mqprio num_tc 8 map 0 1 2 3 4 5 6 7
   queues 2@0 4@2 8@6 16@14 32@30 64@62 128@126 2@254 hw 1 mode channel
```

To set a minimum rate for a TC:

```
# tc qdisc add dev ens4f0 root mqprio num_tc 2 map 0 0 0 0 1 1 1 1
   queues 4@0 8@4 hw 1 mode channel shaper bw_rlimit min_rate 25Gbit
   50Gbit
```

To set a maximum data rate for a TC:

```
# tc qdisc add dev ens4f0 root mqprio num_tc 2 map 0 0 0 0 1 1 1 1
   queues 4@0 8@4 hw 1 mode channel shaper bw_rlimit max_rate 25Gbit
   50Gbit
```

To set both minimum and maximum data rates together:

```
# tc qdisc add dev ens4f0 root mqprio num_tc 2 map 0 0 0 0 1 1 1 1
   queues 4@0 8@4 hw 1 mode channel shaper bw_rlimit min_rate 10Gbit
   20Gbit max_rate 25Gbit 50Gbit
```

Creating Traffic Class Filters



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

For detailed configuration information and example code for switchdev mode on Intel Ethernet 800 Series devices, refer to the configuration guide at <https://cdrdv2.intel.com/v1/dl/getContent/645272>.

To add a GTP filter:

```
# tc filter add dev <ethX> protocol ip parent ffff: prio 1 flower
   src_ip 16.0.0.0/16 ip_proto udp dst_port 5678 enc_dst_port 2152 enc_
   key_id <tunnel_id> skip_sw hw_tc 1
```

Where:

- `dst_port`: inner destination port of application (5678)
- `enc_dst_port`: outer destination port (for GTP user data tunneling occurs on UDP port 2152)
- `enc_key_id`: tunnel ID (vxlan ID)

Using TC Filters to Forward to a Queue

The driver supports directing traffic based on L2/L3/L4 fields in the packet to specific Rx queues, using the TC filter's class ID. Note: This functionality can be used with or without ADQ.

To add filters for the desired queue, use the following tc command:

```
# tc filter add dev <ethX> ingress prio 1 protocol all flower src_
   mac <mac_address> skip_sw classid ffff:<queue_id>
```

Where:

- <mac_address> is the MAC address(es) you want to direct to the Rx queue
- <queue_id> is the Rx queue ID number in hexadecimal

For example, to direct a single MAC address to queue 10:

```
# ethtool -K ens801 hw-tc-offload on
# tc qdisc add dev ens801 clsact
# tc filter add dev ens801 ingress prio 1 protocol all flower src_mac
68:dd:ac:dc:19:00 skip_sw classid ffff:b
```

To direct 4 source MAC addresses to Rx queues 10-13:

```
# ethtool -K ens801 hw-tc-offload on
# tc qdisc add dev ens801 clsact
# tc filter add dev ens801 ingress prio 1 protocol all flower src_mac
68:dd:ac:dc:19:00 skip_sw classid ffff:b
# tc filter add dev ens801 ingress prio 1 protocol all flower src_mac
68:dd:ac:dc:19:01 skip_sw classid ffff:c
# tc filter add dev ens801 ingress prio 1 protocol all flower src_mac
68:dd:ac:dc:19:02 skip_sw classid ffff:d
# tc filter add dev ens801 ingress prio 1 protocol all flower src_mac
68:dd:ac:dc:19:03 skip_sw classid ffff:e
```

Intel® Ethernet Flow Director

The ice driver supports the following flow types:

- IPv4
- TCPv4
- UDPv4
- IPv6
- TCPv6
- UDPv6
- SCTPv6

Each flow type supports valid combinations of IP addresses (source or destination) and UDP/TCP/SCTP ports (source and destination). You can supply only a source IP address, a source IP address and a destination port, or any combination of one or more of these four parameters.

This driver allows you to filter traffic based on a user-defined flexible two-byte pattern and offset by using the ethtool user-def and mask fields. Only L3 and L4 flow types are supported for user-defined flexible filters. For a given flow type, you must clear all Intel Ethernet Flow Director filters before changing the input set (for that flow type).



NOTE: Intel Ethernet Flow Director filters impact only LAN traffic. RDMA filtering occurs before Flow Director, so Intel Ethernet Flow Director filters will not impact RDMA.



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for more information on configuring Intel Ethernet Flow Director and its filters.

RSS Hash Flow

Allows you to set the hash bytes per flow type and any combination of one or more options for Receive Side Scaling (RSS) hash byte configuration.

```
# ethtool -N <ethX> rx-flow-hash <type> <option>
```

Where <type> is:

tcp4 signifying TCP over IPv4

udp4 signifying UDP over IPv4

tcp6 signifying TCP over IPv6

udp6 signifying UDP over IPv6

And <option> is one or more of:

s Hash on the IP source address of the Rx packet.

d Hash on the IP destination address of the Rx packet.

f Hash on bytes 0 and 1 of the Layer 4 header of the Rx packet.

n Hash on bytes 2 and 3 of the Layer 4 header of the Rx packet.

For example:

- To hash on the source and destination IP address for TCP IPv4 traffic, use the following:
ethtool -N <ethX> rx-flow-hash tcp4 sd
- To hash on the source and destination ports for UDP IPv6 traffic, use the following:
ethtool -N <ethX> rx-flow-hash udp6 sdfn

Accelerated Receive Flow Steering (aRFS)

Devices based on the Intel® Ethernet 800 Series support Accelerated Receive Flow Steering (aRFS) on the PF. aRFS is a load-balancing mechanism that allows you to direct packets to the same CPU where an application is running or consuming the packets in that flow.

NOTES:

- aRFS requires that ntuple filtering is enabled via ethtool.
- aRFS support is limited to the following packet types:
 - TCP over IPv4 and IPv6
 - UDP over IPv4 and IPv6
 - Nonfragmented packets
- aRFS only supports Intel Ethernet Flow Director filters, which consist of the source/destination IP addresses and source/destination ports.
- aRFS and ethtool's ntuple interface both use the device's Intel Ethernet Flow Director. aRFS and ntuple features can coexist, but you may encounter unexpected results if there's a conflict between aRFS and ntuple requests.

To set up aRFS:

1. Enable the Intel Ethernet Flow Director and ntuple filters using ethtool.

```
# ethtool -K <ethX> ntuple on
```

2. Set up the number of entries in the global flow table. For example:

```
# NUM_RPS_ENTRIES=16384
# echo $NUM_RPS_ENTRIES > /proc/sys/net/core/rps_sock_flow_
entries
```

3. Set up the number of entries in the per-queue flow table. For example:

```
# NUM_RX_QUEUES=64
# for file in /sys/class/net/$IFACE/queues/rx-*/rps_flow_cnt;
do
# echo $((($NUM_RPS_ENTRIES/$NUM_RX_QUEUES)) > $file;
# done
```

4. Disable the IRQ balance daemon (this is only a temporary stop of the service until the next reboot).

```
# systemctl stop irqbalance
```

5. Configure the interrupt affinity.

```
# set_irq_affinity <ethX>
```

To disable aRFS using ethtool:

```
# ethtool -K <ethX> ntuple off
```



NOTE: This command will disable ntuple filters and clear any aRFS filters in software and hardware.

Example Use Case:

1. Set the server application on the desired CPU (e.g., CPU 4).

```
# taskset -c 4 netserver
```

2. Use netperf to route traffic from the client to CPU 4 on the server with aRFS configured. This example uses TCP over IPv4.

```
# netperf -H <Host IPv4 Address> -t TCP_STREAM
```

Enabling Virtual Functions (VFs) for SR-IOV

Use sysfs to enable virtual functions (VF).

For example, you can create 4 VFs as follows:


```
# echo 4 > /sys/class/net/<ethX>/device/sriov_numvfs
```

To disable VFs, write 0 to the same file:

```
# echo 0 > /sys/class/net/<ethX>/device/sriov_numvfs
```


The maximum number of VFs for the ice driver is 256 total (all ports). To check how many VFs each PF supports, use the following command:

```
# cat /sys/class/net/<ethX>/device/sriov_totalvfs
```

 **NOTE:** The VF driver will not block teaming/bonding/link aggregation (LAG), but this is not a supported feature. Do not expect failover or load balancing on the VF interface.

SR-IOV Live Migration


You can use VFIO Device Migration to move an active virtual machine (VM) between different physical machines so it does not lose its network connection. After migrating, the virtual function (VF) will continue most Ethernet operations without further interruption. During migration, data and VIRTCHNL operations are sent to a buffer so they can be recreated when the migration completes. If the memory allocated for the command buffer is exceeded, the system will drop the buffer and disable the live migration capability for the VF. You must reset the VF for live migration to be re-enabled.

 **NOTE:**

- Live migration requires kernel version 5.15 to 5.17
- You cannot migrate a VM if it has a VF that is using RDMA.
- You can only migrate the VF to a device in the same family with a similar firmware version. For example, you can migrate a VF from one 810 device to another, but not from an 810 device to an 820 device.
- Any VF properties that are set by the PF will not be migrated. Make sure that both devices have the same PF-set properties.

Intel® Scalable I/O Virtualization Support

Refer to "Intel® Scalable I/O Virtualization Support" on page 62 for an overview of this feature.

 **NOTE:**

- This feature is not available in the kernel driver. Download and install the current OOT driver to use this feature.
- The VF driver will not block teaming/bonding/link aggregation (LAG), but this is not a supported feature. Do not expect failover or load balancing on the VF interface.

Enabling Intel® Scalable I/O Virtualization Virtual Devices

Use the following steps to enable Intel® Scalable I/O Virtualization (Intel® Scalable IOV) virtual devices (VDEVs):

1. Create an mdev

```
# echo "83b8f4f2-509f-382f-3c1e-e6bfe0fa1001" | sudo tee
/sys/class/mdev_bus/0000:38:00.0/mdev_supported_types/ice-
ivdm/create
```

2. Use qemu to launch a VM with four processors


```
# sudo ./x86_64-softmmu/qemu-system-x86_64 \
    -enable-kvm \
    -m 1G \
    -smp 4 \
    -device
    vfio-pci,sysfsdev=/sys/bus/mdev/devices/83b8f4f2-
    509f-382f-3c1e-e6bfe0fa1001 \
    -drive file=../../img/test.qcow2 \
    -nic user,hostfwd=tcp::5555-:22 \
    -monitor stdio
```

Refer to "Intel® Scalable I/O Virtualization Support" on page 62 for additional information on this feature.

Displaying VF Statistics on the PF

Use the following command to display the statistics for the PF and its VFs:

```
# ip -s link show dev <ethX>
```



NOTE: The output of this command is very large due to the maximum number of possible VFs.

The PF driver will display a subset of the VF's statistics, as provided by the VF driver, for all VFs that are configured. The PF will always print a statistics block for each of the possible VFs, and it will show a zero for all unconfigured VFs.

Enabling a VF Link if the Port Is Disconnected

If the physical function (PF) link is down, you can force link up (from the host PF) on any virtual functions (VF) bound to the PF. Note that this requires kernel support and associated iproute2 user space support. If the following command does not work, it may not be supported by your system. The following command forces link up on VF 0 bound to PF eth0:

```
# ip link set eth0 vf 0 state enable
```

Setting the MAC Address for a VF

To change the MAC address for the specified VF:

```
# ip link set <ethX> vf 0 mac <address>
```

For example:

```
# ip link set <ethX> vf 0 mac 00:01:02:03:04:05
```

This setting lasts until the PF is reloaded.



NOTE: For untrusted VFs, assigning a MAC address for a VF from the host will disable any subsequent requests to change the MAC address from within the VM. This is a security feature. The VM is not aware of this restriction, so if this is attempted in the VM, it will trigger MDD events. Trusted VFs are allowed to change the MAC address from within the VM.

Trusted VFs and VF Promiscuous Mode

This feature allows you to designate a particular VF as trusted and allows that trusted VF to request selective promiscuous mode on the Physical Function (PF).

To set a VF as trusted or untrusted, enter the following command in the Hypervisor:

```
# ip link set dev <ethX> vf 1 trust [on|off]
```



NOTE: It's important to set the VF to trusted before setting promiscuous mode. If the VM is not trusted, the PF will ignore promiscuous mode requests from the VF. If the VM becomes trusted after the VF driver is loaded, you must make a new request to set the VF to promiscuous.

Once the VF is designated as trusted, use the following commands in the VM to set the VF to promiscuous mode.

- For promiscuous all: # ip link set <ethX> promisc on
Where <ethX> is a VF interface in the VM
- For promiscuous Multicast: # ip link set <ethX> allmulticast on
Where <ethX> is a VF interface in the VM



NOTE: By default, the ethtool private flag `vf-true-promisc-support` is set to "off," meaning that promiscuous mode for the VF will be limited. To set the promiscuous mode for the VF to true promiscuous and allow the VF to see all ingress traffic, use the following command:

```
# ethtool --set-priv-flags <ethX> vf-true-promisc-support on
```

The `vf-true-promisc-support` private flag does not enable promiscuous mode; rather, it designates which type of promiscuous mode (limited or true) you will get when you enable promiscuous mode using the 'ip link' commands above. You can toggle the `vf-true-promisc-support` flag separately for all PFs.

Next, add a VLAN interface on the VF interface. For example:

```
# ip link add link eth2 name eth2.100 type vlan id 100
```

Note that the order in which you set the VF to promiscuous mode and add the VLAN interface does not matter (you can do either first). The result in this example is that the VF will get all traffic that is tagged with VLAN 100.

Virtual Function (VF) Tx Rate Limit

Use the `ip` command to configure the maximum or minimum Tx rate limit for a VF from the PF interface.

For example, to set a maximum Tx rate limit of 8000Mbps for VF 0:

```
# ip link set eth0 vf 0 max_tx_rate 8000
```

For example, to set a minimum Tx rate limit of 1000Mbps for VF 0:

```
# ip link set eth0 vf 0 min_tx_rate 1000
```

 **NOTE:**

- If DCB or ADQ are enabled on a PF, you cannot set a minimum Tx rate on the VFs associated with that PF.
- If both DCB and ADQ are disabled on a PF, then you can set a minimum Tx rate on the VFs associated with that PF.
- If you set a minimum Tx rate limit on a PF for SR-IOV VFs and then apply a DCB or ADQ configuration, the PF cannot guarantee the minimum Tx rate limits for those VFs.
- If you set a minimum Tx rate on VFs across multiple ports that have an aggregate bandwidth over 100Gbps, the PFs cannot guarantee the minimum Tx rate set for the VFs.

Malicious Driver Detection (MDD) for VFs

Some Intel Ethernet devices use Malicious Driver Detection (MDD) to detect malicious traffic from the VF and disable Tx/Rx queues or drop the offending packet until a VF driver reset occurs. You can view MDD messages in the PF's system log using the `dmesg` command.

- If the PF driver logs MDD events from the VF, confirm that the correct VF driver is installed.
- To restore functionality, you can manually reload the VF or VM or enable automatic VF resets.
- When automatic VF resets are enabled, the PF driver will immediately reset the VF and reenables queues when it detects MDD events on the receive path.
- If automatic VF resets are disabled, the PF will not automatically reset the VF when it detects MDD events.

To enable or disable automatic VF resets, use the following command:

```
# ethtool --set-priv-flags <ethX> mdd-auto-reset-vf on|off
```

MAC and VLAN Anti-Spoofing Feature for VFs

When a malicious driver on a Virtual Function (VF) interface attempts to send a spoofed packet, it is dropped by the hardware and not transmitted.

This feature can be disabled for a specific VF:

```
# ip link set <pf dev> vf <vf id> spoofchk {off|on}
```

VLAN Pruning

The ice driver allows you to enable or disable VLAN pruning for the VF VSI using the `ethtool` private flag `vf-vlan-pruning`.

 **NOTE:**

- You cannot change this private flag while any VFs are active.
- If a port VLAN is configured, VLAN pruning will always be enabled.
- When VLAN pruning is enabled, the interface will:
 - Discard all packets with a VLAN tag when Rx VLAN filtering is disabled.
 - Discard untagged packets when Rx VLAN filtering is enabled.

To disable or enable VLAN pruning on all VFs, do the following:

1. Deinitialize any VFs.
2. On the PF, use the following command:

```
# ethtool --set-priv-flags <ethX> vf-vlan-pruning on|off
```

Where:

- on: enables VLAN pruning
- off: disables VLAN pruning (default)

3. Initialize and configure any VFs.

VLAN pruning will then be disabled or enabled on any of these VFs, depending on the flag you set.

Switchdev mode

The PF driver supports legacy and switchdev eSwitch modes. Switchdev mode allows the driver to create additional port representor netdevs that enable a control plane running on the host to configure filters for the VFs and also handle default/exception traffic from the uplink and the VFs.

The driver loads in legacy mode by default. You can configure eSwitch modes independently per physical port using the devlink command. You can change between eSwitch modes only if no VFs have been created. If SR-IOV is enabled and VFs are bound to the PF, you must do the following before changing between switchdev and legacy mode:

- Unload all VFs that were bound
- Set the number of VFs on the PF to zero



NOTE:

- ADQ, trusted VFs, L2 forwarding, and S-IOV are not supported in switchdev mode.
- Switchdev mode is not persistent across reboots or driver reloads.

To configure the device in switchdev mode, enter the following, where `<pci/0000:##:##.##>` is the PCI address of the PF:

```
# devlink dev eswitch set <pci/0000:##:##.##> mode switchdev
```

For example:

```
# devlink dev eswitch set pci/0000:17:00.0 mode switchdev
```

To configure the device in legacy mode:

```
# devlink dev eswitch set <pci/0000:##:##.##> mode legacy
```

To check the current eSwitch mode:

```
# devlink dev eswitch show <pci/0000:##:##.##>
```

The ice driver supports the following hardware offloads in switchdev mode:

- Supported filter conditions:
 - **L2:** Source/Destination MAC addresses, VLAN ID
 - **L3:** Source/Destination IP addresses (IPv4, IPv6), IP protocol (TCP, UDP), ToS (IPv4), Traffic Class (IPv6), TTL (IPv4)
 - **L4:** Source and Destination port

- **VXLAN/GRETAP/GENEVE:** VNI/GRE Key, Outer Destination IP, Inner Source IP, Inner Destination IP, Inner Destination MAC, TCP/UDP Source port and Destination port
- **GTP:** TEID, PDU type, QFI, Outer Destination IP, Outer Source IP
- Supported filter actions: redirect, drop



NOTE: GTP support requires kernel 5.18 and iproute2 5.18 or newer. On older kernel versions, the DCF method provides the same functionality.

For detailed configuration information and example code for switchdev mode on Intel Ethernet 800 Series devices, refer to the configuration guide at <https://cdrdv2.intel.com/v1/dl/getContent/645272>.

At a high level, do the following to offload TC filters to the hardware and create switch rules in switchdev mode:

1. Verify that switchdev mode is enabled.
2. Enable hw-tc-offload on the VF port representor (VF_PR).
3. For tunnel interfaces: Use the ip link command to create the tunnel.
4. Use the tc-flower command to create the switch rule.
5. Verify the offloaded flow in hardware.

Switchdev mode supports the following ip link commands to configure the VF:

- mac
- vlan, vxlan, geneve, gre, nvgre, gtp, qos, proto
- max_tx_rate
- min_tx_rate
- spoofchk
- query_rss
- state
- node_guid
- port_guid



NOTE: trust is not supported. rate is supported but deprecated; use max_tx_rate instead.

To limit the VF's interrupt rate for Rx and Tx in switchdev mode, use the following command, where <vf_pr> is the designated VF port representor and <N> is the desired cap for the interrupt rate:

```
# ethtool -C <vf_pr> rx-usecs-high <N>
```

Jumbo Frames



NOTES:

- The maximum MTU setting for Jumbo Frames is 9702 bytes. This value coincides with the maximum Jumbo Frames size of 9728 bytes.
- This driver will attempt to use multiple page sized buffers to receive each jumbo packet. This should help to avoid buffer starvation issues when allocating receive packets.
- Packet loss may have a greater impact on throughput when you use jumbo frames. If you observe a drop in performance after enabling jumbo frames, enabling flow control may mitigate the issue.

Speed and Duplex Configuration

You cannot set speed, duplex, or autonegotiation settings using `ethtool`.

To see the speed configurations your device supports, run the following:

```
# ethtool <ethX>
```

To have your device advertise supported speeds, use the following:

```
# ethtool -s <ethX> advertise N  
Where N is a bitmask of the desired speeds.
```

For example, to have your device advertise 10000baseSR Full, use:

```
# ethtool -s <ethX> advertise 0x800000000000
```

For more details, please refer to the `ethtool` man page.

Data Center Bridging (DCB)

Note: The kernel assumes that TC0 is available, and will disable Priority Flow Control (PFC) on the device if TC0 is not available. To fix this, ensure TC0 is enabled when setting up DCB on your switch.

DCB is a configuration Quality of Service implementation in hardware. It uses the VLAN priority tag (802.1p) to filter traffic. That means that there are 8 different priorities that traffic can be filtered into. It also enables priority flow control (802.1Qbb) which can limit or eliminate the number of dropped packets during network stress. Bandwidth can be allocated to each of these priorities, which is enforced at the hardware level (802.1Qaz).

DCB is normally configured on the network using the DCBX protocol (802.1Qaz), a specialization of LLDP (802.1AB). The driver supports the following mutually exclusive variants of DCBX support::

- Firmware-based LLDP Agent
- Software-based LLDP Agent

In firmware-based mode, firmware intercepts all LLDP traffic and handles DCBX negotiation transparently for the user. In this mode, the adapter operates in "willing" DCBX mode, receiving DCB settings from the link partner (typically a switch). The local user can only query the negotiated DCB configuration. For information on configuring DCBX parameters on a switch, please consult the switch manufacturer's documentation.

In software-based mode, LLDP traffic is forwarded to the network stack and user space, where a software agent can handle it. In this mode, the adapter can operate in either "willing" or "nonwilling" DCBX mode and DCB configuration can be both queried and set locally. This mode requires the FW-based LLDP Agent to be disabled.

 **NOTE:**

- You can enable and disable the firmware-based LLDP Agent using an ethtool private flag. Refer to the "FW-LLDP (Firmware Link Layer Discovery Protocol)" section in this README for more information.
- In software-based DCBX mode, you can configure DCB parameters using software LLDP/DCBX agents that interface with the Linux kernel's DCB Netlink API. We recommend using OpenLLDP as the DCBX agent when running in software mode. For more information, see the OpenLLDP man pages and <https://github.com/intel/openlldp>.
- The driver implements the DCB netlink interface layer to allow the user space to communicate with the driver and query DCB configuration for the port.
- iSCSI with DCB is not supported.

L3 QoS mode

The ice driver supports setting DSCP-based Layer 3 Quality of Service (L3 QoS) in the PF driver. The driver initializes in L2 QoS mode. L3 QoS mode is:

- Automatically enabled when the first DSCP/ToS to TC mapping is defined
- Automatically disabled when the last DSCP/ToS to TC mapping is removed

The following is an example of how to map a DSCP/ToS to a TC:

```
# lldptool -T -i <ethX> -V APP app=<prio>,<sel>,<pid>
```

where:

<prio>: The TC assigned to the DSCP/ToS code point

<sel>: 5 for DSCP to TC mapping

<pid>: The DSCP/ToS code point

For example, to map packets containing DSCP value 63 to traffic class 0 on interface eth0:

```
# lldptool -T -i eth0 -V APP app=63,5,0
```

To remove a mapping, use the following:

```
# lldptool -T -I <ethX> -V APP -d app=<prio>,<sel>,<pid>
```

To view the currently configured mappings, use the following:

```
# lldptool -t -i <ethX> -V APP -c
```

 **NOTE:**

- L3 QoS mode is not available when FW-LLDP is enabled. You also cannot enable FW-LLDP if L3 QoS mode is active. Disable FW-LLDP before switching to L3 QoS mode. Refer to the "FW-LLDP (Firmware Link Layer Discovery Protocol)" section in this README for more information on disabling FW-LLDP.
- Once a mapping has been submitted for a DSCP value, another mapping for that value will not be accepted until the first one has been deleted.

FW-LLDP (Firmware Link Layer Discovery Protocol)

Use ethtool to change FW-LLDP settings. The FW-LLDP setting is per port and persists across boots.

To enable LLDP:

```
# ethtool --set-priv-flags <ethX> fw-lldp-agent on
```

To disable LLDP:

```
# ethtool --set-priv-flags <ethX> fw-lldp-agent off
```

To check the current LLDP setting:

```
# ethtool --show-priv-flags <ethX>
```



NOTE: You must enable the UEFI HII "LLDP Agent" attribute for this setting to take effect. If "LLDP AGENT" is set to disabled, you cannot enable it from the OS.

Forward Error Correction (FEC)

Allows you to set the Forward Error Correction (FEC) mode. FEC improves link stability, but increases latency. Many high quality optics, direct attach cables, and backplane channels provide a stable link without FEC.



NOTE: For devices to benefit from this feature, link partners must have FEC enabled.

If you enable the flag `allow-no-fec-modules-in-auto`, Auto FEC negotiation will include 'No FEC' in case your link partner does not have FEC enabled or is not FEC capable.

```
# ethtool --set-priv-flags <ethX> allow-no-fec-modules-in-auto on
```

On kernels older than 4.14, use the following private flags to disable FEC modes:

- `rs-fec` (0 to disable, 1 to enable)
- `base-r-fec` (0 to disable, 1 to enable)

On kernel 4.14 or later, use `ethtool` to get/set the following FEC modes:

- No FEC
- Auto FEC
- BASE-R FEC
- RS FEC

Link-Level Flow Control (LFC)

You may encounter issues with link-level flow control (LFC) after disabling DCB. The LFC status may show as enabled but traffic is not paused. To resolve this issue, disable and reenables LFC using `ethtool`:

```
# ethtool -A <ethX> rx off tx off
```

```
# ethtool -A <ethX> rx on tx on
```

Limiting the Maximum Bitrate for a Transmit Queue

The ice driver supports limiting the transmit queue bit rate with the `tx_maxrate` sysfs entry. Use this entry to set a maximum bitrate in Mbps. A value of zero means no limiting.

Setting the bit rate for transmit queue 1 to 300 Mbps:


```
# echo 300 > /sys/class/<ethx>/queues/tx-1/tx_maxrate
```

Removing the limit:

```
# echo 0 > /sys/class/<ethx>/queues/tx-1/tx_maxrate
```

MACVLAN

This driver supports MACVLAN. Kernel support for MACVLAN can be tested by checking if the MACVLAN driver is loaded. You can run 'lsmod | grep macvlan' to see if the MACVLAN driver is loaded or run 'modprobe macvlan' to try to load the MACVLAN driver.



NOTE:

- In passthru mode, you can only set up one MACVLAN device. It will inherit the MAC address of the underlying PF (Physical Function) device.

ice devices support L2 Forwarding Offload. This will offload the processing required for L2 Forwarding from the system processors to the ice device.

Perform the following steps to enable L2 Forwarding Offload:

1. Enable L2 Forwarding offload:

```
# ethtool -K <ethX> l2-fwd-offload on
```
2. Create the MACVLAN netdevs and bind them to the PF.
3. Bring up/enable the MACVLAN netdevs.



NOTE: MACVLAN offloads and ADQ are mutually exclusive. System instability may occur if you enable l2-fwd-offload and then set up ADQ, or if you set up ADQ and then enable l2-fwd-offload.

IEEE 802.1ad (QinQ) Support

The IEEE 802.1ad standard, informally known as QinQ, allows for multiple VLAN IDs within a single Ethernet frame. VLAN IDs are sometimes referred to as "tags," and multiple VLAN IDs are thus referred to as a "tag stack." Tag stacks allow L2 tunneling and the ability to separate traffic within a particular VLAN ID, among other uses.

The following are examples of how to configure 802.1ad (QinQ):

```
# ip link add link eth0 eth0.24 type vlan proto 802.1ad id 24  
# ip link add link eth0.24 eth0.24.371 type vlan proto 802.1Q id 371
```

Where "24" and "371" are example VLAN IDs.

 **NOTES:**

- 802.1ad (QinQ) is supported in 3.19 and later kernels.
- 802.1ad (QinQ) and RDMA are not compatible.
- VLAN protocols use the following EtherTypes:
 - 802.1Q = EtherType 0x8100
 - 802.1ad = EtherType 0x88A8
- For QinQ traffic to work at MTU 1500, the L2 peer (switch port or another NIC) should be able to receive Ethernet frames of 1526 bytes. Some third-party NICs support a maximum Ethernet frame size of 1522 bytes at MTU 1500, which will cause QinQ traffic to fail. To work around this issue, restrict the MTU on the Intel Ethernet device to 1496.

Double VLANs

Devices based on the Intel Ethernet 800 Series can process up to two VLANs in a packet when all the following are installed:

- ice driver version 1.4.0 or later
- NVM version 2.4 or later
- ice DDP package version 1.3.21 or later

If you don't use the versions above, the only supported VLAN configuration is single 802.1Q VLAN traffic.

When two VLAN tags are present in a packet, the outer VLAN tag can be either 802.1Q or 802.1ad. The inner VLAN tag must always be 802.1Q.

 **NOTE THE FOLLOWING LIMITATIONS:**

- For each VF, the PF can only allow VLAN hardware offloads (insertion and stripping) of one type, either 802.1Q or 802.1ad.
- You can't enable or disable outer or single 802.1Q or 802.1ad filtering separately. They are either both on or both off.
- In SR-IOV mode, the VF may not receive all network traffic based on the inner VLAN header when VF true promiscuous mode (vf-true-promisc-support) and double VLANs are enabled.

To enable outer or single 802.1Q VLAN insertion and stripping and disable 802.1ad VLAN insertion and stripping:

```
# ethtool -K <ethX> rxvlan on txvlan on rx-vlan-stag-hw-parse off
tx-vlan-stag-hw-insert off
```

To enable outer or single 802.1ad VLAN insertion and stripping and disable 802.1Q VLAN insertion and stripping:

```
# ethtool -K <ethX> rxvlan off txvlan off rx-vlan-stag-hw-parse on
tx-vlan-stag-hw-insert on
```

To enable outer or single VLAN filtering:

```
# ethtool -K <ethX> rx-vlan-filter on rx-vlan-stag-filter on
```

To disable outer or single VLAN filtering:

```
# ethtool -K <ethX> rx-vlan-filter off rx-vlan-stag-filter off
```

Combining QinQ with SR-IOV VFs

We recommend you always configure a port VLAN for the VF from the PF. If a port VLAN is not configured, the VF driver may only offload VLANs via software. The PF allows all VLAN traffic to reach the VF, and the VF manages all VLAN traffic.

When the device is configured for double VLANs and the PF has configured a port VLAN:

- The VF can only offload guest VLANs for 802.1Q traffic.
- The VF can only configure VLAN filtering rules for guest VLANs using 802.1Q traffic.

However, when the device is configured for double VLANs and the PF has NOT configured a port VLAN:

- You must use iavf driver version 4.1.0 or later to offload and filter VLANs.
- The PF turns on VLAN pruning and antispoof in the VF's VSI by default. The VF will not transmit or receive any tagged traffic until the VF requests a VLAN filter.
- The VF can offload (insert and strip) the outer VLAN tag of 802.1Q or 802.1ad traffic.
- The VF can create filter rules for the outer VLAN tag of both 802.1Q and 802.1ad traffic.

If the PF does not support double VLANs, the VF can hardware offload single 802.1Q VLANs without a port VLAN.

When the PF is enabled for double VLANs, for iavf drivers before version 4.1.x:

- VLAN hardware offloads and filtering are supported only when the PF has configured a port VLAN.
- VLAN filtering, insertion, and stripping will be software offloaded when no port VLAN is configured.

To see VLAN filtering and offload capabilities, use the following command:

```
# ethtool -k <ethX> | grep vlan
```

IEEE 1588 Precision Time Protocol (PTP) Hardware Clock (PHC)



NOTE: You can only change the state of the Clock Generation Unit (CGU) Fast Lock flag (cgu-fast-lock) on PF0.

The following devices support this advanced functionality:

- Intel® Ethernet 100G 2P E810-C-st Adapter
- Intel® Ethernet 100G 2P E810-C-stg Adapter
- Intel® Ethernet 25G 4P E810-XXV-st Adapter
- Intel® Ethernet 25G 4P E810-XXV-stg Adapter

See "User Guides for Specific Devices" on page 6 for links to detailed configuration user guides for this feature.

Some devices support hardware-generated timestamps. The driver uses these timestamps to synchronize clocks on the platform and report precise timestamps on packets. Use the following `hwstamp_ctl` command, which is available in the `linuxptp` utility, to enable this setting:

```
# hwstamp_ctl -i <ethX> -t 1 -r 1
```

GNSS Support

Some Intel Ethernet devices have integrated Global Navigation Satellite System (GNSS) functionality, to aid in high-precision timing synchronization across the network.

The following devices support this advanced functionality:

- Intel® Ethernet 100G 2P E810-C-stg Adapter
- Intel® Ethernet 25G 4P E810-XXV-stg Adapter

See "User Guides for Specific Devices" on page 6 for links to detailed configuration user guides for this feature.

SyncE Support

On hardware that supports Synchronous Ethernet (SyncE), the ice driver has interfaces that allow you to synchronize frequencies with other SyncE-supported ports. After you manually configure SyncE, the device dynamically selects the best quality signal from the ones that are available. Then, once the signal is locked, it synchronizes its frequency clock to it. The best quality signal is determined based on the topology configured with the ice SyncE interfaces.



NOTE: You can only change the state of the Clock Generation Unit (CGU) Fast Lock flag (cgu-fast-lock) on PF0.

The following devices support this advanced functionality:

- Intel® Ethernet 100G 2P E810-C-st Adapter
- Intel® Ethernet 100G 2P E810-C-stg Adapter
- Intel® Ethernet 25G 4P E810-XXV-st Adapter
- Intel® Ethernet 25G 4P E810-XXV-stg Adapter

See "User Guides for Specific Devices" on page 6 for links to detailed configuration user guides for this feature.

Tunnel/Overlay Stateless Offloads

Supported tunnels and overlays include VXLAN, GENEVE, and others depending on hardware and software configuration. Stateless offloads are enabled by default.

To view the current state of all offloads:

```
# ethtool -k <ethX>
```

UDP Segmentation Offload

Allows the adapter to offload transmit segmentation of UDP packets with payloads up to 64K into valid Ethernet frames. Because the adapter hardware is able to complete data segmentation much faster than operating system software, this feature may improve transmission performance.

In addition, the adapter may use fewer CPU resources.



NOTES:

- UDP transmit segmentation offload requires Linux kernel 4.18 or later.
- The application sending UDP packets must support UDP segmentation offload.

To enable/disable UDP Segmentation Offload, issue the following command:

```
# ethtool -K <ethX> tx-udp-segmentation [off|on]
```

Runtime Control of CRC/FCS Stripping

The frame check sequence (FCS) is a four-octet cyclic redundancy check (CRC) that allows the driver to detect corrupted data within a received Ethernet frame.

The ice driver allows you to disable or enable FCS/CRC stripping using the ethtool command.



NOTES:

- FCS/CRC stripping is enabled by default.
- The driver enforces valid combinations of FCS/CRC and VLAN stripping. You can only disable FCS/CRC stripping if VLAN stripping is also disabled on the PF.
- Disabling FCS/CRC stripping may help when debugging issues. XDP programs can also use FCS/CRC for their purposes.

Use the following ethtool command to enable or disable FCS/CRC stripping:

```
# ethtool -K <ethX> rx-fcs on|off
```

To check the status of FCS/CRC stripping, look for the 'rx-fcs' information reported from ethtool:

```
# ethtool -k <ethX>
```

Port Split Configuration Using Devlink

Most Intel Ethernet 800 Series devices support changing their port split configuration to suit your needs. For example, a dual port device may support two 100Gbps links, two 50 Gbps links, and (with the correct cables) four 25 Gbps links, etc. The supported port split configurations are defined in the device's NVM. You can use a tool like Intel's Ethernet Port Configuration Tool (EPCT) to query and set this configuration. If no such tool is available, you can use devlink to cycle through a device's possible port split configurations. If you use devlink to change the configuration, you must check the log to determine which configuration was selected. If you use devlink, you specify the number of ports you want configured on the device. Each time you call devlink with that port count, the driver will check the device's current configuration and then move to the next configuration with the specified number of ports. For example, if your device has two four-port configurations defined in its NVM, the first time you called devlink, it would select the first configuration. The second time you called devlink, it would select the second configuration. If you called devlink again, it would select the first configuration. There is no direct feedback mechanism; you must check the log to determine which configuration was set. Use the following command:

```
# devlink port split <pci/D:b:d.f>/0 count <num>
```

Where:

- <pci/D:b:d.f>/0 is the PCI address of the device (pci/Domain:bus:device.function).
/0 is the PORT_INDEX.
- <num> is the desired port split count.

 **NOTE:**

- If you successfully change a port's configuration, the driver logs an information message: "Reboot required to finish port split" and the port split configuration selected. This is the only indication of success.
- If you request an unsupported count value parameter in devlink port split, the driver logs an information message: "Port split requested unsupported port config."
- If you try to change the configuration on a PF that is not PF 0, the driver returns the error "Port cannot be split."

For example, if your device had the following configurations defined in its NVM:

```
ice 0000:16:00.0:  Status  Split      Quad 0      Quad 1
ice 0000:16:00.0:      count  L0  L1  L2  L3  L4  L5  L6  L7
ice 0000:16:00.0: Active  2   100 -  -  -  100 -  -  -
ice 0000:16:00.0:      2    50 -  50 -  -  -  -  -
ice 0000:16:00.0:      4    25 25 25 25 -  -  -
ice 0000:16:00.0:      4    25 25 -  - 25 25 -  -
ice 0000:16:00.0:      8    10 10 10 10 10 10 10 10
ice 0000:16:00.0:      1   100 -  -  -  -  -  -  -
```

If you call:

```
# devlink port split pci/0000.16:00.0/0 count 4
```

Your device will be configured for:

```
ice 0000:16:00.0:      4    25 25 25 25 -  -  -  -
```

If you call the same command again, your device will be configured for:


```
ice 0000:16:00.0:      4    25 25 -  - 25 25 -  -
```

If you call the same command a third time, your device will cycle back to the top of its 4-port configurations (because there are only two 4-port configurations defined in its NVM) and will be set to:

```
ice 0000:16:00.0:      4    25 25 25 25 -  -  -  -
```

Firmware Logs

The ice driver allows you to generate firmware logs for supported categories of events, to help debug issues with Customer Support. Firmware logs are enabled by default.

 **NOTE:** Refer to "Firmware Logging" on page 178 for an overview of this feature and additional tips.

Firmware logs are printed to dmesg. The driver groups these events into categories, called "modules." Supported modules include:

- 00000001 - General (Bit 0)
- 00000002 - Control (Bit 1)
- 00000004 - Link Management (Bit 2)
- 00000008 - Link Topology Detection (Bit 3)
- 00000010 - Link Control Technology (Bit 4)
- 00000020 - I2C (Bit 5)
- 00000040 - SDP (Bit 6)
- 00000080 - MDIO (Bit 7)
- 00000100 - Admin Queue (Bit 8)
- 00000200 - Host DMA (Bit 9)
- 00000400 - LLDP (Bit 10)
- 00000800 - DCBx (Bit 11)
- 00001000 - DCB (Bit 12)
- 00002000 - XLR (function-level resets; Bit 13)
- 00004000 - NVM (Bit 14)
- 00008000 - Authentication (Bit 15)
- 00010000 - VPD (Vital Product Data; Bit 16)
- 00020000 - IOSF (Intel On-Chip System Fabric, Bit 17)
- 00040000 - Parser (Bit 18)
- 00080000 - Switch (Bit 19)
- 00100000 - Scheduler (Bit 20)
- 00200000 - TX Queue Management (Bit 21)
- 00400000 - ACL (Access Control List; Bit 22)
- 00800000 - Post (Bit 23)
- 01000000 - Watchdog (Bit 24)
- 02000000 - Task Dispatcher (Bit 25)
- 04000000 - Manageability (Bit 26)
- 08000000 - SyncE (Bit 27)
- 10000000 - Health (Bit 28)
- 20000000 - Time Sync (Bit 29)
- 40000000 - PF Registration (Bit 30)
- 80000000 - Module Version (Bit 31)

You can change the verbosity level of the firmware logs. You can set only one log level per module, and each level includes the verbosity levels lower than it. For instance, setting the level to "normal" will also log warning and error messages. Available verbosity levels are:

- 0 = none
- 1 = error
- 2 = warning
- 3 = normal
- 4 = verbose

 **NOTE:**

- See below for all commands to configure firmware logging.
- Firmware logs can overrun the dmesg buffer. Before loading the driver, redirect dmesg to a file.
- Use a bitmap to set the desired verbosity level for the module(s). You must have dynamic debug enabled in the kernel.
- You cannot change firmware log parameters at runtime. You must reload the driver for changes to take effect.

At a high level, do the following to capture a firmware log in Linux:

1. Remove the driver:

```
# rmmod ice
```


2. Redirect the firmware log from dmesg to a file:

```
# dmesg -w > filename.log
```

3. Load the driver using the following command, changing the events and level values as needed:

```
# sudo insmod ice.ko dyndbg="+p" fwlog_events=<bitmask> fwlog_
level=<level 0-4>
```

4. Perform the necessary steps to generate the issue you're trying to debug.
5. Work with Customer Support to decode your firmware log file and debug the issue.

 **NOTE:** To disable firmware logging completely, remove the driver and reload it. Firmware logging will remain disabled until you enable it again.

Code Examples:

To set all events to log warning messages, use the following command:

```
# sudo insmod ice.ko dyndbg="+p" fwlog_events=0x0FFFFFFF fwlog_
level=2
```

To log verbose, normal, warning, and error messages for the ACL (Bit 22), Switch (Bit 19), and Parser (Bit 18) modules, for example, use the following:

```
# sudo insmod ice.ko dyndbg="+p" fwlog_events=0x4C0000 fwlog_level=4
```

To dump the firmware logging configuration to dmesg, use the following commands:

```
# echo dump fwlog > command
# dmesg
```

Hierarchical QoS (HQoS) Transmit Scheduler

You can configure a custom transmit scheduler tree structure to shape transmit traffic for specific needs. You change the tree structure by creating parent nodes on the device and then assigning child nodes (VFs) to the parent node. You can also change the transmit rate management configuration for each node.

 **NOTES:**

- Reconfiguring the scheduler topology should only be done by an expert. Modifying the scheduler topology may adversely impact your device's network availability and throughput. Do not do this unless you are willing to take these risks. After modifying the scheduler topology, if your device does not perform as expected, you should return the device to the default topology.
- Modifying the Hierarchical QoS (HQoS) Transmit Scheduler requires Kernel 6.2, or later.
- Modifying the Hierarchical QoS (HQoS) Transmit Scheduler is not compatible with ADQ, DCB, RDMA, or other custom scheduler tree features.

To create a devlink-rate parent group:

```
# devlink port function rate add <dev/port>/<group>
```

where:

- <dev/port> is the pci bus:device:function of the device
- <group> is a new parent group

For example:

```
# devlink port function rate add pci/0000:03:00.0/operators
```

creates the "operators" group on the specified device

To create a new child node in a parent group:

```
# devlink port function rate add <dev/port>/<child> parent <group>
```

where:

- <dev/port> is the pci bus:device:function of the device
- <child> is a new child node
- <group> is an existing parent group

For example:

```
# devlink port function rate add pci/0000:03:00.0/class_1 parent operators
```

creates the "class_1" child node in the "operators" parent group.

To display a device's current tree structure:

```
# devlink port function rate show <dev/port>
```

where <dev/port> is the pci bus:device:function of the device

For example:

```
# devlink port function rate show pci/0000:03:00.0
```

Example output:

```
pci/0000:03:00.0/node_0 type node (root)
pci/0000:03:00.0/operators type node tx_share 20Mbit tx_max 100Mbit tx_
priority 2 tx_weight 5
pci/0000:03:00.0/class_1 type node parent operators
```

```
pci/0000:03:00.0/1 type leaf parent class_1
```

Refer to the devlink-rate MAN page and other documentation for details.

MSI-X Vector Allocation

The ice driver automatically allocates MSI-X vectors for PF, VF, and RDMA from a pool of 2048 vectors. If there are 8, or fewer, local node CPU threads, the driver will automatically allocate 8 vectors for each PF. This scales up by allocating one vector per local node CPU thread, up to 64 vectors. The driver will not automatically allocate more than 64 MSI-X vectors for each PF. RDMA requires one more MSI-X vector than the PF allocation, so the driver will automatically allocation 9-65 MSI-X vectors for RDMA.

Setting MSI-X Vector Allocation

You can use sysfs to override the automatic MSI-X vector allocation for a particular PF or RDMA function, or for the pool of vectors used by the VFs bound to a PF.

```
# devlink resource set <pci/D:b:d.f> msix/<parameter> size <num>
```

Where:

- <pci/D:b:d.f> is the PCI address of the device (pci/Domain:bus:device.function)
- <parameter> is one of the following:
 - For a PF, use the `msix_eth` parameter
 - For an RDMA function, use the `msix_rdma` parameter
 - For the pool of vectors used by the VFs use the `msix_vf` parameter
- <num> is the number of MSI-X vector to assign to the function

For example, to set a PF to use 320 MSI-X vectors:

```
# devlink resource set pci/0000:31:00.1 msix/msix_eth size 320
```



NOTE: For this change to take affect you must reinitialize the driver after you make this change. Reinitializing the driver may drop some netdev configurations, including reset or downtime. Refer to the Devlink Reload documentation for more information.

You can set the allocation for a particular VF with the `sriov_vf_msix_count` sysfs parameter.

```
# echo <num> > /sys/bus/pci/devices/D:b:d.f/sriov_vf_msix_count
```

Where:

- <D:b:d.f> is the PCI address of the device (Domain:bus:device.function)
- <num> is the number of MSI-X vector allocate to the particular VF

For example, to set a VF to 64 MSI-X vectors, use:

```
# echo 64 > /sys/bus/pci/devices/0000:31:00.2/sriov_vf_msix_count
```

Current MSI-X Allocation

You can check the current MSI-X vector allocation by using the `devlink resource show` parameter. For example:

```
# devlink resource show pci/0000:31:00.1
```

Might return:

```
name: msix size 520 occ 262 unit entry dpipe_tables none
resources:
  name msix_misc size 4 unit entry dpipe_tables none
  name: msix_eth size 48 occ 24 unit
  name: msix_vf size 48 occ 24 unit
  name: msix_rdma size 48 occ 24 unit
```

Increasing the automatic allocation limit

The ice driver supports changing the automatic MSI-X vector allocation for PFs and VFs to spread the RSS load across more cores. Each PF has its own LUT, while all VFs use the global LUT. Each PF LUT allows for 2048 MSI-X vectors. The VF default is a limit of 64 MSI-X vectors, but you can increase this to 512 vectors if there are enough resources in the global LUT. You can also assign a PF's LUT to a bound VF, increasing the VF's MSI-X vector limit to 2048, but decreasing the PF's limit to 512. Use the `rss_lut_pf_attr` and `rss_lut_vf_attr` sysfs parameters to manage this.



NOTES:

- Before changing `rss_lut_vf_attr`, you must first set `sriov_drivers_autoprobe` to zero. After changing `rss_lut_vf_attr`, you can set `sriov_drivers_autoprobe` back to 1.
- You must reload the iavf driver after making these changes.

Set a VF's limit to 512, using the global LUT:

```
# echo 0 > /sys/bus/pci/devices/<ethx>/sriov_drivers_autoprobe
# echo 512 > /sys/bus/pci/devices/<ethx>/rss_lut_vf_attr
```

Set a VF to use its PF's LUT:

```
# echo 0 > /sys/bus/pci/devices/<ethx>/sriov_drivers_autoprobe
# echo 512 > /sys/bus/pci/devices/<ethx>/rss_lut_pf_attr
# echo 2048 > /sys/bus/pci/devices/<ethx>/rss_lut_vf_attr
```

Set a PF back to using its PF LUT.

```
# echo 0 > /sys/bus/pci/devices/<ethx>/sriov_drivers_autoprobe
# echo 512 > /sys/bus/pci/devices/<ethx>/rss_lut_vf_attr
# echo 2048 > /sys/bus/pci/devices/<ethx>/rss_lut_pf_attr
```

6.5.1.4 Performance Optimization - ice

The driver defaults are meant to fit a wide variety of workloads. If further optimization is required, we recommend experimenting with the following settings.

Transmit/Receive Queue Allocation

The driver allocates a number of transmit/receive queue pairs equal to the number of local node CPU threads with the following constraints:

- The driver will allocate a minimum of 8 queue pairs, or the total number of CPUs, whichever is lower
- The driver will allocate a maximum of 64 queue pairs. Or 256 for the iavf driver.

You can set the number of queues symmetrical or asymmetrical using the `ethtool -L` command. For example:

- Setting 16 queue pairs for the interface:


```
# ethtool -L <ethX> combined 16
# ethtool -L <ethX> tx 16 rx 16
```
- Setting 16 Tx queues and 8 Rx queues:


```
# ethtool -L <ethX> tx 16 rx 8
```



NOTE:

- You cannot configure less than 1 Rx or 1 Tx queue. Attempts to do so will be rejected by the driver.
- You cannot configure more Tx/Rx queues than there are MSI-X interrupts available. Attempts to do so will be rejected by the driver.

IRQ to Adapter Queue Alignment

Pin the adapter's IRQs to specific cores by disabling the irqbalance service and using the included `set_irq_affinity` script. Please see the script's help text for further options.

The following settings will distribute the IRQs across all the cores evenly:

```
# scripts/set_irq_affinity -X all <interface1> , [ <interface2>, ...
]
```

The following settings will distribute the IRQs across all the cores that are local to the adapter (same NUMA node):

```
# scripts/set_irq_affinity -X local <interface1> , [ <interface2>,
... ]
```

For very CPU-intensive workloads, we recommend pinning the IRQs to all cores.

Rx Descriptor Ring Size

To reduce the number of Rx packet discards, increase the number of Rx descriptors for each Rx ring using `ethtool`.

Check if the interface is dropping Rx packets due to buffers being full (`rx_dropped.nic` means there is no PCIe bandwidth):

```
# ethtool -S <interface> | grep "rx_dropped"
```

If the previous command shows drops on queues, it may help to increase the number of descriptors using `ethtool -G`:

```
# ethtool -G rx <N>
```

Where `<N>` is the desired number of ring entries/descriptors.

This can provide temporary buffering for issues that create latency while the CPUs process descriptors.

Interrupt Rate Limiting

This driver supports an adaptive interrupt throttle rate (ITR) mechanism that is tuned for general workloads. The user can customize the interrupt rate control for specific workloads, via ethtool, adjusting the number of microseconds between interrupts.

To set the interrupt rate manually, you must disable adaptive mode:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off
```

For lower CPU utilization:

- Disable adaptive ITR and lower Rx and Tx interrupts. The examples below affect every queue of the specified interface.
- Setting rx-usecs and tx-usecs to 80 will limit interrupts to about 12,500 interrupts per second per queue:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs 80 tx-usecs 80
```

For reduced latency:

- Disable adaptive ITR and ITR by setting rx-usecs and tx-usecs to 0 using ethtool:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs 0 tx-usecs 0
```

Per-queue interrupt rate settings:

- The following examples are for queues 1 and 3, but you can adjust other queues.
- To disable Rx adaptive ITR and set static Rx ITR to 10 microseconds or about 100,000 interrupts/second, for queues 1 and 3:

```
# ethtool --per-queue <ethX> queue_mask 0xa --coalesce adaptive-rx off rx-usecs 10
```

- To show the current coalesce settings for queues 1 and 3:

```
# ethtool --per-queue <ethX> queue_mask 0xa --show-coalesce
```

Bounding interrupt rates using rx-usecs-high:

- Valid Range: 0-236 (0=no limit)
The range of 0-236 microseconds provides an effective range of 4,237 to 250,000 interrupts per second. The value of rx-usecs-high can be set independently of rx-usecs and tx-usecs in the same ethtool command, and is also independent of the adaptive interrupt moderation algorithm. The underlying hardware supports granularity in 4-microsecond intervals, so adjacent values may result in the same interrupt rate.
- The following command would disable adaptive interrupt moderation, and allow a maximum of 5 microseconds before indicating a receive or transmit was complete. However, instead of resulting in as many as 200,000 interrupts per second, it limits total interrupts per second to 50,000 via the rx-usecs-high parameter.

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs-high 20 rx-usecs 5 tx-usecs 5
```

Virtualized Environments

In addition to the other suggestions in this section, the following may be helpful to optimize performance in VMs.

- Disable XPS on both ends by using the included virt_perf_default script or by running the following command as root:

```
for file in `ls /sys/class/net/ethX/queues/tx-*/xps_cpus`;  
do echo 0 > $file; done
```

- Using the appropriate mechanism (vcpupin) in the VM, pin the CPUs to individual LCPUs, making sure to use a set of CPUs included in the device's local_cpulist:
/sys/class/net/<ethX>/device/local_cpulist.
- Configure as many Rx/Tx queues in the VM as available. (See the iavf driver documentation for the number of queues supported.) For example:

```
# ethtool -L <virt_interface> rx <max> tx <max>
```

Transmit Balancing

Refer to "Transmit Balancing" on page 39 for an overview of this feature.

To set the transmit balancing feature in Linux via devlink:

```
# devlink dev param set <pci/D:b:d.f> name txbalancing value  
<setting> cmode permanent
```

Where:

- <pci/D:b:d.f> is the PCI address of the PF.
- <setting> is true to enable transmit balancing, or false to disable transmit balancing.

To show the current transmit balancing setting:

```
# devlink dev param show [ <pci> name txbalancing ]
```

6.5.2 i40e Linux Driver for the Intel Ethernet 700 Series

The i40e Linux base driver supports the Intel Ethernet 700 Series of adapters.

6.5.2.1 Important Notes - i40e

TC0 must be enabled when setting up DCB on a switch

The kernel assumes that TC0 is available, and will disable Priority Flow Control (PFC) on the device if TC0 is not available. To fix this, ensure TC0 is enabled when setting up DCB on your switch.

Enabling a VF Link if the Port Is Disconnected

If the physical function (PF) link is down, you can force link up (from the host PF) on any virtual functions (VF) bound to the PF. Note that this requires kernel support and associated iproute2 user space support. If the following command does not work, it may not be supported by your system. The following command forces link up on VF 0 bound to PF eth0:

```
# ip link set eth0 vf 0 state enable
```

6.5.2.2 Command Line Parameters - i40e

In general, ethtool and other OS specific commands are used to configure user changeable parameters after the driver is loaded. The i40e driver only supports the max_vfs kernel parameter on older kernels that do not have the standard sysfs interface. The only other module parameter is the debug parameter that can control the default logging verbosity of the driver.

If the driver is built as a module, the following optional parameters are used by entering them on the command line with the modprobe command using this syntax:


```
# modprobe i40e [<option>=<VAL1>]
```

For example:

```
# modprobe i40e max_vfs=7
```

The default value for each parameter is generally the recommended setting, unless otherwise noted.

The following table contains parameters and possible values for modprobe commands:

Parameter Name	Valid Range/Settings	Default	Description
max_vfs	1-32 (Intel Ethernet Controller X710 based devices) 1-64 (Intel Ethernet Controller XXV710/XL710 based devices)	0	<p>This parameter adds support for SR-IOV. It causes the driver to spawn up to max_vfs worth of virtual functions.</p> <p>NOTE: This parameter is only used on kernel 3.7.x and below. On kernel 3.8.x and above, use sysfs to enable VFs. Use sysfs for Red Hat distributions.</p> <p>For example, you can create 4 VFs as follows:</p> <pre># echo 4 > /sys/class/net/<ethX>/device/sriov_numvfs</pre> <p>To disable VFs, write 0 to the same file:</p> <pre># echo 0 > /sys/class/net/<ethX>/device/sriov_numvfs</pre> <p>The parameters for the driver are referenced by position. Thus, if you have a dual port adapter, or more than one adapter in your system, and want N virtual functions per port, you must specify a number for each port with each parameter separated by a comma. For example:</p> <pre># modprobe i40e max_vfs=4</pre> <p>This will spawn 4 VFs on the first port.</p> <pre># modprobe i40e max_vfs=2,4</pre> <p>This will spawn 2 VFs on the first port and 4 VFs on the second port.</p> <p> NOTES:</p> <ul style="list-style-type: none"> • Caution must be used in loading the driver with these parameters. Depending on your system configuration, number of slots, etc., it is impossible to predict in all cases where the positions would be on the command line. • Neither the device nor the driver control how VFs are mapped into config space. Bus layout will vary by operating system. On operating systems that support it, you can check sysfs to find the mapping. <p>Some hardware configurations support fewer SR-IOV instances, as the whole Intel Ethernet Controller XL710 (all functions) is limited to 128 SR-IOV interfaces in total.</p> <p>When SR-IOV mode is enabled, hardware VLAN filtering and VLAN tag stripping/insertion will remain enabled. Please remove the old VLAN filter before the new VLAN filter is added. For example:</p> <pre># ip link set eth0 vf 0 vlan 100 // set vlan 100 for VF 0</pre> <pre># ip link set eth0 vf 0 vlan 0 // Delete vlan 100</pre> <pre># ip link set eth0 vf 0 vlan 200 // set a new vlan 200 for VF 0</pre>

6.5.2.3 Additional Configurations - i40e



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

Displaying VF Statistics on the PF

Use the following command to display the statistics for all VFs on the PF:


```
# ethtool -S <ethX>
```



NOTE: The output of this command is very large due to the large number of VF statistics and the maximum number of possible VFs.

The PF driver will display a subset of the VF's statistics, as provided by the VF driver, for all VFs that are configured. The PF will always print a statistics block for each of the possible VFs, and it will show a zero for all unconfigured VFs.

VF stats are listed in a single block at the end of the PF statistics, using the following naming convention:

```
vf<XXX>.<statistic name>
```

Where:

- <XXX> is the VF number (for example, vf008).
- <statistic name> is the name of the statistic as supplied by the VF driver.

For example:

```
vf008.rx_bytes: 0
vf008.rx_unicast: 0
vf008.rx_multicast: 0
vf008.rx_broadcast: 0
vf008.rx_discards: 0
vf008.rx_unknown_protocol: 0
vf008.tx_bytes: 0
vf008.tx_unicast: 0
vf008.tx_multicast: 0
vf008.tx_broadcast: 0
vf008.tx_discards: 0
vf008.tx_errors: 0
```

Setting the MAC Address for a VF

To change the MAC address for the specified VF:

```
# ip link set <ethX> vf 0 mac <address>
```

For example:

```
# ip link set <ethX> vf 0 mac 00:01:02:03:04:05
```

This setting lasts until the PF is reloaded.



NOTE: For untrusted VFs, assigning a MAC address for a VF from the host will disable any subsequent requests to change the MAC address from within the VM. This is a security feature. The VM is not aware of this restriction, so if this is attempted in the VM, it will trigger MDD events. Trusted VFs are allowed to change the MAC address from within the VM.

Trusted VFs and VF Promiscuous Mode

This feature allows you to designate a particular VF as trusted and allows that trusted VF to request selective promiscuous mode on the Physical Function (PF).

To set a VF as trusted or untrusted, enter the following command in the Hypervisor:

```
# ip link set dev <ethX> vf 1 trust [on|off]
```



NOTE: It's important to set the VF to trusted before setting promiscuous mode. If the VM is not trusted, the PF will ignore promiscuous mode requests from the VF. If the VM becomes trusted after the VF driver is loaded, you must make a new request to set the VF to promiscuous.

Once the VF is designated as trusted, use the following commands in the VM to set the VF to promiscuous mode.

- For promiscuous all: # ip link set <ethX> promisc on
Where <ethX> is a VF interface in the VM
- For promiscuous Multicast: # ip link set <ethX> allmulticast on
Where <ethX> is a VF interface in the VM



NOTE: By default, the ethtool private flag `vf-true-promisc-support` is set to "off," meaning that promiscuous mode for the VF will be limited. To set the promiscuous mode for the VF to true promiscuous and allow the VF to see all ingress traffic, use the following command:

```
# ethtool --set-priv-flags <ethX> vf-true-promisc-support on
```

The `vf-true-promisc-support` private flag does not enable promiscuous mode; rather, it designates which type of promiscuous mode (limited or true) you will get when you enable promiscuous mode using the 'ip link' commands above. Note that this is a global setting that affects the entire device. However, the `vf-true-promisc-support` private flag is only exposed to the first PF of the device. The PF remains in limited promiscuous mode (unless it is in MFP mode) regardless of the `vf-true-promisc-support` setting.

Next, add a VLAN interface on the VF interface. For example:

```
# ip link add link eth2 name eth2.100 type vlan id 100
```

Note that the order in which you set the VF to promiscuous mode and add the VLAN interface does not matter (you can do either first). The result in this example is that the VF will get all traffic that is tagged with VLAN 100.

Virtual Function (VF) Tx Rate Limit

Use the ip command to configure the Tx rate limit for a VF from the PF interface.

For example, to set a Tx rate limit of 1000Mbps for VF 0:

```
# ip link set eth0 vf 0 rate 1000
```

Malicious Driver Detection (MDD) for VFs

Some Intel Ethernet devices use Malicious Driver Detection (MDD) to detect malicious traffic from the VF and disable Tx/Rx queues or drop the offending packet until a VF driver reset occurs. You can view MDD messages in the PF's system log using the `dmesg` command.

- If the PF driver logs MDD events from the VF, confirm that the correct VF driver is installed.
- To restore functionality, you can manually reload the VF or VM or enable automatic VF resets.
- When automatic VF resets are enabled, the PF driver will immediately reset the VF and reenable queues when it detects MDD events on the receive path.
- If automatic VF resets are disabled, the PF will not automatically reset the VF when it detects MDD events.

To enable or disable automatic VF resets, use the following command:

```
# ethtool --set-priv-flags <ethX> mdd-auto-reset-vf on|off
```

MAC and VLAN Anti-Spoofing Feature for VFs

When a malicious driver on a Virtual Function (VF) interface attempts to send a spoofed packet, it is dropped by the hardware and not transmitted.

This feature can be disabled for a specific VF:

```
# ip link set <pf dev> vf <vf id> spoofchk {off|on}
```

VLAN Pruning

The i40e driver allows you to enable or disable VLAN pruning for the VF VSI using the ethtool private flag `vf-vlan-pruning`.



NOTE:

- You cannot change this private flag while any VFs are active.
- If a port VLAN is configured, VLAN pruning will always be enabled.
- When VLAN pruning is enabled, the interface will:
 - Discard all packets with a VLAN tag when Rx VLAN filtering is disabled.
 - Discard untagged packets when Rx VLAN filtering is enabled.

To disable or enable VLAN pruning on all VFs, do the following:

1. Deinitialize any VFs.
2. On the PF, use the following command:

```
# ethtool --set-priv-flags <ethX> vf-vlan-pruning on|off
```

Where:

`on`: enables VLAN pruning

`off`: disables VLAN pruning (default)

3. Initialize and configure any VFs.

VLAN pruning will then be disabled or enabled on any of these VFs, depending on the flag you set.

Intel® Ethernet Flow Director

The i40e driver supports the following flow types:

- IPv4
- TCPv4
- UDPv4
- IPv6
- TCPv6
- UDPv6
- SCTPv6

Each flow type supports valid combinations of IP addresses (source or destination) and UDP/TCP ports (source and destination). You can supply only a source IP address, a source IP address and a destination port, or any combination of one or more of these four parameters.

This driver allows you to filter traffic based on a user-defined flexible two-byte pattern and offset by using the `ethtool` `user-def` and `mask` fields. Only L3 and L4 flow types are supported for user-defined flexible filters. For a given flow type, you must clear all Intel Ethernet Flow Director filters before changing the input set (for that flow type).



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for more information on configuring Intel Ethernet Flow Director and its filters.

Application Targeted Routing (ATR) Perfect Filters



NOTE: This functionality is supported only on Intel Ethernet 700 Series devices.

Intel Ethernet Flow Director ATR is enabled by default when the kernel is in multiple transmit queue mode. A rule is added when a TCP flow starts and is deleted when the flow ends. Because this would interfere with sideband TCP rules, the driver automatically disables ATR when a TCP rule is added via `ethtool` (sideband). ATR is automatically re-enabled when all TCP sideband rules are deleted or when sideband is disabled.

You can disable or enable ATR using the `ethtool` private flags interface. To view the current setting:

```
# ethtool --show-priv-flags <ethX>
```

To change the setting:

```
# ethtool --set-priv-flags <ethX> flow-director-atr [off|on]
```

Packets that match the ATR rules will increment the `port.fdir_atr_match` stat in `ethtool`. The current operational state of ATR is reflected by the stat `port.fdir_atr_status`.

Cloud Filter Support



NOTE: This functionality is supported only on Intel Ethernet 700 Series devices.

On a complex network that supports multiple types of traffic (such as for storage as well as cloud), cloud filter support allows you to send one type of traffic (for example, the storage traffic) to the Physical Function (PF) and another type (say, the cloud traffic) to a Virtual Function (VF). Because cloud networks are typically VXLAN/GENEVE-based, you can define a cloud filter to identify VXLAN/GENEVE packets and send them to a queue in the VF to be processed by the virtual machine (VM). Similarly, other cloud filters can be designed for various other traffic tunneling.

 **NOTE:**

- Cloud filters are only supported when the underlying device is in Single Function per Port mode.
- The "action -1" option, which drops matching packets in regular Intel Ethernet Flow Director filters, is not available to drop packets when used with cloud filters.
- For IPv4 and ether flow-types, cloud filters cannot be used for TCP or UDP filters.
- Cloud filters can be used as a method for implementing queue splitting in the PF.
- Queue 0xffff, set through either `queue 0xffff` or `action 0x100ffff`, is used for RSS.

The following filters are supported:

- Cloud Filters
 - Inner MAC, Inner VLAN (for NVGRE, VXLAN or GENEVE packets)
 - Inner MAC, Inner VLAN, Tenant ID (for NVGRE, VXLAN or GENEVE packets)
 - Inner MAC, Tenant ID (NVGRE packet or VXLAN/GENEVE packets)
 - Outer MAC L2 filter
 - Inner MAC filter
 - Outer MAC, Tenant ID, Inner MAC
 - Application Destination IP
 - Application Source-IP, Inner MAC (see NOTES below)
 - Destination Port: TCP, UDP, or both, depending on module parameter
 - ToQueue: Use MAC, VLAN to point to a queue
- L3 filters
 - Application Destination IP

 **NOTE:**

- Cloud filters are not compatible with ADQ.
- The Destination Port cloud filter is a load time option; use the 'l4mode' module parameter to enable it. The l4mode module parameter supports the following settings:
 - parameter not present = destination port filters disabled
 - 0 = UDP cloud filter mode enabled; destination port applies to UDP
 - 1 = TCP cloud filter mode enabled; destination port applies to TCP
 - 2 = both TCP and UDP filters are enabled; destination port applies to both UDP and TCP protocols
 - **Note:** When the l4mode parameter specifies TCP (1) or both (2), the driver will disable Application Targeted Routing (ATR). The driver will reenables ATR when the last Destination Port cloud filter rule is removed.
- The Application Source-IP, Inner MAC filter is not available when the Destination Port cloud filter is selected.
- To change back to default mode (which supports the Application Source-IP, Inner MAC filter listed above), you must reboot the system.

Cloud filters are specified using ethtool's ntuple interface, but the driver uses user-def to determine whether to treat the filter as a cloud filter or a regular filter. To enable a cloud filter, set the highest bit of the user-def field, "user-def 0x8000000000000000" to enable the cloud features described below. This specifies to the driver to treat the filter specially and not treat it like the regular filters described above. Note that cloud filters also read the other bits in the user-def field separately so you cannot use the flexible data feature described above.

For regular Intel Ethernet Flow Director filters:

- No user-def specified or highest bit (bit 63) is 0. For example:

```
# ethtool -U enp130s0 flow-type ip4 src-ip 192.168.0.1 dst-ip 192.168.0.109 action
6 loc <N>
```

For L3 filters (non-tunneled packets):

- "user-def 0x8000000000000000" (no Tenant ID/VNI specified in remaining bits of the user-def field)
- Only L3 parameters (src-IP, dst-IP) are considered
- For example, to redirect traffic coming from 192.168.42.13 with destination 192.168.42.33 into VF id 1, and call this "rule 3":

```
# ethtool -U enp130s0 flow-type ip4 src-ip 192.168.42.13 dst-ip 192.168.42.33 /
src-port 12344 dst-port 12344 user-def 0x8000000000000000 action /
0x200000000 loc 3
```

For cloud filters (tunneled packets):

- All other filters, including where Tenant ID/VNI is specified. The lower 32 bits of the user-def field can carry the tenant ID/VNI if required.
- The 'loc' parameter specifies the rule number of the filter as being stored in the base driver.
- The VF can be specified using the "action" field, just as regular filters described in the Flow Director Filter section above.
- To forward tunneled GRE packets directly to the VF, set bit 24 of the "user-def" field.
- Cloud filters can be defined with inner MAC, outer MAC, inner IP address, inner VLAN, and VNI as part of the cloud tuple. Cloud filters filter on destination (not source) MAC and IP. The destination and source MAC address fields in the ethtool command are overloaded as dst = outer, src = inner MAC address to facilitate tuple definition for a cloud filter.
- Examples:
 - To redirect traffic on VXLAN using tunnel id 34 (hex 0x22) coming from outer MAC address 8b:9d:ed:6a:ce:43 and inner MAC address 1d:44:9d:54:da:de into VF id 1 and call this "rule 38":


```
# ethtool -U enp130s0 flow-type ether dst 8b:9d:ed:6a:ce:43 \
src 1d:44:9d:54:da:de user-def 0x8000000000000022 loc 38 \
action 0x200000000
```
 - To forward tunneled GRE packets to VF 0:


```
# ethtool -U enp130s0 flow-type ether user-def 0x8000000001000000 action
0x10000ffff
```
 - To redirect traffic to L4 destination port 4789 to VF 0, when the l4mode module parameter is set to UDP:


```
# ethtool -U enp130s0 flow-type udp4 dst-port 4789 action 0xffffffff00000000
```

RSS Hash Flow

Allows you to set the hash bytes per flow type and any combination of one or more options for Receive Side Scaling (RSS) hash byte configuration.

```
# ethtool -N <ethX> rx-flow-hash <type> <option>
```

Where <type> is:

tcp4 signifying TCP over IPv4

udp4 signifying UDP over IPv4

tcp6 signifying TCP over IPv6

udp6 signifying UDP over IPv6

And <option> is one or more of:

s Hash on the IP source address of the Rx packet.

d Hash on the IP destination address of the Rx packet.

f Hash on bytes 0 and 1 of the Layer 4 header of the Rx packet.

n Hash on bytes 2 and 3 of the Layer 4 header of the Rx packet.

For example:

- To hash on the source and destination IP address for TCP IPv4 traffic, use the following:
ethtool -N <ethX> rx-flow-hash tcp4 sd
- To hash on the source and destination ports for UDP IPv6 traffic, use the following:
ethtool -N <ethX> rx-flow-hash udp6 sdfn

Application Device Queues (ADQ)

Application Device Queues (ADQ) allow you to dedicate one or more queues to a specific application. This can reduce latency for the specified application, and allow Tx traffic to be rate limited per application.



NOTE: The content in this section applies only to Intel Ethernet 700 Series devices. For information on ADQ support on other Intel devices, refer to the [ice](#) and [iavf](#) "Additional Configurations" subsections of this chapter.

Requirements:

- Kernel version 4.19.58 or later
- The sch_mqprio, act_mirred and cls_flower modules must be loaded. For example:

```
# modprobe sch_mqprio
# modprobe act_mirred
# modprobe cls_flower
```

- The latest version of iproute2. We recommend the following installation method:

```
# cd iproute2
# ./configure
# make DESTDIR=/opt/iproute2 install
# ln -s /opt/iproute2/sbin/tc /usr/local/sbin/tc
```

- NVM version 6.01 or later
- ADQ cannot be enabled when the following features are enabled: Data Center Bridging (DCB), Multiple Functions per Port (MFP), or Sideband Filters.
- If another driver (for example, DPDK) has set cloud filters, you cannot enable ADQ.

Creating Traffic Class Filters



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.



NOTE:

- Tunnel filters are not supported in ADQ. If encapsulated packets do arrive in non-tunnel mode, filtering will be done on the inner headers. For example, for VXLAN traffic in non-tunnel mode, if PCTYPE is identified as a VXLAN encapsulated packet, then the outer headers are ignored. Therefore, inner headers are matched.
- If a TC filter on a PF matches traffic over a VF (on the PF), that traffic will be routed to the appropriate queue of the PF, and will not be passed on the VF. Such traffic will end up getting dropped higher up in the TCP/IP stack as it does not match PF address data.
- If traffic matches multiple TC filters that point to different TCs, that traffic will be duplicated and sent to all matching TC queues. The hardware switch mirrors the packet to a VSI list when multiple filters are matched.

RDMA (Remote Direct Memory Access)

Remote Direct Memory Access, or RDMA, allows a network device to transfer data directly to and from application memory on another system, increasing throughput and lowering latency in certain networking environments.

The i40e driver supports the following RDMA protocols:

- iWARP (Internet Wide Area RDMA Protocol)



NOTES:

- RDMA requires auxiliary bus support.

For detailed installation and configuration information, see the README file in the RDMA driver tarball.

Auxiliary Bus

Inter-Driver Communication (IDC) is the mechanism in which LAN drivers (such as i40e) communicate with peer drivers (such as irdma). Starting in kernel 5.11, Intel LAN and RDMA drivers use an auxiliary bus mechanism for IDC.

RDMA functionality requires use of the auxiliary bus.

If your kernel supports the auxiliary bus, the LAN and RDMA drivers will use the inbox auxiliary bus for IDC. For kernels lower than 5.11, the base driver will automatically install an out-of-tree auxiliary bus module.

Source Pruning

The i40e driver allows you to enable or disable source MAC address pruning of Ethernet packets. You can use this feature without a bond created or when active-backup bonding is configured with ARP monitoring.

Use the `ethtool` private flag `disable-source-pruning` to enable or disable source MAC pruning on received packets:

```
# ethtool --set-priv-flags <ethX> disable-source-pruning on|off
```

Where:

- `off` - discards packets with source MAC address matching the source MAC address of the interface (default)
- `on` - receives packets with source MAC address matching the source MAC address of the interface



NOTE: When you set `disable-source-pruning` to `on` (enabled):

- The driver will disable spoof check.
- The network interface will receive all packets with a source MAC address matching the network interface's MAC address.

The i40e driver allows you to enable or disable source pruning on a specified VF using the `vf-source-pruning` private flag. You must also disable MAC anti-spoofing for the VF and enable trust mode for the VF. All three must be set for source pruning on a VF to function. This will allow you to configure VRRP (virtual router redundancy protocol) where one VF is designated as the primary device and all other VFs are secondary devices.

```
# ethtool --set-priv-flags <ethX> vf-source-pruning on|off
```

Where:

- `on` - discards packets with source MAC address matching the source MAC address of the interface (default)
- `off` - receives packets with source MAC address matching the source MAC address of the interface



NOTE: Enabling `vf-source-pruning` will not automatically set anti-spoofing and trust mode automatically. You must perform these steps yourself. They can be performed in any order. For example:

```
# ip link set <ethX> vf <vf id> spoofchk off
# ip link set <ethX> vf <vf id> trust on
# ethtool --set-priv-flag <ethX> vf-source-pruning off
```

For more information, refer to "MAC and VLAN Anti-Spoofing Feature for VFs" and "Trusted VFs and VF Promiscuous Mode" in this section.

EEE (Energy Efficient Ethernet)

A link between two EEE-compliant devices will result in periodic bursts of data followed by periods where the link is in an idle state. This Low Power Idle (LPI) state is supported at 2.5 Gbps and 5 Gbps link speeds.

 **NOTES:**

- EEE support requires auto-negotiation.
- Both link partners must support EEE.
- EEE is not supported on all Intel® Ethernet Network devices or at all link speeds.

Example:

```
# ethtool --show-eee <ethX>
# ethtool --set-eee <ethX> [eee on|off]
```

Disabling Physical Link When the Interface Is Brought Down

When the link-down-on-close private flag is set to "on", the port's link will go down when the interface is brought down using the 'ip link set <ethX> down' command.

Use ethtool to view and set link-down-on-close, as follows:

```
# ethtool --show-priv-flags <ethX>
# ethtool --set-priv-flags <ethX> link-down-on-close [on|off]
```

Jumbo Frames

 **NOTES:**

- The maximum MTU setting for Jumbo Frames is 9710 bytes. This value coincides with the maximum Jumbo Frames size of 9728 bytes.
- This driver will attempt to use multiple page sized buffers to receive each jumbo packet. This should help to avoid buffer starvation issues when allocating receive packets.
- Packet loss may have a greater impact on throughput when you use jumbo frames. If you observe a drop in performance after enabling jumbo frames, enabling flow control may mitigate the issue.

Speed and Duplex Configuration

The i40e driver supports setting the link speed using ethtool. You can only set speeds that the device actually supports. You cannot set duplex settings using ethtool.

To see the speed configurations your device supports, run the following:

```
# ethtool <ethX>
```

To set your device to a supported speed on Intel Ethernet 710 Series devices, use the following:

```
# ethtool -s <ethX> speed <desired speed in Mbps>
```

For example, to set the speed to 10 Gbps, use:

```
# ethtool -s <ethX> speed 10000
```

Alternately, you can use the `advertise` parameter to set the link speed. The `advertise` method is supported on all Intel Ethernet 700 Series devices.

To have your device advertise supported speeds, use the following:

```
# ethtool -s <ethX> advertise N
Where N is a bitmask of the desired speeds.
```

For example, to have your device advertise 10000baseSR Full, use:

```
# ethtool -s <ethX> advertise 0x800000000000
```

For more details, please refer to the ethtool man page.

IEEE 802.1ad (QinQ) Support

The IEEE 802.1ad standard, informally known as QinQ, allows for multiple VLAN IDs within a single Ethernet frame. VLAN IDs are sometimes referred to as "tags," and multiple VLAN IDs are thus referred to as a "tag stack." Tag stacks allow L2 tunneling and the ability to separate traffic within a particular VLAN ID, among other uses.

The following are examples of how to configure 802.1ad (QinQ):

```
# ip link add link eth0 eth0.24 type vlan proto 802.1ad id 24
# ip link add link eth0.24 eth0.24.371 type vlan proto 802.1Q id 371
```

Where "24" and "371" are example VLAN IDs.



NOTES:

- 802.1ad (QinQ) is supported in 3.19 and later kernels.
- Receive checksum offloads, cloud filters, and VLAN acceleration are not supported for 802.1ad (QinQ) packets.
- VLAN protocols use the following EtherTypes:
 - 802.1Q = EtherType 0x8100
 - 802.1ad = EtherType 0x88A8

Tunnel/Overlay Stateless Offloads

Supported tunnels and overlays include VXLAN, GENEVE, and others depending on hardware and software configuration. Stateless offloads are enabled by default.

To view the current state of all offloads:

```
# ethtool -k <ethX>
```

For more information on configuring your network for overlay HW offloading support on Intel Ethernet 700 Series devices, refer to Intel's [VXLAN Configuration Guide](#).

Data Center Bridging (DCB)

Note: The kernel assumes that TC0 is available, and will disable Priority Flow Control (PFC) on the device if TC0 is not available. To fix this, ensure TC0 is enabled when setting up DCB on your switch.

DCB is a configuration Quality of Service implementation in hardware. It uses the VLAN priority tag (802.1p) to filter traffic. That means that there are 8 different priorities that traffic can be filtered into. It also enables priority flow control (802.1Qbb) which can limit or eliminate the number of dropped packets during network stress. Bandwidth can be allocated to each of these priorities, which is enforced at the hardware level (802.1Qaz).

DCB is normally configured on the network using the DCBX protocol (802.1Qaz), a specialization of LLDP (802.1AB). The driver supports the following mutually exclusive variants of DCBX support::

- Firmware-based LLDP Agent
- Software-based LLDP Agent

In firmware-based mode, firmware intercepts all LLDP traffic and handles DCBX negotiation transparently for the user. In this mode, the adapter operates in "willing" DCBX mode, receiving DCB settings from the link partner (typically a switch). The local user can only query the negotiated DCB configuration. For information on configuring DCBX parameters on a switch, please consult the switch manufacturer's documentation.

In software-based mode, LLDP traffic is forwarded to the network stack and user space, where a software agent can handle it. In this mode, the adapter can operate in either "willing" or "nonwilling" DCBX mode and DCB configuration can be both queried and set locally. This mode requires the FW-based LLDP Agent to be disabled.

NOTE:

- You can enable and disable the firmware-based LLDP Agent using an ethtool private flag. Refer to the "FW-LLDP (Firmware Link Layer Discovery Protocol)" section in this README for more information.
- In software-based DCBX mode, you can configure DCB parameters using software LLDP/DCBX agents that interface with the Linux kernel's DCB Netlink API. We recommend using OpenLLDP as the DCBX agent when running in software mode. For more information, see the OpenLLDP man pages and <https://github.com/intel/openlldp>.
- The driver implements the DCB netlink interface layer to allow the user space to communicate with the driver and query DCB configuration for the port.

FW-LLDP (Firmware Link Layer Discovery Protocol)

Use ethtool to change FW-LLDP settings. The FW-LLDP setting is per port and persists across boots.

To enable LLDP:


```
# ethtool --set-priv-flags <ethX> disable-fw-lldp off
```

To disable LLDP:

```
# ethtool --set-priv-flags <ethX> disable-fw-lldp on
```


To check the current LLDP setting:


```
# ethtool --show-priv-flags <ethX>
```

 **NOTE:** You must enable the UEFI HII "LLDP Agent" attribute for this setting to take effect. If "LLDP AGENT" is set to disabled, you cannot enable it from the OS.

Forward Error Correction (FEC)

Allows you to set the Forward Error Correction (FEC) mode. FEC improves link stability, but increases latency. Many high quality optics, direct attach cables, and backplane channels provide a stable link without FEC.

 **NOTE:** For devices to benefit from this feature, link partners must have FEC enabled.

 **NOTE:** Intel® Ethernet Controller XXV710 devices support all FEC modes listed below.

On kernels older than 4.14, use the following private flags to disable FEC modes:

- `rs-fec` (0 to disable, 1 to enable)
- `base-r-fec` (0 to disable, 1 to enable)

On kernel 4.14 or later, use `ethtool` to get/set the following FEC modes:

- No FEC
- Auto FEC
- BASE-R FEC
- RS FEC

Dynamic Device Personalization

Dynamic Device Personalization (DDP) allows you to change the packet processing pipeline of a device by applying a profile package to the device at runtime. Profiles can be used to, for example, add support for new protocols, change existing protocols, or change default settings. DDP profiles can also be rolled back without rebooting the system.

Requirements:

- Intel Ethernet X710/XXV710/XL710 adapter
- Firmware 6.0 or newer
- RHEL 7.5 or later or Linux Kernel 4.0.1 or newer

To apply a profile, copy it first to the `intel/i40e/ddp` directory relative to your firmware root (usually `/lib/firmware` or `/lib/firmware/updates`).

For example:

```
/lib/firmware/intel/i40e/ddp
```

Then use the `ethtool -f|--flash` flag with region 100:

```
# ethtool -f <ethX> <profile name> 100
```

For example:

```
# ethtool -f eth0 gtp.pkgo 100
```

You can roll back to a previously loaded profile using `'-'` instead of profile name:

```
# ethtool -f <ethX> - 100
```

For example:

```
# ethtool -f eth0 - 100
```

For every rollback request one profile will be removed, from last to first (LIFO) order.

 **NOTE:**

- DDP profiles are loaded only on the interface corresponding to first physical function of the device (PF0), but the configuration is applied to all ports of the adapter.
- DDP profiles are not persistent. A system reboot will reset the device to its default configuration.
- DDP profiles are NOT automatically unloaded when the driver is unbound/unloaded. Please note that subsequent driver reload may corrupt the profile configuration during its initialization and is NOT recommended.
- DDP profiles should be manually rolled-back before driver unload/unbind if the intention is to start with clean HW configuration.
- Exercise caution while loading DDP profiles. Attempting to load files other than DDP profiles provided by Intel may cause system instability, system crashes, or system hangs.

More details about Dynamic Device Personalization can be found on the Intel Developer Zone site: <https://software.intel.com/en-us/articles/dynamic-device-personalization-for-intel-ethernet-700-series>

SR-IOV Hypervisor Management Interface

The sysfs file structure below supports the SR-IOV hypervisor management interface.

/sys/class/net/<ethX>/device/sriov (see 1 below)

```

+-- qos
| +-- [TC, 0-7]
|| +-- priority
|| +-- lsp
|| +-- max_bw
| +-- apply
+-- egress_mirror
+-- ingress_mirror
+-- tpid
+-- [VF-id, 0 .. 255] (see 2 below)
| +-- vlan_mirror
| +-- trunk
| +-- allow_untagged
| +-- egress_mirror
| +-- ingress_mirror
| +-- loopback
| +-- mac

```

```
| +--- mac_list
| +--- promisc
| +--- vlan_strip
| +--- enable
| +--- link_state
| +--- queue_type
| +--- num_queues
| +--- max_tx_rate
| +--- stats
|| +--- rx_bytes
|| +--- rx_packets
|| +--- rx_dropped
|| +--- tx_bytes
|| +--- tx_packets
|| +--- tx_dropped
|| +--- tx_errors
| +--- reset_stats
| +--- trust
| +--- qos
|| +--- [TC, 0-7]
|| +--- share
|| +--- Max_TC_TX_Rate
|| +--- share
```

**NOTES:**

1. kobject started from "sriov" is not available from existing kernel sysfs, and it requires device driver to implement this interface.
2. maximum number of SR-IOV instances is 256. The actual number of instances created depends on the value set for /sys/bus/pci/devices/<device pci address>/sriov_numvfs

SR-IOV hypervisor functions:

Parameter	Description and Examples
priority	<p>Sets the list of priority code point (PCP) values to map to the traffic class.</p> <p>Example 1: set priority 0 and 1 to traffic class 0. # echo 0,1 > /sys/class/net/plp1/device/sriov/qos/0/priority</p> <p>Example 2: display current setting for TC3. # cat /sys/class/net/plp1/device/sriov/qos/3/priority</p>
lsp	<p>Sets Link Strict Priority (LSP) for the traffic class.</p> <p>Example 1: set LSP for traffic class 0. # echo on > /sys/class/net/plp1/device/sriov/qos/0/lsp</p> <p>Example 2: display current LSP setting for TC0. # cat /sys/class/net/plp1/device/sriov/qos/0/lsp</p>
max_bw	<p>Sets the maximum bandwidth in Mbps for the traffic class on the PF.</p> <p>Example 1: set max bandwidth of 2Gbps for traffic class 2. # echo 2000 > /sys/class/net/plp1/device/sriov/qos/2/max_bw</p> <p>Example 2: display current setting for TC0. # cat /sys/class/net/plp1/device/sriov/qos/0/max_bw</p>
qos/apply	Applies the VF bandwidth configuration for the port. See "qos/share" below for more information.
egress_mirror	<p>Mirrors egress traffic from the PF to the specified VF on the same PF.</p> <p>Example 1: add egress traffic mirroring on PF p1p2 to VF 7. # echo add 7 > /sys/class/net/plp2/device/sriov/egress_mirror</p> <p>Example 2: remove egress traffic mirroring on PF p1p2 to VF 7. # echo rem 7 > /sys/class/net/plp2/device/sriov/egress_mirror</p>
ingress_mirror	<p>Mirrors ingress traffic from the PF to the specified VF on the same PF.</p> <p>Example 1: add ingress traffic mirroring on PF p1p2 to VF 7. # echo add 7 > /sys/class/net/plp2/device/sriov/ingress_mirror</p> <p>Example 2: remove ingress traffic mirroring on PF p1p2 to VF 7. # echo rem 7 > /sys/class/net/plp2/device/sriov/ingress_mirror</p>
tpid	<p>Specifies the TPID of the outer VLAN tag (S-tag). Can be set to 0x88A8 or 0x8100. This setting affects all VFs configured on the specified PF. Changing the TPID on PF0 results in a device-wide change and will restart all underlying VFs; you must manually reconfigure the TPID to the same value on all PFs associated with the NIC.</p> <p>Example 1: set TPID to 0x88A8. # echo 0x88a8 > /sys/class/net/plp0/device/sriov/tpid</p> <p>Example 2: show the configured value. # cat /sys/class/net/plp0/device/sriov/tpid</p>
vlan_mirror	<p>Supports both ingress and egress traffic mirroring. Supports two operations, add and rem:</p> <ul style="list-style-type: none"> • add: adds one or more VLAN IDs to a mirror list for a given VF. • rem: removes VLAN IDs from the mirror list for a given VF. <p>Example 1: mirror traffic based upon VLANs 2,4,6,18-22 to VF 3 of PF p1p1. # echo add 2,4,6,18-22 > /sys/class/net/plp1/device/sriov/3/vlan_mirror</p> <p>Example 2: remove VLAN 4, 15-17 from traffic mirroring at destination VF 3. # echo rem 15-17 > /sys/class/net/plp1/device/sriov/3/vlan_mirror</p>

Parameter	Description and Examples
	<p>Example 3: remove all VLANs from mirroring at VF 3.</p> <pre># echo rem 0 - 4095 > /sys/class/net/plp1/device/sriov/3/vlan_mirror</pre>
trunk	<p>Lists the VLANs to filter on. Supports two operations, add and rem:</p> <ul style="list-style-type: none"> • add: adds one or more VLAN IDs into VF VLAN filtering. • rem: removes VLAN IDs from the VF VLAN filtering list. <p>Example 1: add multiple VLAN tags, VLANs 2,4,5,10-20, by PF, p1p2, on a selected VF, 1, for filtering, with the sysfs support:</p> <pre># echo add 2,4,5,10-20 > /sys/class/net/plp2/device/sriov/1/trunk</pre> <p>Example 2: remove VLANs 5, 11-13 from PF p1p2 VF 1 with sysfs:</p> <pre># echo rem 5,11-13 > /sys/class/net/plp2/device/sriov/1/trunk</pre> <p>Note: for rem, if VLAN ID is not on the VLAN filtering list, the VLAN ID will be ignored.</p>
allow_untagged	<p>Supports enabling and disabling the filtering of untagged frames to the specified VF.</p> <p>Example 1: allow untagged packet to VF 1 on p1p2.</p> <pre># echo on > /sys/class/net/plp2/device/sriov/1/allow_untagged</pre> <p>Example 2: disable untagged frames.</p> <pre># echo off > /sys/class/net/plp2/device/sriov/1/allow_untagged</pre>
egress_mirror	<p>Supports egress traffic mirroring from this VF to the specified VF.</p> <p>Example 1: add egress traffic mirroring on PF p1p2 VF 1 to VF 7.</p> <pre># echo add 7 > /sys/class/net/plp2/device/sriov/1/egress_mirror</pre> <p>Example 2: remove egress traffic mirroring on PF p1p2 VF 1 to VF 7.</p> <pre># echo rem 7 > /sys/class/net/plp2/device/sriov/1/egress_mirror</pre>
ingress_mirror	<p>Supports ingress traffic mirroring from this VF to the specified VF.</p> <p>Example 1: mirror ingress traffic on PF p1p2 VF 1 to VF 7.</p> <pre># echo add 7 > /sys/class/net/plp2/device/sriov/1/ingress_mirror</pre> <p>Example 2: show current ingress mirroring configuration for VF 1.</p> <pre># cat /sys/class/net/plp2/device/sriov/1/ingress_mirror</pre>
loopback	<p>Supports Enable/Disable VEB/VEPA (Local loopback).</p> <p>Example 1: allow traffic switching between VFs on the same PF.</p> <pre># echo ON > /sys/class/net/plp2/device/sriov/loopback</pre> <p>Example 2: send Hairpin traffic to the switch to which the PF is connected.</p> <pre># echo OFF > /sys/class/net/plp2/device/sriov/loopback</pre> <p>Example 3: show loopback configuration.</p> <pre># cat /sys/class/net/plp2/device/sriov/loopback</pre>
mac	<p>Supports setting default MAC address. If MAC address is set by this command, the PF will not allow VF to change it using an MBOX request.</p> <p>Example 1: set default MAC address to VF 1.</p> <pre># echo "00:11:22:33:44:55" > /sys/class/net/plp2/device/sriov/1/mac</pre> <p>Example 2: show default MAC address.</p> <pre># cat /sys/class/net/plp2/device/sriov/1/mac</pre>
mac_list	<p>Supports adding additional MACs to the VF. The default MAC is taken from "ip link set p1p2 vf 1 mac 00:11:22:33:44:55" if configured. If not, a random address is assigned to the VF by the NIC. If the MAC is configured using the IP LINK command, the VF cannot change it via MBOX/AdminQ</p>

Parameter	Description and Examples
	<p>requests.</p> <p>Example 1: add mac 00:11:22:33:44:55 and 00:66:55:44:33:22 to PF p1p2 VF 1. <pre># echo add "00:11:22:33:44:55,00:66:55:44:33:22" > /sys/class/net/plp2/device/sriov/1/mac_list</pre></p> <p>Example 2: delete mac 00:11:22:33:44:55 from above VF device. <pre># echo rem 00:11:22:33:44:55 > /sys/class/net/plp2/device/sriov/1/mac_list</pre></p> <p>Example 3: display a VF MAC address list. <pre># cat /sys/class/net/plp2/device/sriov/1/mac_list</pre></p>
promisc	<p>Supports setting/unsetting VF device unicast promiscuous mode and multicast promiscuous mode.</p> <p>Example 1: set MCAST promiscuous on PF p1p2 VF 1. <pre># echo add mcast > /sys/class/net/plp2/device/sriov/1/promisc</pre></p> <p>Example 2: set UCAST promiscuous on PF p1p2 VF 1. <pre># echo add ucast > /sys/class/net/plp2/device/sriov/1/promisc</pre></p> <p>Example 3: unset MCAST promiscuous on PF p1p2 VF 1. <pre># echo rem mcast > /sys/class/net/plp2/device/sriov/1/promisc</pre></p> <p>Example 4: show current promiscuous mode configuration. <pre># cat /sys/class/net/plp2/device/sriov/1/promisc</pre></p> <p>Note: VFs set to promiscuous via this sysfs interface may not receive packets addressed to another VF on the same port. For another VF to receive the packets, you must enable VF true promiscuous mode via ethtool. See "Trusted VFs and VF Promiscuous Mode" in the README for more information on enabling true promiscuous mode.</p>
vlan_strip	<p>Supports enabling/disabling VF device outer VLAN stripping.</p> <p>Example 1: enable VLAN strip on VF 3. <pre># echo ON > /sys/class/net/plp1/device/sriov/3/vlan_strip</pre></p> <p>Example 2: disable VLAN striping VF 3. <pre># echo OFF > /sys/class/net/plp1/device/sriov/3/vlan_strip</pre></p>
enable	<p>Enables or disables the VF device.</p> <p>Notes:</p> <ul style="list-style-type: none"> • Enabling a VF will trigger a VF reset. • Enabling a VF does not start any queues in the hardware. • Disabling a VF will forcibly stop the queues and may lead to Tx timeouts on the VF or VM. • This feature is not designed to manage traffic flow. It's intended to help prevent or handle error conditions. <p>Example 1: enable VF 3. <pre># echo on > /sys/class/net/plp1/device/sriov/3/enable</pre></p> <p>Example 2: disable VF 3. <pre># echo off > /sys/class/net/plp1/device/sriov/3/enable</pre></p> <p>Example 3: show VF 3 enable state. <pre># cat /sys/class/net/plp1/device/sriov/3/enable</pre></p>
link_state	<p>Sets/displays link status.</p> <p>Example 1: display link status on link speed. <pre># cat /sys/class/net/plp2/device/sriov/1/link_state</pre></p>

Parameter	Description and Examples
	<p>Example 2: set VF 1 to track status of PF link. # echo auto > /sys/class/net/plp2/device/sriov/1/link_state</p> <p>Example 3: disable VF 1. # echo disable > /sys/class/net/plp2/device/sriov/1/link_state</p>
queue_type	<p>Sets the type of queues (0 RSS, 1 QoS).</p> <p>Example 1: set queue type RSS for VF 3. # echo 0 > /sys/class/net/plp1/device/sriov/3/queue_type</p> <p>Example 2: set type QoS for VF 3. # echo 1 > /sys/class/net/plp1/device/sriov/3/queue_type</p> <p>Example 3: show queue type for VF 3. # cat /sys/class/net/plp1/device/sriov/3/queue_type</p>
num_queues	<p>Sets the number of queues allocated for the VF. To change the number of queues, queue_type must be RSS.</p> <p>Note: Changing this value will trigger a VF reset, which may disrupt traffic. We recommend configuring this setting before traffic starts, not during runtime.</p> <p>Example 1: set 8 queues for VF 5 if queue_type is RSS. # echo 8 > /sys/class/net/plp1/device/sriov/5/num_queues</p> <p>Example 2: show VF 5 number of queues for VF 5 type. # cat /sys/class/net/plp1/device/sriov/5/num_queues</p>
max_tx_rate	<p>Sets the maximum transmit rate in Mbps for the VF.</p> <p>Note: This is ignored if TC QoS is in use. The maximum transmit rate limit is cleared and cannot be set once you configure Max_TC_TX_Rate limits for any of the TCs on the VF.</p> <p>Example 1: set 200Mbps limit for VF 3. # echo 200 > /sys/class/net/plp1/device/sriov/3/max_tx_rate</p> <p>Example 2: show max_tx_rate for VF 3. # cat /sys/class/net/plp1/device/sriov/3/max_tx_rate</p>
stats	<p>Supports getting VF statistics:</p> <ul style="list-style-type: none"> • rx_bytes • rx_packets • rx_dropped • tx_bytes • tx_packets • tx_dropped • tx_errors <p>Example 1: display anti-spoofing violations counter for VF 1. # cat /sys/class/net/plp2/device/sriov/1/stats/tx_error</p>
reset_stats	<p>Resets the VF's stats counters.</p> <p>Example 1: reset stats for VF 1. # echo 1 > /sys/class/net/plp2/device/sriov/1/stats/reset_stats</p>
Max_TC_TX_Rate	<p>Sets the maximum bandwidth in Mbps for the traffic class per VF.</p> <p>Example 1: set max sending rate for VF 0 TC2 to 2000Mbps. # echo 2000 > /sys/class/net/plp2/device/sriov/0/qos/2/max_tc_tx_rate</p>

Parameter	Description and Examples
	<pre># cat /sys/class/net/plp2/device/sriov/0/qos/2/max_tc_tx_rate</pre>
qos/share	<p>Sets the share of bandwidth for the specified VF(s).</p> <p>Note: This feature is limited to TC0 only. You cannot revert the share back to 0 once it has been set. You need to apply a change to hardware using the related sysfs node qos/apply. The qos/apply PF attribute applies the traffic shares to all VFs and all applicable TCs at once.</p> <p>Example 1: allocate 10% of bandwidth to VF 0, 20% to VF 1, and the remaining 70% of bandwidth shared equally among the other VFs plus PF. Note: For all the unspecified VFs (or PFs), the default value for "share" is 0.</p> <pre># echo 10 > /sys/class/net/plp1/device/sriov/0/qos/share # echo 20 > /sys/class/net/plp1/device/sriov/1/qos/share # echo 1 > /sys/class/net/plp1/device/sriov/qos/apply (kicks off a recalculation based upon bandwidth distribution parameters specified through qos/share sysfs)</pre> <p>Example 2: display current bandwidth allocation for each VF.</p> <pre># cat /sys/class/net/plp1/device/sriov/0/qos/share (return 10) # cat /sys/class/net/plp1/device/sriov/1/qos/share (return 20)</pre>

PHY Register Debug Dump

i40e devices support Phy register debug dump, which allows you to obtain runtime register values from the PHY and then write the results to a single dump file, for debugging connection and link issues.

This debug dump contains a snapshot of the PHY's existing configuration, such as the PCS Link Control Register, PCS Link Status registers, and other information.

To generate a PHY register debug dump file, do the following:

1. Use `ethtool -d` to dump the PHY registers. Refer to the `ethtool` man page for more information. For example:

```
# ethtool -d <ethX> dump.txt
```

NOTE:

- If a register does not exist or if the register cannot be read, the driver will return `0xAABBCCDD` for the register.
- The contents of the debug dump are not human-readable. You must work with Customer Support to decode the file.

6.5.2.4 Performance Optimization - i40e

The driver defaults are meant to fit a wide variety of workloads. If further optimization is required, we recommend experimenting with the following settings.

Small Frame Sizes

For better performance when processing small (64B) frame sizes:

1. Try enabling Hyper threading in the BIOS in order to increase the number of logical cores in the system.
2. Increase the number of queues available to the adapter:

```
# ethtool -L
```

IRQ to Adapter Queue Alignment

Pin the adapter's IRQs to specific cores by disabling the irqbalance service and using the included set_irq_affinity script. Please see the script's help text for further options.

The following settings will distribute the IRQs across all the cores evenly:

```
# scripts/set_irq_affinity -X all <interface1> , [ <interface2>, ...
]
```

The following settings will distribute the IRQs across all the cores that are local to the adapter (same NUMA node):

```
# scripts/set_irq_affinity -X local <interface1> , [ <interface2>,
... ]
```

For very CPU-intensive workloads, we recommend pinning the IRQs to all cores.

Rx Descriptor Ring Size

To reduce the number of Rx packet discards, increase the number of Rx descriptors for each Rx ring using ethtool.

Check if the interface is dropping Rx packets due to buffers being full (rx_dropped.nic means there is no PCIe bandwidth):

```
# ethtool -S <interface> | grep "rx_dropped"
```

If the previous command shows drops on queues, it may help to increase the number of descriptors using ethtool -G:

```
# ethtool -G rx <N>
```

Where <N> is the desired number of ring entries/descriptors.

This can provide temporary buffering for issues that create latency while the CPUs process descriptors.

Interrupt Rate Limiting

This driver supports an adaptive interrupt throttle rate (ITR) mechanism that is tuned for general workloads. The user can customize the interrupt rate control for specific workloads, via ethtool, adjusting the number of microseconds between interrupts.

To set the interrupt rate manually, you must disable adaptive mode:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off
```

For IP forwarding:

- Disable adaptive ITR and lower Rx and Tx interrupts per queue using ethtool.
- Setting rx-usecs and tx-usecs to 125 will limit interrupts to about 8000 interrupts per second per queue:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs 125 tx-usecs 125
```

For lower CPU utilization:

- Disable adaptive ITR and lower Rx and Tx interrupts. The examples below affect every queue of the specified interface.

- Setting rx-usecs and tx-usecs to 80 will limit interrupts to about 12,500 interrupts per second per queue:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs 80 tx-usecs 80
```

For reduced latency:

- Disable adaptive ITR and ITR by setting rx-usecs and tx-usecs to 0 using ethtool:

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs 0 tx-usecs 0
```

Per-queue interrupt rate settings:

- The following examples are for queues 1 and 3, but you can adjust other queues.
- To disable Rx adaptive ITR and set static Rx ITR to 10 microseconds or about 100,000 interrupts/second, for queues 1 and 3:

```
# ethtool --per-queue <ethX> queue_mask 0xa --coalesce adaptive-rx off rx-usecs 10
```

- To show the current coalesce settings for queues 1 and 3:

```
# ethtool --per-queue <ethX> queue_mask 0xa --show-coalesce
```

Bounding interrupt rates using rx-usecs-high:

- Valid Range: 0-235 (0=no limit)
The range of 0-235 microseconds provides an effective range of 4,310 to 250,000 interrupts per second. The value of rx-usecs-high can be set independently of rx-usecs and tx-usecs in the same ethtool command, and is also independent of the adaptive interrupt moderation algorithm. The underlying hardware supports granularity in 4-microsecond intervals, so adjacent values may result in the same interrupt rate.
- The following command would disable adaptive interrupt moderation, and allow a maximum of 5 microseconds before indicating a receive or transmit was complete. However, instead of resulting in as many as 200,000 interrupts per second, it limits total interrupts per second to 50,000 via the rx-usecs-high parameter.

```
# ethtool -C <ethX> adaptive-rx off adaptive-tx off rx-usecs-high 20 rx-usecs 5 tx-usecs 5
```

Virtualized Environments

In addition to the other suggestions in this section, the following may be helpful to optimize performance in VMs.

- Disable XPS on both ends by using the included virt_perf_default script or by running the following command as root:

```
for file in `ls /sys/class/net/ethX/queues/tx-*/xps_cpus`;
do echo 0 > $file; done
```

- Using the appropriate mechanism (vcpupin) in the VM, pin the CPUs to individual LCPUs, making sure to use a set of CPUs included in the device's local_cpulist:
/sys/class/net/<ethX>/device/local_cpulist.
- Configure as many Rx/Tx queues in the VM as available. (See the iavf driver documentation for the number of queues supported.) For example:

```
# ethtool -L <virt_interface> rx <max> tx <max>
```

6.5.3 iavf Linux Driver

The iavf (Intel Adaptive Virtual Function) driver is the VF driver for Intel Ethernet 700 Series and 800 Series devices. The iavf driver allows you to upgrade your hardware without needing to upgrade the virtual function driver in each of the VMs running on top of the hardware.

To enable SR-IOV on your system:

1. Ensure both Virtualization and SR-IOV are enabled in the BIOS.
2. Install the Linux operating system. You can verify that the KVM driver is loaded by typing: `lsmod | grep -i kvm`
3. Load the Linux Base Driver using the `modprobe` command: `modprobe <driver name> option max_vfs=xx,yy`

xx and yy are the number of virtual functions you want to create. You must specify a number for each port with each parameter separated by a comma. For example, xx is the number of virtual functions for port 1; and yy, for port 2. You can create up to 63 functions per port.
4. Compile and install the iavf driver for SR-IOV. This is loaded against the virtual functions created.



NOTE:

- For VLANs, there is a limit of a total of 32 shared VLANs to 1 or more virtual functions.
- The MTU size set on a VF should match the MTU size set on the PF. A mismatch in MTU sizes may cause unexpected results.

6.5.3.1 Adaptive Virtual Function

Adaptive Virtual Function (AVF) allows the virtual function driver, or VF, to adapt to changing feature sets of the physical function driver (PF) with which it is associated. This allows system administrators to update a PF without having to update all the VFs associated with it. All AVFs have a single common device ID and branding string.

AVFs have a minimum set of features known as "base mode," but may provide additional features depending on what features are available in the PF with which the AVF is associated. The following are base mode features:

- 4 Queue Pairs (QP) and associated Configuration Status Registers (CSRs) for Tx/Rx
- iavf descriptors and ring format
- Descriptor write-back completion
- 1 control queue, with iavf descriptors, CSRs and ring format
- 5 MSI-X interrupt vectors and corresponding iavf CSRs
- 1 Interrupt Throttle Rate (ITR) index
- 1 Virtual Station Interface (VSI) per VF
- 1 Traffic Class (TC), TC0
- Receive Side Scaling (RSS) with 64 entry indirection table and key, configured through the PF
- 1 unicast MAC address reserved per VF
- 8 MAC address filters for each VF on an Intel Ethernet 800 Series device
- 16 MAC address filters for each VF on an Intel Ethernet 700 Series device

- Stateless offloads - non-tunneled checksums
- AVF device ID
- HW mailbox is used for VF to PF communications (including on Windows)

6.5.3.2 Command Line Parameters - iavf

The iavf driver does not support any command line parameters.

6.5.3.3 Additional Configurations - iavf



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

Setting VLAN Tag Stripping

If you have applications that require Virtual Functions (VFs) to receive packets with VLAN tags, you can disable VLAN tag stripping for the VF. The Physical Function (PF) processes requests issued from the VF to enable or disable VLAN tag stripping. Note that if the PF has assigned a VLAN to a VF, then requests from that VF to set VLAN tag stripping will be ignored.

To enable/disable VLAN tag stripping for a VF, issue the following command from inside the VM in which you are running the VF:

```
# ethtool -K <ethX> rxvlan on/off
```

or alternatively:

```
# ethtool --offload <ethX> rxvlan on/off
```

IEEE 802.1ad (QinQ) Support

The IEEE 802.1ad standard, informally known as QinQ, allows for multiple VLAN IDs within a single Ethernet frame. VLAN IDs are sometimes referred to as "tags," and multiple VLAN IDs are thus referred to as a "tag stack." Tag stacks allow L2 tunneling and the ability to separate traffic within a particular VLAN ID, among other uses.

The following are examples of how to configure 802.1ad (QinQ):

```
# ip link add link eth0 eth0.24 type vlan proto 802.1ad id 24
```

```
# ip link add link eth0.24 eth0.24.371 type vlan proto 802.1Q id 371
```

Where "24" and "371" are example VLAN IDs.



NOTES:

- 802.1ad (QinQ) is supported in 3.19 and later kernels.
- VLAN protocols use the following EtherTypes:
 - 802.1Q = EtherType 0x8100
 - 802.1ad = EtherType 0x88A8
- For QinQ traffic to work at MTU 1500, the L2 peer (switch port or another NIC) should be able to receive Ethernet frames of 1526 bytes. Some third-party NICs support a maximum Ethernet frame size of 1522 bytes at MTU 1500, which will cause QinQ traffic to fail. To work around this issue, restrict the MTU on the Intel Ethernet device to 1496.

Double VLANs

Devices based on the Intel Ethernet 800 Series can process up to two VLANs in a packet when all the following are installed:

- ice driver version 1.4.0 or later
- NVM version 2.4 or later
- ice DDP package version 1.3.21 or later

If you don't use the versions above, the only supported VLAN configuration is single 802.1Q VLAN traffic.

When two VLAN tags are present in a packet, the outer VLAN tag can be either 802.1Q or 802.1ad. The inner VLAN tag must always be 802.1Q.

NOTE THE FOLLOWING LIMITATIONS:

- For each VF, the PF can only allow VLAN hardware offloads (insertion and stripping) of one type, either 802.1Q or 802.1ad.

To enable outer or single 802.1Q VLAN insertion and stripping and disable 802.1ad VLAN insertion and stripping:

```
# ethtool -K <ethX> rxvlan on txvlan on rx-vlan-stag-hw-parse off
tx-vlan-stag-hw-insert off
```

To enable outer or single 802.1ad VLAN insertion and stripping and disable 802.1Q VLAN insertion and stripping:

```
# ethtool -K <ethX> rxvlan off txvlan off rx-vlan-stag-hw-parse on
tx-vlan-stag-hw-insert on
```

To enable outer or single VLAN filtering if the VF supports modifying VLAN filtering:

```
# ethtool -K <ethX> rx-vlan-filter on rx-vlan-stag-filter on
```

To disable outer or single VLAN filtering if the VF supports modifying VLAN filtering:

```
# ethtool -K <ethX> rx-vlan-filter off rx-vlan-stag-filter off
```

Combining QinQ with SR-IOV VFs

We recommend you always configure a port VLAN for the VF from the PF. If a port VLAN is not configured, the VF driver may only offload VLANs via software. The PF allows all VLAN traffic to reach the VF, and the VF manages all VLAN traffic.

When the device is configured for double VLANs and the PF has configured a port VLAN:

- The VF can only offload guest VLANs for 802.1Q traffic.
- The VF can only configure VLAN filtering rules for guest VLANs using 802.1Q traffic.

However, when the device is configured for double VLANs and the PF has NOT configured a port VLAN:

- You must use iavf driver version 4.1.0 or later to offload and filter VLANs.
- The PF turns on VLAN pruning and antispoof in the VF's VSI by default. The VF will not transmit or

receive any tagged traffic until the VF requests a VLAN filter.

- The VF can offload (insert and strip) the outer VLAN tag of 802.1Q or 802.1ad traffic.
- The VF can create filter rules for the outer VLAN tag of both 802.1Q and 802.1ad traffic.

If the PF does not support double VLANs, the VF can hardware offload single 802.1Q VLANs without a port VLAN.

When the PF is enabled for double VLANs, for iavf drivers before version 4.1.x:

- VLAN hardware offloads and filtering are supported only when the PF has configured a port VLAN.
- VLAN filtering, insertion, and stripping will be software offloaded when no port VLAN is configured.

To see VLAN filtering and offload capabilities, use the following command:

```
# ethtool -k <ethX> | grep vlan
```

Application Device Queues (ADQ)

Application Device Queues (ADQ) allow you to dedicate one or more queues to a specific application. This can reduce latency for the specified application, and allow Tx traffic to be rate limited per application.

For Devices Based on the Intel Ethernet 800 Series

For requirements and configuration information for Intel Ethernet 800 Series devices, refer to the [Intel® Ethernet Controller E810 Application Device Queues \(ADQ\) Configuration Guide](#).

For Devices Based on the Intel Ethernet 700 Series

Requirements:

- Kernel version: Varies by feature and the underlying PF device. Refer to the E810 ADQ Configuration Guide for more information on required kernel versions for different ADQ features on Intel(R) Ethernet 800 Series devices.
- Depending on the underlying PF device, ADQ cannot be enabled when the following features are enabled: Data Center Bridging (DCB), Multiple Functions per Port (MFP), or Sideband Filters.
- If another driver (for example, DPDK) has set cloud filters, you cannot enable ADQ.

RDMA in the VF

Devices based on the Intel Ethernet 800 Series support RDMA in a Linux VF, on supported Windows or Linux hosts.

The iavf driver supports the following RDMA protocols in the VF:

- iWARP (Internet Wide Area RDMA Protocol)
- RoCEv2 (RDMA over Converged Ethernet)

Refer to the README inside the irdma driver tarball for details on configuring RDMA in the VF.



NOTE: To support VF RDMA, load the irdma driver on the host before creating VFs. Otherwise VF RDMA support may not be negotiated between the VF and PF driver.

The iavf driver allocates MSI-X resources for the VF RDMA instance (irdma). The LAN iavf driver gets first priority and any leftover MSI-X interrupts are used for VF RDMA.

Auxiliary Bus

Inter-Driver Communication (IDC) is the mechanism in which LAN drivers (such as iavf) communicate with peer drivers (such as irdma). Starting in kernel 5.11, Intel LAN and RDMA drivers use an auxiliary bus mechanism for IDC.

RDMA functionality requires use of the auxiliary bus.

If your kernel supports the auxiliary bus, the LAN and RDMA drivers will use the inbox auxiliary bus for IDC. For kernels lower than 5.11, the base driver will automatically install an out-of-tree auxiliary bus module.

6.5.4 ixgbe Linux Driver for Intel Ethernet 10 Gigabit Server Adapters

This section describes the ixgbe Linux base driver for 10 Gigabit Intel® Network Connections.

6.5.4.1 Command Line Parameters - ixgbe

If the driver is built as a module, the following optional parameters are used by entering them on the command line with the modprobe command using this syntax:


```
# modprobe ixgbe [<option>=<VAL1>,<VAL2>,...]
```




For example:







```
# modprobe ixgbe InterruptThrottleRate=16000,16000
```


The default value for each parameter is generally the recommended setting, unless otherwise noted.


The following table contains parameters and possible values for modprobe commands:



Parameter Name	Valid Range/Settings	Default	Description
RSS	0 - 16	1	<p>0 = Assign up to the lesser value of the number of CPUs or the number of queues</p> <p>X = Assign X queues, where X is less than or equal to the maximum number of queues</p> <p>RSS also effects the number of transmit queues allocated on 2.6.23 and newer kernels with CONFIG_NET_MULTIQUEUE set in the kernel .config file. CONFIG_NETDEVICES_MULTIQUEUE is only supported in kernels 2.6.23 to 2.6.26. For kernels 2.6.27 or newer, other options enable multiqueue.</p>
Multiqueue	0, 1	1	<p>0 = Disables Multiple Queue support</p> <p>1 = Enables Multiple Queue support (a prerequisite for RSS)</p>
Direct Cache Access (DCA)	0, 1		<p>0 = Disables DCA support in the driver</p> <p>1 = Enables DCA support in the driver</p> <p>If the driver is enabled for DCA, this parameter allows load-time control of the feature.</p> <p> NOTE: DCA is not supported on X550-based adapters.</p>
IntMode	0 - 2	2	<p>Interrupt mode controls the allowed load time control over the type of interrupt registered for by the driver. MSI-X is required for multiple queue support. Some kernels and combinations of kernel .config options will force a lower level of interrupt support. 'cat/proc/interrupts' will show different values for each type of interrupt.</p> <p>0 = Legacy Interrupts</p> <p>1 = MSI Interrupts</p>

Parameter Name	Valid Range/Settings	Default	Description
			2 = MSI-X interrupts
InterruptThrottleRate	956 - 488,281 (0=off, 1=dynamic)	1	<p>0=off</p> <p>1=dynamic</p> <p><min_ITR>-<max_ITR></p> <p>Interrupt Throttle Rate controls the number of interrupts each interrupt vector can generate per second. Increasing ITR lowers latency at the cost of increased CPU utilization, though it may help throughput in some circumstances.</p> <ul style="list-style-type: none"> 0 = Setting InterruptThrottleRate to 0 turns off any interrupt moderation and may improve small packet latency. However, this is generally not suitable for bulk throughput traffic due to the increased CPU utilization of the higher interrupt rate. 1 = Setting InterruptThrottleRate to Dynamic mode attempts to moderate interrupts per vector while maintaining very low latency. This can sometimes cause extra CPU utilization. If planning on deploying this driver in a latency sensitive environment, this parameter should be considered. <min_ITR>-<max_ITR> = Setting InterruptThrottleRate to a value greater or equal to <min_ITR> will program the adapter to send at most that many interrupts per second, even if more packets have come in. This reduces interrupt load on the system and can lower CPU utilization under heavy load, but will increase latency as packets are not processed as quickly. <p> On 82599 and X550-based adapters, disabling InterruptThrottleRate will also result in the driver disabling HW RSC.</p> <p> On 82598-based adapters, disabling InterruptThrottleRate will also result in disabling LRO (Large Receive Offloads).</p>
LLI			<p>Low Latency Interrupts (LLI) allow for immediate generation of an interrupt upon processing receive packets that match certain criteria as set by the parameters described below. LLI parameters are not enabled when Legacy interrupts are used. You must be using MSI or MSI-X (see cat /proc/interrupts) to successfully use LLI.</p> <p> NOTE: LLI is not supported on X550-based adapters.</p>
LLIPort	0 - 65535	0 (disabled)	<p>LLI is configured with the LLIPort command line parameter, which specifies which TCP port should generate Low Latency Interrupts.</p> <p>For example, using LLIPort=80 would cause the board to generate an immediate interrupt upon receipt of any packet sent to TCP port 80 on the local machine.</p>

Parameter Name	Valid Range/Settings	Default	Description
			 WARNING: Enabling LLI can result in an excessive number of interrupts/second that may cause problems with the system and in some cases may cause a kernel panic.  NOTE: LLI is not supported on X550-based adapters.
LLIPush	0 - 1	0 (disabled)	LLIPush can be set to enabled or disabled (default). It is most effective in an environment with many small transactions.  NOTE: Enabling LLIPush may allow a denial of service attack. LLI is not supported on X550-based adapters.
LLISize	0 - 1500	0 (disabled)	LLISize causes an immediate interrupt if the board receives a packet smaller than the specified size.  NOTE: LLI is not supported on X550-based adapters.
LLIType	0 - x8FFF	0 (disabled)	This parameter specifies the Low Latency Interrupt (LLI) Ethernet protocol type.  NOTE: LLI is not supported on X550-based adapters.
LLIVLANP	0 - 7	0 (disabled)	This parameter specifies the LLI on VLAN priority threshold.  NOTE: LLI is not supported on X550-based adapters.
FdirPballoc	1 - 3	1 (64k)	Specifies the Intel Ethernet Flow Director allocated packet buffer size. 1 = 64k 2 = 128k 3 = 256k
AtrSampleRate	0 - 255	20	This parameter is used with the Flow Director and is the software ATR transmit packet sample rate. For example, when AtrSampleRate is set to 20, every 20th packet looks to see if the packet will create a new flow. A value of 0 indicates that ATR should be disabled and no samples will be taken.
max_vfs	1 - 63	0	This parameter adds support for SR-IOV. It causes the driver to spawn up to max_vfs worth of virtual functions. If the value is greater than 0 it will also force the VMDq parameter to be 1 or more. NOTE: This parameter is only used on kernel 3.7.x and below.

Parameter Name	Valid Range/Set-tings	Default	Description
			<p>On kernel 3.8.x and above, use sysfs to enable VFs. Use sysfs for Red Hat distributions.</p> <p>For example, you can create 4 VFs as follows:</p> <pre># echo 4 > /sys/class/net/<ethX>/device/sriov_numvfs</pre> <p>To disable VFs, write 0 to the same file:</p> <pre># echo 0 > /sys/class/net/<ethX>/device/sriov_numvfs</pre> <p>The parameters for the driver are referenced by position. Thus, if you have a dual port adapter, or more than one adapter in your system, and want N virtual functions per port, you must specify a number for each port with each parameter separated by a comma. For example:</p> <pre># modprobe ixgbe max_vfs=4</pre> <p>This will spawn 4 VFs on the first port.</p> <pre># modprobe ixgbe max_vfs=2,4</pre> <p>This will spawn 2 VFs on the first port and 4 VFs on the second port.</p> <p> NOTES:</p> <ul style="list-style-type: none"> • Caution must be used in loading the driver with these parameters. Depending on your system configuration, number of slots, etc., it is impossible to predict in all cases where the positions would be on the command line. • Neither the device nor the driver control how VFs are mapped into config space. Bus layout will vary by operating system. On operating systems that support it, you can check sysfs to find the mapping. <p>When either SR-IOV mode or VMDq mode is enabled, hardware VLAN filtering and VLAN tag stripping/insertion will remain enabled. Please remove the old VLAN filter before the new VLAN filter is added. For example:</p> <pre># ip link set eth0 vf 0 vlan 100 // set vlan 100 for VF 0</pre> <pre># ip link set eth0 vf 0 vlan 0 // Delete vlan 100</pre> <pre># ip link set eth0 vf 0 vlan 200 // set a new vlan 200 for VF 0</pre> <p>With kernel 3.6, the driver supports the simultaneous usage of max_vfs and DCB features, subject to the constraints described below. Prior to kernel 3.6, the driver did not support the simultaneous operation of max_vfs greater than 0 and the DCB features (multiple traffic classes utilizing Priority Flow Control and Extended Transmission Selection).</p> <p>When DCB is enabled, network traffic is transmitted and received through multiple traffic classes (packet buffers in the NIC). The traffic is associated with a specific class based on</p>

Parameter Name	Valid Range/Set-tings	Default	Description
			<p>priority, which has a value of 0 through 7 used in the VLAN tag. When SR-IOV is not enabled, each traffic class is associated with a set of receive/transmit descriptor queue pairs. The number of queue pairs for a given traffic class depends on the hardware configuration. When SR-IOV is enabled, the descriptor queue pairs are grouped into pools. The Physical Function (PF) and each Virtual Function (VF) is allocated a pool of receive/transmit descriptor queue pairs. When multiple traffic classes are configured (for example, DCB is enabled), each pool contains a queue pair from each traffic class. When a single traffic class is configured in the hardware, the pools contain multiple queue pairs from the single traffic class.</p> <p>The number of VFs that can be allocated depends on the number of traffic classes that can be enabled. The configurable number of traffic classes for each enabled VF is as follows:</p> <ul style="list-style-type: none"> • 0 - 15 VFs = Up to 8 traffic classes, depending on device support • 16 - 31 VFs = Up to 4 traffic classes • 32 - 63 VFs = 1 traffic class <p>When VFs are configured, the PF is allocated one pool as well. The PF supports the DCB features with the constraint that each traffic class will only use a single queue pair. When zero VFs are configured, the PF can support multiple queue pairs per traffic class.</p>
LRO	0-1		<p>0=off, 1=on</p> <p>Large Receive Offload (LRO) is a technique for increasing inbound throughput of high-bandwidth network connections by reducing CPU overhead. It works by aggregating multiple incoming packets from a single stream into a larger buffer before they are passed higher up the networking stack, thus reducing the number of packets that have to be processed. LRO combines multiple Ethernet frames into a single receive in the stack, thereby potentially decreasing CPU utilization for receives.</p> <p>This technique is also referred to as Hardware Receive Side Coalescing (HW RSC). X550-based adapters support HW RSC. The LRO parameter controls HW RSC enablement.</p> <p>You can verify that the driver is using LRO by looking at these counters in ethtool:</p> <ul style="list-style-type: none"> • hw_rsc_aggregated - counts total packets that were combined • hw_rsc_flushed - counts the number of packets flushed out of LRO <p> NOTE: IPv6 and UDP are not supported by LRO.</p>
EEE	0-1		<p>0 = Disables EEE</p> <p>1 = Enables EEE</p> <p>A link between two EEE-compliant devices will result in</p>

Parameter Name	Valid Range/Set-tings	Default	Description
			<p>periodic bursts of data followed by periods where the link is in an idle state. This Low Power Idle (LPI) state is supported at 1 Gbps and 10 Gbps link speeds.</p> <p> NOTES:</p> <ul style="list-style-type: none"> • EEE support requires auto-negotiation. • Both link partners must support EEE. • EEE is not supported on all Intel® Ethernet Network devices or at all link speeds.
DMAC	0, 41-10000		<p>This parameter enables or disables DMA Coalescing feature. Values are in microseconds and set the internal DMA Coalescing internal timer.</p> <p> NOTE: DMAC is available on Intel® X550 (and later) based adapters.</p> <p>DMA (Direct Memory Access) allows the network device to move packet data directly to the system's memory, reducing CPU utilization. However, the frequency and random intervals at which packets arrive do not allow the system to enter a lower power state. DMA Coalescing allows the adapter to collect packets before it initiates a DMA event. This may increase network latency but also increases the chances that the system will enter a lower power state.</p> <p>Turning on DMA Coalescing may save energy with kernel 2.6.32 and later. DMA Coalescing must be enabled across all active ports in order to save platform power.</p> <p>InterruptThrottleRate (ITR) should be set to dynamic. When ITR=0, DMA Coalescing is automatically disabled.</p> <p>A guide containing information on how to best configure your platform is available on the Intel website.</p>
MDD	0-1	1 (enabled)	<p>0 = Disabled 1 = Enabled</p> <p>This parameter is only relevant for devices operating in SR-IOV mode. When this parameter is set, the driver detects malicious VF driver and disables its Tx/Rx queues until a VF driver reset occurs.</p>

6.5.4.2 Additional Configurations - ixgbe



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

Jumbo Frames



NOTES:

- The maximum MTU setting for Jumbo Frames is 9710 bytes. This value coincides with the maximum Jumbo Frames size of 9728 bytes.
- This driver will attempt to use multiple page sized buffers to receive each jumbo packet. This should help to avoid buffer starvation issues when allocating receive packets.
- Packet loss may have a greater impact on throughput when you use jumbo frames. If you observe a drop in performance after enabling jumbo frames, enabling flow control may mitigate the issue.
- For 82599-based network connections, if you are enabling jumbo frames in a virtual function (VF), jumbo frames must first be enabled in the physical function (PF). The VF MTU setting cannot be larger than the PF MTU.

Speed and Duplex Configuration

In addressing speed and duplex configuration issues, you need to distinguish between copper-based adapters and fiber-based adapters.

In the default mode, an Intel® Ethernet Network Adapter using copper connections will attempt to auto-negotiate with its link partner to determine the best setting. If the adapter cannot establish link with the link partner using auto-negotiation, you may need to manually configure the adapter and link partner to identical settings to establish link and pass packets. This should only be needed when attempting to link with an older switch that does not support auto-negotiation or one that has been forced to a specific speed or duplex mode. Your link partner must match the setting you choose. 1 Gbps speeds and higher cannot be forced. Use the autonegotiation advertising setting to manually set devices for 1 Gbps and higher.

Speed, duplex, and autonegotiation advertising are configured through the ethtool utility. To see the speed configurations your device supports, run the following:

```
# ethtool <ethX>
```

By default, devices based on the Intel® Ethernet Controller x550 do not advertise 2.5 Gbps or 5 Gbps. To have your device advertise these speeds, use the following:

```
# ethtool -s <ethx> advertise N
```

Where **N** is a combination of the following:

100baseTFull	0x008
1000baseTFull	x020
2500baseTFull	0x800000000000
5000baseTFull	0x1000000000000
10000baseTFull	0x1000

For example, to turn on all modes:

```
# ethtool -s <ethX> advertise 0x1800000001028
```

For more details, please refer to the ethtool man page.

note



NOTE: On Linux systems with INTERFACES(5), this can be specified as a pre-up command in `/etc/network/interfaces` so that the interface is always brought up with NBASE-T support. For example:

```
# iface <ethX> inet dhcp
pre-up ethtool -s <ethX> advertise 0x1800000001028 || true
```



CAUTION: Only experienced network administrators should force speed and duplex or change autonegotiation advertising manually. The settings at the switch must always match the adapter settings. Adapter performance may suffer or your adapter may not operate if you configure the adapter differently from your switch.

An Intel® Ethernet Network Adapter using fiber-based connections will not attempt to auto-negotiate with its link partner since those adapters operate only in full duplex and only at their native speed.



NOTE: For the Intel® Ethernet Connection X552 10 GbE SFP+, you must specify the desired speed.

Intel® Ethernet Flow Director

The ixgbe driver supports the following flow types:

- IPv4
- TCPv4
- UDPv4
- TCPv6
- UDPv6

Each flow type supports valid combinations of IP addresses (source or destination) and UDP/TCP ports (source and destination). You can supply only a source IP address, a source IP address and a destination port, or any combination of one or more of these four parameters. NOTE: This driver does not support IPv6 source or destination IP addresses.



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for more information on configuring Intel Ethernet Flow Director and its filters.

Support for UDP RSS

This feature adds an ON/OFF switch for hashing over certain flow types. Only UDP can be turned on. The default setting is disabled.

Only support for enabling/disabling hashing on ports for UDP over IPv4 (UDP4) or IPv6 (UDP6) is supported.



NOTE: Fragmented packets may arrive out of order when RSS UDP support is configured.

Supported ethtool Commands and Options

```
-n --show-nfc
```

Retrieves the receive network flow classification configurations.

```
rx-flow-hash tcp4|udp4|ah4|esp4|sctp4|tcp6|udp6|ah6|esp6|sctp6
```

Retrieves the hash options for the specified network traffic type.

```
-N --config-nfc
```

Configures the receive network flow classification.

```
rx-flow-hash tcp4|udp4|ah4|esp4|sctp4|tcp6|udp6|ah6|esp6|sctp6 m|v|t|s|d|f|n|r...
```

Configures the hash options for the specified network traffic type.

```
udp4 UDP over IPv4
```

```
udp6 UDP over IPv6
```

```
f Hash on bytes 0 and 1 of the Layer 4 header of the rx packet.
```

```
n Hash on bytes 2 and 3 of the Layer 4 header of the rx packet.
```

Parameters FdirPballoc and AtrSampleRate impact Intel Ethernet Flow Director.

Data Center Bridging (DCB)

Note: The kernel assumes that TC0 is available, and will disable Priority Flow Control (PFC) on the device if TC0 is not available. To fix this, ensure TC0 is enabled when setting up DCB on your switch.

DCB is a configuration Quality of Service implementation in hardware. It uses the VLAN priority tag (802.1p) to filter traffic. That means that there are 8 different priorities that traffic can be filtered into. It also enables priority flow control (802.1Qbb) which can limit or eliminate the number of dropped packets during network stress. Bandwidth can be allocated to each of these priorities, which is enforced at the hardware level (802.1Qaz).

DCB is normally configured on the network using the DCBX protocol (802.1Qaz), a specialization of LLDP (802.1AB). The driver supports the following variants of DCBX support::

- Software-based DCBX mode only

In software-based mode, LLDP traffic is forwarded to the network stack and user space, where a software agent can handle it. In this mode, the adapter can operate in either "willing" or "nonwilling" DCBX mode and DCB configuration can be both queried and set locally.

NOTE:

- In software-based DCBX mode, you can configure DCB parameters using software LLDP/DCBX agents that interface with the Linux kernel's DCB Netlink API. We recommend using OpenLLDP as the DCBX agent when running in software mode. For more information, see the OpenLLDP man pages and <https://github.com/intel/openlldp>.

Controlling the VF Link State

The ixgbe PF driver allows a root user to disable the VF link state using iproute2.

To control the VF link state, use the following:

```
# ip link set <ethX> vf <number> state auto|disable
```

Where:

`auto`: automatically reflects the link state from the PF to the VF (default)

`disable`: tells the hardware to drop any packets sent by the VF



NOTE: The `ixgbe` driver does not support the `enable` option.

If the command does not work, it may not be supported by your system.

Virtual Function (VF) Tx Rate Limit

Use the `ip` command to configure the Tx rate limit for a VF from the PF interface.

For example, to set a Tx rate limit of 1000Mbps for VF 0:

```
# ip link set eth0 vf 0 rate 1000
```

Note that the limit is set per queue and not for the entire VF interface.

Malicious Driver Detection (MDD) for VFs

Some Intel Ethernet devices use Malicious Driver Detection (MDD) to detect malicious traffic from the VF and disable Tx/Rx queues or drop the offending packet until a VF driver reset occurs. You can view MDD messages in the PF's system log using the `dmesg` command.

- If the PF driver logs MDD events from the VF, confirm that the correct VF driver is installed.
- To restore functionality, you can manually reload the VF or VM.

MAC and VLAN Anti-Spoofing Feature for VFs

When a malicious driver on a Virtual Function (VF) interface attempts to send a spoofed packet, it is dropped by the hardware and not transmitted.

An interrupt is sent to the PF driver notifying it of the spoof attempt. When a spoofed packet is detected, the PF driver will send the following message to the system log (displayed by the "`dmesg`" command):

```
ixgbe <ethX>: ixgbe_spoof_check: n spoofed packets detected
```

where "X" is the PF interface number and "n" is number of spoofed packets.

This feature can be disabled for a specific VF:

```
# ip link set <pf dev> vf <vf id> spoofchk {off|on}
```

Setting MAC Address, VLAN and Rate Limit Using IProute2 Tool

You can set a MAC address of a Virtual Function (VF), a default VLAN and the rate limit using the IProute2 tool. Download the latest version of the `iproute2` tool from Sourceforge if your version does not have all the features you require.

Wake on LAN (WoL) Support



NOTE: The Intel® Ethernet Converged Network Adapter X550-T1 and Intel® Ethernet Converged Network Adapter X550-T2 have a manageability/AUX power connector. These devices only support WoL if AUX power is supplied via this connector. Note that this is system and adapter specific. Some with this connector do not support WoL. Some systems do not provide the correct power connection. See your system documentation for details.

Tunnel/Overlay Stateless Offloads

Supported tunnels and overlays include VXLAN, GENEVE, and others depending on hardware and software configuration. Stateless offloads are enabled by default.

To view the current state of all offloads:

```
# ethtool -k <ethX>
```

For more information on configuring your network for overlay HW offloading support on Intel Ethernet 500 Series devices, refer to the Intel technical brief, "[Creating Overlay Networks Using Intel Ethernet Converged Network Adapters.](#)"

Interrupt Rate Limiting

This driver supports an adaptive interrupt throttle rate (ITR) mechanism that is tuned for general workloads. The user can customize the interrupt rate control for specific workloads, via ethtool, adjusting the number of microseconds between interrupts.

Syntax:

```
# ethtool -C <ethX> rx-usecs N
```

Values for N:

- 0 - no limit
- 1 - adaptive (default)
- 2-1022 - minimum microseconds between each interrupt

The range of 0-1022 microseconds provides an effective range of 978 to 500,000 interrupts per second. The underlying hardware supports granularity in 2us intervals at 1Gbps and 10Gbps and 20us at 100Mbps, so adjacent values may result in the same interrupt rate.

For lower CPU utilization:

- Lower Rx and Tx interrupts per queue using ethtool.
- Setting rx-usecs to 125 will limit interrupts to about 8,000 interrupts per second per queue:

```
# ethtool -C <ethX> rx-usecs 125
```

For reduced latency:

- Disable ITR by setting rx-usecs to 0 using ethtool:

```
# ethtool -C <ethX> rx-usecs 0
```

6.5.5 ixgbev Linux Driver for Intel Ethernet 10 Gigabit Server Adapters

The ixgbev driver supports X550 virtual function devices that can only be activated on kernels supporting SR-IOV. SR-IOV requires the correct platform and OS support. The ixgbev driver should be loaded on both the host and VMs.

The ixgbev driver requires the ixgbe driver. The ixgbev driver supports virtual functions generated by the ixgbe driver with a `max_vfs` value of 1 or greater. For more information on the `max_vfs` parameter, see "Additional Configurations - ixgbe" on page 153.

The guest OS loading the ixgbev driver must support MSI-X interrupts.

To enable SR-IOV on your system:

1. Ensure both Virtualization and SR-IOV are enabled in the BIOS.
2. Install the Linux operating system. You can verify that the KVM driver is loaded by typing: `lsmod | grep -i kvm`
3. Load the Linux Base Driver using the `modprobe` command: `modprobe ixgbe option max_vfs=xx,yy`

`xx` and `yy` are the number of virtual functions you want to create. You must specify a number for each port with each parameter separated by a comma. For example, `xx` is the number of virtual functions for port 1; and `yy`, for port 2. You can create up to 63 functions per port.

4. Compile and install the ixgbev driver for SR-IOV. This is loaded against the virtual functions created.



NOTE:

- For VLANs, there is a limit of a total of 32 shared VLANs to 1 or more virtual functions.
- The MTU size set on a VF should match the MTU size set on the PF. A mismatch in MTU sizes may cause unexpected results.

6.5.5.1 Command Line Parameters - ixgbev

If the driver is built as a module, the following optional parameters are used by entering them on the command line with the `modprobe` command using this syntax:



```
# modprobe ixgbev [<option>=<VAL1>,<VAL2>,...]
```

For example:

```
# modprobe ixgbev InterruptThrottleRate=16000,16000
```

The default value for each parameter is generally the recommended setting, unless otherwise noted.

The following table contains parameters and possible values for `modprobe` commands:

Parameter Name	Valid Range/Settings	Default	Description
InterruptThrottleRate	0, 1, 956 - 488,281	8000	<p>0=off</p> <p>1=dynamic</p> <p><min_ITR>-<max_ITR></p> <p>Use ethtool to control InterruptThrottleRate, as shown below:</p> <pre># ethtool -C <ethX> rx-usecs N</pre> <p>where N is the time in microseconds between each interrupt.</p> <p>Interrupt Throttle Rate controls the number of interrupts each interrupt vector can generate per second. Increasing ITR lowers latency at the cost of increased CPU utilization, though it may help throughput in some circumstances.</p> <ul style="list-style-type: none"> 0 = Setting InterruptThrottleRate to 0 turns off any interrupt moderation and may improve small packet latency. However, this is generally not suitable for bulk throughput traffic due to the increased CPU utilization of the higher interrupt rate. 1 = Setting InterruptThrottleRate to Dynamic mode attempts to moderate interrupts per vector while maintaining very low latency. This can sometimes cause extra CPU utilization. If planning on deploying this driver in a latency sensitive environment, this parameter should be considered. <min_ITR>-<max_ITR> = Setting InterruptThrottleRate to a value greater or equal to <min_ITR> will program the adapter to send at most that many interrupts per second, even if more packets have come in. This reduces interrupt load on the system and can lower CPU utilization under heavy load, but will increase latency as packets are not processed as quickly. <p> On 82599 and X550-based adapters, disabling InterruptThrottleRate will also result in the driver disabling HW RSC.</p> <p> On 82598-based adapters, disabling InterruptThrottleRate will also result in disabling LRO (Large Receive Offloads).</p>

**NOTES:**

- For more information about the InterruptThrottleRate parameter, see the application note at <http://www.intel.com/design/network/applnots/ap450.htm>.
- A descriptor describes a data buffer and attributes related to the data buffer. This information is accessed by the hardware.

6.5.5.2 Additional Configurations - ixgbev



NOTE: Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

MACVLAN

This driver supports MACVLAN. Kernel support for MACVLAN can be tested by checking if the MACVLAN driver is loaded. You can run 'lsmod | grep macvlan' to see if the MACVLAN driver is loaded or run 'modprobe macvlan' to try to load the MACVLAN driver.

NOTE:

- In passthru mode, you can only set up one MACVLAN device. It will inherit the MAC address of the underlying PF (Physical Function) device.

6.5.6 igb Linux Driver for Intel Ethernet Gigabit Adapters

The igb driver supports the Intel Ethernet 300 Series of devices.

6.5.6.1 Command Line Parameters - igb

If the driver is built as a module, the following optional parameters are used by entering them on the command line with the modprobe command using this syntax:



```
# modprobe igb [<option>=<VAL1>,<VAL2>,...]
```


A value (<VAL#>) must be assigned to each network port in the system supported by this driver. The values are applied to each instance, in function order. For example:



```
# modprobe igb InterruptThrottleRate=16000,16000
```


In this case, there are two network ports supported by igb in the system. The default value for each parameter is generally the recommended setting, unless otherwise noted.

The following table contains parameters and possible values for modprobe commands:


Parameter Name	Valid Range/Settings	Default	Description
InterruptThrottleRate	0, 1, 3, 100-100000	3	<p>0=off</p> <p>1=dynamic</p> <p>3=dynamic conservative</p> <p><min_ITR>-<max_ITR></p> <p>Interrupt Throttle Rate controls the number of interrupts each interrupt vector can generate per second. Increasing ITR lowers latency at the cost of increased CPU utilization, though it may help throughput in some circumstances.</p> <ul style="list-style-type: none"> • 0 = Setting InterruptThrottleRate to 0 turns off any interrupt moderation and may improve small packet latency. However, this is generally not suitable for bulk throughput traffic due to the increased CPU utilization of the higher interrupt rate. • 1 = Setting InterruptThrottleRate to Dynamic mode attempts to moderate interrupts per vector while maintaining very low latency. This can sometimes cause extra CPU utilization. If planning on deploying this driver in a latency sensitive environment, this parameter should be considered. • <min_ITR>-<max_ITR> = Setting InterruptThrottleRate to a value greater or equal to <min_ITR> will program the adapter to send at most that many interrupts per second, even if more packets have come in. This reduces interrupt load on the system and can lower CPU utilization under heavy load, but will increase latency as packets are not processed as quickly. <p> NOTE: Unsupported Adapters: InterruptThrottleRate is NOT supported by 82542, 82543, or 82544-based adapters.</p>
LLI			<p>Low Latency Interrupts (LLI) allow for immediate generation of an interrupt upon processing receive packets that match certain criteria as set by the parameters described below. LLI parameters are not enabled when Legacy interrupts are used. You must be using MSI or MSI-X (see cat /proc/interrupts) to successfully use LLI.</p>
LLIPort	0-65535	0 (disabled)	<p>LLI is configured with the LLIPort command line parameter, which specifies which TCP port should generate Low Latency Interrupts.</p> <p>For example, using LLIPort=80 would cause the board to generate an immediate interrupt upon receipt of any packet sent to TCP port 80 on the local machine.</p> <p> WARNING: Enabling LLI can result in an excessive number of interrupts/second that may cause problems with the system and in some cases may cause a kernel panic.</p>

Parameter Name	Valid Range/Settings	Default	Description																									
LLIPush	0-1	0 (disabled)	<p>LLIPush can be set to enabled or disabled (default). It is most effective in an environment with many small transactions.</p> <p> NOTE: Enabling LLIPush may allow a denial of service attack.</p>																									
LLISize	0-1500	0 (disabled)	LLISize causes an immediate interrupt if the board receives a packet smaller than the specified size.																									
IntMode	0-2	2	<p>Interrupt mode controls the allowed load time control over the type of interrupt registered for by the driver. MSI-X is required for multiple queue support. Some kernels and combinations of kernel .config options will force a lower level of interrupt support. 'cat/proc/interrupts' will show different values for each type of interrupt.</p> <p>0 = Legacy Interrupts 1 = MSI Interrupts 2 = MSI-X interrupts</p>																									
RSS	0-8	1	<p>0 = Assign up to the lesser value of the number of CPUs or the number of queues</p> <p>X = Assign X queues, where X is less than or equal to the maximum number of queues</p> <p>The maximum number of queues allowed are:</p> <ul style="list-style-type: none"> • I350-based adapters: 8 queues • 82575-based adapters: 4 queues • 82576-based and newer adapters: 8 queues • I210-based adapters: 4 queues • I211-based adapters: 2 queues <p>This parameter is also affected by the VMDq parameter in that it will limit the queues more.</p> <table border="1" data-bbox="800 1331 1414 1598"> <thead> <tr> <th></th> <th colspan="4">VMDQ</th> </tr> <tr> <th>Model</th> <th>0</th> <th>1</th> <th>2</th> <th>3+</th> </tr> </thead> <tbody> <tr> <td>82575</td> <td>4</td> <td>4</td> <td>3</td> <td>1</td> </tr> <tr> <td>82576</td> <td>8</td> <td>2</td> <td>2</td> <td>2</td> </tr> <tr> <td>82580</td> <td>8</td> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table>		VMDQ				Model	0	1	2	3+	82575	4	4	3	1	82576	8	2	2	2	82580	8	1	1	1
	VMDQ																											
Model	0	1	2	3+																								
82575	4	4	3	1																								
82576	8	2	2	2																								
82580	8	1	1	1																								
VMDQ	0-4 on 82575-based adapters 0-8 for 82576/82580-based adapters	0	<p>Supports enabling VMDq pools, which is needed to support SR-IOV.</p> <p>0 = Disabled 1 = Sets the netdev as pool 0 2+ = Add additional queues but they currently are not used</p>																									


Parameter Name	Valid Range/Settings	Default	Description
			<p>This parameter is forced to 1 or more if the max_vfs module parameter is used. In addition, the number of queues available for RSS is limited if this is set to 1 or greater.</p> <p> NOTE: When either SR-IOV mode or VMDq mode is enabled, hardware VLAN filtering and VLAN tag stripping/insertion will remain enabled.</p>
max_vfs	0-7	0	<p>This parameter adds support for SR-IOV. It causes the driver to spawn up to max_vfs worth of virtual functions.</p> <p>If the value is greater than 0 it will also force the VMDq parameter to be 1 or more.</p> <p>The parameters for the driver are referenced by position. Thus, if you have a dual port adapter, or more than one adapter in your system, and want N virtual functions per port, you must specify a number for each port with each parameter separated by a comma. For example:</p> <pre># modprobe igb max_vfs=4</pre> <p>This will spawn 4 VFs on the first port.</p> <pre># modprobe igb max_vfs=2,4</pre> <p>This will spawn 2 VFs on the first port and 4 VFs on the second port.</p> <p> NOTES:</p> <ul style="list-style-type: none"> • Caution must be used in loading the driver with these parameters. Depending on your system configuration, number of slots, etc., it is impossible to predict in all cases where the positions would be on the command line. • Neither the device nor the driver control how VFs are mapped into config space. Bus layout will vary by operating system. On operating systems that support it, you can check sysfs to find the mapping. <p>When either SR-IOV mode or VMDq mode is enabled, hardware VLAN filtering and VLAN tag stripping/insertion will remain enabled. Please remove the old VLAN filter before the new VLAN filter is added. For example:</p> <pre># ip link set eth0 vf 0 vlan 100 // set vlan 100 for VF 0</pre> <pre># ip link set eth0 vf 0 vlan 0 // Delete vlan 100</pre> <pre># ip link set eth0 vf 0 vlan 200 // set a new vlan 200 for VF 0</pre>
QueuePairs	0-1	1	<p>If set to 0, when MSI-X is enabled, the Tx and Rx will attempt to occupy separate vectors.</p> <p>This option can be overridden to 1 if there are not sufficient interrupts available. This can occur if any combination of RSS, VMDQ, and max_vfs results in more than 4 queues being used.</p>

Parameter Name	Valid Range/Settings	Default	Description
Node	0-n, -1	-1 (off)	<p>0-n: where n is the number of the NUMA node that should be used to allocate memory for this adapter port.</p> <p>-1: uses the driver default of allocating memory on whichever processor is running modprobe.</p> <p>The Node parameter allows you to choose which NUMA node you want to have the adapter allocate memory from. All driver structures, in-memory queues, and receive buffers will be allocated on the node specified. This parameter is only useful when interrupt affinity is specified; otherwise, part of the interrupt time could run on a different core than where the memory is allocated causing slower memory access and impacting throughput, CPU, or both.</p>
EEE	0-1	1 (enabled)	<p>0 = Disables EEE</p> <p>1 = Enables EEE</p> <p>A link between two EEE-compliant devices will result in periodic bursts of data followed by periods where the link is in an idle state. This Low Power Idle (LPI) state is supported at 1 Gbps and 100 Mbps link speeds.</p> <p> NOTES:</p> <ul style="list-style-type: none"> • EEE support requires auto-negotiation. • Both link partners must support EEE. • EEE is not supported on all Intel® Ethernet Network devices or at all link speeds.
DMAC	0, 250, 500, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000, 9000, 10000	0 (disabled)	<p>This parameter enables or disables DMA Coalescing feature. Values are in microseconds and set the internal DMA Coalescing internal timer.</p> <p>DMA (Direct Memory Access) allows the network device to move packet data directly to the system's memory, reducing CPU utilization. However, the frequency and random intervals at which packets arrive do not allow the system to enter a lower power state. DMA Coalescing allows the adapter to collect packets before it initiates a DMA event. This may increase network latency but also increases the chances that the system will enter a lower power state.</p> <p>Turning on DMA Coalescing may save energy with kernel 2.6.32 and later. DMA Coalescing must be enabled across all active ports in order to save platform power.</p>
MDD	0-1	1 (enabled)	<p>0 = Disabled</p> <p>1 = Enabled</p> <p>This parameter is only relevant for devices operating in SR-IOV mode. When this parameter is set, the driver detects malicious VF driver and disables its Tx/Rx queues until a VF driver reset occurs.</p>

6.5.6.2 Additional Configurations - igb

 **NOTE:** Refer to "Additional Configurations (All Drivers)" on page 79 for information common to multiple drivers.

Jumbo Frames

-  **NOTES:**
- The maximum MTU setting for Jumbo Frames is 9216 bytes. This value coincides with the maximum Jumbo Frames size of 9234 bytes.
 - Using Jumbo Frames at 10 or 100 Mbps may result in poor performance or loss of link.
 - Packet loss may have a greater impact on throughput when you use jumbo frames. If you observe a drop in performance after enabling jumbo frames, enabling flow control may mitigate the issue.


Speed and Duplex Configuration

In addressing speed and duplex configuration issues, you need to distinguish between copper-based adapters and fiber-based adapters.

In the default mode, an Intel® Ethernet Network Adapter using copper connections will attempt to auto-negotiate with its link partner to determine the best setting. If the adapter cannot establish link with the link partner using auto-negotiation, you may need to manually configure the adapter and link partner to identical settings to establish link and pass packets. This should only be needed when attempting to link with an older switch that does not support auto-negotiation or one that has been forced to a specific speed or duplex mode. Your link partner must match the setting you choose. 1 Gbps speeds and higher cannot be forced. Use the autonegotiation advertising setting to manually set devices for 1 Gbps and higher.


Speed, duplex, and autonegotiation advertising are configured through the `ethtool` utility. To see the speed configurations your device supports, run the following:

```
# ethtool <ethX>
```

 **CAUTION:** Only experienced network administrators should force speed and duplex or change autonegotiation advertising manually. The settings at the switch must always match the adapter settings. Adapter performance may suffer or your adapter may not operate if you configure the adapter differently from your switch.

An Intel® Ethernet Network Adapter using fiber-based connections will not attempt to auto-negotiate with its link partner since those adapters operate only in full duplex and only at their native speed.

Wake on LAN (WoL) Support

 **NOTES:** Wake on LAN is only supported on port A of multi-port devices.

Multiqueue

In this mode, a separate MSI-X vector is allocated for each queue and one for "other" interrupts such as link status change and errors. All interrupts are throttled via interrupt moderation. Interrupt moderation must be used to avoid interrupt storms while the driver is processing one interrupt. The moderation value

should be at least as large as the expected time for the driver to process an interrupt. Multiqueue is off by default.

REQUIREMENTS: MSI-X support is required for Multiqueue. If MSI-X is not found, the system will fallback to MSI or to Legacy interrupts. This driver supports multiqueue in kernel versions 2.6.24 and newer. This driver supports receive multiqueue on all kernels that support MSI-X.

NOTES:

- Do not use MSI-X with the 2.6.19 or 2.6.20 kernels.
- On some kernels a reboot is required to switch between single queue mode and multiqueue mode or vice-versa.

Large Receive Offload (LRO)


Large Receive Offload (LRO) is a technique for increasing inbound throughput of high-bandwidth network connections by reducing CPU overhead. It works by aggregating multiple incoming packets from a single stream into a larger buffer before they are passed higher up the networking stack, thus reducing the number of packets that have to be processed. LRO combines multiple Ethernet frames into a single receive in the stack, thereby potentially decreasing CPU utilization for receives.

IGB_NO_LRO is a compile time flag. The user can enable it at compile time to add support for LRO from the driver. The flag is used by adding `CFLAGS_EXTRA="-DIGB_NO_LRO"` to the make file when it's being compiled.


```
# make CFLAGS_EXTRA="-DIGB_NO_LRO" install
```

You can verify that the driver is using LRO by looking at these counters in ethtool:

- `Iro_aggregated` - counts total packets that were combined
- `Iro_flushed` - counts the number of packets flushed out of LRO
- `Iro_recycled` - counts the number of buffers returned to the ring from recycling


 **NOTE:** IPv6 and UDP are not supported by LRO.

IEEE 1588 Precision Time Protocol (PTP) Hardware Clock (PHC)

 **NOTE:** PTP requires a 3.0.0 or later kernel version with PTP support enabled in the kernel and a user-space software daemon.

IGB_PTP is a compile time flag. The user can enable it at compile time to add support for PTP from the driver. The flag is used by editing the make file as follows when it is being compiled:

```
# make CFLAGS_EXTRA="-DIGB_PTP" install
```

 **NOTE:** The driver will fail to compile if your kernel does not support PTP.

You can verify that the driver is using PTP by looking at the system log to see whether a PHC was attempted to be registered or not. If you have a kernel and version of ethtool with PTP support, you can check the PTP support in the driver by executing:

```
# ethtool -T <ethX>
```

MAC and VLAN Anti-Spoofing Feature for VFs

When a malicious driver on a Virtual Function (VF) interface attempts to send a spoofed packet, it is dropped by the hardware and not transmitted.

An interrupt is sent to the PF driver notifying it of the spoof attempt. When a spoofed packet is detected, the PF driver will send the following message to the system log (displayed by the "dmesg" command):

```
Spoof event(s) detected on VF(n)
```

Where n=the VF that attempted to do the spoofing.

Setting MAC Address, VLAN and Rate Limit Using IProute2 Tool

You can set a MAC address of a Virtual Function (VF), a default VLAN and the rate limit using the IProute2 tool. Download the latest version of the iproute2 tool from Sourceforge if your version does not have all the features you require.

7. VMware* ESXi* Drivers and Support

7.1 Driver types

Intel provides the following types of drivers for VMware ESX:

- Native mode drivers are the default driver for the VMware ESX environment. They are interrupt driven and developed using VMware's native mode API.
- Enhanced Network Stack (ENS) drivers are intended for use in VMware NSX-T deployments. These drivers are polling mode drivers.
- Unified drivers support both interrupt and poll mode operation. Depending on the deployment model, the driver automatically utilizes the appropriate mode.

This release contains Native, ENS, and Unified drivers as follows:

Driver	Device Family	ESXi 8.0 U1	ESXi 7.0 U3
icen	Intel® Ethernet 800 Series	Unified	Unified
i40en	Intel® Ethernet 700 Series	Unified	Unified
ixgben	Intel® Ethernet X550 Series	Native, ENS	Native, ENS
igbn	Intel® Gigabit I350 Series	Native	Native

You can check your hardware compatibility at <http://www.vmware.com/resources/compatibility/search.php>

7.2 Installation and Configuration

For detailed information on installation, driver fixes, known issues, and other enhancements, please refer to the readme and release notes files included with the driver.

For information on VMware ENS configuration, please refer to VMware documentation under the topics of Enhanced Network Stack (ENS) or Enhanced Data Path. To use an Intel ENS capable driver, you must install VMware NSX-T on the system hosting the adapters. VMware's NSX manager must then be used to create logical switches and attach ENS capable drivers to the adapter ports.



NOTE: To install the ENS driver, you must first install the native driver. You must leave the native driver installed for the ENS driver to function.

8. Remote Boot

Remote Boot allows you to boot a system using only an Ethernet adapter. You connect to a server that contains an operating system image and use that to boot your local system.

8.1 Intel® Boot Agent

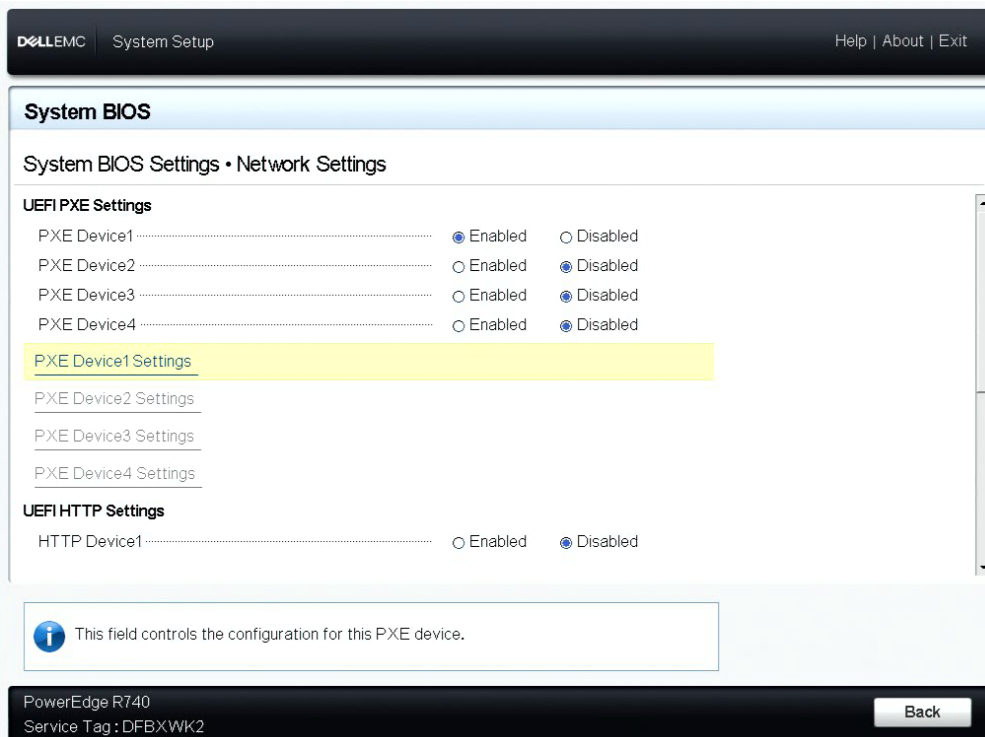
The Intel® Boot Agent is a software product that allows your networked client computer to boot using a program code image supplied by a remote server. Intel Boot Agent complies with the Pre-boot eXecution Environment (PXE) Version 2.1 Specification. It is compatible with legacy boot agent environments that use BOOTP protocol.

For all devices supported in this release, PXE is enabled through the UEFI environment (HII interface). **We recommend you use the default UEFI boot mode.**

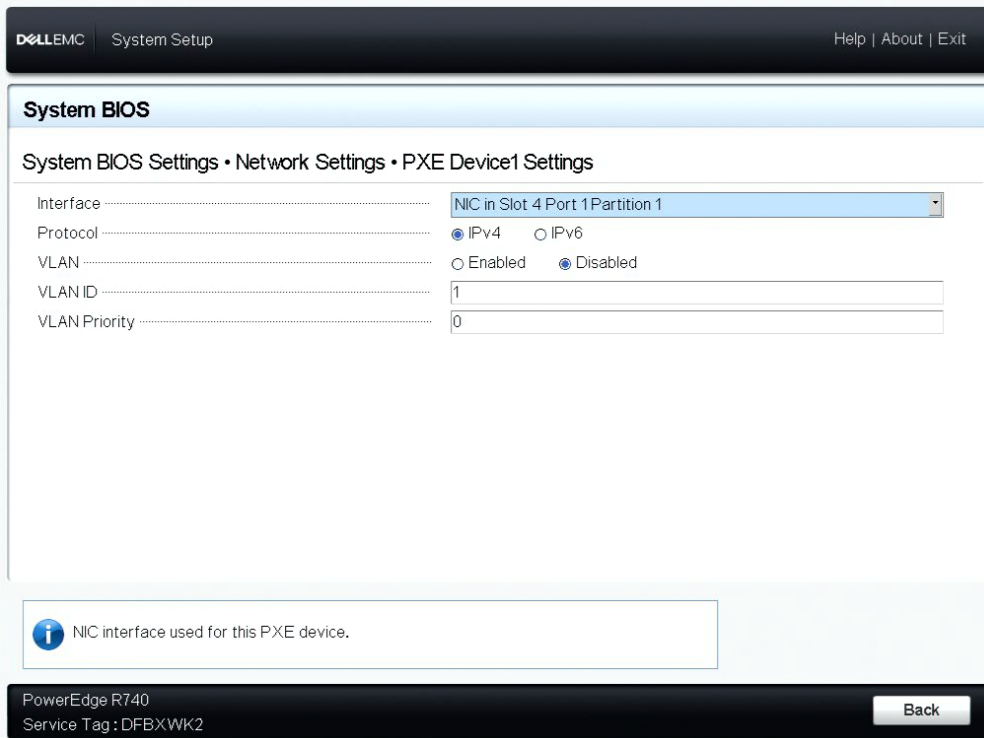
8.1.1 Configuring UEFI PXE Settings

To enable PXE boot in UEFI boot mode:

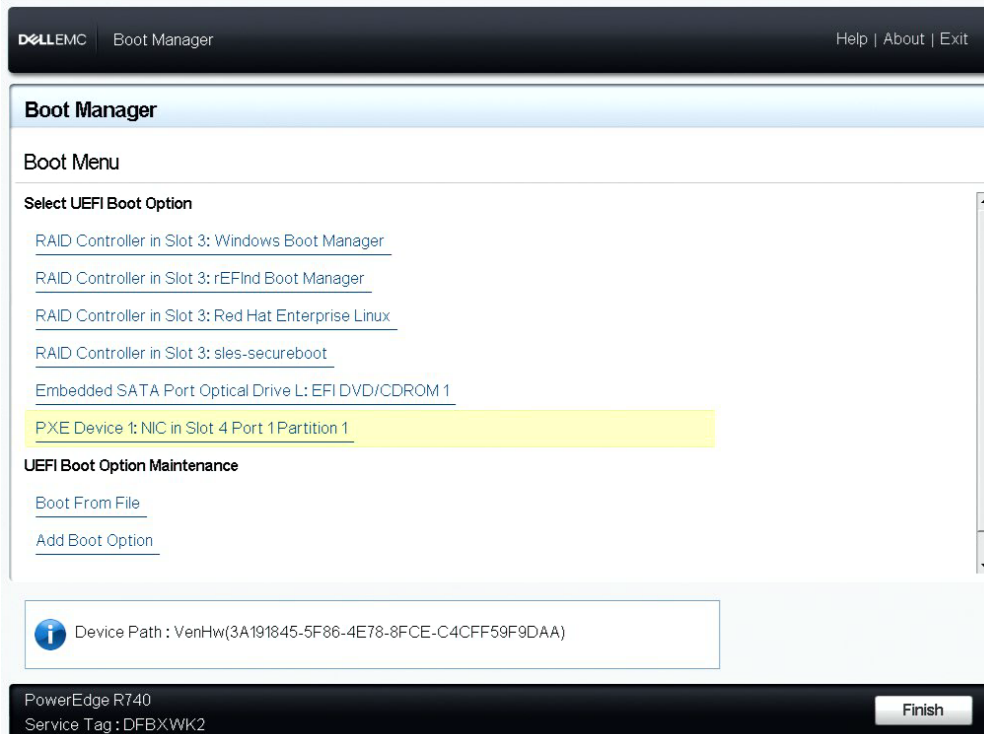
1. Open the System BIOS.
2. Under Network Settings, enable PXE for your desired device.



3. Configure the PXE settings for your device. For example, for PXE Device1 Settings:



After you enable the UEFI PXE options in the BIOS, the PXE device appears in the Boot Menu :



8.1.2 Intel Boot Agent Client Configuration

The Intel Boot Agent is enabled and configured from HII.



CAUTION: If spanning tree protocol is enabled on a switch port through which a port is trying to use PXE, the delay before the port starts forwarding can cause a DHCP timeout. Either disable spanning tree or turn on the feature that allows the port to begin forwarding of packets immediately, rather than wait until the spanning tree discovery is complete.

8.1.3 Intel Boot Agent Target/Server Setup

For the Intel Boot Agent software to perform its intended job, there must be a server set up on the same network as the client computer. That server must recognize and respond to the PXE or BOOTP boot protocols that are used by the Intel Boot Agent software.

8.1.3.1 Linux Server Setup

Consult your Linux vendor for information about setting up the Linux Server.

8.1.3.2 Windows Deployment Services

Nothing is needed beyond the standard driver files supplied on the media. Microsoft owns the process and associated instructions for Windows Deployment Services. For more information on Windows Deployment Services perform a search of Microsoft articles at: <http://technet.microsoft.com/en-us/library/default.aspx>

8.1.4 Intel Boot Agent Troubleshooting

The following list of problems and associated solutions covers a representative set of problems that you might encounter while using the Intel Boot Agent.

After booting, my computer experiences problems

After the Intel Boot Agent product has finished its sole task (remote booting), it no longer has any effect on the client computer operation. Thus, any issues that arise after the boot process is complete are most likely not related to the Intel Boot Agent product.

If you are having problems with the local (client) or network operating system, contact the operating system manufacturer for assistance. If you are having problems with some application program, contact the application manufacturer for assistance. If you are having problems with any of your computer's hardware or with the BIOS, contact your computer system manufacturer for assistance.

There are configuration/operation problems with the boot process

If your PXE client receives a DHCP address, but then fails to boot, you know the PXE client is working correctly. Check your network or PXE server configuration to troubleshoot the problem. Contact [Customer Support](#) if you need further assistance.

9. Firmware

Firmware is a layer of software that is programmed into a device's memory. It provides low level functionality for the device. In most cases you will not notice the firmware on your device at all. Firmware error states usually occur because of an unsuccessful update.

9.1 Firmware Security

Intel or your equipment manufacturer will occasionally release a firmware security patch. We recommend that you update your firmware to the latest version available for your device to take advantage of these security patches. Firmware updates for Intel Ethernet devices will have a Security Revision number (SRev).

9.1.1 Minimum Security Revision Enforcement

Firmware security updates can be undone if you install a previous version of the firmware onto your device. Intel firmware releases include a Minimum Security Revision (MinSRev) enforcement feature. This means you can block someone from installing a lower revision of the firmware onto your device. This will limit the rollback capabilities of your device. The firmware update process will block the update if the supplied firmware has a lower security revision (SRev) than the MinSRev value of the firmware currently loaded on the device. Only update the MinSRev value if you are certain you will not need to roll the firmware back to an earlier version.

You can update the MinSRev value during the firmware update process, locking the current security version in as the new MinSRev baseline, by using the `-optinminsrev` command line option.



CAUTION: The MinSRev value on a device can never be decreased. Once the MinSRev is increased, NVM downgrades attempting to install a lower Security revision (SRev) than the current MinSRev will be rejected by the device. Users who want to downgrade firmware without regard to security revisions should not use this feature.

This applies to devices based on the following:


- Intel® Ethernet 800 Series
- Intel® Ethernet 700 Series
- Intel® Ethernet Controller X550

9.1.2 SRev and MinSRev Examples

9.1.2.1 View your device's current Minimum Security Revision

Use these steps to view the Minimum Security Revision that is set on the card.

1. During system boot, press the F2 key to enter the **System Setup** menu.
2. Under **System Setup Main Menu**, select **Device Settings**.
3. Select your adapter from the list to get to the **Device Configuration Menu**.
4. Under **Main Configuration Page**, select **Firmware Image Properties**.
5. View the **Minimum Security Revision** attribute.

 **NOTE:** The minimum security revision shown in HII and in the NVM Update tool may not match. HII displays the value in decimal while NVM Update tool displays it in hex.

9.1.2.2 Update your device's MinSRev


To update your device's MinSRev, you must first extract nvupdate from the FW DUP. In Windows, this can be done by clicking "Extract" after starting the firmware DUP in GUI mode. In Linux, this can be done by using the "--extract <path>" parameters from the command line. After you extract the package, perform the update from the command line as shown in the following example:

Windows: `nvupdatew64e -u -optinminsrev -l update.log -o results.xml -c nvupdate.cfg`

Linux: `nvupdate64e -u -optinminsrev -l update.log -o results.xml -c nvupdate.cfg`

Where:

- u -- Sets nvupdate to update mode.
- optinminsrev -- Tells the tool to update the MinSRev value.
- l update.log -- Specifies the name of the log file.
- o results.xml -- Specifies the name of the results file. This is an XML file that contains the inventory/update results.
- c nvupdate.cfg -- Specifies the name of the configuration file. This is a text file that contains descriptions of networking devices and firmware versions for those devices.


 **NOTE:** The minimum security revision shown in HII and in the NVM Update tool may not match. HII displays the value in decimal while NVM Update tool displays it in hex.

9.2 Updating the Firmware via PLDM

Intel Ethernet 810 Series devices support updating the firmware using the Platform Level Data Model (PLDM) interface to the firmware.

To update the firmware via PLDM:

1. Go to the iDRAC web interface.
2. In iDRAC, load the Dell Update Package (DUP) with the desired firmware.
3. In iDRAC, click the **Install** button.
4. When the installation is complete, the job under the iDRAC Job Queue and related entry in the Lifecycle Log will report that the firmware update was successful.

 **NOTE:** PCIe Ethernet adapters based on the Intel Ethernet E810 Series support iDRAC rebootless updates. This feature allows firmware updates on the adapters without rebooting the host server to initiate and perform the update. However, the adapters may require a power cycle for the firmware to take effect. See the Lifecycle Controller or job queues to check if a reboot is needed.

9.3 System Lockdown Mode

Some Intel® Ethernet devices support system lockdown, where the management controller blocks certain requests to modify the device firmware. If system lockdown is enabled, no updates to firmware are allowed from any interface.

9.4 Firmware Rollback Mode

When a port is in firmware rollback mode it may have reduced functionality. Usually a device enters firmware rollback mode when a firmware update does not complete correctly. Rebooting or power cycling the system may allow the port to use the previous firmware. You may need to reapply the firmware update to regain full functionality on the device. Use the appropriate NVM Update Package to update the device's firmware. Download the latest NVM Update Package from your vendor's support website and follow the instructions in it. After restoring the NVM image, you may need to perform an A/C power cycle of the system.

9.5 Firmware Recovery Mode

A device will enter Firmware Recovery mode if it detects a problem that requires the firmware to be reprogrammed. When a device is in Firmware Recovery mode it will not pass traffic or allow any configuration; you can only attempt to recover the device's firmware.

9.5.1 Affected Products

Ethernet Device	New NVM Version	Software Driver and Tools
Intel® Ethernet 800 Series	All firmware versions	All driver versions
Intel® Ethernet 700 Series	Intel® NIC Family Version 18.8.0 Firmware and newer	Intel® NIC Family Version 18.8.0 and newer
Intel® Ethernet Controller X550	Intel® NIC Family Version 18.8.0 Firmware and newer	Intel® NIC Family Version 18.8.0 and newer

9.5.2 Recovery Mode Detection

During initialization, a device can enter recovery mode if the device firmware detects a problem with the LAN device, mandating NVM reprogramming to restore normal operation. After thorough internal testing of the NVM (typically less than 10 minutes, but in some cases longer), the NIC enters Recovery Mode.

9.5.3 Firmware Recovery Mode Errors and Messages

When a device is in Firmware Recovery mode, the device drivers, preboot software, and utilities may log or display messages such as the following:

- **Firmware recovery mode detected. Limiting functionality. Refer to the Intel® Ethernet Adapters and Devices User Guide for details on firmware recovery mode.**
- **Firmware recovery mode detected. The underlying hardware has been deactivated. Refer to the Intel® Ethernet Adapters and Devices User Guide for details on firmware recovery mode.**
- **Firmware recovery mode detected. Initialization failed.**
- **Firmware recovery mode detected. Limiting functionality.**

- **Initialization failure due to repeated FW resets.** This message is usually an indication that the device is about to enter Recovery Mode. The device may be able to return to normal functionality without intervention. This may take several minutes. No action is required unless the device does enter Recovery Mode.

9.5.4 Resolving Firmware Recovery Mode Issues

If your device is in Firmware Recovery mode, you can restore it to factory defaults using the process for resolution of Firmware Recovery Mode Issues as outlined in the sub-sections below.

9.5.4.1 NVM Self Check

The process begins after power-on or reboot. At this time, the firmware will perform tests to assess whether there is damage or corruption of the device NVM image.

Actions:

- If NVM image damage or corruption **is not** detected, the device will initialize and operate normally. No further action is required.
- If NVM image damage or corruption **is** detected, the device will not initialize. Proceed with the additional recovery steps listed under Recovery Mode below.

9.5.4.2 Recovery Mode

The device NVM image has exhibited damage or corruption.

Actions:

1. Wait 10 minutes for the NVM self-check process to complete. If during this period normal operation is achieved, the device will initialize and operate normally. No further action is required.
2. If after 10 minutes normal operation is *not* achieved:
 - a. Check the System Event log for Windows OSs or driver message and kernel logs for Linux and ESXi based distributions. Recovery Mode is confirmed by presence of message/log entries as listed in the Firmware Recovery Mode Errors and Messages section above.
 - b. Reboot the system and proceed with the additional recovery steps listed under NVM Image Restoration below.

 **NOTES:**

- While in Recovery Mode, for Windows OSs, clicking on the adapter in device manager may present a dialog box indicating that Firmware Recovery Mode is active.
 - Once the dialog is dismissed, while the device appears to be functioning normally, it is in fact limited to only enable NVM image recovery.
- If the system is rebooted (versus power cycled), the driver status may not show a Code 10/yellow bang status as expected. Refer to events logged in System Event log for Windows OSs or driver message and kernel logs for Linux and ESXi based distributions to accurately assess the adapter status.
- When the adapter is in recovery mode, the link LED will not be lit and the adapter will not appear in the following locations:
 - F2 System Setup > Device Settings
 - System BIOS as a NIC for PXE Boot in UEFI boot mode
 - Lifecycle Controller > Network Settings
 - iDRAC Web GUI > Firmware Inventory

9.5.4.3 NVM Image Restoration

At this point, the device is in Firmware Recovery mode and its functionality is limited to only supporting restoration of the NVM image.

Actions:

1. Before initiating device recovery, the integrity of the host operating system, device drivers and firmware utilities must be verified and reinstalled if necessary. Fully functional operating system, device drivers and tools are required for device recovery. Please consult your operating system specific instructions on how to scan and repair potentially damaged system files.
2. If your device is in Firmware Recovery mode, you can restore it to factory defaults using the latest Dell Update Package for Intel Adapter Firmware (FW-DUP). Download the latest Dell Update Package for Intel Adapter Firmware (FW-DUP) from Dell's support website and follow the instructions in them. The Dell Update Package for Intel Adapter Firmware (FW-DUP) must be executed in an operating system to recover the device, not in the Dell Lifecycle Controller or iDRAC.
3. After restoring the NVM image, perform an A/C power cycle of the system. Details for this are in the **Other General Notes** section below.

 **NOTES:**

- If a device is in recovery mode when the Dell DUP package is executed for inventory, the Firmware Family Version (FFV) will display "0.0.0". This is expected behavior.
- Running the FW-DUP in recovery mode does not update the option ROM. A/C power cycling and running the FW-DUP a second time will correct this.
- After running the FW-DUP in recovery mode, the firmware version is incorrect. Updating the firmware via the Dell Lifecycle Controller or iDRAC resolves the issue.
- User configured settings (i.e. iSCSI target information, user defined port/alternate MAC addresses) will not be restored to pre-recovery mode values.

9.5.4.4 Other General Notes

NOTES:

- To perform an AC power cycle, do the following:
 - Shut down the system if its is powered up.
 - Unplug all AC power cords from the system.
 - Leave the AC power cords unplugged for 15 seconds to allow the system power supply to discharge completely.
 - Plug in AC power cords to the system.

9.6 Firmware Logging

Intel® Ethernet 800 Series devices allow you to generate firmware logs for supported categories of events, to help debug complex issues with [Customer Support](#). Firmware logging is enabled by default.


NOTE:

- Both the device and the driver need to support firmware logging for the functionality to work. If you are not able to set the configuration and the problem persists, reinstall the driver.
- You must have the latest base driver and firmware installed.
- Firmware logging events and levels are device-wide settings. Changing the log configuration on one port will apply it to all ports on a device.

9.6.1 Capturing a Firmware Log

To capture a firmware log, you must do the following:

1. Set the configuration for the firmware log. See the following for more information :
 - "Configuring Settings for Firmware Logs" below
 - "Tips for Firmware Logs" on page 180
 - "OS-Specific Information" on page 180
2. Perform the necessary steps to generate the issue you're trying to debug.
3. Capture the firmware log. (Exact steps will vary by operating system.)
4. Stop capturing the firmware log.
5. Reset your firmware log settings as needed.
6. Work with Customer Support to debug your issue.

 **NOTE:** Firmware logs are generated in a binary format and must be decoded by Customer Support. Information collected is related only to firmware and hardware for debug purposes.

9.6.2 Configuring Settings for Firmware Logs

Firmware logs capture information about different categories of events (called "modules"). A module corresponds to a general category of functionality, such as link topology detection or manageability.

The device's NVM sets default verbosity levels for each module. You can change the verbosity level per module; refer to "OS-Specific Information" on the next page for more details. You can set only one log level per module, and each level includes the verbosity levels lower than it. Available verbosity levels are:

- 0 = none
- 1 = error
- 2 = warning
- 3 = normal
- 4 = verbose

If you see errors or suspect the issue could fall into the below categories, setting the firmware logs to capture more verbosity for the corresponding module(s) in the right column will provide more information in the firmware log.

Category of Event	Corresponding Module
Initialization	Control
NVM	NVM Authentication VPD
I/O	I2C SDP MDIO
Link Management	Link Management Link Control Technology Link Topology Detection
Rx	Parser Switch ACL Post
Tx	Scheduler Tx Queue Management
AQ Interface	Admin Queue HDMA
Manageability	Manageability
Protocols	LLDP DCBx
Infrastructure	Watchdog Task Dispatcher General

Category of Event	Corresponding Module
	IOSF PF Registration Module Versions
XLR	XLR
QoS	DCB
Diagnostics	SyncE Health
TimeSync	Time Sync

9.6.3 Tips for Firmware Logs

- Firmware logs are for the hardest issues to debug. If you are experiencing issues, refer to the following sections for preliminary methods to diagnose problems:
 - "Firmware" on page 173
 - "Troubleshooting" on page 185
 - "Health Status Messages" on page 187
- We generally do not recommend to capture firmware logs at all times. If you suspect an issue, set the module(s) for the suspected event to a higher verbosity level, capture the firmware log, and then stop the log.
- Collecting firmware logs should not materially impact performance or CPU utilization.
- In general, set the logging level to Verbose only for the configuration group(s) or module(s) you need to debug. Setting too many modules to Verbose can overrun the buffer.
- You can try writing logs to a remote location or an external storage device, if your disk is full or your system does not have sufficient storage.

9.6.4 OS-Specific Information

Linux


In Linux, firmware logs are printed to dmesg. Refer to the README in the driver tarball or the [Linux section](#) of this guide for more information on configuring firmware logs.


Windows

In Windows, you use PowerShell and Intel Ethernet cmdlets to configure firmware logging and capture firmware logs. At a high level, do the following to capture a firmware log in Windows:

1. Set the configuration for the firmware log, using the `Set-IntelEthernetLogConfig` cmdlet in PowerShell.
2. Perform the necessary steps to generate the issue you're trying to debug.

3. Start capturing the firmware log, using the `Start-IntelEthernetLog` cmdlet.
4. Stop capturing the firmware log, using the `Stop-IntelEthernetLog` cmdlet.
5. Work with Customer Support to decode your firmware log file and debug the issue.

 **NOTE:** Firmware logs will be captured in the file you designated with `Start-IntelEthernetLog`.

 **NOTE:** To disable firmware logging, use the `Disable-IntelEthernetLogConfig` cmdlet. To verify that firmware logging is disabled, run the `Get-IntelEthernetLogConfig` cmdlet; its results should say "Disabled."


Refer to the following for more information:

- "About Intel® Ethernet Cmdlets" on page 13
- The cmdlet help for Intel Ethernet cmdlets (see "Help Information for PowerShell Cmdlets" on page 16)
- The readme file inside the Intel Ethernet cmdlet directory on your disk (available after installing or upgrading the base drivers; see "Installing Intel Ethernet Cmdlets" on page 14 for its location)


ESXi

In ESXi, use `esxcfg-module` to set the configuration for firmware logs. Firmware logs are printed to kernel logs, with the tag `FWLOG`; use `dmesg` or read the file at `/var/log/vmkernel.log`.

At a high level, do the following to capture a firmware log in ESXi:

 **NOTE:** Refer to the table after these steps for all commands and parameters.

1. Set the configuration for the firmware log, using `esxcfg-module`. The ESXi driver uses the following module parameters for firmware logging:
 - `FWLogEnable`: Enables firmware logging functionality on the designated PF (0 = Disable, 1 = Enable). Use commas to separate the values for each PF; the first value is for PF0, second for PF1, and so on.
 - `FWLogEvents`: Designates the firmware events to log, using a bitmask. Binary math is required to set.
 - `FWLogLevel`: Sets the verbosity level for the firmware event's log.
2. Redirect the kernel log or `dmesg` to a separate file for capturing the firmware log.
3. Reboot the system for changes to take effect.
4. After the system has rebooted, perform the necessary steps to generate the issue you're trying to debug.
5. Work with Customer Support to decode your firmware log file and debug the issue.

 **NOTE:** Firmware logs will be captured in the file you designated in step 2.

Use the following commands in ESXi for tasks related to firmware logging:

Task	Example Code
Enable firmware logging and set the verbosity level for your desired events	<pre># esxcfg-module icen -s 'FWLogEnable=<values> FWLogEvents=<bitmask> FWLogLevel=<value>'</pre> <p>For example, to enable firmware logging on PF0 and set all events to log warning messages, use:</p> <pre># esxcfg-module icen -s 'FWLogEnable=1,0,0,0,0,0,0,0 FWLogEvents=255 FWLogLevel=2'</pre>
Show the current configuration of the firmware log parameters	<pre># esxcfg-module -g <driver name></pre> <p>NOTE: If firmware logging is disabled, the FWLogEnable parameter should say "0" (disabled).</p>
Show a description of module parameters for firmware logging	<pre># esxcfg-module -i <driver name></pre> <p>NOTE: Look for the parameters that begin with FWLog.</p>
Redirect the firmware log to a file	<pre># tail -f /var/log/vmkernel.log > filename.log</pre>
Disable firmware logging	<pre># esxcfg-module icen -s 'FWLogEnable=0 FWLogEvents=0 FWLogLevel=0'</pre>

9.7 Debug Dump

Intel Ethernet 810 Series devices support debug dump, which allows you to obtain runtime register values from the firmware for "clusters" of events and then write the results to a single dump file, for debugging complicated issues in the field.

This debug dump contains a snapshot of the device and its existing hardware configuration, such as switch tables, transmit scheduler tables, and other information. Debug dump captures the current state of the specified cluster(s) and is a stateless snapshot of the whole device.



NOTE:

- Like with [firmware logs](#), the contents of the debug dump are not human-readable. You must work with Customer Support to decode the file.
- Debug dump is per device, not per PF.
- Debug dump writes all information to a single file.
- This feature is not currently supported on Linux.

Exact steps will vary by OS, but do the following to generate a debug dump log file:

1. Using the method appropriate for your OS (see "OS-Specific Information" on the next page), specify one or more clusters that you want to dump the hardware configuration for.
 - **NOTE:** Firmware will return an error if you call the command without specifying at least one cluster. Available clusters include:
 - Switch
 - ACL
 - Tx Scheduler
 - Profile Configuration

- Link
 - DCB
 - L2P
2. Specify the path and filename for the dump file to be written to (optional depending on your OS).
 3. Execute the command to write the debug dump file.
 4. After the log file is written, work with Customer Support to decode the dump file.

9.7.1 OS-Specific Information

Use the following tools or commands to write the debug dump results to a dump file:

OS	Method to Generate Debug Dump File
Windows Server Azure Stack HCI	Use either of the following: <ul style="list-style-type: none"> • Write-IntelEthernetDebugDump Ethernet cmdlet in PowerShell (see the cmdlet help for more information) • Intel® Ethernet Inspector
Windows	Not supported
Linux	Not supported
ESXi	Use esxcli (see below)
FreeBSD	Use sysctl (see the FreeBSD base driver README for more information)

ESXi



NOTE: For this functionality to work, you must have installed version 1.10.x or higher of the intnet tool, which is a plugin to the esxcli tool. You can download the latest version from the Intel Download Center [here](#).

In esxcli, use the following command to generate the debug dump file for your specified cluster(s):

```
# esxcli intnet debug fw dump <Cmd options>
```

Where <Cmd options> include:

- -n|--vmnic <string>: Specifies the vmnic name to operate on. **NOTE:** This field is required.
- -c|--clusters <string>: Specifies the clusters to dump. You must specify at least one cluster.
 - To specify multiple clusters, enclose a single string in quotes, separated by commas with no spaces. For example:


```
# esxcli intnet debug fw dump -n vmnic0 --clusters "ACL,L2P"
```
- -l|--list: Displays the complete list of valid clusters on the screen.

To show the complete list of valid clusters, use the following:

```
# esxcli intnet debug fw dump -n <vmnicX> -l
```

esxcli will output the debug dump results to a single file in the `/scratch/core` directory. The file naming convention is `vmnicX-<time-stamp>-dump.bin`, where `vmnicX` is the VMware device alias of the affected device.

10. Troubleshooting


10.1 Common Problems and Solutions

There are many simple, easy-to-fix problems related to network problems. Review each one of these before going further.

- Check for recent changes to hardware, software, or the network that may have disrupted communications.
- Check the driver software.
 - Make sure you are using the latest appropriate drivers for your adapter from the [Dell support website](#).
 - Disable (or unload), then re-enable (reload) the driver or adapter.
 - Check for conflicting settings. Disable advanced settings to see if it corrects the problem.
 - Re-install the drivers.
- Check the cable. Use the best available cabling for the intended data rate.
 - Check that the cable is securely attached at both points.
 - Make sure the cable length does not exceed specifications.
 - Perform a cable test.
 - Replace the cable.
- Check the link partner (switch, hub, etc.).
 - Make sure the link partner is active and can send and receive traffic.
 - Make sure the adapter and link partner settings match one another, or are set to auto-negotiate.
 - Make sure the port is enabled.
 - Re-connect to another available port or another link partner.
- Look for adapter hardware problems.
 - Re-seat the adapter.
 - Insert the adapter in another slot.
 - Check for conflicting or incompatible hardware devices and settings.
 - Replace the adapter.
- Check the [Dell support website](#) for possible documented issues.
 - Select your adapter from the adapter family list.
 - Check the Frequently Asked questions section.
 - Check the Knowledge Base.
- Check your process monitor and other system monitors.
 - Check to see that there is sufficient processor and memory capacity to perform networking activity.
 - Look for any unusual activity (or lack of activity).
 - Use network testing programs to check for basic connectivity.

- Check your BIOS version and settings.
 - Use the latest appropriate BIOS for your computer.
 - Make sure the settings are appropriate for your computer.

The following troubleshooting table assumes that you have already reviewed the common problems and solutions.

Problem	Solution
Your computer cannot find the adapter	Make sure your adapter slots are compatible for the type of adapter you are using.
Diagnostics pass but the connection fails	<p>Make sure the cable is securely attached, is the proper type and does not exceed the recommended lengths.</p> <p>Make sure the duplex mode and speed setting on the adapter matches the setting on the switch.</p>
Adapter unable to connect to switch at correct speed. Gigabit adapter connects at 100 Mbps and 10 gigabit adapter connects at 1000 Mbps.	<p><i>This is applicable only to copper-based connections.</i></p> <p>Make sure the adapter and the link partner are set to auto-negotiate.</p> <p>Verify that you are running the latest operating system revision for your switch and that the switch is compliant with the proper IEEE standard:</p> <ul style="list-style-type: none"> • IEEE 802.3ad-compliant (gigabit over copper) • IEEE 802.3an-compliant (10 gigabit over copper)
The device does not connect at the expected speed.	When Gigabit PHY Mode is forced to Primary mode on both the Intel adapter and its link partner, the link speed obtained by the Intel adapter may be lower than expected or link may not be established.
The adapter stops working without apparent cause	Run the adapter and network tests described in "Diagnostics in Intel PROSet" on page 190.
The Link indicator light is off	<p>Run the adapter and network tests described in "Diagnostics in Intel PROSet" on page 190.</p> <p>Make sure the proper (and latest) driver is loaded.</p> <p>Make sure that the link partner is configured to auto-negotiate (or forced to match adapter)</p> <p>Verify that the switch is IEEE 802.3ad-compliant.</p>
The link light is on, but communications are not properly established	<p>Make sure the proper (and latest) driver is loaded.</p> <p>Both the adapter and its link partner must be set to either auto-detect or manually set to the same speed and duplex settings.</p> <p> NOTE: The adapter's link indicator light may be on even if communications between the adapter and its link partner have not been properly established. Technically, the link indicator light represents the presence of a carrier signal but not necessarily the ability to properly communicate with a link partner. This is expected behavior and is consistent with IEEE's specification for physical layer operation.</p>
RX or TX light is off	Network may be idle; try creating traffic while monitoring the lights.
The diagnostic utility reports the adapter is	The PCI BIOS isn't configuring the adapter correctly. See "PCI / PCI-X /

Problem	Solution
"Not enabled by BIOS"	PCI Express Configuration" later in this table.
The computer hangs when the drivers are loaded	Try changing the PCI BIOS interrupt settings. See "PCI / PCI-X / PCI Express Configuration" later in this table.
PCI / PCI-X / PCI Express Configuration	<p>If the adapter is not recognized by your OS or if it does not work you may need to change some BIOS settings. Try the following only if you are having problems with the adapter and are familiar with BIOS settings.</p> <ul style="list-style-type: none"> • Check to see that the "Plug-and-Play" setting is compatible with the operating system you are using. • Make sure the slot is enabled. • Configure interrupts for level-triggering, as opposed to edge-triggering. • Reserve interrupts and/or memory addresses. This prevents multiple buses or bus slots from using the same interrupts. Check the BIOS for IRQ options for PCI / PCI-X / PCIE.
Driver message: "Rx/Tx is disabled on this device because an unsupported SFP+ module type was detected."	You installed an unsupported module in the device. See "Supported SFP+, SFP28, QSFP+, and QSFP28 Modules" on page 9 for a list of supported modules.

10.2 Multiple Adapters

When configuring a multi-adapter environment, you must upgrade all Intel adapters in the computer to the latest software.

If the computer has trouble detecting all adapters, consider the following:

- If you enable Wake on LAN* (WoL) on more than two adapters, the Wake on LAN feature may overdraw your system's auxiliary power supply, resulting in the inability to boot the system and other unpredictable problems. For multiple desktop/management adapters, it is recommended that you install one adapter at a time and use the IBAUtil utility (ibautil.exe in \APPS\BOOTAGNT) to disable the WoL feature on adapters that do not require WoL capabilities. On server adapters, the WoL feature is disabled by default.
- Adapters with Intel Boot Agent enabled will require a portion of the limited start up memory for each adapter enabled. Disable the service on adapters that do not need to boot Pre-Boot Execution Environment (PXE).

10.3 Health Status Messages

Intel® Ethernet 800 Series devices support asynchronous health status messages, which help you to debug system-level issues and diagnostics in the field. This feature is enabled by default and cannot be shut off.

When the firmware detects an abnormal event during initialization, it will push health status information to the base driver's system event log, such as dmesg or the Windows Event Log. Health status messages could encompass issues related to:

- Unsupported modules
- The NVM or option ROM
- Invalid link configuration

- Port speed
- Link partner
- Other issues

The system log will identify the device experiencing the issue, list information about the problem, and suggest a possible solution, such as updating to the latest NVM image or checking the cable or module.

See your system log for more information, if you are experiencing issues on Intel Ethernet 800 Series devices.

10.4 Safe Mode

Adapters based on the Intel® Ethernet 800 Series require a [Dynamic Device Personalization \(DDP\)](#) package file to enable advanced and performance features. If the driver detects a missing or incompatible DDP package file, the driver will go into Safe Mode. Safe Mode supports only basic traffic and minimal functionality, such as updating the NVM or downloading a new driver or DDP package.



NOTES:

- Safe Mode only applies to the affected physical function and does not impact any other PFs.
- [Firmware Recovery Mode](#) takes precedence over Safe Mode.

10.4.1 Safe Mode Errors and Messages

When the driver is in Safe Mode, the device drivers and utilities may log or display messages to help with troubleshooting. The following conditions may cause the driver to enter Safe Mode:

- The DDP package file was not found or couldn't be read.
- The DDP package file's version number, signature, or other metadata aren't valid or aren't supported by the driver.
- An unknown error occurred when loading the DDP package.
- The driver couldn't load the DDP package file because a compatible DDP package is already present on the device.
- The device has a DDP package that isn't supported by the driver.

10.4.2 Resolving Safe Mode Issues

The device drivers and utilities may display the action to take to get out of Safe Mode, depending on the underlying cause. Possible actions could include the following:

- Wait for the device to reset.
- Install the latest driver.
- Download a new DDP package.
- Restart the adapter. If the problem persists, install the latest driver.
- Reboot the system. If the problem persists, update the NVM.

You can download the latest drivers and DDP packages from the [Dell support website](#).

10.5 PF Message Queue Overflow

The device driver can detect some types of anomalous behavior. When it does, it will log the VF MAC address and associated PF MAC address. Using this information, you can check the virtual machine (VM) that is using the VF MAC address to ensure that the VM is operating correctly.

10.6 Possible Misconfiguration of the Ethernet Port

You may see an informational message stating that a potential misconfiguration of the Ethernet port was detected. This is to alert you that your device is being underutilized. If this was intentional, you may ignore this message. For example, setting your Intel® Ethernet 100G 2P E810-C Adapter to 2x2x25 is valid, but it does not use the full capabilities of the device. If you see this message, and the configuration was not intentional, you may use the Ethernet Port Configuration Tool (EPCT) to correct the configuration.

10.7 Other Performance Issues

Attaining the best speed requires that many components are operating at peak efficiency. Among them are the following:

- **Cable quality and length** - Do not exceed the maximum recommended length for your cable type. Shorter lengths often provide better results. Check for loose or damaged connectors. Check the cable for kinked or damaged sections.
- **Bus speed and traffic** - The PCI bus speed accommodates the slowest PCI card installed. Check to see if you have a card that is slowing down your system.
- **Processor and Memory** - Check your performance monitoring programs to see if traffic is being affected by your processor speed, available memory or other processes.
- **Transmission frame size** - Your network performance may be enhanced by adjusting or maximizing the transmission frame size. Operating systems, switches and adapters will impose varying limits on maximum frame size. See the discussion on Jumbo Frames for your OS.
- **Operating System** - Networking feature implementation will vary by operating system version, such as offloading and multiprocessor threading.

10.8 Diagnostics

Multiple utilities are available to help troubleshoot and diagnose issues with your Intel Ethernet devices or Dell system.

10.8.1 Dell Diagnostic Utilities

One diagnostic tool that Dell provides is the Enhanced Pre-boot System Assessment (ePSA), which is accessible through the Dell Boot Manager. Refer to your system documentation for more information on diagnostic tests available through this tool.



NOTE: Intel Ethernet 700 Series and 800 Series devices may list an incorrect speed under the driver name in ePSA. This behavior is expected and does not impact the functionality of the device.

10.8.2 Linux Diagnostics

Intel Ethernet drivers use the ethtool interface for driver configuration and diagnostics, as well as displaying statistical information. The latest ethtool version is required for this functionality. Download it at <https://kernel.org/pub/software/network/ethtool/>.

10.8.3 Diagnostics in Intel PROSet

Intel's diagnostic software lets you test the adapter to see if there are problems with the adapter hardware, the cabling, or the network connection. Refer to "About Intel PROSet®" on page 13 for an overview of this software.

Intel PROSet allows you to run the following types of diagnostic tests on supported Windows operating systems.

Type of Test	Description
Connection Test	Verifies network connectivity by pinging the DHCP server, WINS server, and gateway.
Cable Tests	Provides information about cable properties. ¹
Hardware Tests	Determines if the adapter is functioning properly. ²
Note: <ul style="list-style-type: none"> The Cable Test is not supported on all adapters and will not run on Direct Attached Cables (DAC) or Fiber. The Cable Test will only be available on adapters that support it. 	

In Intel PROSet ACU, use the Diagnostics panel.

The availability of these tests is dependent on the adapter and operating system. Tests may be disabled if:

- The port is used as a manageability port.
- The tests are being run from a virtual machine.



NOTE: At this time, Windows diagnostics are not supported on ports based on an Intel Ethernet Controller I225 and will fail.

10.8.3.1 Testing from Windows PowerShell

Intel provides two [PowerShell cmdlets](#) for testing your device.

- Test-IntelNetDiagnostics runs the specified test suite on the specified device. See the Test-IntelNetDiagnostics help inside PowerShell plus the following table for more information.
- Test-IntelNetIdentifyAdapter blinks the LED on the specified device.

The following table describes the possible result codes from the Test-IntelNetDiagnostics cmdlet (where <X> below indicates a value returned in the message).

Result Code	Description
00	Successfully pinged gateway address <X>.
01	Successfully pinged DHCP address <X>.
02	Successfully pinged DNS address <X>.
03	Successfully pinged WINS address <X>.
04	An instance of this test is already running. You can only have one test running at a time.

Result Code	Description
05	TCP/IP protocol is not configured. To run this test, configure this connection to use the TCP/IP protocol.
07	The IP address for this connection is invalid. Possible cause: The system may be waiting for a response from a DHCP server.
09	The IP address for this connection is invalid.
18	You cannot run this test while the device is included in a team or VLAN.
19	Auto-negotiation is not enabled on the device. The device has been configured to force a lower speed.
20	Auto-negotiation is not complete on this device. Please wait and try again later.
21	A Category 5 (or better) cable is required to run at 1 Gbps. The cable connected to the device either is not Category 5 (or better) or has faulty wires.
22	Link speed: <X> Mbps. The link partner is not capable of higher speeds.
23	The link partner is not advertising a compatible speed. Please check that the link partner supports 1 Gbps.
24	Link speed has been reduced because a Power Saver option is enabled. Disable power saving in your OS settings and run the test again.
25	The device is configured to force a lower speed.
26	No cable problems detected.
27	Could not run the test at this time. Please try again later.
33	The test detected a bad connection. Distance to problem: <X> meters.
34	Passed
35	Failed
36	Cable quality is unknown.
37	Passed
38	Failed
39	Cable quality is unknown.
40	Cable polarity is normal.
41	Cable polarity is reversed.
42	Cable length: <X> meters.
43	This device is running at full speed.
44	Cable quality is poor or no cable is connected. Possible causes: Faulty cable, faulty connector, or a speed/duplex mismatch. Verify that the speed/duplex setting on the switch/hub is configured for auto-negotiation.
45	Cable quality is poor or no cable is connected. Possible causes: Faulty cable, faulty connector, or a speed/duplex mismatch.

Result Code	Description
46	Cable quality is adequate.
47	Cable quality is good.
48	Cable quality is excellent.
49	Cable quality is unknown.
50	The test detected a frequency response that does not meet IEEE specifications.
51	This device does not have link. Make sure the cable is connected and the speed and duplex settings are configured correctly on the device and link partner.
52	Passed
53	Passed
54	Passed
55	Passed
56	Passed
57	Passed
58	Failed
59	Failed
60	Failed
61	Failed
62	Failed
63	Failed
64	Passed
65	Failed
66	This device requires a restart.
67	Could not run the test at this time. The device may be connected to a remote target. Disconnecting the cable will enable the test.
68	Could not run the test at this time. This device is used as a manageability port. Disconnecting the cable will enable the test.
69	The device does not support this test.
70	This test relies on a response from a gateway, DNS, DHCP, or WINS server and no such response was received. Any such server for this connection may be unavailable or misconfigured.
71	Cable integrity is unknown.

Result Code	Description
72	<p>This test relies on a response from a gateway, DNS, DHCP, or WINS server and no such response was received. Any such server for this connection may be unavailable or misconfigured.</p> <p>This device is configured to automatically obtain an IP address but no DHCP server is present on the network. Windows selected an IP address using alternate private IP addressing.</p>
73	Temperature is normal.
74	<p><i>Possible values:</i></p> <p>The device overheated and was stopped.</p> <p>The device overheated. Link speed was reduced.</p>

10.9 Event Log in Microsoft Windows

The following tables shows the Windows event log service names for each family of devices:

Intel® Ethernet Controller	NDIS Driver File Names	Windows Event Log Service Name
I350	E1r*.sys	e1repress
I354	E1s*.sys	e1sexpress
X550	Ixs*.sys	ixgbs
710 Series	I40ea*.sys	i40ea
810 Series	icea.sys	icea
823 Series	scea.sys	scea

11. Known Issues

Fiber optics and auto-negotiation

Modules based on 100GBASE-SR4, 40GBASE-SR4, 25GBASE-SR, active optical cable (AOC), and active copper cable (ACC) do not support auto-negotiation per the IEEE specification. To obtain link with these modules, you must turn off auto-negotiation on the link partner's switch ports.

Link issues at speeds faster than 10 Gbps

If you are having link issues (including no link) at link speeds faster than 10 Gbps, check your switch configuration and/or specifications. Many optical connections and direct attach cables require RS-FEC for connection speeds faster than 10 Gbps. One of the following may resolve the issue:

- Configure your switch to use RS-FEC mode.
- Specify a 10 Gbps, or slower, link speed connection.
- If you are attempting to connect at 25 Gbps, try using an SFP28 CA-S or CS-N Direct Attach cable. These cables do not require RS-FEC.
- If your switch does not support RS-FEC mode, check with your switch vendor for the availability of a SW or FW upgrade.

iDRAC Firmware Rollback

In certain circumstances, the firmware rollback feature in iDRAC may not be available. In this situation, please download and rerun the DUP with the desired firmware level from the Dell website.

The get-netadaptervmq PowerShell cmdlet displays less than the expected number of receive queues

After installing the Dell Update Package (DUP), the get-netadaptervmq PowerShell cmdlet reports 31 queues per port. This is expected behavior. The DUP changes the queue pooling default from pairs to groups of four. Pre-DUP, queues are paired into pools of two. After the DUP is installed, queues put into groups of four. This decreases the number of queues displayed by the get-netadaptervmq cmdlet.

NVM update utilities exit with error on Linux kernel 4.16 or higher

On Linux kernel 4.16 or higher, if you update the ixgbe, igb, or i40e driver and then run any of the NVM update utilities (NVMUpdate or NVMCheck), the utility may exit with the error "The selected adapter cannot be updated due to strict MMIO memory settings in the kernel." To fix this, set the iomem kernel parameter to "relaxed" (i.e., iomem=relaxed) and reboot the system before running the tool again. On kernel 4.16 or higher, the iomem parameter is set to "strict" by default, which prevents the NVM update utilities from accessing the MMIO of the device.

"Rx/Tx is disabled on this device because the module does not meet thermal requirements." error during POST

This error is caused by installing a module in an X710 based device that does not meet thermal requirements for that device. To resolve the issue, please install a module that meets the device's thermal requirements. See "Supported SFP+, SFP28, QSFP+, and QSFP28 Modules" on page 9 in this user guide for more information.

"Rx/Tx is disabled on this device because an unsupported SFP+ module type was detected." error during POST

This error is caused by installing an unsupported module in an X710/XL710 based device. You will not be able to send or receive traffic on this device. To resolve the issue, please install a supported module. See "Supported SFP+, SFP28, QSFP+, and QSFP28 Modules" on page 9 in this user guide for more information.

Throughput Reduction After Hot-Replace

If an Intel gigabit adapter is under extreme stress and is hot-swapped, throughput may significantly drop. This may be due to the PCI property configuration by the Hot-Plug software. If this occurs, throughput can be restored by restarting the system.

CPU Utilization Higher Than Expected

Setting RSS Queues to a value greater than 4 is only advisable for large servers with several processors. Values greater than 4 may increase CPU utilization to unacceptable levels and have other negative impacts on system performance.

Supported SFP or SFP+ Module Not Recognized by the System

If you try to install an unsupported module, the port may no longer install any subsequent modules, regardless of whether the module is supported or not. The port will show a yellow bang under Windows Device Manager and an event id 49 (unsupported module) will be added to the system log when this issue occurs. To resolve this issue, the system must be completely powered off.

Receive Error counts may be higher than the actual packet error count

When a packet is received with more than one error, two bad packets may be reported. This affects all devices based on 10G, or faster, controllers.

Logical switch creation fails

When you use Microsoft SCVMM (System Center Virtual Machine Manager) to create an encapsulation-enabled (VXLAN, NVGRE, etc.) logical switch, it may fail with error code 0x80041001. To resolve the issue, after the failure is seen, perform the following:

1. Disable the "Hyper-V Extensible Virtual Switch" property on the Intel Ethernet device.
2. Recreate the logical switch.

This affects hosts running Microsoft Windows Server 2019.

11.1 Windows Known Issues

Intel PROSet ACU does not show Virtualization profile with Hyper-V installed

On a system running Microsoft Windows Server, you may not see the Virtualization profile in Intel PROSet ACU when Hyper-V is installed. To see and set a Virtualization profile or Storage + Virtualization profile for Intel Ethernet devices, you must enable SR-IOV. If Virtualization profiles are not available for the adapter, check that Hyper-V is installed on the system and that SR-IOV is enabled on the Intel NIC in the HII and BIOS.

Less than the expected number of RSS queues are assigned

In a Microsoft Windows Server 2019 or Windows Server 2022 VM, if the host has hyper-threading enabled, when you assign RSS queues to a virtual adapter, less than the requested number of queues may be assigned. For example, if you have 16 RSS CPUs available, and try to set NumberOfReceiveQueues to 16, only 8 RSS queues will be assigned. This is due to a known issue in the operating system. Disabling hyper-threading on the host may mitigate the issue. Customers should contact Microsoft via the appropriate support channel for a solution.

Unable to shutdown virtual machine

Multiple VF failover events may leave a VM in an unstable state. You may not be able to shutdown the VM. Rebooting the host will resolve the issue.

hv_vmbus probe error on a Linux guest in a Windows Server system

On a system running Microsoft Windows Server 2019 on the host and Linux in the VF, you may see an "hv_vmbus: probe failed for device X" error in dmesg after you change a vSwitch from VMQ to SRIOV. This is due to a known timing issue in the operating system. There is no functionality loss, and the VF will successfully start after a few failed probes.

Incomplete branding string displayed in the event log

Some branding strings are too long to be displayed fully in the event log. In these cases, the branding string will be truncated and the port's PCI Bus/Device/Function are appended to the string. For example: Intel(R) Ethernet Converged Network Ad... [129,0,1].

PcieLinkSpeed is Unknown

When you install an Intel® Ethernet 800 Series device in a PCI Gen 4 slot, the operating system may report PcieLinkSpeed as Unknown. This does not affect the operation of the device.

Port is missing from Lifecycle Controller : Network Settings

If a port is configured for iSCSI boot, and it successfully connected to its boot target, then you cannot modify the port settings in the Lifecycle Controller.

Procedure for Installing and Upgrading Drivers and Utilities

Intel does not recommend installing or upgrading drivers and Intel® PROSet software over a network connection. Instead, install or upgrade drivers and utilities from each system.

Adapter Settings Change While Traffic is Running

In the Adapter Settings panel of Intel PROSet ACU, parameters should not be modified under heavy network loads. Otherwise, a reboot may be required to make the changes effective.

Intel drivers must be installed by Dell Update Package before configuring Microsoft Hyper-V features

Prior to configuring the Microsoft* Hyper-V features, the Intel® NIC drivers must be installed by the Dell Update Package.

Application Error Event IDs 789 and 790 in the Event Log

If Data Center Bridging (DCB) is enabled, and the enabled port loses link, the following three events may be logged in the event log:

- Event ID 789: Enhanced Transmission Selection feature on a device has changed to non-operational
- Event ID 790: Priority Flow Control feature on a device has changed to non-operational

This is the expected behavior when a DCB enabled port loses link. DCB will begin working again as soon as link is reestablished. A port will lose link if the cable is disconnected, the driver or software package is updated, if the link partner goes down, or for other reasons.

"Malicious script detected" Warning from Norton AntiVirus During PROSet Uninstall

The Intel PROSet uninstall process uses a Visual Basic script as part of the process. Norton AntiVirus and other virus scanning software may mistakenly flag this as a malicious or dangerous script. Letting the script run allows the uninstall process to complete normally.

RSS Load Balancing Profile Advanced Setting

Setting the "RSS load balancing profile" Advanced Setting to "ClosestProcessor" may significantly reduce CPU utilization. However, in some system configurations (such as a system with more Ethernet ports than processor cores), the "ClosestProcessor" setting may cause transmit and receive failures. Changing the setting to "NUMAScalingStatic" will resolve the issue.

11.2 Linux Known Issues

11.2.1 All Linux Drivers

Unless noted otherwise, this section describes issues that apply to multiple Intel Ethernet Linux drivers.

Link LEDs May Be Off During Offline Installation of Linux OS

When performing an offline installation (installation without a NIC connected to a valid network) of SUSE Linux Enterprise Server 15, the PHY link is disabled in the OS even if the Ethernet cable is plugged in. The PHY link is disabled due to the Intel driver disabling the link for power savings when the network is not in use. Configuring the network setting during or after the installation restores link LEDs.

Software Issues

After installing the driver, if your Intel® Ethernet Network Connection is not working, verify that you have installed the correct driver. Intel® Active Management Technology 2.0, 2.1, and 2.5 are not supported in conjunction with the Linux driver.

Compiling the Driver

When trying to compile the driver by running `make install`, the following error may occur: "Linux kernel source not configured - missing version.h"

To solve this issue, create the version.h file by going to the Linux source tree and entering:

```
# make include/linux/version.h
```

Multiple Interfaces on Same Ethernet Broadcast Network

Due to the default ARP behavior on Linux, it is not possible to have one system on two IP networks in the same Ethernet broadcast domain (non-partitioned switch) behave as expected. All Ethernet interfaces will respond to IP traffic for any IP address assigned to the system. This results in unbalanced receive traffic.

If you have multiple interfaces in a server, either turn on ARP filtering by entering:

```
# echo 1 > /proc/sys/net/ipv4/conf/all/arp_filter
```



NOTE: This setting is not saved across reboots. The configuration change can be made permanent by adding the following line to the file `/etc/sysctl.conf`:

```
net.ipv4.conf.all.arp_filter = 1
```

Another alternative is to install the interfaces in separate broadcast domains (either in different switches or in a switch partitioned to VLANs).

MAC Address of Virtual Function Changes Unexpectedly

If a Virtual Function's MAC address is not assigned in the host, then the VF (virtual function) driver will use a random MAC address. This random MAC address may change each time the VF driver is reloaded. You can assign a static MAC address in the host machine. This static MAC address will survive a VF driver reload.

SR-IOV virtual functions have identical MAC addresses

When you create multiple SR-IOV virtual functions, the VFs may have identical MAC addresses. Only one VF will pass traffic, and all traffic on other VFs with identical MAC addresses will fail. This is related to the "MACAddressPolicy=persistent" setting in `/usr/lib/systemd/network/99-default.link`.

To resolve this issue, edit the `/usr/lib/systemd/network/99-default.link` file and change the MACAddressPolicy line to "MACAddressPolicy=none". For more information, see the `systemd` documentation.

Rx Page Allocation Errors

'Page allocation failure. order:0' errors may occur under stress with kernels 2.6.25 and newer. This is caused by the way the Linux kernel reports this stressed condition.

On `igb` devices, unloading the PF driver causes the system to reboot when the VM is running and VF is loaded on the VM. Do not unload the PF driver (`igb`) while VFs are assigned to guests.

Lower Than Expected Performance

Some PCIe x8 slots are actually configured as x4 slots. These slots have insufficient bandwidth for full line rate with dual port and quad port devices. In addition, if you put a PCIe v4.0 or v3.0-capable adapter into a PCIe v2.x slot, you cannot get full bandwidth. The driver detects this situation and writes one of the following messages in the system log:

```
"PCI-Express bandwidth available for this card is not sufficient for optimal performance. For optimal performance a x8 PCI-Express slot is required."
```

or

```
"PCI-Express bandwidth available for this device may be insufficient for optimal performance. Please move the device to a different PCI-e link with more lanes and/or higher transfer rate."
```

If this error occurs, moving your adapter to a true PCIe v3.0 x8 slot will resolve the issue. For best performance on the Intel Ethernet 800 Series, the device needs to be installed in a PCIe v4.0 x8 or v3.0 x16 slot.

Unplugging Network Cable While ethtool -p is Running

In some newer kernel versions, unplugging the network cable while ethtool -p is running will cause the system to become unresponsive to keyboard commands, except for control-alt-delete. Restarting the system appears to be the only remedy.

11.2.2 iavf Known Issues



NOTE: Refer to "Linux Known Issues" on page 197 for additional issues common to multiple Linux drivers.

Linux bonding fails with Virtual Functions bound to an Intel® Ethernet 700 Series device

If you bind Virtual Functions (VFs) to an Intel® Ethernet 700 Series device, the VF targets may fail when they become the active target. If the MAC address of the VF is set by the PF (Physical Function) of the device, when you add a target, or change the active-backup target, Linux bonding tries to sync the backup target's MAC address to the same MAC address as the active target. Linux bonding will fail at this point. This issue will not occur if the VF's MAC address is not set by the PF.

When using bonding mode 5 (i.e., balance-tlb or adaptive transmit load balancing), if you add multiple VFs to the bond, they are assigned duplicate MAC address. When the VFs are joined with the bond interface, the Linux bonding driver sets the MAC address for the VFs to the same value. The MAC address is based on the first active VF added to that bond. This results in balance-tlb mode not functioning as expected. PF interfaces behave as expected. The presence of duplicate MAC addresses may cause further issues, depending on your switch configuration.

Traffic Is Not Being Passed Between VM and Client

You may not be able to pass traffic between a client system and a Virtual Machine (VM) running on a separate host if the Virtual Function (VF, or Virtual NIC) is not in trusted mode and spoof checking is enabled on the VF. Note that this situation can occur in any combination of client, host, and guest operating system. See the readme for the PF driver for information on spoof checking and how to set the VF to trusted mode.

Using four traffic classes fails

Do not try to reserve more than three traffic classes in the iavf driver. Doing so will fail to set any traffic classes and will cause the driver to write errors to stdout. Use a maximum of three queues to avoid this issue.

Unexpected errors in dmesg when adding TCP filters on the VF

When ADQ is configured and the VF is not in trusted mode, you may see unexpected error messages in dmesg on the host when you try to add TCP filters on the VF. This is due to the asynchronous design of the iavf driver. The VF does not know whether it is trusted and appears to set the filter, while the PF blocks the request and reports an error. See the dmesg log in the host OS for details about the error.

Multiple log error messages on iavf driver removal

If you have several VFs and you remove the iavf driver, several instances of the following log errors are written to the log:

```
Unable to send opcode 2 to PF, err I40E_ERR_QUEUE_EMPTY, aq_err ok
```

```
Unable to send the message to VF 2 aq_err 12
```

```
ARQ Overflow Error detected
```

11.2.3 ice Known Issues



NOTE: Refer to "Linux Known Issues" on page 197 for additional issues common to multiple Linux drivers.

Dynamic Debug

If you encounter unexpected issues during driver load, some of the most useful information for developers to receive in a bug report can include driver logging. This logging uses a kernel feature called Dynamic Debug, which is generally enabled in most kernel configurations (`CONFIG_DYNAMIC_DEBUG=y`).

To load the driver with dynamic debug enabled, run `modprobe` with the `dyndbg` parameter:

```
# modprobe ice dyndbg=+p
```

The driver will then load and print debugging information into the kernel log (`dmesg`) and is usually logged into the system log viewable by `journalctl` or in `/var/log/messages`. Saving this information to a file and attaching it to any bug report can help shorten the reproduction and debugging time for a developer.

To enable dynamic debug during runtime operation of the driver, use this command:

```
# echo "module ice +p" > /sys/kernel/debug/dynamic_debug/control
```

For more details, see the Dynamic Debug documentation included in the Linux kernel instructions.

PF Message Queue Overflow

The device driver can detect some types of anomalous behavior. When it does, it will log the VF MAC address and associated PF MAC address. Using this information, you can check the virtual machine (VM) that is using the VF MAC address to ensure that the VM is operating correctly.

'ethtool -S' does not display Tx/Rx packet statistics

Issuing the command `'ethtool -S'` does not display Tx/Rx packet statistics. This is by convention. Use other tools (such as the `ip` command) that display standard netdev statistics such as Tx/Rx packet statistics.

'ethtool -S' rx_bytes and ip stats rx_bytes don't match statistics

The `rx_bytes` value of `ethtool` does not match the `rx_bytes` value of `Netdev`, due to the 4-byte CRC being stripped by the device. The difference between the two `rx_bytes` values will be 4 x the number of Rx packets. For example, if Rx packets are 10 and `Netdev` (software statistics) displays `rx_bytes` as "X", then `ethtool` (hardware statistics) will display `rx_bytes` as "X+40" (4 bytes CRC x 10 packets).

'ethtool -a' autonegotiate result may vary between drivers

For kernel versions 4.6 or higher, 'ethtool -a' will show the advertised and negotiated autoneg settings. For kernel versions below 4.6, ethtool will only report the negotiated link status.

The issue is cosmetic and does not affect functionality. Installing the latest ice driver and upgrading your kernel to version 4.6 or higher will resolve the issue.

AF_XDP fails to allocate buffers

On kernels older than 5.3, you may see an undesirable CPU load during packet processing if you enable AF_XDP in native mode and the Rx ring size is larger than the UMEM fill queue. This is due to a known issue in the kernel and was fixed in 5.3. To address the issue, upgrade your kernel to 5.3 or newer.

SCTP checksum offloads aren't indicated on Geneve tunnel

For SCTP traffic over a Geneve tunnel, the SCTP checksum isn't offloaded to the device, even when tx-checksum-sctp is on. This is due to a limitation in the Linux kernel. However, for Rx traffic, the SCTP checksum is verified if rx-checksumming is on. For both Tx and Rx traffic, you can offload the outer UDP checksum to the device.

Incorrect link speed reported on older VF drivers

Linux distributions with older iavf or i40evf drivers (including Red Hat Enterprise Linux 8) may show an incorrect link speed on VF interfaces. This issue is cosmetic and does not affect VF functionality. To resolve the issue, download the latest iavf driver.

'VF X failed opcode 24' error message in dmesg on host

With a Microsoft Windows Server 2019 guest machine running on a Linux host, you may see 'VF <vf_number> failed opcode 24' error messages in dmesg on the host. This error is benign and does not affect traffic. Installing the latest iavf driver in the guest will resolve the issue.

Windows guest OSs on a Linux host may not pass traffic across VLANs

The VF is not aware of the VLAN configuration if you use Load Balancing and Failover (LBFO) to configure VLANs in a Windows guest. VLANs configured using LBFO on a VF driver may result in failure to pass traffic.

MDD events in dmesg when creating maximum number of VLANs on the VF

When you create the maximum number of VLANs on the VF, you may see MDD events in dmesg on the host. This is due to the asynchronous design of the iavf driver. It always reports success to any VLAN requests, but the requests may fail later. The guest OS could try to send traffic on a VLAN that is not configured on the VF, which will cause a Malicious Driver Detection (MDD) event in dmesg on the host.

This issue is cosmetic. You do not need to reload the PF driver.

'ip address' or 'ip link' command displays an error on a single-port NIC with 245+ VFs

When you use the 'ip address' or 'ip link' command on a Linux host configured with 245 or more VFs on a single-port adapter, you may encounter a "Buffer too small for object" error. This is due to a known issue in the iproute2 tools. Please use ifconfig instead of iproute2. You can install ifconfig via the net-tools-deprecated package.

11.2.4 i40e Known Issues



NOTE: Refer to "Linux Known Issues" on page 197 for additional issues common to multiple Linux drivers.

Linux bonding fails with Virtual Functions bound to an Intel® Ethernet 700 Series device

If you bind Virtual Functions (VFs) to an Intel® Ethernet 700 Series device, the VF targets may fail when they become the active target. If the MAC address of the VF is set by the PF (Physical Function) of the device, when you add a target, or change the active-backup target, Linux bonding tries to sync the backup target's MAC address to the same MAC address as the active target. Linux bonding will fail at this point. This issue will not occur if the VF's MAC address is not set by the PF.

ip link show command shows incorrect VF MAC if VF MAC was set from VF side

Executing the command "ip link show" only shows MAC addresses if they are set by the PF. Otherwise, it shows all zeros.

This is expected behavior. The PF driver is passing zeroes to the VF driver that the VF driver can generate its own random MAC address and report it to the guest OS. Without this feature, some guest operating systems will incorrectly assign the VF a new interface name each time they reboot.

IPv6/UDP checksum offload does not work on some older kernels

Some distributions with older kernels do not properly enable IPv6/UDP checksum offload. To use IPv6 checksum offload, it may be necessary to upgrade to a newer kernel.

depmod warning messages about unknown symbol during installation

During driver installation, you may see depmod warning messages referring to unknown symbols `i40e_register_client` and `i40e_unregister_client`. These messages are informational only; no user action is required. The installation should complete successfully.

Error: <ethX> selects TX queue XX but real number of TX queues is YY


When configuring the number of queues under heavy traffic load, you may see an error message stating "<ethX> selects TX queue XX, but real number of TX queues is YY". This message is informational only and does not affect functionality.

Fixing Performance Issues When Using IOMMU in Virtualized Environments

The I/O Memory Management Unit (IOMMU) feature of the processor prevents I/O devices from accessing memory outside the boundaries set by the OS. It also allows devices to be directly assigned to a Virtual Machine. However, IOMMU may affect performance, both in latency (each DMA access by the device must be translated by the IOMMU) and in CPU utilization (each buffer assigned to every device must be mapped in the IOMMU).

If you experience significant performance issues with IOMMU, try using it in "passthrough" mode by adding the following to the kernel boot command line:

```
intel_iommu=on iommu=pt
```

 **NOTE:** This mode enables remapping for assigning devices to VMs, providing near-native I/O performance, but does not provide the additional memory protection.

If you plan to direct-assign devices to a VM in Linux, you must enable IOMMU support for SR-IOV to function correctly. Use the kernel boot parameters "intel_iommu=on" for system boards with Intel processors or "amd_iommu=on" for systems boards with AMD processors, and "iommu=pt" to enable IOMMU support. For the best memory protection, use "intel_iommu=on." For the best performance, use both parameters ("intel_iommu=on iommu=pt").

In Red Hat and most other Linux distributions, append these parameters to the `GRUB_CMDLINE_LINUX` entry in the `/etc/default/grub` configuration file. For systems booting in UEFI mode, run `grub2-mkconfig -o /etc/grub2-efi.cfg`. For systems booting in legacy BIOS mode, run `grub2-mkconfig -o /boot/grub2/grub.cfg`.

In SUSE based Linux distributions, add these parameters by opening Yast and then opening the Boot Loader and clicking the Kernel Parameters tab. Add the optional parameters in the Optional Kernel Command Line Parameter field. This adds the options for either boot mode.

You will need to reboot for these changes to take effect.

Transmit hangs leading to no traffic

Disabling flow control while the device is under stress may cause tx hangs and eventually lead to the device no longer passing traffic. You must reboot the system to resolve this issue.

Bad checksum counter incorrectly increments when using VxLAN

When passing non-UDP traffic over a VxLAN interface, the `port.rx_csum_bad` counter increments for the packets.

Statistic counters reset when promiscuous mode is changed

Changing promiscuous mode triggers a reset of the physical function driver. This will reset the statistic counters.

Changing the number of Rx or Tx queues with `ethtool -L` may cause a kernel panic

Changing the number of Rx or Tx queues with `ethtool -L` while traffic is flowing and the interface is up may cause a kernel panic. Bring the interface down first to avoid the issue. For example:

```
# ip link set <ethX> down
# ethtool -L <ethX> combined 4
```

Intel Ethernet Flow Director Sideband Logic adds duplicate filter

The Intel Ethernet Flow Director Sideband Logic adds a duplicate filter in the software filter list if the location is not specified or the specified location differs from the previous location but has the same filter criteria. In this case, the second of the two filters that appear is the valid one in hardware and it decides the filter action.

UDP Stress Test Dropped Packet Issue

Under small packet UDP stress with the i40e driver, the system may drop UDP packets due to socket buffers being full. Setting the driver Intel Ethernet Flow Control variables to the minimum may resolve the

issue. You may also try increasing the kernel's default buffer sizes by changing the values in `/proc/sys/net/core/rmem_default` and `rmem_max`

'ethtool -a' autonegotiate result may vary between drivers

For kernel versions 4.6 or higher, 'ethtool -a' will show the advertised and negotiated autoneg settings. For the i40e driver and kernel versions below 4.6, ethtool will only report the negotiated link status.

The issue is cosmetic and does not affect functionality.

Running ethtool -t ethX command causes break between PF and test client

When there are active VFs, "ethtool -t" performs a full diagnostic. In the process, it resets itself and all attached VFs. The VF drivers encounter a disruption but are able to recover.

Unable to obtain DHCP lease on boot with Red Hat

In configurations where the auto-negotiation process takes more than 5 seconds, the boot script may fail with the following message:

```
"ethX: failed. No link present. Check cable?"
```

This error may occur even though the presence of link can be confirmed using `ethtool ethx`. In this case, try setting `"LINKDELAY=30"` in `/etc/sysconfig/network-scripts/ifdfg-ethx`.

The same issue can occur during a network boot (via PXE) on Red Hat distributions that use the dracut script:

```
"Warning: No carrier detected on interface <interface_name>"
```

In this case add `"rd.net.timeout.carrier=30"` at the kernel command line.



NOTE: Link time can vary. Adjust LINKDELAY value accordingly.

Alternatively, NetworkManager can be used to configure the interfaces, which avoids the set timeout. For configuration instructions of NetworkManager refer to the documentation provided by your distribution.

Loading i40e driver in 3.2.x and newer kernels displays kernel tainted message

Due to recent kernel changes, loading an out of tree driver causes the kernel to be tainted.

VLAN pruning doesn't work when traffic is sent from local system

VLAN tagged traffic sent from other VFs attached to a NIC can be seen on other VFs. This is due to an issue with NIC settings and offload settings of VFs.

To work around this issue, disable Tx VLAN offload on VFs:

```
# ethtool -K <ethX> tx-vlan-offload off
```

This does not affect egress traffic from the outside NIC.

'VF X failed opcode 24' error message in dmesg on host

With a Microsoft Windows Server 2019 guest machine running on a Linux host, you may see 'VF <vf_number> failed opcode 24' error messages in `dmesg` on the host. This error is benign and does not affect traffic. Installing the latest iavf driver in the guest will resolve the issue.

Windows guest OSs on a Linux host may not pass traffic across VLANs

The VF is not aware of the VLAN configuration if you use Load Balancing and Failover (LBFO) to configure VLANs in a Windows guest. VLANs configured using LBFO on a VF driver may result in failure to pass traffic.

11.2.5 ixgbe Known Issues



NOTE: Refer to "Linux Known Issues" on page 197 for additional issues common to multiple Linux drivers.

HeaderDataSplit

HeaderDataSplit is not supported in 82599-based adapters.

UDP Stress Test Dropped Packet Issue

Under small packet UDP stress with the ixgbe driver, the system may drop UDP packets due to socket buffers being full. Setting the driver Intel Ethernet Flow Control variables to the minimum may resolve the issue. You may also try increasing the kernel's default buffer sizes by changing the values in `/proc/sys/net/core/rmem_default` and `rmem_max`

DCB: Generic segmentation offload on causes bandwidth allocation issues

In order for DCB to work correctly, Generic Segmentation Offload (GSO), also known as software TSO, must be disabled using `ethtool`. Since the hardware supports TSO (hardware offload of segmentation), GSO will not be running by default. The GSO state can be queried with `ethtool` using `ethtool -k ethX`. When using 82598-based network connections, ixgbe driver only supports 16 queues on a platform with more than 16 cores.

Due to known hardware limitations, RSS can only filter in a maximum of 16 receive queues.

82599 and X550-based network connections support up to 64 queues.

Running `ethtool -t ethX` command causes break between PF and test client

When there are active VFs, "`ethtool -t`" will only run the link test. The driver will also log in `syslog` that VFs should be shut down to run a full diagnostic test.

Unable to obtain DHCP lease on boot with Red Hat

In configurations where the auto-negotiation process takes more than 5 seconds, the boot script may fail with the following message:

```
"ethX: failed. No link present. Check cable?"
```

This error may occur even though the presence of link can be confirmed using `ethtool ethx`. In this case, try setting "`LINKDELAY=30`" in `/etc/sysconfig/network-scripts/ifdfg-ethx`.

The same issue can occur during a network boot (via PXE) on Red Hat distributions that use the `dracut` script:

```
"Warning: No carrier detected on interface <interface_name>"
```

In this case add "`rd.net.timeout.carrier=30`" at the kernel command line.



NOTE: Link time can vary. Adjust `LINKDELAY` value accordingly.

11.2.6 ixgbevf Known Issues



NOTE: Refer to "Linux Known Issues" on page 197 for additional issues common to multiple Linux drivers.

dmesg: Unable to start - perhaps the PF Driver isn't up yet

This message is posted when the PF interface is down when you try to change the number of Tx or Rx queues on the VF interface. To resolve the issue, bring the PF interface up and reload the VF driver.

11.2.7 igb Known Issues



NOTE: Refer to "Linux Known Issues" on page 197 for additional issues common to multiple Linux drivers.

Detected Tx Unit Hang in Quad Port Adapters

In some cases, ports 3 and 4 don't pass traffic and report "Detected Tx Unit Hang" followed by "NETDEV WATCHDOG: <ethX>: transmit timed out" errors. Ports 1 and 2 do not show any errors and will pass traffic.

This issue may be resolved by updating to the latest kernel and BIOS. You should use an OS that fully supports Message Signaled Interrupts (MSI) and make sure that MSI is enabled in your system's BIOS.

Performance Degradation with Jumbo Frames

Degradation in throughput performance may be observed in some Jumbo frames environments. If this is observed, increasing the application's socket buffer size and/or increasing the `/proc/sys/net/ipv4/tcp_*mem` entry values may help.

See the specific application manual and `/usr/src/linux*/Documentation/networking/ip-sysctl.txt` for more details.

Disable rx Flow Control with ethtool

In order to disable receive flow control using ethtool, you must turn off auto-negotiation on the same command line:

```
# ethtool -A <ethX> autoneg off rx off
```

Do Not Use LRO When Routing Packets

Due to a known general compatibility issue with LRO and routing, do not use LRO when routing packets.

11.3 Power Management Known Issues

Intel® Ethernet Controller X710 devices do not support Wake on LAN in multicast mode

Devices based on the Intel Ethernet Controller X710 do not support Wake on LAN in multicast mode.

System does not wake on link

On a driver-only installation, if you change 'Wake on Link Settings' to Forced and change 'Wake on Magic Packet' and 'Wake on Pattern Match' to Disabled, the system may not wake up when expected. In order to

"Wake on Link" successfully, check Adapter Settings panel in Intel PROSet ACU and make sure that "Allow this device to wake the computer" is checked. You may also need to change 'Wake on Magic Packet' or 'Wake on Pattern Match' to Enabled.

System Wakes-Up from a Removed VLAN

If a system goes into standby mode, and a directed packet is sent to the IP address of the removed VLAN, the system will wake-up. This occurs because a directed packet bypasses VLAN filtering.

Intel Adapters ignore consecutive Wake Up signals while transitioning into standby mode

While sending a system into standby, occasionally a wake up packet arrives before the system completes the transition into standby mode. When this happens, the system ignores consecutive wake up signals and remains in standby mode until manually powered up using the mouse, keyboard, or power button.

Low power link speed slower than expected

If you disable the "Reduce Power During Standby" setting and remove power from the system, your system may link at 10Mbps when power is restored, instead of 100Mbps or faster. The system will continue to link at 10Mbps until the operating system is loaded. This setting will be restored when the OS loads.

System Wakes Unexpectedly

On a driver only install, if you uncheck the "Allow this device to bring the computer out of standby" option on the Power Management tab, the adapter will still wake the system from Standby or Hibernate. The "Wake on Settings" option on the Advanced tab must also be set to Disabled.

11.4 Intel Ethernet 800 Series Known Issues

RDMA (Remote Direct Memory Access)

Devices based on the Intel® Ethernet 800 Series do not support RDMA when operating in multiport mode with more than 4 ports.

VMQ Support

On Intel(R) Ethernet 800 Series devices, Intel regularly tests up to 128 VMQs per NIC. Edge case testing shows that binding more than 512 VMQs per NIC may cause system instability or a system crash.

SR-IOV Support

On Intel(R) Ethernet 800 Series devices, Intel regularly tests up to 128 Virtual Functions (VFs) per NIC. Edge case testing shows that binding more than 128 VFs per NIC may cause system instability or a system crash.

A port in willing mode does not apply DCB settings received from the connected switch

When operating in multiport mode with more than 4 ports, Intel(R) Ethernet 800 Series devices support a maximum of 4 traffic classes (TCs) per port. If a port is in willing mode and connected to a switch that advertises more than 4 TCs, the port will not map to the advertised TCs. Instead it will map to the operating system's default TC configuration (usually 1 TC).

"Insufficient PCI-Express bandwidth available for device" on Intel Ethernet E823 Series

In Windows Server 2019 and 2022, devices based on the Intel Ethernet E823 Series may show the warning "Insufficient PCI-Express bandwidth available for device" in the event log. The E823 Series does not use a PCI-Express interface, and this appears to only be a cosmetic logging issue.

iDRAC Update Messaging

After updating the firmware on an Intel Ethernet 800 Series device using iDRAC, a Lifecycle Controller (LC) log message may appear that shows the NVM version followed by the ETrack ID, instead of showing the Dell family firmware version. For example, the log message could show: "SUP 0200 : The firmware 4.20 (0x80017848) update on the device NIC in slot 1 port 1 partition 1 will become effective after restarting the server." This messaging discrepancy is expected and can be ignored.

11.5 Intel Ethernet 700 Series Known Issues

Some devices based on the Intel® Ethernet Controller X710 report a subdevice ID of 0x0000 and may display a generic branding string. Port 0 reports the correct subvendor ID and displays the correct branding string.

Intel X710 based devices may maintain link on any and all ports as long as power is provided to the device, regardless of the device's or system's power state.

Unexpected IntelDCB errors in the Windows Application Event Log

After upgrading your X710 drivers, you may see several IntelDCB errors in the Windows Application Event Log. These errors are erroneous and can be ignored.

Lower than expected throughput on X710/XL710 based devices

If you have an X710 or XL710 based device installed in a four CPU socket system, receive and transmit traffic may be significantly lower than expected. Setting your interrupt rate to High may mitigate the issue.

Cable tests unavailable with Broadcom BCM84886 transceiver

The cable diagnostic tests may be unavailable if you have a Broadcom BCM84886 transceiver installed on a port. The transceiver does not support TDR diagnostics. See the BCM84886 datasheet (84886-DS103; October 27, 2017 revision). This issue affects the following devices:

- Intel® Ethernet Converged Network Adapter X710-T
- Intel® Ethernet 25G 2P XXV710 Mezz
- Intel® Ethernet 10G 2P X710-T2L-t Adapter
- Intel® Ethernet 10G 4P X710-T4L-t Adapter
- Intel® Ethernet 10G 2P X710-T2L-t OCP
- Intel® Ethernet 10G 4P X710-T4L-t OCP

Wake on LAN erroneously available in iDRAC/racadm

The Intel® Ethernet Converged Network Adapter X710-2 only supports WoL on port 1. If you view WoL status with iDRAC/racadm, WoL may erroneously appear as available on other ports and partitions.

Intel® Ethernet 10G 2P/4P X710-k bNDC does not have link and is not displayed in Windows Device Manager

If you install an Intel® Ethernet 10G 2P X710-k bNDC or an Intel® Ethernet 10G 4P X710-k bNDC onto a Dell PowerEdge M630/M830 blade server, and install that blade into an M1000e chassis, the bNDC may not have link and may display a yellow bang, or may not be displayed at all, in Windows Device Manager. This is limited to the 1.0 version of the M1000e Midplane.

100 Mbps reported as a supported speed on Intel Ethernet 700 Series devices

In Windows, Linux, and VMware ESXi operating systems, Intel Ethernet 700 Series adapters may report 100 Mbps as a supported speed but this speed is not supported by Dell.

Limitation of firmware downgrade from host OS

On devices based on the Intel Ethernet 700 Series and 550 Series, you cannot downgrade from the latest firmware to the previous firmware from the host OS. Use iDRAC with Lifecycle Controllers to downgrade the firmware.

11.6 Intel Ethernet 500 Series Known Issues

ETS Bandwidth Allocations Don't Match Settings

When Jumbo Frames is set to 9K with a 10GbE adapter, a 90%/10% ETS traffic split will not actually be attained on any particular port, despite settings being made on the DCB switch. When ETS is set to a 90%/10% split, an actual observed split of 70%/30% is more likely.

Link Loss on 10GbE Devices with Jumbo Frames Enabled

You must not lower Receive_Buffers or Transmit_Buffers below 256 if jumbo frames are enabled on an Intel® 10GbE Device. Doing so will cause loss of link.

When trying to identify the adapter, the Activity LED blinks and the Link LED is solid

If you use the Identify Adapter feature with the following adapters, the Activity LED blinks instead of the Link LED. The Link LED may display a solid green light for 10G ports even if a network link is not present.

- All Intel® Ethernet X550 10GbE devices
- Some Intel® Gigabit I350 LOM devices

Limitation of firmware downgrade from host OS

On devices based on the Intel Ethernet 700 Series and 550 Series, you cannot downgrade from the latest firmware to the previous firmware from the host OS. Use iDRAC with Lifecycle Controllers to downgrade the firmware.

11.7 Intel Ethernet 300 Series Known Issues

OS driver state shows as operational when the OS is not running

The OS driver for Intel Ethernet I350 Series devices shows as operational in iDRAC even when the system is not booted into the operating system. This is a known issue with the firmware.

11.7.1 Intel® Gigabit 4P I350-t Adapter Known Issues

Downshifting

When connecting to any Gigabit switch via a faulty CAT 5 cable where one pair is broken, the adapter does not downshift from 1 Gig to 100Mbps. For the adapter to downshift, it must identify two broken pairs in the cable.

System does not boot

Your system may run out of I/O resources and fail to boot if you install more than four quad port server adapters. Moving the adapters to different slots or rebalancing resources in the system BIOS may resolve the issue. This issue affects the following Adapters:

- Intel® Gigabit 4P I350-t Adapter

12. Regulatory Compliance Statements

12.1 FCC Class A Products

12.1.1 100 Gigabit Ethernet Products

- Intel® Ethernet 100G 2P E810-C-st Adapter
- Intel® Ethernet 100G 2P E810-C-stg Adapter

12.1.2 40 Gigabit Ethernet Products

- Intel® Ethernet 40G 2P XL710 QSFP+ rNDC
- Intel® Ethernet Converged Network Adapter XL710-Q2

12.1.3 25 Gigabit Ethernet Products

- Intel® Ethernet 25G 2P E810-XXV OCP
- Intel® Ethernet 25G 4P E810-XXV OCP
- Intel® Ethernet 25G 2P E810-XXV-k Mezz
- Intel® Ethernet 25G 4P E810-XXV-st Adapter
- Intel® Ethernet 25G 4P E810-XXV-stg Adapter
- Intel® Ethernet 25G 2P XXV710 Mezz
- Intel® Ethernet 25G 2P XXV710 Adapter

12.1.4 10 Gigabit Ethernet Products

- Intel® Ethernet 10G 2P X550-t Adapter
- Intel® Ethernet 10G 4P X550 rNDC
- Intel® Ethernet 10G 4P X550/I350 rNDC
- Intel® Ethernet 10G 4P X710-k bNDC
- Intel® Ethernet 10G 2P X710-k bNDC
- Intel® Ethernet Converged Network Adapter X710
- Intel® Ethernet Converged Network Adapter X710-T
- Intel® Ethernet 10G 4P X710/I350 rNDC
- Intel® Ethernet 10G 4P X710 SFP+ rNDC
- Intel® Ethernet Server Adapter X710-DA2 for OCP
- Intel® Ethernet 10G 2P X710 OCP
- Intel® Ethernet 10G 4P X710 OCP
- Intel® Ethernet 10G 2P X710-T2L-t OCP
- Intel® Ethernet 10G 4P X710-T4L-t OCP

12.1.5 Gigabit Ethernet Products

- Intel® Ethernet 1G 4P I350-t OCP
- Intel® Gigabit 4P X550/I350 rNDC
- Intel® Gigabit 4P I350-t rNDC
- Intel® Gigabit 4P I350-t Mezz
- Intel® Gigabit 4P X710/I350 rNDC
- Intel® Gigabit 4P I350 bNDC

12.2 FCC Class B Products

12.2.1 100 Gigabit Ethernet Products

- Intel® Ethernet 100G 2P E810-C Adapter

12.2.2 25 Gigabit Ethernet Products

- Intel® Ethernet 25G 2P E810-XXV Adapter
- Intel® Ethernet 25G 4P E810-XXV Adapter

12.2.3 10 Gigabit Ethernet Products

- Intel® Ethernet 10G 2P X710-T2L-t Adapter
- Intel® Ethernet 10G 4P X710-T4L-t Adapter

12.2.4 Gigabit Ethernet Products

- Intel® Gigabit 2P I350-t Adapter
- Intel® Gigabit 4P I350-t Adapter

12.3 Safety Compliance

The following safety standards apply to all products listed above:

- UL 60950-1, 2nd Edition, 2011-12-19 (Information Technology Equipment - Safety - Part 1: General Requirements)
- UL 62368-1 2nd Edition (Information Technology Equipment - Safety requirements)
- CSA C22.2 No. 60950-1-07, 2nd Edition, 2011-12 (Information Technology Equipment - Safety - Part 1: General Requirements)
- CAN/CSA C22.2 European Group Differences and National Differences according to 62368-1-14 - Audio/video, information and communication technology equipment - Part 1: Safety requirements
- EN 60950-1:2006/A11:2009/A1:2010/A12:2011 (European Union)
- IEC 60950-1:2005 (2nd Edition); Am 1:2009 (International)
- EU LVD Directive 2006/95/EC

12.4 EMC Compliance

The following standards may apply.

12.4.1 Class A Products

- FCC Part 15 – Radiated & Conducted Emissions (USA)
- CAN ICES-3(A)/NMB-3(A) – Radiated & Conducted Emissions (Canada)
- CISPR 22 – Radiated & Conducted Emissions (International)
- EN55022: 2010 – Radiated & Conducted Emissions (European Union)
- EN55024: 2010 +A1:2001+A2:2003 – Immunity (European Union)
- EN55032: 2015 Class A Radiated and Conducted Emissions requirements (European Union)
- EMC Directive 2004/108/EC (European Union)
- VCCI (Class A)– Radiated & Conducted Emissions (Japan)
- CNS13438 – Radiated & Conducted Emissions (Taiwan)
- AS/NZS CISPR 22:2009 + A1:2010 Class A and CISPR 32:2015 for Radiated and Conducted Emissions requirements (Australia/New Zealand)
- NRR No. 2012-13 (2012.06.28), NRR Notice No. 2012-14 (2012.06.28) (Korea)

12.4.2 Class B Products

- FCC Part 15 (Class B) – Radiated & Conducted Emissions (USA)
- CAN ICES-3(B)/NMB-3(B) – Radiated & Conducted Emissions (Canada)
- CISPR 22 – Radiated & Conducted Emissions (International)
- EN55022: 2010 – Radiated & Conducted Emissions (European Union)
- EN55024: 2010 – Immunity (European Union)
- EN55032: 2015 Class B Radiated and Conducted Emissions requirements (European Union)
- EMC Directive 2004/108/EC (European Union)
- VCCI (Class B)– Radiated & Conducted Emissions (Japan) (excluding optics)
- CNS13438 (Class B)-2006 – Radiated & Conducted Emissions (Taiwan) (excluding optics)
- AS/NZS CISPR 22:2009 + A1:2010 Class B and CISPR 32:2015 for Radiated and Conducted Emissions requirements (Australia/New Zealand)
- KN22; KN24 – Korean emissions and immunity
- NRR No. 2012-13 (2012.06.28), NRR Notice No. 2012-14 (2012.06.28) (Korea)

12.5 Hazardous Substances Compliance

The following standards may apply:

- EU REACH directive
- EU WEEE directive
- EU RoHS directive
- China RoHS directive
- BSMI CNS15663: Taiwan RoHS

12.6 Regulatory Compliance Markings

When required, these products are provided with the following Product Certification Markings:

- UL Recognition Mark for USA and Canada
- CE Mark
- EU WEEE Logo
- FCC markings
- VCCI marking
- Australian C-Tick Mark
- Korea MSIP mark
- Taiwan BSMI mark
- People's Republic of China "EFUP" mark

12.7 FCC Class A User Information

The Class A products listed above comply with Part 15 of the FCC Rules. Operation is subject to the following two conditions:

1. This device may not cause harmful interference.
2. This device must accept any interference received, including interference that may cause undesired operation.



NOTE: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.



CAUTION: If the device is changed or modified without permission from Intel, the user may void his or her authority to operate the equipment.

12.7.1 Canadian Compliance (Industry Canada)

CAN ICES-3(A)/NMB-3(A)

12.7.2 VCCI Class A Statement

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

VCCI-A

12.7.3 BSMI Class A Statement

警告使用者:

此為甲類資訊技術設備，於居住環境中使用時，可能會造成射頻擾動，在此種情況下，使用者會被要求採取某些適當的對策。

12.7.4 KCC Notice Class A (Republic of Korea Only)

<p>A급 기기 (업무용 방송통신기기)</p> <p>CLASS A device (commercial broadcasting and communication equipment)</p>	<p>이 기기는 업무용(A급)으로 전자파적합등록을 한 기기이오니 판매자 또는 사용자는 이 점을 주의하시기 바라며, 가정외의 지역에서 사용하는 것을 목적으로 합니다.</p> <p>This device has been approved by EMC registration. Distributors or users pay attention to this point. This device is usually aimed to be used in other area except at home.</p>
--	--

12.7.5 BSMI Class A Notice (Taiwan)

警告使用者:

此為甲類資訊技術設備，於居住環境中使用時，可能會造成射頻擾動，在此種情況下，使用者會被要求採取某些適當的對策。

12.8 FCC Class B User Information

This equipment has been tested and found to comply with the limits for a Class B digital device pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation.

If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.



CAUTION: If the device is changed or modified without permission from Intel, the user may void his or her authority to operate the equipment.



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

12.8.1 Electromagnetic Compatibility Notices

12.8.1.1 FCC Declaration of Conformity Statement

The following products have been tested to Comply with FCC Standards for Home or Office Use:

PRO/1000 PT, PRO/1000 GT, Gigabit PT, I210-T1, I340-T2/T4, and I350-T2/T4.

12.8.1.2 Canadian Compliance (Industry Canada)

CAN ICES-3 (B)/NMB-3 (B)

12.8.2 VCCI Class B Statement (Japan)

この装置は、クラスB 情報技術装置です。この装置は、家庭環境で使用することを目的としていますが、この装置がラジオやテレビジョン受信機に近接して使用されると、受信障害を引き起こすことがあります。

取扱説明書に従って正しい取り扱いをして下さい。

VCCI-B

12.8.3 KCC Notice Class B (Republic of Korea Only)

<p>B급 기기 (가정용 방송통신기기)</p> <p>CLASS B device residential broadcasting and communication equipment</p>	<p>이 기기는 가정용(B급)으로 전자파적합등록을 한 기기로서 주로 가정에서 사용하는 것을 목적으로 하며, 모든 지역에서 사용할 수 있습니다.</p> <p>This device has been approved by EMC Registration and is usually aimed to be used in a residential area so that it can be used in all other location as well as at home.</p>
---	---

12.9 EU WEEE Logo



12.10 Manufacturer Declaration European Community



12.10.1 Manufacturer Declaration

Intel Corporation declares that the equipment described in this document is in conformance with the requirements of the European Council Directive listed below:

- Low Voltage Directive 2006/95/EC
- EMC Directive 2004/108/EC
- RoHS Directive 2011/65/EU

These products follow the provisions of the European Directive 1999/5/EC.

Dette produkt er i overensstemmelse med det europæiske direktiv 1999/5/EC.

Dit product is in navolging van de bepalingen van Europees Directief 1999/5/EC.

Tämä tuote noudattaa EU-direktiivin 1999/5/EC määräyksiä.

Ce produit est conforme aux exigences de la Directive Européenne 1999/5/EC.

Dieses Produkt entspricht den Bestimmungen der Europäischen Richtlinie 1999/5/EC.

Þessi vara stenst reglugerð Evrópska Efnahags Bandalagsins númer 1999/5/EC.

Questo prodotto è conforme alla Direttiva Europea 1999/5/EC.

Dette produktet er i henhold til bestemmelsene i det europeiske direktivet 1999/5/EC.

Este produto cumpre com as normas da Diretiva Europeia 1999/5/EC.

Este producto cumple con las normas del Directivo Europeo 1999/5/EC.

Denna produkt har tillverkats i enlighet med EG-direktiv 1999/5/EC.

This declaration is based upon compliance of the Class A products listed above to the following standards:

EN 55022:2010 (CISPR 22 Class A) RF Emissions Control.

EN 55024:2010 (CISPR 24) Immunity to Electromagnetic Disturbance.

EN 60950-1:2006/A11:2009/A1:2010/A12:2011 Information Technology Equipment- Safety-Part 1: General Requirements.

EN 50581:2012 - Technical documentation for the assessment of electrical and electronic products with respect to the restriction of hazardous substances.

This declaration is based upon compliance of the Class B products listed above to the following standards:

EN 55022:2010 (CISPR 22 Class B) RF Emissions Control.

EN 55024:2010 (CISPR 24) Immunity to Electromagnetic Disturbance.

EN 60950-1:2006/A11:2009/A1:2010/A12:2011 Information Technology Equipment- Safety-Part 1: General Requirements.

EN 50581:2012 - Technical documentation for the assessment of electrical and electronic products with respect to the restriction of hazardous substances.



WARNING: In a domestic environment, Class A products may cause radio interference, in which case the user may be required to take adequate measures.

Responsible Party

Intel Corporation, Mailstop JF3-446
5200 N.E. Elam Young Parkway
Hillsboro, OR 97124-6497
Phone 1-800-628-8686

12.11 China RoHS Declaration

关于符合中国《电子信息产品污染控制管理办法》的声明
**Management Methods on Control of Pollution From
 Electronic Information Products
 (China RoHS declaration)**

产品中有毒有害物质的名称及含量

部件名称	有毒有害物质或元素					
	铅 (Pb)	汞 (Hg)	镉 (Cd)	六价铬 (Cr(VI))	多溴联苯 (PBB)	多溴二苯醚 (PBDE)
印刷板组件	X	○	○	○	○	○
<p>○：表示该有毒有害物质在该部件所有均质材料中的含量均在 SJ/T 11363-2006 标准规定的限量要求以下。</p> <p>X：表示该有毒有害物质至少在该部件的某一均质材料中的含量超出 SJ/T 11363-2006 标准规定的限量要求。</p>						

12.12 Class 1 Laser Products

Server adapters listed above may contain laser devices for communication use. These devices are compliant with the requirements for Class 1 Laser Products and are safe in the intended use. In normal operation the output of these laser devices does not exceed the exposure limit of the eye and cannot cause harm.

For continued safe operation in case of an abnormal circumstance, always have the provided laser connector cover in place or a compatible fiber optics cable properly connected when power is available to the product.

The Laser device must be factory serviced ONLY by the responsible manufacturer! NO adjustments, service or maintenance is to be performed otherwise.



CAUTION: Use of controls or adjustments or performance of procedures other than those specified herein may result in hazardous radiation exposure.

These Class 1 Laser devices:

Comply with FDA/CDRH per CFR21, subchapter J.
 Comply with IEC 60825-1:2007

12.13 End-of-Life / Product Recycling

Product recycling and end-of-life take-back systems and requirements vary by country.



Contact the retailer or distributor of this product for information about product recycling and/or take-back.

13. Specifications

13.1 Device Specifications

See "Indicator Lights" on page 238 for descriptions of LED behavior.

13.1.1 Intel® 100 Gigabit Network Adapter Specifications

Feature	Intel® Ethernet 100G 2P E810-C Adapter	Intel® Ethernet 100G 2P E810-C-st Adapter Intel® Ethernet 100G 2P E810-C-stg Adapter
Bus Connector	PCI Express 4.0	PCI Express 4.0
Bus Speed	x16	x16
Transmission Mode/Connector	QSFP28	QSFP28
Cabling	25GBase-CR, Twinax DAC	25GBase-CR, Twinax DAC
Power Requirements	25 W maximum @ +12 V 1.35 W maximum @ +3.3 V	42 W @ +12 V 1 W @ +3.3 V
Dimensions (excluding bracket)	2.70 x 6.67 in 6.87 x 16.93 cm	4.37 x 6.67 in 11.10 x 16.94 cm
Operating Temperature	23 - 131 deg. F (-5 - 55 deg. C)	32 - 131 deg. F with required airflow (passive heat sink) (0-55 deg. C with required airflow (passive heat sink))
MTBF at 55°C	199 years	-stg Adapter: 44 years -st Adapter: 103 years
Available Speeds	100 Gbps/50 Gbps/25 Gbps/10 Gbps/1 Gbps Note: This device supports multiple port options that can be configured with the Ethernet Port Configuration Tool. Some speeds are only supported in specific port configurations.	100 Gbps/10 Gbps
Duplex Modes	Full only	Full only
Standards Conformance	PCI Express 4.0 SFF-8419 IEEE 802.3	PCI Express 4.0 IEEE Standard for Ethernet - 802.3-2015 IEEE 802.3by SFF-8402, Revision 1.1 SFF-8431, Revision 2.1 ITU-T G.703
Regulatory and Safety	Safety Compliance <ul style="list-style-type: none"> UL 62368-1, 2nd Ed, 2014-12-01 CAN/CSA C22.2 No. 62368-1-14, 2nd Ed 	Safety Compliance <ul style="list-style-type: none"> UL 62368-1, 2nd Ed, 2014-12-01 (Audio/video, information and communication technology equipment)

Feature	Intel® Ethernet 100G 2P E810-C Adapter	Intel® Ethernet 100G 2P E810-C-st Adapter Intel® Ethernet 100G 2P E810-C-stg Adapter
	(USA/Canada) EMC Compliance <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • EN 55032:2013 - Radiated & Conducted Emissions (European Union) • EN 55024:2010 Immunity Requirements (European Union) • KN32 Radiated and Conducted Emissions, KN35 Immunity (Korea) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI-CISPR 32:2016 - Radiated & Conducted Emissions (Japan) • BSMI CNS13438: 2006 - Radiated & Conducted Emissions (Taiwan) • REACH, WEEE, RoHS Directives (European Union) • RoHS Directive (China) 	Part 1: Safety requirements) <ul style="list-style-type: none"> • CAN/CSA C22.2 No. 62368-1-14, 2nd Ed (Audio/video, information and communication technology equipment Part 1: Safety requirements) EMC Compliance <ul style="list-style-type: none"> • FCC, 47 CFR Part 15, Class A digital device (USA) • ICES-003:2016, Class A (Canada) • EN 55032:2012/AC:2013 • EN 55035: 2017 Immunity requirements for European Union (EU) • KS C 9832 Radiated and Conducted (Korea) • Emissions KS C 9835 Immunity (Korea) • AS/NZS CISPR 32:2015 Class A (Australia/New Zealand) • CE directives for the CE Mark and the Radio Electronics Directive RED • VCCI-CISPR 32: 2016 Class A (Japan) • BSMI CNS13438: 2006 (complete) Class A Radiated and Conducted Emissions requirements (Taiwan) • REACH, WEEE, RoHS Directives (European Union) • RoHS Directive (China) • UK S.I. 1101, 1091 and 3032 directives (United Kingdom)

13.1.2 Intel® 40 Gigabit Network Adapter Specifications

Feature	Intel® Ethernet Converged Network Adapter XL710-Q2
Bus Connector	PCI Express 3.0
Bus Speed	x8
Transmission Mode/Connector	QSFP+
Cabling	40GBase-SR4, Twinax DAC (7m max)
Power Requirements	6.5 W Maximum @ +12 V
Dimensions (excluding bracket)	5.21 x 2.71 in 13.3 x 6.9 cm
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)

Feature	Intel® Ethernet Converged Network Adapter XL710-Q2
MTBF at 55°C	159 years
Available Speeds	40 Gbps
Duplex Modes	Full only
Standards Conformance	IEEE 802.3ba SFF-8436 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) • EN 60 950 (European Union) • IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 22 - Radiated & Conducted Emissions (International) • EN55022-1998 - Radiated & Conducted Emissions (European Union) • EN55024 - 1998 - (Immunity) (European Union) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea)

13.1.3 Intel® 40GbE Network Daughter Cards (NDC) Specifications

Feature	Intel® Ethernet 40G 2P XL710 QSFP+ rNDC
Bus Connector	PCI Express 3.0
Bus Speed	x8
Transmission Mode/Connector	QSFP+
Cabling	40GBase-SR4, Twinax DAC (7m max)
Power Requirements	6.2 W Maximum @ +12 V
Dimensions (excluding bracket)	3.66 x6.081 in 9.3 x 15.5 cm
Operating Temperature	32 - 140 deg. F (0 - 60 deg. C)
MTBF at 55°C	112 years
Available Speeds	40 Gbps/10 Gbps

Feature	Intel® Ethernet 40G 2P XL710 QSFP+ rNDC
Duplex Modes	Full only
Standards Conformance	IEEE 802.3ba SFF-8436 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) • EN 60 950 (European Union) • IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 22 - Radiated & Conducted Emissions (International) • EN55022-1998 - Radiated & Conducted Emissions (European Union) • EN55024 - 1998 - (Immunity) (European Union) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea)

13.1.4 Intel® 25 Gigabit Network Adapter Specifications

Feature	Intel® Ethernet 25G 4P E810-XXV-st Adapter Intel® Ethernet 25G 4P E810-XXV-stg Adapter
Bus Connector	PCI Express 4.0
Bus Speed	x16
Transmission Mode/Connector	SFP28
Cabling	25GBase-CR, Twinax DAC
Power Requirements	34 W Max @ +12 V
Dimensions (excluding bracket)	4.37 in x 6.67 in 11.1 cm x 16.93 cm
Operating Temperature	32 - 149 deg. F with required airflow (passive heat sink) (0 - 65 deg. C with required airflow (passive heat sink))
MTBF at 55°C	-stg Adapter: 43 years -st Adapter: 1650 years
Available Speeds	25 Gbps/10 Gbps/1 Gbps

Feature	Intel® Ethernet 25G 4P E810-XXV-st Adapter Intel® Ethernet 25G 4P E810-XXV-stg Adapter
Duplex Modes	Full only
Standards Conformance	PCI Express 4.0 SFF-8402, SFF-8431 IEEE 802.3, 802.3by
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> UL 62368-1, 2nd Ed, 2014-12-01 (USA) CAN/CSA C22.2 No. 62368-1-14, 2nd Ed (Canada) <p>EMC Compliance</p> <ul style="list-style-type: none"> FCC, 47 CFR Part 15, Class A digital device (USA) ICES-003:2016, Class A (Canada) EN 55032:2012/AC:2013 (European Union) EN 55035: 2017 Immunity requirements for European Union (EU) KS C 9832 Radiated and Conducted (Korea) Emissions KS C 9835 Immunity (Korea) AS/NZS CISPR 32:2015 Class A (Australia/New Zealand) VCCI-CISPR 32: 2016 Class A (Japan) BSMI CNS13438: 2006 (complete) Class A Radiated and Conducted Emissions requirements (Taiwan) <p>Other Compliance</p> <ul style="list-style-type: none"> CE Mark and the Radio Electronics Directive (RED) directives (European Union) REACH, WEEE, and RoHS directives (European Union) RoHS directive (China) UK S.I. 1101, 1091 and 3032 directives (United Kingdom)

Feature	Intel® Ethernet 25G 2P E810-XXV Adapter	Intel® Ethernet 25G 2P E810-XXV OCP	Intel® Ethernet 25G 4P E810-XXV Adapter
Bus Connector	PCI Express 4.0 PCI Express 3.0	OCP NIC 3.0	PCI Express 4.0
Bus Speed	x8	x8	x16
Transmission Mode/Connector	SFP28	SFP28	SFP28
Cabling	25GBase-CR, Twinax DAC (3m max)	25GBase-CR, Twinax DAC (3m max)	25GBase-CR, Twinax DAC
Power Requirements	25 W maximum @ +12 V 3.63 W maximum @ +3.3 V	25 W maximum @ +12 V 3.63 W maximum @ +3.3 V	25 W maximum @ +12 V 3.63 W maximum @ +3.3 V
Dimensions (excluding bracket)	2.54 in. x 6.6 in. 6.44 cm x 16.76 cm	4.53 in. x 2.99 in. 11.5 cm x 7.6 cm	4.37 in. x 6.67 in. 11.1 cm x 16.93 cm
Operating Temperature	32 - 140°F (0 - 60°C)	41 - 149°F (5 - 65°C)	23 - 131°F (-5 - 55°C)

Feature	Intel® Ethernet 25G 2P E810-XXV Adapter	Intel® Ethernet 25G 2P E810-XXV OCP	Intel® Ethernet 25G 4P E810-XXV Adapter
MTBF at 55°C	271 years	266 years	189 years
Available Speeds	25 Gbps/10 Gbps/1 Gbps	25 Gbps/10 Gbps/1 Gbps	25 Gbps/10 Gbps/1 Gbps/100 Mbps
Duplex Modes	Full only	Full only	Full only
Standards Conformance	PCI Express 4.0 SFF-8419 IEEE 802.3	PCI Express 4.0 SFF-8431 IEEE 802.3 OCP NIC 3.0	PCI Express 4.0 SFF-8419 IEEE 802.3
Regulatory and Safety	EMC Compliance <ul style="list-style-type: none"> FCC Part 15 - Radiated & Conducted Emissions (USA) ICES-003 - Radiated & Conducted Emissions (Canada) CISPR 22 - Radiated & Conducted Emissions (International) EN55032-2015- Radiated & Conducted Emissions (European Union) EN55024 - 2010- (Immunity) (European Union) REACH, WEEE, RoHS Directives (European Union) VCCI - Radiated & Conducted Emissions (Japan) CNS13438 - Radiated & Conducted Emissions (Taiwan) AS/NZS CISPR - Radiated & Conducted Emissions (Australia/New Zealand) KN22 -Radiated & Conducted Emissions (Korea) RoHS (China) 		

Feature	Intel® Ethernet 25G 4P E810-XXV OCP	Intel® Ethernet 25G 2P XXV710 Adapter
Bus Connector	OCP NIC 3.0	PCI Express 3.0
Bus Speed	x16	x8
Transmission Mode/Connector	SFP28	SFP28
Cabling	100GBase-CR4 CA-25G-N (Breakout) CA-25G-S (Breakout) CA-25G-L (Breakout) 100GBASE-SR4	25GBase-CR, Twinax DAC (3m max)
Power Requirements	35 W maximum @ +12 V 3.63 W maximum @ +3.3 V	6.5 W maximum @ +12 V
Dimensions (excluding bracket)	4.53 in. x 2.99 in 11.5 cm x 7.6 cm	2.70 x 2.02 in 6.86 x 5.12 cm
Operating Temperature	41 - 149 deg. F (5 - 65 deg. C)	32 - 131 deg. F (0 - 55 deg. C)

Feature	Intel® Ethernet 25G 4P E810-XXV OCP	Intel® Ethernet 25G 2P XXV710 Adapter
MTBF at 55°C	193 years	239 years
Available Speeds	25 Gbps/10 Gbps/1 Gbps	25 Gbps/10 Gbps/1 Gbps
Duplex Modes	Full only	Full only
Standards Conformance	PCI Express 4.0 SFF-8431 IEEE 802.3 OCP NIC 3.0	IEEE 802.3-2015 SFF-8431 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> UL/CSA 60950-1-07 2nd Edition EN 60 950 (European Union) IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> FCC Part 15 - Radiated & Conducted Emissions (USA) ICES-003 - Radiated & Conducted Emissions (Canada) CISPR 22 - Radiated & Conducted Emissions (International) EN55032-2015- Radiated & Conducted Emissions (European Union) EN55024 - 2010- (Immunity) (European Union) REACH, WEEE, RoHS Directives (European Union) VCCI - Radiated & Conducted Emissions (Japan) CNS13438 - Radiated & Conducted Emissions (Taiwan) AS/NZS CISPR - Radiated & Conducted Emissions (Australia/New Zealand) KN32 -Radiated & Conducted Emissions (Korea) KN35 - (Immunity) (Korea) RoHS 	

13.1.5 Intel® 25 Gigabit Network Mezzanine Card Specifications

Feature	Intel® Ethernet 25G 2P E810-XXV-k Mezz	Intel® Ethernet 25G 2P XXV710 Mezz
Bus Connector	PCI Express 3.0	PCI Express 3.0
Bus Speed	x8	x8
Transmission Mode/Connector	10GBase-KR, 25GBase-KR	SFP28
Cabling	n/a	25GBase-CR, Twinax DAC (3m max)
Power Requirements	10.2 W @ +12V D0 5.7 W @ +12V D3	9.78 W @ +12V
Dimensions (excluding bracket)	3.78 x 3.15 in. 9.60 x 8.001 cm	3.78 x 3.15 in. 9.60 x 8.001 cm
Operating Temperature	86 - 149 deg. F (30 to 65 deg. C)	105° F max

Feature	Intel® Ethernet 25G 2P E810-XXV-k Mezz	Intel® Ethernet 25G 2P XXV710 Mezz
MTBF at 55°C	231 years	353 years
Available Speeds	25 Gbps/10 Gbps	25 Gbps/10 Gbps
Duplex Modes	Full only	Full only
Standards Conformance	IEEE 802.3 backplane standards	IEEE 802.3 backplane standards
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> EN IEC 62368-1:2020 + A11:2020 (Ed: 3) - Safety Audio/Video International EN IEC 62368-3:2020 - Safety Audio/Video International IEC 62368-1:2018 (Ed: 3) - Safety Audio/Video International IEC 62368-3:2017 - Safety Audio/Video International UL 62368-1:2020 (3rd Ed) - Safety Audio/Video CSA C22.2 No 62368-1:19 (3rd Ed) - Safety Audio/Video <p>EMC Compliance</p> <ul style="list-style-type: none"> AS/NZS CISPR 32 - EMC Emissions EN 55032:2015+A11:2020 - EMC Emissions EN 55035:2017/A11:2020 - EMC Immunity FCC Part 15, Subpart B (Class A) - EMC Emissions ICES-003 (Class A) - EMC Emissions KS C 9832:2019 - EMC Emissions KS C 9835:2019 - EMC Immunity VCCI 32-1:2016 (CISPR 32) - EMC Emissions 	<p>EMC Compliance</p> <ul style="list-style-type: none"> FCC Part 15 - Radiated & Conducted Emissions (USA) ICES-003 - Radiated & Conducted Emissions (Canada) CISPR 22 - Radiated & Conducted Emissions (International) EN55032-2015- Radiated & Conducted Emissions (European Union) EN55024 - 2010- (Immunity) (European Union) REACH, WEEE, RoHS Directives (European Union) VCCI - Radiated & Conducted Emissions (Japan) CNS13438 - Radiated & Conducted Emissions (Taiwan) AS/NZS CISPR - Radiated & Conducted Emissions (Australia/New Zealand) KN22 -Radiated & Conducted Emissions (Korea) RoHS (China)

13.1.6 Intel® 10 Gigabit Network Adapter Specifications

Feature	Intel® Ethernet 10G 2P X710 OCP Intel® Ethernet 10G 4P X710 OCP	Intel® Ethernet 10G 2P X710-T2L-t OCP	Intel® Ethernet 10G 4P X710-T4L-t OCP
Bus Connector	PCI Express 3.0	OCP NIC 3.0	OCP NIC 3.0
Bus Speed	x8	x8 PCI Express v3.0	x8 PCI Express v3.0
Transmission Mode/Connector	10G/SFP+	RJ45 BASE-T Connector	RJ45 BASE-T Connector
Cabling	10GBASE-SR	10GBASE-T: CAT6A (100m)	10GBASE-T: CAT6A (100m)

Feature	Intel® Ethernet 10G 2P X710 OCP Intel® Ethernet 10G 4P X710 OCP	Intel® Ethernet 10G 2P X710-T2L-t OCP	Intel® Ethernet 10G 4P X710-T4L-t OCP
	10GBASE-LR SFP+ Direct Attach Cables	max), CAT6 (55m max) 1000BASE-T: CAT6A, CAT6, CAT5e (100m max)	max), CAT6 (55m max) 1000BASE-T: CAT6A, CAT6, CAT5e (100m max)
Power Requirements	6 W max, optics not included	9.0 W Maximum @ +12 V	16.3 W Maximum @ +12 V
Dimensions (excluding bracket)	Standard OCP3.0 Small Form Factor 2.99 x 4.53 in (7.6 x 11.5 cm)	Standard OCP3.0 Small Form Factor 2.99 x 4.53 in (7.6 x 11.5 cm)	Standard OCP3.0 Small Form Factor 2.99 x 4.53 in (7.6 x 11.5 cm)
Operating Temperature	23 - 149 deg. F (-5 - 65 deg. C)	32 - 131 deg. F (0 - 55 deg. C)	32 - 131 deg. F (0 - 55 deg. C)
MTBF at 55°C	376	376	223
Available Speeds	10 Gbps/1 Gbps	10 Gbps/1 Gbps	10 Gbps/1 Gbps
Duplex Modes	Full only	Full only	Full only
Standards Conformance	PCI Express 3.0 SFF-8431 IEEE 802.3ae OCP NIC 3.0	IEEE 802.3 PCI Express 3.0	IEEE 802.3 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL/ CSA 62368-1: 2014 2nd Edition • EN 62368 (European Union) • IEC 62368 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 32 - Radiated & Conducted Emissions (International) • EN55032-2015 - Radiated & Conducted Emissions (European Union) • EN55035: 2017 - (Immunity) (European Union) • REACH, WEEE, RoHS Directives (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZ CISPR - Radiated & Conducted Emissions (Australia/New Zealand) • KN32 -Radiated & Conducted Emissions (Korea)KN35 - (Immunity) (Korea) 		

Feature	Intel® Ethernet 10G 2P X710-T2L-t Adapter	Intel® Ethernet 10G 4P X710-T4L-t Adapter
Bus Connector	PCI Express 3.0	PCI Express 3.0
Bus Speed	x8	x8
Transmission Mode/Connector	RJ45 BASE-T Connector	RJ45 BASE-T Connector
Cabling	10GBASE-T: CAT6A (100m max), CAT6 (55m max) 1000BASE-T: CAT6A, CAT6, CAT5e (100m max)	10GBASE-T: CAT6A (100m max), CAT6 (55m max) 1000BASE-T: CAT6A, CAT6, CAT5e (100m max)
Power Requirements	9.6 W Maximum @ +12 V	14.2 W Maximum @ +12 V
Dimensions (excluding bracket)	2.70 x 6.74 in (6.86 x 17.12 cm)	2.70 x 6.63 in (6.86 x 16.84 cm)
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)	32 - 131 deg. F (0 - 55 deg. C)
MTBF at 55°C	356	237
Available Speeds	10 Gbps/1 Gbps	10 Gbps/1 Gbps
Duplex Modes	Full only	Full only
Standards Conformance	IEEE 802.3 PCI Express 3.0	IEEE 802.3 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL/ CSA 62368-1: 2014 2nd Edition • EN 62368 (European Union) • IEC 62368 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 32 - Radiated & Conducted Emissions (International) • EN55032-2015 - Radiated & Conducted Emissions (European Union) • EN55035: 2017 - (Immunity) (European Union) • REACH, WEEE, RoHS Directives (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZ CISPR - Radiated & Conducted Emissions (Australia/New Zealand) • KN32 -Radiated & Conducted Emissions (Korea)KN35 - (Immunity) (Korea) 	

Feature	Intel® Ethernet Converged Network Adapter X710-T	Intel® Ethernet Converged Network Adapter X710	Intel® Ethernet Server Adapter X710-DA2 for OCP
Bus Connector	PCI Express 3.0	PCI Express 3.0	PCI Express 3.0
Bus Speed	x8	x8	x8
Transmission Mode/Connector	10GBase-T/RJ-45	SFP+	SFP+
Cabling	10GBase-T (Category 6A)	Twinax 10GBase-SR/LR	Direct Attach 10GBASE-SR
Power Requirements	8.53 W (idle) @ 12V Main	6.7 Watts (maximum) @ 12 V	3.08 Watts (max) @ 5V Main
Dimensions (excluding bracket)	6.578 x 4.372 in 16.708 x 11.107 cm	6.578 x 4.372 in 16.708 x 11.107 cm	2.67 x 4.59 in 6.78 x 11.658 cm
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)	41 - 131 deg. F (5 - 55 deg. C)	32 - 131 deg. F (0 - 55 deg. C)
MTBF	493 years	491 years	1276 years
Available Speeds	10 Gbps/1 Gbps	10 Gbps/1 Gbps	10 Gbps/1 Gbps
Duplex Modes	Full only	Full Only	Full only
Standards Conformance	PCI Express 3.0 SFF-8431 IEEE 802.3z IEEE 802.3ae	PCI Express 3.0 SFF-8431 IEEE 802.3z IEEE 802.3ae	PCI Express 3.0 SFF-8431 IEEE 802.3z IEEE 802.3ae
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) • EN 60 950 (European Union) • IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 22 - Radiated & Conducted Emissions (International) • EN55022-1998 - Radiated & Conducted Emissions (European Union) • EN55024 - 1998 - (Immunity) (European Union) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea) 		

Feature	Intel® Ethernet 10G 2P X550-t Adapter
Bus Connector	PCI Express 3.0
Bus Speed	x8
Transmission Mode/Connector	10GBase-T/RJ-45
Cabling	10GBase-T (Category 6A)
Power Requirements	13W Maximum @ +12 V
Dimensions (excluding bracket)	5.13 x 2.7 in 13.0 x 6.9 cm
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)
MTBF at 55°C	127 years
Available Speeds	10 Gbps/1 Gbps
Duplex Modes	Full only
Standards Conformance	IEEE 802.1p IEEE 802.1Q IEEE 802.3an IEEE 802.3ac IEEE 802.3ad IEEE 802.3x ACPI v1.0 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) • EN 60 950 (European Union) • IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 22 - Radiated & Conducted Emissions (International) • EN55022-1998 - Radiated & Conducted Emissions (European Union) • EN55024 - 1998 - (Immunity) (European Union) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea)

13.1.7 Intel® 10GbE Network Daughter Cards (NDC) Specifications

Feature	Intel® Ethernet 10G 4P X550/I350 rNDC	Intel® Ethernet 10G 4P X550 rNDC
Bus Connector	PCI Express 3.0	PCI Express 3.0
Bus Speed	x8	x8
Transmission Mode/Connector	Twisted copper/RJ-45	Twisted copper/RJ-45
Cabling	Cat 6A (10 Gbps)/Cat 5e (1 Gbps)	Cat 6A
Power Requirements	15.39 Watts (max) @12 V	33.6 Watts (maximum) @ 12 V
Dimensions	4.34 x 4.012 in 11.04 x 10.19 cm	4.37 x 5.86 in 11.10 x 14.883 cm
Operating Temperature	60° F	60° F
MTBF at 55°C	445	436
Available Speeds	10 Gbps/1 Gbps	10 Gbps/1 Gbps
Duplex Modes	Full only	Full only
Standards Conformance	IEEE 802.1p IEEE 802.1Q IEEE 802.3ac IEEE 802.3ad IEEE 802.3ae IEEE 802.3x ACPI v1.0 PCI Express 3.0	IEEE 802.1p IEEE 802.1Q IEEE 802.3ac IEEE 802.3ad IEEE 802.3ae IEEE 802.3x ACPI v1.0 PCI Express 3.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) • EN 60 950 (European Union) • IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 22 - Radiated & Conducted Emissions (International) • EN55022-1998 - Radiated & Conducted Emissions (European Union) • EN55024 - 1998 - (Immunity) (European Union) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea) 	

Feature	Intel® Ethernet 10G 2P X710-k bNDC Intel® Ethernet 10G 4P X710-k bNDC	Intel® Ethernet 10G 4P X710/I350 rNDC	Intel® Ethernet 10G 4P X710 SFP+ rNDC
Bus Connector	Dell bNDC	PCI Express 3.0	PCI Express 3.0
Bus Speed	x8	x8	x8
Transmission Mode/Connector	KX/KR	SFP+	SFP+
Cabling	Backplane	Twinax 10GBase-SR/LR	Twinax 10GBase-SR/LR
Power Requirements	3.3 Watts @ 3.3 V (AUX), 12.6 Watts @ 12 V (AUX)	10.7 Watts Maximum @ +12 V	9.5 Watts Maximum @ +12 V
Dimensions	3.000x2.449 in 7.62x6.220cm	4.331x3.661 in 11.0x9.298 cm	4.331x3.661 in 11.0x9.298 cm
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)	32 - 131 deg. F (0 - 55 deg. C)	32 - 131 deg. F (0 - 55 deg. C)
MTBF at 55°C	828 years	108 years	505 years
Available Speeds	10 Gbps/1 Gbps	10 Gbps/1 Gbps	10 Gbps/1 Gbps
Duplex Modes	Full only	Full only	Full only
Standards Conformance	PCI Express 3.0 IEEE 802.3ap	PCI Express 3.0 SFF-8431 IEEE 802.3z IEEE 802.3ae	PCI Express 3.0 SFF-8431 IEEE 802.3z IEEE 802.3ae
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) EN 60 950 (European Union) IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> FCC Part 15 - Radiated & Conducted Emissions (USA) ICES-003 - Radiated & Conducted Emissions (Canada) CISPR 22 - Radiated & Conducted Emissions (International) EN55022-1998 - Radiated & Conducted Emissions (European Union) EN55024 - 1998 - (Immunity) (European Union) CE - EMC Directive (89/336/EEC) (European Union) VCCI - Radiated & Conducted Emissions (Japan) CNS13438 - Radiated & Conducted Emissions (Taiwan) AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea) 		

13.1.8 Intel® Gigabit Network Adapter Specifications

Feature	Intel® Ethernet 1G 4P I350-t OCP	Intel® Gigabit 2P I350-t Adapter Intel® Gigabit 4P I350-t Adapter
Bus Connector	PCI Express 2.1	PCI Express 2.0
Bus Speed	x4	x4
Transmission Mode/Connector	1GBase-T/RJ-45	Twisted copper/RJ-45
Cabling	Cat 5e	1000Base-T (Category 3 or Category 5)
Power Requirements	25.2 W Maximum @ +12 V	2P Adapter: 4.8 Watts @ 12 V 4P Adapter: 6.0 Watts @ 12 V
Dimensions (excluding bracket)	Standard OCP3.0 Small Form Factor 2.99 x 4.53 in (7.6 x 11.5 cm)	5.3 x 2.7 in. 13.5 x 6.9 cm
Operating Temperature	23 - 149 deg. F (-5 to 65 deg. C))	32 - 131 deg. F (0 - 55 deg. C)
MTBF at 55°C	335 years	68 years
Available Speeds	1 Gbps/100 Mbps auto-negotiate	1 Gbps/100 Mbps auto-negotiate
Duplex Modes	Full or half at 10/100 Mbps; full only at 1 Gbps	Full or half at 10/100 Mbps; full only at 1 Gbps
Standards Conformance	IEEE 802.3 IEEE 802.3ab IEEE 802.3u PCI Express 2.1 OCP NIC 3.0	IEEE 802.1p IEEE 802.1Q IEEE 802.3ab IEEE 802.3ac IEEE 802.3ad IEEE 802.3az IEEE 802.3u IEEE 802.3x IEEE 802.3z ACPI v1.0 PCI Express 2.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) EN 60 950 (European Union) IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> FCC Part 15 - Radiated & Conducted Emissions (USA) ICES-003 - Radiated & Conducted Emissions (Canada) CISPR 22 - Radiated & Conducted Emissions (International) EN55022-1998 - Radiated & Conducted Emissions (European Union) EN55024 - 1998 - (Immunity) (European Union) CE - EMC Directive (89/336/EEC) (European Union) VCCI - Radiated & Conducted Emissions (Japan) CNS13438 - Radiated & Conducted Emissions (Taiwan) 	

Feature	Intel® Ethernet 1G 4P I350-t OCP	Intel® Gigabit 2P I350-t Adapter Intel® Gigabit 4P I350-t Adapter
	<ul style="list-style-type: none"> AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea) 	

13.1.9 Intel® Gigabit Network Mezzanine Card Specifications

Feature	Intel® Gigabit 4P I350-t Mezz
Bus Connector	PCI Express 2.0
Bus Speed	x4
Power Requirements	3.425 Watts (maximum) @ 3.3 V
Dimensions	3.65 x 3.3 in.
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)
MTBF at 55°C	108 years
Available Speeds	Full only at 1 Gbps
Duplex Modes	Full at 1 Gbps
Standards Conformance	IEEE 802.1p IEEE 802.1Q IEEE 802.3ab IEEE 802.3ac IEEE 802.3ad IEEE 802.3x ACPI v1.0 PCI Express 2.0
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) EN 60 950 (European Union) IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> FCC Part 15 - Radiated & Conducted Emissions (USA) ICES-003 - Radiated & Conducted Emissions (Canada) CISPR 22 - Radiated & Conducted Emissions (International) EN55022-1998 - Radiated & Conducted Emissions (European Union) EN55024 - 1998 - (Immunity) (European Union) CE - EMC Directive (89/336/EEC) (European Union) VCCI - Radiated & Conducted Emissions (Japan) CNS13438 - Radiated & Conducted Emissions (Taiwan) AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea)

13.1.10 Intel® Gigabit Network Daughter Cards Specifications

Feature	Intel® Gigabit 4P X710/I350 rNDC	Intel® Gigabit 4P X550/I350 rNDC	Intel® Gigabit 4P I350-t rNDC
Bus Connector	PCI Express 2.0	PCI Express 2.0	PCI Express 2.0
Bus Speed	x8	x8	x8
Transmission Mode/Connector	Twisted copper/RJ-45	Twisted copper/RJ-45	Twisted copper/RJ-45
Cabling	Cat-5e	Cat-5e	Cat-5e
Power Requirements	10.7W Maximum @ +12 V	15.39 W (max) @ +12 V	5.5W (max) @ +3.3 V
Dimensions (excluding bracket)	4.331 x 3.661 in 11.007 x 9.298 cm	5.86 x 4.35 in 14.882 x 11.04 cm	5.33 x 2.71 in 13.54 x 6.59 cm
Operating Temperature	32 - 131 deg. F (0 - 55 deg. C)	32 - 60 deg. F (0- 16 deg C.)	32 - 60 deg. F (0 - 16 deg. C)
MTBF at 55°C	108 years	251 years	117 years
Available Speeds	1 Gbps/100 Mbps	1 Gbps/100 Mbps	1 Gbps/100 Mbps
Duplex Modes	Full only	Full only	Full only
Standards Conformance	IEEE 802.3i IEEE 802.3ab IEEE 802.3u IEEE 802.3ad IEEE 802.3az PCI Express 2.1	IEEE 802.1p IEEE 802.1Q IEEE 802.3ac IEEE 802.3ad IEEE 802.3ae IEEE 802.3x ACPI v1.0 PCI Express 2.1	IEEE 802.1p IEEE 802.1Q IEEE 802.3ac IEEE 802.3ad IEEE 802.3ae IEEE 802.3x ACPI v1.0 PCI Express 2.1
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950 Third Edition- CAN/CSA-C22.2 No.60950-00 (USA/Canada) • EN 60 950 (European Union) • IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • CISPR 22 - Radiated & Conducted Emissions (International) • EN55022-1998 - Radiated & Conducted Emissions (European Union) • EN55024 - 1998 - (Immunity) (European Union) • CE - EMC Directive (89/336/EEC) (European Union) • VCCI - Radiated & Conducted Emissions (Japan) • CNS13438 - Radiated & Conducted Emissions (Taiwan) • AS/NZS3548 - Radiated & Conducted Emissions (Australia/New Zealand) • MIC notice 1997-41, EMI and MIC notice 1997-42 - EMS (Korea) 		

Feature	Intel® Gigabit 4P I350 bNDC
Bus Connector	PCI Express 2.1
Bus Speed	x4
Transmission Mode/Connector	1000Base-KX
Cabling	Backplane
Power Requirements	6.3 W (max)
Dimensions (excluding bracket)	3.00 x 2.45 in 7.62 x 6.22 cm
Operating Temperature	32 - 149 deg. F (0 - 65 deg. C)
MTBF at 55°C	200.5 years
Available Speeds	1 Gbps
Duplex Modes	Full only
Standards Conformance	PCI Express 2.1 IEEE 802.3-2008, Clause 70
Regulatory and Safety	<p>Safety Compliance</p> <ul style="list-style-type: none"> • UL 60950-1 Second Edition- Amendment 1 • CAN/CSA-C22.2 No.60950-1-07 (USA/Canada) • EN 60950-1:2006/A11:2009/A1:2010/A12:2011 Safety requirements for European Union (EU) IEC 60 950 (International) <p>EMC Compliance</p> <ul style="list-style-type: none"> • FCC Part 15 - Radiated & Conducted Emissions (USA) • ICES-003 - Radiated & Conducted Emissions (Canada) • AS/NZS CISPR 22:2009 + A1:2010 - Radiated & Conducted Emissions (International) • EN55022-2010 - Radiated & Conducted Emissions (European Union) • EN55024 - 2010 - (Immunity) (European Union) • CE - EMC Directive (2004/108/EEC) (European Union) • VCCI:2013-04 - Radiated & Conducted Emissions (Japan) • KCC KN22, KN 24 Class A (Korea)

13.2 Indicator Lights

Intel Ethernet adapters feature indicator lights on the adapter backplate that serve to indicate activity and the status of the adapter board. The following tables define the meaning for the possible states of the indicator lights for each adapter board.

Subsections are organized by number of ports and types of connections.

All drawings are representational.

13.2.1 Dual Port QSFP28 Adapters

Drawing	Label	Indication	Meaning	
	Link	Green	Operating at maximum port speed	
		Yellow	Operating at less than maximum port speed	
	Activity	Blinking On/Off	Actively transmitting or receiving data	
		Off	No link	
	<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 100G 2P E810-C-st Adapter</p> <p>Intel® Ethernet 100G 2P E810-C-stg Adapter</p>			

Drawing	Label	Indication	Meaning
	ACT/LNK	Green	Linked at maximum port speed
		Yellow	Linked at less than maximum port speed
		Blinking On/Off	Actively transmitting or receiving data
		Off	No link
	<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 100G 2P E810-C Adapter</p>		

13.2.2 Dual Port QSFP+ Adapters

Drawing	Label	Indication	Meaning
	ACT/LNK	Green	Linked at 40 Gb
		Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet Converged Network Adapter XL710-Q2</p>			

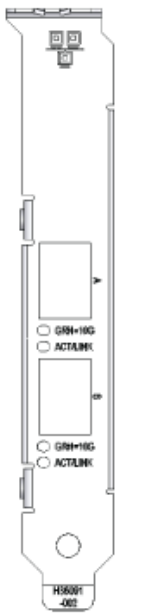
13.2.3 Dual Port SFP28 Adapters

Drawing	Label	Indication	Meaning
	GRN 25G	Green	Linked at maximum port speed
		Yellow	Linked at less than maximum port speed
	ACTIVITY	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 25G 2P E810-XXV Adapter</p> <p>Intel® Ethernet 25G 2P XXV710 Adapter</p>			

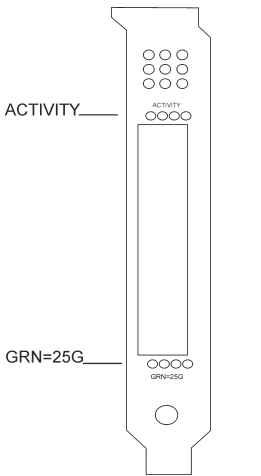
Drawing	Label	Indication	Meaning
	LNK	Green	Operating at maximum port speed
		Yellow	Linked at less than maximum port speed
	ACT	Green flashing	Data activity
		Off	No activity
<p>These indicator lights apply to the following devices: Intel® Ethernet 25G 2P E810-XXV OCP</p>			

13.2.4 Dual Port SFP/SFP+ Adapters

Drawing	Label	Indication	Meaning
	LNK	Green	Operating at maximum port speed
		Yellow	Linked at less than maximum port speed
	ACT	Green flashing	Data activity
		Off	No activity
<p>These indicator lights apply to the following devices: Intel® Ethernet 10G 2P X710 OCP</p>			

Drawing	Label	Indication	Meaning
	LNK	Green	Linked at 10 Gb
		Yellow	Linked at 1 Gb
	ACT	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet Converged Network Adapter X710</p>			

13.2.5 Quad Port SFP28 Adapters

Drawing	Label	Indication	Meaning
	GRN 25G	Green	Linked at 25 Gb
		Yellow	Linked at 10 Gb or 1 Gb
	ACTIVITY	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 25G 4P E810-XXV Adapter</p>			

Drawing	Label	Indication	Meaning
	Link	Green	Operating at maximum port speed
		Yellow	Operating at less than maximum port speed
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 25G 4P E810-XXV-st Adapter Intel® Ethernet 25G 4P E810-XXV-stg Adapter</p>			

Drawing	Label	Indication	Meaning
	Link	Green	Linked at maximum port speed
		Yellow	Linked at less than maximum port speed
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 25G 4P E810-XXV OCP</p>			

13.2.6 Quad Port SFP/SFP+ Adapters

Drawing	Label	Indication	Meaning
	Link	Green	Linked at maximum port speed
		Yellow	Linked at less than maximum port speed
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 10G 4P X710 OCP</p>			

13.2.7 Dual Port Copper Adapters

Drawing	Label	Indication	Meaning
	Link	Green	Linked at 10 Gbps
		Yellow	Linked at slower than 10 Gbps
		Off	No link
	Activity	Blinking On/Off	Actively transmitting or receiving data
Off		No link	
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 10G 2P X710-T2L-t OCP</p>			

Drawing	Label	Indication	Meaning
	Link	Green	Linked at 10 Gbps
		Yellow	Linked at slower than 10 Gbps
		Off	No link
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 10G 2P X710-T2L-t Adapter</p>			

Drawing	Label	Indication	Meaning
	Link	Green	Linked at 10 Gbps
		Yellow	Linked at slower than 10 Gbps
		Off	No link
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 10G 2P X550-t Adapter</p>			

Drawing	Label	Indication	Meaning	
	ACT/LNK	Green on	The adapter is connected to a valid link partner	
		Green flashing	Data activity	
		Off	No link	
	10/100/1000	10/100/1000	Off	10 Mbps
			Green	100 Mbps
			Yellow	1000 Mbps
			Orange flashing	Identity. Use the "Identify Adapter" button in Intel PROSet to control blinking. See Intel PROSet Help for more information.
	<p>These indicator lights apply to the following devices:</p> <p>Intel® Gigabit 2P I350-t Adapter</p>			

13.2.8 Quad Port Copper Adapters

Drawing	Label	Indication	Meaning
	Link	Green	Linked at 10 Gbps
		Yellow	Linked at slower than 1 Gbps
		Off	No link
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 10G 4P X710-T4L-t OCP</p>			

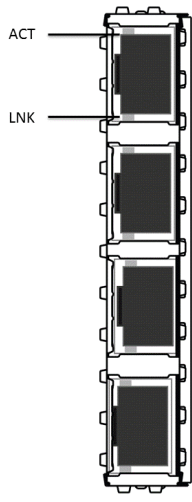
Drawing	Label	Indication	Meaning
<p>LINK 100=GRN OTHER=YLW</p> <p>LINK</p> <p>ACT</p> <p>ACT=GRN</p>	Link	Green	Linked at 10 Gbps
		Yellow	Linked at slower than 10 Gbps
		Off	No link
	Activity	Blinking On/Off	Actively transmitting or receiving data
		Off	No link
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 10G 4P X710-T4L-t Adapter</p>			

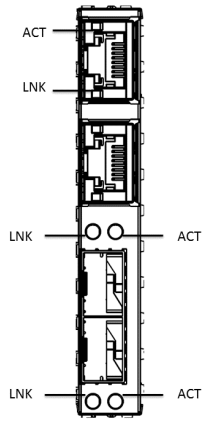
Drawing	Label	Indication	Meaning
<p>ACT</p> <p>LNK</p>	ACT	Green on	The adapter is connected to a valid link partner
		Green flashing	Data activity
		Off	No link
	LNK	Green	10 Gbps
		Yellow	1 Gbps
		Off	100 Mbps
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet Converged Network Adapter X710</p> <p>Intel® Ethernet Converged Network Adapter X710-T</p>			

Drawing	Label	Indication	Meaning
	ACT/LNK	Green on	The adapter is connected to a valid link partner
		Green flashing	Data activity
		Off	No link
	10/100/1000	Green	100 Mbps
		Yellow	1000 Mbps
		Orange flashing	Identity. Use the "Identify Adapter" button in Intel® PROSet to control blinking. See Intel PROSet Help for more information.
		Off	10 Mbps
	<p>These indicator lights apply to the following devices:</p> <p>Intel® Gigabit 4P I350-t Adapter</p>		

13.2.9 rNDC (Rack Network Daughter Cards)

Drawing	Label	Indication	Meaning
	LNK (green/yellow)	Green on	Operating at maximum port speed
		Off	No link
	ACT (green)	Green flashing	Data activity
		Off	No activity
<p>These indicator lights apply to the following devices:</p> <p>Intel® Ethernet 40G 2P XL710 QSFP+ rNDC</p>			

Drawing	Label	Indication	Meaning
	LNK (green/yellow)	Green on	Operating at maximum port speed
		Yellow on	Operating at lower port speed
		Off	No link
	ACT (green)	Green flashing	Data activity
		Off	No activity
	<p>These indicator lights apply to the following devices:</p> <ul style="list-style-type: none"> Intel® Ethernet 1G 4P I350-t OCP Intel® Ethernet 10G 4P X550/I350 rNDC Intel® Gigabit 4P X550/I350 rNDC Intel® Ethernet 10G 4P X550 rNDC Intel® Gigabit 4P I350-t rNDC 		

Drawing	Label	Indication	Meaning
	LNK (green/yellow)	Green on	Operating at maximum port speed
		Yellow on	Operating at lower port speed
		Off	No link
	ACT (green)	Green flashing	Data activity
		Off	No activity
	<p>These indicator lights apply to the following devices:</p> <ul style="list-style-type: none"> Intel® Ethernet Gigabit 4P x710/I350 rNDC Intel® 10G 4P X710/I350 rNDC 		

14. Legal Disclaimers

14.1 Software License Agreement

INTEL SOFTWARE LICENSE AGREEMENT (Final, License)

IMPORTANT - READ BEFORE COPYING, INSTALLING OR USING.

Do not use or load this software and any associated materials (collectively, the "Software") until you have carefully read the following terms and conditions. By loading or using the Software, you agree to the terms of this Agreement. If you do not wish to so agree, do not install or use the Software.

LICENSES

Please Note:

- If you are a network administrator, the "Site License" below shall apply to you.
- If you are an end user, the "Single User License" shall apply to you.

SITE LICENSE. You may copy the Software onto your organization's computers for your organization's use, and you may make a reasonable number of back-up copies of the Software, subject to these conditions:

1. **This Software is licensed for use only in conjunction with Intel component products. Use of the Software in conjunction with non-Intel component products is not licensed hereunder.**
2. You may not copy, modify, rent, sell, distribute or transfer any part of the Software except as provided in this Agreement, and you agree to prevent unauthorized copying of the Software.
3. You may not reverse engineer, decompile, or disassemble the Software.
4. You may not sublicense or permit simultaneous use of the Software by more than one user.
5. The Software may include portions offered on terms in addition to those set out here, as set out in a license accompanying those portions.

SINGLE USER LICENSE. You may copy the Software onto a single computer for your personal, noncommercial use, and you may make one back-up copy of the Software, subject to these conditions:

1. **This Software is licensed for use only in conjunction with Intel component products. Use of the Software in conjunction with non-Intel component products is not licensed hereunder.**
2. You may not copy, modify, rent, sell, distribute or transfer any part of the Software except as provided in this Agreement, and you agree to prevent unauthorized copying of the Software.
3. You may not reverse engineer, decompile, or disassemble the Software.
4. You may not sublicense or permit simultaneous use of the Software by more than one user.
5. The Software may include portions offered on terms in addition to those set out here, as set out in a license accompanying those portions.

OWNERSHIP OF SOFTWARE AND COPYRIGHTS. Title to all copies of the Software remains with Intel or its suppliers. The Software is copyrighted and protected by the laws of the United States and other countries, and international treaty provisions. You may not remove any copyright notices from the Software. Intel may make changes to the Software, or to items referenced therein, at any time without

notice, but is not obligated to support or update the Software. Except as otherwise expressly provided, Intel grants no express or implied right under Intel patents, copyrights, trademarks, or other intellectual property rights. You may transfer the Software only if the recipient agrees to be fully bound by these terms and if you retain no copies of the Software.

LIMITED MEDIA WARRANTY. If the Software has been delivered by Intel on physical media, Intel warrants the media to be free from material physical defects for a period of ninety days after delivery by Intel. If such a defect is found, return the media to Intel for replacement or alternate delivery of the Software as Intel may select.

EXCLUSION OF OTHER WARRANTIES. EXCEPT AS PROVIDED ABOVE, THE SOFTWARE IS PROVIDED "AS IS" WITHOUT ANY EXPRESS OR IMPLIED WARRANTY OF ANY KIND INCLUDING WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. Intel does not warrant or assume responsibility for the accuracy or completeness of any information, text, graphics, links or other items contained within the Software.

LIMITATION OF LIABILITY. IN NO EVENT SHALL INTEL OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, OR LOST INFORMATION) ARISING OUT OF THE USE OF OR INABILITY TO USE THE SOFTWARE, EVEN IF INTEL HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SOME JURISDICTIONS PROHIBIT EXCLUSION OR LIMITATION OF LIABILITY FOR IMPLIED WARRANTIES OR CONSEQUENTIAL OR INCIDENTAL DAMAGES, SO THE ABOVE LIMITATION MAY NOT APPLY TO YOU. YOU MAY ALSO HAVE OTHER LEGAL RIGHTS THAT VARY FROM JURISDICTION TO JURISDICTION.

TERMINATION OF THIS AGREEMENT. Intel may terminate this Agreement at any time if you violate its terms. Upon termination, you will immediately destroy the Software or return all copies of the Software to Intel.

APPLICABLE LAWS. Claims arising under this Agreement shall be governed by the laws of California, excluding its principles of conflict of laws and the United Nations Convention on Contracts for the Sale of Goods. You may not export the Software in violation of applicable export laws and regulations. Intel is not obligated under any other agreements unless they are in writing and signed by an authorized representative of Intel.

GOVERNMENT RESTRICTED RIGHTS. The Software is provided with "RESTRICTED RIGHTS." Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 *et seq.* or its successor. Use of the Software by the Government constitutes acknowledgment of Intel's proprietary rights therein. Contractor or Manufacturer is Intel.

14.1.0.1 Third-party Licenses

Portions of this release may include software distributed under the following licenses.

Open Toolkit Library (OpenTK)

The Open Toolkit library license

Copyright (c) 2006 - 2009 The Open Toolkit library.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Third parties

* The Open Toolkit library includes portions of the Mono class library, which are covered by the following license:

Copyright (c) 2004 Novell, Inc.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

* Half-to-Single and Single-to-Half conversions are covered by the following license:

Copyright (c) 2002, Industrial Light & Magic, a division of Lucas Digital Ltd. LLC. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- Neither the name of Industrial Light & Magic nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT OWNER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

RSA Data Security-MD5 Message

RSA Data Security

Copyright (C) 1991-2, RSA Data Security, Inc. Created 1991. All rights reserved.

License to copy and use this software is granted provided that it is identified as the "RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing this software or this function.

License is also granted to make and use derivative works provided that such works are identified as "derived from the RSA Data Security, Inc. MD5 Message-Digest Algorithm" in all material mentioning or referencing the derived work.

RSA Data Security, Inc. makes no representations concerning either the merchantability of this software or the suitability of this software for any particular purpose. It is provided "as is" without express or implied warranty of any kind.

These notices must be retained in any copies of any part of this documentation and/or software.

14.2 Restrictions and Disclaimers

Information in this document is subject to change without notice.

Copyright © 2008-2023, Intel Corporation. All rights reserved.

Trademarks used in this text: *Dell* and the *Dell* logo are trademarks of Dell, Inc.; Intel is a trademark of Intel Corporation or its subsidiaries in the U.S. and/or other countries.

* Other trademarks and trade names may be used in this document to refer to either the entities claiming the marks and names or their products. Intel Corporation disclaims any proprietary interest in trademarks and trade names other than its own.

14.2.1 Restrictions and Disclaimers

The information contained in this document, including all instructions, cautions, and regulatory approvals and certifications, is provided by the supplier and has not been independently verified or tested by Dell. Dell cannot be responsible for damage caused as a result of either following or failing to follow these instructions.

All statements or claims regarding the properties, capabilities, speeds or qualifications of the part referenced in this document are made by the supplier and not by Dell. Dell specifically disclaims knowledge of the accuracy, completeness or substantiation for any such statements. All questions or comments relating to such statements or claims should be directed to the supplier.

14.2.2 Export Regulations

Customer acknowledges that these Products, which may include technology and software, are subject to the customs and export control laws and regulations of the United States (U.S.) and may also be subject to the customs and export laws and regulations of the country in which the Products are manufactured and/or received. Customer agrees to abide by those laws and regulations. Further, under U.S. law, the Products may not be sold, leased or otherwise transferred to restricted end users or to restricted countries. In addition, the Products may not be sold, leased or otherwise transferred to, or utilized by an end-user engaged in activities related to weapons of mass destruction, including without limitation, activities related to the design, development, production or use of nuclear weapons, materials, or facilities, missiles or the support of missile projects, and chemical or biological weapons.

September 21, 2023