

---

# Solarflare® Server Adapter User Guide

---

The information disclosed to you hereunder (the “Materials”) is provided solely for the selection and use of Xilinx products. To the maximum extent permitted by applicable law: (1) Materials are made available “AS IS” and with all faults, Xilinx hereby DISCLAIMS ALL WARRANTIES AND CONDITIONS, EXPRESS, IMPLIED, OR STATUTORY, INCLUDING BUT NOT LIMITED TO WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, OR FITNESS FOR ANY PARTICULAR PURPOSE; and (2) Xilinx shall not be liable (whether in contract or tort, including negligence, or under any other theory of liability) for any loss or damage of any kind or nature related to, arising under, or in connection with, the Materials (including your use of the Materials), including for any direct, indirect, special, incidental, or consequential loss or damage (including loss of data, profits, goodwill, or any type of loss or damage suffered as a result of any action brought by a third party) even if such damage or loss was reasonably foreseeable or Xilinx had been advised of the possibility of the same. Xilinx assumes no obligation to correct any errors contained in the Materials or to notify you of updates to the Materials or to product specifications. You may not reproduce, modify, distribute, or publicly display the Materials without prior written consent. Certain products are subject to the terms and conditions of Xilinx’s limited warranty, please refer to Xilinx’s Terms of Sale which can be viewed at <https://www.xilinx.com/legal.htm#tos>; IP cores may be subject to warranty and support terms contained in a license issued to you by Xilinx. Xilinx products are not designed or intended to be fail-safe or for use in any application requiring fail-safe performance; you assume sole risk and liability for use of Xilinx products in such critical applications, please refer to Xilinx’s Terms of Sale which can be viewed at <https://www.xilinx.com/legal.htm#tos>.

## **AUTOMOTIVE APPLICATIONS DISCLAIMER**

AUTOMOTIVE PRODUCTS (IDENTIFIED AS “XA” IN THE PART NUMBER) ARE NOT WARRANTED FOR USE IN THE DEPLOYMENT OF AIRBAGS OR FOR USE IN APPLICATIONS THAT AFFECT CONTROL OF A VEHICLE (“SAFETY APPLICATION”) UNLESS THERE IS A SAFETY CONCEPT OR REDUNDANCY FEATURE CONSISTENT WITH THE ISO 26262 AUTOMOTIVE SAFETY STANDARD (“SAFETY DESIGN”). CUSTOMER SHALL, PRIOR TO USING OR DISTRIBUTING ANY SYSTEMS THAT INCORPORATE PRODUCTS, THOROUGHLY TEST SUCH SYSTEMS FOR SAFETY PURPOSES. USE OF PRODUCTS IN A SAFETY APPLICATION WITHOUT A SAFETY DESIGN IS FULLY AT THE RISK OF CUSTOMER, SUBJECT ONLY TO APPLICABLE LAWS AND REGULATIONS GOVERNING LIMITATIONS ON PRODUCT LIABILITY.

## **Copyright**

© Copyright 2020 Xilinx, Inc. Xilinx, the Xilinx logo, OpenOnload®, EnterpriseOnload®, Cloud Onload®, and other designated brands included herein are trademarks of Xilinx in the United States and other countries. All other trademarks are the property of their respective owners.

SF-103837-CD

Issue 28

Last revised: November 2020

# Table of Contents

<b>1 Introduction</b>	<b>1</b>
1.1 Virtual NIC Interface	1
1.2 Product Specifications	5
1.3 Software Driver Support	16
1.4 Solarflare AppFlex™ Technology	17
1.5 Open Source Licenses	17
1.6 Support and Driver Download	18
1.7 Regulatory Information and Approvals	18
<b>2 Installation</b>	<b>19</b>
2.1 Solarflare Network Adapter Products	20
2.2 CPU/PCIe Requirements	21
2.3 Fitting a Full Height Bracket (optional)	21
2.4 Inserting an Adapter in a PCI Express (PCIe) Slot	22
2.5 Inserting an adapter in an OCP 2.0 mezzanine slot	23
2.6 Inserting an adapter in an OCP 3.0 mezzanine slot	23
2.7 Attaching a Cable (RJ-45)	25
2.8 Attaching a Cable (SFP+, QSFP+, SFP28, QSFP28)	26
2.9 Cables and Transceivers	28
2.10 Supported Speed and Mode	40
2.11 Forward Error Correction	42
2.12 LED States	43
2.13 Port Modes	43
2.14 Single Optical Fiber - RX Configuration	49
2.15 Solarflare Precision Time Synchronization Adapters	49
2.16 Solarflare ApplicationOnload™ Engine	50

<b>3 Solarflare Adapters on Linux</b>	<b>51</b>
3.1 System Requirements	52
3.2 Linux Platform Driver Feature Set	52
3.3 About the In-tree Driver	54
3.4 Getting the Adapter Driver	54
3.5 Installing the DKMS RPM	55
3.6 Installing the Source RPM	56
3.7 Configuring the Solarflare Adapter	58
3.8 Setting Up VLANs	61
3.9 Setting Up Teams	61
3.10 NIC Partitioning	62
3.11 NIC Partitioning with SR-IOV	67
3.12 Receive Side Scaling (RSS)	70
3.13 Receive Flow Steering (RFS)	72
3.14 Transmit Packet Steering (XPS)	74
3.15 Linux Utilities RPM	76
3.16 Configuring the Boot Manager with sfboot	78
3.17 Upgrading Adapter Firmware with sfupdate	86
3.18 Installing an activation key with sfkey	92
3.19 Performance Tuning on Linux	95
3.20 Web Server - Driver Optimization	103
3.21 Interrupt Affinity	105
3.22 Module Parameters	115
3.23 Linux ethtool Statistics	117
3.24 Reading sensors	126
3.25 Driver Logging Levels	128
3.26 Running Adapter Diagnostics	129
3.27 Running Cable Diagnostics	130

<b>4 Solarflare Adapters on Windows</b> .....	<b>131</b>
4.1 Windows 2012 R2 / 2016 / 2019 Driver .....	132
4.2 Legacy Driver .....	132
4.3 System Requirements .....	132
4.4 Driver Certification .....	132
4.5 Minimum Driver and Firmware Packages .....	133
4.6 Firmware Variants .....	133
4.7 Windows Feature Set .....	134
4.8 Installing Solarflare Driver Package .....	135
4.9 Install SolarflareTools .....	137
4.10 Using SolarflareTools .....	138
4.11 Configuration & Management .....	143
4.12 Adapter Configuration .....	144
4.13 Flow Control .....	148
4.14 Configuring FEC .....	149
4.15 Jumbo Frames .....	149
4.16 Checksum Offload .....	150
4.17 Interrupt Moderation (Interrupt Coalescing) .....	152
4.18 NUMA Node .....	152
4.19 Receive Side Scaling (RSS) .....	154
4.20 Receive and Transmit Buffers .....	157
4.21 Virtual Machine Queue .....	159
4.22 Teaming and VLANs .....	160
4.23 Adapter Statistics .....	163
4.24 Performance Tuning on Windows .....	164
4.25 List Installed Adapters .....	172
4.26 Startup/Boot time Errors .....	172

<b>5 Solarflare Adapters on VMware</b>	<b>173</b>
5.1 Native ESXi Driver (VMkernel API)	174
5.2 Legacy Driver (vmklinux API)	174
5.3 System Requirements	174
5.4 Distribution Packages	175
5.5 VMware Feature Set	176
5.6 Install Solarflare Drivers	178
5.7 Driver Configuration	180
5.8 Adapter Configuration	181
5.9 Granting access to the NIC from the Virtual Machine	181
5.10 NIC Teaming	182
5.11 Configuring VLANs	183
5.12 Performance Tuning on VMware	184
5.13 Interface Statistics	194
5.14 vSwitch/VM Network Statistics	194
5.15 CIM Provider	197
5.16 Adapter Firmware Upgrade - sfupdate_esxi	198
5.17 Adapter Configuration - sfboot_esxi	201
5.18 ESXCLI Extension	203
5.19 vSphere Client Plugin	212
5.20 Fault Reporting - Diagnostics	222
5.21 Network Core Dump	223
5.22 Adapter Diagnostic Selftest	223
<b>6 SR-IOV Virtualization Using KVM</b>	<b>224</b>
6.1 Introduction	224
6.2 SR-IOV	229
6.3 KVM Network Architectures	231
6.4 PF-IOV	245
6.5 General Configuration	247
6.6 Feature Summary	248
6.7 Limitations	249
<b>7 SR-IOV Virtualization Using ESXi</b>	<b>250</b>
7.1 Configuration Procedure - SR-IOV	252
7.2 Configuration Procedure - DirectPath I/O	252
7.3 Install Solarflare Drivers in the Guest	252
7.4 Install the Solarflare Driver on the ESXi host	252
7.5 Solarflare Utilities for legacy driver on the ESXi host	252
7.6 Solarflare Utilities for native driver on the ESXi host	254
7.7 Configure VFs on the Host/Adapter	255
7.8 Virtual Machine	256

<b>8 Solarflare Boot Manager</b> .....	<b>257</b>
8.1 Introduction .....	257
8.2 Solarflare Boot Manager.....	258
8.3 iPXE Support .....	259
8.4 sfupdate Options for PXE upgrade/downgrade .....	259
8.5 Starting PXE Boot.....	261
8.6 iPXE Image Create .....	265
8.7 Multiple PF - PXE Boot .....	267
8.8 Default Adapter Settings.....	270
<b>9 Unattended Installations</b> .....	<b>272</b>
9.1 Unattended Installation - Red Hat Enterprise Linux .....	274
9.2 Unattended Installation - SUSE Linux Enterprise Server .....	275

# 1

## Introduction

This is the User Guide for Solarflare® Server Adapters. This chapter covers the following topics:

- [Virtual NIC Interface on page 1](#)
- [Product Specifications on page 5](#)
- [Software Driver Support on page 16](#)
- [Solarflare AppFlex™ Technology on page 17](#)
- [Open Source Licenses on page 17](#)
- [Support and Driver Download on page 18](#)
- [Regulatory Information and Approvals on page 18.](#)



**NOTE:** Throughout this guide the term Onload refers to both OpenOnload® and EnterpriseOnload® unless otherwise stated. Users of Onload should refer to the *Onload User Guide, SF-104474-CD*, which describes procedures for download and installation of the Onload distribution, accelerating and tuning the application using Onload to achieve minimum latency and maximum throughput.

### 1.1 Virtual NIC Interface

Solarflare’s VNIC architecture provides the key to efficient server I/O and is flexible enough to be applied to multiple server deployment scenarios. These deployment scenarios include:

- **Kernel Driver** – This deployment uses an instance of a VNIC per CPU core for standard operating system drivers. This allows network processing to continue over multiple CPU cores in parallel. The virtual interface provides a performance-optimized path for the kernel TCP/IP stack and contention-free access from the driver, resulting in extremely low latency and reduced CPU utilization.
- **Accelerated Virtual I/O** – The second deployment scenario greatly improves I/O for virtualized platforms. The VNIC architecture can provide a VNIC per Virtual Machine, giving over a thousand protected interfaces to the host system, granting any virtualized (guest) operating system direct access to the network hardware. Solarflare’s hybrid SR-IOV technology, unique to Solarflare Ethernet controllers, is the only way to provide bare-metal I/O performance to virtualized guest operating systems whilst retaining the ability to live migrate virtual machines.

- **OpenOnload™** – The third deployment scenario aims to leverage the host CPU(s) to full capacity, minimizing software overheads by using a vNIC per application to provide a kernel bypass solution. Solarflare has created both an open-source and Enterprise class high-performance application accelerator that delivers lower and more predictable latency and higher message rates for TCP and UDP-based applications, all with no need to modify applications or change the network infrastructure. To learn more about the open source OpenOnload project or EnterpriseOnload, download the Onload user guide (SF-104474-CD) or contact your reseller.

## Advanced Features and Benefits

<b>Virtual NIC support</b>	<p>The core of Solarflare technology. Protected vNIC interfaces can be instantiated for each running guest operating system or application, giving it a direct pipeline to the Ethernet network. This architecture provides the most efficient way to maximize network and CPU efficiency. The Solarflare Ethernet controller supports up to 1024 vNIC interfaces per port.</p> <p>On IBM System p servers equipped with Solarflare adapters, each adapter is assigned to a single Logical Partition (LPAR) where all vNICs are available to the LPAR.</p>
<b>PCI Express</b>	Implements PCI Express 3.1.
<b>High Performance</b>	Support for 40G Ethernet interfaces and a new internal datapath micro architecture.
<b>Hardware Switch Fabric</b>	Full hardware switch fabric in silicon capable of steering any flow based on Layer 2, Layer 3 or application level protocols between physical and virtual interfaces. Supporting an open software defined network control plane with full PCI-IOV virtualization acceleration for high performance guest operating systems and virtual applications.
<b>Improved flow processing</b>	The addition of dedicated parsing, filtering, traffic shaping and flow steering engines which are capable of operating flexibly and with an optimal combination of a full hardware data plane with software based control plane.
<b>TX PIO</b>	Transmit Programmed input/output is the direct transfer of data to the adapter without CPU involvement. As an alternative to the usual bus master DMA method, TX PIO improves latency and is especially useful for smaller packets.



<b>CTPIO</b>	Cut Through PIO - TX packets are streamed directly from the PCIe interface to the adapter port bypassing the main TX datapath to deliver lowest TX latency.
<b>Multicast Replication</b>	Received multicast packets are replicated in hardware and delivered to multiple receive queues.
<b>Sideband management</b>	NCSI RMI2 interface for base board management integration.  SMBus interface for legacy base board management integration.
<b>PCI Single-Root-IOV, SR-IOV, capable</b>	16 Physical functions and up to 240 Virtual functions per adapter.  Flexible deployment of 1024 channels between Virtual and Physical Functions.  Support Alternate Routing ID (ARI).  SR-IOV is not supported for Solarflare adapters on IBM System p servers.
<b>10 Gigabit Ethernet</b>	Supports the ability to design a cost effective, high performance 10 Gigabit Ethernet solution.
<b>25 Gigabit Ethernet</b>	Supported on X2 series and U25 adapters.  <a href="#">See Supported Speed and Mode on page 40</a>
<b>Receive Side Scaling (RSS)</b>	IPv4 and IPv6 RSS raises the utilization levels of multi-core servers dramatically by distributing I/O load across all CPUs and cores.
<b>Stateless offloads</b>	Through the addition of hardware based TCP segmentation and reassembly offloads, VLAN, VxLAN, NVGRE and GENEVE offloads.
<b>Jumbo frame support</b>	Support for up to 9216 byte jumbo frames.
<b>MSI-X support</b>	2048 MSI-X interrupt support enables higher levels of performance.  Can also work with legacy line based interrupts.
<b>Ultra low latency</b>	Cut through architecture. < 7µs end to end latency with standard kernel drivers, < 1µs with Onload drivers.

<b>Remote boot</b>	<p>Support for PXE boot 2.1 and UEFI Boot provides flexibility in cluster design and diskless servers (see <a href="#">Solarflare Boot Manager on page 257</a>).</p> <p>Network boot is not supported for Solarflare adapters on IBM System p servers.</p>
<b>MAC address filtering</b>	<p>Enables the hardware to steer packets based on the MAC address to a VNIC.</p>
<b>Hardware timestamps</b>	<p>The Solarflare Flareon™ and XtremeScale™ series adapters can support hardware timestamping for all packets, sent and received - including PTP.</p> <p>The adapters incorporate a highly accurate stratum 3 compliant oscillator with drift of 0.37 PPM per day (c. 32ms/day).</p>
<b>FEC</b>	<p>Supported on X2 series and U25 25GbE adapters.</p> <p>25G link Forward Error Correction employs redundancy in channel coding as a technique used to reduce bit errors (BER) in noisy or unreliable communications channels.</p> <p>See <a href="#">Forward Error Correction on page 42</a></p>
<b>AN/LT</b>	<p>Supported on X2 series and U25 25GbE adapters.</p> <p>Auto-negotiation/Link Training</p> <p>See <a href="#">X2-series and U25 adapters on page 41</a></p>

## 1.2 Product Specifications

### Xilinx Alveo™ Series SFP28 Network Adapters

Xilinx Alveo™ U25 Dual-Port 25GbE SFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	U25
<b>Controller silicon</b>	SFC9250
<b>Power</b>	50W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>FEC - Forward Error Correction</b>	Yes
<b>Network ports</b>	2 x SFP28 (1G/10G/25G)

## Solarflare XtremeScale™ X2 Series QSFP28 Network Adapters

### Solarflare XtremeScale™ X2542 Dual-Port 100GbE QSFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	X2542 or X2542-Plus
<b>Controller silicon</b>	SFC9250
<b>Power</b>	13.2W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>FEC - Forward Error Correction</b>	Yes
<b>Network ports</b>	2 x QSFP28 (1G/10G/25G/40G/50G/100G)

### Solarflare XtremeScale™ X2541 Single-Port 100GbE QSFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	X2541 or X2541-Plus
<b>Controller silicon</b>	SFC9250
<b>Power</b>	13.2W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>FEC - Forward Error Correction</b>	Yes
<b>Network ports</b>	1 x QSFP28 (1G/10G/25G/50G/100G)

## Solarflare XtremeScale™ X2 Series SFP28 Network Adapters

### Solarflare XtremeScale™ X2562 Dual-Port 25GbE SFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	X2562-10G/25G or X2562-10G25G-Plus
<b>Controller silicon</b>	SFC9250
<b>Power</b>	18.5W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>FEC - Forward Error Correction</b>	Yes
<b>Network ports</b>	2 x SFP28 (1G/10G/25G)

### Solarflare XtremeScale™ X2552 Dual-Port 25GbE SFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	X2522-25G or X2522-25G-Plus
<b>Controller silicon</b>	SFC9250
<b>Power</b>	13.2W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>FEC - Forward Error Correction</b>	Yes
<b>Network ports</b>	2 x SFP28 (1G/10G/25G)

### Solarflare XtremeScale™ X2522-25G Dual-Port 25GbE SFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	X2522-25G or X2522-25G-Plus
<b>Controller silicon</b>	SFC9250
<b>Power</b>	13.2W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>FEC - Forward Error Correction</b>	Yes
<b>Network ports</b>	2 x SFP28 (1G/10G/25G)

### Solarflare XtremeScale™ X2522-10G Dual-Port 10GbE SFP28 PCIe 3.1 Server Adapter

<b>Part numbers</b>	X2522-10G or X2522-10G-Plus
<b>Controller silicon</b>	SFC9250
<b>Power</b>	13.2W typical
<b>PCI Express</b>	8/16 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Yes
<b>SR-IOV</b>	Yes
<b>CTPIO</b>	Yes
<b>Network ports</b>	2 x SFP28 (1G/10G)

## Solarflare XtremeScale™ 8000 Series Network Adapters

### Solarflare XtremeScale™ SFN8722 Dual-Port 10GbE SFP+ PCIe 3.1 OCP Server Adapter

<b>Part numbers</b>	SFN8722
<b>Controller silicon</b>	SFC9240
<b>Power</b>	10.5W typical
<b>PCI Express</b>	8 lanes Gen 3.1 (8.0GT/s)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	No
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)

### Solarflare XtremeScale™ SFN8542 Dual-Port 40GbE QSFP+ PCIe 3.1 Server I/O Adapter

<b>Part numbers</b>	SFN8542 or SFN8542-Plus
<b>Controller silicon</b>	SFC9240
<b>Power</b>	12.5W typical
<b>PCI Express</b>	16 lanes Gen 3.1 (8.0GT/s), x16 edge connector
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes (factory enabled for the Plus version)
<b>PTP and hardware timestamps</b>	Yes (factory enabled for the Plus version)
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x QSFP+ (40G/10G)

## Solarflare XtremeScale™ SFN8522M Dual-Port 10GbE SFP+ PCIe 3.1 Server I/O Adapter

<b>Part numbers</b>	SFN8522M, SFN8522M-Onload, or SFN8522M-Plus
<b>Controller silicon</b>	SFC9240
<b>Power</b>	10.5W typical
<b>PCI Express</b>	8 lanes Gen 3.1 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes (factory enabled for the Onload and Plus versions)
<b>PTP and hardware timestamps</b>	Yes (factory enabled for the Plus version)
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)

## Solarflare XtremeScale™ SFN8522 Dual-Port 10GbE SFP+ PCIe 3.1 Server I/O Adapter

<b>Part numbers</b>	SFN8522, SFN8522-Onload, or SFN8522-Plus
<b>Controller silicon</b>	SFC9240
<b>Power</b>	10.5W typical
<b>PCI Express</b>	8 lanes Gen 3.1 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes (factory enabled for the Onload and Plus versions)
<b>PTP and hardware timestamps</b>	Yes (factory enabled for the Plus version)
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)



## Solarflare XtremeScale™ SFN8042 Dual-Port 40GbE QSFP+ PCIe 3.1 Server I/O Adapter

<b>Part numbers</b>	SFN8042
<b>Controller silicon</b>	SFC9240
<b>Power</b>	12.5W typical
<b>PCI Express</b>	8 lanes Gen 3.1 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes
<b>PTP and hardware timestamps</b>	Yes
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x QSFP+ (40G/10G)

## Solarflare Flareon™ Network Adapters

### Solarflare Flareon™ Ultra SFN7322F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7322F
<b>Controller silicon</b>	SFC9120
<b>Power</b>	5.9W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes (factory enabled)
<b>PTP and hardware timestamps</b>	Yes (factory enabled)
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)

### Solarflare Flareon™ Ultra SFN7142Q Dual-Port 40GbE QSFP+ PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7142Q
<b>Controller silicon</b>	SFC9140
<b>Power</b>	13W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes (factory enabled)
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x QSFP+ (40G/10G)

### Solarflare Flareon™ Ultra SFN7124F Quad-Port 10GbE SFP+ PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7124F
<b>Controller silicon</b>	SFC9140
<b>Power</b>	13W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Yes (factory enabled)
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	4 x SFP+ (10G/1G)

## Solarflare Flareon™ Ultra SFN7122F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7122F
<b>Controller silicon</b>	SFC9120
<b>Power</b>	5.9W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	1Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts.
<b>Supports OpenOnload</b>	Yes (factory enabled)
<b>PTP and hardware timestamps</b>	AppFlex™ activation key required
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed.
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)

## Solarflare Flareon™ SFN7042Q Dual-Port 40GbE QSFP+ PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7042Q
<b>Controller silicon</b>	SFC9140
<b>Power</b>	13W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Enabled by installing AppFlex activation key
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x QSFP+ (40G/10G)

## Solarflare Flareon™ Ultra SFN7024F Quad-Port 10GbE SFP+ PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7024F
<b>Controller silicon</b>	SFC9140
<b>Power</b>	13W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Enabled by installing AppFlex activation key
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	No
<b>SR-IOV</b>	Yes
<b>Network ports</b>	4 x SFP+ (10G/1G)

## Solarflare Flareon™ Ultra SFN7022F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7022F
<b>Controller silicon</b>	SFC9120
<b>Power</b>	5.9W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts.
<b>Supports OpenOnload</b>	Enabled by installing AppFlex activation key
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed.
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)

## Solarflare Flareon™ SFN7004F Quad-Port 10GbE SFP+ PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7004F
<b>Controller silicon</b>	SFC9140
<b>Power</b>	13W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts
<b>Supports OpenOnload</b>	Enabled by installing AppFlex activation key
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	No
<b>SR-IOV</b>	Yes
<b>Network ports</b>	4 x SFP+ (10G/1G)

## Solarflare Flareon™ SFN7002F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter

<b>Part number</b>	SFN7002F
<b>Controller silicon</b>	SFC9120
<b>Power</b>	5.9W typical
<b>PCI Express</b>	8 lanes Gen 3 (8.0GT/s), x8 edge connector (usable in x8 and x16 slots)
<b>PCIe features support</b>	Per adapter: 16 PF, 240 VF, 2048 VI, 2048 MSI-X Interrupts.
<b>Supports OpenOnload</b>	Enabled by installing AppFlex activation key
<b>PTP and hardware timestamps</b>	Enabled by installing AppFlex activation key
<b>1PPS</b>	Optional bracket and cable assembly – not factory installed.
<b>SR-IOV</b>	Yes
<b>Network ports</b>	2 x SFP+ (10G/1G)

## 1.3 Software Driver Support

The software driver is currently supported on the following distributions:

- Red Hat Enterprise Linux 6.10
- Red Hat Enterprise Linux 7.5, 7.6, 7.7
- Red Hat Enterprise Linux 8.0, 8.1
- SUSE Linux Enterprise Server 12 SP3 and SP4
- SUSE Linux Enterprise Server 15 base release and SP1
- Canonical Ubuntu Server 16.04.x LTS
- Canonical Ubuntu Server 18.04 LTS
- Debian 8 “Jessie” 8.10
- Debian 9 “Stretch” 9.4
- Linux® KVM
- Kernel.org Linux kernels 3.0 to 5.4

and for all adapters except the Xilinx Alveo U25:

- Windows® Server 2012 R2
- Windows® Server 2016
- Windows® Server 2019
- VMware® ESXi™ 6.0-u3e (and higher 6.0-uX versions) ESXi™ 6.5, ESXi™ 6.7.

Support includes all minor updates/releases/service packs of the above major releases, for which the distributor has not yet declared end of life/support.

The Solarflare accelerated network middleware, OpenOnload and EnterpriseOnload, is supported on all Linux, Ubuntu, and Debian variants listed above, and is available for all Solarflare Onload network adapters. Solarflare are not aware of any issues preventing OpenOnload installation on other Linux variants such as Centos and Fedora.

## 1.4 Solarflare AppFlex™ Technology

Solarflare AppFlex technology allows Solarflare server adapters to be selectively configured to enable on-board applications. AppFlex activation keys are required to enable selected functionality on the Solarflare XtremeScale™ and Flareon™ adapters and on the AOE ApplicationOnload™ Engine.

Customers can obtain access to AppFlex applications via their Solarflare sales channel by obtaining the corresponding AppFlex authorization code. The authorization code allows the customer to generate activation keys at the MyAppFlex page at <https://support.solarflare.com/myappflex>.

The sfkey utility application is used to install the generated activation key file on selected adapters. For detailed instructions for sfkey and key file installation refer to [Installing an activation key with sfkey on page 92](#).

## 1.5 Open Source Licenses

Solarflare network adapters include software that is published under open source licenses.

### Solarflare Boot Manager

The Solarflare Boot Manager is installed in the adapter's flash memory. This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

## Controller Firmware

The firmware running on the SFC9xxx controller includes a modified version of libcoroutine. This software is free software published under a BSD license reproduced below:

Copyright (c) 2002, 2003 Steve Dekorte

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 the author nor the names of other 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 REGENTS 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.

## 1.6 Support and Driver Download

Solarflare network drivers, RPM packages and documentation are available for download from <https://support.solarflare.com/>.

## 1.7 Regulatory Information and Approvals

Refer to [www.solarflare.com/quickstart](http://www.solarflare.com/quickstart).



## 2

# Installation

This chapter covers the following topics:

- [Solarflare Network Adapter Products on page 20](#)
- [CPU/PCIe Requirements on page 21](#)
- [Fitting a Full Height Bracket \(optional\) on page 21](#)
- [Inserting an Adapter in a PCI Express \(PCIe\) Slot on page 22](#)
- [Inserting an adapter in an OCP 2.0 mezzanine slot on page 23](#)
- [Inserting an adapter in an OCP 3.0 mezzanine slot on page 23](#)
- [Attaching a Cable \(RJ-45\) on page 25](#)
- [Attaching a Cable \(SFP+, QSFP+, SFP28, QSFP28\) on page 26](#)
- [Cables and Transceivers on page 28](#)
- [Supported Speed and Mode on page 40](#)
- [Forward Error Correction on page 42](#)
- [LED States on page 43](#)
- [Port Modes on page 43](#)
- [Single Optical Fiber - RX Configuration on page 49](#)
- [Solarflare Precision Time Synchronization Adapters on page 49](#)
- [Solarflare ApplicationOnload™ Engine on page 50.](#)



**CAUTION:** Servers contain high voltage electrical components. Before removing the server cover, disconnect the mains power supply to avoid the risk of electrocution.



**CAUTION:** Static electricity can damage computer components. Before handling computer components, discharge static electricity from yourself by touching a metal surface, or wear a correctly fitted anti-static wrist band.

## 2.1 Solarflare Network Adapter Products

### Xilinx Alveo™ adapters

- Xilinx Alveo U25 Dual-Port 10GbE/25GbE SFP28 PCIe 3.1 Server Adapter.

### Solarflare XtremeScale™ adapters

- Solarflare XtremeScale X2542 Dual-Port 100GbE QSFP28 PCIe 3.1 Server Adapter
- Solarflare XtremeScale X2541 Single-Port 100GbE QSFP28 PCIe 3.1 Server Adapter
- Solarflare XtremeScale X2522-25G Dual-Port 10GbE/25GbE SFP28 PCIe 3.1 Server Adapter
- Solarflare XtremeScale X2522 Dual-Port 10GbE SFP28 PCIe 3.1 Server Adapter
- Solarflare XtremeScale SFN8722 Dual-Port 10GbE SFP+ PCIe 3.1 OCP Server Adapter
- Solarflare XtremeScale SFN8542 Dual-Port 40GbE PCIe 3.1 QSFP+ Server Adapter
- Solarflare XtremeScale SFN8522M Dual-Port 10GbE PCIe 3.1 SFP+ Server Adapter
- Solarflare XtremeScale SFN8522 Dual-Port 10GbE PCIe 3.1 SFP+ Server Adapter
- Solarflare XtremeScale SFN8042 Dual-Port 40GbE PCIe 3.1 QSFP+ Server Adapter.

### Solarflare Flareon™ adapters

- Solarflare Flareon Ultra SFN7322F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter
- Solarflare Flareon Ultra SFN7142Q Dual-Port 40GbE PCIe 3.0 QSFP+ Server Adapter
- Solarflare Flareon Ultra SFN7124F Quad-Port 10GbE PCIe 3.0 SFP+ Server Adapter
- Solarflare Flareon Ultra SFN7122F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter
- Solarflare Flareon SFN7042Q Dual-Port 40GbE PCIe 3.0 QSFP+ Server Adapter
- Solarflare Flareon Ultra SFN7024F Quad-Port 10GbE PCIe 3.0 SFP+ Server Adapter
- Solarflare Flareon Ultra SFN7022F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter
- Solarflare Flareon SFN7004F Quad-Port 10GbE PCIe 3.0 SFP+ Server Adapter
- Solarflare Flareon SFN7002F Dual-Port 10GbE PCIe 3.0 Server I/O Adapter.

## 2.2 CPU/PCIe Requirements

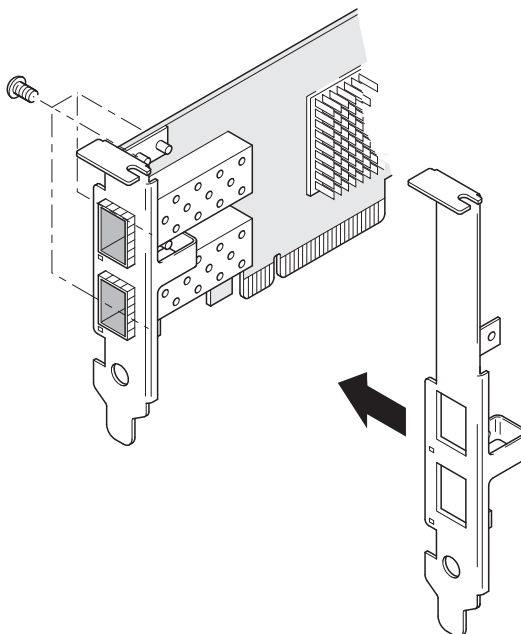
Solarflare network adapters can be installed on Intel/AMD x86 based 32 bit or 64 bit servers. The network adapter must be inserted into a PCIe x8 OR PCIe x 16 slot for maximum performance.

## 2.3 Fitting a Full Height Bracket (optional)

Solarflare adapters are supplied with a low-profile bracket fitted to the adapter. A full height bracket has also been supplied for PCIe slots that require this type of bracket.

To fit a full height bracket to the Solarflare adapter:

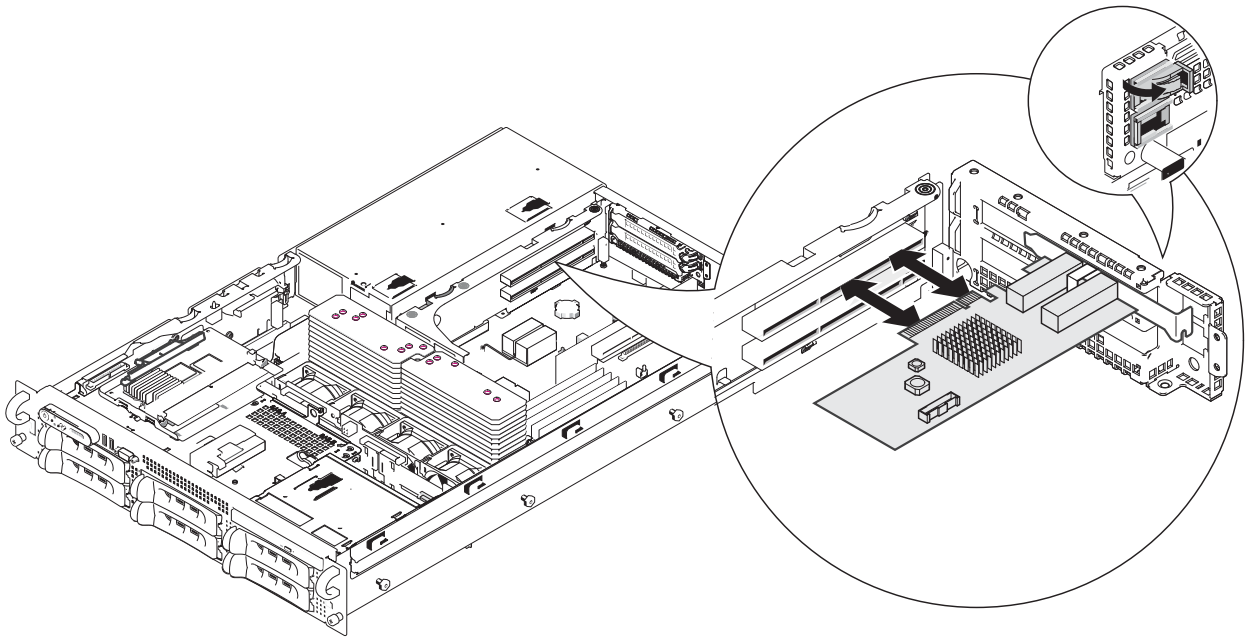
- 1 From the back of the adapter, remove the screws securing the bracket.
- 2 Slide the bracket away from the adapter.
- 3 Taking care not to over-tighten the screws, attach the full height bracket to the adapter.



## 2.4 Inserting an Adapter in a PCI Express (PCIe) Slot

To insert an adapter in a PCI Express (PCIe) slot:

- 1 Shut down the server and unplug it from the mains.
- 2 Remove the server cover to access the PCIe slots in the server.
- 3 Locate an 8-lane or 16-lane PCIe slot (refer to the server manual if necessary).
- 4 Insert the adapter.
- 5 Secure the adapter bracket in the slot.

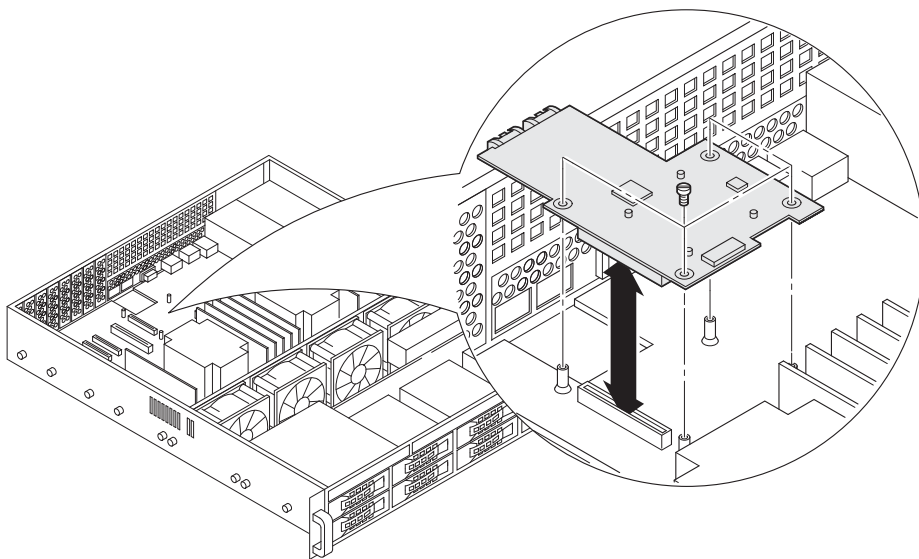


- 6 Replace the cover and restart the server.
- 7 After restarting the server, the host operating system may prompt you to install drivers for the new hardware. Click Cancel or abort the installation and refer to the relevant chapter in this manual for how to install the Solarflare adapter drivers for your operating system.

## 2.5 Inserting an adapter in an OCP 2.0 mezzanine slot

To insert an adapter in an OCP 2.0 mezzanine slot:

- 1 Shut down the server and unplug it from the mains.
- 2 Remove the server cover to access the mezzanine slot in the server.
- 3 Locate the mezzanine slot and the SFP+ port slots (refer to the server manual if necessary).
- 4 Align the SFP+ cages with the port slots and seat the adapter in the mezzanine slot.
- 5 Secure the adapter to the standoffs.



**Figure 1: Installing the OCP 2.0 Mezzanine Adapter**

- 6 Replace the cover and restart the server.
- 7 After restarting the server, the host operating system may prompt you to install drivers for the new hardware. Click Cancel or abort the installation and refer to the relevant chapter in this manual for how to install the Solarflare adapter drivers for your operating system.

## 2.6 Inserting an adapter in an OCP 3.0 mezzanine slot

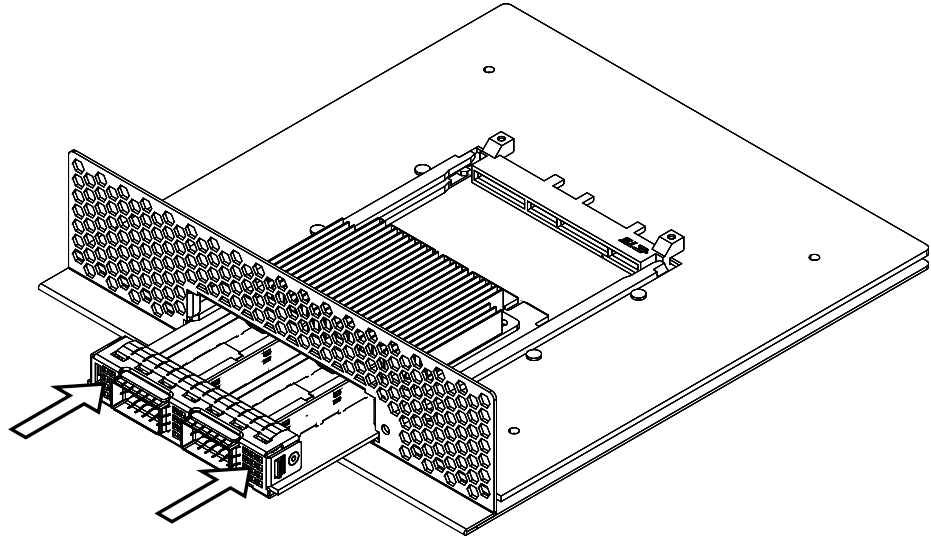
To insert an adapter in an OCP 3.0 mezzanine slot:

- 1 Shut down the server and unplug it from its power source.
- 2 Remove the server cover so you can access the internal latches for the slot in the server.
- 3 Locate the OCP 3.0 SFF slot (refer to the server manual if necessary).

- 4 Insert the Solarflare server adapter into the slot, positioning the edge of the PCB in the side rails.
- 5 **Pushing on the outer edges of the faceplate only** (as shown in the diagram below), slide the adapter into the server until it is fully engaged in the socket:



**CAUTION:** To avoid damage **do not push on the heatsink**.



- 6 Secure the adapter using the internal latches (refer to the server manual if necessary).
- 7 Replace the cover and restart the server.
- 8 After restarting the server, the host operating system may prompt you to install drivers for the new hardware. Click Cancel or abort the installation and refer to the relevant chapter in this manual for how to install the Solarflare adapter drivers for your operating system.

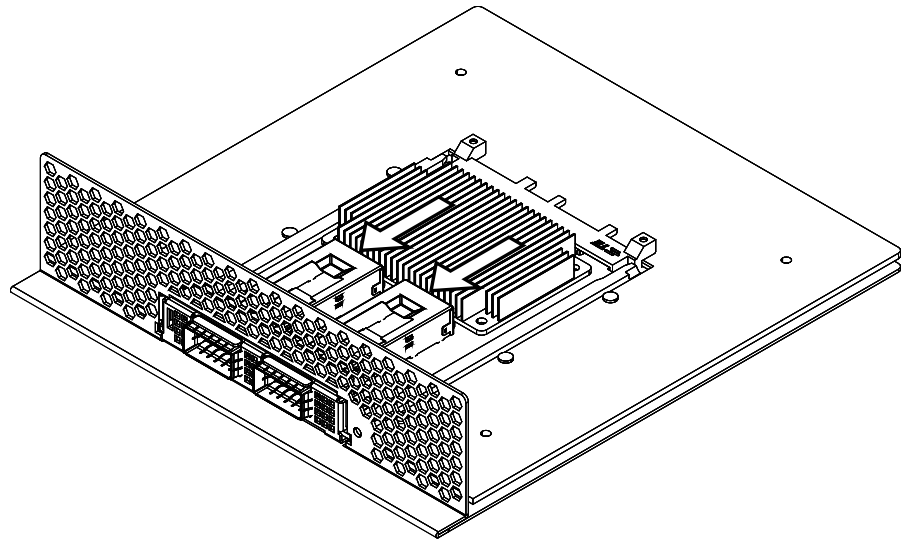
## Removing the adapter

To remove an adapter from an OCP 3.0 mezzanine slot:

- 1 Shut down the server and unplug it from its power source.
- 2 Remove the server cover so you can access the internal latches for the slot in the server.
- 3 Release the internal latches that are securing the adapter (refer to the server manual if necessary).
- 4 **Pushing on the back of the SFP cages only** (as shown in the diagram below), slide the adapter out of the server:



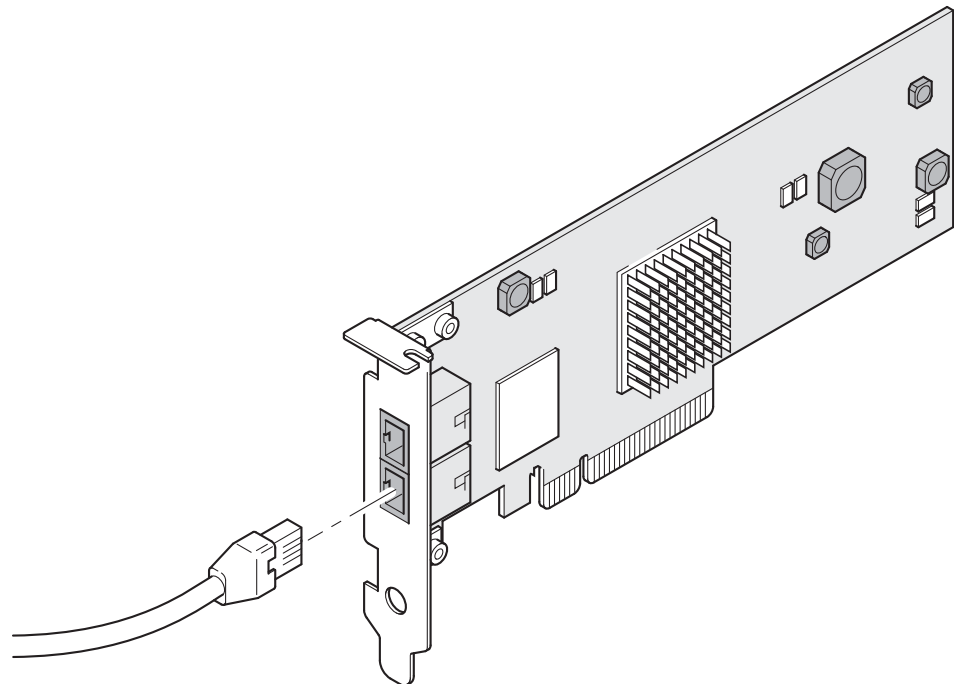
**CAUTION:** To avoid damage **do not push on the heatsink**, or pull on any cables or transceivers.



- 5 Replace the cover and restart the server.

## 2.7 Attaching a Cable (RJ-45)

Solarflare 10GBASE-T Server Adapters connect to the Ethernet network using a copper cable fitted with an RJ-45 connector (shown below).



## RJ-45 Cable Specifications

Table 1 below lists the recommended cable specifications for various Ethernet port types. Depending on the intended use, attach a suitable cable. For example, to achieve 10 Gb/s performance, use a Category 6 cable. To achieve the desired performance, the adapter must be connected to a compliant link partner, such as an IEEE 802.3an-compliant gigabit switch.

**Table 1: RJ-45 Cable Specification**

Port type	Connector	Media Type	Maximum Distance
10GBASE-T	RJ-45	Category 6A	100m (328 ft.)
		Category 6 unshielded twisted pairs (UTP)	55m (180 ft.)
		Category 5E	55m (180 ft.)
1000BASE-T	RJ-45	Category 5E, 6, 6A UTP	100m (328 ft.)
100BASE-TX	RJ-45	Category 5E, 6, 6A UTP	100m (328 ft.)

## 2.8 Attaching a Cable (SFP+, QSFP+, SFP28, QSFP28)

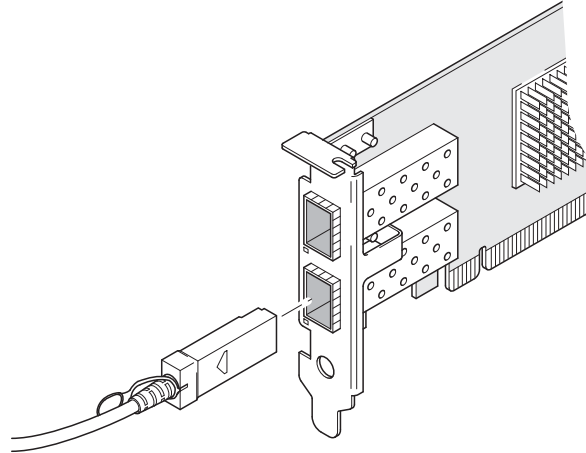
Solarflare SFP+, QSFP+, SFP28 and QSFP28 Server Adapters can be connected to the network using either a Direct Attach cable or a fiber optic cable.

### Attaching a Direct Attach Cable

To attach a Direct Attach cable:

- 1 Turn the cable so that the connector retention tab and gold fingers are on the same side as the network adapter retention clip.
- 2 Push the cable connector straight in to the adapter socket until it clicks into place.





### Removing a Direct Attach Cable

To remove a Direct Attach cable:

- 1 Pull straight back on the release ring to release the cable retention tab. Alternatively, you can lift the retention clip on the adapter to free the cable if necessary.
- 2 Slide the cable free from the adapter socket.

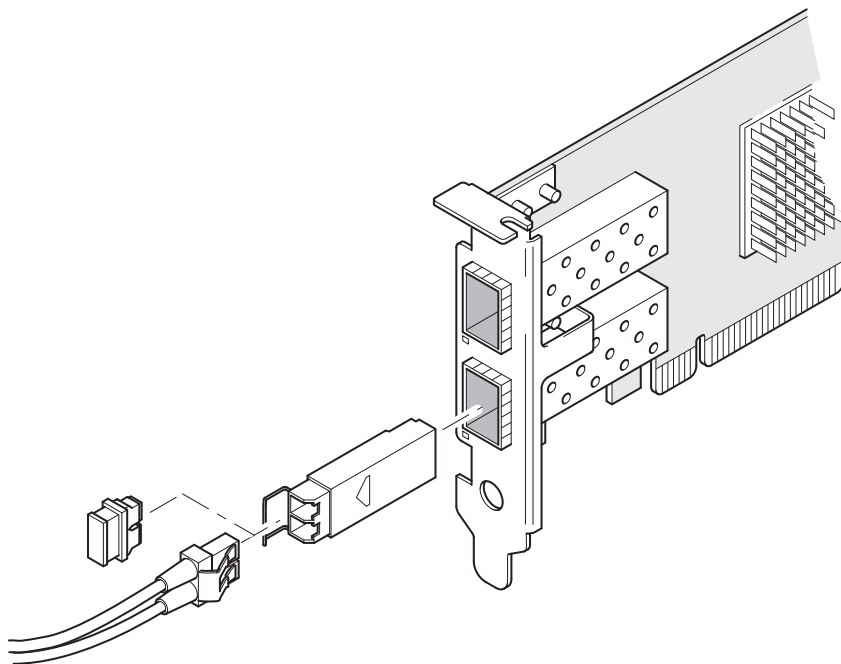
### Attaching a fiber optic cable



**WARNING:** Do not look directly into the fiber transceiver or cables as the laser beams can damage your eyesight.

To attach a fiber optic cable:

- 1 Remove and save the fiber optic connector cover.
- 2 Insert a fiber optic cable into the ports on the network adapter bracket as shown. Most connectors and ports are keyed for proper orientation. If the cable you are using is not keyed, check to be sure the connector is oriented properly (transmit port connected to receive port on the link partner, and vice versa).



### Removing a fiber optic cable



**WARNING:** *Do not* look directly into the fiber transceiver or cables as the laser beams can damage your eyesight.

To remove a fiber optic cable:

- 1 Remove the cable from the adapter bracket and replace the fiber optic connector cover.
- 2 Pull the plastic or wire tab to release the adapter bracket.
- 3 Hold the main body of the adapter bracket and remove it from the adapter.

## 2.9 Cables and Transceivers

The following tables identify adapter cables and transceiver modules that have been tested by Solarflare Communications.



**NOTE:** Cables may not work with all switches.

See:

- [QSFP28 100G Direct Attach Cables on page 29](#)
- [QSFP28 100G SR Optical Transceivers on page 30](#)
- [QSFP28 to SFP28 Breakout Direct Attach Cables on page 30](#)
- [SFP28 25G Direct Attach Cables on page 31](#)
- [SFP28 25G SR Optical Transceivers on page 32](#)
- [QSFP+ 40G Direct Attach Cables on page 33](#)

- [QSFP+ 40G Active Optical Cables on page 34](#)
- [QSFP+ 40G SR4 Optical Transceivers on page 34](#)
- [QSFP+ to SFP+ Breakout Direct Attach Cables on page 35](#)
- [QSFP+ to SFP+ Breakout Active Optical Cables on page 36](#)
- [SFP+ 10G Direct Attach Cables on page 36](#)
- [SFP+ 10G SR Optical Transceivers on page 38](#)
- [SFP+ 10G LR Optical Transceivers on page 39](#)
- [SFP 1000BASE-T Transceivers on page 39](#)
- [1G Optical Transceivers on page 40](#)
- [SFP 10GBASE-T Transceivers on page 40.](#)

## QSFP28 100G Direct Attach Cables

This is a list of supported QSFP28 direct attach cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables (of up to 5m in length) with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 2: Supported QSFP28 100G Direct Attach Cables**

Manufacturer	Product Code	Notes	X254x
Amphenol	NDAAFF-0001	1m 30AWG	✓
Amphenol	NDAAFF-0002	2m 30AWG	✓
Amphenol	NDAAFJ-0005	2m 26AWG	✓
Amphenol	NDAAFF-0003	3m 30AWG	✓
Amphenol	NDAAFJ-0002	3m 26AWG	✓
Amphenol	NDAAFJ-0004	5m 26AWG	✓
Arista	CAB-Q-Q-100G-1M	1m 30AWG	✓
Arista	CAB-Q-Q-100G-3M	3m 30AWG	✓
Arista	CAB-Q-Q-100G-5M	5m 26AWG	✓
Fiberstore	Q28-PC01	1m 26AWG	✓
Fiberstore	Q28-PC02	2m 26AWG	✓
Fiberstore	Q28-PC03	3m 26AWG	✓
Legrand	QSFP100GPDAC3M-LEG	3m 26AWG	✓

**Table 2: Supported QSFP28 100G Direct Attach Cables (continued)**

Manufacturer	Product Code	Notes	X254x
Legrand	100G2X50GPD3M-LEG	2m 28AWG	✓
ProLabs	QSFP-100G-CU3M-C	3m 28AWG	✓
ProLabs	QSFP28-2QSFP28- PDAC3M-12-34-C	3m 30AWG	✓
ProLabs	QSFP28-1QSFP28- PDAC3M-12-C	3m 30AWG	✓

## QSFP28 100G SR Optical Transceivers

This is a list of supported QSFP28 SR optical transceivers that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of transceivers with Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 3: Supported QSFP28 100G SR Optical Transceivers**

Manufacturer	Product Code	Notes	X254x
Arista	QSFP-100G-SR4		✓
Avago	AFBR-89CDDZ		✓
Finisar	FTLC9551REPM		✓
Juniper	JNP-QSFP-100G-SR4		✓

## QSFP28 to SFP28 Breakout Direct Attach Cables

This is a list of supported QSFP28 to SFP28 breakout cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables (of up to 5m in length) with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 4: Supported QSFP28 to SFP28 Breakout Direct Attach Cables**

Manufacturer	Product Code	Notes	X2522-25G	X254x
Amphenol	NDAQGF-0001	1m 30AWG (CA-N)	✓	✓
Amphenol	NDAQGJ-0002	2m 26AWG (CA-N)	✓	✓

**Table 4: Supported QSFP28 to SFP28 Breakout Direct Attach Cables (continued)**

Manufacturer	Product Code	Notes	X2522-25G	X254x
Amphenol	NDAQGF-0002	2m 30AWG (CA-S)	✓	✓
Amphenol	NDAQGF-0003	3m 30AWG (CA-L)	✓	✓
Amphenol	NDAQGJ-0005	5m 26AWG (CA-L)	✓	✓
Fiberstore	Q-4S2801	1m 30AWG	✓	✓
Legrand	100G4SFPPDAC3M-LEG	3m 28AWG		✓
Legrand	100G2X25GPD2M-LEG	2m 30AWG		✓
ProLabs	QSFP28-2SFP28-PDAC3M-13-C	3m 30AWG		✓
Siemon	Q4S28P300.5-01P	0.5m 30AWG (CA-N)	✓	✓
Siemon	Q4S28P301.0-01P	1m 30AWG (CA-N)	✓	✓
Siemon	Q4S28P301.5-01P	1.5m 30AWG (CA-N)	✓	✓
Siemon	Q4S28P302.0-01P	2m 30AWG (CA-N)	✓	✓
Siemon	Q4S28P263.0-01P	3m 26AWG (CA-N)	✓	✓
Siemon	Q4S28P265.0-01P	5m 26AWG (CA-L)	✓	✓

## SFP28 25G Direct Attach Cables

This is a list of supported SFP28 direct attach cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables (of up to 5m in length) with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 5: Supported SFP28 25G Direct Attach Cables**

Manufacturer	Product Code	Notes	X2522-25G	Alveo U25
Amphenol	NDCCGF-0006	0.5m 30AWG (CA-N)	✓	
Amphenol	NDCCGF-0001	1m 30AWG (CA-N)	✓	✓
Amphenol	NDCCGJ-0002	2m 26AWG (CA-N)	✓	
Amphenol	NDCCGJ-0003	3m 26AWG (CA-N)	✓	✓

**Table 5: Supported SFP28 25G Direct Attach Cables (continued)**

Manufacturer	Product Code	Notes	X2522-25G	Alveo U25
Amphenol	NDCCGF-0003	2m 30AWG (CA-S)	✓	✓
Amphenol	NDCCGF-0005	3m 30AWG (CA-L)	✓	✓
Amphenol	NDCCGJ-0005	5m 26AWG (CA-L)	✓	✓
Arista	CAB-S-S-25G-1M	1m 30AWG (CA-N)	✓	
Arista	CAB-S-S-25G-3M	3m 26AWG (CA-S)	✓	
Arista	CAB-S-S-25G-5M	5m 26AWG (CA-L)	✓	
Fiberstore	S28-PC01	1m 30AWG	✓	
Siemon	S1S28P301.0-01P	30AWG (CA-N)	✓	
Siemon	S1S28P302.0-01P	30AWG (CA-N)	✓	
Siemon	S1S28P265.0-01P	26AWG (CA-L)	✓	

## SFP28 25G SR Optical Transceivers

This is a list of supported SFP28 SR optical transceivers that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of transceivers with Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 6: Supported SFP28 25G SR Optical Transceivers**

Manufacturer	Product Code	Notes	X2522-25G
Cisco	SFP-25G-SR-S		✓
Fiberstore	SFP28-25GSR-85		✓
Finisar	FTLF8536P4BCL		✓
Finisar	FTLF8538P4BCL		✓

## QSFP+ 40G Direct Attach Cables

This is a list of supported QSFP+ direct attach cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables (of up to 5m in length) with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 7: Supported QSFP+ 40G Direct Attach Cables**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8x42	X254x
Arista	CAB-Q-Q-3M	3m	✓	✓	
Arista	CAB-Q-Q-5M	5m	✓		
Cisco	QSFP-H40G-CU3M	3m	✓	✓	
FCI	10093084-2010LF	1m	✓	✓	
FCI	10093084-3030LF	3m	✓	✓	
Molex	74757-1101	1m	✓		
Molex	74757-2101	1m		✓	
Molex	74757-2301	3m	✓	✓	
Panduit	40GBASE-CR4-PQSFPXA3MBU	3m		✓	
Siemon	QSFP30-01	1m	✓		
Siemon	QSFP30-03	3m	✓	✓	
Siemon	QSFP26-05	5m	✓	✓	

## QSFP+ 40G Active Optical Cables

This is a list of supported QSFP+ active optical cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 8: Supported QSFP+ 40G Active Optical Cables (AOC)**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8x42	X254x
Avago	AFBR-7QER05Z	3m	✓		
Finisar	FCBG410QB1C03	3m	✓	✓	
Finisar	FCBN410QB1C05	5m	✓		

## QSFP+ 40G SR4 Optical Transceivers

This is a list of supported QSFP+ SR4 optical transceivers that have been tested by Solarflare<sup>1</sup>. Solarflare is not aware of any issues preventing the use of other brands of transceivers with Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 9: Supported QSFP+ 40G SR4 Transceivers**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8x42	X254x
Solarflare	SFM-40G-SR4		✓		✓
Arista	AFBR-79E4Z		✓	✓	
Avago	AFBR-79EADZ		✓		
Avago	AFBR-79EIDZ		✓	✓	
Avago	AFBR-79EQDZ		✓	✓	
Avago	AFBR-79EQPZ		✓		
Finisar	FTL410QE2C		✓		
JDSU	JQP-04SWAA1		✓		
JDSU	JDSU-04SRAB1		✓		

1. Tested at standard 100m (OM3 Multimode range)



## QSFP+ to SFP+ Breakout Direct Attach Cables

QSFP+ to SFP+ breakout cables enable users to connect Solarflare dual-port QSFP+ server I/O adapters to work as a quad-port SFP+ server I/O adapters. They support 2 lanes of 10 Gb/s per QSFP+ port. The breakout cables offer a cost-effective option to support connectivity flexibility in high-speed data center applications.

This is a list of supported QSFP+ to SFP+ breakout direct attach cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables (of up to 5m in length) with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 10: Supported QSFP+ to SFP+ Breakout Direct Attach Cables**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8x42	X2522-10G	X2522-25G
Solarflare	SOLR-QSFP2SFP-1M	1m Compliant with the SFF-8431, SFF-8432, SFF-8436, SFF-8472 and IBTA Volume 2 Revision 1.3 specifications	✓			
Solarflare	SOLR-QSFP2SFP-3M	3m Compliant with the SFF-8431, SFF-8432, SFF-8436, SFF-8472 and IBTA Volume 2 Revision 1.3 specifications	✓			
Arista	CAB-Q-S-3M	3m		✓	✓	✓
Arista	CAB-Q-S-5M	5m		✓	✓	✓
Mellanox	MC2609130-003	3m		✓		
Panduit	PHQ4SFPXA1MBL	1m		✓		
Prolabs	CU1.0M-QSFP-2SFP-NS-13-C	1m		✓		
Prolabs	CU1.5M-QSFP-2SFP-NS-13-C	1.5m		✓		
Siemon	SFPPQSFP30-01	1m		✓	✓	✓
Siemon	SFPPQSFP28-03	3m		✓	✓	✓
Siemon	SFPPQSFP28-05	5m		✓	✓	✓
10GTek	CAB-QSFP.4SFP-P1M	1m		✓	✓	✓

**Table 10: Supported QSFP+ to SFP+ Breakout Direct Attach Cables (continued)**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8x42	X2522-10G	X2522-25G
10GTek	CAB-QSFP.4SFP-P3M	3m		✓	✓	✓
10GTek	CAB-QSFP.4SFP-P5M	5m		✓	✓	✓

## QSFP+ to SFP+ Breakout Active Optical Cables

QSFP+ to SFP+ breakout cables enable users to connect Solarflare dual-port QSFP+ server I/O adapters to work as a quad-port SFP+ server I/O adapters. They support 2 lanes of 10 Gb/s per QSFP+ port. The breakout cables offer a cost-effective option to support connectivity flexibility in high-speed data center applications.

This is a list of supported QSFP+ to SFP+ breakout active optical cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 11: Supported QSFP+ to SFP+ Breakout Active Optical Cables**

Manufacturer	Product Code	Notes	SFN8522M
Finisar	FCBN510QE2C07	7m Tested with Cisco NEXUS 5648Q	✓

## SFP+ 10G Direct Attach Cables

This is a list of supported SFP+ direct attach cables that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of cables (of up to 5m in length) with Solarflare network adapters. However, only cables in the table below have been fully verified and are therefore supported.

**Table 12: Supported SFP+ Direct Attach Cables**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8xxx	X2522-10G	X2522-25G
Arista	CAB-SFP-SFP-1M	1m	✓	✓	✓	✓
Arista	CAB-SFP-SFP-3M	3m	✓	✓	✓	✓
Arista	CAB-SFP-SFP-5M	5m		✓	✓	✓

**Table 12: Supported SFP+ Direct Attach Cables (continued)**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8xxx	X2522-10G	X2522-25G
Cisco	SFP-H10GB-CU1M	1m	✓	✓	✓	✓
Cisco	SFP-H10GB-CU3M	3m	✓	✓	✓	✓
Cisco	SFP-H10GB-CU5M	5m	✓	✓	✓	✓
HP	J9283A/B Procurve	3m	✓	✓		
HPE	MergeOptics GmbH 10119467-3030LF	3m			✓	✓
Juniper	EX-SFP-10GE-DAC-1m	1m	✓	✓		
Juniper	EX-SFP-10GE-DAC-3m	3m	✓	✓		
Molex	74752-1101	1m	✓	✓	✓	✓
Molex	74752-2301	3m	✓	✓	✓	✓
Molex	74752-3501	5m	✓	✓	✓	✓
Molex	74752-9093	1m	✓			
Molex	74752-9094	3m	✓			
Molex	74752-9096	5m	✓			
Panduit	PSF1PXA1M	1m	✓	✓	✓	✓
Panduit	PSF1PXA3M	3m	✓	✓	✓	✓
Panduit	PSF1PXD5MBU	5m	✓	✓	✓	✓
Siemon	SFPP30-01	1m	✓	✓	✓	✓
Siemon	SFPP30-02	2m	✓	✓		
Siemon	SFPP30-03	3m	✓	✓		
Siemon	SFPP28-05	5m	✓	✓	✓	✓
Tyco	2032237-2 D	1m	✓	✓		
Tyco	2032237-4	3m		✓		

## SFP+ 10G SR Optical Transceivers

This is a list of supported SFP+ SR optical transceivers that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of transceivers with Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 13: Supported SFP+ 10G SR Optical Transceivers**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8xxx	X2522-10G	X2522-25G
Solarflare	SFM-10G-SR	10G	✓	✓	✓	✓
Arista	SFP-10G-SR	10G	✓			
Arista	XVR-00002-02	10G		✓	✓	✓
Arista	XVR-10002-20	10G		✓		
Avago	AFBR-703SDZ	10G	✓	✓	✓	✓
Avago	AFBR-703SDDZ	Dual speed 1G/10G optic.	✓			
Avago	AFBR-703SMZ	10G	✓			
Avago	AFBR-709SMZ-SF1	10G		✓		
DELL	PLRXPL-SC-S43-811	10G			✓	
Fibrestore	10GBASE-SR-CO	10G			✓	✓
Finisar	FTLX8571D3BCL	10G	✓	✓	✓	✓
Finisar	FTLX8571D3BCV	Dual speed 1G/10G optic.	✓	✓		
Finisar	FTLX8574D3BCL	10G	✓	✓	✓	✓
HP	456096-001		✓	✓		
HP	455883-B21		✓	✓		
HP	455885-001		✓	✓		
Intel	AFBR-703SDZ	10G	✓	✓		
JDSU	PLRXPL-SC-S43-22-N	10G	✓			
Juniper	AFBR-700SDZ-JU1	10G	✓			
MergeOptics	TRX10GVP2010	10G	✓	✓	✓	✓
Vorboss	VBO-PXG-SR-300	10G	✓			

## SFP+ 10G LR Optical Transceivers

This is a list of supported SFP+ LR optical transceivers that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of transceivers with Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 14: Supported SFP+ 10G LR Optical Transceivers**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8xxx	X2522-10G	X2522-25G
Avago	AFCT-701SDZ	10G single mode fiber	✓			
Finisar	FTLX1471D3BCL	10G single mode fiber	✓		✓	✓

## SFP 1000BASE-T Transceivers

This is a list of supported SFP 1000BASE-T transceivers that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of transceivers with the Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 15: Supported SFP 1000BASE-T Transceivers**

Manufacturer	Product Code	Notes	SFN7xxx	SFN8xxx	X2522-10G	X2522-25G
Arista	SFP-1G-BT		✓			
Avago	ABCU-5710RZ		✓	✓	✓	✓
Cisco	30-1410-03		✓			
Dell	FCMJ-8521-3-(DL)		✓	✓	✓	✓
Finisar	FCLF-8521-3		✓	✓		
Finisar	FCMJ-8521-3		✓			
Finisar	FCLF8522P2BTL			✓	✓	✓
HP	453156-001		✓	✓	✓	✓
HP	453154-B21		✓	✓		
3COM	3CSFP93		✓	✓		

## 1G Optical Transceivers

This is a list of supported 1G transceivers that have been tested by Solarflare. Solarflare is not aware of any issues preventing the use of other brands of transceivers with Solarflare network adapters. However, only transceivers in the table below have been fully verified and are therefore supported.

**Table 16: Supported 1G Optical Transceivers**

Manufacturer	Product Code	Type	SFN7xxx	SFN8xxx	X2522-10G	X2522-25G
Avago	AFBR-5710PZ	1000Base-SX	✓			
Cisco	GLC-LH-SM	1000Base-LX/LH	✓			
Cisco	30-1299-01	1000Base-LX		✓		
Finisar	FTLF8519P2BCL	1000Base-SX	✓	✓		
Finisar	FTLF8519P3BNL	1000Base-SX	✓			
Finisar	FTLF1318P2BCL	1000Base-LX	✓			
Finisar	FTLF1318P3BTL	1000Base-LX	✓	✓		
HP	453153-001	1000Base-SX	✓			
HP	453151-B21	1000Base-SX	✓			

## SFP 10GBASE-T Transceivers

Solarflare adapters do not support 10GBASE-T transceiver modules.

## 2.10 Supported Speed and Mode

Solarflare network adapters support either SFP+, QSFP+, SFP28 or QSFP28 standards. The table below summarizes the speeds supported by adapters:

Standard	Auto neg speed	Speed	Comment
QSFP28	Yes	100G, 50G, 40G, 25G or 10G	X2-series adapters (X2541, X2542)
SFP28	Yes	25G, 10G or 1G	X2-series and U25 adapters

Standard	Auto neg speed	Speed	Comment
QSFP+	No	40G or 10G	8000- and 7000-series adapters (SFN8542, SFN8042, SFN7142Q, SFN7042Q)
SFP+	No	10G or 1G	8000- and 7000-series adapters

The speed(s) available are those that are supported by both the network adapter, and the transceiver module being used:

- QSFP28 modules typically operate at 100Gbps or 40Gbps.
- SFP28 modules typically operate at 25Gbps or 10Gbps.
- QSFP+ modules typically only operate at 40Gbps.
- SFP+ modules typically only operate at 10Gbps.  
Some SFP+ optical modules are dual speed. These run at the maximum (10G) link speed unless explicitly configured to operate at a lower speed (1G).
- SFP optical modules typically operate at 1Gbps.
- SFP 1000BASE-T modules operate at 1Gbps. They will not link at 100Mbps.

## Setting the speed

X2-series and U25 adapters typically set the speed automatically, whereas older adapters require manual setting.

### X2-series and U25 adapters

X2-series and U25 adapters have Auto-Negotiation (AN) enabled by default. They automatically detect the link speed and configuration of the link partner (switch), and typically no configuration is necessary:

- If the link partner/switch also has AN enabled, the adapter will determine this from the AN protocol, and use this to set the speed.
- If auto-negotiation is disabled or fails, the adapter will instead analyze the received signal using 'parallel detect', and establish a link at the highest available speed.

Parallel detect is required on 10G transceivers and for some switches that do not support AN/LT.

Solarflare recommends, in the majority of cases, to configure the link properties on the connected switch port and leave the X2 series or U25 adapter in its default state. However, the adapter can also be configured manually if this is required, as described in [Port Modes on page 43](#).



**NOTE:** 25G links (or  $n \times 25\text{G}$  for QSFP28) are only attempted when the DAC cable or transceiver module is rated for that speed. For example, 25G links speed will not be attempted when a using a standard 10G DAC cable/transceiver.

### 8000- and 7000-series adapters

On 8000- and 7000-series adapters, the speed must be set manually. For more information on default speeds and how to change them, see [Port Modes on page 43](#).



**NOTE:** AN/LT is typically present on 40G links, and is supported by 8000- and 7000-series adapters.

## 2.11 Forward Error Correction

On 25G links (or  $n \times 25\text{G}$  for QSFP28), Forward Error Correction (FEC) is used. This employs redundancy in the channel coding as a technique to lower the bit error rate (BER) in noisy or unreliable communications channels, and over long cables. The receiver is able to detect and correct errors without the need for a reverse channel or data re-transmission.



**NOTE:** FEC can potentially impact latency with an additional error correction overhead of a few hundred nanoseconds.

Auto-negotiation is used to establish whether to use FEC, and what type of FEC to apply on a link. This can be overridden with manual configuration:

- For Linux, see [Configuring FEC on page 60](#)
- For Windows, see [Configuring FEC on page 149](#)
- For VMware, see [Configuring FEC on page 204](#).

### Direct Attach Cables

Direct attach cables have the following FEC requirements:

DAC Cable	FEC Requirement
CA-25G-N up to 3m	Can work with no FEC (the default), BASE-R FEC or RS-FEC.
CA-25G-S up to 3m	Requires either BASE-R FEC (the default) or RS-FEC.
CA-25G-L up to 5m	Requires RS-FEC

### Optical Cables

Optical cables do not support auto-negotiation, and use RS-FEC by default.



## 2.12 LED States

There are two LEDs on the Solarflare network adapter transceiver module. LED states are as follows

**Table 17: LED States**

Adapter Type	LED Description	State
QSFP+, SFP/ SFP+, SFP28	Speed	Green (solid) at all speeds
	Activity	Flashing green when network traffic is present
		LEDs are OFF when there is no link present
BASE-T	Speed	Green (solid) 10Gbps Yellow (solid) 100/1000Mbps
	Activity	Flashing green when network traffic is present
		LEDs are OFF when there is no link present

## 2.13 Port Modes

Port modes are configured using the `sfboot` utility available for Linux or ESXi systems and the `SfConfig` utility on Windows systems.

The `port-mode` is a GLOBAL option and applies to all ports on the adapter.

A server reboot (power off/on) is required following changes to port modes.

Adapter Model	port-mode
<i>SFN7000 series</i>	
7x22	[1x10G], [1x10G][1x10G]*
7x24	[1x10G][1x10G], [4x10G]*
7x42	[1x10G][1x10G], [4x10G], [1x40G][1x40G]*
<i>SFN8000 series</i>	
8x22	[1x10G][1x10G]*
8x41	[1x40G]*
8x42	[4x10G], [2x10G][2x10G], [1x40G][1x40G]*

Adapter Model	port-mode
<i>X2 series</i>	
X2522	[1x10/25G][1x10/25G]*
X2541	[4x10/25G], [2x50G], [1x100G]*
X2542	[4x10/25G], [2x10/25G][2x10/25G], [2x50G], [1x50G][1x50G]*, [1x100G]
<i>Alveo series</i>	
U25	[1x10/25G][1x10/25G]*
The mode annotated with * is the Default port mode for that model	

Port modes are illustrated in the sections below:

- [SFN7x42Q QSFP+ Adapters on page 45](#)
- [SFN8x42 QSFP+ Adapters on page 46](#)
- [X2542 and X2541 QSFP28 adapters on page 47](#)
  - [X2542, X2541 – \[1x100G\] on page 47](#)
  - [X2542, X2541 – \[4x10/25G\] on page 47](#)
  - [X2542, X2541 – \[2x50G\] on page 48](#)
  - [X2542 – \[1x50G\]\[1x50G\] on page 48](#)
  - [X2542 – \[2x10/25G\]\[2x10/25G\] on page 49.](#)

## SFN7x42Q QSFP+ Adapters

SFN7x42Q adapters can operate as

- 1 x 40Gbps per QSFP+ port  
sfboot port-mode=[1x40G][1x40G]
- 2 x 10Gbps per QSFP+ port  
sfboot port-mode=[4x10G]
- 1 x 10Gbps per QSFP+ port  
sfboot port-mode=[1x10G][1x10G]
- A configuration of 1 x 40G and 2 x 10G ports is not supported.

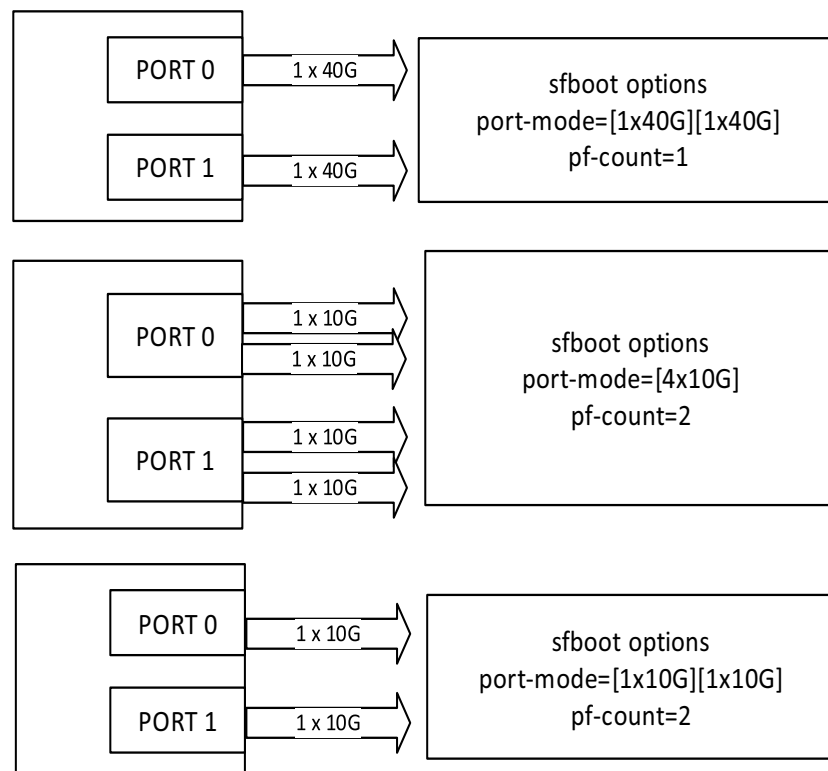


Figure 2: Port Configuration: SFN7x42Q

### BreakOut Cables

The Solarflare 40G breakout cable has only 2 physical cables. Cables from other suppliers may have 4 physical cables. When connecting a third party breakout cable to the SFN7x42Q 40G QSFP+ cage (in 10G mode), **only cables 1 and 3 are active**.

## SFN8x42 QSFP+ Adapters

SFN8x42 adapters can operate as

- 1 x 40G per QSFP+ port  
sfboot port-mode=[1x40G][1x40G]
- 4 x 10G on one of the QSFP+ ports – 2nd cage is disabled  
sfboot port-mode=[4x10G]
- 2 x 10G per QSFP+ port  
sfboot port-mode=[2x10G][2x10G]
- A configuration of 1 x 40G and 4 x 10G ports is not supported.

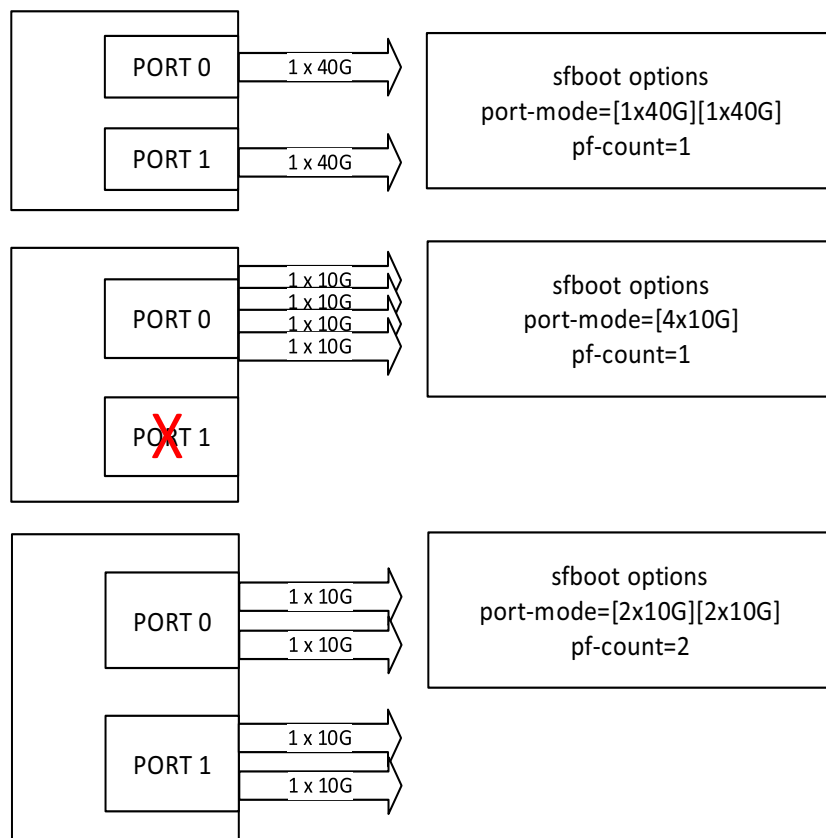


Figure 3: Port Configuration: SFN8x42

## X2542 and X2541 QSFP28 adapters

The Solarflare X2541 and X2542 adapters have 4 x 10G/25G network ports, 2 x 50G ports or a single 100G port and can operate at 1G, 10G, 25G, 40G, 50G and 100G.

### X2542, X2541 – [1x100G]

- 1 x 100G on the first QSFP28 port. On the X2542 adapter, the 2nd cage is disabled.
- Exposes a single PCIe PF to the OS.

`sfboot port-mode=[1x100G] pf-count=1`

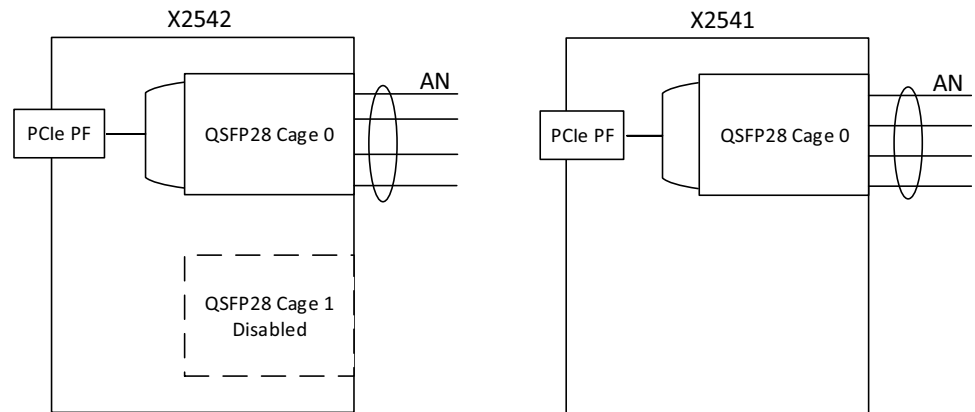


Figure 4: X254x – [1x100G]

### X2542, X2541 – [4x10/25G]

- Uses one QSFP28 port as 4 separate SFP28 ports. On the X2542 adapter, the 2nd cage is disabled.
- Four PCIe PFs exposed to the OS – each port can operate at 1G, 10G or 25G.

`sfboot-mode=[4x10/25G] pf-count=4`

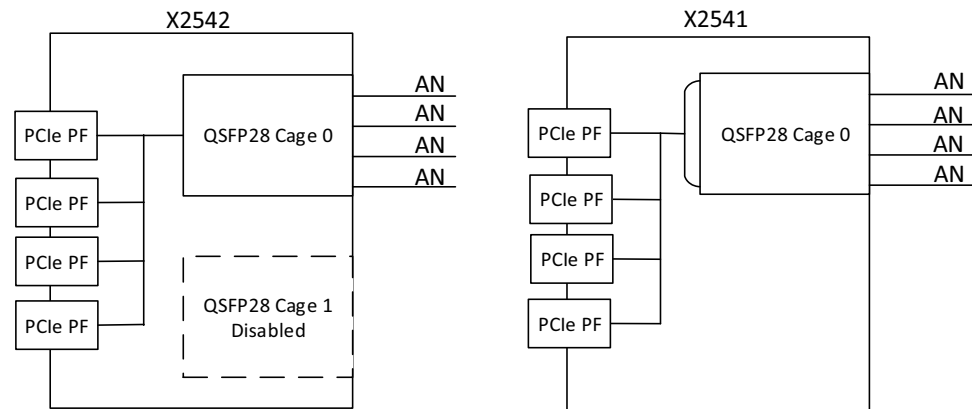


Figure 5: X254x – [4x10/25G]

### X2542, X2541 – [2x50G]

- X2 supports 2 x 50G MACs – each MAC needs 2 x 25G lanes.
- Uses one physical QSFP28 port as 2 x 50G ports. On the X2542 adapter, the 2nd cage is disabled.
- Two PCIe PFs exposed to the OS – each port can operate at 1G, 10G, 25G or 50G.

sfboot port-mode=[2x50G] pf-count=2

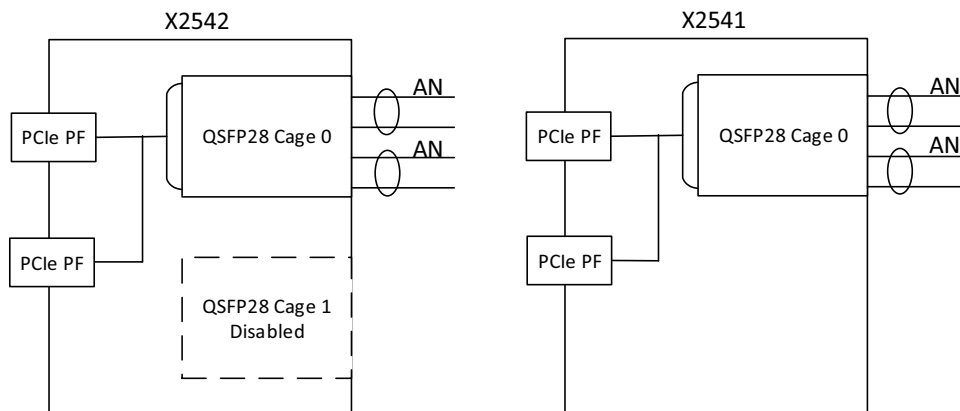


Figure 6: X254x – [2x50G]

### X2542 – [1x50G][1x50G]

- Uses both QSFP28 ports.
- Two PCIe PFs exposed to the OS – each port can operate at 1G, 10G, 25G or 50G.
- NIC does not advertise 40G or 100G on either port.

sfboot port-mode=[1x50G][1x50G] pf-count=2

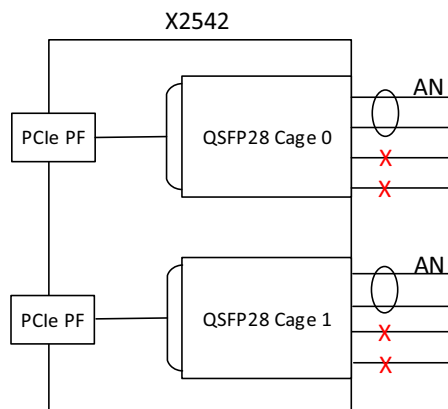


Figure 7: X2542 – [1x50G][1x50G]

### X2542 – [2x10/25G][2x10/25G]

- Uses both QSFP28 ports.
- Four PCIe PFs exposed to the OS – each port can operate at 1G, 10G or 25G.

sfboot port-mode=[2x10/25G][2x10/25G] pf-count=2

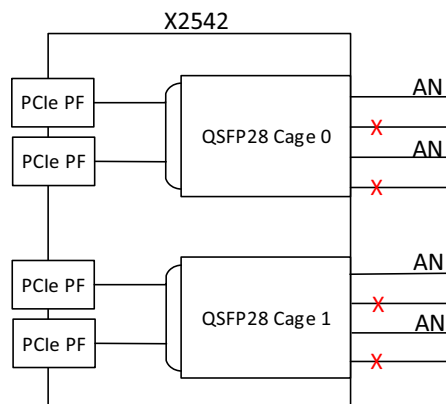


Figure 8: X2542 – [2X10/25G][2x10/25G]

## 2.14 Single Optical Fiber - RX Configuration

The Solarflare adapter will support a receive (RX) only fiber cable configuration when the adapter is required only to receive traffic, but have no transmit link. This can be used, for example, when the adapter is to receive traffic from a fiber tap device.

Solarflare have successfully tested this configuration on a 10G link on Flareon series and XtremeScale series adapters when the link partner is configured to be TX only (this will always be the case with a fiber tap). Some experimentation might be required when splitting the light signal to achieve a ratio that will deliver sufficient signal strength to all endpoints.



**NOTE:** Solarflare adapters do not support a receive only configuration on 1G links.

## 2.15 Solarflare Precision Time Synchronization Adapters

Solarflare adapters can generate hardware timestamps for PTP packets in support of a network precision time protocol deployment compliant with the IEEE 1588-2008 specification.

Some adapters require an additional AppFlex activation key to enable PTP/HW timestamping.

Customers requiring configuration instructions for these adapters and Solarflare PTP in a PTP deployment should refer to the *Solarflare Enhanced PTP User Guide* (SF-109110-CD).

## 2.16 Solarflare ApplicationOnload™ Engine

The ApplicationOnload™ Engine (AOE) SFA7942Q is a half-length, full-height PCIe form factor adapter combining the ultra-low latency dual-port 40GbE adapter with an Altera Stratix V FPGA.

For details of the SFA7942Q adapter refer to the *Solarflare ApplicationOnload Users Guide* (SF-115020-CD).



# 3

## Solarflare Adapters on Linux

This chapter covers the following topics on the Linux® platform:

- [System Requirements on page 52](#)
- [Linux Platform Driver Feature Set on page 52](#)
- [About the In-tree Driver on page 54](#)
- [Getting the Adapter Driver on page 54](#)
- [Installing the DKMS RPM on page 55](#)
- [Installing the Source RPM on page 56](#)
- [Configuring the Solarflare Adapter on page 58](#)
- [Setting Up VLANs on page 61](#)
- [Setting Up Teams on page 61](#)
- [NIC Partitioning on page 62](#)
- [NIC Partitioning with SR-IOV on page 67](#)
- [Receive Side Scaling \(RSS\) on page 70](#)
- [Receive Flow Steering \(RFS\) on page 72](#)
- [Transmit Packet Steering \(XPS\) on page 74](#)
- [Linux Utilities RPM on page 76](#)
- [Configuring the Boot Manager with sfbboot on page 78](#)
- [Upgrading Adapter Firmware with sfupdate on page 86](#)
- [Installing an activation key with sfkey on page 92](#)
- [Performance Tuning on Linux on page 95](#)
- [Web Server - Driver Optimization on page 103](#)
- [Interrupt Affinity on page 105](#)
- [Module Parameters on page 115](#)
- [Linux ethtool Statistics on page 117](#)
- [Reading sensors on page 126](#)
- [Driver Logging Levels on page 128](#)
- [Running Adapter Diagnostics on page 129](#)
- [Running Cable Diagnostics on page 130.](#)

## 3.1 System Requirements

Refer to [Software Driver Support on page 16](#) for supported Linux Distributions.

## 3.2 Linux Platform Driver Feature Set

The table below shows the feature set for the Linux platform driver.

**Table 18: Linux Feature Set**

<b>Fault diagnostics</b>	<p>Support for comprehensive adapter and cable fault diagnostics and system reports.</p> <ul style="list-style-type: none"> <li>• See <a href="#">Linux Utilities RPM on page 76</a></li> </ul>
<b>Firmware updates</b>	<p>Support for Boot ROM, Phy transceiver and adapter firmware upgrades.</p> <ul style="list-style-type: none"> <li>• See <a href="#">Upgrading Adapter Firmware with sfupdate on page 86</a></li> </ul>
<b>Hardware Timestamps</b>	<p>The Xilinx Alveo U25 adapter, Solarflare XtremeScale X2522, SFN8542-Plus, SFN8522-Plus and SFN8042<sup>1</sup> adapters, and Solarflare Flareon SFN7322F, SFN7142Q<sup>1</sup>, SFN7124F<sup>1</sup>, SFN7122F<sup>1</sup>, SFN7042Q<sup>1</sup>, SFN7024F<sup>1</sup>, SFN7022F<sup>1</sup> adapters support the hardware timestamping of all received packets - including PTP packets.</p> <p>The Linux kernel must support the SO_TIMESTAMPING socket option (2.6.30+) to allow the driver to support hardware packet timestamping. Therefore hardware packet timestamping is not available in RHEL 5.</p>
<b>Jumbo frames</b>	<p>Support for MTUs (Maximum Transmission Units) from 1500 bytes to 9216 bytes.</p> <ul style="list-style-type: none"> <li>• See <a href="#">Configuring Jumbo Frames on page 60</a></li> </ul>
<b>PXE and UEFI booting</b>	<p>Support for diskless booting to a target operating system via PXE or UEFI boot.</p> <ul style="list-style-type: none"> <li>• See <a href="#">Configuring the Boot Manager with sfboot on page 78</a></li> <li>• See <a href="#">Solarflare Boot Manager on page 257</a></li> </ul> <p>PXE or UEFI boot are not supported for Solarflare adapters on IBM System p servers.</p>
<b>Receive Side Scaling (RSS)</b>	<p>Support for RSS multi-core load distribution technology.</p> <ul style="list-style-type: none"> <li>• See <a href="#">Receive Side Scaling (RSS) on page 70</a>.</li> </ul>

**Table 18: Linux Feature Set (continued)**

<b>ARFS</b>	Linux Accelerated Receive Flow Steering.  Improve latency and reduce jitter by steering packets to the core where a receiving application is running.  See <a href="#">Receive Flow Steering (RFS) on page 72</a> .
<b>Transmit Packet Steering (XPS)</b>	Supported on Linux 2.6.38 and later kernels. Selects the transmit queue when transmitting on multi-queue devices.  See <a href="#">Transmit Packet Steering (XPS) on page 74</a> .
<b>NIC Partitioning</b>	Each physical port on the adapter can be exposed as up to 8 PCIe Physical Functions (PF).  See <a href="#">NIC Partitioning on page 62</a> .
<b>SR-IOV</b>	Support for Linux KVM SR-IOV.  <ul style="list-style-type: none"> <li>See <a href="#">SR-IOV Virtualization Using KVM on page 224</a></li> </ul> SR-IOV is not supported for Solarflare adapters on IBM System p servers.
<b>Task offloads</b>	Support for TCP Segmentation Offload (TSO), Large Receive Offload (LRO), and TCP/UDP/IP checksum offload for improved adapter performance and reduced CPU processing requirements.  <ul style="list-style-type: none"> <li>See <a href="#">Configuring Task Offloading on page 59</a></li> </ul>
<b>TX PIO</b>	Use of programmed IO buffers in order to reduce latency for small packet transmission.  <ul style="list-style-type: none"> <li>See <a href="#">TX PIO on page 102</a>.</li> </ul>
<b>CTPIO</b>	Cut Through PIO - TX packets are streamed directly from the PCIe interface to the adapter port bypassing the main TX datapath to deliver lowest TX latency.  For details refer to the Onload User Guide (SF-104474-CD).
<b>Teaming</b>	Improve server reliability and bandwidth by combining physical ports, from one or more Solarflare adapters, into a team, having a single MAC address and which function as a single port providing redundancy against a single point of failure.  <ul style="list-style-type: none"> <li>See <a href="#">Setting Up Teams on page 61</a></li> </ul>
<b>Virtual LANs (VLANs)</b>	Support for multiple VLANs per adapter.  <ul style="list-style-type: none"> <li>See <a href="#">Setting Up VLANs on page 61</a></li> </ul>

- Requires an AppFlex activation key - for details refer to [Solarflare AppFlex™ Technology on page 17](#).

### 3.3 About the In-tree Driver



**CAUTION:** Linux (and Linux based OS distributions) already include a version of the Solarflare adapter driver. This is known as the OS ‘in-tree’ driver.

The in-tree driver is good for normal networking operation, but does not support the following operations:

- hardware timestamping
- sfptpd (Solarflare PTP daemon)
- sfkey (feature activation key utility)
- sfctool
- other more advanced/recent features available with later drivers.

To identify if the in-tree driver is being used by the adapter:

```
# ethtool -i <interface>
driver: sfc
version: 4.0
```

The in-tree driver is being used if the version is reported as either 4.0 or 4.1.



**CAUTION:** A later version of the driver can be installed without removing the in-tree driver, but the in-tree driver will automatically reload unless the initramfs is rebuilt using the `dracut -f` command after a later version driver is installed and loaded.

### 3.4 Getting the Adapter Driver

The Solarflare adapter driver can be installed from the following packages:

- Source DKMS (SF-104979-LS)
- Source RPM (SF-103848-LS)
- OpenOnload or EnterpriseOnload distribution.

These packages are available from: [support.solarflare.com](http://support.solarflare.com).



**NOTE:** Solarflare recommend that the DKMS driver package is installed on the Ubuntu server and *not* the source RPM package.



**NOTE:** Onload users only need to install the Onload package - there is no need to install the driver again from RPM or DKMS.

## 3.5 Installing the DKMS RPM

Dynamic Kernel Module Support is a framework where device driver source can reside outside the kernel source tree. This supports an easy method to rebuild modules when kernels are upgraded.

### Requirements

DKMS must be installed on the server. If the following command returns nothing, then DKMS is not installed. Refer to Linux online documentation to install DKMS.

```
# dkms --version
```

### Install the driver

To install the Solarflare driver DKMS package:

- If you *are not* using an Ubuntu/Debian server:

**a)** Execute the following command:

```
# rpm -i sfc-dkms-<version>.noarch.rpm
```

- If you *are* using an Ubuntu/Debian server:

**a)** Create the .deb file:

```
sudo alien -c sfc-dkms-<version>.sf.1.noarch.rpm
```

This command generates the sfc-dkms\_<version>\_all.deb file.



**NOTE:** The -c option is required to convert source scripts and build the driver.

**b)** Install the deb file:

```
sudo dpkg -i -dkms_<version>_all.deb
```

### Reload the Driver

```
modprobe -r sfc
modprobe sfc
```

### Confirm

To check the adapter is using the newly installed driver (check version):

```
# ethtool -i <interface>
```

## 3.6 Installing the Source RPM

To install the driver from the source RPM package, the binary driver must be built from the source RPM, then installed and then loaded.

### Requirements

Kernel headers for the running kernel must be installed at `/lib/modules/<kernel-version>/build`:

- On Red Hat systems, install the appropriate `kernel-smp-devel` or `kernel-devel` package
- On SUSE systems install the `kernel-source` package.

### Additional SUSE requirements

The adapter drivers are currently classified as 'unsupported' by SUSE Enterprise Linux:

- To allow unsupported drivers to load in SLES 10:
  - a) Edit the following file:  
`/etc/sysconfig/hardware/config`
  - b) Find the line:  
`LOAD_UNSUPPORTED_MODULES_AUTOMATICALLY=no`
  - c) Change no to yes :  
`LOAD_UNSUPPORTED_MODULES_AUTOMATICALLY=yes`
- For SLES 11 or later:
  - a) Edit the unsupported modules file in:  
`/etc/modprobe.d/unsupported-modules`
  - b) Find the line:  
`allow_unsupported_modules 0`
  - c) Change 0 to 1:  
`allow_unsupported_modules 1`

### Build the Binary RPM

To build the binary RPM:



**NOTE:** This is an example, and version numbers for your distribution might be different.

- 1 Copy the driver distribution package to the server and unzip to reveal the source RPM file:

```
# unzip SF-103848-LS-46_Solarflare_NET_driver_source_RPM.zip
Archive: SF-103848-LS-46_Solarflare_NET_driver_source_RPM.zip
  inflating: sfc-4.13.1.1034-1.src.rpm
```

**2** Build the binary:

```
# rpmbuild --rebuild sfc-4.13.1.1034-1.src.rpm
```

**3** The build procedure will generate a lot of console output. Towards the end of the build a **'Wrote'** line identifies the location of the built binary driver file:

```
Wrote: /root/rpmbuild/RPMS/x86_64/kernel-module-sfc-RHEL7-3.10.0-514.26.2.el7.x86_64-4.13.1.1034-1.x86_64.rpm
```

## Install the Binary RPM

Copy the location from the previous build step to install the binary driver:

```
# rpm -ivh /root/rpmbuild/RPMS/x86_64/kernel-module-sfc-RHEL7-3.10.0-514.26.2.el7.x86_64-4.13.1.1034-1.x86_64.rpm
```

## Reload the Driver

```
modprobe -r sfc
modprobe sfc
```

## Confirm

To check the adapter is using the newly installed driver (check version):

```
# ethtool -i <interface>
```

### Using YaST on SUSE

On SUSE, YaST is used to configure the adapter. When the Ethernet Controller is selected, the **Configuration Name** will take one of the following forms:

- eth-bus-pci-dddd:dd:dd.N where N is either 0 or 1.
- eth-id-00:0F:53:XX:XX:XX

Once configured, the **Configuration Name** for the correct Ethernet Controller will change to the second form, and an ethX interface will appear on the host.

If the incorrect Ethernet Controller is chosen and configured, then the **Configuration Name** will remain as eth-bus-pci-dddd:dd:dd.1 after configuration by YaST, and an ethX interface will not appear on the system. If this happens, you should remove the configuration for this Ethernet Controller, and configure the other Ethernet Controller of the pair.

## Building for a different kernel

To build for a different kernel to the running system, enter the following command to identify the target kernel.

```
# rpmbuild --define 'kernel <kernel version>' --rebuild <package_name>
```

## 3.7 Configuring the Solarflare Adapter

Ethtool is a standard Linux tool to set, view and change Ethernet adapter settings.

```
ethtool <-option> <interface>
```

Root permissions are required to configure the adapter.

### Hardware Timestamps

The Solarflare Flareon series and XtremeScale series adapters can support hardware timestamping for all received network packets.

The Linux kernel must support the SO\_TIMESTAMPING socket option (2.6.30+) therefore hardware packet timestamping is not supported on RHEL 5.

For more information about using the kernel timestamping API, users should refer to the Linux documentation: <http://lxr.linux.no/linux/Documentation/networking/timestamping.txt>

### Configuring Speed and Modes

Solarflare adapters by default automatically negotiate the connection speed to the maximum supported by the link partner.

- On the 10GBASE-T adapters “auto” instructs the adapter to negotiate the highest speed supported in common with its link partner.
- On SFP28, SFP+ adapters, “auto” instructs the adapter to use the highest link speed supported by the inserted SFP+ module.

On 10GBASE-T and SFP+ adapters, any other value specified will fix the link at that speed, regardless of the capabilities of the link partner, which may result in an inability to establish the link. Dual speed SFP+ modules operate at their maximum (10G) link speed unless explicitly configured to operate at a lower speed (1G).

The following commands demonstrate ethtool to configure the network adapter Ethernet settings.

- Identify interface configuration settings:  
`ethtool ethX`
- Set link speed:  
`ethtool -s ethX speed 1000|100`
- To return the connection speed to the default auto-negotiate, enter:  
`ethtool -s <ethX> autoneg on`
- Configure auto negotiation:  
`ethtool -s ethX autoneg [on|off]`
- Set auto negotiation advertised speed 1G:  
`ethtool -s ethX advertise 0x20`



- Set autonegotiation advertised speed 10G:  
`ethtool -s ethX advertise 0x1000`
- Set autonegotiation advertised speeds 1G and 10G:  
`ethtool -s ethX advertise 0x1020`
- Identify interface auto negotiation pause frame setting:  
`ethtool -a ethX`
- Configure auto negotiation of pause frames:  
`ethtool -A ethX autoneg on [rx on|off] [tx on|off]`



**NOTE:** Due to a limitation in ethtool, when auto-negotiation is enabled, the user must specify both speed and duplex mode or speed and set an advertise mask otherwise speed configuration will not function.

## Configuring Task Offloading

Solarflare adapters support transmit (Tx) and receive (Rx) checksum offload, as well as TCP segmentation offload. To ensure maximum performance from the adapter, all task offloads should be enabled, which is the default setting on the adapter. For more information, see [Performance Tuning on Linux on page 95](#).

To change offload settings for Tx and Rx, use the ethtool command:

```
ethtool --offload <ethX> [rx on|off] [tx on|off]
```

## Configuring Receive/Transmit Ring Buffer Size

By default receive and transmit ring buffers on the Solarflare adapter support 1024 descriptors. The user can identify and reconfigure ring buffer sizes using the ethtool command.

To identify the current ring size:

```
ethtool -g ethX
```

To set the new transmit or receive ring size to value N

```
ethtool -G ethX [rx N] tx N]
```

The ring buffer size must be a value between 128 and 4096. On the SFN7000, SFN8000 and X2 series adapters, and the U25 adapter, the maximum TX buffer size is restricted to 2048. Buffer size can also be set directly in the `modprobe.conf` file or add the options line to a file under the `/etc/modprobe.d` directory e.g.

```
options sfc rx_ring=4096
```

Using the `modprobe` method sets the value for all Solarflare interfaces. Then reload the driver for the option to become effective:

```
modprobe -r sfc
modprobe sfc
```

## Configuring Jumbo Frames

Solarflare adapters support frame sizes from 1500 bytes to 9216 bytes. For example, to set a new frame size (MTU) of 9000 bytes, enter the following command:

```
ifconfig <ethX> mtu 9000
```

To make the changes permanent, edit the network configuration file for <ethX>; for example, /etc/sysconfig/network-scripts/ifcfg-eth1 and append the following configuration directive, which specifies the size of the frame in bytes:

```
MTU=9000
```

## Configuring FEC

For information about FEC, see [Forward Error Correction on page 42](#).

FEC settings can be configured with ethtool from version 4.8 on RHEL7.5 (kernel 3.10.0-862.14.4.el7.x86\_64) or later RHEL versions or using generic kernel 4.14+.

- Identify FEC settings  
# ethtool --show-fec <interface>
- Set FEC  
# ethtool --set-fec <interface> [encoding auto|off|rs|baser]

The auto setting means adapter firmware will attempt to use the FEC type required by the DAC or optical cable.

## Configuring FCS forwarding

To enable FCS forwarding:

```
echo 1 > /sys/class/net/<interface>/device/forward_fcs
```

On recent drivers, you can also do this using:

```
ethtool -K <interface> rx-fcs on
```

## Using sfctool

Solarflare sfctool is a network driver utility that adopts some of the more advanced ethtool features, making these available on older Linux kernels. sfctool is available from the Solarflare Linux Utilities package. See [Linux Utilities RPM on page 76](#)

Users should ensure they have a recent Solarflare adapter driver. sfctool is not supported with the OS 'in-tree' driver (sfc driver versions 4.0 or 4.1).

sfctool uses the same command format as ethtool – replace 'ethtool' with 'sfctool'.

## 3.8 Setting Up VLANs

VLANs offer a method of dividing one physical network into multiple broadcast domains. In enterprise networks, these broadcast domains usually match with IP subnet boundaries, so that each subnet has its own VLAN. The advantages of VLANs include:

- Performance
- Ease of management
- Security
- Trunks
- You don't have to configure any hardware device, when physically moving your server to another location.

To set up VLANs, consult the following documentation:

- To configure VLANs on SUSE Linux Enterprise Server, see:  
<http://www.novell.com/support/viewContent.do?externalId=3864609>
- To configure tagged VLAN traffic only on Red Hat Enterprise Linux, see:  
<http://kbase.redhat.com/faq/docs/DOC-8062>
- To configure mixed VLAN tagged and untagged traffic on Red Hat Enterprise Linux, see:  
<http://kbase.redhat.com/faq/docs/DOC-8064>

## 3.9 Setting Up Teams

Teaming network adapters (network bonding) allows a number of physical adapters to act as one, virtual adapter. Teaming network interfaces, from the same adapter or from multiple adapters, creates a single virtual interface with a single MAC address.

The virtual adapter or virtual interface can assist in load balancing and providing failover in the event of physical adapter or port failure.

Teaming configuration support provided by the Linux bonding driver includes:

- 802.3ad Dynamic link aggregation
- Static link aggregation
- Fault Tolerant

To set up an adapter team, consult the following documentation:

- General:  
<http://www.kernel.org/doc/Documentation/networking/bonding.txt>
- RHEL 5:  
[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.4/html/Deployment\\_Guide/s2-modules-bonding.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.4/html/Deployment_Guide/s2-modules-bonding.html)
- RHEL6:  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Deployment\\_Guide/s2-networkscripts-interfaces-chan.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s2-networkscripts-interfaces-chan.html)
- SLES:  
[http://www.novell.com/documentation/sles11/book\\_sle\\_admin/data/sec\\_basicnet\\_yast.html#sec\\_basicnet\\_yast\\_netcard\\_man](http://www.novell.com/documentation/sles11/book_sle_admin/data/sec_basicnet_yast.html#sec_basicnet_yast_netcard_man)

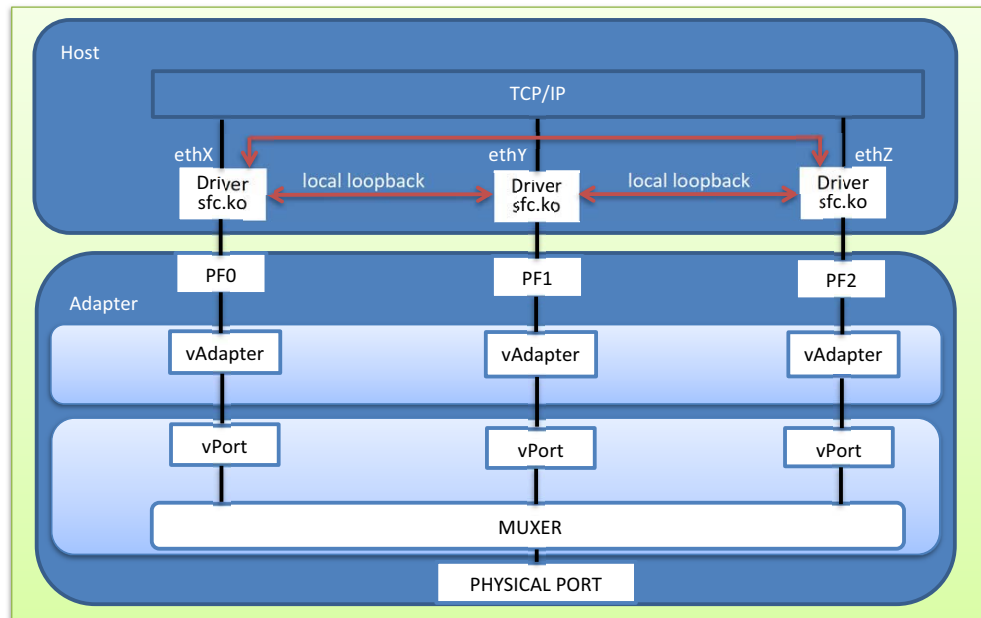
## 3.10 NIC Partitioning

NIC Partitioning is a feature supported on Solarflare adapters starting with the SFN7000 series. By partitioning the NIC, each physical network port can be exposed to the host as multiple PCIe Physical Functions (PF) with each having a unique interface name and unique MAC address.

When the Solarflare NET driver (sfc.ko) is loaded in the host, each PF is backed by a virtual adapter connected to a virtual port. A switching function supports the transport of network traffic between virtual ports (vport) and the physical port. Partitioning is particularly useful when, for example, splitting a single 40GbE interface into multiple PFs.

- Up to 16 PFs and 16 MAC addresses are support PER ADAPTER.
- On a 10GbE dual-port adapter each physical port can be exposed as a maximum 8 PFs.
- On a 40GbE dual-port adapter (in 2\*40G mode) each physical port can be exposed as a maximum 8 PFs.
- On a 40GbE dual-port adapter (in 4\*10G mode) each physical port can be exposed as a maximum 4 PFs.

## NIC Partitioning Without VLANs



**Figure 9: NIC Partitioning - without VLANs**

- Configured without VLANs, all PFs are in the same Ethernet layer 2 broadcast domain i.e. a packet broadcast from any one PF would be received by all other PFs.
- Transmitted packets go directly to the wire. Packets sent between PFs are routed through the local TCP/IP stack loopback interface without touching the sfc driver.
- Received broadcast packets are replicated to all PFs.
- Received multicast packets are delivered to each subscriber.
- Received unicast packets are delivered to the PF with a matching MAC address. Because the TCP/IP stack has multiple network interfaces on the same broadcast domain, there is always the possibility that any interface could respond to an ARP request. To avoid this the user should use `arp_ignore=2` to avoid ARP cache pollution ensuring that ARP responses are only sent from an interface if the target IP address in the ARP request matches the interface address with both sender/receiver IP addresses in the same subnet.
- To set `arp_ignore` for the current session:  

```
echo 2 >/proc/sys/net/ipv4/conf/all/arp_ignore
```
- To set `arp_ignore` permanently (does not affect the current session), add the following line to the `/etc/sysctl.conf` file:  

```
net.ipv4.conf.all.arp_ignore = 2
```

- The MUXER function is a layer2 switching function for received traffic enabled in adapter firmware. When the OS delivers traffic to local interfaces via the loopback interface, the MUXER acts as a layer2 switch for both transmit and receive.

## VLAN Support

When PFs are configured with VLAN tags each PF must be in a different VLAN. The MUXER function acts as a VLAN aggregator such that transmitted packets are sent to the wire and received packets are demultiplexed based on the VLAN tags. VLAN tags are added/stripped by the adapter firmware transparent to the OS and driver. VLAN tags can be assigned when PFs are enabled using the `sfboot` command. A single PF can be assigned VLAN tag 0 allowing it to receive untagged traffic.

```
# sfboot switch-mode=partitioning pf-count=3 pf-vlans=0,200,300
```

The first VLAN ID in the `pf-vlans` comma separated list is assigned to the first PF of the physical port and thereafter tags are assigned to PFs in lowest MAC address order.

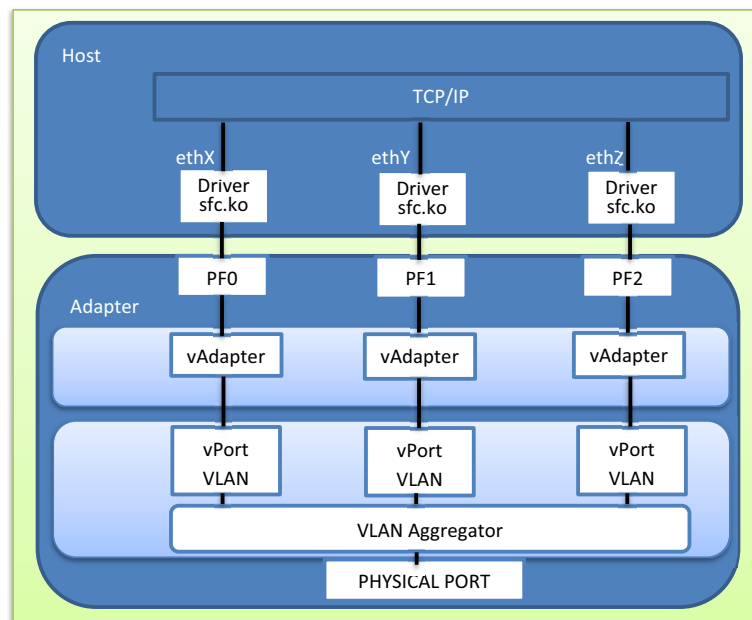


Figure 10: NIC Partitioning - VLAN Support

## NIC Partitioning Configuration

Up to 16 PFs and 16 MAC addresses are supported per adapter. The PF count value applies to all physical ports. Ports cannot be configured individually.

- 1 Ensure the Solarflare adapter driver (`sfc.ko`) is installed on the host.
- 2 The `sfboot` utility (`pf-count`) from the Solarflare Linux Utilities package (SF-107601-LS) is used to partition physical interfaces to the required number of PFs.

**3** To partition all ports (example configures 4 PFs per port):

```
# sfboot switch-mode=partitioning pf-count=4
Solarflare boot configuration utility [v4.5.0]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
```

```
eth2:
  Boot image                Option ROM only
  Link speed                Negotiated automatically
  Link-up delay time        5 seconds
  Banner delay time         2 seconds
  Boot skip delay time      5 seconds
  Boot type                 Disabled
  Physical Functions per port 4
  MSI-X interrupt limit     32
  Number of Virtual Functions 0
  VF MSI-X interrupt limit  8
  Firmware variant         full feature / virtualization
  Insecure filters          Disabled
  MAC spoofing              Disabled
  VLAN tags                 None
  Switch mode               Partitioning
```

*A cold reboot of the server is required for sfboot changes to be effective.*

**4** Following reboot each PF will be visible using the `lspci` command:

```
# lspci -d 1924:
07:00.0 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.1 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.2 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.3 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.4 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.5 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.6 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.7 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
```

- If more than 8 functions are required the server must support ARI - see [Alternative Routing-ID Interpretation \(ARI\) on page 226](#).
- Solarflare also recommend setting `pci=realloc` in the kernel configuration grub file - refer to [Kernel Configuration on page 226](#) for details.

**5** To identify which physical port a given network interface is using:

```
# cat /sys/class/net/eth<N>/device/physical_port
```

**6** If the Solarflare driver is loaded, PFs will also be visible using the `ifconfig` command where each PF is listed with a unique MAC address.

## Software Requirements

The server must have the following (minimum) net driver and firmware versions to enable NIC Partitioning:

```
# ethtool -i eth<N>  
driver: sfc  
version: 4.4.1.1017  
firmware-version: 4.4.2.1011 rx0 tx0
```

The adapter must be using the *full-feature* firmware variant which can be selected using the *sfboot* utility and confirmed with *rx0 tx0* appearing after the version number in the output from *ethtool* as shown above.

The firmware update utility (*sfupdate*) and boot ROM configuration tool (*sfboot*) are available in the Solarflare Linux Utilities package (SF-107601-LS issue 28 or later).



### 3.11 NIC Partitioning with SR-IOV

When combining NIC partitioning with SR-IOV, every partition (PF) must be in a separate VLAN. The user is able to create a number of PFs per physical port and associate a number of VFs with each PF. Within this layer2 broadcast domain there is switching between a PF and its associated VFs.

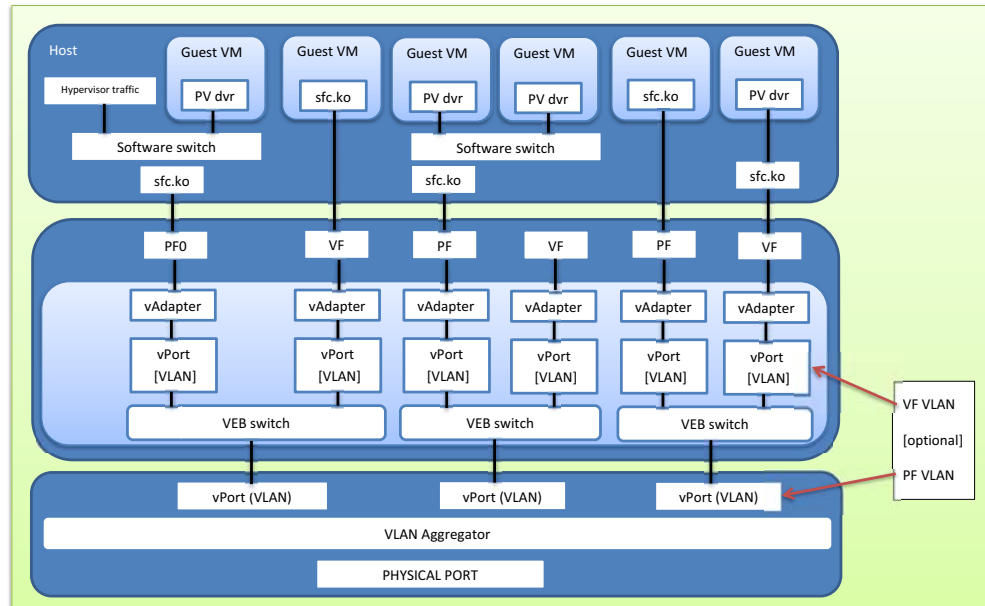


Figure 11: NIC Partitioning with SR-IOV

#### Configuration

- 1 Use the `sfboot` utility to set the firmware switch-mode, create PFs, assign unique VLAN ID to each PF and assign a number of VFs for each PF. In the following example 4 PFs are configured per physical port and 2 VFs per PF:

```
# sfboot switch-mode=partitioning-with-sriov pf-count=4 /
pf-vlans=0,100,110,120 vf-count=2
```

eth10:

Interface-specific boot options are not available. Adapter-wide options are available via eth4 (00-0F-53-21-00-60).

eth11:

Interface-specific boot options are not available. Adapter-wide options are available via eth4 (00-0F-53-21-00-60).

eth12:

Interface-specific boot options are not available. Adapter-wide options are available via eth4 (00-0F-53-21-00-60).

eth13:

Interface-specific boot options are not available. Adapter-wide

options are available via eth4 (00-0F-53-21-00-60).

eth14:

Interface-specific boot options are not available. Adapter-wide options are available via eth4 (00-0F-53-21-00-60).

eth15:

Interface-specific boot options are not available. Adapter-wide options are available via eth4 (00-0F-53-21-00-60).

eth4:

Boot image	Option ROM only
Link speed	Negotiated automatically
Link-up delay time	5 seconds
Banner delay time	2 seconds
Boot skip delay time	5 seconds
Boot type	Disabled
Physical Functions per port	4
MSI-X interrupt limit	32
Number of Virtual Functions	2
VF MSI-X interrupt limit	8
Firmware variant	full feature / virtualization
Insecure filters	Disabled
MAC spoofing	Disabled
VLAN tags	0,100,110,120
Switch mode	Partitioning with SRIOV

eth5:

Boot image	Option ROM only
Link speed	Negotiated automatically
Link-up delay time	5 seconds
Banner delay time	2 seconds
Boot skip delay time	5 seconds
Boot type	Disabled
Physical Functions per port	4
MSI-X interrupt limit	32
Number of Virtual Functions	2
VF MSI-X interrupt limit	8
Firmware variant	full feature / virtualization
Insecure filters	Disabled
MAC spoofing	Disabled
VLAN tags	0,100,110,120
Switch mode	Partitioning with SRIOV

**2** PF interfaces are visible in the host using the `ifconfig` command:

```
eth4      Link encap:Ethernet HWaddr 00:0F:53:21:00:60
eth5      Link encap:Ethernet HWaddr 00:0F:53:21:00:61
eth10     Link encap:Ethernet HWaddr 00:0F:53:21:00:64
eth11     Link encap:Ethernet HWaddr 00:0F:53:21:00:65
eth12     Link encap:Ethernet HWaddr 00:0F:53:21:00:66
eth13     Link encap:Ethernet HWaddr 00:0F:53:21:00:63
eth14     Link encap:Ethernet HWaddr 00:0F:53:21:00:62
eth15     Link encap:Ethernet HWaddr 00:0F:53:21:00:67
```

- The output from steps 1 and 2 above identifies a server with 2 physical interfaces (eth4/eth5), 4 PFs per physical port and identifies the following PF-VLAN configuration:

**Table 19: PF-VLAN Configuration**

Interface	MAC Address	PF	VLAN ID
eth4	00:0F:53:21:00:60	PF0	0
eth10	00:0F:53:21:00:64	PF4	110
eth12	00:0F:53:21:00:66	PF6	120
eth14	00:0F:53:21:00:62	PF2	100
eth5	00:0F:53:21:00:61	PF1	0
eth11	00:0F:53:21:00:65	PF5	110
eth13	00:0F:53:21:00:63	PF3	100
eth15	00:0F:53:21:00:67	PF7	120

- Refer to [SR-IOV Configuration on page 230](#) for procedures to create VMs and VFs.

## VLAN Configuration

When using partitioning with SR-IOV, all PFs must have a unique VLAN tag. A single PF from each physical port can use tag 0 (zero) to receive untagged traffic. VLAN tags are transparently inserted/stripped by the adapter firmware.

## LACP Bonding

LACP Bonding is not currently supported using the NIC Partitioning configuration mode as the LACP partner i.e. the switch will be unaware of the configured partitions.

Users are advised to refer to the sfc driver release notes for current limitations when using the NIC partitioning features.

## 3.12 Receive Side Scaling (RSS)

Solarflare adapters support Receive Side Scaling (RSS). RSS enables packet receive-processing to scale with the number of available CPU cores. RSS requires a platform that supports MSI-X interrupts. RSS is enabled by default.

When RSS is enabled the controller uses multiple receive queues to deliver incoming packets. The receive queue selected for an incoming packet is chosen to ensure that packets within a TCP stream are all sent to the same receive queue – this ensures that packet-ordering within each stream is maintained. Each receive queue has its own dedicated MSI-X interrupt which ideally should be tied to a dedicated CPU core. This allows the receive side TCP processing to be distributed amongst the available CPU cores, providing a considerable performance advantage over a conventional adapter architecture in which all received packets for a given interface are processed by just one CPU core. RSS can be restricted to only process receive queues on the NUMA node local to the Solarflare adapter. To configure this the driver module option `rss_numa_local` should be set to 1.

By default the driver enables RSS and configures one RSS Receive queue per CPU core. The number of RSS Receive queues can be controlled via the driver module parameter `rss_cpus`. The following table identifies `rss_cpus` options.

**Table 20: `rss_cpus` Options**

Option	Description	Interrupt Affinity (MSI-X)
<code>&lt;num_cpus&gt;</code>	Indicates the number of RSS queues to create.	A separate MSI-X interrupt for a receive queue is affinitized to each CPU.
<code>packages</code>	An RSS queue will be created for each multi-core CPU package. The first CPU in the package will be chosen.	A separate MSI-X interrupt for a receive queue, is affinitized to each of the designated package CPUs.
<code>cores</code>	An RSS queue will be created for each CPU. The first hyperthread instance (If CPU has hyperthreading) will be chosen.  The default option.	A separate MSI-X interrupt for a receive queue, is affinitized to each of the CPUs.
<code>hyperthreads</code>	An RSS queue will be created for each CPU hyperthread (hyperthreading must be enabled).	A separate MSI-X interrupt for a receive queue, is affinitized to each of the hyperthreads.

Add the following line to `/etc/modprobe.conf` file or add the options line to a user created file under the `/etc/modprobe.d` directory. The file should have a `.conf` extension:

```
options sfc rss_cpus=<option>
```

To set `rss_cpus` equal to the number of CPU cores:

```
options sfc rss_cpus=cores
```

Sometimes, it can be desirable to disable RSS when running single stream applications, since all interface processing may benefit from taking place on a single CPU:

```
options sfc rss_cpus=1
```

The driver must be reloaded to enable option changes:



**NOTE:** The association of RSS receive queues to a CPU is governed by the receive queue's MSI-X interrupt affinity. See [Interrupt Affinity on page 105](#) for more details.

```
rmmod sfc  
modprobe sfc
```



**NOTE:** RSS also works for UDP packets. For UDP traffic the Solarflare adapter will select the Receive CPU based on IP source and destination addresses. Solarflare adapters support IPv4 and IPv6 RSS.

## 3.13 Receive Flow Steering (RFS)

RFS will attempt to steer packets to the core where a receiving application is running. This reduces the need to move data between processor caches and can significantly reduce latency and jitter. Modern NUMA systems, in particular, can benefit substantially from RFS where packets are delivered into memory local to the receiving thread.

Unlike RSS which selects a CPU from a CPU affinity mask set by an administrator or user, RFS will store the application's CPU core identifier when the application process calls `recvmsg()` or `sendmsg()`.

- A hash is calculated from a packet's addresses or ports (2-tuple or 4-tuple) and serves as the consistent hash for the flow associated with the packet.
- Each receive queue has an associated list of CPUs to which RFS may enqueue the received packets for processing.
- For each received packet, an index into the CPU list is computed from the flow hash modulo the size of the CPU list.

There are two types of RFS implementation; Soft RFS and Hardware (or Accelerated) RFS.

Soft RFS is a software feature supported since Linux 2.6.35 that attempts to schedule protocol processing of incoming packets on the same processor as the user thread that will consume the packets.

Accelerated RFS requires Linux kernel version 2.6.39 or later, with the Linux `sfc` driver or Solarflare v3.2 network adapter driver.

RFS can dynamically change the allowed CPUs that can be assigned to a packet or packet stream and this introduces the possibility of out of order packets. To prevent out of order data, two tables are created that hold state information used in the CPU selection.

- **Global\_flow\_table:** Identifies the number of simultaneous flows that are managed by RFS.
- **Per\_queue\_table:** Identifies the number of flows that can be steered to a queue. This holds state as to when a packet was last received.

The tables support the steering of incoming packets from the network adapter to a receive queue affinitized to a CPU where the application is waiting to receive them. The Solarflare accelerated RFS implementation requires configuration through the two tables and the `ethtool -K` command.

The following sub-sections identify the RFS configuration procedures:

## Kernel Configuration

Before using RFS the kernel must be compiled with the kconfig symbol CONFIG\_RPS enabled. Accelerated RFS is only available if the kernel is compiled with the kconfig symbol CONFIG\_RFS\_ACCEL enabled.

## Global Flow Count

Configure the number of simultaneous flows that will be managed by RFS. The suggested flow count will depend on the expected number of active connections at any given time and may be less than the number of open connections. The value is rounded up to the nearest power of two.

```
# echo 32768 > /proc/sys/net/core/rps_sock_flow_entries
```

## Per Queue Flow Count

For each adapter interface there will exist a 'queue' directory containing one 'rx' or 'tx' subdirectory for each queue associated with the interface. For RFS only the receive queues are relevant.

```
# cd /sys/class/net/eth3/queue
```

Within each 'rx' subdirectory, the rps\_flow\_cnt file holds the number of entries in the per-queue flow table. If only a single queue is used then rps\_flow\_cnt will be the same as rps\_sock\_flow\_entries. When multiple queues are configured the count will be equal to rps\_sock\_flow\_entries/*N* where *N* is the number of queues, for example:

rps\_sock\_flow\_entries = 32768 and there are 16 queues then rps\_flow\_cnt for each queue will be configured as 2048.

```
# echo 2048 > /sys/class/net/eth3/queues/rx-0/rps_flow_cnt
# echo 2048 > /sys/class/net/eth3/queues/rx-1/rps_flow_cnt
```

## Disable RFS

To turn off RFS using the following command:

```
# ethtool -K <devname> ntuple off
```

## 3.14 Transmit Packet Steering (XPS)

Transmit Packet Steering (XPS) is supported in Linux 2.6.38 and later. XPS is a mechanism for selecting which transmit queue to use when transmitting a packet on a multi-queue device.

XPS is configured on a per transmit queue basis where a bitmap of CPUs identifies the CPUs that may use the queue to transmit.

### Kernel Configuration

Before using XPS the kernel must be compiled with the kconfig symbol CONFIG\_XPS enabled.

### Configure CPU/Hyperthreads

Within in each `/sys/class/net/<interface>/queues/tx-N` directory there exists an `xps_cpus` file which contains a bitmap of CPUs that can use the queue to transmit. In the following example transmit queue 0 can be used by the first two CPUs and transmit queue 1 can be used by the following two CPUs:

```
# echo 3 > /sys/class/net/eth3/queues/tx-0/xps_cpus
# echo c > /sys/class/net/eth3/queues/tx-1/xps_cpus
```

If hyperthreading is enabled, each hyperthread is identified as a separate CPU, for example if the system has 16 cores but 32 hyperthreads then the transmit queues should be paired with the hyperthreaded cores:

```
# echo 30003 > /sys/class/net/eth3/queues/tx-0/xps_cpus
# echo c000c > /sys/class/net/eth3/queues/tx-1/xps_cpus
```

### XPS - Example Configuration

#### System Configuration:

- Single Solarflare adapter
- 2 x 8 core processors with hyperthreading enabled to give a total of 32 cores
- `rss_cpus=8`
- Only 1 interface on the adapter is configured
- The IRQ Balance service is disabled



### Identify interrupts for the configured interface:

```
# cat /proc/interrupts | grep 'eth3\ | CPU'
```

```
> cat /proc/irq/132/smp_affinity
00000000,00000000,00000000,00000001
> cat /proc/irq/133/smp_affinity
00000000,00000000,00000000,00000100
> cat /proc/irq/134/smp_affinity
00000000,00000000,00000000,00000002
[...snip...]
> cat /proc/irq/139/smp_affinity
00000000,00000000,00000000,00000800
```

The output identifies that IRQ-132 is the first queue and is routed to CPU0. IRQ-133 is the second queue routed to CPU8, IRQ-134 to CPU2 and so on.

### Map TX queue to CPU

Hyperthreaded cores are included with the associated physical core:

```
> echo 110011 > /sys/class/net/eth3/queues/tx-0/xps_cpus
> echo 11001100 > /sys/class/net/eth3/queues/tx-1/xps_cpus
> echo 220022 > /sys/class/net/eth3/queues/tx-2/xps_cpus
> echo 22002200 > /sys/class/net/eth3/queues/tx-3/xps_cpus
> echo 440044 > /sys/class/net/eth3/queues/tx-4/xps_cpus
> echo 44004400 > /sys/class/net/eth3/queues/tx-5/xps_cpus
> echo 880088 > /sys/class/net/eth3/queues/tx-6/xps_cpus
> echo 88008800 > /sys/class/net/eth3/queues/tx-7/xps_cpus
```

### Configure Global and Per Queue Tables

- The flow count (number of active connections at any one time) = 32768
- Number of queues = 8 (rss\_cpus)
- So the flow count for each queue will be 32768/8

```
> echo 32768 > /proc/sys/net/core/rps_sock_flow_entries
> echo 4096 > /sys/class/net/eth3/queues/rx-0/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-1/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-2/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-3/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-4/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-5/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-6/rps_flow_cnt
> echo 4096 > /sys/class/net/eth3/queues/rx-7/rps_flow_cnt
```

## 3.15 Linux Utilities RPM

The Solarflare Linux Utilities RPM contains:

- A boot ROM utility.  
See [Configuring the Boot Manager with sboot on page 78](#).
- A flash firmware update utility.  
See [Upgrading Adapter Firmware with sfupdate on page 86](#).
- A firmware feature activation key install utility.  
See [Installing an activation key with sfkey on page 92](#).

The RPM package, is supplied as 64bit and 32bit binaries compiled to be compatible with GLIBC versions for all supported distributions. The RPM is signed from version 6.2.1.1000 onwards. The Solarflare utilities RPM file can be downloaded from the following location:

<https://support.solarflare.com/>

- SF-104451-LS is a 32bit binary RPM package.
- SF-107601-LS is a 64bit binary RPM package.

### Uninstall a previous Utilities RPM

The rpm install command will warn if another version of the Utilities RPM is already installed on the system.

- To identify this RPM:  

```
# rpm -qa | grep sfutils
sfutils-<version>.x86_64
```

 or if you are using an Ubuntu/Debian server:  

```
# dpkg -l | grep sfutils
sfutils-<version>.deb
```
- To remove the sfutils RPM:  

```
# rpm -e sfutils-<version>.x86_64
```

 or if you are using an Ubuntu/Debian server:  

```
# dpkg -r sfutils-<version>.deb
```

## Install Utilities

To install the Linux Utilities package:

**1** Download and copy the zipped binary RPM package to the required directory.

**2** Unzip the package:

```
# unzip SF-107601-LS-<version>_Solarflare_Linux_Utilities_RPM_64bit.zip
```

**3** Install the binary RPM:

```
# rpm -Uvh sfutils-<version>.x86_64.rpm
Preparing...      ##### [100%]
1:sfutils         ##### [100%]
```

or if you are using an Ubuntu/Debian server:

**a)** Create the .deb file:

```
sudo alien sfutils-<version>.x86_64.rpm
```

This command generates the sfutils\_<version>\_amd64.deb file.

**b)** Install the deb file:

```
sudo dpkg -i -sfutils_<version>_amd64.deb
```

**4** Check that the RPM installed correctly:

```
# rpm -q sfutils
sfutils-<version>.x86_64
```

or if you are using an Ubuntu/Debian server:

```
# dpkg -i sfutils
```

## Using the Solarflare Utilities bootable Linux disk image

If you are unable to boot the system, you can instead use the Solarflare Utilities bootable Linux disk image. This is available from:

[https://support.solarflare.com/index.php?id=1960&option=com\\_cognidox](https://support.solarflare.com/index.php?id=1960&option=com_cognidox)

This package is supplied as a ISO 9660 disk image. It contains the same utilities as the Solarflare Linux Utilities RPM, as well as a 4.0 driver for Solarflare adapters.



**CAUTION:** The driver contained in this package has only undergone QA for Utilities use, and should therefore not be used outside of this package or to carry production network traffic.

## 3.16 Configuring the Boot Manager with sfboot

- [Sfboot: Command Usage on page 78.](#)
- [Sfboot: Command Line Options on page 79.](#)
- [Sfboot: Examples on page 84.](#)

Sfboot is a command line utility for configuring Solarflare adapter Boot Manager options, including PXE and UEFI booting. Using sfboot is an alternative to using **Ctrl + B** to access the Boot ROM agent during server startup.

See [Solarflare Boot Manager on page 257](#) for more information on the Boot Rom agent.

PXE and UEFI network boot is not supported for Solarflare adapters on IBM System p servers.

### **Sfboot: SLES 11 Limitation**

Due to limitations in SLES 11 using kernel versions prior to 2.6.27.54 it is necessary to reboot the server after running the sfboot utility.

### **Sfboot: Command Usage**

The general usage for sfboot is as follows (as root):

```
sfboot [--adapter=eth<N>] [options] [parameters]
```

When the --adapter option is not specified, the sfboot command applies to all adapters present in the target host.

The format for the parameters are:

```
<parameter>=<value>
```

## Sfboot: Command Line Options

Table 21 lists the options for `sfboot`, Table 22 lists the available global parameters, and Table 23 lists the available per-adapter parameters. Note that command line options are case insensitive and may be abbreviated.



**NOTE:** Abbreviations in scripts should be avoided, since future updates to the application may render abbreviated scripts invalid.

**Table 21: Sfboot Options**

Option	Description
<code>-h, --help</code>	Displays command line syntax and provides a description of each <code>sfboot</code> option.
<code>-V, --version</code>	Shows detailed version information and exits.
<code>-v, --verbose</code>	Shows extended output information for the command entered.
<code>-y, --yes</code>	Update without prompting.
<code>-s, --quiet</code> Aliases: <code>--silent</code>	Suppresses all output, except errors; no user interaction. The user should query the completion code to determine the outcome of commands when operating silently.
<code>-l, --list</code>	Lists all available Solarflare adapters. This option shows the ifname and MAC address.  Note: this option may not be used in conjunction with any other option. If this option is used with configuration parameters, those parameters will be silently ignored.
<code>-i, --adapter =&lt;ethX&gt;</code>	Performs the action on the identified Solarflare network adapter. The adapter identifier <code>ethX</code> can be the ifname or MAC address, as output by the <code>--list</code> option. If <code>--adapter</code> is not included, the action will apply to all installed Solarflare adapters.
<code>-c, --clear</code>	Resets all adapter configuration options to their default values. If an adapter is specified, options for the given adapter are reset, but global options (shown in Table 22) are not reset. Note that <code>--clear</code> can also be used with parameters, allowing you to reset to default values, and then apply the parameters specified.
<code>-r, --repair</code>	Restore firmware configuration settings to default values. The <code>sfboot</code> option should only be used if a firmware upgrade/downgrade using <code>sfboot</code> has failed.

The following global parameters are used to control the configurable parameters for the Boot ROM driver when running prior to the operating system booting.

**Table 22: Sfboot Global Parameters**

Parameter	Description
boot-image= all optionrom uefi disabled	Specifies which boot firmware images are served-up to the BIOS during start-up. This parameter can not be used if the --adapter option has been specified. This is a global option and applies to all ports on the NIC.
port-mode=refer to <a href="#">Port Modes on page 43</a>	<p>Configure the port mode to use. This is for SFN7000, SFN8000 and X2 series adapters and the U25 adapter only. The values specify the connectors available after using any splitter cables. The usable values are adapter-dependent.</p> <p>For details of port-modes refer to <a href="#">Port Modes on page 43</a></p> <p>Changes to this setting with sfboot require a cold reboot to become effective. MAC address assignments may change after altering this setting.</p>
firmware-variant= full-feature ultra-low-latency  capture-packed-stream auto	<p>Configure the firmware variant to use. This is for SFN7000, SFN8000 and X2 series adapters and the U25 adapter only:</p> <ul style="list-style-type: none"> <li>the SFN7002F adapter is factory set to full-feature</li> <li>all other adapters are factory set to auto.</li> </ul> <p>Default value = auto - means the driver will select a variant that meets its needs:</p> <ul style="list-style-type: none"> <li>the VMware driver always uses full-feature</li> <li>otherwise, ultra-low-latency is used.</li> </ul> <p>The ultra-low-latency variant produces best latency without support for TX VLAN insertion or RX VLAN stripping (not currently used features). It is recommended that Onload customers use the ultra-low-latency variant. This is a global option and applies to all ports on the NIC.</p>
insecure-filters= enabled disabled	If enabled bypass filter security on non-privileged functions. This is for SFN7000 and SFN8000 series adapters only. This reduces security in virtualized environments. The default is disabled. When enabled a function (PF or VF) can insert filters not qualified by their own permanent MAC address. This is a requirement and should be enabled when using Onload or when using bonded interfaces. This is a global option and applies to all ports on the NIC.

**Table 22: Sfboot Global Parameters (continued)**

Parameter	Description
mac-spoofing=default enabled disabled	<p>If enabled, non-privileged functions can create unicast filters for MAC addresses that are not associated with them. This is for SFN7000, SFN8000 and X2 series adapters and the U25 adapter only.</p> <p>The default is disabled.</p> <p>Changes to this setting with sfboot require a cold reboot to become effective. This is a global option and applies to all ports on the NIC.</p>
rx-dc-size=8 16 32 64	<p>Specifies the size of the descriptor cache for each receive queue. This is for SFN7000, SFN8000 and X2 series adapters and the U25 adapter only. The default is:</p> <ul style="list-style-type: none"> <li>• 32 if the port-mode supports the maximum number of connectors for the adapter</li> <li>• 64 if the port-mode supports a reduced number of connectors.</li> </ul>
change-mac= default enabled disabled	<p>This is for SFN7000, SFN8000 and X2 series adapters and the U25 adapter only. Change the unicast MAC address for a non-privileged function on this port. This is a global option and applies to all physical ports on the NIC.</p>
tx-dc-size=8 16 32 64	<p>Specifies the size of the descriptor cache for each transmit queue. This is for SFN7000, SFN8000 and X2 series adapters and the U25 adapter only. The default is:</p> <ul style="list-style-type: none"> <li>• 16 if the port-mode supports the maximum number of connectors for the adapter</li> <li>• 32 if the port-mode supports a reduced number of connectors.</li> </ul>
vi-count=<vi count>	<p>Sets the total number of virtual interfaces that will be available on the NIC.</p>
event-merge-timeout=<timeout in nanoseconds>	<p>Specifies the timeout in nanoseconds for RX event merging. A timeout of 0 means that event merging is disabled.</p>
permit-fw-downgrade=yes no	<p>For X2522-10G, X2522-25G, X2541 and X2542 adapters only, setting this option to yes allows firmware to be downgraded to pre-2020 releases. It has no effect on other adapters.</p> <p>The default value is no.</p> <p>This is a global option and applies to all ports on the NIC.</p>

The following per-adapter parameters are used to control the configurable parameters for the Boot ROM driver when running prior to the operating system booting.

**Table 23: Sfboot Per-adapter Parameters**

Parameter	Description
<code>link-speed=auto 10g 1g 100m</code>	<p>Specifies the network link speed of the adapter used by the Boot ROM. The default is auto. On the 10GBASE-T adapters, auto instructs the adapter to negotiate the highest speed supported in common with its link partner. On SFP+ adapters, auto instructs the adapter to use the highest link speed supported by the inserted SFP+ module. On 10GBASE-T and SFP+ adapters, any other value specified will fix the link at that speed, regardless of the capabilities of the link partner, which may result in an inability to establish the link.</p> <p>auto Auto-negotiate link speed (default)</p> <p>10G 10G bit/sec</p> <p>1G 1G bit/sec</p> <p>100M 100M bit/sec</p>
<code>linkup-delay= &lt;delay time in seconds&gt;</code>	<p>Specifies the delay (in seconds) the adapter defers its first connection attempt after booting, allowing time for the network to come up following a power failure or other restart. This can be used to wait for spanning tree protocol on a connected switch to unblock the switch port after the physical network link is established. The default is 5 seconds.</p>
<code>banner-delay= &lt;delay time in seconds&gt;</code>	<p>Specifies the wait period for Ctrl-B to be pressed to enter adapter configuration tool.</p> <p>&lt;delay time in seconds&gt; = 0-256</p>
<code>bootskip-delay= &lt;delay time in seconds&gt;</code>	<p>Specifies the time allowed for Esc to be pressed to skip adapter booting.</p> <p>&lt;delay time in seconds&gt; = 0-256</p>
<code>boot-type=pxe disabled</code>	<p>Sets the adapter boot type – effective on next boot.</p> <p>pxe – PXE (Preboot eXecution Environment) booting</p> <p>disabled – Disable adapter booting</p>



**Table 23: Sfboot Per-adapter Parameters (continued)**

Parameter	Description
pf-count=<pf count>	<p>This is the number of available PCIe PFs per physical network port. This setting is applied to all ports on the adapter.</p> <p>Changes to this setting with sfboot require a cold reboot to become effective. MAC address assignments may change after altering this setting.</p>
msix-limit= 8 16 32 64 128 256 512 1024	<p>Specifies the maximum number of MSI-X interrupts that each PF will use. The default is 32.</p> <p>Note: Using the incorrect setting can impact the performance of the adapter. Contact Solarflare technical support before changing this setting.</p>
vf-count=<vf count>	<p>The number of virtual functions (VF) advertised to the operating system for each Physical Function on this physical network port.</p> <p>Adapters support 2048 interrupts</p> <p>Adapters support a total limit of 127 virtual functions per port.</p> <p>Depending on the values of msix-limit and vf-msix-limit, some of these virtual functions may not be configured.</p> <p>Enabling all 127 VFs per port with more than one MSI-X interrupt per VF may not be supported by the host BIOS - in which case you may get 127 VFs on one port and none on others. Contact your BIOS vendor or reduce the VF count.</p> <p>The sriov parameter is implied if vf-count is greater than zero.</p> <p>Changes to this setting with sfboot require a cold reboot to become effective.</p>
vf-msix-limit= 1 2 4 8 16 32 64 128 256	<p>The maximum number of interrupts a virtual function may use.</p>

**Table 23: Sfboot Per-adapter Parameters (continued)**

Parameter	Description
<code>pf-vlans=&lt;tag&gt;[,&lt;tag&gt;[,...]] none</code>	<p>Comma separated list of VLAN tags for each PF in the range 0-4094 - see <code>sfboot --help</code> for details.</p> <p>Setting <code>pf-vlans=none</code> will clear all VLAN tags on the port. <code>pf-vlans</code> should be included after the <code>pf-count</code> option on the <code>sfboot</code> command line.</p> <p>If the number of PFs is changed then the VLAN tags will be cleared.</p>
<code>switch-mode=</code> <code>default sriov partitioning </code> <code>partitioning-with-sriov pfiov</code>	<p>Specifies the mode of operation that the port will be used in:</p> <p><code>default</code> - single PF created, zero VFs created.</p> <p><code>sriov</code> - SR-IOV enabled, single PF created, VFs configured with <code>vf-count</code>.</p> <p><code>partitioning</code> - PFs configured with <code>pf-count</code>, VFs configured with <code>vf-count</code>. See <a href="#">NIC Partitioning on page 62</a> for details.</p> <p><code>partitioning-with-sriov</code> - SR-IOV enabled, PFs configured with <code>pf-count</code>, VFs configured with <code>vf-count</code>. See <a href="#">NIC Partitioning on page 62</a> for details.</p> <p><code>pfiov</code> - PFIOV enabled, PFs configured with <code>pf-count</code>, VFs not supported.</p> <p>Changes to this setting with <code>sfboot</code> require a cold reboot to become effective.</p>

## Sfboot: Examples

- Show the current boot configuration for all adapters:

```
sfboot
# ./sfboot
Solarflare boot configuration utility [v4.3.1]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005

eth4:
  Boot image                Option ROM only
  Link speed                Negotiated automatically
  Link-up delay time        5 seconds
  Banner delay time         2 seconds
  Boot skip delay time      5 seconds
  Boot type                  Disabled
  Physical Functions per port 1
  MSI-X interrupt limit     32
  Number of Virtual Functions 0
  VF MSI-X interrupt limit  8
```

Firmware variant	full feature / virtualization
Insecure filters	Disabled
VLAN tags	None
Switch mode	Default

- List all Solarflare adapters installed on the localhost:

```
sfboot --list
```

```
./sfboot -l
Solarflare boot configuration utility [v4.3.1]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
Adapter list:
eth4
eth5
```

- Enable Firmware Variant

```
sfboot firmware-variant=full-feature
```

```
Solarflare boot configuration utility [v4.3.1]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
eth4:
Boot image                Option ROM only
Link speed                 Negotiated automatically
Link-up delay time        7 seconds
Banner delay time         3 seconds
Boot skip delay time      6 seconds
Boot type                  PXE
MSI-X interrupt limit     32
Number of Virtual Functions 0
VF MSI-X interrupt limit  1
Firmware variant          full feature / virtualization
```

- SR-IOV enabled and using Virtual Functions

```
sfboot switch-mode=sriov vf-count=4
```

```
Solarflare boot configuration utility [v4.3.1]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
eth4:
Boot image                Option ROM only
Link speed                 Negotiated automatically
Link-up delay time        5 seconds
Banner delay time         2 seconds
Boot skip delay time      5 seconds
Boot type                  Disabled
Physical Functions per port 1
MSI-X interrupt limit     32
Number of Virtual Functions 4
VF MSI-X interrupt limit  8
Firmware variant          full feature / virtualization
Insecure filters          Disabled
VLAN tags                  None
Switch mode                SRIOV
```

## 3.17 Upgrading Adapter Firmware with sfupdate

- [Sfupdate: Command Usage on page 86.](#)
- [Sfupdate: Command Line Options on page 90.](#)
- [Sfupdate: Examples on page 91.](#)

Sfupdate is a command line utility to manage and upgrade the Solarflare adapter Boot ROM, Phy and adapter firmware. Embedded within the sfupdate executable are firmware images for the Solarflare adapter - the exact updates available via sfupdate depend on the specific adapter type.

See [Solarflare Boot Manager on page 257](#) for more information on the Boot Rom agent.



**CAUTION:** All Applications accelerated with OpenOnload should be terminated before updating the firmware with sfupdate.



**CAUTION:** Solarflare PTP (sfptpd) should be terminated before updating firmware.

### Sfupdate: Command Usage

The general usage for sfupdate is as follows (as root):

```
# sfupdate [--adapter=eth<N>] [options]
```

where:

- ethN is the interface name (ifname) of the Solarflare adapter to be upgraded.
- option is one of the command options listed in [Table 24](#).

The format for the options are:

```
<option>=<parameter>
```

Running the command sfupdate with no additional parameters will show the current firmware version for all Solarflare adapters and identifies whether the firmware version within sfupdate is more up to date. To update the firmware for all Solarflare adapters run the command sfupdate --write

Solarflare recommend the following procedure:

- 1 Run sfupdate to check that the firmware on all adapters is up to date.
- 2 Run sfupdate --write to update the firmware on all adapters.

```
# sfupdate --adapter=<interface> --write [--backup|--force]
```

## Sfupdate: Bundles

From 2020 onwards, `sfupdate` supports bundles.

A bundle is a single binary, that is specific to a particular Solarflare adapter. It is a signed collection of all the firmware and images used for an update. It enables adapters to be updated remotely, where the host system provides a compatible manageability interface over which to make the update.

All X2 series adapters can use bundles:

- Solarflare X2522-10G, X2522-25G, X2541 and X2542 adapters with pre-2020 firmware do not use bundles. When they are updated to use 2020 or later firmware, they are also migrated to use bundles. All future updates must use bundles (but see [Downgrades](#) below).
- X2552, X2562, and OEM X2-series adapters use only bundles (and have always done so).

SFN8000-series and earlier adapters do not use bundles, and are not affected by this change.

### Using the in-tree driver

The in-tree driver might not support the features required for bundle updates. This can result in updates that either do not start, or that timeout.

Do not use an in-tree driver to update an X2-series adapter that is using bundles.

See [About the In-tree Driver on page 54](#).

### Downgrades

It might be necessary to downgrade an X2522-10G, X2522-25G, X2541 or X2542 adapter to pre-2020 firmware, and hence for it to stop using bundles. To do so:

- 1 Use the `sfboot` command with the `permit-fw-downgrade` parameter to modify the adapter, so that it will accept images that are not part of a bundle:  
`sfboot permit-fw-downgrade=yes`
- 2 Restart the management controller and reload the drivers, for example by restarting the server.
- 3 Use an old version of `sfupdate` to forcibly write the downgraded firmware:  
`sfupdate --write --force --yes`

## Sfupdate: Linux MTD Limitations

The driver supplied “inbox” within RedHat and Novell distributions has a limitation on the number of adapters that sfupdate can support. This limitation is removed from RHEL 6.5 onwards. The Solarflare supplied driver is no longer subject to this limitation on any distro/kernel.

Linux kernel versions prior to 2.6.20 support up to 16 MTD (flash) devices. Solarflare adapters are equipped with 6 flash partitions. If more than two adapters are deployed within a system a number of flash partitions will be inaccessible during upgrade.

The limit was raised to 32 in Linux kernel version 2.6.20 and removed altogether in 2.6.35.

If issues are encountered during sfupdate, the user should consider one of the following options when upgrading firmware on systems equipped with more than two Solarflare adapters:

- Upgrade two adapters at a time with the other adapters removed.
- Upgrade the kernel.
- Rebuild the kernel, raising the value of MAX\_MTD\_DEVICES in include/linux/mtd/mtd.h.
- Download an *Sfutils bootable image* from:  
[https://support.solarflare.com/index.php?id=1960&option=com\\_cognidox](https://support.solarflare.com/index.php?id=1960&option=com_cognidox)

## Overcome Linux MTD Limitations

An alternative method is available to upgrade the firmware without removing the adapters.

**1** Unbind all interfaces from the drivers:

```
# for bdf in $(lspci -D -d 1924: | awk '{ print $1 }'); do \
  echo -n ${bdf}\ > /sys/bus/pci/devices/${bdf}/driver/unbind; done
```

**2** Identify the bus/device/function for all Solarflare interfaces.

Using ifconfig -a will not discover any Solarflare interfaces. Use lspci:

```
# lspci -D -d 1924:
```

Output similar to the following will be produced (5 NICs installed in this example):

```
# lspci -D -d 1924:
0000:02:00.0 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:02:00.1 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:03:00.0 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:03:00.1 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:04:00.0 Ethernet controller: Solarflare Communications SFL9021 [Solarstorm]
0000:04:00.1 Ethernet controller: Solarflare Communications SFL9021 [Solarstorm]
0000:83:00.0 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:83:00.1 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:84:00.0 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
0000:84:00.1 Ethernet controller: Solarflare Communications SFC9020 [Solarstorm]
```

- 3 There are enough resources to upgrade two NICs at a time, so re-bind interfaces in groups of four (2x2NICs):

```
# echo -n "0000:02:00.0" > /sys/bus/pci/drivers/sfc/bind
# echo -n "0000:02:00.1" > /sys/bus/pci/drivers/sfc/bind
# echo -n "0000:03:00.0" > /sys/bus/pci/drivers/sfc/bind
# echo -n "0000:03:00.1" > /sys/bus/pci/drivers/sfc/bind
```

- 4 Run sfupdate to update these NICs (command options may vary):

```
# sfupdate --write --yes --force
```

- 5 Run the command to unbind the interfaces again. There will be failures reported because some of the interfaces are not bound:

```
# for bdf in $(lspci -D -d 1924: | awk '{ print $1 }'); do \
  echo -n "${bdf}" > /sys/bus/pci/devices/${bdf}/driver/unbind; done
```

- 6 Repeat the process for the other interfaces (0000:04:00.x; 0000:83:00.x and 0000:84:00.x) doing so in pairs until all the NICs have been upgraded.

- 7 Rebind all interfaces, doing so en-mass and ignoring errors from those already bound:

```
# for bdf in $(lspci -D -d 1924: | awk '{ print $1 }'); do \
  echo -n "${bdf}" > /sys/bus/pci/drivers/sfc/bind; done
```

Alternatively reload the sfc driver:

```
# onload_tool reload
```

or:

```
# modprobe -r sfc
# modprobe sfc
```

- 8 Run ifconfig -a again to find that all the interfaces are reported and all have been firmware upgraded without having to physically touch the server or change the kernel.

## Sfupdate: SLES 11 Limitation

Due to limitations in SLES 11 using kernel versions prior to 2.6.27.54 it is necessary to reboot the server after running the sfupdate utility to upgrade server firmware.

## Sfupdate: Command Line Options

Table 24 lists the options for `sfupdate`.

**Table 24: Sfupdate Options**

Option	Description
<code>-h, --help</code>	Shows help for the available options and command line syntax.
<code>-i, --adapter=ethX</code>	Specifies the target adapter when more than one adapter is installed in the localhost.  ethX = Adapter ifname or MAC address (as obtained with <code>--list</code> ).
<code>--list</code>	Shows the adapter ID, adapter name and MAC address of each adapter installed in the localhost.
<code>--write</code>	Re-writes the firmware from the images embedded in the <code>sfupdate</code> tool. To re-write using an external image, specify <code>--image=&lt;filename&gt;</code> in the command.  <code>--write</code> fails if the embedded image is the same or a previous version. To force a write in this case, specify <code>-force</code> in the command.
<code>--force</code>	Force the update of all firmware, even if the installed firmware version is the same as, or more recent than, the firmware embedded in <code>sfupdate</code> .
<code>--backup</code>	Backup existing firmware image before updating. This option may be used with <code>--write</code> and <code>--force</code> .
<code>--image=(filename)</code>	Update the firmware using the binary image from the given file rather than from those embedded in the utility.
<code>--ipxe-image=(filename)</code>	Install an iPXE image from the given file, replacing the Solarflare boot ROM image. <code>sfupdate</code> will not automatically replace the iPXE image in subsequent flash updates unless the <code>--restore-bootrom</code> option is used.
<code>--restore-bootrom</code>	Replace an iPXE image in flash with the standard Solarflare Boot Manager PXE image included in <code>sfupdate</code> .
<code>-y, --yes</code>	Update without prompting. This option can be used with the <code>--write</code> and <code>--force</code> options.
<code>-v, --verbose</code>	Verbose mode.



**Table 24: Sfupdate Options (continued)**

Option	Description
-s, --silent	Suppress output while the utility is running; useful when the utility is used in a script.
-V, --version	Display version information and exit.

## Sfupdate: Examples

- Display firmware versions for all adapters:

```
sfupdate
```

```
Solarstorm firmware update utility [v4.3.1]
Copyright Solarflare Communications 2006-2013, Level 5 Networks 2002-2005
```

```
eth4 - MAC: 00-0F-53-21-00-61
      Controller type:   Solarflare SFC9100-family
      Controller versoin: unknown
      Boot ROM version:  unknown
```

```
This utility contains more recent Boot ROM firmware [v4.2.1.1000]
```

```
- run "sfupdate --write" to perform an update
```

```
This utility contains more recent controller firmware [v4.2.1.1010]
```

```
- run "sfupdate --write" to perform an update
```

```
eth5 - MAC: 00-0F-53-21-00-60
      Controller type:   Solarflare SFC9100-family
      Controller version: unknown
      Boot ROM version:  unknown
```

```
This utility contains more recent Boot ROM firmware [v4.2.1.1000]
```

```
- run "sfupdate --write" to perform an update
```

```
This utility contains more recent controller firmware [v4.2.1.1010]
```

```
- run "sfupdate --write" to perform an update
```

- Update adapter firmware:
 

```
# sfupdate --adapter=<interface> --write
```
- Update adapter firmware + create a backup firmware image:
 

```
# sfupdate --adapter=<interface> --write --backup
```
- Update firmware to an earlier or the same version:
 

```
# sfupdate --adapter=<interface> --write --force
```

A backup firmware image file can be restored to the adapter using the '--image' option.

## 3.18 Installing an activation key with sfkey

The sfkey utility is distributed with the Linux Utilities RPM package. This utility is used to install Solarflare AppFlex™ activation keys and enable selected on-board services for Solarflare adapters. For more information about activation key requirements see [Solarflare AppFlex™ Technology on page 17](#).

### sfkey: Command Usage

```
# sfkey [--adapter=eth<N>] [options]
```

If the adapter option is not specified, operations will be applied to all installed adapters.

- To view all sfkey options:  
# sfkey --help
- To list (by key ID) all adapters that support activation keys:  
# sfkey --inventory --all

```
eth2: 714100101282140148200014
```

- To display an adapter's activation keys:  
# sfkey --adapter=eth2 --report

```
eth2: 714100101282140148200014 (Flareon)
Product name          Solarflare SFN7141Q QSFP+ Flareon Ultra Server Adapter
Installed keys      Onload
```

- To install a activation key:  
Copy the activation key to a .txt file on the target server. All keys can be in the same key file and the file applied on multiple servers. The following example uses an activation key file called keys.txt created on the local server.

```
# sfkey --adapter=eth2 --install keys.txt
```

```
Reading keys...
```

```
Writing all keys to eth2...
```

```
eth2: 714100101282140148200014 (Flareon)
Product name          Solarflare SFN7141Q QSFP+ Flareon Ultra Server Adapter
Installed keys      Onload, SolarCapture Pro, Capture SolarSystem
```

### Activation Keys Inventory

Use the combined --inventory and --keys options to identify the activation keys installed on an adapter.

```
# sfkey --adapter=eth2 --inventory --keys
```

```
eth2: 714100101282140148200014 (Flareon), $0NL, !PTP, !SCL, SCP, CSS, !SSFE, !PM, !NAC
```

Activation key information is displayed in *[Prefix] [AppID] [Suffix]* format.

Prefix:	<none>	Feature is active
	\$	Factory-fitted
	!	Not present
AppID:	An	Application ID number
	<name>	Application acronym
Suffix:	<none>	Feature is active
	+	Site activated
	~	Evaluation key
	*	Inactive key
	@	Inactive site key
	-	No state available

## sfkey Options

Table 25 describes all sfkey options.

**Table 25: sfkey options**

Option	Description
<code>--backup</code>	Output a report of the installed keys in all adapters. The report can be saved to file and later used with the <code>--install</code> option.
<code>--install &lt;filename&gt;</code>	Install activation keys from the given file and report the result. To read from stdin use “-” in place of filename. Keys are installed to an adapter, so if an adapter’s ports are eth4 and eth5, both ports will be affected by the keys installed.  <i>sfc driver reload is required after sfkey installs certain feature (e.g. a PTP key).</i>  To reload the sfc driver:  # <code>modprobe -r sfc; modprobe sfc</code>  or when Onload is installed:  # <code>onload_tool reload</code>
<code>--inventory</code>	List the adapters that support activation keys. To list all adapters use the <code>--all</code> option. To list keys use the <code>--keys</code> option.

**Table 25: sfkey options (continued)**

<b>Option</b>	<b>Description</b>
--keys	Include keys in --inventory output - see Inventory above.
--noevaluationupdate	Do not update any evaluation keys.
-a, --all	Apply sfkey operation to all adapters that support licensing.
-c, --clear	Delete all existing activation keys from an adapter - except factory installed keys.
-h, --help	Display all sfkey options.
-i, --adapter	identify specific adapter to apply sfkey operation to.
-r, --report	Display an adapter serial number and current activation key status (see example above).  Use with --all or with --adapter.  If an installed or active key is reported as 'An' (where n is a number), it indicates a key unknown to this version of sfkey - use an updated sfkey version.
-s, --silent	Silent mode, output errors only.
-v, --verbose	Verbose mode.
-V, --version	Display sfkey version and exit.
-x, --xml	Report formatted as XML.

## 3.19 Performance Tuning on Linux

- [Introduction on page 95](#)
- [Tuning settings on page 95](#)
- [Other Considerations on page 108](#)

### Introduction

The Solarflare family of network adapters are designed for high-performance network applications. The adapter driver is pre-configured with default performance settings that have been designed to give good performance across a broad class of applications. Occasionally, application performance can be improved by tuning these settings to best suit the application.

There are three metrics that should be considered when tuning an adapter:

- Throughput
- Latency
- CPU utilization

Different applications may be more or less affected by improvements in these three metrics. For example, transactional (request-response) network applications can be very sensitive to latency whereas bulk data transfer applications are likely to be more dependent on throughput.

The purpose of this section is to highlight adapter driver settings that affect the performance metrics described. This section covers the tuning of all Solarflare adapters.

Latency will be affected by the type of physical medium used: 10GBase-T, twinaxial (direct-attach), fiber or KX4. This is because the physical media interface chip (PHY) used on the adapter can introduce additional latency. Likewise, latency can also be affected by the type of SFP/SFP+/QSFP module fitted.

In addition, you may need to consider other issues influencing performance, such as application settings, server motherboard chipset, CPU speed, cache size, RAM size, additional software installed on the system, such as a firewall, and the specification and configuration of the LAN. Consideration of such issues is not within the scope of this guide.

### Tuning settings

#### Port mode

The selected port mode for SFN7000, SFN8000 and X2 series adapters and the U25 adapter should correspond to the speed and number of connectors in use, after using any splitter cables. If a restricted set of connectors is configured, the driver can then transfer resources from the unused connectors to those configured, potentially improving performance.

## Adapter MTU (Maximum Transmission Unit)

The default MTU of 1500 bytes ensures that the adapter is compatible with legacy 10/100Mbps Ethernet endpoints. However if a larger MTU is used, adapter throughput and CPU utilization can be improved. CPU utilization is improved, because it takes fewer packets to send and receive the same amount of data. Solarflare adapters support an MTU of up to 9216 bytes (this does not include the Ethernet preamble or frame-CRC).

Since the MTU should ideally be matched across all endpoints in the same LAN (VLAN), and since the LAN switch infrastructure must be able to forward such packets, the decision to deploy a larger than default MTU requires careful consideration. It is recommended that experimentation with MTU be done in a controlled test environment.

The MTU is changed dynamically using `ifconfig`, where `ethX` is the interface name and `<size>` is the MTU size in bytes:

```
# /sbin/ifconfig <ethX> mtu <size>
```

Verification of the MTU setting may be performed by running `ifconfig` with no options and checking the MTU value associated with the interface. The change in MTU size can be made to persist across reboots by editing the file `/etc/sysconfig/network-scripts/ifcfg-ethX` and adding `MTU=<mtu>` on a new line.

## Interrupt Moderation (Interrupt Coalescing)

*Interrupt moderation* reduces the number of interrupts generated by the adapter by coalescing multiple received packet events and/or transmit completion events together into a single interrupt.

The *interrupt moderation interval* sets the minimum time (in microseconds) between two consecutive interrupts. Coalescing occurs only during this interval:

- When the driver generates an interrupt, it starts timing the moderation interval.
- Any events that occur before the moderation interval expires are coalesced together into a single interrupt, that is raised only when the interval expires. A new moderation interval then starts, during which no interrupt is raised.
- An event that occurs after the moderation interval has expired gets its own dedicated interrupt, that is raised immediately. A new moderation interval then starts, during which no interrupt is raised.

Solarflare adapters, by default, use an *adaptive algorithm* where the interrupt moderation delay is automatically adjusted between zero (no interrupt moderation) and 60 microseconds. The adaptive algorithm detects latency sensitive traffic patterns and adjusts the interrupt moderation interval accordingly.

Interrupt moderation settings are **critical for tuning adapter latency**:

- Disabling the adaptive algorithm will:
  - reduce jitter
  - allow setting the moderation interval as required to suit conditions.
- Increasing the interrupt moderation interval will:
  - generate less interrupts
  - reduce CPU utilization (because there are less interrupts to process)
  - increase latency
  - improve peak throughput.
- Decreasing the interrupt moderation interval will:
  - generate more interrupts
  - increase CPU utilization (because there are more interrupts to process)
  - decrease latency
  - reduce peak throughput.
- Turning off interrupt moderation will:
  - generate the most interrupts
  - give the highest CPU utilization
  - give the lowest latency
  - give the biggest reduction in peak throughput.

For many transaction request-response type network applications, the benefit of reduced latency to overall application performance can be considerable. Such benefits typically outweigh the cost of increased CPU utilization. It is recommended that:

- Interrupt moderation is disabled for applications that require best latency and jitter performance, such as market data handling.
- Interrupt moderation is enabled for high throughput single (or few) connection TCP streaming applications, such as iSCSI.

Interrupt moderation can be changed using `ethtool`, where `ethX` is the interface name. Before adjusting the interrupt moderation interval, it is recommended to disable adaptive moderation:

```
ethtool -C <ethX> adaptive-rx off
```

To set the RX interrupt moderation interval in microseconds ( $\mu$ s):

```
ethtool -C <ethX> rx-usecs <interval>
```

To turn off interrupt moderation, set an interval of zero (0):

```
ethtool -C <ethX> rx-usecs 0
```

The above example also sets the transmit interrupt moderation interval, unless the driver module parameter `separate_tx_channels` is enabled. (Normally packet RX and TX completions will share interrupts, so RX and TX interrupt moderation intervals must be equal, and the adapter driver automatically adjusts tx-usecs to match rx-usecs.) Refer to [Table 30 on page 115](#).

To set the TX interrupt moderation interval, if `separate_tx_channels` is enabled:

```
ethtool -C <ethX> tx-usecs <interval>
```

Interrupt moderation settings can be checked using `ethtool -c`.



**NOTE:** The performance benefits of TCP Large Receive Offload are limited if interrupt moderation is disabled. See [TCP Large Receive Offload \(LRO\) on page 99](#).

### TCP/IP Checksum Offload

Checksum offload moves calculation and verification of IP Header, TCP and UDP packet checksums to the adapter. The driver has all checksum offload features enabled by default. Therefore, there is no opportunity to improve performance from the default.

Checksum offload is controlled using `ethtool`:

- Receive Checksum:  
# `/sbin/ethtool -K <ethX> rx <on|off>`
- Transmit Checksum:  
# `/sbin/ethtool -K <ethX> tx <on|off>`

Verification of the checksum settings may be performed by running `ethtool` with the `-k` option.



**NOTE:** Solarflare recommend you do not disable checksum offload.

### TCP Segmentation Offload (TSO)

TCP Segmentation Offload (TSO) offloads the splitting of outgoing TCP data into packets to the adapter. TSO benefits applications using TCP. Applications using protocols other than TCP will not be affected by TSO.

Enabling TSO will reduce CPU utilization on the transmit side of a TCP connection and improve peak throughput, if the CPU is fully utilized. Since TSO has no effect on latency, it can be enabled at all times. The driver has TSO enabled by default. Therefore, there is no opportunity to improve performance from the default.

TSO is controlled using `ethtool`:

```
# /sbin/ethtool -K <ethX> tso <on|off>
```

Verification of the TSO settings may be performed by running `ethtool` with the `-k` option.

TCP and IP checksum offloads must be enabled for TSO to work.



**NOTE:** Solarflare recommend that you do not disable this setting.



## TCP Large Receive Offload (LRO)

TCP Large Receive Offload (LRO) is a feature whereby the adapter coalesces multiple packets received on a TCP connection into a single larger packet before passing this onto the network stack for receive processing. This reduces CPU utilization and improves peak throughput when the CPU is fully utilized. The effectiveness of LRO is bounded by the interrupt moderation delay, and is limited if interrupt moderation is disabled (see [Interrupt Moderation \(Interrupt Coalescing\) on page 96](#)). Enabling LRO does not itself negatively impact latency.



**NOTE:** The Solarflare network adapter driver enables LRO by default. By its design, LRO is of greater benefit when working with smaller packets. For Solarflare adapter, LRO will become disabled if the MTU is set larger than 3979. When the MTU is set larger than 3978, LRO cannot be enabled and will be reported as 'fixed disabled' by ethtool.



**NOTE:** LRO should **NOT** be enabled when using the host to forward packets from one interface to another. For example, if the host is performing IP routing.



**NOTE:** It has been observed that as RHEL6 boots the libvirtd daemon changes the default forwarding setting such that LRO is disabled on all network interfaces. This behavior is undesirable as it will potentially lower bandwidth and increase CPU utilization - especially for high bandwidth streaming applications.

To determine if LRO is enabled on an interface:

```
ethtool -k ethX
```

If IP forwarding is not required on the server, Solarflare recommends either:

- Disabling the libvirtd service (if this is not being used),
- Or, as root before loading the Solarflare driver:  

```
sysctl -w net.ipv4.conf.default.forwarding=0
```

 (This command can be loaded into `/etc/rc.local`),
- Or, after loading the Solarflare driver, turn off forwarding for only the Solarflare interfaces and re-enable LRO:  

```
sysctl -w net.ipv4.conf.ethX.forwarding=0
```

```
ethtool -K ethX lro on
```

 (where X is the id of the Solarflare interface).

Disabling the libvirtd service is a permanent solution, whereas the other recommendations are temporary and will not persist over reboot.

LRO should not be enabled if IP forwarding is being used on the same interface as this could result in incorrect IP and TCP operation.

LRO can be controlled using the module parameter `lro`. Add the following line to `/etc/modprobe.conf` or add the options line to a file under the `/etc/modprobe.d` directory to disable LRO:

```
options sfc lro=0
```

Then reload the driver so it picks up this option:

```
rmmod sfc
modprobe sfc
```

The current value of this parameter can be found by running:

```
cat /sys/module/sfc/parameters/lro
```

LRO can also be controlled on a per-adapter basis by writing to this file in sysfs:

```
/sys/class/net/ethX/device/lro
```

- To disable LRO:  
`echo 0 > /sys/class/net/ethX/device/lro`
- To enable LRO:  
`echo 1 > /sys/class/net/ethX/device/lro`
- To show the current value of the per-adapter LRO state:  
`cat /sys/class/net/ethX/device/lro`

Modifying this file instantly enables or disables LRO, no reboot or driver reload is required. This setting takes precedence over the `lro` module parameter

Current LRO settings can be identified with Linux `ethtool` e.g.

```
ethtool -k ethX
```

TCP and IP checksum offloads must be enabled for LRO to work.

### **The performance\_profile module option**

A performance profile can be set to optimize adapter performance either for low latency or for high throughput.

Set the `performance_profile` module parameter to `latency` or `throughput` as required. Add the following line to `/etc/modprobe.conf`, or add the options line to a file under the `/etc/modprobe.d` directory to optimize for throughput:

```
options sfc performance_profile=throughput
```

Then reload the driver so it picks up this option:

```
rmmod sfc
modprobe sfc
```

The current value of this parameter can be found by running:

```
cat /sys/module/sfc/parameters/performance_profile
```

For further information, see [Table 30 on page 115](#).

## TCP Protocol Tuning

TCP Performance can also be improved by tuning kernel TCP settings. Settings include adjusting send and receive buffer sizes, connection backlog, congestion control, etc.

For Linux kernel versions, including 2.6.16 and later, initial buffering settings should provide good performance. However for earlier kernel versions, and for certain applications even on later kernels, tuning buffer settings can significantly benefit throughput. To change buffer settings, adjust the `tcp_rmem` and `tcp_wmem` using the `sysctl` command:

- Receive buffering:  
`sysctl net.ipv4.tcp_rmem="<min> <default> <max>"`
- Transmit buffering:  
`sysctl net.ipv4.tcp_wmem="<min> <default> <max>"`

(`tcp_rmem` and `tcp_wmem` can also be adjusted for IPV6 and globally with the `net.ipv6` and `net.core` variable prefixes respectively).

Typically it is sufficient to tune just the max buffer value. It defines the largest size the buffer can grow to. Suggested alternate values are `max=500000` (1/2 Mbyte). Factors such as link latency, packet loss and CPU cache size all influence the affect of the max buffer size values. The minimum and default values can be left at their defaults `minimum=4096` and `default=87380`.

## Buffer Allocation Method

The Solarflare driver has a single optimized buffer allocation strategy. This replaces the two different methods controlled with the `rx_alloc_method` driver module parameter which were available using 3.3 and previous drivers.

The net driver continues to expose the `rx_alloc_method` module option, but the value is ignored and it only exists to not break existing customer configurations.

## TX PIO

PIO (programmed input/output) describes the process where data is directly transferred by the CPU to or from an I/O device. It is an alternative technique to the I/O device using bus master DMA to transfer data without CPU involvement.

SFN7000, SFN8000 and X2 series adapters and the U25 adapter support TX PIO, where packets on the transmit path can be “pushed” to the adapter directly by the CPU. This improves the latency of transmitted packets but can cause a very small increase in CPU utilization. TX PIO is therefore especially useful for smaller packets.

The TX PIO feature is enabled by default for packets up to 256 bytes. The maximum packet size that can use PIO can be configured with the driver module option `piobuf_size`.

## CTPIO

Supported on the XtremeScale X2 series adapters and the U25 adapter, cut-through PIO delivers the lowest transmit latency when packets are transmitted on the wire while still being streamed over the PCI interface from the host.

For further details, refer to the Onload User Guide (SF-104474-CD).

## 3.20 Web Server - Driver Optimization

### Introduction

The Solarflare net driver from version 4.4.1.1017 includes optimizations aimed specifically at web service providers and cloud based applications.

Tuning recommendations are documented in [Table 26](#) for users concerned with Content Delivery Networks (CDN), HTTP web hosting application technologies such as HA Proxy, nginx and HTTP web servers.

Performance improvements have been observed in the following areas:

- increased the rate at which servers can process new HTTP connections
- increased the rate at which servers can process HTTP requests
- increased sustained throughput when processing large files via HTTP
- improved kernel throughput performance

Customers requiring further details or to access test data should send an email to [support@solarflare.com](mailto:support@solarflare.com).

### Driver Tuning

Whilst most driver enhancements are internal changes, transparent and non-configurable by the user, the following driver module options can be used to tune the driver for particular user applications.

- `rss_numa_local`  
Using the 4.4.1.1017 driver this option is enabled by default. This will restrict RSS to use CPU cores only on the NUMA node closest to the adapter. This is particularly important for processors supporting DDIO.  
RSS channels not on the local NUMA node can still be accessed using the `ethtool -U` commands to identify a core (action) on which to process the specified `ethtool ntuple` filter traffic. For example if `rss_cpus=cores`, then an RSS receive channel and associated MSI-X interrupt is created for every core.
- `rx_recycle_ring_size`  
The default value for the maximum number of receive buffers to recycle pages for has been changed to 512, and in newer drivers will be further increased to 1024.
- `rx_copybreak`  
A default value of 192 bytes has been selected as the maximum size of packet (bytes) that will be copied directly to the network stack.

Driver module options can be enabled in a user-created file (e.g `sfc.conf`) in the `/etc/modprobe.d` directory, for example:

```
options sfc rss_numa_local=1
options sfc rx_recycle_ring_size=512
```

For further descriptions and to list all sfc driver module options:

```
# modinfo sfc
```

## nginx Tuning

**Table 26: nginx Server Tuning**

Tuning	Notes
SO_REUSEPORT	Solarflare testing involving nginx used version v1.7.9 with applied patch to support so_reuseport. See the following link for details: <a href="http://forum.nginx.org/read.php?29,241283,241283">http://forum.nginx.org/read.php?29,241283,241283</a> .
rss_cpus=N	Create N receive queues where N=(number of logical cores)/2.  See <a href="#">Receive Side Scaling (RSS) on page 70</a> for options.
rss_numa_local=1	On SMP systems it is recommended to have all interrupts on the NUMA node local to the Solarflare adapter: <code>rss_numa-local=1</code> , and pin nginx threads to the free CPUs even when these are on the non-local node.  When this is not possible, CPU cores can be divided equally between interrupts and nginx threads.  <code>rss_numa_local=1</code> is the default setting.
Pinning threads	Application threads and interrupts should not be pinned to the same CPU cores.
<code>ethtool -C &lt;interface&gt; adaptive-rx off</code>	Disable the irq-balance service to prevent re-distribution of interrupts by the kernel. Disable adaptive interrupt moderation before setting the interrupt moderation interval.
<code>ethtool -C &lt;interface&gt; rx-usecs 60</code>	Set the interrupt moderation interval.  When processing smaller packets it is generally better to set a higher interval i.e. 60µsecs and for larger packets a lower interval or even zero to disable interrupt moderation.  See <a href="#">Interrupt Moderation (Interrupt Coalescing) on page 96</a> .

## Adapters - Software Support

To benefit from recent driver optimizations, the following (minimum) net driver and firmware versions should be used:

```
# ethtool -i eth<N>
driver: sfc
version: 4.4.1.1017
firmware-version: 4.4.2.1011 rx1 tx1
```

For latency sensitive applications, the adapter firmware variant should be set with the `sfboot` utility to ultra-low-latency:

```
# sfboot --adapter=eth<N> firmware-variant=ultra-low-latency
```

The ultra-low-latency firmware variant is being used when the output from `ethtool` (above) shows the `rx1` and `tx1` values.

*A reboot of the server is required after changes using `sfboot`.*

### 3.21 Interrupt Affinity

Interrupt affinity describes the set of host CPUs that may service a particular interrupt.

This affinity therefore dictates the CPU context where received packets will be processed and where transmit packets will be freed once sent. If the application can process the received packets in the same CPU context by being affinitized to the relevant CPU, then latency and CPU utilization can be improved. This improvement is achieved because well tuned affinities reduce inter-CPU communication.

Tuning interrupt affinity is most relevant when MSI-X interrupts and RSS are being used. The `irqbalance` service, which typically runs by default in most Linux distributions, is a service that automatically changes interrupt affinities based on CPU workload.

In many cases the `irqbalance` service hinders rather than enhances network performance. It is therefore necessary to disable it and then set interrupt affinities.

- To disable `irqbalance` permanently, run:  
`/sbin/chkconfig -level 12345 irqbalance off`
- To see whether `irqbalance` is currently running, run:  
`/sbin/service irqbalance status`
- To disable `irqbalance` temporarily, run:  
`/sbin/service irqbalance stop`

Once the `irqbalance` service has been stopped, the Interrupt affinities can be configured manually.



**NOTE:** The Solarflare driver will evenly distribute interrupts across the available host CPUs (based on the `rss_cpus` module parameter).

To use the Solarflare driver default affinities (recommended), the irqbalance service must be disabled before the Solarflare driver is loaded (otherwise it will immediately overwrite the affinity configuration values set by the Solarflare driver).

### Example 1:

How affinities should be manually set will depend on the application. For a single streamed application such as Netperf, one recommendation would be to affinity all the Rx queues and the application on the same CPU. This can be achieved with the following steps:

- 1 Determine which interrupt line numbers the network interface uses. Assuming the interface is eth0, this can be done with:

```
# cat /proc/interrupts | grep eth0-
123:      13302      0      0      0      PCI-MSI-X  eth0-0
131:         0       24      0      0      PCI-MSI-X  eth0-1
139:         0        0     32      0      PCI-MSI-X  eth0-2
147:         0        0      0     21      PCI-MSI-X  eth0-3
```

This output shows that there are four channels (rows) set up between four CPUs (columns).

- 2 Determine the CPUs to which these interrupts are assigned to:

```
# cat /proc/irq/123/smp_affinity
00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000001
# cat /proc/irq/131/smp_affinity
00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000002
# cat /proc/irq/139/smp_affinity
00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000004
# cat /proc/irq/147/smp_affinity
00000000,00000000,00000000,00000000,00000000,00000000,00000000,00000008
```

This shows that RXQ[0] is affinity to CPU[0], RXQ[1] is affinity to CPU[1], and so on. With this configuration, the latency and CPU utilization for a particular TCP flow will be dependant on that flow's RSS hash, and which CPU that hash resolves onto.



**NOTE:** Interrupt line numbers and their initial CPU affinity are not guaranteed to be the same across reboots and driver reloads. Typically, it is therefore necessary to write a script to query these values and apply the affinity accordingly.

- 3 Set all network interface interrupts to a single CPU (in this case CPU[0]):

```
# echo 1 > /proc/irq/123/smp_affinity
# echo 1 > /proc/irq/131/smp_affinity
# echo 1 > /proc/irq/139/smp_affinity
# echo 1 > /proc/irq/147/smp_affinity
```



**NOTE:** The read-back of /proc/irq/N/smp\_affinity will return the old value until a new interrupt arrives.

- 4 Set the application to run on the same CPU (in this case CPU[0]) as the network interface's interrupts:

```
# taskset 1 netperf
# taskset 1 netperf -H <host>
```





**NOTE:** The use of taskset is typically only suitable for affinity tuning single threaded, single traffic flow applications. For a multi threaded application, whose threads for example process a subset of receive traffic, taskset is not suitable. In such applications, it is desirable to use RSS and Interrupt affinity to spread receive traffic over more than one CPU and then have each receive thread bind to each of the respective CPUs. Thread affinities can be set inside the application with the `shed_setaffinity()` function (see Linux man pages). Use of this call and how a particular application can be tuned is beyond the scope of this guide.

If the settings have been correctly applied, all interrupts from `eth0` are being handled on `CPU[0]`. This can be checked:

```
# cat /proc/interrupts | grep eth0-
123:      13302          0          0          0          PCI-MSI-X eth0-0
131:         0          24          0          0          PCI-MSI-X eth0-1
139:         0          0          32          0          PCI-MSI-X eth0-2
147:         0          0          0          21          PCI-MSI-X eth0-3
```

### Example 2:

An example of affinitizing each interface to a CPU on the same package:

First identify which interrupt lines are servicing which CPU and IO device:

```
# cat /proc/interrupts | grep eth0-
123:      13302          0  1278131          0          PCI-MSI-X eth0-0
# cat /proc/interrupts | grep eth1-
131:         0          24          0          0          PCI-MSI-X eth1-0
```

Find CPUs on same package (have same 'package-id'):

```
# more /sys/devices/system/cpu/cpu*/topology/physical_package_id
:::
/sys/devices/system/cpu/cpu0/topology/physical_package_id
:::
1
:::
/sys/devices/system/cpu/cpu10/topology/physical_package_id
:::
1
:::
/sys/devices/system/cpu/cpu11/topology/physical_package_id
:::
0
...
```

Having determined that `cpu0` and `cpu10` are on package 1, we can assign each `ethX` interface's MSI-X interrupt to its own CPU on the same package. In this case we choose package 1:

```
# echo 1 > /proc/irq/123/smp_affinity      # 1hex is bit 0 = CPU0
# echo 400 > /proc/irq/131/smp_affinity    # 400hex is bit 10 = CPU10
```

## Other Considerations

### PCI Express Lane Configurations

The PCI Express (PCIe) interface used to connect the adapter to the server can function at different speeds and widths. This is independent of the physical slot size used to connect the adapter. The possible widths are multiples x1, x2, x4, x8 and x16 lanes of (2.5Gbps for PCIe Gen 1, 5.0 Gbps for PCIe Gen 2 and 8.0Gbps for PCIe Gen 3) in each direction. *Solarflare adapters are designed for x8 or x16 lane operation.*

On some server motherboards, choice of PCIe slot is important. This is because some slots (including those that are physically x8 or x16 lanes) may only electrically support x4 lanes. In x4 lane slots, Solarflare PCIe adapters will continue to operate, but not at full speed. The Solarflare driver will warn if it detects that the adapter is plugged into a PCIe slot which electrically has fewer than x8 lanes.

Solarflare adapters require a PCIe Gen 3 x8 or x16 slot for optimal performance. The Solarflare driver will warn if it detects that the adapter is placed in a sub-optimal slot.

Warning messages can be viewed in `dmesg` from `/var/log/messages`.

The `lspci` command can be used to discover the currently negotiated PCIe lane width and speed:

```
lspci -d 1924: -vv
02:00.1 Class 0200: Unknown device 1924:0710 (rev 01)
...
Link: Supported Speed 2.5Gb/s, Width x8, ASPM L0s, Port 1
Link: Speed 2.5Gb/s, Width x8
```



**NOTE:** The Supported speed may be returned as 'unknown', due to older `lspci` utilities not knowing how to determine that a slot supports PCIe Gen. 2.0/5.0 Gb/s or PCIe Gen 3.0/8.0 Gb/s.

In addition, the latency of communications between the host CPUs, system memory and the Solarflare PCIe adapter may be PCIe slot dependent. Some slots may be “closer” to the CPU, and therefore have lower latency and higher throughput. If possible, install the adapter in a slot which is local to the desired NUMA node

Please consult your server user guide for more information.

### CPU Speed Service

Most Linux distributions will have the `cpuspeed` service running by default. This service controls the CPU clock speed dynamically according to current processing demand. For latency sensitive applications, where the application switches between having packets to process and having periods of idle time waiting to receive a packet, dynamic clock speed control may increase packet latency. Solarflare recommend disabling the `cpuspeed` service if minimum latency is the main consideration.

The service can be disabled temporarily:

```
/sbin/service cpuspeed stop
```

The service can be disabled across reboots:

```
/sbin/chkconfig --level 12345 cpuspeed off
```

### **CPU Power Service**

On RHEL7 systems, `cpuspeed` is replaced with `cpupower`. Solarflare recommend disabling the `cpupower` service if minimum latency is the main consideration. The service is controlled via `systemctl`:

```
systemctl stop cpupower  
systemctl disable cpupower
```

### **Tuned Service**

On RHEL7 systems, it may be beneficial to disable the `tuned` service if minimum latency is the main consideration. Users are advised to experiment. The service is controlled via `systemctl`:

```
systemctl stop tuned  
systemctl disable tuned
```

### **Busy poll**

If the kernel supports the *busy poll* features (Linux 3.11 or later), and minimum latency is the main consideration, Solarflare recommend that the `busy_poll` socket options should be enabled with a value of 50 microseconds as follows:

```
sysctl net.core.busy_poll=50 && sysctl net.core.busy_read=50
```

Only sockets having a non-zero value for `SO_BUSY_POLL` will be polled, so the user should do one of the following:

- set the poll timeout with the global `busy_read` option, as shown above,
- set the per-socket `SO_BUSY_POLL` socket option on selected sockets.

Setting `busy_read` also sets the default value for the `SO_BUSY_POLL` option.

### **Memory bandwidth**

Many chipsets use multiple channels to access main system memory. Maximum memory performance is only achieved when the chipset can make use of all channels simultaneously. This should be taken into account when selecting the number of memory modules (DIMMs) to populate in the server. For optimal memory bandwidth in the system, it is likely that:

- all DIMM slots should be populated
- all NUMA nodes should have memory installed.

Please consult the motherboard documentation for details.

## Intel® QuickData / NetDMA

On systems that support Intel I/OAT (I/O Acceleration Technology) features such as QuickData (a.k.a NetDMA), Solarflare recommend that these are enabled as they are rarely detrimental to performance.

Using Intel® QuickData Technology allows data copies to be performed by the system and not the operating system. This enables data to move more efficiently through the server and provide fast, scalable, and reliable throughput.

### Enabling QuickData

- On some systems the hardware associated with QuickData must first be enabled (once only) in the BIOS
- Load the QuickData drivers with `modprobe ioatdma`

### Server Motherboard, Server BIOS, Chipset Drivers

Tuning or enabling other system capabilities may further enhance adapter performance. Readers should consult their server user guide. Possible opportunities include tuning PCIe memory controller (PCIe Latency Timer setting available in some BIOS versions).

## Tuning Recommendations

The following tables provide recommendations for tuning settings for different applications.

- Throughput - [Table 27 on page 110](#)
- Latency - [Table 28 on page 112](#)
- Forwarding - [Table 29 on page 113](#)

### Recommended Throughput Tuning

**Table 27: Throughput Tuning Settings**

Tuning Parameter	How?
MTU Size	Configure to maximum supported by network: <code>/sbin/ifconfig &lt;ethX&gt; mtu &lt;size&gt;</code>
Interrupt moderation	Leave at default (Enabled).
TCP/IP Checksum Offload	Leave at default (Enabled).
TCP Segmentation Offload	Leave at default (Enabled).
TCP Large Receive Offload	Leave at default (Enabled).
<code>performance_profile</code>	Set to throughput.

**Table 27: Throughput Tuning Settings (continued)**

<b>Tuning Parameter</b>	<b>How?</b>
TCP Protocol Tuning	<p>Leave at default for 2.6.16 and later kernels.</p> <p>For earlier kernels:</p> <pre>sysctl net.core.tcp_rmem 4096 87380 524288 sysctl net.core.tcp_wmem 4096 87380 524288</pre>
Receive Side Scaling (RSS)	Application dependent
Interrupt affinity & irqbalance service	<p>Interrupt affinity settings are application dependent</p> <p>Stop irq balance service:</p> <pre>/sbin/service irqbalance stop</pre> <p>Reload the drivers to use the driver default interrupt affinity.</p>
Buffer Allocation Method	<p>Leave at default. Some applications may benefit from specific setting.</p> <p>The Solarflare driver now supports a single optimized buffer allocation strategy and any value set by the rx_alloc_method parameter is ignored.</p>
PCI Express Lane Configuration	Ensure the adapter is in an x8 slot (2.0 or later), and that current speed (not the supported speed) reads back as “x8 and 5GT/s”, or “x8 and 8GT/s”, or “x8 and Unknown”.
CPU Speed Service (cpuspeed)	Leave enabled.
Memory bandwidth	Ensure memory utilizes all memory channels on system motherboard.
Intel QuickData (Intel chipsets only)	<p>Enable in BIOS and install driver:</p> <pre>modprobe ioatdma</pre>

## Recommended Latency Tuning

Table 28 shows recommended tuning settings for latency:

**Table 28: Latency Tuning Settings**

Tuning Parameter	How?
MTU Size	Configure to maximum supported by network: <code>/sbin/ifconfig &lt;ethX&gt; mtu &lt;size&gt;</code>
Interrupt moderation	Disable with: <code>ethtool -C &lt;ethX&gt; rx-usecs-inq 0</code>
TCP/IP Checksum Offload	Leave at default (Enabled).
TCP Segmentation Offload	Leave at default (Enabled).
TCP Large Receive Offload	Disable using sysfs: <code>echo 0 &gt; /sys/class/net/ethX/device/lro</code>
TCP Protocol Tuning	Leave at default, but changing does not impact latency.
Receive Side Scaling	Application dependent.
Interrupt affinity & irqbalance service	Interrupt affinity settings are application dependent Stop irq balance service: <code>/sbin/service irqbalance stop</code> Reload the drivers to use the driver default interrupt affinity.
Buffer Allocation Method	Leave at default. Some applications may benefit from specific setting.  The Solarflare driver now supports a single optimized buffer allocation strategy and any value set by the <code>rx_alloc_method</code> parameter is ignored.
PCI Express Lane Configuration	Ensure the adapter is in an x8 slot (2.0 or later), and that current speed (not the supported speed) reads back as “x8 and 5GT/s”, or “x8 and 8GT/s”, or “x8 and Unknown”.
CPU Speed Service (cpuspeed)	Disable with: <code>/sbin/service cpuspeed stop</code>
CPU Power Service (cpupower)	Disable with: <code>systemctl stop cpupower</code> <code>systemctl disable cpupower</code>

**Table 28: Latency Tuning Settings (continued)**

Tuning Parameter	How?
Tuned Service	Experiment disabling this with:  systemctl stop tuned systemctl disable tuned
Busy poll (Linux 3.11 and later)	Enable with a value of 50µs:  sysctl net.core.busy_poll=50 \ && sysctl net.core.busy_read=50
Memory bandwidth	Ensure memory utilizes all memory channels on system motherboard.
Intel QuickData (Intel chipsets only)	Enable in BIOS and install driver:  modprobe ioatdma

### Recommended Forwarding Tuning

Table 29 shows recommended tuning settings for forwarding

**Table 29: Forwarding Tuning Settings**

Tuning Parameter	How?
MTU Size	Configure to maximum supported by network:  /sbin/ifconfig <ethX> mtu <size>
Interrupt moderation	Configure an explicit interrupt moderation interval by setting the following driver options (see <a href="#">Driver Tuning on page 103</a> ):  irq_adapt_enable=0 tx_irq_mod_usec=150
TCP/IP Checksum Offload	Leave at default (Enabled).
TCP Segmentation Offload	Leave at default (Enabled).
TCP Large Receive Offload	Disable using sysfs:  echo 0 > /sys/class/net/ethX/device/lro
performance_profile	Set to latency.
TCP Protocol Tuning	Leave at default for 2.6.16 and later kernels.  For earlier kernels:  sysctl net.core.tcp_rmem 4096 87380 524288 sysctl net.core.tcp_wmem 4096 87380 524288

**Table 29: Forwarding Tuning Settings (continued)**

Tuning Parameter	How?
Receive Side Scaling (RSS)	<p>Leave the <code>rss_cpus</code> option at the default, to use all CPUs for RSS.</p> <p>Ensure the <code>rss_numa_local</code> driver option is set to its default value of 1 (see <a href="#">Driver Tuning on page 103</a>).</p>
Interrupt affinity & irqbalance service	<p>Interrupt affinity. Affinitize each ethX interface to its own CPU (if possible select CPU's on the same Package). Refer to <a href="#">Interrupt Affinity on page 105</a>.</p> <p>Stop irqbalance service:</p> <pre>/sbin/service irqbalance stop</pre>
Buffer Allocation Method	<p>Leave at default. Some applications may benefit from specific setting.</p> <p>The Solarflare driver now supports a single optimized buffer allocation strategy and any value set by the <code>rx_alloc_method</code> parameter is ignored.</p>
Buffer Recycling	<p>Make receive buffer recycling more aggressive by setting the following driver option (see <a href="#">Driver Tuning on page 103</a>):</p> <pre>rx_recycle_ring_size=256</pre>
PIO	<p>Disable PIO by setting the following driver option (see <a href="#">Driver Tuning on page 103</a>):</p> <pre>piobuf_size=0</pre>
Transmit push	<p>Disable transmit push by setting the following driver option (see <a href="#">Driver Tuning on page 103</a>):</p> <pre>tx_push_max_fill=0</pre>



**Table 29: Forwarding Tuning Settings (continued)**

Tuning Parameter	How?
Direct copying	Disable copying directly from the network stack for transmits by setting the following driver option (see <a href="#">Driver Tuning on page 103</a> ):  tx_copybreak=0
Ring sizes	Change the number of descriptor slots on each ring by setting the following driver options (see <a href="#">Driver Tuning on page 103</a> ):  tx_ring=512 rx_ring=512  Note that as the tx_irq_mod_usec interrupt moderation interval increases, the number of required tx_ring and rx_ring descriptor slots also increases. Insufficient descriptor slots will cause dropped packets.

## 3.22 Module Parameters

[Table 30](#) lists the available parameters in the Solarflare Linux driver module (`modinfo sfc`):

**Table 30: Driver Module Parameters**

Parameter	Description	Possible Value	Default Value
piobuf_size	Identify the largest packet size that can use PIO.  Setting this to zero effectively disables PIO	uint	256 bytes
rx_alloc_method	Allocation method used for RX buffers.  The Solarflare driver now supports a single optimized buffer allocation strategy and any value set by the rx_alloc_method parameter is ignored.  See <a href="#">Buffer Allocation Method on page 102</a> .	uint	AVN(0) new kernels.  PAGE(2) old kernels
rx_refill_threshold	RX descriptor ring fast/slow fill threshold (%).	uint	90
lro_table_size <sup>1</sup>	Size of the LRO hash table. Must be a power of 2.	uint	128

**Table 30: Driver Module Parameters (continued)**

Parameter	Description	Possible Value	Default Value
lro_chain_max <sup>1</sup>	Maximum length of chains in the LRO hash table.	uint	20
lro_idle_jiffies <sup>1</sup>	Time (in jiffies) after which an idle connection's LRO state is discarded.	uint	101
lro_slow_start_packets <sup>1</sup>	Number of packets that must pass in-order before starting LRO.	uint	20000
lro_loss_packets <sup>1</sup>	Number of packets that must pass in-order following loss before restarting LRO.	uint	20
rx_desc_cache_size	Set RX descriptor cache size.	int	64
tx_desc_cache_size	Set TX descriptor cache size.	int	16
rx_xoff_thresh_bytes	RX fifo XOFF threshold.	int	-1 (auto)
rx_xon_thresh_bytes	RX fifo XON threshold.	int	-1 (auto)
lro	Large receive offload acceleration	int	1
separate_tx_channels	Use separate channels for TX and RX	uint	0
rss_cpus	Number of CPUs to use for Receive-Side Scaling, or 'packages', 'cores' or 'hyperthreads'	uint or string	<empty>
irq_adapt_enable	Enable adaptive interrupt moderation	uint	1
irq_adapt_low_thresh	Threshold score for reducing IRQ moderation	uint	10000
irq_adapt_high_thresh	Threshold score for increasing IRQ moderation	uint	20000
irq_adapt_irqs	Number of IRQs per IRQ moderation adaptation	uint	1000
napi_weight	NAPI weighting	uint	64
rx_irq_mod_usec	Receive interrupt moderation (microseconds)	uint	60
tx_irq_mod_usec	Transmit interrupt moderation (microseconds)	uint	150
allow_load_on_failure	If set then allow driver load when online self-tests fail	uint	0
onload_offline_selftest	Perform offline self-test on load	uint	1
interrupt_mode	Interrupt mode (0=MSIX, 1=MSI, 2=legacy)	uint	0
falcon_force_internal_sram	Force internal SRAM to be used	int	0

**Table 30: Driver Module Parameters (continued)**

Parameter	Description	Possible Value	Default Value
rss_numa_local	Constrain RSS to use CPU cores on the NUMA node local the Solarflare adapter.  Set to 1 to restrict, 0 otherwise.	0 1	1
max_vfs	Enable VFs in the net driver.  When specified as a single integer the VF count will be applied to all PFs.  When specified as a comma separated list, the first VF count is assigned to the PF with the lowest index i.e. the lowest MAC address, then the PF with the next highest MAC address etc.	uint	0
performance_profile	Tune settings for different performance profiles: <ul style="list-style-type: none"> <li>'throughput' for high throughput</li> <li>'latency' for low latency</li> <li>'auto' to select one of the above profiles based on the installed activation keys.</li> </ul>	string	auto

1. Check OS documentation for availability on SUSE and RHEL versions.

## 3.23 Linux ethtool Statistics

The Linux command `ethtool` will display an extensive range of statistics originated from the MAC on the Solarflare network adapter. To display statistics use the following command:

```
ethtool -S ethX
```

(where X is the ID of the Solarflare interface)

Using a Solarflare net driver earlier than version 4.4.1.1017, the `ethtool` statistics counters can be reset by reloading the `sfc` driver:

```
# modprobe -r sfc
# modprobe sfc
```

Drivers from version 4.4.1.1017 (included in `onload-201502`) have to manage multi-PF configurations and for this reason statistics are not reset by reloading the driver. The only methods currently available to reset stats is to cold-reboot (power OFF/ON) the server or reload the firmware image.

Per port statistics (`port_`) are from the physical adapter port. Other statistics are from the specified PCIe function.

Table 31 below lists the complete output from the `ethtool -S` command.



**NOTE:** `ethtool -S` output depends on the features supported by the adapter type

**Table 31: Ethtool -S output**

Field	Description
<code>port_tx_bytes</code>	Number of bytes transmitted.
<code>port_tx_packets</code>	Number of packets transmitted.
<code>port_tx_pause</code>	Number of pause frames transmitted with valid pause <code>op_code</code> .
<code>port_tx_control</code>	Number of control frames transmitted. Does not include pause frames.
<code>port_tx_unicast</code>	Number of unicast packets transmitted. Includes flow control packets.
<code>port_tx_multicast</code>	Number of multicast packets transmitted.
<code>port_tx_broadcast</code>	Number of broadcast packets transmitted.
<code>port_tx_lt64</code>	Number of frames transmitted where the length is less than 64 bytes.
<code>port_tx_64</code>	Number of frames transmitted where the length is exactly 64 bytes.
<code>port_tx_65_to_127</code>	Number of frames transmitted where the length is between 65 and 127 bytes
<code>port_tx_128_to_255</code>	Number of frames transmitted where the length is between 128 and 255 bytes
<code>port_tx_256_to_511</code>	Number of frames transmitted where the length is between 256 and 511 bytes
<code>port_tx_512_to_1023</code>	Number of frames transmitted where length is between 512 and 1023 bytes
<code>port_tx_1024_to_15xx</code>	Number of frames transmitted where the length is between 1024 and 1518 bytes (1522 with VLAN tag).
<code>port_tx_15xx_to_jumbo</code>	Number of frames transmitted where length is between 1518 bytes (1522 with VLAN tag) and 9000 bytes.
<code>port_rx_bytes</code>	Number of bytes received. Not include collided bytes.
<code>port_rx_good_bytes</code>	Number of bytes received without errors. Excludes bytes from flow control packets.
<code>port_rx_bad_bytes</code>	Number of bytes with invalid FCS. Includes bytes from packets that exceed the maximum frame length.

**Table 31: Ethtool -S output (continued)**

Field	Description
port_rx_packets	Number of packets received.
port_rx_good	Number of packets received with correct CRC value and no error codes.
port_rx_bad	Number of packets received with incorrect CRC value.
port_rx_pause	Number of pause frames received with valid pause op_code.
port_rx_control	Number of control frames received. Does not include pause frames.
port_rx_unicast	Number of unicast packets received.
port_rx_multicast	Number of multicast packets received.
port_rx_broadcast	Number of broadcasted packets received.
port_rx_lt64	Number of packets received where the length is less than 64 bytes.
port_rx_64	Number of packets received where the length is exactly 64 bytes.
port_rx_65_to_127	Number of packets received where the length is between 65 and 127 bytes.
port_rx_128_to_255	Number of packets received where the length is between 128 and 255 bytes.
port_rx_256_to_511	Number of packets received where the length is between 256 and 511 bytes.
port_rx_512_to_1023	Number of packets received where the length is between 512 and 1023 bytes.
port_rx_1024_to_15xx	Number of packets received where the length is between 1024 and 1518 bytes (1522 with VLAN tag).
port_rx_15xx_to_jumbo	Number of packets received where the length is between 1518 bytes (1522 with VLAN tag) and 9000 bytes.
port_rx_gtjumbo	Number of packets received with a length is greater than 9000 bytes.
port_rx_bad_gtjumbo	Number of packets received with a length greater than 9000 bytes, but with incorrect CRC value.
port_rx_overflow	Number of packets dropped by receiver because of FIFO overrun.

**Table 31: Ethtool -S output (continued)**

Field	Description
port_rx_nodesc_drop_cnt port_rx_nodesc_drops	<p>Number of packets dropped by the network adapter because of a lack of RX descriptors in the RX queue.</p> <p>Packets can be dropped by the NIC when there are insufficient RX descriptors in the RX queue to allocate to the packet. This problem occurs if the receive rate is very high and the network adapter receive cycle process has insufficient time between processing to refill the queue with new descriptors.</p> <p>A number of different steps can be tried to resolve this issue:</p> <ul style="list-style-type: none"> <li>• Disable the irqbalance daemon in the OS</li> <li>• Distribute the traffic load across the available CPU/cores by setting rss_cpus=cores. Refer to Receive Side Scaling section</li> <li>• Increase receive queue size using ethtool.</li> </ul>
port_rx_pm_trunc_bb_overflow	Overflow of the packet memory burst buffer - should not occur.
port_rx_pm_discard_bb_overflow	Number of packets discarded due to packet memory buffer overflow.
port_rx_pm_trunc_vfifo_full	<p>Number of packets truncated or discarded because there was not enough packet memory available to receive them. Happens when packets cannot be delivered as quickly as they arrive due to:</p> <ul style="list-style-type: none"> <li>• packet rate exceeds maximum supported by the adapter.</li> <li>• adapter is inserted into a low speed or low width PCI slot – so the PCIe bus cannot support the required bandwidth.</li> <li>• packets are being replicated by the adapter and the resulting bandwidth cannot be handled by the PCIe bus.</li> <li>• host memory bandwidth is being used by other devices resulting in poor performance for the adapter.</li> </ul>
port_rx_pm_discard_vfifo_full	Count of the number of packets dropped because of a lack of main packet memory on the adapter to receive the packet into.
port_rx_pm_trunc_qbb	Not currently supported.
port_rx_pm_discard_qbb	Not currently supported.
port_rx_pm_discard_mapping	Number of packets dropped because they have an 802.1p priority level configured to be dropped

**Table 31: Ethtool -S output (continued)**

Field	Description
port_rx_dp_q_disabled_packets	Increments when the filter indicates the packet should be delivered to a specific rx queue which is currently disabled due to configuration error or error condition.
port_rx_dp_di_dropped_packets	Number of packets dropped because the filters indicate the packet should be dropped. Can happen because: <ul style="list-style-type: none"> <li>the packet does not match any filter.</li> <li>the matched filter indicates the packet should be dropped.</li> </ul>
port_rx_dp_streaming_packets	Number of packets directed to RXDP streaming bus which is used if the packet matches a filter which directs it to the MCP. Not currently used.
port_rx_dp_hlb_fetch	Count the number of times the adapter descriptor cache is empty and a fetch operation is triggered to refill with more descriptors.
port_rx_dp_hlb_wait	Packet arrives while adapter descriptor cache is empty, refill is in progress, but not yet complete.
rx_unicast	Number of unicast packets received.
rx_unicast_bytes	Number of unicast bytes received.
rx_multicast	Number of multicast packets received.
rx_multicast_bytes	Number of multicast bytes received.
rx_broadcast	Number of broadcast packets received.
rx_broadcast_bytes	Number of broadcast bytes received.
rx_bad	Number of packets received with incorrect CRC value.
rx_bad_bytes	Number of bytes received from packets with incorrect CRC value.
rx_overflow	Number of packets dropped by receiver because of FIFO overrun.
tx_unicast	Number of unicast packets transmitted.
tx_unicast_bytes	Number of unicast bytes transmitted.
tx_multicast	Number of multicast packets transmitted.
tx_multicast_bytes	Number of multicast bytes transmitted.
tx_broadcast	Number of broadcast packets transmitted.
tx_broadcast_bytes	Number of broadcast bytes transmitted.
tx_bad	Number of bad packets transmitted.

**Table 31: Ethtool -S output (continued)**

Field	Description
tx_bad_bytes	Number of bad bytes transmitted.
tx_overflow	Number of packets dropped by transmitter because of FIFO overrun.
fec_uncorrected_errors	Number of uncorrected errors (RS-FEC)
fec_corrected_errors	Number of corrected errors (RS-FEC)
fec_corrected_symbols_lane0	per 25G lane corrected symbols
fec_corrected_symbols_lane1	
fec_corrected_symbols_lane2	
fec_corrected_symbols_lane3	
ctpio_vi_busy_fallback	When a CTPIO push occurs from a VI, but the VI DMA datapath is still busy with packets in flight or waiting to be sent. The packet is sent over the DMA datapath.
ctpio_long_write_success	Host wrote excess data beyond 32-byte boundary after frame end, but the CTPIO send was successful.
ctpio_missing_dbell_fail	When CTPIO push is not accompanied by a TX doorbell.
ctpio_overflow_fail	When the host pushes packet bytes too fast and overflows the CTPIO buffer.
ctpio_underflow_fail	When the host fails to push packet bytes fast enough to match the adapter port speed.  The packet is truncated and data transmitted as a poisoned packet.
ctpio_timeout_fail	When host fails to send all bytes to complete the packet to be sent by CTPIO before the VI inactivity timer expires.  The packet is truncated and data transmitted as a poisoned packet.
ctpio_noncontig_wr_fail	A non-sequential address (for packet data) is encountered during CTPIO, caused when packet data is sent over PCIe interface as out-of-order or with gaps.  Packet is truncated and transmitted as a poisoned packet.
ctpio_frm_clobber_fail	When a CTPIO push from one VI would have 'clobbered' a push already in progress by the same VI or another VI. One or both packets are sent over the DMA datapath - no packets are dropped.



**Table 31: Ethtool -S output (continued)**

Field	Description
ctpio_invalid_wr_fail	<p>If packet length is less than length advertised in the CTPIO header the CTPIO fails.</p> <p>Or packet write is not aligned to (or multiple of) 32-bytes,</p> <p>Packet maybe transmitted as a poisoned packet if sending has already started. Or erased if send has not already started.</p>
ctpio_vi_clobber_fallback	<p>When a ctpio collided with another already in progress. The in-progress packet succeeds, other packet is sent via DMA.</p>
ctpio_unqualified_fallback	<p>When the VI is not enabled to send using ctpio or first write is not the packet header.</p> <p>The packet is sent using DMA datapath.</p>
ctpio_runt_fallback	<p>Length in header &lt; 29 bytes.</p> <p>The packet is sent using DMA datapath.</p>
ctpio_success	<p>Number of successful ctpio tx events</p>
ctpio_fallback	<p>Number of instances when CTPIO push was rejected. This can occur because:</p> <ul style="list-style-type: none"> <li>• the VI legacy datapath is still busy</li> <li>• another CTPIO is in progress</li> <li>• VI is not enabled to use CTPIO</li> <li>• push request for illegal sized frame</li> </ul> <p>Fallback events do not result in poison packets. Rejected packets will use the DMA datapath path.</p>
ctpio_poison	<p>When the packet send has started, if CTPIO has to abort this packet, a corrupt CRC is attached to the packet.</p> <p>A poisoned packet may be sent over the wire - depending on the mode.</p> <p>The packet is sent using DMA datapath.</p>
ctpio_erase	<p>Before a packet send has started. Corrupt, undersized or poisoned packets are erased from the CTPIO datapath.</p> <p>The packet is sent using DMA datapath.</p>
tx_merge_events	<p>The number of TX completion events where more than one TX descriptor was completed.</p>
tx_tso_bursts	<p>Number of times when outgoing TCP data is split into packets by the adapter driver. Refer to <a href="#">TCP Segmentation Offload (TSO) on page 98</a>.</p>
tx_tso_long_headers	<p>Number of times TSO is applied to packets with long headers.</p>

**Table 31: Ethtool -S output (continued)**

Field	Description
tx_tso_packets	Number of physical packets produced by TSO.
tx_tso_fallbacks	0
tx_pushes	Number of times a packet descriptor is 'pushed' to the adapter from the network adapter driver.
tx_pio_packets	Number of packets sent using PIO.
tx_cb_packets	0
rx_reset	0
rx_tobe_disc	Number of packets marked by the adapter to be discarded because of one of the following: <ul style="list-style-type: none"> <li>• Mismatch unicast address and unicast promiscuous mode is not enabled.</li> <li>• Packet is a pause frame.</li> <li>• Packet has length discrepancy.</li> <li>• Due to internal FIFO overflow condition.</li> <li>• Length &lt; 60 bytes.</li> </ul>
rx_[inner outer]ip_hdr_chksum_err	Number of packets received with IP header Checksum error.
rx_[inner outer]tcp_udp_chksum_err	Number of packets received with TCP/UDP checksum error.
rx_eth_crc_err	Number of packets received where the CRC did not match the internally generated CRC value. This is the total of all receive channels receiving CRC errors.
rx_mcast_mismatch	Number of unsolicited multicast packets received. Unwanted multicast packets can be received because a connected switch simply broadcasts all packets to all endpoints or because the connected switch is not able or not configured for IGMP snooping - a process from which it learns which endpoints are interested in which multicast streams.
rx_frm_trunc	Number of frames truncated because an internal FIFO is full. As a packet is received it is fed by the MAC into a 128K FIFO. If for any reason the PCI interface cannot keep pace and is unable to empty the FIFO at a sufficient rate, the MAC will be unable to feed more of the packet to the FIFO. In this event the MAC will truncate the frame - marking it as such and discard the remainder. The driver on seeing a 'partial' packet which has been truncated will discard it.
rx_merge_events	Number of RX completion events where more than one RX descriptor was completed.
rx_merge_packets	Number of packets delivered to the host through merge events.

**Table 31: Ethtool -S output (continued)**

Field	Description
tx-N.tx_packets	Per TX queue transmitted packets.
rx_N.rx_packets	Per RX queue received packets.
rx_no_skb_drops	Number of packets dropped by the adapter when there are insufficient socket buffers available to receive packets into.  See also port_rx_nodesc_drop_cnt and port_rx_nodesc_drops above.
rx_nodesc_trunc	Number of frames truncated when there are insufficient descriptors to receive data into. Truncated packets will be discarded by the adapter driver.
ptp_good_syncs	These PTP stats counters relate to the mechanism used by sfptpd to synchronize the system clock and adapter clock(s) in a server.  For each synchronization event sfptpd will select a number of system clock times to be compared to the adapter clock time. If the times can be synchronized, the good_syncs counter is incremented, otherwise the bad_syncs counter is incremented. If sfptpd is unable to synchronize the clocks at this event, the sync_timeout counter is incremented.  sfptpd will synchronize clocks 16 times per second - so incrementing counters does not necessarily indicate bad synchronization between local server clocks and an external PTP master clock.
ptp_fast_syncs	
ptp_bad_syncs	
ptp_sync_timeouts	
ptp_no_time_syncs	
ptp_invalid_sync_windows	
ptp_undersize_sync_windows	
ptp_oversize_sync_windows	
ptp_rx_no_timestamp	Number of PTP packets received for which a hardware timestamp was not recovered from the adapter.
ptp_tx_timestamp_packets	Number of PTP packets transmitted for which the adapter generated a hardware timestamp.
ptp_rx_timestamp_packets	Number of PTP packets received for which the adapter generated a hardware timestamp.
ptp_timestamp_packets	Total number of PTP packets for which the adapter generated a hardware timestamp.
ptp_filter_matches	Number of PTP packets hitting the PTP filter.
ptp_non_filter_matches	Number of PTP packets which did not match the PTP filter.



**NOTE:** The adapter will double count packets less than 64 bytes (port\_rx\_1t64) as also being a CRC error. This can result in port\_rx\_bad => rx\_eth\_crc\_err counter. The difference should be equal to the port\_rx\_1t64 counter.

## 3.24 Reading sensors

Solarflare adapters have various sensors for temperature, voltage and currents. Their values are output using standard Linux conventions:

- The `sensors` command provides a formatted view of the values
- Alternatively, you can access the raw values via the filesystem.

### Using the sensors command

For a formatted view of the sensor values, use the `sensors` command from the `lm_sensors` package.

If the package is not already installed:

```
# yum install lm_sensors
```

By default, the `sensors` command shows sensors for all cores and other devices. By filtering its output you can get a list of all cores and devices in its output:

```
$ sensors | awk 'NF==1 {print}'
```

or a list of all adapters using the Solarflare `sfc` driver:

```
$ sensors | grep sfc
sfc-pci-0400
sfc-pci-0401
```

Then, to get output for a specific adapter such as `sfc-pci-0400`:

```
# sensors sfc-pci-0400
sfc-pci-0400
Adapter: PCI adapter
1.2V supply:                N/A
3.3V supply:                +3.22 V (min = +3.00 V, max = +3.60 V)
12.0V supply:              +12.14 V (min = +11.04 V, max = +12.96 V)
0.9V supply (ext. ADC):    +1.03 V (min = +0.50 V, max = +1.10 V)
                           (crit max = +1.15 V)
0.9V phase A supply:      N/A
PHY overcurrent:          N/A
ERROR: Can't get value of subfeature temp1_alarm: Can't read
PHY temp.:                N/A
AOE FPGA temp.:          +68.0°C (low = +0.0°C, high = +95.0°C)
                           (crit = +105.0°C)
Ambient temp.:            +56.0°C (low = +0.0°C, high = +75.0°C)
                           (crit = +85.0°C)
Controller die (TDIODE) temp.: +77.0°C (low = +0.0°C, high = +95.0°C)
                           (crit = +105.0°C)
Board front temp.:        +59.0°C (low = +0.0°C, high = +75.0°C)
                           (crit = +85.0°C)
Board back temp.:         +62.0°C (low = +0.0°C, high = +75.0°C)
                           (crit = +85.0°C)
1.2V supply current:      N/A
0.9V phase A supply current: N/A
3.3V supply current:      N/A
12V supply current:       N/A
```

## Using the filesystem

Sensors for Linux devices output their values into the `/sys/class/hwmon` hierarchy, in a directory named `/sys/class/hwmon/hwmon<n>/device`.

To determine which of the `hwmon<n>/device` directories contain sensor data from Solarflare adapters, search for a driver file within the directory that is a soft link to the `sfc` driver:

```
$ ls -l /sys/class/hwmon/*/device/driver | egrep sfc\$
lrwxrwxrwx 1 root root 0 Mar 20 10:43 /sys/class/hwmon/hwmon1/device/
driver -> ../../../../bus/pci/drivers/sfc
lrwxrwxrwx 1 root root 0 Mar 20 10:43 /sys/class/hwmon/hwmon2/device/
driver -> ../../../../bus/pci/drivers/sfc
```

So in the above example, the `/sys/class/hwmon/hwmon1/device` and `/sys/class/hwmon/hwmon2/device` directories contain sensor data from Solarflare adapters.

Within these directories, each sensor has the following files:

- `<sensor>_label` contains a description of the sensor
- `<sensor>_input` contains the current value of the sensor, in thousandths of the base unit (°C, V or A)
- `<sensor>_max` contains the maximum value for the sensor, in thousandths of the base unit (°C, V or A)
- `<sensor>_min` contains the minimum value for the sensor, in thousandths of the base unit (°C, V or A)
- `<sensor>_crit` contains the critical maximum value of the sensor, in thousandths of the base unit (°C, V or A)
- `<sensor>_alarm` contains 1 if an alarm condition exists for the sensor, else 0.

Read these files to get the data. For example:

```
$ cd /sys/class/hwmon/hwmon1/device
$ for file in temp4_*; do echo -en "${file}:\t" ; cat ${file}; done
temp4_alarm:      0
temp4_crit:       85000
temp4_input:      56000
temp4_label:      Ambient temp.
temp4_max:        75000
temp4_min:        0
```

In the above example, the ambient temperature is currently 56°C.

## 3.25 Driver Logging Levels

For the Solarflare net driver, two settings affect the verbosity of log messages appearing in dmesg output and /var/log/messages:

- The kernel console log level
- The netif message per network log level

The kernel console log level controls the overall log message verbosity and can be set with the command `dmesg -n` or through the `/proc/sys/kernel/printk` file:

```
echo 6 > /proc/sys/kernel/printk
```

Refer to 'man 2 syslog' for log levels and `Documentation/sysctl/kernel.txt` for a description of the values in `/proc/sys/kernel/printk`.

The netif message level provides additional logging control for a specified interface. These message levels are documented in `Documentation/networking/netif-msg.txt`. A message will only appear on the terminal console if both the kernel console log level and netif message level requirements are met.

The current netif message level can be viewed using the following command:

```
ethtool <iface> | grep -A 1 'message level:'
      Current message level: 0x000020f7 (8439)
      drv probe link ifdown ifup rx_err tx_err hw
```

Changes to the netif message level can be made with `ethtool`. Either by name:

```
ethtool -s <iface> msglvl rx_status on
```

or by bit mask:

```
ethtool -s <iface> msglvl 0x7fff
```

The initial setting of the netif msg level for all interfaces is configured using the debug module parameter e.g.

```
modprobe sfc debug=0x7fff
```

```
ethtool <iface> | grep -A 1 'message level:'
      Current message level: 0x00007fff (32767)
      drv probe link timer ifdown ifup rx_err
tx_err tx_queued intr tx_done rx_status pktdata hw wol
```

## 3.26 Running Adapter Diagnostics

You can use ethtool to run adapter diagnostic tests. Tests can be run offline (default) or online. Offline runs the full set of tests, which can interrupt normal operation during testing. Online performs a limited set of tests without affecting normal adapter operation.



**CAUTION:** Offline tests should not be run while sfptpd is running. The PTP daemon should be terminated before running the offline test.

As root user, enter the following command:

```
ethtool --test ethX offline|online
```

The tests run by the command are as follows:

**Table 32: Adapter Diagnostic Tests**

Diagnostic Test	Purpose
<b>core.nvram</b>	Verifies the flash memory 'board configuration' area by parsing and examining checksums.
<b>core.registers</b>	Verifies the adapter registers by attempting to modify the writable bits in a selection of registers.
<b>core.interrupt</b>	Examines the available hardware interrupts by forcing the controller to generate an interrupt and verifying that the interrupt has been processed by the network driver.
<b>tx/rx.loopback</b>	Verifies that the network driver is able to pass packets to and from the network adapter using the MAC and Phy loopback layers.
<b>core.memory</b>	Verifies SRAM memory by writing various data patterns (incrementing bytes, all bit on and off, alternating bits on and off) to each memory location, reading back the data and comparing it to the written value.
<b>core.mdio</b>	Verifies the MII registers by reading from PHY ID registers and checking the data is valid (not all zeros or all ones). Verifies the MMD response bits by checking each of the MMDs in the Phy is present and responding.
<b>chanX eventq.poll</b>	Verifies the adapter's event handling capabilities by posting a software event on each event queue created by the driver and checking it is delivered correctly.  The driver utilizes multiple event queues to spread the load over multiple CPU cores (RSS).
<b>phy.bist</b>	Examines the PHY by initializing it and causing any available built-in self tests to run.

## 3.27 Running Cable Diagnostics

Cable diagnostic data can be gathered from the Solarflare 10GBASE-T adapters physical interface using the `ethtool -t` command which runs a comprehensive set of diagnostic tests on the controller, PHY, and attached cables. To run the cable tests enter the following command:

```
ethtool -t ethX [online | offline]
```

Online tests are non-intrusive and will not disturb live traffic.



**CAUTION:** Offline tests should not be run while `sftptd` is running. The PTP daemon should be terminated before running the offline test.

The following is an extract from the output of the `ethtool` diagnostic offline tests:

```
phy  cable.pairA.length      9
phy  cable.pairB.length      9
phy  cable.pairC.length      9
phy  cable.pairD.length      9
phy  cable.pairA.status      1
phy  cable.pairB.status      1
phy  cable.pairC.status      1
phy  cable.pairD.status      1
```

Cable length is the estimated length in metres. A length value of 65535 indicates length not estimated due to pair busy or cable diagnostic routine not completed successfully.

The cable status can be one of the following values:

0 - invalid, or cable diagnostic routine did not complete successfully

1 - pair ok, no fault detected

2 - pair open or  $R_t > 115$  ohms

3 - intra pair short or  $R_t < 85$  ohms

4 - inter pair short or  $R_t < 85$  ohms

9 - pair busy or link partner forces 100Base-Tx or 1000Base-T test mode.



# 4

## Solarflare Adapters on Windows

This chapter documents procedures for the configuration and management of Solarflare adapters on **Windows Server 2012 R2**, **Windows Server 2016** and **Windows Server 2019**.

- [Windows 2012 R2 / 2016/ 2019 Driver on page 132](#)
- [Legacy Driver on page 132](#)
- [System Requirements on page 132](#)
- [Driver Certification on page 132](#)
- [Minimum Driver and Firmware Packages on page 133](#)
- [Firmware Variants on page 133](#)
- [Windows Feature Set on page 134](#)
- [Installing Solarflare Driver Package on page 135](#)
- [Install SolarflareTools on page 137](#)
- [Using SolarflareTools on page 138](#)
- [Configuration & Management on page 143](#)
- [Adapter Configuration on page 144](#)
- [Flow Control on page 148](#)
- [Configuring FEC on page 149](#)
- [Jumbo Frames on page 149](#)
- [Checksum Offload on page 150](#)
- [Interrupt Moderation \(Interrupt Coalescing\) on page 152](#)
- [NUMA Node on page 152](#)
- [Receive Side Scaling \(RSS\) on page 154](#)
- [Receive and Transmit Buffers on page 157](#)
- [Virtual Machine Queue on page 159](#)
- [Teaming and VLANs on page 160](#)
- [Adapter Statistics on page 163](#)
- [Performance Tuning on Windows on page 164](#)
- [List Installed Adapters on page 172](#)
- [Startup/Boot time Errors on page 172.](#)

## 4.1 Windows 2012 R2 / 2016 / 2019 Driver

The Solarflare adapter driver package includes Windows NDIS drivers for **Windows Server 2012 R2**, **Windows Server 2016** and **Windows Server 2019**.

- The driver is not supported on Windows Server versions earlier than 2012 R2.
- The driver is not supported on Windows Client OS versions, this includes Windows 10 clients.

The Solarflare driver currently supports the following Solarflare adapters:

- SFN8000 series adapters
- X2522 (10G) and X2522-25G adapters
- X2541 and X2542 adapters
- X2552 and X2562 adapters.

The driver is distributed as a single .zip package containing the .inf installer and uses standard Windows tools for installation, adapter configuration and management.

## 4.2 Legacy Driver

Earlier generations of Solarflare adapters; SFN5000, SFN6000, SFN7000 and SFN8000 series, are supported on Windows Server platforms, including Windows Server 2016, using the Solarflare legacy Bus/NDIS based driver.

A previous issue of this user manual (Issue 22 or earlier), documenting the configuration/management of Solarflare adapters using the Bus/NDIS driver, is available from [support@solarflare.com](mailto:support@solarflare.com).



**NOTE:** The legacy driver is no longer under active development for new features, but continues to be maintained for security updates and bug fixes.

## 4.3 System Requirements

Refer to [Software Driver Support on page 16](#) for details of supported Windows versions.

## 4.4 Driver Certification

The Solarflare Windows driver and Solarflare SFN8000 series and X2 series adapters are certified compatible with the Windows Hardware Compatibility Program (WHCP).

Drivers are available via Windows update and from <https://support.solarflare.com/>

## 4.5 Minimum Driver and Firmware Packages

- SF-121281-LS issue 1 containing driver 1.0.0.1012
  - For X2522 models
- SF-121281-LS issue 2 containing driver 1.2.1.1004
  - For X2522 models and X254x models
- SF-121281-LS issue 3 containing driver 1.4.1.1000
  - For X2522 models and X254x models
- SF-121281-LS issue 7 containing driver 1.8.0.1013
  - For X2552 models and X2562 models
- Firmware: 7.4.0.1021

## 4.6 Firmware Variants

The Solarflare adapter firmware can be configured into different variants of the adapter firmware. This is configurable using the *SfConfig* utility

Firmware variant	Limitations
Full-feature	None.
Ultra-low-latency	Overlay acceleration, VLAN strip/insert, SRIOV are disabled.
Auto	Selects Full-feature.

## 4.7 Windows Feature Set

The following table lists the features supported by Solarflare adapters on Windows.

**Table 33: Solarflare Windows Features**

<b>Flow Control</b>	Refer to <a href="#">Flow Control on page 148</a>
<b>Jumbo frames</b>	MTUs (Maximum Transmission Units) from 1500 bytes to 9216 bytes. <ul style="list-style-type: none"> <li>Refer to <a href="#">Jumbo Frames on page 149</a></li> </ul>
<b>Task offloads</b>	Large Segmentation Offload (LSO) and TCP/UDP/IP checksum offload for improved adapter performance and reduced CPU processing requirements. <ul style="list-style-type: none"> <li><a href="#">Checksum Offload on page 150</a> and <a href="#">Large Send Offload (LSO) on page 151</a></li> </ul>
<b>Receive Side Scaling (RSS)</b>	RSS multi-core load distribution technology. <ul style="list-style-type: none"> <li><a href="#">Receive Side Scaling (RSS) on page 154</a></li> </ul>
<b>Receive Segment Coalescing</b>	RSC coalesce multiple received TCP packets. <ul style="list-style-type: none"> <li><a href="#">Receive Segment Coalescing (RSC) on page 151</a></li> </ul>
<b>Interrupt Moderation</b>	Interrupt Moderation to reduce the number of interrupts on the host processor from packet events. <ul style="list-style-type: none"> <li><a href="#">Interrupt Moderation (Interrupt Coalescing) on page 152</a></li> </ul>
<b>Teaming and/or Link Aggregation</b>	Improve server reliability and bandwidth by bonding physical ports, from one or more Solarflare adapters, into a team, having a single MAC address and which function as a single port providing redundancy against a single point of failure. <ul style="list-style-type: none"> <li><a href="#">Teaming and VLANs on page 160</a></li> </ul>
<b>Virtual LANs (VLANs)</b>	Support for multiple VLANs per adapter: <ul style="list-style-type: none"> <li><a href="#">Teaming and VLANs on page 160</a></li> </ul>
<b>State and statistics analysis</b>	<ul style="list-style-type: none"> <li><a href="#">Adapter Statistics on page 163</a></li> </ul>
<b>FEC</b>	Support for Forward Error Correction: <ul style="list-style-type: none"> <li><a href="#">Configuring FEC on page 149</a></li> </ul>
<b>VMQ</b>	Support for Virtual Machine Queues: <ul style="list-style-type: none"> <li><a href="#">Virtual Machine Queue on page 159</a></li> </ul>

## 4.8 Installing Solarflare Driver Package

The adapter drivers and SolarflareTools are supplied as a single .zip package **SF-121281-LS** available from: support@solarflare.com.

A device driver package must be added (staged) to the Windows driver store before it can be installed on any device. Use the pnputil utility from command prompt or Powershell prompt to add (stage) or remove the Solarflare driver package.

### Identify installed driver

```
PS> pnputil /enum-drivers
```

```
Published Name:    oem5.inf
Original Name:     sfn.inf
Provider Name:     Solarflare
Class Name:        Network adapters
Class GUID:        {4d36e972-e325-11ce-bfc1-08002be10318}
Driver Version:    10/05/2018 1.0.0.1012
Signer Name:       Microsoft Windows Hardware Compatibility Publisher
```

This will enumerate/list all third-party drivers in the driver store. This command does not list drivers packaged as part of the Windows OS distribution.

### Remove and uninstall driver

```
PS> pnputil /delete-driver oem#.inf [/force]
```

# is the enumerated driver .inf file as identified from the /enum-drivers option.

### Add and install driver

- 1 Copy the .zip file driver package to a directory on the Windows Server - **in this install example we create and use the directory C:\sfdriver**

- 2 Set location to the directory:

```
PS> Set-Location C:\sfdriver
```

- 3 Expand archive:

```
PS C:\sfdriver> Expand-Archive "SF-121281-LS-Solarflare XtremeScale X2 Series Windows Driver Package.zip" -DestinationPath .
```

The following should be present in the directory:

```
license.txt
readme.txt
WS2012R2/
WS2016/
WS2019/
```

Each OS-specific folder contains the following files:

```
sfn.cat
sfn.inf
sfn.man
sfn.sys
sfn.wprp
```

**4** Install Driver:

```
PS C:\sfdriver> pnputil /add-driver sfn.inf /install /subdirs
```

Microsoft PnP Utility

```
Adding driver package: sfn.inf
Driver package added successfully.
Published Name: oem5.inf
Driver package installed on matching devices.
Total driver packages: 1
Added driver packages: 1
```

This command will add the driver to the driver store and, using the /install directive means the driver will also be installed on adapters.

**5** For Windows Server 2012 R2 and Windows Server 2016, the manifest for management event logging needs to be installed manually. For example, for Windows Server 2016:

```
PS C:\sfdriver> wevtutil.exe im WS2016\sfn.man /
rf:"${Env:SystemRoot}\system32\drivers\sfn.sys" /
mf:"${Env:SystemRoot}\system32\drivers\sfn.sys"
```

**6** Identify installed driver version:

```
PS C:\sfdriver> Get-NetAdapter |? DriverProvider -eq Solarflare |
Format-Table -View Driver
```

Name	InterfaceDescription	DriverFileName
Ethernet 4	Solarflare XtremeScale X2522 (10G) ...#2	SFN.sys
DriverDate	DriverVersion	NdisVersion
2018-10-05	1.0.0.1012	6.60

## 4.9 Install SolarflareTools

The SolarflareTools are powershell cmdlets utilities to configure and manage the Solarflare adapter. Utility tools are available for SFN8000 series and X2 series adapters, and include:

- *SfUpdate* – firmware management
- *SfConfig* – adapter configuration
- *SfReport* – server/adapter diagnostics script.

SolarflareTools are distributed as a PowerShellGet package available from [support.solarflare.com](http://support.solarflare.com).

### Requirements

Solarflare XtremeScale X2 tools require Windows PowerShell 5.1 which is installed by default on Windows Server 2016 and later. PowerShell 5.1 is available as an optional update for Windows Server 2012 R2.

### Install

Refer to the Solarflare XtremeScale X2 Series Tools Package **SF-123005-LS** for install instructions.

### Uninstall

The SolarflareTools bundle can be uninstalled using the following command:

```
PS> Uninstall-Module -Name SolarflareTools [-RequiredVersion <version>]
```

or to remove from current powershell session:

```
PS> Remove-Module -Name SolarflareTools
```



**NOTE:** It may be useful to retain multiple SolarflareTools versions on the local server. In the event of a firmware issue, a previous firmware image can be restored to the adapter.

### Identify the installed SolarflareTools package

```
PS> Get-Module -Name SolarflareTools -ListAvailable
```

```
Directory: C:\Program Files\WindowsPowerShell\Modules
ModuleType Version Name ExportedCommands
Script 1.5.4 SolarflareTools
```

```
ExportedCommands
{Get-SfNetAdapterDiagnosticReport, Invoke-SfNetAdapterUpdate, Get-SfNetAdapterFirmwareInformation, Update-SfNetAdapterFirmware...}
```

## List Available Commands

```
PS> Get-Command -Module SolarflareTools -CommandType All
```

CommandType	Name	Version
Source		
Alias	SfConfig -> Invoke-SfNetAdapterConfiguration	1.5.4
SolarflareTools		
Alias	SfReport -> Get-SfNetAdapterDiagnosticReport	1.5.4
SolarflareTools		
Alias	SfUpdate -> Invoke-SfNetAdapterUpdate	1.5.4
SolarflareTools		
Function	Get-SfNetAdapterDiagnosticReport	1.5.4
SolarflareTools		
Function	Get-SfNetAdapterFirmwareInformation	1.5.4
SolarflareTools		
Function	Get-SfNetAdapterGlobalConfiguration	1.5.4
SolarflareTools		
Function	Invoke-SfNetAdapterConfiguration	1.5.4
SolarflareTools		
Function	Invoke-SfNetAdapterUpdate	1.5.4
SolarflareTools		
Function	Set-SfNetAdapterGlobalConfiguration	1.5.4
SolarflareTools		
Function	Update-SfNetAdapterFirmware	1.5.4
SolarflareTools		

## 4.10 Using SolarflareTools

### SfUpdate - Firmware management

The *SfUpdate* utility is used to manage firmware on the Solarflare adapter.

The firmware bundle location can be identified as the directory listed with the following command:

```
PS> Get-Module SolarflareTools -ListAvailable
```



**NOTE:** There is no requirement to unzip the firmware.zip file.

Firmware can be installed on the adapter from the following sources:

- the firmware bundle installed on the local server (see above path)
- a firmware image copied to the local server



## Identify current firmware version

PS> sfupdate

Solarflare firmware update utility  
(c) 2018-2019 Solarflare Communications Inc. All rights reserved.

Solarflare XtremeScale X2522-25G Adapter:  
Boot ROM: v5.2.1.1000 - update to v5.2.2.1004  
Adapter: v7.6.1.1004 - update to v7.6.2.1006  
UEFI ROM: v2.8.6.5 - downgrade to v2.7.8.5  
SUC: v2.1.1.1004 - update to v2.1.1.1012

If the firmware bundle version supplied with the driver package is newer than the firmware components installed on the adapter, the following options are displayed.

Firmware update available  
There is an upgrade to Solarflare XtremeScale X2522-25G Adapter:Boot ROM.  
Do you want to install?

[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help  
(default is "Y"):

Y - Continue with only the next step of the operation.  
A - Continue with all the steps of the operation.  
N - Skip this operation and proceed with the next operation.  
L - Skip this operation and all subsequent operations.  
S - Pause the current pipeline and return to the command prompt. Type "exit" to resume the pipeline.

Select an option to upgrade adapter firmware or use the following commands for more options:

PS> Get-Help SfUpdate -[Detailed|Examples|Full]

## SfConfig - Adapter Configuration

The *SfConfig* utility is used to configure the adapter hardware port mode or firmware variant. *SfConfig* can also be used to restore adapter default settings.

Identify current configuration:

PS> SfConfig

Solarflare configuration utility  
(c) 2020 Xilinx, Inc. All rights reserved. (c) 2018-2019 Solarflare Communications Inc.

InterfaceDescription : Solarflare XtremeScale X2542 Adapter  
PortMode :  
FirmwareVariant :  
VlanTags :  
PfCount : 1  
MsixLimit : {Default}  
Partitioning :

```
InterfaceDescription : Solarflare XtremeScale X2542 Adapter #2
PortMode             :
FirmwareVariant     : Auto
VlanTags            :
PfCount             : 1
MsixLimit           : {Default}
Partitioning        :
```

The global configuration settings (PortMode, FirmwareVariant and Partitioning) are only configurable on the first adapter of the first port of the NIC. The other settings are per-port and are configurable on the first adapter of each port of the NIC.

Use the following commands for more options:

```
PS> Get-Help SfConfig -[Detailed|Examples|Full]
```

### Port Mode

```
PS> SfConfig -Name "<interface>" -PortMode [1x10G/25G][1x10G/25G]
```

The port mode string for X2 series adapters can be one of the following values:

- Default
- [4x10/25G]
- [2x10/25G][2x10/25G]
- [2x50G]
- [1x50G][1x50G]
- [1x100G]

The port mode string for 8000-series adapters can be one of the following values:

- Default
- [1x40G][1x40G]
- [4x10G]
- [2x10G][2x10G]

### Firmware Variant

Refer to [Firmware Variants on page 133](#).

```
PS> sfconfig -ifDesc "Solarflare XtremeScale X2542 Adapter #2" -FirmwareVariant "Full feature"
```

Solarflare configuration utility

(c) 2020 Xilinx, Inc. All rights reserved. (c) 2018-2019 Solarflare Communications Inc.

```
InterfaceDescription : Solarflare XtremeScale X2542 Adapter #2
PortMode             :
FirmwareVariant     : Full feature
VlanTags            :
PfCount             : 1
MsixLimit           : {Default}
Partitioning        :
```

## Partitioning

Enable the Partitioning setting to apply a global configuration that is compatible with port configuration for NIC Partitioning (see [NIC Partitioning on page 62](#) for details). This includes setting the Firmware Variant to Full feature.

```
PS> sfconfig -Partitioning Enabled
```

```
Solarflare configuration utility
(c) 2020 Xilinx, Inc. All rights reserved. (c) 2018-2019 Solarflare Communications Inc.
WARNING: Adapter Solarflare XtremeScale X2542 Adapter #2 is being rebooted.
Waiting for restart on: Solarflare XtremeScale X2542 Adapter #2 ...
```

```
InterfaceDescription : Solarflare XtremeScale X2542 Adapter
PortMode             :
FirmwareVariant      :
VlanTags             :
PfCount              : 1
MsixLimit            : {Default}
Partitioning         :
```

```
InterfaceDescription : Solarflare XtremeScale X2542 Adapter #2
PortMode             :
FirmwareVariant      : Full feature
VlanTags             :
PfCount              : 1
MsixLimit            : {Default}
Partitioning         : Enabled
```

A port is partitioned by configuring PfCount to greater than 1. To partition with VLAN support, specify a list of distinct VLAN tag values corresponding to each PF:

```
PS> sfconfig -ifdesc "Solarflare XtremeScale X2542 Adapter" -PfCount 2 -VlanTags
@(101,102)
```

```
Solarflare configuration utility
(c) 2020 Xilinx, Inc. All rights reserved. (c) 2018-2019 Solarflare Communications Inc.
WARNING: Computer must be powered off for changes to take effect
```

```
InterfaceDescription : Solarflare XtremeScale X2542 Adapter
PortMode             :
FirmwareVariant      :
VlanTags             : {101, 102}
PfCount              : 2
MsixLimit            : {Default, Default}
Partitioning         :
```

Optionally, the maximum number of MSI-X interrupts that each PF will use can be specified.



**NOTE:** Using incorrect settings can impact the performance of the adapter. Contact Solarflare technical support before changing this setting.

## More Help

To list all options – including port-mode settings per adapter model – use the following command:

```
PS> Get-Help SfConfig -Detailed
```

## Restore Adapter Default Settings

To restore an adapter to default settings (does not change firmware) - use the following command:

```
PS> SfConfig -Name "<interface>" -Clear
```

## SfReport - Diagnostic script

*SfReport* will generate a diagnostic file detailing configuration aspects of the server and Solarflare adapters.

If problems are encountered, the SfReport can be run and a report generated **before any server reboot occurs**.

The generated (HTML) file should be returned to support@solarflare.com.

```
PS> SfReport
```

Solarflare report utility

(c) 2018-2019 Solarflare Communications Inc. All rights reserved.

Report saved as SfReport-localhost-20181005035359.html

The output (HTML) file will be generated in the local directory.

To view the report directly in web browser:

```
PS> SfReport | Invoke-Item
```



**NOTE:** Users may consider reading the SfReport output file to redact sensitive information before sending the file to Solarflare support.

## 4.11 Configuration & Management

Use Windows Powershell cmdlets or the Windows Device Manager GUI interface. Refer to Microsoft documentation for information on standard adapter cmdlets:

<https://docs.microsoft.com/powershell/module/netadapter>

### Powershell Cmdlets

For description and usage help on any cmdlet:

`Get-Help <cmdlet-name>`

For detailed description/usage information

`Get-Help <cmdlet-name> [-Detailed|-Examples|-Full]`

### GUI

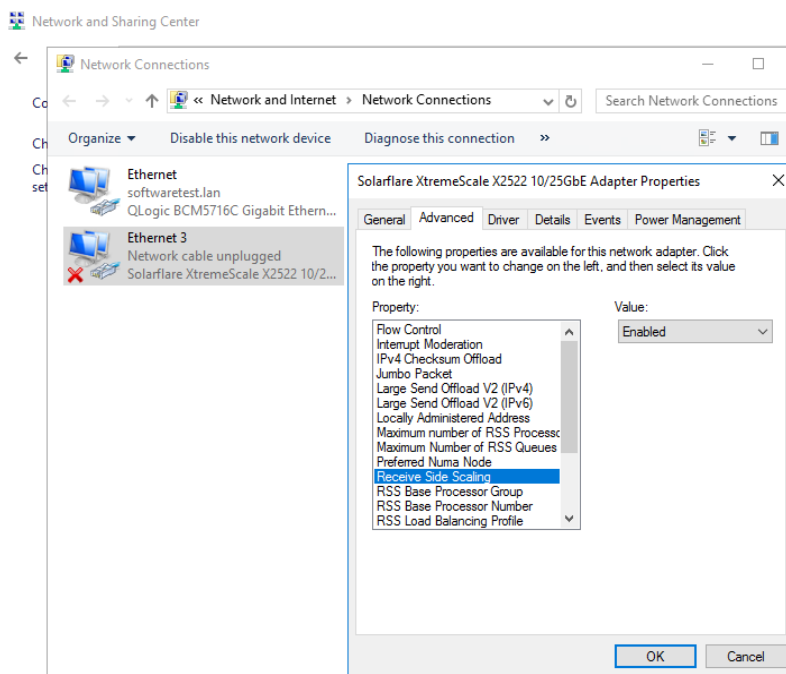
Launch the Network Connections window from command line or from Device Manager window:

`PS > ncpa.cpl`

or from control panel:

**Control Panel > Network and Internet > Network and Sharing Center > Change adapter settings**

Select an adapter (right click) to open the Properties window and use the Configure button to open the Advanced Properties window.



## 4.12 Adapter Configuration

Identify installed adapters

```
PS > Get-NetAdapter
```

Name	InterfaceDescription	ifIndex	Status
Ethernet 4	Solarflare XtremeScale X2522 10/25...#2	21	Up

MacAddress	LinkSpeed
00-0F-53-64-4F-11	10 Gbps

### Adapter PCIe Information

```
PS > Get-NetAdapterHardwareInfo -Name "<interface>"
```

Name	Segment	Bus	Device	Function	Slot	NumaNode	PcieLinkSpeed	Width
Eth 4	0	1	0	1	1		8.0 GT/s	8

### Adapter hardware properties

For an extensive list of adapter, adapter port and server hardware properties for the specified Solarflare interface.

```
PS > Get-NetAdapter -Name "<interface>" | Format-List -Property *
MacAddress           : 00-0F-53-65-1A-61
Status               : Up
LinkSpeed            : 25 Gbps
MediaType            : 802.3
PhysicalMediaType    : 802.3
AdminStatus          : Up
MediaConnectionState : Connected
DriverInformation    : Driver Date 2019-02-07
Version 1.2.1.1004 NDIS 6.60
DriverFileName       : SFN.sys
NdisVersion           : 6.60
ifOperStatus         : Up
ifAlias              : SLOT 1 Port 2
InterfaceAlias        : SLOT 1 Port 2
ifIndex              : 16
ifDesc               : Solarflare XtremeScale
X2522-25G Adapter #2
ifName                : ethernet_32772
DriverVersion         : 1.2.1.1004
LinkLayerAddress     : 00-0F-53-65-1A-61
Caption              :
Description           :
ElementName          :
InstanceID           : {F80F01D6-639D-4E1A-8C69-7F511E1234B5}
CommunicationStatus  :
DetailedStatus       :
HealthState          :
InstallDate          :
```

```

Name : SLOT 1 Port 2
OperatingStatus :
OperationalStatus :
PrimaryStatus :
StatusDescriptions :
AvailableRequestedStates :
EnabledDefault : 2
EnabledState : 5
OtherEnabledState :
RequestedState : 12
TimeOfLastStateChange :
TransitioningToState : 12
AdditionalAvailability :
Availability :
CreationClassName : MSFT_NetAdapter
DeviceID : {F80F01D6-639D-4E1A-
8C69-7F511E1234B5}
ErrorCleared :
ErrorDescription :
IdentifyingDescriptions :
LastErrorCode :
MaxQuiesceTime :
OtherIdentifyingInfo :
PowerManagementCapabilities :
PowerManagementSupported :
PowerOnHours :
StatusInfo :
SystemCreationClassName : CIM_NetworkPort
SystemName : ADMINISO-
B6S85GK.softwaretest.lan
TotalPowerOnHours :
MaxSpeed :
OtherPortType :
PortType :
RequestedSpeed :
Speed : 25000000000
UsageRestriction :
ActiveMaximumTransmissionUnit : 1500
AutoSense :
FullDuplex : True
LinkTechnology :
NetworkAddresses : {000F53651A61}
OtherLinkTechnology :
OtherNetworkPortType :
PermanentAddress : 000F53651A61
PortNumber : 0
SupportedMaximumTransmissionUnit :
AdminLocked : False
ComponentID :
PCI\VEN_1924&DEV_0B03&SUBSYS_80281924
ConnectorPresent : True
DeviceName : \Device\{F80F01D6-639D-
4E1A-8C69-7F511E1234B5}
DeviceWakeUpEnable : False
DriverDate : 2019-02-07
DriverDateData : 131939712000000000
DriverDescription : Solarflare XtremeScale

```

```

X2522-25G Adapter
DriverMajorNdisVersion           : 6
DriverMinorNdisVersion           : 60
DriverName                        :
\SystemRoot\System32\drivers\SFN.sys
DriverProvider                    : Solarflare
DriverVersionString               : 1.2.1.1004
EndPointInterface                 : False
HardwareInterface                 : True
Hidden                            : False
HigherLayerInterfaceIndices       : {17}
IMFilter                          : False
InterfaceAdminStatus              : 1
InterfaceDescription              : Solarflare XtremeScale
X2522-25G Adapter #2
InterfaceGuid                     : {F80F01D6-639D-4E1A-
8C69-7F511E1234B5}
InterfaceIndex                    : 16
InterfaceName                     : ethernet_32772
InterfaceOperationalStatus        : 1
InterfaceType                     : 6
iSCSIInterface                   : False
LowerLayerInterfaceIndices        :
MajorDriverVersion                : 1
MediaConnectState                 : 1
MediaDuplexState                  : 2
MinorDriverVersion                : 2
MtuSize                           : 1500
NdisMedium                        : 0
NdisPhysicalMedium                : 14
NetLuid                           : 1689399683186688
NetLuidIndex                      : 32772
NotUserRemovable                  : False
OperationalStatusDownDefaultPortNotAuthenticated : False
OperationalStatusDownInterfacePaused : False
OperationalStatusDownLowPowerState : False
OperationalStatusDownMediaDisconnected : False
PnPDeviceID                       :
PCI\VEN_1924&DEV_0B03&SUBSYS_80281924&REV_01\000F53FFFF651A6001
PromiscuousMode                   : False
ReceiveLinkSpeed                  : 25000000000
State                              : 2
TransmitLinkSpeed                  : 25000000000
Virtual                            : False
VlanID                            :
WdmInterface                       : False
PSComputerName                    :
CimClass                           : ROOT/
StandardCimv2:MSFT_NetAdapter
CimInstanceProperties              : {Caption, Description,
ElementName, InstanceID...}
CimSystemProperties                :
Microsoft.Management.Infrastructure.CimSystemProperties

```



## Adapter advanced (configuration) properties

For adapter configurable properties:

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>" [-AllProperties]
```

DisplayName	DisplayValue	RegKeyword	RegValue
Speed & Duplex	Auto Negotiation	*SpeedDuplex	{0}
Forward Error Correction	Auto	Fec	{0}
Jumbo Packet	1518	*JumboPacket	{1518}
Flow Control	Auto Negotiation	*FlowControl	{4}
Interrupt Moderation	Enabled	*InterruptMo...	{1}
Interrupt Moderation Time	60	InterruptMod...	{60}
IPv4 Checksum Offload	Rx & Tx Enabled	*IPChecksumO...	{3}
TCP Checksum Offload (IPv4)	Rx & Tx Enabled	*TCPChecksum...	{3}
UDP Checksum Offload (IPv4)	Rx & Tx Enabled	*UDPChecksum...	{3}
TCP Checksum Offload (IPv6)	Rx & Tx Enabled	*TCPChecksum...	{3}
UDP Checksum Offload (IPv6)	Rx & Tx Enabled	*UDPChecksum...	{3}
Large Send Offload V2 (IPv4)	Enabled	*LSOv2IPv4	{1}
Large Send Offload V2 (IPv6)	Enabled	*LSOv2IPv6	{1}
Recv Segment Coalescing (IPv4)	Enabled	*RSCIPv4	{1}
Recv Segment Coalescing (IPv6)	Enabled	*RSCIPv6	{1}
Receive Side Scaling	Enabled	*Rss	{1}
Maximum number of RSS Proce...	16	*MaxRssProce...	{16}
Maximum Number of RSS Queues	8	*NumRSSQueues	{8}
RSS Load Balancing Profile	NUMA Scaling Static	*RssProfile	{4}
RSS Base Processor Group	0	*RssBaseProc...	{0}
RSS Base Processor Number	0	*RssBaseProc...	{0}
RSS Max Processor Group	9	*RssMaxProcG...	{9}
RSS Max Processor Number	63	*RssMaxProcN...	{63}
Preferred Numa Node	All	*NumaNodeId	{65535}
Receive Buffers	1152	*ReceiveBuffers	{1152}
Transmit Buffers	1024	*TransmitBuf...	{1024}

## 4.13 Flow Control

Ethernet flow control allows two communicating devices to inform each other when they are being overloaded by received data. This prevents one device from overwhelming the other device with network packets. For instance, when a switch is unable to keep up with forwarding packets between ports. Solarflare adapters can auto-negotiate flow control settings with the link partner.

Flow control can be configured using adapter advanced properties.

### Identify current settings

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>"
```

DisplayName	DisplayValue	RegKeyword	RegValue
Flow Control	Auto Negotiation	*FlowControl	{4}

### Change settings

```
PS > Set-NetAdapterAdvancedProperty -Name "<interface>" -Displayname "Flow Control" -Displayvalue "Rx Enabled"
```

Option	Description
<b>Auto-negotiation</b>	Flow control is auto-negotiated between the devices. This is the default setting, preferring <b>Rx &amp; Tx Enabled</b> if the link partner is capable.
<b>Rx &amp; Tx Enabled</b>	Adapter generates and responds to flow control messages.
<b>Tx Enabled</b>	Adapter responds to flow control messages but is unable to generate messages if it becomes overwhelmed.
<b>Rx Enabled</b>	Adapter generates flow control messages but is unable to respond to incoming messages and will keep sending data to the link partner.
<b>Disabled</b>	Ethernet flow control is disabled on the adapter. Data will continue to flow even if the adapter or link partner is overwhelmed.

## 4.14 Configuring FEC

For information about FEC, see [Forward Error Correction on page 42](#).

FEC can be configured using adapter advanced properties.

### Identify current settings

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>"
```

### Change settings

```
PS > Set-NetAdapterAdvancedProperty -Name <interface> -DisplayName  
"Forward Error Correction" -DisplayValue <value>
```

where <value> is one of the following:

Auto | Disabled | Prefer FEC | BASE-R/FC FEC | RS FEC

## 4.15 Jumbo Frames

Solarflare adapters support Jumbo frames up to 9216 bytes. Jumbo frames can be configured using adapter advanced properties.

### Identify current settings

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>"
```

DisplayName	DisplayValue	RegKeyword	RegValue
Jumbo Packet	1518	*JumboPacket	{1518}

### Change settings

```
PS > Set-NetAdapterAdvancedProperty -Name "<interface>" -DisplayName  
"Jumbo Packet" -DisplayValue "9216"
```

## 4.16 Checksum Offload

Checksum offloading is supported for IP, TCP and UDP packets. Before transmitting a packet, a checksum is generated by the adapter and appended to the packet. At the receiving end, the calculation is performed by the adapter against the received packet. Offloading checksum calculation to the network adapter decreases the work load on server CPUs.

### Identify current settings

```
PS > Get-NetAdapterChecksumOffload -Name "<interface>"
```

Name	IpIPv4Enabled	TcpIPv4Enabled	TcpIPv6Enabled
Ethernet 4	RxTxEnabled	RxTxEnabled	RxTxEnabled
UdpIPv4Enabled	UdpIPv6Enabled		
RxTxEnabled	RxTxEnabled		

Checksums can be enabled/disabled in both transmit and receive directions:

TxEnabled, RxEnabled, RxTxEnabled, Disabled

### Set values

```
PS > Set-NetAdapterChecksumOffload -Name "<interface>" -IpIPv4Enabled  
RxTxEnabled
```

### Disable All

```
PS > Disable-NetAdapterChecksumOffload -Name "<interface>"
```

Returns all checksums to Disabled.

### Enable All

```
PS > Enable-NetAdapterChecksumOffload -Name "<interface>"
```

Enabling returns all checksum to RxTxEnabled.



**NOTE:** Changing the Checksum Offload settings can impact the performance of the adapter. Solarflare recommend that these remain at the default values. Disabling Checksum Offload disables Large Send Offload.

## Large Send Offload (LSO)

LSO offloads to the adapter the task of splitting large outgoing TCP data into smaller packets. This improves throughput performance and has no effect on latency.

### Identify current settings

```
PS > Get-NetAdapterLso -Name "<interface>"
```

Name	Version	V1IPv4Enabled	IPv4Enabled	IPv6Enabled
Eth 4	LSO Version 2	False	True	True

### Enable LSO

```
PS > Enable-NetAdapterLso -Name "<interface>"
```

### Disable LSO

```
PS > Disable-NetAdapterLso -Name "<interface>"
```



**NOTE:** LSO is enabled by default and there is generally no reason to disable this feature.

## Receive Segment Coalescing (RSC)

When RSC is enabled the adapter will coalesce multiple received TCP packets on a TCP connection into a single call to the TCP/IP stack. This reduces CPU use and improves peak performance. RSC has minimal impact on latency. If a host is forwarding received packets from one interface to another, Windows will automatically disable RSC. RSC is enabled by default.

### Identify current settings

```
PS > Get-NetAdapterRsc -Name "<interface>"
```

### Enable RSC

```
PS > Enable-NetAdapterRsc -Name "<interface>"
```

### Disable RSC

```
PS > Disable-NetAdapterRsc -Name "<interface>"
```

## 4.17 Interrupt Moderation (Interrupt Coalescing)

Reduces the number of interrupts generated by the adapter by combining multiple received packet events and/or transmit completion events into a single interrupt thereby reducing the number of interrupts sent to the CPU and reducing the CPU workload.

### Identify current settings

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>"
```

Name	DisplayName	DisplayValue	RegistryKeyword	RegistryValue
Eth 4	Interrupt Moderation	Enabled	*InterruptMo...	{1}

### Enable

```
PS > Set-NetAdapterAdvancedProperty -Name "<interface>" -Displayname "Interrupt Moderation Packet" -Displayvalue "Enabled"
```

### Disable

```
PS > Set-NetAdapterAdvancedProperty -Name "<interface>" -Displayname "Interrupt Moderation Packet" -Displayvalue "Disabled"
```

## 4.18 NUMA Node

The adapter driver can select a subset of available CPU cores to handle transmit and receive processing. The preferred NUMA node setting can be used to constrain the set of CPU cores used to those on a specific NUMA Node.

To force processing onto a particular NUMA Node, set the preferred NUMA node value in the adapter advanced properties.

The NUMA distance of the cores used for the RSS queue and for the network application will influence performance.

### NUMA Distance

To check the NUMA distance of each core from the interface.

```
PS > Get-NetAdapterRss -Name "<interface>"
```

Name	:	Ethernet 4
InterfaceDescription	:	Solarflare XtremeScale X2522 10/25GbE Adapter #2
Enabled	:	True
NumberOfReceiveQueues	:	8
Profile	:	NUMAStatic
BaseProcessor: [Group:Number]	:	0:0
MaxProcessor: [Group:Number]	:	0:3
MaxProcessors	:	4
RssProcessorArray: [Group:Number/NUMA Distance]	:	0:0/0 0:1/0 0:2/0 0:3/0
IndirectionTable: [Group:Number]	:	

### **RSS Queue - NUMA Node**

For low latency low jitter applications, RSS queues should be mapped to NUMA nodes that are local to the interface. The local NUMA node is always selected automatically when either of the following RSS profiles is selected:

- ClosestProcessor
- ClosestProcessorStatic

### **Application - NUMA Node**

For low latency low jitter, run the network applications on a NUMA node local to the interface.

```
> start /affinity <hexmask> <command>
```

or

```
> start /node <num> <command>
```

or

```
> start /node <num> /affinity <hexmask> <command>
```

### **Preferred NUMA Node**

Assign the adapter to a specific NUMA node where the registry value is a number between 0-15 or use 65535 to use ALL.

```
PS > Set-NetAdapterAdvancedProperty -Name "<interface>" -Registrykeyword '*numanodeid' -Registryvalue '0'
```

### **RSS - NICs sharing a NUMA node**

When adapters share a NUMA node, RSS, for each adapter, can be limited to a subset of processors within the node.

```
PS > Set-NetAdapterRss -Name "<interface>" -NumaNode 0 -BaseProcessorNumber 1 -MaxProcessorNumber 4
```

Allows the adapter to use processors 1,2,3,4 only on the NUMA node 0.

## 4.19 Receive Side Scaling (RSS)

RSS attempts to dynamically distribute data processing across the available host CPUs in order to spread the workload. RSS is enabled by default and can significantly improve the performance of the host CPU when handling large amounts of network data.

RSS cmdlets allow per-adapter RSS configuration, so different adapters can have different RSS configurations.

### Identify current settings

```
PS > Get-NetAdapterRss -Name "<interface>"
```

```
Name : Ethernet 4
InterfaceDescription Solarflare XtremeScale X2522 10/25GbE Adapter #2
Enabled : True
NumberOfReceiveQueues : 8
Profile : NUMAStatic
BaseProcessor: [Group:Number] : 0:0
MaxProcessor: [Group:Number] : 0:3
MaxProcessors : 4
RssProcessorArray: [Group:Number/NUMA Distance] : 0:0/0 0:1/0 0:2/0
0:3/0
IndirectionTable: [Group:Number] :
```

### Enable RSS

```
PS > Set-NetAdapterRss -Name "<interface>" -Enabled 1
```

### Disable RSS

```
PS > Set-NetAdapterRss -Name "<interface>" -Enabled 0
```

### Number of Receive Queues

```
PS > St-NetAdapterRss -Name "<interface>" -NumberOfReceiveQueues 4
```



## RSS Profile

Determines the logical processors that can be used by a network adapter for RSS.

```
PS > Set-NetAdapterRss -Name "<interface>" -Profile closest
```

RSS Profile	Description
<b>Closest</b>	Processors near the network adapter's base RSS processor are preferred. Windows may rebalance processors dynamically based on load.
<b>ClosestStatic</b>	Processors near the network adapter's base RSS processor are preferred. Windows will <b>not</b> rebalance processors dynamically based on load.
<b>NUMA</b>	Will select processors on different NUMA nodes. Windows may rebalance processors dynamically based on load.
<b>NUMAStatic</b>	<b>Default.</b> Will select processors on different NUMA nodes. Windows will <b>not</b> rebalance processors dynamically based on load.
<b>Conservative</b>	Use as few processors as possible to sustain the load. This option can help to reduce the number of interrupts.

Refer to [NUMA Node on page 152](#) for further NUMA node considerations.



**NOTE:** Changing the RSS profile requires a restart of the adapter.

## RSS Base Processor

For a Solarflare network adapter, the RSS base processor is 0 (zero), which means it starts processing on CPU core 0 which is also the default processor for all other general Windows processes and will likely be the default for all other network adapters in the server.

To avoid this unnecessary contention, set the adapter RSS base processor to another processor on the NUMA node the Solarflare adapter is assigned to.

### **cmdlet:**

```
PS > Set-NetAdapterRss -Name "<interface>" -BaseProcessorNumber 1 -  
Numanode 0
```

### RSS Max Processor

Used with the BaseProcessorNumber, this identifies the range of processors that can be used by RSS.

**cmdlet:**

```
PS > Set-NetAdapterRss -Name "<interface>" -Numanode 0 -
BaseProcessorNumber 4 -MaxProcessorNumber 7
```

The above example means that lowest processor that can be used for RSS is processor 4, and it can use processors 4,5,6,7.

### RSS Base Processor Group

On systems with more than 64 logical processors - identify the processor group. Systems with ≤ 64 processors have only the single group 0.

**cmdlet:**

```
PS > Set-NetAdapterRss -Name "<interface>" -BaseProcessorGroup <value>
```



**NOTE:** Setting the 'numanode' parameter will automatically set the correct base processor group.

### RSS Max Processors

Set the number of processors to be used by RSS.

**cmdlet:**

```
PS > Set-NetAdapterRss -Name "<interface>" -MaxProcessors 4
```

Also refer to the following section: [Receive and Transmit Buffers on page 157](#).

## 4.20 Receive and Transmit Buffers

The **Receive Buffers** configuration option sets the number of receive packet buffers allocated for each RSS receive queue. An appropriate hardware receive queue size is chosen by the driver based on this setting. Increasing this value may improve receive performance and reduce occurrences of packet discards under heavy load, but consumes additional system memory.



**CAUTION:** Setting this value higher than necessary can waste limited system resources, impact overall system performance and may cause the driver to fail initialization.

### Identify current settings

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>"
```

Receive Buffers	1152	*ReceiveBuffers {1152}
Transmit Buffers	1024	*TransmitBuf... {1024}

### Change Settings

```
PS > Set-NetAdapterAdvancedProperty -Name "Ethernet 8" -DisplayName  
"Receive Buffers" -DisplayValue 2048
```

- Receive buffer values must be in 128 increments in the range 640-6144.
- Transmit buffer values must be in 512 increments in the range 512-2048.

The **Transmit Buffers** configuration option sets the number of descriptors that can be queued simultaneously on each hardware transmit queue. An appropriate hardware transmit queue size is chosen by the driver based on this setting. As each packet may require multiple descriptors the number of packets that can be queued simultaneously may be lower than the queue size. Increasing this value may improve transmit performance, but also consumes more system resources.

## Ethernet Frame Length

The maximum Ethernet frame length used by the adapter to transmit data is (or should be) closely related to the MTU (maximum transmission unit) of your network. The network MTU determines the maximum frame size that your network is able to transmit across all devices in the network.



**NOTE:** For optimum performance set the Ethernet frame length to your network MTU.

## Ethernet Link Speed

Generally, it is not necessary to configure the link speed of the adapter. The adapter by default will negotiate the link speed dynamically, connecting at the maximum, supported speed. However, if the adapter is unable to connect to the link partner, set a fixed link speed through the advanced properties.

```
PS > Get-NetAdapterAdvancedProperty -Name "<interface>" | Format-List -
Property "*"

```

```
ValueName           : *SpeedDuplex
ValueData           : {0}
ifAlias             : Ethernet 4
InterfaceAlias      : Ethernet 4
ifDesc              : Solarflare XtremeScale X2522 10/25GbE Adapter
#2
Caption             : MSFT_NetAdapterAdvancedPropertySettingData
'Solarflare XtremeScale X2522 10/25GbE Adapter #2'
Description         : Speed & Duplex
ElementName         : Speed & Duplex
InstanceID          : {F6B41E13-80D5-4BE0-B45C-
CC314CCADB6C}::*SpeedDuplex
InterfaceDescription : Solarflare XtremeScale X2522 10/25GbE Adapter
#2
Name                : Ethernet 4
Source              : 3
SystemName          : <server>.<domain>.lan
DefaultDisplayValue : Auto Negotiation
DefaultRegistryValue : 0
DisplayName         : Speed & Duplex
DisplayParameterType : 5
DisplayValue        : Auto Negotiation
NumericParameterBaseValue :
NumericParameterMaxValue :
NumericParameterMinValue :
NumericParameterStepValue :
Optional           : False
RegistryDataType   : 1
RegistryKeyword    : *SpeedDuplex
RegistryValue      : {0}
ValidDisplayValues : {Auto Negotiation, 1.0 Gbps Full Duplex, 10
Gbps Full Duplex}
ValidRegistryValues : {0, 6, 7}
PSComputerName     :
CimClass           : ROOT/
StandardCimv2:MSFT_NetAdapterAdvancedPropertySettingData
CimInstanceProperties : {Caption, Description, ElementName,
InstanceID...}
CimSystemProperties :
Microsoft.Management.Infrastructure.CimSystemProperties

```

```
PS > Set-NetAdapterAdvancedProperty -Name "<interface>" -Displayname
"Speed & Duplex" -Displayvalue "10 Gbps Full Duplex"

```

## 4.21 Virtual Machine Queue

Solarflare adapters support VMQ to offload the classification and delivery of network traffic destined for Hyper-V virtual machines to the network adapter thereby reducing the CPU load on Hyper-V hosts. Dynamic VMQ will dynamically distribute received network traffic across available CPUs while adjusting for network load by, if necessary, bringing in more processors or releasing processors under light load conditions.

VMQ supports the following features:

- Classification of received network traffic in hardware by using the destination MAC address and the VLAN identifier to route packets to different receive queues dedicated to each virtual machine.
- Using the network adapter to directly transfer received network traffic to a virtual machine's shared memory avoiding a potential software-based copy from the Hyper-V host to the virtual machine.
- Scaling to multiple processors by processing network traffic destined for different virtual machines on different processors.

### Enable VMQ

```
PS > Set-NetAdapterVmq -Name "<interface>" -Enabled 1
```

### Disable VMQ

```
PS > Set-NetAdapterVmq -Name "<interface>" -Enabled 0
```

### Get the VMQ properties

```
PS> Get-NetAdapterVmq
```

### Get the VMQs allocated

```
PS> Get-NetAdapterVmqQueue
```

### Get the virtual network adapter of the virtual machines

```
PS> Get-VMNetworkAdapter
```

## 4.22 Teaming and VLANs

### About Teaming

Solarflare adapters use the native Windows **NetLbfo** module for teaming configuration and management. The following teaming configurations are supported:

- IEEE 802.1AX (802.3ad) Dynamic link aggregation.
- Static link aggregation.
- Fault tolerant teams.

### Team Configurations

- All Solarflare adapter ports on all installed Solarflare adapters.
- Selected ports e.g. from a dual port Solarflare adapter, the first port could be a member of team A and the second port a member of team B or both ports members of the same team.
- Mixed Solarflare and non-Solarflare adapters.
- A port can be a member of more than one team.
- A port can be assigned more than one VLAN.

### Link Aggregation

A mechanism supporting load balancing and fault tolerance across a team of network adapters and intermediate switch.

- Requires configuration at both ends of the link.
- All links in the team are bonded into a single virtual link with a single MAC address.

Two or more physical links can increase the potential throughput available between the link partners, and improve resilience against link failures.

- All links in the team must be between the same two link partners.
- Links must be full-duplex.
- Traffic is distributed evenly to all links connected to the same switch.
- In case of link failover, traffic on the failed link will be re-distributed to the remaining links.

Link aggregation offers the following functionality:

- Teams can be built from mixed media (i.e. UTP and Fiber).
- All protocols can be load balanced without transmit or receive modifications to frames.

- Multicast and broadcast traffic can be load balanced.
- Short recovery time in case of failover.
- Solarflare supports up to 64 link aggregation port groups per system.
- Solarflare supports up to 64 ports and VLANs in a link aggregation port group.

## Dynamic Link Aggregation

Uses the Link Aggregation Control Protocol (LACP) (IEEE 802.1AX - previously called 802.3ad) to negotiate the ports that will make up the team.

- LACP must be enabled at both ends of the link for a team to be operational.
- LACP will automatically determine which physical links can be aggregated.
- Provides fault tolerance and load balancing.
- Standby links are supported, but are not considered part of a link aggregation until a link within the aggregation fails.
- VLANs are supported within 802.1AX teams.
- In the event of failover, the load on the failed link is redistributed over the remaining links.



**NOTE:** A switch must support 802.1AX (802.3ad) dynamic link aggregation to use this method of teaming.

## Fault-Tolerant Teams

Fault tolerant teaming can be implemented on any switch. It can also be used with each team member network link connected to separate switches.

A fault-tolerant team is a set of one or more network adapters bound together by the teaming driver. The team improves network availability by providing standby adapters. At any one moment no more than one of the adapters will be active with the remainder either in standby or in a fault state.



**NOTE:** All adapters in a fault-tolerant team must be part of the same broadcast domain.

### Failover

The teaming driver monitors the state of the active adapter and, in the event that its physical link is lost (down) or that it fails in service, swaps to one of the standby adapters. A link in a failed state will not be available as a standby while the failed state persists.

## VLANs and Teaming

VLANs are used to divide a physical network into multiple broadcast domains and are supported on all Solarflare adapter teaming configurations.

## Teaming - Configuration

Teams can be configured using NetLbfo specific powershell Cmdlets or via the NIC Teaming dialog (GUI).

### Teaming - Cmdlets

The Windows teaming module, NetLbfo, can be configured using Powershell cmdlets.

```
PS > Get-Command -Module NetLbfo
```

Name	Version	Source
Add-NetLbfoTeamMember	2.0.0.0	NetLBFO
Add-NetLbfoTeamNic	2.0.0.0	NetLBFO
Get-NetLbfoTeam	2.0.0.0	NetLBFO
Get-NetLbfoTeamMember	2.0.0.0	NetLBFO
Get-NetLbfoTeamNic	2.0.0.0	NetLBFO
New-NetLbfoTeam	2.0.0.0	NetLBFO
Remove-NetLbfoTeam	2.0.0.0	NetLBFO
Remove-NetLbfoTeamMember	2.0.0.0	NetLBFO
Remove-NetLbfoTeamNic	2.0.0.0	NetLBFO
Rename-NetLbfoTeam	2.0.0.0	NetLBFO
Set-NetLbfoTeam	2.0.0.0	NetLBFO
Set-NetLbfoTeamMember	2.0.0.0	NetLBFO
Set-NetLbfoTeamNic	2.0.0.0	NetLBFO

For further information - use the Powershell cmdlet help:

```
PS > Get-Help [Add|Remove|Rename|Get|Set]-NetLbfo
```

A complete list describing NetLbfo (teaming) cmdlets can also be found with the following links:

- [Windows-teaming-documentation](https://docs.microsoft.com/en-us/powershell/module/netlbfo/windows-teaming-documentation)  
<https://docs.microsoft.com/en-us/powershell/module/netlbfo/>

### Teaming - GUI

Enter the following at a command prompt or powershell prompt:

```
PS > LbfoAdmin
```

Select the adapter(s) and then use the TASKS tab to create/configure/add/remove teams.



## 4.23 Adapter Statistics

Networking statistics for an adapter can be viewed with the following cmdlet.

```
PS > Get-NetAdapterStatistics -Name "<interface>"
```

```
Name           RxBytes RxUnicastPackets TxBytes TxUnicastPackets
Ethernet 4     0         0             0         0
```

```
PS > Get-NetAdapterStatistics -Name "<interface>" | Format-List -Property "*"
```

```
ifAlias           : Ethernet 4
InterfaceAlias    : Ethernet 4
ifDesc            : Solarflare XtremeScale X2522 10/25GbE Adapter #2
Caption           : MSFT_NetAdapterStatisticsSettingData
'Solarflare XtremeScale X2522 10/25GbE Adapter #2'
Description       : Solarflare XtremeScale X2522 10/25GbE Adapter #2
ElementName       : Solarflare XtremeScale X2522 10/25GbE Adapter #2
InstanceID        : {F6B41E13-80D5-4BE0-B45C-CC314CCADB6C}
InterfaceDescription : Solarflare XtremeScale X2522 10/25GbE Adapter #2
Name              : Ethernet 4
Source            : 2
SystemName        : <server>.<domain>.lan
OutboundDiscardedPackets : 0
OutboundPacketErrors : 0
RdmaStatistics    :
ReceivedBroadcastBytes : 0
ReceivedBroadcastPackets : 0
ReceivedBytes     : 0
ReceivedDiscardedPackets : 0
ReceivedMulticastBytes : 0
ReceivedMulticastPackets : 0
ReceivedPacketErrors : 0
ReceivedUnicastBytes : 0
ReceivedUnicastPackets : 0
RscStatistics     : MSFT_NetAdapter_RscStatistics
SentBroadcastBytes : 0
SentBroadcastPackets : 0
SentBytes         : 0
SentMulticastBytes : 0
SentMulticastPackets : 0
SentUnicastBytes  : 0
SentUnicastPackets : 0
SupportedStatistics : 4163583
```

## 4.24 Performance Tuning on Windows

### Introduction

Solarflare network adapters are designed for high-performance network applications. The adapter driver is pre-configured with default performance settings designed for optimum performance across a broad class of applications.

Occasionally, application performance can be improved by additional tuning to best suit the application.

There are three metrics that should be considered when tuning an adapter:

- Throughput
- Latency
- CPU utilization

Transactional (request-response) network applications can be very sensitive to latency whereas bulk data transfer applications are more dependent on throughput.

The tuning recommendations should be considered in conjunction the following Microsoft performance tuning guides:

- [Performance Tuning Guidelines for Windows Server 2016](#).

### Max Frame Size

A larger maximum frame size will improve adapter throughput and CPU utilization. CPU utilization is improved, because it takes fewer packets to send and receive the same amount of data. Solarflare adapters support maximum frame sizes up to 9216 bytes (this does not include the Ethernet preamble or frame check sequence).



**NOTE:** The maximum frame size setting should include the Ethernet frame header. The Solarflare drivers support 802.1p. This allows Solarflare adapters on Windows to optionally transmit packets with 802.1Q tags for QoS applications. It requires an Ethernet frame header size of 18 bytes (6 bytes source MAC address, 6 bytes destination MAC address, 2 bytes 802.1Q tag protocol identifier, 2 bytes 802.1Q tag control information, and 2 bytes EtherType). The default maximum frame size is therefore 1518 bytes.

The maximum frame size is changed by changing the Max Frame Size setting in the Network Adapter's Advanced Properties Page.

## **Interrupt Moderation (Interrupt Coalescing)**

*Interrupt moderation* reduces the number of interrupts generated by the adapter by coalescing multiple received packet events and/or transmit completion events together into a single interrupt.

The *interrupt moderation interval* sets the minimum time (in microseconds) between two consecutive interrupts. Coalescing occurs only during this interval:

- When the driver generates an interrupt, it starts timing the moderation interval.
- Any events that occur before the moderation interval expires are coalesced together into a single interrupt, that is raised only when the interval expires. A new moderation interval then starts, during which no interrupt is raised.
- An event that occurs after the moderation interval has expired gets its own dedicated interrupt, that is raised immediately. A new moderation interval then starts, during which no interrupt is raised.

Interrupt moderation settings are **critical for tuning adapter latency**:

- Disabling the adaptive algorithm will:
  - reduce jitter
  - allow setting the moderation interval as required to suit conditions.
- Increasing the interrupt moderation interval will:
  - generate less interrupts
  - reduce CPU utilization (because there are less interrupts to process)
  - increase latency
  - improve peak throughput.
- Decreasing the interrupt moderation interval will:
  - generate more interrupts
  - increase CPU utilization (because there are more interrupts to process)
  - decrease latency
  - reduce peak throughput.
- Turning off interrupt moderation will:
  - generate the most interrupts
  - give the highest CPU utilization
  - give the lowest latency
  - give the biggest reduction in peak throughput.

For many transaction request-response type network applications, the benefit of reduced latency to overall application performance can be considerable. Such benefits typically outweigh the cost of increased CPU utilization. It is recommended that:

- Interrupt moderation is disabled for applications that require best latency and jitter performance, such as market data handling.
- Interrupt moderation is enabled for high throughput single (or few) connection TCP streaming applications, such as iSCSI.

Interrupt moderation and time interval value can be disabled or enabled using the Interrupt Moderation setting in the Network Adapter's Advanced Properties Page.

### Receive Side Scaling (RSS)

RSS is enabled by default for best networking performance.

The number of RSS queues can be adjusted to suit the workload:

- The number of RSS CPUs is limited by the number of RSS queues. The driver does not target multiple RSS queues to the same CPU. Therefore:
  - It is best to set the maximum number of RSS queues to be equal to the maximum number of RSS CPUs (or the next higher setting if the equal option is unavailable).
  - The number of queues can be reduced in order to isolate CPU cores for application processing.
  - The number of queues can be increased to spread the load over more cores. This will also increase the amount of receive buffering due to a larger number of RX queues.



**NOTE:** If hyper-threading is enabled, RSS will only select one thread from each CPU core.

- For low latency low jitter applications select the NUMA scaling static RSS profile. Set both the maximum number of RSS processors and the number of RSS queues to be equal to the number of CPU cores

In multi-port scenarios, restrict each port to a subset of RSS processors using the base and max processor settings.

- For other applications use as few RSS processors as required to cope with the traffic load, leaving other CPUs free for other tasks.

## Adapter RX/TX Descriptor Ring Buffers

Adapter Receive and Transmit descriptor rings are buffers on the adapter used to receive and transmit network packets.

Ring buffers settings can be changed via regedit or with Powershell.

### RX/TX Queue Sizes - Registry Setting:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\services\SFCBUS\Parameters
ReceiveQueueInitialFill
    Type DWORD
    Default (decimal): 256
    Valid Values (decimal): 1 to ReceiveQueueSize
ReceiveQueueSize
    Type: DWORD
    Default (decimal): 1024
    Valid Values (decimal): 512, 1024, 2048 or 4096
TransmitQueueSize
    Type: DWORD
    Default (decimal): 1024
    Valid Values (decimal): 512, 1024, 2048 or 4096
```

### Changing RX/TX Queue Size - Powershell:

- 1 In PowerShell, check if entries already exist:  

```
PS > Get-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\SFCBUS\Parameters
```
- 2 If these registry entries do not exist, they can be created and set with:  

```
PS > New-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\SFCBUS\Parameters
-Name <name> -PropertyType dword -Value <value>
```
- 3 When the registry entries does exist, it can be modified:  

```
PS > Set-ItemProperty -Path HKLM:\SYSTEM\CurrentControlSet\Services\SFCBUS\Parameters
-Name <name> -Value <value>
```

where:

  - <name> is ReceiveQueueSize or TransmitQueueSize
  - <value> is one of 512, 1024, 2048 or 4096.

### Refill Batch Size

The refill batch size affects how many descriptors are required on each ring refill. Setting this to a smaller value allows the ring buffer to be refilled quicker.

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\services\SFCBUS\Parameters
ReceiveQueueBatch
    Type: DWORD
    Default (decimal): 128
    Valid Values (decimal): multiples of 8, try values 8 to 128.
```

## Other Considerations

### PCI Express Lane Configurations

Solarflare adapters require a PCIe Gen 3.x x16 slot for optimal performance. The Solarflare driver will insert a warning in the Windows Event Log if it detects that the adapter is placed in a sub-optimal slot.

In addition, the latency of communications between the host CPUs, system memory and the Solarflare PCIe adapter may be PCIe slot dependent. Some slots may be “closer” to the CPU, and therefore have lower latency and higher throughput:

- If possible, install the adapter in a slot which is local to the desired NUMA node.

### Memory bandwidth

Many chipsets use multiple channels to access main system memory. Maximum memory performance is only achieved when the chipset can make use of all channels simultaneously. This should be taken into account when selecting the number of memory modules (DIMMs) to populate in the server. For optimal memory bandwidth in the system, it is likely that:

- all DIMM slots should be populated
- all NUMA nodes should have memory installed.

Please consult the motherboard documentation for details.

### Intel Hyper-Threading Technology

On systems that support Intel Hyper-Threading Technology users should consider benchmarking or application performance data when deciding whether to adopt hyper-threading on a particular system and for a particular application. Solarflare have identified that hyper-threading is generally beneficial.

### TCP/IP Options

TCP timestamps, window scaling and selective acknowledgments are enabled by default on supported platforms, and include receive window tuning and congestion control algorithms that automatically adapt to 10 gigabit connections. There is therefore no need to change these settings.

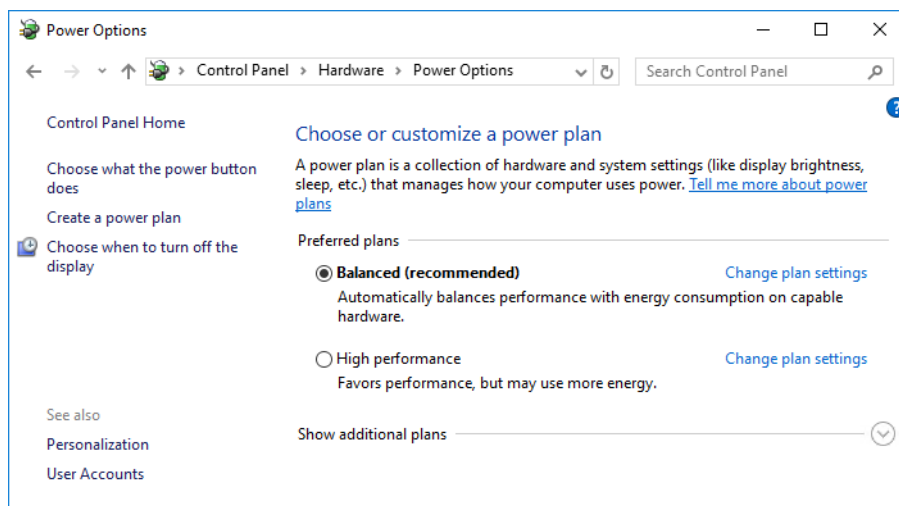
### Power Saving Mode

Modern processors utilize design features that enable a CPU core to drop into low power states when instructed by the operating system that the CPU core is idle. When the OS schedules work on the idle CPU core (or when other CPU cores or devices need to access data currently in the idle CPU core’s data cache) the CPU core is signaled to return to the fully on power state. These changes in CPU core power states create additional network latency and jitter. Solarflare recommend to achieve the lowest latency and lowest jitter that the “C1E power state” or “CPU power saving mode” is disabled within the system BIOS.

In general the user should examine the system BIOS settings and identify settings that favor performance over power saving. In particular look for settings to disable:

- C states / Processor sleep/idle states
- Enhanced C1 CPU sleep state (C1E)
- Any deeper C states (C3 through to C6)
- P states / Processor throttling
- Processor Turbo mode
- Ultra Low Power State
- PCIe Active State Power Management (ASPM)
- Unnecessary SMM/SMI features

The latency can be improved by selecting the highest performance power plan from the Control Panel > Hardware > Power Options:



The powercfg.exe utility that is installed with Windows can also be used to select the power scheme.

- List all power schemes in the current user's environment:

```
PS > PowerCfmg /LIST
```

```
Existing Power Schemes (* Active)
```

```
Power Scheme GUID: 381b4222-f694-41f0-9685-ff5bb260df2e (Balanced) *
Power Scheme GUID: 8c5e7fda-e8bf-4a96-9a85-a6e23a8c635c (High
performance)
Power Scheme GUID: a1841308-3541-4fab-bc81-f71556f20b4a (Power saver)
```

```
PS > PowerCfmg /SETACTIVE <GUID>
```

## Firewalls and anti-virus software

Depending on the system configuration, the following software may have a significant impact on throughput and CPU utilization, in particular when receiving multicast UDP traffic:

- the built-in Windows Firewall and Base Filtering Engine
- other third-party firewall or network security products
- anti-virus checkers.

This is the case even if the software has no rules configured but is still active.

Where high throughput is required on a particular port, the performance will be improved by disabling the software on that port:



**CAUTION:** The Windows (or any third party) Firewall should be disabled with caution. The network administrator should be consulted before making any changes.

## Tuning Recommendations

The following tables provide recommendations for tuning settings for different performance requirements.

**Table 34: Throughput Tuning Settings**

Tuning Parameter	How?
Firewall	Disable the Base Filter Engine.
Interrupt Moderation	Leave at default (Enabled).
Large Send Offloads	Leave at default (Enabled).
Max Frame Size	Configure to maximum supported by network in Network Adapter's Advanced Properties.
Memory bandwidth	Ensure memory utilizes all memory channels on system motherboard.
Offload Checksums	Leave at default.
PCI Express Lane Configuration	Ensure the adapter is in an x8 or x16 slot (PCIe Gen 2.0 or PCIe Gen 3.x).
Power Saving Mode	Leave at default.
Receive Side Scaling (RSS)	Leave at default.



**Table 34: Throughput Tuning Settings (continued)**

<b>Tuning Parameter</b>	<b>How?</b>
RSS NUMA Node	Leave at default (All).
TCP Protocol Tuning	Leave at default (install with “Optimize Windows TCP/IP protocol settings for 10G networking” option selected).

**Table 35: Latency Tuning Settings**

<b>Tuning Parameter</b>	<b>How?</b>
Firewall	Disable the Base Filter Engine.
Interrupt Moderation	Disable in Network Adapter’s Advanced Properties.
Interrupt Moderation Time	Leave at default (60µs). This setting is ignored when interrupt moderation is disabled.
Large Receive Offloads	Disable in Network Adapter’s Advanced Properties.
Large Send Offloads	Leave at default (Enabled).
Max Frame Size	Configure to maximum supported by network in Network Adapter’s Advanced Properties.
Memory bandwidth	Ensure memory utilizes all memory channels on system motherboard.
Offload Checksums	Leave at default (Enabled).
PCI Express Lane Configuration	Ensure the adapter is in an x8 slot (PCIe Gen 2.0 or PCIe Gen 3.x) or x16 slot.
Power Saving Mode	Disable C1E and other CPU sleep modes to prevent OS from putting CPUs into lowering power modes when idle.
Receive Side Scaling	Application dependent
RSS NUMA Node	Leave at default (All).
TCP Protocol Tuning	Leave at default (install with “Optimize Windows TCP/IP protocol settings for 10G networking” option selected).
TCP/IP Checksum Offload	Leave at default

## 4.25 List Installed Adapters

From the Powershell command line, use the following command to list Solarflare adapters.

```
PS> Get-NetAdapter [|? DriverProvider -eq Solarflare]
```

### To identify adapter serial number:

Use the following command in Powershell to identify the adapter serial number:

```
PS> (Get-WmiObject -Namespace root\wmi  
Solarflare_VPD).Solarflare_VPD_ReadStringKeyword(16, 0x4E53).StrData
```

### To identify adapter model:

This is exposed by various powershell commands e.g.

```
PS> C:\sfdriver> Get-NetAdapter |? DriverProvider -eq Solarflare | Format-  
Table -View Driver
```

Name	InterfaceDescription	DriverFileName
Ethernet 4	Solarflare XtremeScale X2522-25G	SFN.sys

The adapter model is also visible through the Windows standard Device Manager GUI interface.

## 4.26 Startup/Boot time Errors

If drivers fail to load or adapters are not listed following a reboot, check the Windows Event Viewer for issues relating to driver configuration and installation.

- **Event Viewer > Applications and Services > Solarflare > Network Adapters** for messages related to Solarflare drivers.
- **Event Viewer > Custom Views > Device Manager - Solarflare** for events related to Solarflare adapters.

# 5

## Solarflare Adapters on VMware

This chapter documents procedures for installation and configuration of Solarflare adapters on VMware® vSphere 2016 ESXi 6.0u3e™ (and later ux versions), ESXi 6.5™ and later versions.

- [Native ESXi Driver \(VMkernel API\) on page 174](#)
- [Legacy Driver \(vmklinux API\) on page 174](#)
- [System Requirements on page 174](#)
- [Distribution Packages on page 175](#)
- [VMware Feature Set on page 176](#)
- [Install Solarflare Drivers on page 178](#)
- [Driver Configuration on page 180](#)
- [Adapter Configuration on page 181](#)
- [Granting access to the NIC from the Virtual Machine on page 181](#)
- [NIC Teaming on page 182](#)
- [Configuring VLANs on page 183](#)
- [Performance Tuning on VMware on page 184](#)
- [Interface Statistics on page 194](#)
- [vSwitch/VM Network Statistics on page 194](#)
- [CIM Provider on page 197](#)
- [Adapter Firmware Upgrade - sfupdate\\_esxi on page 198](#)
- [Adapter Configuration - sfboot\\_esxi on page 201](#)
- [ESXCLI Extension on page 203](#)
- [vSphere Client Plugin on page 212](#)
- [Fault Reporting - Diagnostics on page 222](#)
- [Network Core Dump on page 223](#)
- [Adapter Diagnostic Selftest on page 223.](#)

## 5.1 Native ESXi Driver (VMkernel API)

The Solarflare VMware driver package includes a *native* ESXi driver using the VMware *VMkernel* API.

- VMkernel is VMware's new recommended API for network drivers.
- All future Solarflare development will focus on this native driver.

The Solarflare native driver supports the following Solarflare adapters:

- SFN8042M
- SFN8522M, SFN8522 (SFP+)
- SFN8522-Onload (SFP+), SFN8522-Plus (SFP+)
- SFN8542 (QSFP+), SFN8542-Plus (QSFP+)
- SFN8722 (OCP mezzanine adapter)
- X2522, X2522-Plus (10G)
- X2522-25G, X2522-25G-Plus
- X2541, X2542 models
- X2552 OCP 2.0

The Solarflare native driver requires ESXi 6.0u3e (and higher ux versions), 6.5 or 6.7

## 5.2 Legacy Driver (vmklinux API)

There is also a *legacy* driver package, that uses the deprecated *vmklinux* API.

- The older vmklinux API will be removed in future ESXi versions.
- This legacy driver is no longer under active development.

The Solarflare legacy driver supports the following Solarflare adapters:

- SFN5000 series
- SFN6000 series
- SFN7000 series
- SFN8000 series.

The Solarflare legacy driver supports ESXi versions up to and including 6.5.

## 5.3 System Requirements

Refer to [Software Driver Support on page 16](#) for supported VMware host platforms.

## 5.4 Distribution Packages

VMware drivers, firmware and utilities are available from [support.solarflare.com](http://support.solarflare.com).



**NOTE:** To prevent file deletion when the ESXi host is rebooted, files can be copied to a directory created by the user in any host datastore under /vmfs, for example: /vmfs/volumes/datastore1/solarflare

### Packages for ESXi 6.5 (and later versions)

The following packages are available for ESXi 6.5 (and later versions):

Part Number	Description
SF-118824-LS	Solarflare VMware ESXi 6.5 / 6.7 Native Driver (VIB)
SF-118825-LS	Solarflare VMware ESXi 6.5 / 6.7 Native Driver (Offline Bundle)
SF-120055-LS	Solarflare VMware Utilities CIM Provider for Native Driver
SF-120056-LS	Solarflare vSphere client plugin Windows Installer <b>Available for ESXi 6.5 or later versions.</b>
SF-120054-LS	Solarflare Linux Utilities (32-bit) for use with the ESXi CIM Provider for Native Driver
SF-120773-LS	ESXCLI extensions (VIB)
SF-121528-LS	Solarflare Firmware VIB

### Packages for ESXi 6.0-u3e (and later ux versions)

The following packages are available for ESXi 6.0-u3e (and later ux versions):

Part Number	Description
SF-120732-LS	Solarflare VMware ESXi 6.0 Native Driver (VIB)
SF-120733-LS	Solarflare VMware ESXi 6.0 Native Driver (Offline Bundle)
SF-120055-LS	Solarflare VMware Utilities CIM Provider for Native Driver
SF-120054-LS	Solarflare Linux Utilities (32-bit) for use with the ESXi CIM Provider for Native Driver
SF-120773-LS	ESXCLI extensions (VIB)
SF-121528-LS	Solarflare Firmware VIB

## 5.5 VMware Feature Set

The table below lists the features available from the VMware host.

**Table 36: VMware Host Feature Set**

<b>Basic Driver Features</b>	
<b>Jumbo frames</b>	Support for MTUs (Maximum Transmission Units) from 1500 bytes to 9000 bytes. <ul style="list-style-type: none"> <li>See <a href="#">Adapter MTU (Maximum Transmission Unit) on page 187</a></li> </ul>
<b>Teaming</b>	Improve server reliability by creating teams on either the host vSwitch, Guest OS or physical switch to act as a single adapter. <p>See <a href="#">NIC Teaming on page 182</a></p>
<b>Virtual LANs (VLANs)</b>	Support for VLANs on the host, guest OS and virtual switch. <ul style="list-style-type: none"> <li>See <a href="#">Configuring VLANs on page 183</a></li> </ul>
<b>VLAN tag insertion</b>	Support offload of vlan tag insertion to hardware (firmware). The NIC must use the full-feature or auto firmware variants. If the firmware variant is not full-feature or auto, vlan tag insertion offload is not available.
<b>FEC</b>	Forward Error Correction employs redundancy in the channel coding as a technique used to reduce bit errors (BER) in noisy or unreliable communications channels. <ul style="list-style-type: none"> <li>See <a href="#">ESXCLI Extension on page 203</a></li> </ul>
<b>Fault diagnostics</b>	Support for comprehensive adapter and cable fault diagnostics and system reports. <ul style="list-style-type: none"> <li>See <a href="#">CIM Provider on page 197</a></li> </ul>
<b>Interrupt moderation</b>	Coalesce multiple received packets events or transmit completion events into a single interrupt. <ul style="list-style-type: none"> <li>See <a href="#">Interrupt Moderation (Interrupt Coalescing) on page 188</a></li> </ul>
<b>Pause frames</b>	Separate control for receive and transmit.
<b>RX/TX ring buffers</b>	Set the adapter RX/TX buffer sizes. <ul style="list-style-type: none"> <li>See <a href="#">Adapter RX/TX ring buffer size on page 181</a></li> </ul>
<b>Network core dumping</b>	Transfer core dump file to vCenter Server Appliance after host panic. <ul style="list-style-type: none"> <li>See <a href="#">Network Core Dump on page 223</a> for configuration detail.</li> </ul>
<b>SR-IOV</b>	Supported by legacy and native drivers for all adapter models and a maximum 63 Virtual Functions per PF. <ul style="list-style-type: none"> <li>See <a href="#">SR-IOV Virtualization Using ESXi on page 250</a></li> </ul>
<b>Firmware Features</b>	
<b>Port level stats</b>	<ul style="list-style-type: none"> <li>See <a href="#">ESXCLI Extension on page 203</a></li> </ul>
<b>Cable Type</b>	<ul style="list-style-type: none"> <li>See <a href="#">Adapter Diagnostic Selftest on page 223</a></li> </ul>
<b>PHY Address</b>	<ul style="list-style-type: none"> <li>See <a href="#">Adapter Diagnostic Selftest on page 223</a></li> </ul>

**Table 36: VMware Host Feature Set (continued)**

<b>Transceiver Type</b>	<ul style="list-style-type: none"> <li>See <a href="#">Adapter Diagnostic Selftest on page 223</a></li> </ul>
<b>Offload Features</b>	
<b>Checksum offload</b>	<p>TCP/UDP over IPv4/IPv6 checksum.</p> <ul style="list-style-type: none"> <li>See <a href="#">TCP/UDP Checksum Offload on page 190</a></li> </ul>
<b>TSO</b>	<p>Support for TCP Segmentation Offload (TSO).</p> <ul style="list-style-type: none"> <li>See <a href="#">TCP Segmentation Offload (TSO) on page 190</a></li> </ul>
<b>Overlay support VXLAN</b>	<p>Support for VXLAN, checksum offload against these packets and RSS support for encapsulated inner layer 3/4 headers.</p>
<b>Overlay support GENEVE</b>	<p>Support for Geneve, checksum offload against these packets and RSS support for encapsulated inner layer 3/4 headers.</p> <p><b>Not supported on ESXi 6.0.</b></p>
<b>Performance Features</b>	
<b>NetQueue</b>	<p>Configurable number of NetQueues</p> <ul style="list-style-type: none"> <li>See <a href="#">Driver Configuration on page 180</a></li> <li>See <a href="#">VMware ESXi NetQueue on page 184</a></li> </ul>
<b>RSS</b>	<p>Configurable number of RSS queues</p> <ul style="list-style-type: none"> <li>See <a href="#">Driver Configuration on page 180</a></li> <li>See <a href="#">Receive Side Scaling (RSS) on page 192</a></li> </ul>
<b>Management Features</b>	
<b>vSphere Client Plugin</b>	<p>Registered with a vCenter Server Appliance, for configuration and management of Solarflare adapters.</p> <p><b>Not supported on ESXi 6.0.</b></p> <ul style="list-style-type: none"> <li>See <a href="#">vSphere Client Plugin on page 212</a></li> </ul>
<b>ESXCLI extension</b>	<p>Solarflare extensions to the command line interface.</p> <ul style="list-style-type: none"> <li>See <a href="#">ESXCLI Extension on page 203</a></li> </ul>
<b>Firmware update</b>	<p>Support for Boot ROM and Phy transceiver firmware upgrades for in-field upgradeable adapters.</p> <p><b>The firmware update utility, <code>sfupdate_esxi</code>, is provided through the supplied CIM provider package.</b></p> <p>See <a href="#">Adapter Firmware Upgrade - <code>sfupdate_esxi</code> on page 198</a>.</p> <p>Firmware can also be upgraded through the vSphere Client Plugin. See <a href="#">vSphere Client Plugin on page 212</a>.</p> <p>Firmware can also be upgraded through the extensions command line. See <a href="#">ESXCLI Extension on page 203</a>.</p>
<b>Adapter hardware and bootROM configuration</b>	<p>Adapter configuration with <code>sfboot</code>.</p> <ul style="list-style-type: none"> <li>See <a href="#">Adapter Configuration - <code>sfboot_esxi</code> on page 201</a>.</li> </ul>
<b>Sensors</b>	<p>Read adapter voltage and temperature sensors.</p> <ul style="list-style-type: none"> <li>See <a href="#">Sensors on page 209</a>.</li> </ul>

## 5.6 Install Solarflare Drivers



**CAUTION:** The Solarflare native ESXi driver is a host driver only. This driver should NOT be installed on a Virtual Machine.

The Solarflare adapter driver on the ESXi host is named **sfvmk**.

### Identify an installed driver vib version

```
esxcli software vib list | grep [sfvmk|sfc]
sfvmk 2.2.0.1000-10EM.650.0.0.4598673 SFC VMwareCertified 2019-02-05
```

Remove an installed driver vib

```
esxcli software vib remove --vibName=sfvmk
```

To remove the earlier versions of the Solarflare driver:

```
esxcli software vib remove --vibName=net-sfc
```



**NOTE:** When a driver has been removed the ESXi host server must be rebooted.

### Install the vib through the host CLI

```
esxcli software vib install -v <absolute PATH to the .vib>
```

Installation Result

Message: The update completed successfully, but the system needs to be rebooted for the changes to be effective.

**Reboot Required: true**

VIBs Installed: SFC\_bootbank\_sfvmk\_2.2.0.1000-10EM.650.0.0.4598673

VIBs Removed:

VIBs Skipped:



**NOTE:** When a driver has been installed the ESXi host server must be rebooted.

### Install the offline bundle

```
esxcli software vib install -d <absolute PATH to the .zip>
```

### Identify installed/loaded driver module

```
esxcli system module get -m=sfvmk
```

Module: sfvmk

Module File: /usr/lib/vmware/vmkmmod/sfvmk

License: BSD

Version: 2.2.0.1000-10EM.650.0.0.4598673

Build Type: release

Provided Namespaces:

Required Namespaces: com.vmware.vmkapi@v2\_4\_0\_0

Containing VIB: sfvmk VIB Acceptance Level: certified



## To identify adapter driver and firmware versions

### 1 List adapter interfaces:

```
esxcli network nic list
```

Name	PCI Device	Driver	Admin Status	Link Status	Speed	Duplex
vmnic4	0000:04:00.0	sfvmk	Up	Up	10000	Full
MAC Address		MTU	Description			
00:0f:53:29:eb:60		1500	Solarflare SFC9220 10/40G Ethernet Controller			

### 2 Identify adapter driver/firmware/link status:

```
esxcli network nic get -n vmnic4
```

```
Advertised Auto Negotiation: false
Advertised Link Modes: 1000BaseT/Full
Auto Negotiation: false
Cable Type: DA
Current Message Level: 1
Driver Info:
  Bus Info: 0000:04:00:0
  Driver: sfvmk
  Firmware Version: 7.5.0.1016 rx0 tx0
  Version: 2.2.0.1000
Link Detected: true
Link Status: Up
Name: vmnic4
PHYAddress: 0
Pause Autonegotiate: true
Pause RX: false
Pause TX: false
Supported Ports: FIBRE, DA
Supports Auto Negotiation: false
Supports Pause: true
Supports Wakeon: false
Transceiver:
Virtual Address: 00:50:56:5b:94:f1
Wakeon: None
```



**CAUTION:** Drivers, firmware, CIM Provider versions shown in this user guide are for example purposes. Users should check the relevant release notes to ensure that installed drivers and firmware support the features required.

## 5.7 Driver Configuration

### List Adapter Driver Parameters

Run the following command to identify driver module parameters and current values of parameters configurable on the ESXi host:

```
esxcli system module parameters list -m sfvmk
```

**Table 37: Driver Module Parameters**

Diagnostic Test	Purpose
debugMask	Debug logging bit masks
evqType	Optimize driver for low-latency or throughput performance. EVQ type [0:Auto, 1: Throughput (default), 2: Low latency] Invalid value sets evqtype to the default value. Auto will select a setting based on firmware variant.
vxlanOffload	VXLAN offload [0: Disable, 1: Enable (default)]
geneveOffload	GENEVE offload [0: Disable, 1: Enable (default)]
netQCount	NetQ count (includes defQ) [Min:1 Max:15 Default:8] Invalid value sets netQCount to default value.
rssQCount	RSSQ count [Min:1 (RSS disabled) Max:4 Default: RSS disabled] Invalid value sets rssQCount disables RSS.
max_vfs	Number of VFs per PF that the driver should configure. The value should be ≥ the vf-count value on SR-IOV adapters. [Min: 0, Max: 63, Default: 0]. An invalid entry will set max_vfs to 0 (0 means SR-IOV is disabled). Must be set to a valid value for SR-IOV.

**netQCount and rssQCount values are applied to every adapter port. Ports cannot be configured independently.**

### Set Driver Parameters

#### Examples

```
esxcli system module parameters set -m sfvmk -p max_vfs=16
```

```
esxcli system module parameters set -m sfvmk --parameter-string="rssQCount=4"
```

The parameter string can also set multiple values in one command:

```
--parameter-string="rssQCount=4 netQCount=8"
```

Use the list command to view current settings. The ESXi host must be rebooted following changes to the driver module parameters.

## 5.8 Adapter Configuration

### Uplink link state

To change/get uplink state:

```
esxcli network nic [up|down] -n <uplink-interface>
```

For example:

```
esxcli network nic [up|down] -n vmnic4
```

### Adapter RX/TX ring buffer size

Use the following command to get/set adapter RX/TX ring buffers sizes:

```
esxcli network nic ring current set -n vmnic4 -r 2048 -t 1024  
esxcli network nic ring current set -n vmnic4 -r 4096  
esxcli network nic ring current get -n vmnic4  
RX: 4096 RX Mini: 0 RX Jumbo: 0 TX: 1024
```



**NOTE:** Changes are not preserved over reboot.

## 5.9 Granting access to the NIC from the Virtual Machine

Before a guest operating system has access to the Solarflare adapter, the device should be connected to a vSwitch to which the guest also has a connection.

## 5.10 NIC Teaming

A team allows two or more network adapters to be connected to a virtual switch (vSwitch). The main benefits of creating a team are:

- Increased network capacity for the virtual switch hosting the team.
- Passive failover in the event one of the adapters in the team fails.



**NOTE:** The VMware host only supports NIC teaming on a single physical switch or stacked switches.

### To create a team

- 1 From the host web client, select the **Networking** folder.
- 2 Select the required **vSwitch** under **Networking**.
- 3 **Edit** Settings on the vSwitch.
- 4 Select **NIC Teaming** from the Edit Standard virtual switch settings dialog.

The following options are configurable:

- Load Balancing
- Network Failover Detection
- Notify Switches
- Failover
- Failover Order

### Teaming - further reading

Refer to VMware documentation for additional teaming configuration information.

## 5.11 Configuring VLANs

There are three methods for creating VLANs on VMware ESXi:

- 1 Virtual Switch Tagging (VST)
- 2 External Switch Tagging (EST)
- 3 Virtual Guest Tagging (VGT)

For EST and VGT tagging, consult the documentation for the switch or for the guest OS.

### To Configure Virtual Switch Tagging (VST)

With vSwitch tagging:

- All VLAN tagging of packets is performed by the virtual switch, before leaving the VMware ESXi host.
- The host network adapters must be connected to trunk ports on the physical switch.
- The port groups connected to the virtual switch must have an appropriate VLAN ID specified.

To configure vSwitch tagging:

- 1 From the host web client, select the **Networking** folder.
- 2 Select **Port Groups** tab to list port groups.
- 3 Select a **Port Group** and click **Edit Settings**.
- 4 Enter a valid VLAN ID.
  - a) VLAN ID 0 (zero) disables VLAN tagging on the port group (EST mode).
  - b) VLAN ID 4095 enables trunking on the port group (VGT mode).

## 5.12 Performance Tuning on VMware

### Introduction

The Solarflare network adapters are designed for high-performance network applications. The adapter driver is pre-configured with default performance settings that have been designed to provide good performance across a broad class of applications.

### Install VMware Tools in the Guest Platform

Installing VMware tools will deliver greatly improved networking performance in the guest. When VMware Tools are installed, the guest will see virtual adapters of type **vmxnet3** which is a virtual adapter designed to deliver high performance with minimal I/O overheads in VMs.

To check that VMware Tools are installed:

- 1 From the host web client, select the virtual machine.
- 2 **Edit Settings > VM options > VMware Tools.**

If VMware Tools are not installed/enabled, refer to VMware documentation:

<https://kb.vmware.com/s/article/2004754>

### VMware ESXi NetQueue

Solarflare adapters support VMware's NetQueue technology, accelerating network performance in Ethernet virtualized environments. NetQueue is enabled by default in VMKernel releases.

There is usually no reason not to enable NetQueue.



**NOTE:** VMware NetQueue accelerates receive and transmit traffic.

#### NetQueue Filtering

NetQueue distributes traffic among physical queues, with each queue having its own ESX thread for packet processing and each thread represents a CPU core.

Traffic is filtered on MAC address and NetQueue will use the adapter outer MAC address meaning that all traffic having the same MAC address is directed to the same queue.

When using VXLAN, packets addressed to multiple VMs will have the same destination MAC. VXLAN also filters on the inner MAC address.

## Is NetQueue Enabled

```
esxcli system settings kernel list -o netNetqueueEnabled
```

Name	Type	Description
netNetqueueEnabled	Bool	Enable/Disable NetQueue support.
Configured	Runtime	Default
TRUE	TRUE	TRUE

## Configure Number of NetQueues

To configure the Solarflare adapter driver to use NetQueue - specify the number of queues required.

Refer to [List Adapter Driver Parameters on page 180](#) above.

## Check the current NetQueue configuration

```
esxcli network nic queue count get
NIC      Tx netqueue count  Rx netqueue count
-----
vmnic4   8                   8
vmnic5   8                   8
vmnic6   8                   8
vmnic7   8                   8
```

## Binding NetQueue queues and Virtual Machines to CPUs

NetQueue can deliver improved performance when each queue's associated interrupt and the virtual machine are pinned to the same CPU. This is particularly true when workloads with sustained high bandwidth are evenly distributed across multiple virtual machines.

To pin a Virtual Machine to one or more CPUs:

- 1 From the host web client, select the virtual machine.
- 2 **Edit Settings > Virtual Hardware > CPU.**
- 3 In the **Scheduling Affinity** box, enter a comma separated list (or hyphenated range) of CPU(s) to which the virtual machine is to be bound.

## Identify NetQueue Interrupts

Use `esxtop` to identify interrupts assigned to the Solarflare adapter. Interrupts are listed in order: the first interrupt will be for the **default** queue, the second interrupt for the queue dedicated to the first virtual machine to have been started, the third interrupt for the queue dedicated to the second virtual machine to have been started, and so on.

```
esxtop
press i
press f
toggle fields B C D (press B, then C then D then Enter key)
```

The following example lists the NetQueues and associated IRQs from `vmnic4` when four NetQueues are configured.

```
0x14   VMK vmnic4-intr0
0x15   VMK vmnic4-intr1
0x16   VMK vmnic4-intr2
0x17   VMK vmnic4-intr3
```

Toggle the `esxtop` fields ABCDEF as required for different views of interrupts and CPU usage.

### Number of VMs > number CPU

If there are more virtual machine's than CPUs on the host, optimal performance is obtained by pinning each virtual machine and its associated interrupt to the same CPU.

### Number of VMs < number CPU

If there are fewer virtual machines than CPUs, optimal results are obtained by pinning the virtual machine and associated interrupt to two different cores which share an L2 cache.



## Adapter MTU (Maximum Transmission Unit)

The default MTU of 1500 bytes ensures that the adapter is compatible with legacy Ethernet endpoints. However if a larger MTU is used, adapter throughput and CPU utilization can be improved because it takes fewer packets to send and receive the same amount of data.

Solarflare adapters support frame sizes up to 9000 bytes (this does not include the Ethernet preamble or frame-CRC).

Since the MTU should ideally be matched across all endpoints in the same LAN (VLAN), and since the LAN switch infrastructure must be able to forward such packets, the decision to deploy a larger than default MTU requires careful consideration. It is recommended that experimentation with MTU be done in a controlled test environment.

### Commands

```
esxcli network vswitch
Usage: esxcli network vswitch {cmd} [cmd options]
```

Available Namespaces:

```
dvs          Commands to retrieve Distributed Virtual Switch information
standard    Commands to list and manipulate Legacy Virtual Switches on an ESX host.
```

### Set/Change MTU

Changing the MTU on the vSwitch will also change the value on the Solarflare uplink interface(s).

```
esxcli network vswitch standard set -m <MTU size> -v <vSwitch name>
```

### Verify MTU

```
esxcli network vswitch standard list
esxcli network vswitch dvs vmware list
```

A change in MTU size on a vSwitch will persist across reboots of the VMware ESXi host.

### Check Adapter MTU

To check the MTU size on the adapter uplink:

```
esxcli network nic list
```

## Interrupt Moderation (Interrupt Coalescing)

Interrupt moderation reduces the number of interrupts generated by the adapter by combining multiple received packet events and/or transmit completion events into a single interrupt.

The interrupt moderation interval is the minimum time (microseconds) between two consecutive interrupts. Coalescing occurs only during the interval. Setting a moderation interval of zero (0) will disable interrupt moderation.

Interrupt moderation settings are **crucial for tuning adapter latency**:

- Increasing the interrupt moderation interval will:
  - generate less interrupts
  - reduce CPU utilization (because there are less interrupts to process)
  - increase latency
  - improve peak throughput.
- Decreasing the interrupt moderation interval will:
  - generate more interrupts
  - increase CPU utilization (because there are more interrupts to process)
  - decrease latency
  - reduce peak throughput.
- Turning off interrupt moderation will:
  - generate the most interrupts
  - give the highest CPU utilization
  - give the lowest latency
  - give the biggest reduction in peak throughput.

For many transaction request-response type network applications, the benefit of reduced latency to overall application performance can be considerable. Such benefits may outweigh the cost of increased CPU utilization.

- Interrupt moderation should be disabled for applications that require best latency and jitter performance, such as market data handling.
- Interrupt moderation should be enabled for high throughput single (or few) connection TCP streaming applications, such as iSCSI.



**NOTE:** The interrupt moderation interval dictates the minimum gap between two consecutive interrupts. It does not mandate a delay on the triggering of an interrupt on the reception of every packet. For example, an interval of 30μs will not delay the reception of the first packet received, but the interrupt for any following packets will be delayed until 30μs after the reception of that first packet.

## Commands

```
esxcli network nic coalesce
```

```
Usage: esxcli network nic coalesce {cmd} [cmd options]
```

Available Namespaces:

high Commands to access coalesce parameters for a NIC at high packet rate

low Commands to access coalesce parameters for a NIC at low packet rate

Available Commands:

get	Get coalesce parameters
set	Set coalesce parameters on a nic

## Set interrupt moderation

```
esxcli network nic coalesce set -t 30 -n vmnicX
```



**CAUTION:** Settings do not persist over host reboots.

## Get interrupt moderation

```
esxcli network nic coalesce get -n vmnicX
```

## Identify interrupt activity with esxstop

Refer to [Identify NetQueue Interrupts on page 186](#) above.

## Adaptive Moderation

The adaptive interrupt moderation feature is not currently supported.

## TCP/UDP Checksum Offload

Checksum offload moves calculation and verification of TCP and UDP packet checksums to the adapter. The driver by default has checksum offload features enabled.

### Commands

```
esxcli network nic cso
```

```
Usage: esxcli network nic cso {cmd} [cmd options]
```

Available Commands:

get	Get checksum offload settings
set	Set checksum offload settings on a nic

### Enable Checksum Offload

```
esxcli network nic cso set -e=1 -n=<uplink interface>
```

### Disable Checksum Offload

```
esxcli network nic cso set -e=0 -n=<uplink interface>
```

### Verify Checksum Offload

```
esxcli network nic cso get -n <uplink interface e.g. vmnic4>
```

NIC	RX Checksum Offload	TX Checksum Offload
vmnic4	on	on

When configuring checksum offload in the guest, consult the relevant Solarflare section for the guest OS, or documentation for the guest OS.

## TCP Segmentation Offload (TSO)

TCP Segmentation offload (TSO) offloads the splitting of outgoing TCP data into packets to the adapter. TCP segmentation offload benefits applications using TCP. Enabling TCP segmentation offload will reduce CPU utilization on the transmit side of a TCP connection, and so improve peak throughput, if the CPU is fully utilized.

Since TSO has no effect on latency, it can be enabled at all times. The driver has TSO enabled by default.

### Commands

```
esxcli network nic tso
```

```
Usage: esxcli network nic tso {cmd} [cmd options]
```

Available Commands:

get	Get TCP segmentation offload settings
set	Set TCP segmentation offload settings on a nic

### Enable TSO

```
esxcli network nic tso set -e=1 -n vmnic4
```

### Disable TSO

```
esxcli network nic tso set -e=0 -n vmnic4
```

### Verify TSO

```
esxcli network nic tso get -n <uplink interface e.g. vmnic4>
```

```
NIC      Value
vmnic4  on
```



**NOTE:** Non TCP protocol applications will not benefit (but will not suffer) when TSO is enabled.

## TCP Large Receive Offload (LRO)

Solarflare sfvmk does not support LRO offload.

### TSO and LRO Further Reading

Users should refer to [Understanding TCP Segmentation Offload \(TSO\) and Large Receive Offload \(LRO\) in a VMware environment](#).

## TCP Protocol Tuning

TCP Performance can also be improved by tuning kernel TCP settings. Settings include adjusting send and receive buffer sizes, connection backlog, congestion control, etc.

Typically it is sufficient to tune just the max buffer value. It defines the largest size the buffer can grow to. Suggested alternate values are max=500000 (1/2 Mbyte). Factors such as link latency, packet loss and CPU cache size all influence the affect of the max buffer size values. The minimum and default values can be left at their defaults minimum=4096 and default=87380.

When tuning the guest TCP stack consult the documentation for the guest operating system.

## Receive Side Scaling (RSS)

Solarflare adapters support Receive Side Scaling (RSS). RSS enables packet receive-processing to scale with the number of available CPU cores. RSS requires a platform that supports MSI-X interrupts. RSS is disabled by default.

When RSS is enabled the controller uses multiple receive queues into which to deliver incoming packets. The receive queue selected for an incoming packet is chosen in such a way as to ensure that packets within a TCP stream are all sent to the same receive queue – this ensures that packet-ordering within each stream is maintained.

Each receive queue has a dedicated MSI-X interrupt which ideally should be tied to a dedicated CPU core. This allows the receive side TCP processing to be distributed amongst the available CPU cores.

When VXLAN or GENEVE overlay encapsulation is enabled, RSS will distribute traffic based on inner layer 3/4 headers.

RSS can be enabled independently of NetQueue i.e. both can be enabled or either can be enabled.

### Disable RSS

- On the ESXi host (requires ESXi host reboot):

```
esxcli system module parameters set -p rssQCount=1 -m sfvmk
```

### Enable RSS

- On the ESXi host (requires ESXi host reboot):

```
esxcli system module parameters set -p rssQCount=4 -m sfvmk
```

### Identify RSS Queue Count

```
esxcli system module parameters list -m sfvmk | grep rssQCount
```

Also refer to [List Adapter Driver Parameters on page 180](#) above.

## Interrupt Balancing

Interrupt (IRQ) balancing in the hypervisor aims to distribute interrupts over available CPU cores based on CPU workload. When setting interrupt affinity to specific CPU cores it is best to disable IRQ balancing.

### Commands

```
esxcli system settings kernel
Usage: esxcli system settings kernel {cmd} [cmd options]
```

Available Commands:

list	List VMkernel kernel settings.
set	Set a VMKernel setting.

### Enable Balance

```
esxcli system settings kernel set --setting="intrBalancingEnabled" --value="TRUE"
```

### Disable Balance

```
esxcli system settings kernel set --setting="intrBalancingEnabled" --value="FALSE"
```

### Verify Balance

```
esxcli system settings kernel list | grep intrBalancingEnabled
```

## Other Considerations

### PCI Express Lane Configurations

The PCI Express (PCIe) interface used to connect the adapter to the server can function at different widths. This is independent of the physical slot size used to connect the adapter. Widths are multiples x1, x2, x4, x8 and x16 lanes:

- PCIe 1.0 (2.5 GT/s - in each direction)
- PCIe 2.0 (5.0 GT/s - in each direction)
- PCIe 3.x (8.0 GT/s - in each direction)

Solarflare Adapters are designed for x8 and x16 lane operation. When PCIe slots are only configured electrically to support x4 lanes, adapters will continue to operate, but at reduced speed.

### Memory bandwidth

Many chipsets/CPU's use multiple channels to access main system memory. Maximum memory performance is only achieved when the server can make use of all channels simultaneously. This should be taken into account when selecting the number of DIMMs to populate in the server. Consult server/motherboard documentation for details.

## Server Motherboard, Server BIOS, Chipset Drivers

Tuning or enabling other system capabilities may further enhance adapter performance. Readers should consult their server user guide. Possible opportunities include tuning PCIe memory controller (PCIe Latency Timer setting available in some BIOS versions).

## 5.13 Interface Statistics

Use the following VMkernel Sys Info Shell command to list adapter statistics:

```
vsish -e cat /net/pNics/<uplink-interface>/stats
```

e.g

```
vsish -e cat /net/pNics/vmnic4/stats
```

This command generates an extensive list of counters for RX/TX packets, dropped packet counters and per-queue counters.

## 5.14 vSwitch/VM Network Statistics

- 1 Identify the network port for the VM:

```
net-stats -l
```

PortNum	Type	SubType	SwitchName	MACAddress	ClientName
33554434	4	0	vSwitch0	b0:83:fe:e3:88:56	vmnic0
33554436	3	0	vSwitch0	b0:83:fe:e3:88:56	vmk0
33554437	5	9	vSwitch0	00:0c:29:e7:61:11	vmrhe173

- 2 List vSwitch Port Network Statistics

```
esxcli network port stats get -p 33554437
```

```
Packet statistics for port 33554437
```

```

Packets received: 3955
Packets sent: 153
Bytes received: 414779
Bytes sent: 14294
Broadcast packets received: 3829
Broadcast packets sent: 21
Multicast packets received: 4
Multicast packets sent: 8
Unicast packets received: 122
Unicast packets sent: 124
Receive packets dropped: 0
Transmit packets dropped: 0

```



### 3 Identify the PortNum being used by the Solarflare uplink-interface:

```
net-stats -l
```

PortNum	Type	SubType	SwitchName	MACAddress	ClientName
33554434	4	0	vSwitch0	b0:83:fe:e3:88:56	vmnic0
33554436	3	0	vSwitch0	b0:83:fe:e3:88:56	vmk0
33554437	5	9	vSwitch0	00:0c:29:e7:61:11	vmrhe173
50331652	3	0	vdataSW	00:50:56:61:65:f3	vmk1
<b>67108868</b>	4	0	DvsPortset-1	<b>00:0f:53:43:23:f1</b>	vmnic5

Solarflare adapter MAC addresses begin **00:0f:53**.

### 4 Port ClientStats:

```
vsish -e get /net/portsets/DvsPortset-1/ports/67108868/clientStats
```

```
port client stats {
  pktsTxOK:56363495
  bytesTxOK:3720155568
  droppedTx:0
  pktsTsoTxOK:0
  bytesTsoTxOK:0
  droppedTsoTx:0
  pktsSwTsoTx:0
  droppedSwTsoTx:0
  pktsZerocopyTxOK:16
  droppedTxExceedMTU:0
  pktsRxOK:275131192
  bytesRxOK:3999992213561
  droppedRx:4676
  pktsSwTsoRx:313
  droppedSwTsoRx:0
  actions:0
  uplinkRxPkts:2740376953
  clonedRxPkts:0
  pksBilled:0
  droppedRxDueToPageAbsent:0
  droppedTxDueToPageAbsent:0
}
```

### 5 Port Stats Summary:

```
vsish -e get /net/portsets/DvsPortset-1/ports/67108868/vmxnet3/rxSummary
```

```
stats of a vmxnet3 vNIC rx queue {
  LRO pkts rx ok:272133667
  LRO bytes rx ok:3995454566846
  pkts rx ok:275131192
  bytes rx ok:3999992213561
  unicast pkts rx ok:275131186
  unicast bytes rx ok:3999992213201
  multicast pkts rx ok:0
  multicast bytes rx ok:0
  broadcast pkts rx ok:6
  broadcast bytes rx ok:360
  running out of buffers:711
  pkts receive error:0
  1st ring size:256
  2nd ring size:128
  # of times the 1st ring is full:398
  # of times the 2nd ring is full:313
  fail to map a rx buffer:0
}
```

```

request to page in a buffer:0
# of times rx queue is stopped:0
failed when copying into the guest buffer:0
# of pkts dropped due to large hdrs:0
# of pkts dropped due to max number of SG limits:0
pkts rx via data ring ok:0
bytes rx via data ring ok:0
Whether rx burst queuing is enabled:0
current backend burst queue length:0
maximum backend burst queue length so far:0
aggregate number of times packets are requeued:0
aggregate number of times packets are dropped by PktAgingList:0

```

## 6 VM Receive Queues

```

vsish -e ls /net/portsets/DvsPortset-1/ports/67108868/vmxnet3/rxqueues/
0/
1/
2/
3/
4/
5/
6/
7/

```

= 8 Linux VM queues being used by the guest and each has its own RX queue.

## 7 Per Receive-Queue Statistics

Per-receive queue stats are available for each VM receive queue:

```

vsish -e get /net/portsets/DvsPortset-1/ports/67108868/vmxnet3/
rxqueues/<rxqueue-number>/stats

```

Identify the PortNum, in this example it is 67108868, being used by the Solarflare uplink-interface using the following command:

```

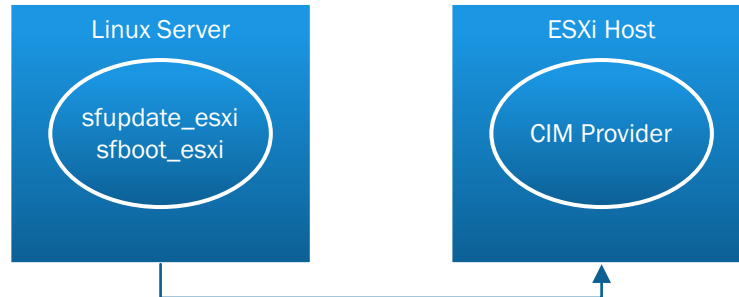
net-stats -l

```

## 5.15 CIM Provider

The Solarflare Common Information Model (CIM) Provider package is available as a VIB for installation on the ESXi host.

The CIM Provider allows remote access to the ESXi host using CIM transport.



### Install CIM Provider

- 1 Remove any existing installed Solarflare CIM .vib

```
esxcli software vib list | grep solarflare
solarflare-cim-provider 2.1-0.19 SLF VMwareAccepted 2019-02-05
esxcli software vib remove --vibName=solarflare-cim-provider
```

- 2 Copy the CIM VIB package, SF-120055-LS, to a directory on the ESXi host.

- 3 Install the .vib

```
esxcli software vib install -v /vmfs/volumes/datastore1/solarflare/
SFC-ESX-solarflare-cim-provider-2.1-0.19.vib
```

Installation Result

Message: The update completed successfully, but the system needs to be rebooted for the changes to be effective.

**Reboot Required: true**

VIBs Installed: SFC\_bootbank\_solarflare-cim-provider\_2.1-0.19

VIBs Removed:

VIBs Skipped:



**NOTE:** When a vib has been installed the ESXi host server must be rebooted.

### Verify CIM Provider

```
esxcli system wbem provider list
Name                Enabled  Loaded
vmw_solarflare-cim-provider  true   true
sfc_base            true    true
```

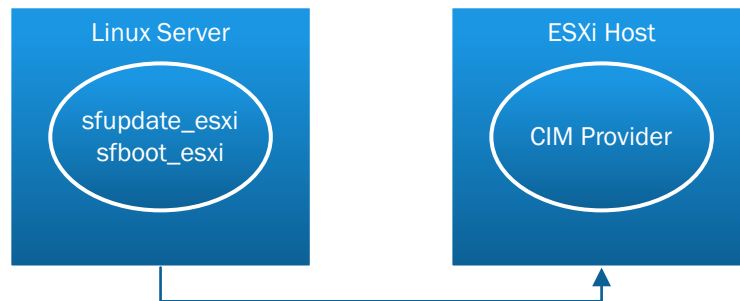
## 5.16 Adapter Firmware Upgrade - sfupdate\_esxi

Adapter firmware can be upgraded using any of the following methods:

- Using the *VCP web plugin* from a VCSA - Refer to [vSphere Client Plugin on page 212](#).
- Locally using the `esxcli` extension firmware command - Refer to [Firmware Images VIB on page 205](#).
- Remotely using `sfupdate_esxi` connected to the CIM Provider.

### sfupdate\_esxi

The CIM Provider must be installed on the ESXi host. The `sfupdate_esxi` utility connects to the CIM Provider from a remote Linux server.



**NOTE:** `sfupdate_esxi` is not provided past CIM version 2.1-0.19. When later CIM versions are installed, firmware upgrade is done with the Solarflare `esxcli` extension commands or by using the VCP web plugin.

See [Firmware Upgrade Examples on page 200](#).

## sfupdate\_esxi Options

Enter the following command to display available options:

```
./sfupdate_esxi_<version> -?
```

**Table 38: sfupdate\_esxi Options**

Option	Description
-h, --help	Display options and usage
-a, --cim-address=STRING	Address of CIM Server e.g. "https://hostname:5989" "hostname " "hostname:5988" Default protocol is HTTP, default port for HTTP is 5988, default port for HTTPS is 5989
-s, --https	Use HTTPS to access CIM Server (has no effect if you specified protocol in --cim-address parameter)
-u, --cim-user=STRING	CIM Server user name
-p, --cim-password=STRING	CIM Server user password
-n, --cim-namespace=STRING	CIM Provider namespace (solarflare/cimv2 by default)
-i, --interface-name=STRING	Interface name (if not specified, firmware for all interfaces would be processed)
--controller	Process Controller firmware
--bootrom	Process BootROM firmware
--uefirom	Process UEFIROM firmware
--sucfw	Process support microprocessor (SUC) adapter firmware
-y, --yes	Do not ask for confirmation before updating firmware
-w, --write	Perform firmware update
--force	Force update of the firmware even if the version of the image is lower than or the same as they image already installed - see examples below. If this option is required, but not on the sfupdate command line, the following is displayed: "won't be applied without --force"
--firmware-url=STRING	URL of firmware image(s) to be used instead of the image included with this version of sfupdate_esxi. Currently support FTP and TFTP - see examples below.
--firmware-path=STRING	Path to firmware image(s) to be used instead of the image included with this version of sfupdate_esxi
--firmware-url-no-local	Do not try to access firmware images from an URL specified from this tool, just pass URL to CIM provider. Version checks will be disabled; --fw-url-use-cim-transfer cannot be used together with this option
--firmware-url-cim-transfer	Do not pass firmware URL to CIM provider but transfer downloaded firmware images via private CIM methods. Useful when there are issues with ESXi firewall or if URL specified is not available on the ESXi target host

## Firmware Upgrade Examples

```
./sfupdate_esxi_v2.1.0.17 --cim-address="https://servername:5989" --write
```

The user will be prompted for the user password.

```
vmnic4 - MAC: 000f53644f10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
SUCFW version: 2.1.1.1003
    Available update: 2.1.1.1001 (won't be applied without --force)
UEFIROM version: 2.7.8.5
    Available update: 2.7.5.0 (won't be applied without --force)
BootROM version: 5.2.1.1000
    Available update: 5.2.0.1004 (won't be applied without --force)
Controller version: 7.5.0.1016 rx0 tx0
    Available update: 7.5.0.1009 (won't be applied without --force)
```

\*

```
./sfupdate_esxi_v2.1.0.17
--cim-address="servername" --https --cim-user=<user>
--cim-password=<password> -i vmnic5 --write
```

```
vmnic5 - MAC: 000f534323f1
NIC model: Solarflare Flareon Ultra 8000 Series 10G Adapter
SUCFW Not Applicable
UEFIROM version: 2.4.4.8
    Available update: 2.4.4.8 (won't be applied without --force)
BootROM version: 5.0.5.1002
    Available update: 5.0.5.1002 (won't be applied without --force)
Controller version: 6.5.1.1023 rx0 tx0
    Available update: 6.5.2.1000
```

Do you want to update Controller firmware on vmnic5? [yes/no]

\*

### Upgrade firmware using FTP (example)

```
./sfupdate_esxi_2.1.0.17
--cim-address="https://root@10.40.128.17:5989"
--interface-name=vmnic2 --controller --write --force
--firmware-url="ftp://Guest:guest123@10.40.30.20/mcfw.dat"
```

### Upgrade firmware using TFTP (example)

```
./sfupdate_esxi_2.1.0.17
--cim-address="https://root@10.40.128.17:5989"
--interface-name=vmnic2 --controller --write --force --yes
--firmware-url=tftp://10.40.128.230/mcfw.dat
```

## 5.17 Adapter Configuration - sfboot\_esxi

sfboot\_esxi is a command line utility for configuring Solarflare adapter Boot Manager options including PXE and UEFI booting. sfboot\_esxi is an alternative to using the Ctrl+B to access the bootROM agent during server restart.

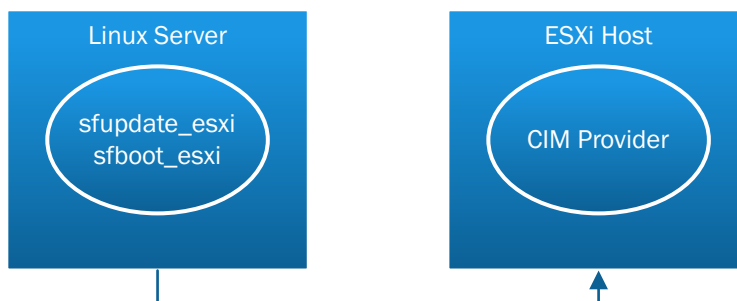
Adapters on the ESXi host can be configured remotely using sfboot\_esxi connected to the CIM Provider.

### sfboot\_esxi

The CIM Provider must be installed on the ESXi host. The sfboot\_esxi utility connects to the CIM Provider from a remote Linux server.

This feature requires:

- Solarflare boot configuration utility [v7.6.0] or later
- Solarflare-CIM-provider [2.1.0.19] or later
- Solarflare sfvmk driver [2.2.0.1000] or later



### sfboot\_esxi Options

```
# ./sfboot_esxi -h
```

For more information about sfboot\_esxi options, refer to [Sfboot: Command Usage on page 78](#).

## Using sfboot\_esxi

### Usage

```
# ./sfboot_esxi -i <interface> -a "https://fully qualified server domain
name:5989" -u root -p <root password>
```

### Example

```
# ./sfboot_esxi -i vmnic6 -a "https://mserv1.companydomaincom.com:5989"
-u root -p tester
```

Solarflare boot configuration utility [v7.6.0]  
Copyright Solarflare Communications 2006-2018, Level 5 Networks 2002-2005

```
vmnic6:
  Boot image                               Option ROM and UEFI
  Link speed                               Negotiated automatically
  Link-up delay time                       5 seconds
  Banner delay time                        2 seconds
  Boot skip delay time                     5 seconds
  Boot type                                 PXE
  Physical Functions on this port          1
  PF MSI-X interrupt limit                 32
  Virtual Functions on each PF             0
  VF MSI-X interrupt limit                 8
  Port mode                                Default
  Firmware variant                         Auto
  Insecure filters                         Default
  MAC spoofing                             Default
  Change MAC                               Default
  VLAN tags                                None
  Switch mode                              Default
  RX descriptor cache size                 32
  TX descriptor cache size                 16
  Total number of VIs                      2048
  Event merge timeout                      1500 nanoseconds
```



**NOTE:** A ESXi host server cold reboot is required after changes with sfboot\_esxi.

### Example for SR-IOV

The following command is entered on a single line:

```
./sfboot_esxi -i vmnic4 -a "https://server1.mycompanycom.com:5989" -u root
-p tester switch-mode=sriov pf-count=1 vf-count=4 firmware-variant=full-
feature
```



## 5.18 ESXCLI Extension

Solarflare provide extensions to the VMware esxcli command line interface.

### Install

Solarflare esxcli extensions are supplied as a VIB package - see [Distribution Packages on page 175](#) above.

```
esxcli software vib install -v <absolute PATH to the .vib>
```

### Identify installed package

```
esxcli software vib list | grep sfvmk
```

```
sfc-esx-sfvmkcli 2.2.0.1000-05 SFC PartnerSupported 2019-02-05
```

The esxcli command will confirm that the Solarflare extensions commands are present:

```
esxcli | grep sfvmk
```

```
sfvmk SFVMK esxcli functionality
```

### List extensions commands

```
esxcli sfvmk
```

Usage: esxcli sfvmk {cmd} [cmd options]

Available Namespaces:

```
fec      esxcli extension to get/ set FEC mode settings
firmware esxcli extension to get firmware version and update
         firmware image
mclog    esxcli extension to get/ set the MC logging enable state
sensor   esxcli extension to get hardware sensor information
stats    esxcli extension to get hardware queue statistics
vpd      esxcli extension to get VPD information
```

## Configuring FEC

For information about FEC, see [Forward Error Correction on page 42](#).

### Identify current FEC setting

```
esxcli sfvmk fec get -n vmnic4
```

```
FEC parameters for vmnic4:
Configured FEC encodings: None
Active FEC encoding: None
```

### Set/Change FEC

```
esxcli sfvmk fec set -n vmnic4
```

Usage: `esxcli sfvmk fec set [cmd options]`

Description:

<code>set</code>	Sets FEC mode settings
------------------	------------------------

Cmd options:

<code>-m --mode=&lt;str&gt;</code>	FEC mode (auto off rs baser [, ...]). (required)
<code>-n --nic-name=&lt;str&gt;</code>	The name of the NIC to configured. (required)

For example, to set the FEC parameters for vmnic4 so it uses auto mode:

```
esxcli sfvmk fec set -n vmnic4 -m auto
```

Or to set the FEC parameters for vmnic4 so it will try to use RS, and if this fails it will fallback to use BASER:

```
esxcli sfvmk fec set -n vmnic4 -m rs,baser
```



**NOTE:** FEC configuration is non-persistent.

## Firmware Images VIB

With the firmware images VIB installed on the ESXi host, adapter firmware can be updated using the esxcli firmware command **-d|--default** option.

### Firmware Components

A firmware images VIB contains all adapter firmware components for all Solarflare adapters:

- Controller (MAC controller) firmware
- BootROM firmware
- UEFI firmware
- SUC (support microprocessor controller) firmware

Installing the images VIB installs the firmware components on the ESXi host.

### Identify installed firmware images VIB

```
esxcli software vib list | grep fw
sfc-fw-images 7.5.0-1019 Solarflare PartnerSupported 2019-02-06
```

### Install the firmware images VIB

```
esxcli software vib install -v /<absolute path to the vib>/fw_images.vib
[--no-sig-check]
```

#### Installation Result

```
Message: Operation finished successfully.
Reboot Required: false
VIBs Installed: Solarflare_bootbank_sfc-fw-images_7.5.0-1019
VIBs Removed:
VIBs Skipped:
```

A firmware image file (.dat) can also be copied to the ESXi host and firmware installed from this file using the esxcli firmware extension with **-f|--file-name** option.

## Firmware Update Options

### set

Sets new firmware image. Either the **-d** or **-f** option must be specified.



**NOTE:** esxcli does not allow interactive cli. The progress of firmware update is not visible to the user and output is only displayed when all images are successfully updated or there is a failure. It may take a few minutes (or more when there are multiple adapters in the server) before the operation completes.

### get

Display current firmware versions.

Option	Description
-d --default	This option assumes a firmware images VIB is installed. Used without other options, this will update all firmware, on all Solarflare NICs from the firmware VIB. <b>Use with:</b> -n to update all firmware on a particular NIC. -t to update a specific firmware type. -w to overwrite the existing firmware image even if the firmware image in the VIB is the same as the firmware image on the adapter.
-f --file-name=<str>	Update a specific firmware image from a firmware image file. <b>Use with:</b> -t will compare the specified firmware image type [controller suc bootrom uefirom] against the file image type and <b>will FAIL if types do not match.</b> -n is mandatory.
-n --nic-name=<str>	The name of the NIC to configure. NIC name is mandatory with <b>-f --file-name</b> option.
-w --overwrite	Overwrites firmware image even if firmware image version being updated is same as the firmware image on the NIC. This is applicable only with <b>-d --default</b> option.
-t --type=<str>	Firmware image type [controller bootrom uefirom suc bundle]. <b>Use with:</b> -d it specifies the firmware image type to be updated. -f it will compare the specified firmware image type against the file image type and <b>will FAIL if types do not match.</b>

## Identify current adapter firmware

```
esxcli sfvmk firmware get [-n <interface e.g. vmnic6>]
```

```
esxcli sfvmk firmware get -n vmnic6
```

```
vmnic6 - MAC: 00:0f:53:64:4f:10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
  Controller version: 7.6.1.1009 rx0 tx0
  BOOTROM version: 5.2.1.1000
  UEFIROM version: 2.8.6.12
  SUC version: 2.1.1.1003
  BUNDLE version: 7.6.7.1001
```

## Update firmware -t -d

Specify the type of firmware [controller|suc|bootrom|uefirom]. The image is taken from the default firmware VIB on the host.

```
esxcli sfvmk firmware set -n vmnic6 -t=suc -d
```

```
vmnic6 - MAC: 00:0f:53:64:4f:10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
Previous firmware version:
  SUC version:          2.1.0.0001
Updated firmware successfully for vmnic6 vmnic7...
Current firmware version:
  SUC version:          2.1.1.1003
```

## Update firmware -t -f

Specify the type of firmware [controller|suc|bootrom|uefirom]. The image is taken from the specified firmware image file which must be present on the host.

This will compare the specified type with the image file type and **will fail if types do not match**.

```
esxcli sfvmk firmware set -n vmnic4 -t=bootrom -f=<absolute path to the
image file>/BOOTROM_2_6_v5.1.0.1005.dat
```

```
vmnic4 - MAC: 00:0f:53:43:25:40
Solarflare Flareon Ultra 8000 Series 10G Adapter
NIC model: Solarflare Flareon Ultra 8000 Series 10G Adapter
BOOTROM version: 5.0.7.1000
Updating firmware...
Updating bootrom firmware for vmnic4...
Firmware was successfully updated!
```

## Update firmware bundle

Will update all firmware components using the bundle firmware from the default location of the firmware vib installed on the host.

```
esxcli sfvmk firmware set -n vmnic6 -t=bundle -d
```

## Update firmware -d

Updates all firmware components on all adapters from the default firmware vib on the host. Can also be used with -n to specify a single adapter.

```
esxcli sfvmk firmware set -n vmnic6 -d
```

```
vmnic6 - MAC: 00:0f:53:64:4f:10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
Previous firmware version:
  SUC version:      2.1.0.1007
Updated firmware successfully for vmnic6 vmnic7...
Current firmware version:
  SUC version:      2.1.1.1003
```

```
vmnic6 - MAC: 00:0f:53:64:4f:10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
Previous firmware version:
  UEFI version:     2.7.2.10
Updated firmware successfully for vmnic6 vmnic7...
Current firmware version:
  UEFI version:     2.7.8.5
```

```
vmnic6 - MAC: 00:0f:53:64:4f:10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
Previous firmware version:
  BOOTROM version:  5.2.0.1004
Updated firmware successfully for vmnic6 vmnic7...
Current firmware version:
  BOOTROM version:  5.2.1.1000
```

```
vmnic6 - MAC: 00:0f:53:64:4f:10
NIC model: Solarflare XtremeScale X2522 10/25GbE Adapter
Previous firmware version:
  Controller version: 7.4.0.1021 rx0 tx0
Updated firmware successfully for vmnic6 vmnic7...
Current firmware version:
  Controller version: 7.5.0.1016 rx0 tx0
```



**NOTE:** esxcli does not allow interactive cli. The progress of firmware update is not visible to the user and output is only displayed when all images are successfully updated or there is a failure. It may take a few minutes (or more when there are multiple adapters in the server) before the operation completes.

## Stats

The sfvmk extensions **stats** option will generate an extensive list of stats for all network packets types sent/received via the adapter. The generated stats list includes:

- per network packet type counters
- per network packet length counters
- categorized errors packets counters
- categorized dropped packet counters
- virtual adapter packet counters
- FEC corrections
- per NetQueue transmitted/received packet counters.

To retrieve NIC stats:

```
esxcli sfvmk stats get -n <interface e.g.vmnic4>
```

## Sensors

The **sensors** option, supported from sfvmk extensions version 2.2.0.0014, displays power and temperature readings from Solarflare adapter sensors.

```
esxcli sfvmk sensor get -n vmnic4
```

Sensor Name	Warn Min.	Warn Max.	Fatal Min.	Fatal Max.	Read Value	Sensor State
0.9v power current: mA	0	8500	0	9500	4264	OK
1.2v power current: mA	0	3500	0	5000	1914	OK
0.9v power voltage (at ADC): mV	500	1100	400	1150	1060	OK
ambient temperature: degC	0	75	0	85	35	OK
Port 0 PHY power switch over-current: bool	--	--	--	--	--	OK
Controller die temperature (TDIODE): degC	0	90	0	100	43	OK
Board temperature (back): degC	0	75	0	85	31	OK

## MLog (MCDI logging)



**CAUTION:** MCDI logging should be used for debugging purposes only and should be enabled only when advised by Solarflare customer support.

The **mlog** option enables logging of MCDI messages between adapter driver and adapter firmware.

Usage: `esxcli sfvmk mlog {cmd} [cmd options]`

Available Commands:

<code>get</code>	Gets MC logging state
<code>set</code>	Sets MC logging state to enable/ disable

Description:

<code>set</code>	Sets MC logging state to enable/ disable
------------------	--

Cmd options:

<code>-e --enable</code>	Enable/ Disable MC logging (y[es], n[o]) (required)
<code>-n --nic-name=&lt;str&gt;</code>	The name of the NIC to configured. (required)

### Enable MCDI logging

```
esxcli sfvmk mlog set -e Y -n <interface>
Enabled
```

### Disable MCDI logging

```
esxcli sfvmk mlog set -e N -n <interface>
Disabled
```

## VPD

The Vital Product Data option identifies the adapter product range, model and serial number data.

```
esxcli sfvmk vpd get -n <interface>
```

```
Product Name: Solarflare Flareon Ultra 8000 Series 10G Adapter
[PN] Part number: SFN8522
[SN] Serial number: 852200201000161724100387
[EC] Engineering changes: PCBR2:CCSA2
[VD] Version: [missing]
```



## ESXCLI extensions via SSH

As with all esxcli commands, ESXCLI extensions commands can be invoked remotely using SSH from a remote Linux server when SSH is enabled on the ESXi host. The following are example command formats:

```
# ssh <esxi-host-server> esxcli sfvmk fec get -n vmnic4
```

```
# ssh <esxi-host-server> esxcli sfvmk firmware set -d -n vmnic6
```

```
# ssh <esxi-host-server> esxcli sfvmk sensor get -n vmnic4
```

When using SSH to upgrade firmware using the **-f** option, the path to the file is the path on the esxi host server.

## ESXCLI extensions via vCLI

With the VMware vCLI package installed on a remote Linux or Windows platform, esxcli commands can be run from the remote machine. Refer to VMware documentation for further vCLI information.

## 5.19 vSphere Client Plugin

The Solarflare VCP is a HTML5 vSphere client plugin for ESXi 6.5 (and later) allowing the user to manage Solarflare adapters via a web browser connecting to a vCenter Server Appliance hosted on an ESXi platform.

The VCP provides a graphical frontend, via vSphere, to manage Solarflare CIM objects i.e. adapter and driver. The plugin can be installed on a 64bit Windows Server 2012 R2 or Windows Server 2016.

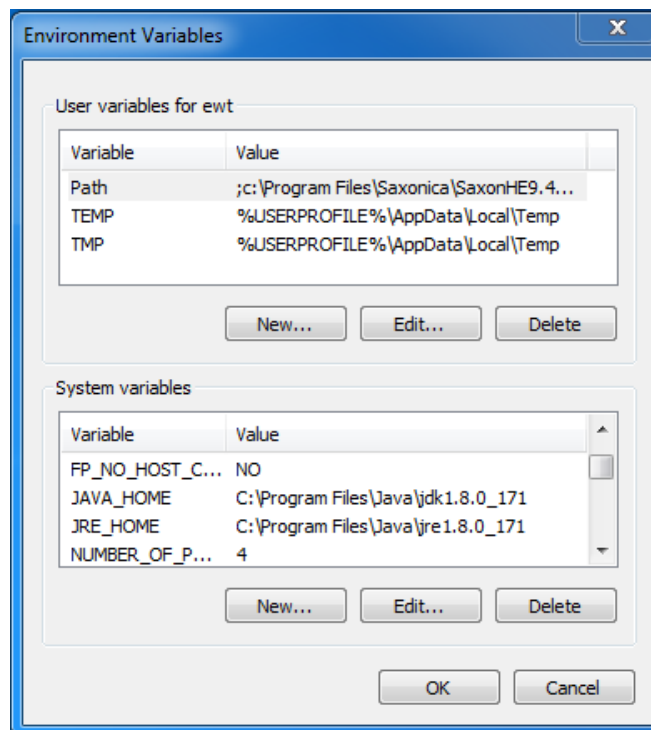
### Requirements



**NOTE:** The Solarflare CIM Provider vib must be installed on all ESXi hosts that will be managed via the vSphere Client Plugin.

The machine from which the plugin installer.msi will run must have the correct value for the **JRE\_HOME** environment variable. The Java version must be 1.8 or later.

Check settings from the Windows **Control Panel > System Properties > Environment Variables** tab:

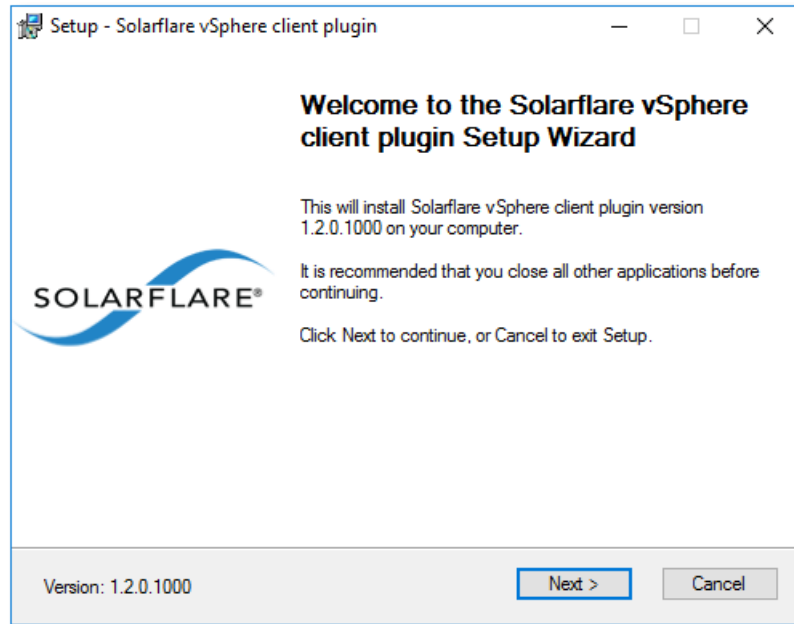


The installer includes all components including Apache Tomcat which is installed on the local server. Changes will be made to the local machine system registry.

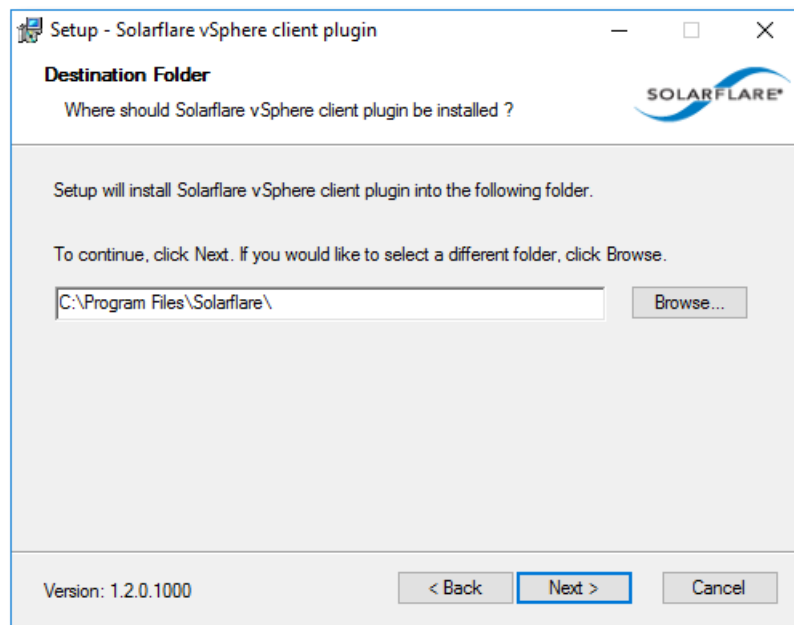
## Plugin Installation

Copy the Solarflare vSphere client plugin Windows Installer package SF-120056-LS to the Windows machine where it will be installed.

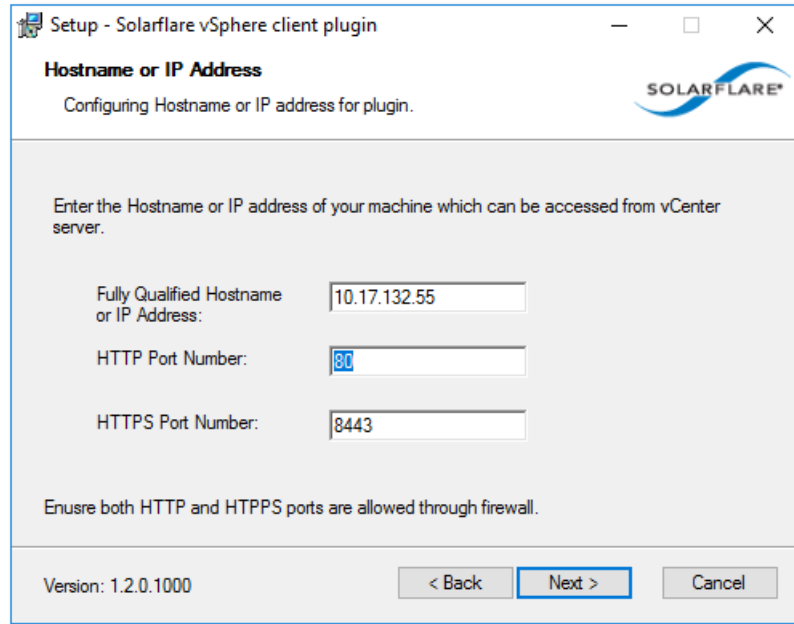
In the *Solarflare VCP-Windows Installer* directory, right click the .msi installer.



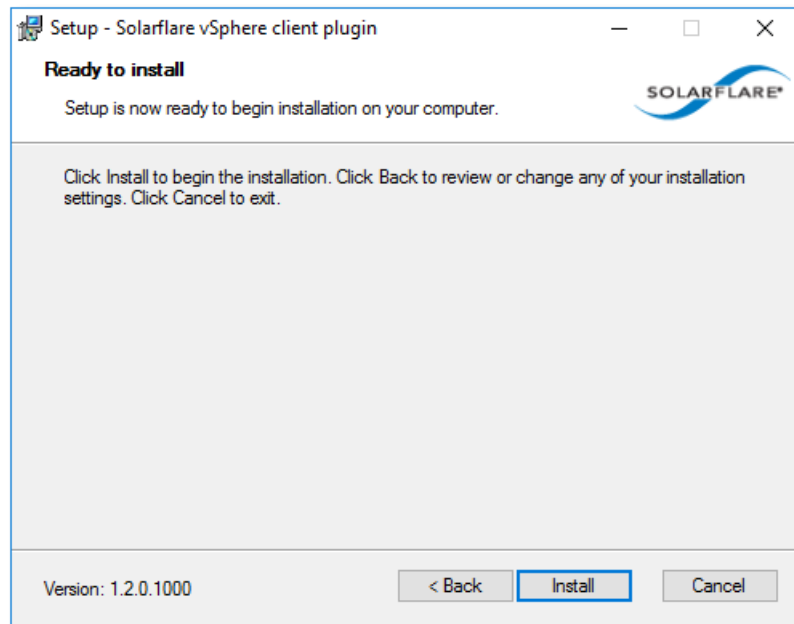
Click **Next**. The Solarflare directory will be created on the local machine. Change the install directory if required and select **Next** to continue.



Enter the hostname or IP address of the local Windows server.

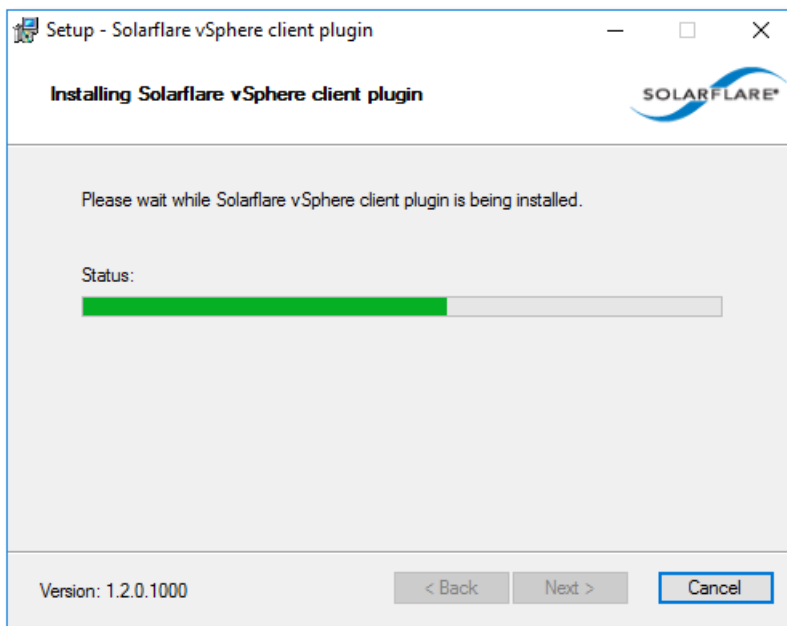


Click **Next** to launch the installation.



Click **Install** to begin installation.

If installation fails - refer to [Plugin Install TroubleShoot on page 219](#).



When installation is complete, the installer will launch the **Plugin Registration** window to register the plugin with the VCSA.

If a plugin is already installed, registration will first prompt the user to unregister the existing plugin.

### Plugin Registration

<b>vCenter Server (VCSA)</b>	<input type="text" value="vCenter server Host Name or IP a"/>
<b>Port</b>	<input type="text" value="443"/> <span style="float: right;">▲▼</span>
<b>User Name</b>	<input type="text" value="vCenter Username"/>
<b>Password</b>	<input type="password" value="vCenter password"/>

Register
Unregister
Cancel

Enter the name of the VCSA and the VCSA user name and password used when the VCSA was created. Select the **Register** button to complete the installation.

A banner message will display the results of the registration process.

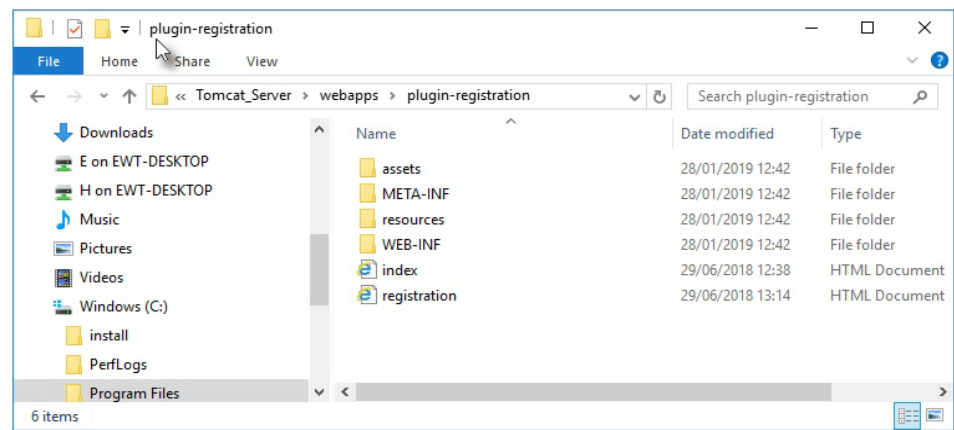
## Verify Installation on the local machine

When the install is complete there will be a Solarflare directory created in the specified location on the local machine, (C:\Program Files\Solarflare by default).

## Register Later

The VCP plugin can be registered with the VCSA at anytime after the plugin has been installed on the local Windows server. Select the registration.html from:

C:\Program Files\Solarflare\Tomcat\_Server\webapps\plugin-registration



## Unregister VCP

To unregister the VCP plugin from the VCSA, launch the **Plugin Registration** dialog window by selecting the registration.html file (see [Register Later](#) above) and select the **Unregister** control.

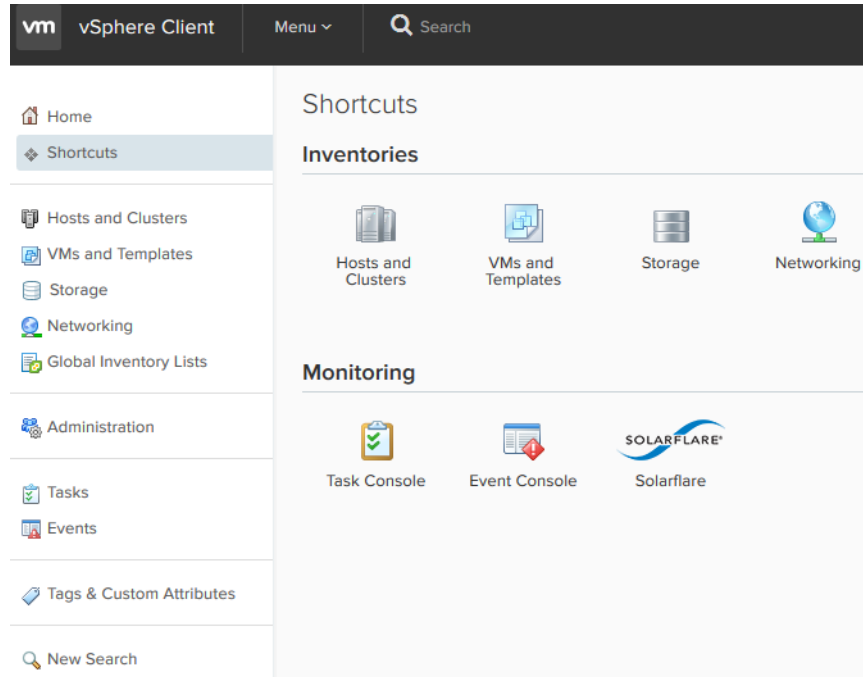
## Uninstall VCP

Unregister the VCP plugin from the VCSA before uninstalling the **Solarflare vSphere client plugin installer** via the **Control Panel > Programs > Uninstall a program**.

When uninstalled the VCP components will not be present under the C:\Program Files\Solarflare\Tomcat\_Server\webapps directory.

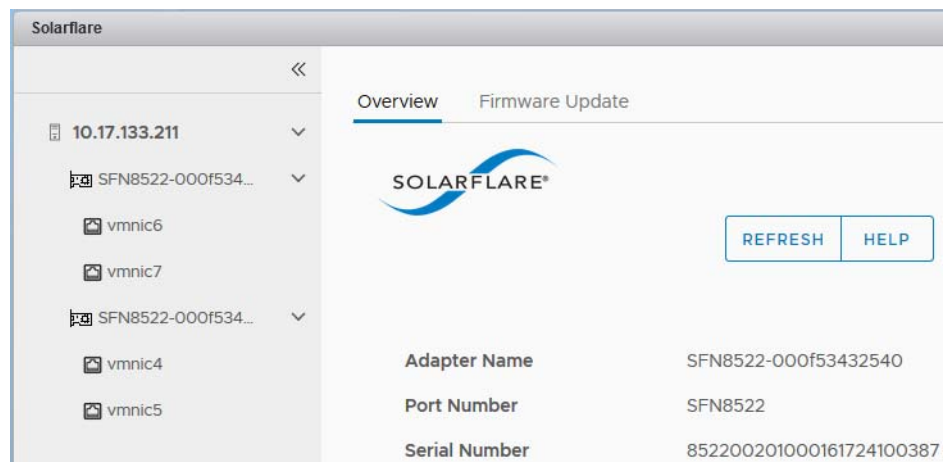
## Verify Plugin on the VCSA

When the plugin has been registered with the VCSA, the Solarflare plugin icon will be visible on the **vSphere Client, Menu > Shortcuts** page.



Click the Solarflare icon to load and display the Solarflare plugin.

Select a Solarflare adapter from the adapter list to display configuration menus.



Before hosts or Solarflare adapters are visible in the left pane, a host(s) must be added to the VCSA under a new or existing datacenter. When a datacenter is created, host(s) can be added under this datacenter. Once the host is added, the VMs and Solarflare adapters on the host should be visible to the VCP.

## Plugin - Configuration Menus

### Host view

- *Overview*
  - Identify the number of adapters present
  - Identify adapter driver version
  - Identify CIM Provider version
  - Identify Solarflare esxcli extensions version
- *Firmware Update*
  - Identify current firmware versions [controller | boot ROM | uefi ROM].
  - Supports per-adapter firmware update
- *Configuration*
  - NetQueue Count
  - RSS Queue Count
  - Driver debug mask
  - Enable/disable VXLAN/GENEVE overlay offload

### Adapter view

- *Overview*
  - Identify adapter model
  - Identify adapter serial number
- *Firmware Update*
  - Identify current firmware version and firmware version available for update [controller | boot ROM | uefi ROM].

### Interface view (vmnic)

- *Overview*
  - Port driver/link status/port speed + hardware vendor information + PCI address
- *Statistics*
  - Timed period stats for packets/bytes sent/received
  - Drop packet count
  - Total multicast/broadcast packets
  - Total receive/transmit error packets



## Plugin Install Troubleshoot

### When plugin install fails before registration.

If installation fails to complete, generate the installer log file by running the following command from Windows PowerShell or command line, in the directory where the installer.msi resides.

```
msiexec /i Solarflare_VCP_<version>_Installer.msi /l*v  
installer_<version>_log.txt
```

e.g:

```
msiexec /i Solarflare_VCP_1.0.5.0_Installer.msi /l*v  
installer_<version>_log.txt
```

(the command should be presented on a single line).

Return [1] the install log file and [2] the failure message/screenshot from the installer by email to support@solarflare.com.

### When plugin install fails during registration

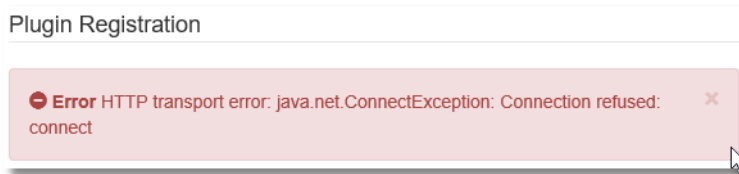
Plugin Information	
Plugin Name	Solarflare
Plugin Key	com.solarflare.vcp
Version	1.147.0
Summary	Solarflare vSphere Client plugin
Plugin URL	<a href="https://10.101.10.132:443/solarflare-vcp/solarflare-1.147.0.zip">https://10.101.10.132:443/solarflare-vcp/solarflare-1.147.0.zip</a>
SSL Thumbprint	D3:4E:92:70:9B:B3:71:FA:C2:3B:DD:E5:91:06:2B:3A:C7:2B:49:27

Check the Plugin URL is valid and accessible from vCenter Server Appliance.

### When plugin registration page does not open in the browser.

- This can occur if the Apache Tomcat server is not properly installed or encountered a problem with a previously running service on the same port.
- Check the default browser is working correctly.
- Ensure ports are free and open in the firewall.

## When registration fails due to URL - HTTPS connect



If registration cannot complete because there is a problem connecting from the local machine URL, reinstall the client plugin on the local machine and specify the local Windows machine by IP address and not by hostname@domain.

Also confirm that the local Windows server can ping the VCSA IP address and ESXi host machine IP address.

## When registration succeeds

Check the correct plugin key (com.solarflare.vcp) and version are present in the vSphere MOB.

Login to the MOB browser:

```
https://<vCenter hostname or IP>/mob
```

The **com.solarflare.vcp** key should be present in the extensionList:

```
extensionList["com.solarflare.vcp"]
```

Select the Solarflare extension list entry to display the Properties window showing the plugin key version.

## Cannot see Shortcuts menu

If the shortcuts menu is not visible after logging in via the 'vSphere Client (HTML5) - partial functionality' link, try opening with the 'vSphere Web Client (Flash)' link.

## Cannot see Solarflare plugin icon.

### Restart UI

If the Solarflare plugin icon is not visible after installing and completing a successful install and plugin registration, try the following procedure to stop/restart the vsphere user interface.

- 1 Login to the server hosting the VCSA.
- 2 From the Navigator plane, select the VCSA - make sure the VCSA is powered on.
- 3 From the VCSA Actions menu - open a browser console.
- 4 Login at the console with root access and password.

- 5 Enter the following commands at the Command prompt:  
service-control --status vsphere-ui  
service-control --stop vsphere-ui  
service-control --start vsphere-ui
- 6 Logout and login to VCSA vSphere Client from a web browser:  
https://<vcsa name>:5480
- 7 Check if the Solarflare plugin icon is visible under **Menu > Shortcuts** page.

### **VCP Installer log**

If a UI restart does not display the plugin icon, collect the vsphere-ui log from the VCSA command console:

At the console Command prompt enter 'shell' to access the VCSA file system.

```
Command> shell
```

Navigate to the following log:

```
/var/log/vmware/vsphere-ui/logs/vsphere_client_virgo.log
```

Return the log to solarflare support.

### **Unable to load left menu pane after Refresh in PluginLandingPage**

This is a known VMware issue. Return to the **Menu > Shortcuts** drop-down menu and reselect the Solarflare icon.

### **Error when fetching data after browser inactivity**

Following a long period of inactivity it maybe necessary to refresh the browser or logout/login to vCenter to update adapter information.

## 5.20 Fault Reporting - Diagnostics

sfreport is a command line utility generating a diagnostic log file identifying configuration and statistical data from the VMware host server and installed Solarflare adapters. The Solarflare VMkernel driver (sfvmk) must be installed on the host server.



**NOTE:** It is advisable to include the sfreport log when reporting issues to Solarflare support.

### Run sfreport on local host

Download the document package SF-120088-LS from: [support.solarflare.com](http://support.solarflare.com).

Copy the sfreport.py file to a directory on the host server.

To prevent file deletion when the host is rebooted, the file should be copied to a directory created by the user in any host datastore under /vmfs e.g

/vmfs/volumes/datastore1/solarflare

Run the sfreport from the esxcli and return the generated HTML file to [support@solarflare.com](mailto:support@solarflare.com).

```
python sfreport.py
```

```
sfreport version: v0.1.0
Solarflare Adapters detected..
Please be patient.
SolarFlare system report generation is in progress...
Generated output file: sfreport-2018-03-13-14-15-51.html
```

### Run sfreport from remote host

sfreport can also be run from a remote server meeting the following requirements:

- A server running the vSphere Management Assistant (VMA) which has vCLI and is compatible with ESXi6.5. The target host must be reachable from the VMA host.
- A server with vCLI installed and able to reach the target server.

## 5.21 Network Core Dump

The native driver network core dump feature allows a core dump file to be transferred to a vCenter Server Appliance following a panic of the host.

Configure in the host for each interface:

```
esxcli system coredump network set --interface-name <vmnicN> --server-ip
<vcsa-server-ip-address>
```

```
esxcli system coredump network set -e 1
```

## 5.22 Adapter Diagnostic Selftest

- 1 Identify the Solarflare Adapter uplink(s):

```
esxcli network nic list
```

```
vmnic4 0000:82:00.0 sfvmk Up Up 10000 Full
00:0f:53:43:23:f0 1500 Solarflare SFC 9220 Ethernet Controller
```

```
vmnic5 0000:82:00.1 sfvmk Up Up 10000 Full
00:0f:53:43:23:f1 1500 Solarflare SFC 9220 Ethernet Controller
```

- 2 Run the adapter diagnostics:

```
esxcli network nic selftest run -n vmnic4
```

```
Item      Value
-----  -
Result    Failed
Info      Phy Test : PASSED
Info      Register Test : PASSED
Info      Memory Test : PASSED
```



**NOTE:** The erroneous 'Failed' result is a known VMware esxcli issue resolved in ESXi 6.7.

# 6

## SR-IOV Virtualization Using KVM

### 6.1 Introduction

This chapter describes SR-IOV and virtualization using Linux KVM and Solarflare adapters.

SR-IOV enabled on Solarflare adapters provides accelerated cut-through performance and is fully compatible with hypervisor based services and management tools.

- PCIe Virtual Functions (VF).

A PCIe physical function, PF, can support a configurable number of PCIe virtual functions. In total 240 VFs can be allocated between the PFs. The adapter can also support a total of 2048 MSI-X interrupts.

- Layer 2 Switching Capability.

A layer 2 switch configured in firmware supports the transport of network packets between PCI physical functions (PF), Virtual functions (VF) and the external network. This allows received packets to be replicated across multiple PFs/VFs and allows packets transmitted from one PF to be received on another PF or VF.

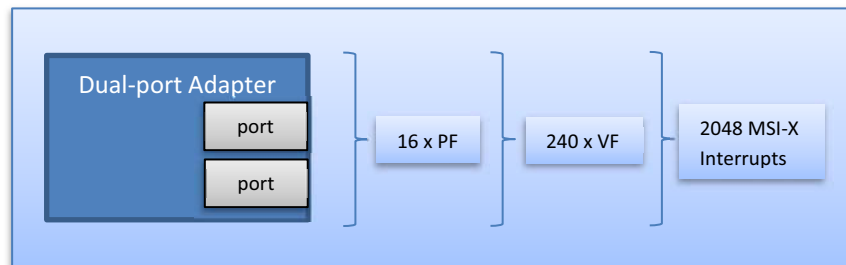


Figure 12: Per Adapter - Configuration Options

## Supported Platforms

### Host

- Red Hat Enterprise Linux 6.5 - 7.6 KVM

### Guest VM

- Red Hat Enterprise Linux 5.x, 6.x and 7.x

Acceleration of guest Virtual Machines (VM) running other (non-Linux) operating systems are not currently supported, however other schemes, for example, a KVM direct bridged configuration using the Windows virtio-net driver could be used.

## Driver/Firmware

Features described in the chapter require the following (minimum) Solarflare driver and firmware versions.

```
# ethtool -i eth<N>
driver: sfc
version: 4.4.1.1017
firmware-version: 4.4.2.1011 rx0 tx0
```

The adapter must be using the *full-feature* firmware variant which can be selected using the *sfbboot* utility and confirmed with **rx0 tx0** appearing after the version number in the output from *ethtool* as shown above.

The firmware update utility (*sfupdate*) and boot ROM configuration tool (*sfbboot*) are available in the Solarflare Linux Utilities package (SF-107601-LS issue 28 or later).

## Platform support - SR-IOV

### BIOS

To use SR-IOV modes, SR-IOV must be enabled in the platform BIOS where the actual BIOS setting can differ between machines, but may be identified as SR-IOV, IOMMU or VT-d and VT-x on an Intel platform.

The following links identify Linux Red Hat documentation for SR-IOV BIOS settings.

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Virtualization\\_Deployment\\_and\\_Administration\\_Guide/index.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Virtualization_Deployment_and_Administration_Guide/index.html)

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Virtualization\\_Administration\\_Guide/sect-Virtualization-Troubleshooting-Enabling\\_Intel\\_VT\\_and\\_AMD\\_V\\_virtualization\\_hardware\\_extensions\\_in\\_BIOS.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Virtualization_Administration_Guide/sect-Virtualization-Troubleshooting-Enabling_Intel_VT_and_AMD_V_virtualization_hardware_extensions_in_BIOS.html)

There may be other BIOS options which should be enabled to support SR-IOV, for example on DELL servers the following BIOS option must also be enabled:

Integrated Devices, SR-IOV Global Enable

*Users are advised to consult the server vendor BIOS options documentation.*

## Kernel Configuration

On an Intel platform, the IOMMU must be explicitly enabled by appending `intel_iommu=on` to the kernel line in the `/boot/grub/grub.conf` file. The equivalent setting on an AMD system is `amd_iommu=on`.

Solarflare recommends that users also enable the `pci=realloc` kernel parameter in the `/boot/grub/grub.conf` file. This allows the kernel to reassign addresses to PCIe apertures (i.e. bridges, ports) in the system when the BIOS does not allow enough PCI apertures for the maximum number of supported VFs.

## KVM - Interrupt Re-Mapping

To use PCIe VF passthrough, the server must support interrupt re-mapping. If the target server does not support interrupt re-mapping it is necessary to set the following option in a user created file e.g. `kvm_iommu_map_guest.conf` in the `/etc/modprobe.d` directory:

```
[RHEL 6] options kvm allow_unsafe_assigned_interrupts=1
```

```
[RHEL 7] options vfio_iommu_type1 allow_unsafe_assigned_interrupts=1
```

## Alternative Routing-ID Interpretation (ARI)

The ARI extension to the PCI Express Base Specification extends the capacity of a PCIe endpoint by increasing the number of accessible functions (PF+VF) from 8, up to 256. Without ARI support - which is a feature of the server hardware and BIOS, a server hosting a virtualized environment will be limited to 8 functions. Solarflare adapters can expose up to 16 PFs and 240 VFs per adapter.

Users should consult the appropriate server vendor documentation to ensure that the host server supports ARI.

## Supported Adapters

All Solarflare adapters fully support SR-IOV.

The `sfbboot` utility allows the user to configure:

- The number of PFs exposed to host and/or Virtual Machine (VM).
- The number VFs exposed to host and/or Virtual Machine (VM).
- The number of MSI-X interrupts assigned to each PF or VF.

The Solarflare implementation uses a single driver (`sfc.ko`) that binds to both PFs and VFs.



## sfboot - Configuration Options

Adapter configuration options are set using the `sfboot` utility *v4.5.0 or later* from the Solarflare Linux Utilities package (SF-107601-LS issue 28 or later). The firmware variant must be set to `full-feature / Virtualization`.

```
# sfboot firmware-variant=full-feature
```

To check the current adapter configuration run the `sfboot` command:

```
# sfboot
Solarflare boot configuration utility [v4.5.0]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
```

```
eth5:
  Boot image                Option ROM only
  Link speed                Negotiated automatically
  Link-up delay time        5 seconds
  Banner delay time         2 seconds
  Boot skip delay time      5 seconds
  Boot type                 Disabled
  Physical Functions per port 1
  MSI-X interrupt limit     32
  Number of Virtual Functions 2
  VF MSI-X interrupt limit  8
  Firmware variant          full feature / virtualization
  Insecure filters          Disabled
  MAC spoofing              Disabled
  VLAN tags                 None
  Switch mode               SRIOV
```

*For some configuration option changes using `sfboot`, the server must be power cycled (power off/power on) before the changes are effective. `sfboot` will display a warning when this is required.*

[Table 39](#) identifies `sfboot` SR-IOV configurable options.

**Table 39: sfboot - SR-IOV options**

Option	Default Value	Description
<code>pf-count=&lt;n&gt;</code>	1	Number of PCIe PFs per physical port.  MAC address assignments may change, after next reboot, following changes with this option.
<code>pf-vlans</code>	None	A comma separated list of VLAN tags for each PF.  <code>sfboot pf-vlans=0,100,110,120</code>  The first tag is assigned to the first PF, thereafter tags are assigned to PFs in (lowest) MAC address order.

**Table 39: sfboot - SR-IOV options (continued)**

Option	Default Value	Description
mac-spoofing	disabled	<p>If enabled, non-privileged functions may create unicast filters for MAC addresses that are not associated with themselves.</p> <p>This should be used when using bonded interfaces where a bond slave inherits the bond master hardware address.</p>
msix-limit=<n>	32	<p>Number of MSI-X interrupts assigned to each PF. The adapter supports a maximum 2048 interrupts. The specified value for a PF must be a power of 2.</p>
switch-mode=<mode>	default	<p>Specifies the mode of operation that the port will be used in:</p> <p>default - single PF created, zero VFs created.</p> <p>sriov - SR-IOV enabled, single PF created, VFs configured with vf-count.</p> <p>partitioning - PFs configured with pf-count, VFs configured with vf-count. See <a href="#">NIC Partitioning on page 62</a> for details.</p> <p>partitioning-with-sriov - SR-IOV enabled, PFs configured with pf-count, VFs configured with vf-count. See <a href="#">NIC Partitioning on page 62</a> for details.</p> <p>pfiov - PFIOV enabled, PFs configured with pf-count, VFs not supported. Layer 2 switching between PFs.</p>
vf-count=<n>	240	<p>Number of virtual functions per PF.</p>

**Table 39: sfbboot - SR-IOV options (continued)**

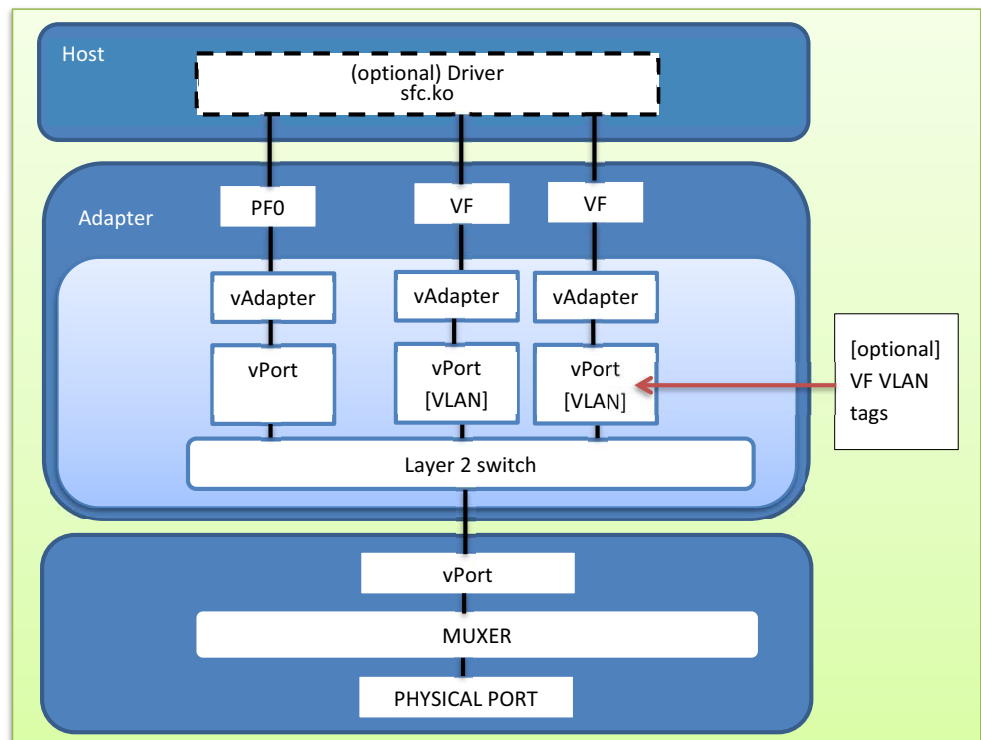
Option	Default Value	Description
vf-msix-limit=<n>	8	Number of MSI-X interrupts per VF. The adapter supports a maximum 2048 interrupts. The specified value for a PF must be a power of 2.
insecure_filters=<enabled disabled>	disabled	When enabled, a function (PF or VF) can insert filters not qualified by its own permanent MAC address.

## 6.2 SR-IOV

In the simplest of SR-IOV supported configurations each physical port is exposed as a single PF (adapter default) and up to 240 VFs.

The Solarflare net driver (sfc.ko) will detect that PF/VFs are present from the sfbboot configuration and automatically configure the virtual adapters and virtual ports as required.

Adapter firmware will also configure the firmware switching functions allowing packets to pass between PF and VFs or from VF to VF.



**Figure 13: SR-IOV - Single PF, Multiple VFs**

- With no VLAN configuration, the PFs and VFs are in the same Ethernet layer 2 broadcast domain i.e. a packet broadcast from the PF would be received by all VFs. VLAN tags can optionally be assigned to VFs using standard libvirt commands.
- The L2 switch supports replication of received/transmitted broadcast packets to all functions.
- The L2 switch will replicate received/transmitted multicast packets to all functions that have subscribed.
- The MUXER function is a firmware enabled layer2 switching function for transmit and receive traffic.

In the example above there are no virtual machines (VM) created. Network interfaces for the PF and each VF will appear in the host. An sfc NIC driver loaded in the host will identify the PF and each VF as individual network interfaces.

## SR-IOV Configuration

Ensure SR-IOV and the IOMMU are enabled on the host server kernel command line  
- Refer to [Platform support - SR-IOV on page 225](#).

- 1 The example configures 1 PF per port (default), 2 VFs per PF):

```
sfboot switch-mode=sriov pf-count=1 vf-count=2
Solarflare boot configuration utility [v4.5.0]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
```

eth8:

Boot image	Option ROM only
Link speed	Negotiated automatically
Link-up delay time	5 seconds
Banner delay time	2 seconds
Boot skip delay time	5 seconds
Boot type	Disabled
<b>Physical Functions per port</b>	<b>1</b>
MSI-X interrupt limit	32
<b>Number of Virtual Functions</b>	<b>2</b>
VF MSI-X interrupt limit	8
<b>Firmware variant</b>	<b>full feature / virtualization</b>
Insecure filters	Disabled
MAC spoofing	Disabled
VLAN tags	None
<b>Switch mode</b>	<b>SRIOV</b>

- 2 Create VFs - see [Enabling Virtual Functions on page 247](#).

- 3 The server should be cold rebooted following changes using `sfboot`. Following the reboot, The PF and VFs will be visible in the host using the `ifconfig` command and `lspci` (the output below is from a dual-port adapter. VFs are shown in bold text):

```
# lspci -d1924:
03:00.0 Ethernet controller: Solarflare Communications SFC9120 (rev 01)
03:00.1 Ethernet controller: Solarflare Communications SFC9120 (rev 01)
03:00.2 Ethernet controller: Solarflare Communications Device 1903 (rev 01)
03:00.3 Ethernet controller: Solarflare Communications Device 1903 (rev 01)
03:00.4 Ethernet controller: Solarflare Communications Device 1903 (rev 01)
03:00.5 Ethernet controller: Solarflare Communications Device 1903 (rev 01)
```

- 4 To identify which physical port a given network interface is using:
 

```
# cat /sys/class/net/eth<N>/device/physical_port
```
- 5 To identify which PF a given VF is associated with use the following command (in this example there are 4 VFs assigned to PF eth4):

```
# ip link show

19: eth4: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc mq state DOWN qlen 1000
    link/ether 00:0f:53:21:00:61 brd ff:ff:ff:ff:ff:ff
    vf 0 MAC 76:c1:36:0a:be:2b
    vf 1 MAC 1e:b8:a8:ea:c7:fb
    vf 2 MAC 52:6e:32:3d:50:85
    vf 3 MAC b6:ad:a0:56:39:94
```

MAC addresses beginning `00:0f:53` are Solarflare designated hardware addresses. MAC addresses assigned to VFs in the above example output have been randomly generated by the host. MAC addresses visible to the host will be replaced by libvirt-generated MAC addresses in a VM.

## 6.3 KVM Network Architectures

This section identifies SR-IOV and the Linux KVM virtualization infrastructure configurations to consume adapter port Physical Functions (PF) and Virtual Functions (VF).

- [KVM libvirt Bridged on page 232](#)
- [KVM Direct Bridged on page 235](#)
- [KVM Libvirt Direct Passthrough on page 238](#)
- [KVM Libvirt Network Hostdev on page 242](#)
- [General Configuration on page 247](#)
- [Enabling Virtual Functions on page 247](#)

When migration is not a consideration, Solarflare recommends the network-hostdev configuration for highest throughput and lowest latency performance

## KVM libvirt Bridged

The traditional method of configuring networking in KVM virtualized environments uses the para-virtualized (PV) driver, `virtio-net`, in the virtual machine and the standard Linux bridge in the host.

The bridge emulates a layer 2 learning switch to replicate multicast and broadcast packets in software and supports the transport of network traffic between VMs and the physical port.

This configuration uses standard Linux tools for configuration and needs only a virtualized environment and guest operating system.

Performance (latency/throughput) will not be as good as a network-hostdev configuration because network traffic must pass via the host kernel.

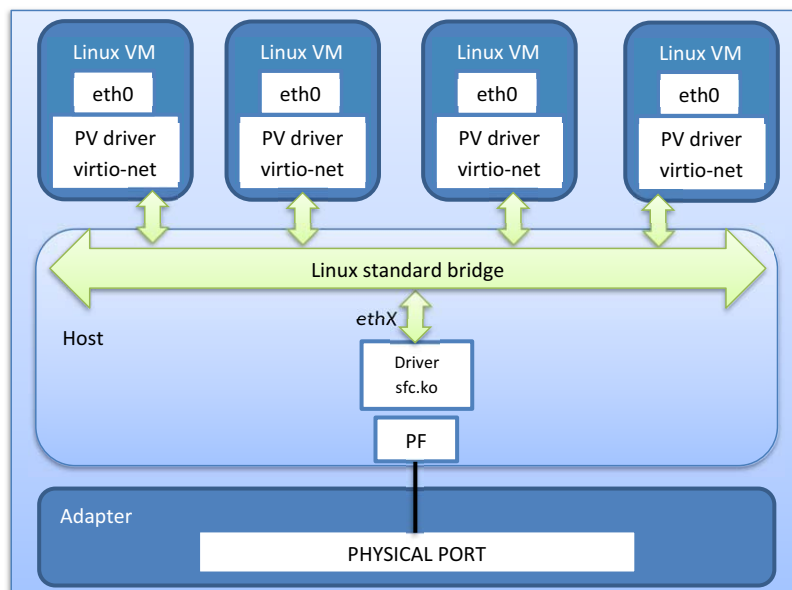


Figure 14: KVM - libvirt bridged

### KVM libvirt bridged - Configuration

1 Ensure the Solarflare adapter driver (`sfc.ko`) is installed on the host.

2 In the host, configure the PF:

```
# sfboot switch-mode=default pf-count=1
```

The `sfboot` settings shown above are the default (shipping state) settings for the SFN7000 series adapter. A cold reboot of the server is only required when changes are made using `sfboot`.

3 Create virtual machines:

VMs can be created from the standard Linux `virt-manager` GUI interface or the equivalent `virsh` command line tool. As root, run the command `virt-manager` from a terminal to start the GUI interface. A VM can also be created from an existing VM XML file.

The following procedure assumes the VM is created. The example procedure will create a bridge 'br1' and network 'host-network' to connect the VM to the Solarflare adapter via the bridge.

- 4 Define a bridge in /etc/sysconfig/network-scripts/ifcfg-br1
 

```
DEVICE=br1
TYPE=Bridge
BOOTPROTO=none
ONBOOT=yes
DELAY=0
NM_CONTROLLED=no
```
- 5 Associate the bridge with the required Solarflare PF (HWADDR) in a config file in /etc/sysconfig/network-scripts/ifcfg-eth4 (this example uses eth4):
 

```
DEVICE=eth4
TYPE=Ethernet
HWADDR=00:0F:53:21:00:60
BOOTPROTO=none
ONBOOT=yes
BRIDGE=br1
```
- 6 Bring up the bridge:
 

```
# service network restart
```
- 7 The bridge will be visible in the host using the ifconfig command:
 

```
# ifconfig -a
br1      Link encap:Ethernet  HWaddr 00:0F:53:21:00:60
         inet6 addr: fe80::20f:53ff:fe21:60/64 Scope:Link
         UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
         RX packets:170 errors:0 dropped:0 overruns:0 frame:0
         TX packets:6 errors:0 dropped:0 overruns:0 carrier:0
         collisions:0 txqueuelen:0
         RX bytes:55760 (54.4 KiB)  TX bytes:468 (468.0 b)
```
- 8 Define a network in an XML file i.e. host-network.xml:
 

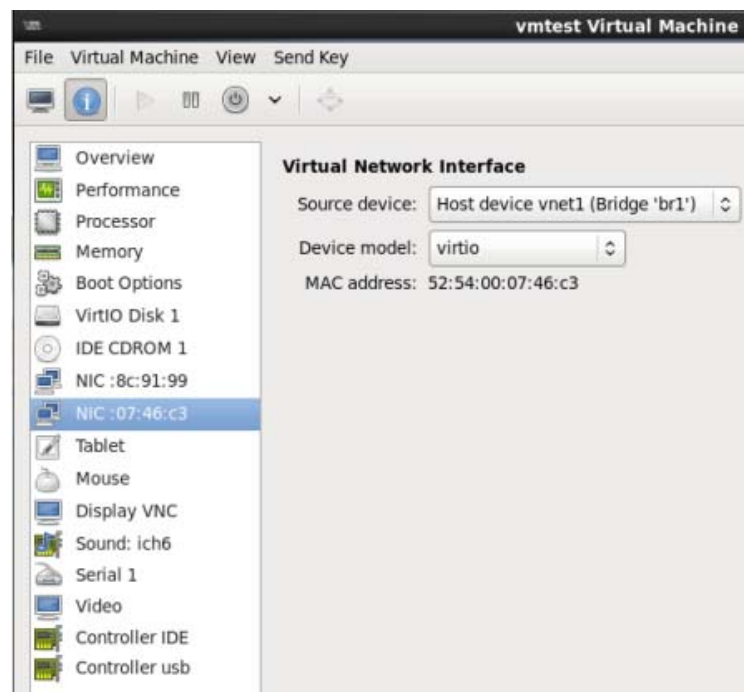
```
<network>
  <name>host-network</name>
  <forward mode='bridge'/>
  <bridge name="br1"/>
</network>
```
- 9 Define and start the network using virsh net-<option> commands:
 

```
# virsh net-define host-network.xml
Network host-network defined from host-network.xml
# virsh net-start host-network
Network host-network started
# virsh net-autostart host-network
Network host-network marked as autostarted
# virsh net-list --all
```

Name	State	Autostart	Persistent
default	active	yes	yes
host-network	active	yes	yes

- 10 On the host machine, edit the VM XML file:  
# virsh edit <vmname>
- 11 Add the network component to the VM XML file:  

```
<interface type='network'>
  <source network='host-network' />
  <model type='virtio' />
</interface>
```
- 12 Restart the VM after editing the XML file.  
# virsh start <vmname>
- 13 The bridged interface is visible in the VM when viewed from the GUI Virtual Machine Manager:



**Figure 15: Virtual Machine Manager - Showing the network/bridged interface**



## XML Description

The following extract is from the VM XML file after the configuration procedure has been applied (line numbers have been added for ease of description):

```

1. <interface type='bridge'>
2.   <mac address='52:54:00:96:0a:8a' />
3.   <source bridge='br1' />
4.   <model type='virtio' />
5.   <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0' />
6. </interface>

```

- 1 Interface type must be specified by the user as 'bridge'.
- 2 The MAC address. If not specified by the user this will be automatically assigned a random MAC address by libvirt.
- 3 The source bridge as created in configuration step 4 above.
- 4 Model type must be specified by the user as 'virtio'.
- 5 The PF PCIe address (as known by the guest) will be added automatically by libvirt.

For further information about the direct bridged configuration and XML formats, refer to the following links:

<http://libvirt.org/formatdomain.html#elementsNICSBridge>

## KVM Direct Bridged

In this configuration multiple macvtap interfaces are bound over the same PF. For each VM created, libvirt will automatically instantiate a macvtap driver instance and the macvtap interfaces will be visible on the host.

Where the KVM libvirt bridged configuration uses the standard Linux bridge, a direct bridged configuration bypasses this providing an internal bridging function and increasing performance.

When using macvtap there is no link state propagation to the guest which is unable to identify if a physical link is up or down.

Macvtap does not currently forward multicast joins from the guests to the underlying network driver with the result that all multicast traffic received by the physical port is forwarded to all guests. Due to this limitation this configuration is not recommended for deployments that use a non-trivial amount of multicast traffic.

Guest migration is fully supported as there is no physical hardware state in the VM guests. A guest can be migrated to a host using a different VF or a host without an SR-IOV capable adapter.

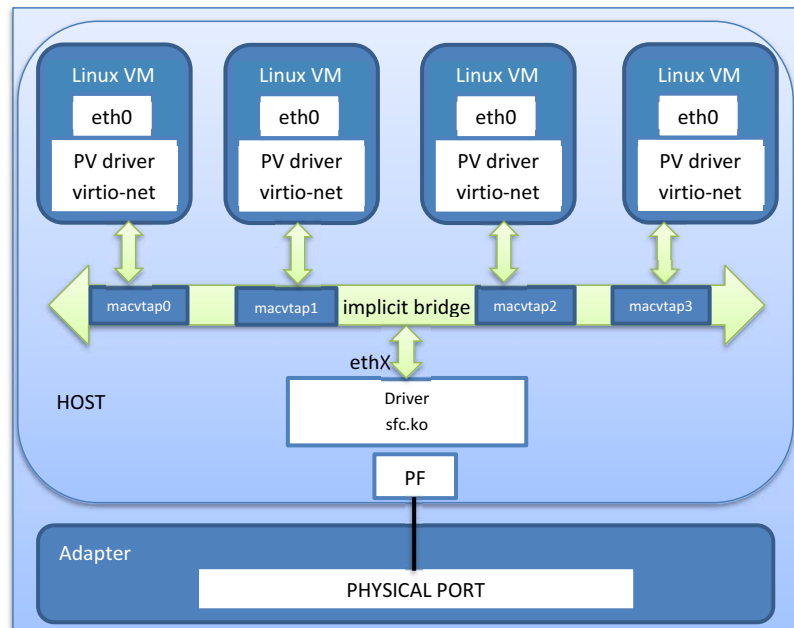


Figure 16: KVM - direct bridged

### KVM direct Bridged - Configuration

- 1 Ensure the Solarflare adapter driver (sfc.ko) is installed on the host.
- 2 In the host, configure the PF.

```
# sfboot switch-mode=default pf-count=1
```

The sfboot settings shown above are the default (shipping state) settings for the SFN7000 series adapter. A cold reboot of the server is only required when changes are made using sfboot.

- 3 Create virtual machines:

VMs can be created from the standard Linux virt-manager GUI interface or the equivalent virsh command line tool. As root, run the command virt-manager from a terminal to start the GUI interface. A VM can also be created from an existing VM XML file.

The following procedure assumes the VM is created. The example procedure will create an interface configuration file and connect the VM directly to the Solarflare adapter.

- 4 Create a configuration file for the required Solarflare PF (HWADDR) in a config file in /etc/sysconfig/network-scripts/ifcfg-eth4 (this example uses eth4):

```
DEVICE=eth4
TYPE=Ethernet
HWADDR=00:0F:53:21:00:60
BOOTPROTO=none
ONBOOT=yes
```

- 5 Bring up the interface:

```
# service network restart
```

- 6 On the host machine, edit the VM XML file:

```
# virsh edit <vmname>
```

- 7 Add the interface component to the VM XML file:

```
<interface type='direct'>
  <source dev='eth4' mode='bridge'/>
  <model type='virtio'/>
</interface>
```

- 8 Restart the VM after editing the XML file.

```
# virsh start <vmname>
```

- 9 The bridged interface is visible when viewed from the GUI Virtual Machine Manager:

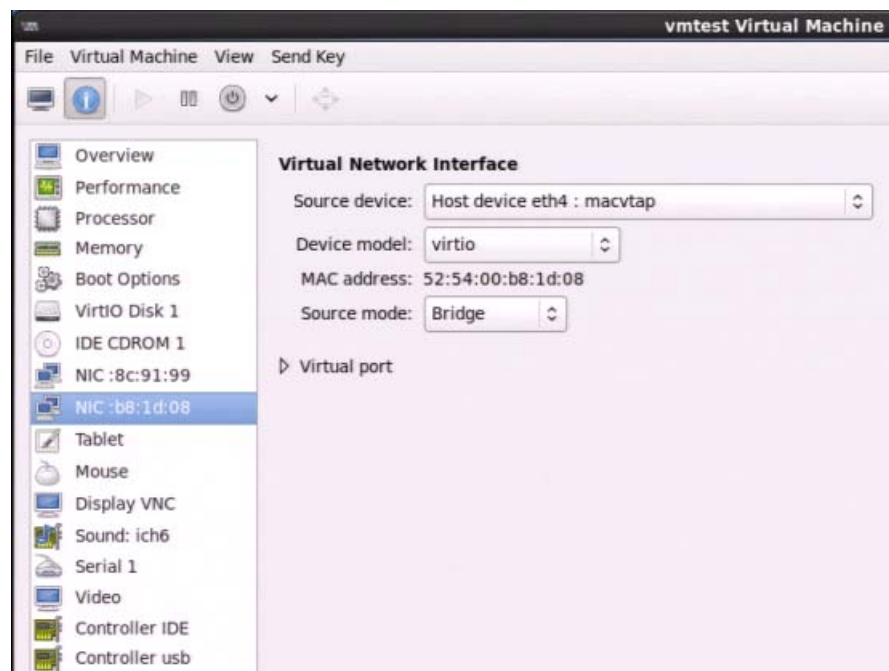


Figure 17: Virtual Machine Manager - Showing the direct bridged interface

### XML Description

The following extract is from the VM XML file after the configuration procedure has been applied (line numbers have been added for ease of description):

1. <interface type='direct'>
2. <mac address='52:54:00:db:ab:ca'/>
3. <source dev='eth4' mode='bridge'/>
4. <model type='virtio'/>
5. <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'/>
6. </interface>

- 1 Interface type must be specified by the user as 'direct'.
- 2 The MAC address. If not specified by the user this will be automatically assigned a random MAC address by libvirt.

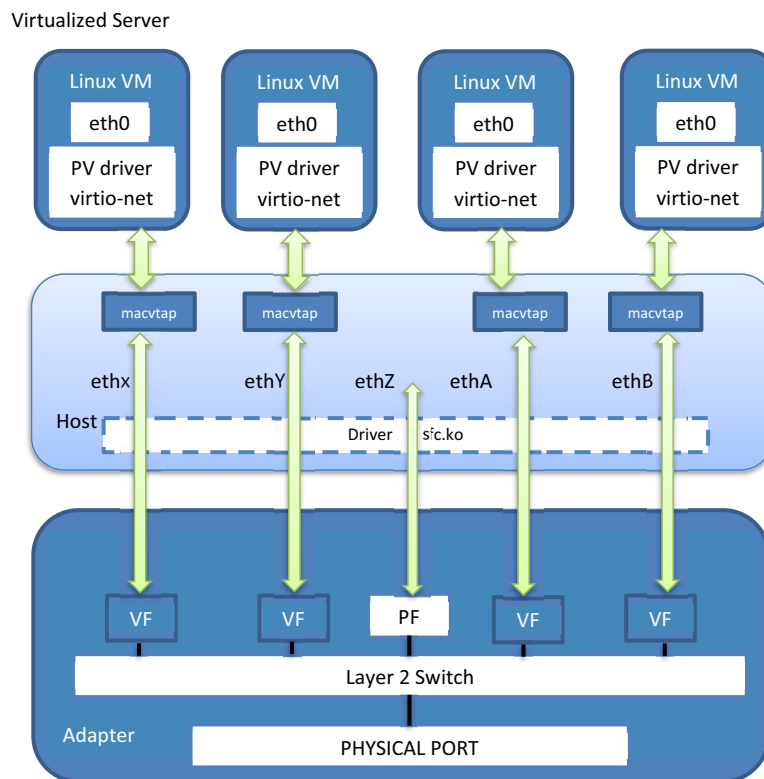
- 3 The source dev is the interface identifier from the host - added by the user. The user should also specify the mode which must be 'bridge'.
- 4 If not specified by the user, the model type will be automatically assigned by libvirt when the guest is started. Use virtio for best performance.
- 5 The PF PCIe address (as known by the guest) will be added automatically by libvirt

For further information about the direct bridged configuration and XML formats, refer to the following link:

<http://libvirt.org/formatdomain.html#elementsNICBridge>

## KVM Libvirt Direct Passthrough

Using a libvirt direct-passthrough configuration, VFs are used in the host OS to provide network acceleration for guest VMs. The guest continues to use a paravirtualized driver and is unaware this is backed with a VF from the network adapter.



**Figure 18: SR-IOV VFs used in the host OS**

- The Solarflare net driver is bound over the top of each VF.
- Each macvtap interface is implicitly created by libvirt over a single VF network interface and is not visible to the host OS.

- Each macvtap instance builds over a different network interface - so there is no implicit macvtap bridge.
- Macvtap does not currently forward multicast joins from the guests to the underlying network driver with the result that all multicast traffic received by the physical port is forwarded to all guests. Due to this limitation this configuration is not recommended for deployments that use a non-trivial amount of multicast traffic.
- Guest migration is fully supported as there is no physical hardware state in the VM guests. A guest can be reconfigured to a host using a different VF or a host without an SR-IOV capable adapter.
- The MAC address from the VF is passed through to the para-virtualized driver.
- Because there is no VF present in a VM, Onload and other Solarflare applications such as SolarCapture cannot be used in the VM.

### KVM Libvirt Direct Passthrough - Configuration

1 Ensure the Solarflare adapter driver (sfc.ko) is installed on the host.

2 In the host, configure the switch-mode, PF and VFs:

```
# sfboot switch-mode=sriov pf-count=1 vf-count=4
```

A cold reboot of the server is required when changes are made using sfboot.

3 Create VFs in the host (example uses PF eth4):

```
echo 2 > /sys/class/net/eth4/device/sriov_numvfs
cat /sys/class/net/eth4/device/sriov_totalvfs
```

For Linux versions earlier than RHEL6.5 see [Enabling Virtual Functions on page 247](#).

4 PFs and VFs will be visible using the lspci command (VFs in **bold**):

```
# lspci -D -d1924:
0000:03:00.0 Ethernet controller: Solarflare Communications SFC9120
0000:03:00.1 Ethernet controller: Solarflare Communications SFC9120
0000:03:00.2 Ethernet controller: Solarflare Communications Device 1903
0000:03:00.3 Ethernet controller: Solarflare Communications Device 1903
0000:03:00.4 Ethernet controller: Solarflare Communications Device 1903
0000:03:00.5 Ethernet controller: Solarflare Communications Device 1903
```

VFs will also be listed using the ifconfig command (abbreviated output below, from a dual port adapter, shows 2 x PF and 4 x VF. (pf-count=1 vf-count=2). VFs are shown in **bold**).

```
eth4      Link encap:Ethernet HWaddr 00:0F:53:21:00:60
eth5      Link encap:Ethernet HWaddr 00:0F:53:21:00:61
eth6     Link encap:Ethernet HWaddr AE:82:AB:C9:67:49
eth7     Link encap:Ethernet HWaddr 86:B4:C8:9E:27:D6
eth8     Link encap:Ethernet HWaddr 72:0B:C7:21:E1:59
eth9     Link encap:Ethernet HWaddr D2:B7:68:54:35:A5
```

**5** Create virtual machines:

VMs can be created from the standard Linux virt-manager GUI interface or the equivalent virsh command line tool. As root, run the command virt-manager from a terminal to start the GUI interface. A VM can also be created from an existing VM XML file.

The following procedure assumes the VM is created. The example procedure will create an interface configuration file for each VF to be passed through to the VM.

**6** For each VF to be passed through to a VM, create a configuration file in the /etc/sysconfig/network-scripts directory i.e. ifcfg-eth6:

```
DEVICE=eth6
TYPE=Ethernet
HWADDR=AE:82:AB:C9:67:49
BOOTPROTO=none
ONBOOT=yes
```

The above example is the file ifcfg-eth6 and identifies the MAC address assigned to the VF. One file is required for each VF.

**7** On the host machine, edit the VM XML file:

```
# virsh edit <vmname>
```

**8** Add the interface component to the VM XML file e.g:

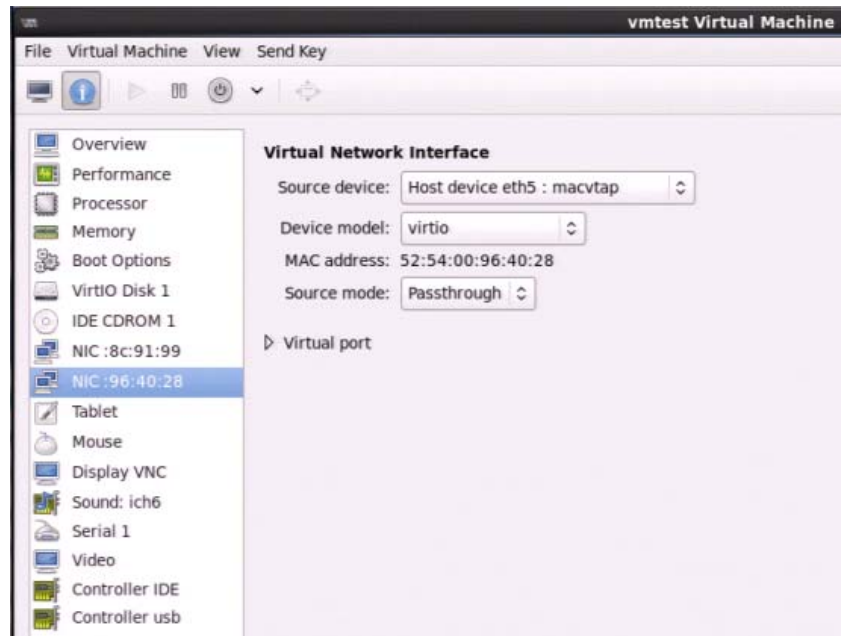
```
<interface type='direct'>
  <source dev='eth6' mode='passthrough' />
  <model type='virtio' />
</interface>
```

One interface type component is required for each VF.

**9** Restart the VM after editing the XML file.

```
# virsh start <vmname>
```

The passed through VF interface is visible when viewed from the GUI Virtual Machine Manager:



**Figure 19: Virtual Machine Manager - Showing the passthrough interface**

### XML Description

The following (example) extract is from the VM XML file after a VF has been passed through to the guest using the procedure above (line numbers have been added for ease of description):

```

1. <interface type='direct'>
2.   <mac address='52:54:00:96:40:28' />
3.   <source dev='eth6' mode='passthrough' />
4.   <model type='virtio' />
5.   <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0' />
6. </interface>

```

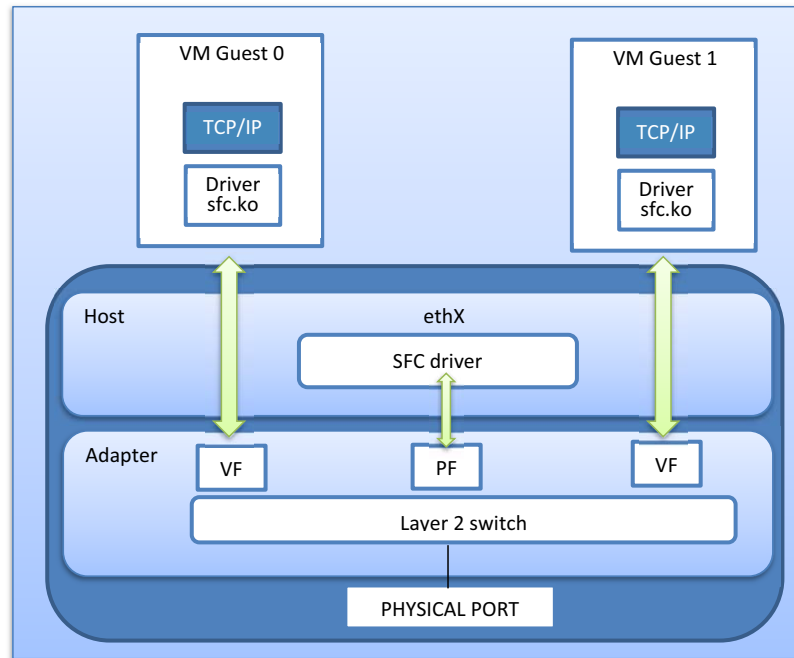
- 1** A description of how the VF interface is managed - added by the user.
- 2** The MAC address. If not specified by the user this will be automatically assigned a random MAC address by the guest OS. The user can specify a MAC address when editing the XML file.
- 3** The source dev is the VF interface identifier - added by the user. The user should also specify the mode which must be 'passthrough'.
- 4** If not specified by the user, the model type will be automatically assigned by libvirt when the guest is started.
- 5** The VF PCIe address (as known by the guest) will be added automatically by libvirt.

For further information about the direct passthrough configuration and XML formats, refer to the following link:

<http://libvirt.org/formatdomain.html#elementsNICSDirect>

## KVM Libvirt Network Hostdev

Network Hostdev exposes VFs directly into guest VMs allowing the data path to fully bypass the host OS and therefore provides maximum acceleration for network traffic.



**Figure 20: SR-IOV VFs passed to guests**

- The hostdev configuration delivers the highest throughput and lowest latency performance. Because the guest is directly linked to the virtual function therefore directly connected to the underlying hardware.
- Migration is not supported in this configuration because the VM has knowledge of the network adapter hardware (VF) present in the server.
- The VF is visible in the guest. This allows applications using the VF interface to be accelerated using OpenOnload or to use other Solarflare applications such as SolarCapture.
- The Solarflare net driver (sfc.ko) needs to be installed in the guest.

### KVM Libvirt network hostdev - Configuration

- 1 Create the VM from the Linux virt-manager GUI interface or the virsh command line tool.
- 2 Install Solarflare network driver (sfc.ko) in the guest and host.
- 3 Create the required number of VFs:  

```
# sfboot switch-mode=sriov vf-count=4
```

A cold reboot of the server is required for this to be effective.



- 4 For the selected PF - configure the required number of VFs e.g:  
# echo 4 > /sys/class/net/eth8/device/sriov\_numvfs
- 5 VFs will now be visible in the host - use ifconfig and the lspci command to identify the Ethernet interfaces and PCIe addresses (VFs shown below in **bold** text):  
# lspci -D -d1924:  
0000:03:00.0 Ethernet controller: Solarflare Communications SFC9120 (rev 01)  
0000:03:00.1 Ethernet controller: Solarflare Communications SFC9120 (rev 01)  
**0000:03:00.2 Ethernet controller: Solarflare Communications Device 1903 (rev 01)**  
**0000:03:00.3 Ethernet controller: Solarflare Communications Device 1903 (rev 01)**  
**0000:03:00.4 Ethernet controller: Solarflare Communications Device 1903 (rev 01)**  
**0000:03:00.5 Ethernet controller: Solarflare Communications Device 1903 (rev 01)**
- 6 Using the PCIe address, unbind the VFs to be passed through to the guest from the host sfc driver e.g.:  
# echo 0000:03:00.5 > /sys/bus/pci/devices/0000\:03\:00.5/driver/unbind
- 7 Check that the required VF interface is no longer visible in the host using ifconfig.
- 8 On the host, stop the virtual machine:  
# virsh shutdown <vmname>
- 9 On the host, edit the virtual machine XML file:  
# virsh edit <vmname>
- 10 For each VF that is to be passed to the guest, add the following <interface type> section to the file identifying the VF PCIe address (use lspci to identify PCIe address):  

```
<interface type='hostdev' managed='yes'>
  <source>
    <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x5' />
  </source>
</interface>
```
- 11 Restart the virtual machine in the host and VF interfaces will be visible in the guest:  
# virsh start <vmname>

The following (example) extract is from the VM XML file after a VF has been passed through to the guest using the procedure above (line numbers have been added for ease of description):

1. <interface type='hostdev' managed='yes'>
2. <mac address='52:54:00:d1:ec:85' />
3. <source>
4. <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x5' />
5. </source>
6. <alias name='hostdev0' />
7. <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0' />
8. </interface>

## XML Description

- 1** A description of how the VF interface is managed - added by user.  
When managed=yes, the VF is detached from the host before being passed to the guest and the VF will be automatically reattached to the host after the guest exits.  
If managed=no, the user must call `virNodeDeviceDetach` (or use the command `virsh nodedev-detach`) before starting the guest or hot-plugging the device and call `virNodeDeviceReAttach` (or use command `virsh nodedev-reattach`) after hot-unplug or after stopping the guest.
- 2** The VF MAC address. If not specified by the user this will be automatically assigned a random MAC address by libvirt The user can specify a MAC address when editing the XML file.
- 3** The VF PCIe address, this is the address of the VF interface as it is identified in the host. This should be entered by the user when editing the XML file.
- 4** If not specified by the user the alias name will be automatically assigned by libvirt The user can supply an alias when editing the XML file.
- 5** The VF PCIe address (as known by the guest) will be added automatically by libvirt.

For further information about the hostdev configuration and XML formats, refer to the following link:

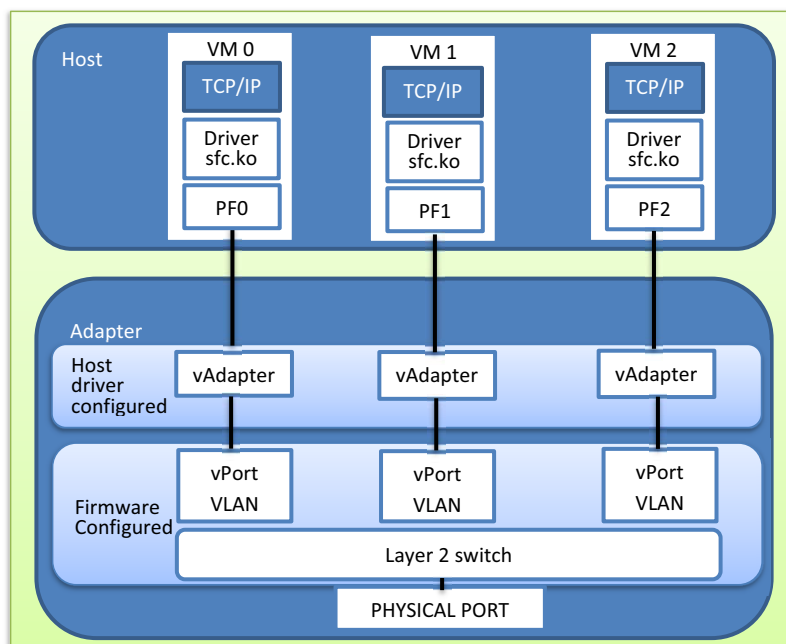
<http://libvirt.org/formatdomain.html#elementsNICSHostdev>

## 6.4 PF-IOV

Physical Function I/O Virtualization allows PFs to be passed to a VM. Although this configuration is not widely used, it is included here for completeness. This mode provides no advantage over “Network Hostdev” and therefore Solarflare recommends that customers deploy “Network hostdev instead of PF-IOV. PF-IOV does not use SR-IOV and does not require SR-IOV hardware support.

Each physical port is partitioned into a number of PFs with each PF passed to a different Virtual Machine (VM). Each VM supports a TCP/IP stack and Solarflare adapter driver (sfc.ko).

This mode allows switching between PFs via the Layer 2 switch function configured in firmware.



**Figure 21: PFIOV**

- Up to 16 PFs and 16 MAC addresses are supported *per adapter*.
- With no VLAN configuration, all PFs are in the same Ethernet layer 2 broadcast domain i.e. a packet broadcast from any one PF would be received by all other PFs.
- PF VLAN tags can optionally be assigned when creating PFs using the sboot utility.
- The layer 2 switch supports replication of received/transmitted broadcast packets to all PFs and to the external network.
- The layer 2 switch supports replication of received/transmitted multicast packets to all subscribers.
- VFs are not supported in this mode.

## PF-IOV Configuration

The `sboot` utility from the Solarflare Linux Utilities package (SF-107601-LS) is used to partition physical interfaces to the required number of PFs.

- Up to 16 PFs and 16 MAC addresses are supported per adapter.
- The PF setting applies to all physical ports. Ports cannot be configured individually.
- `vf-count` must be zero.

**1** To partition all ports (example configures 4 PFs per port):

```
# sboot switch-mode=pfiov pf-count=4
```

```
Solarflare boot configuration utility [v4.3.1]
Copyright Solarflare Communications 2006-2014, Level 5 Networks 2002-2005
```

```
eth5:
```

```
  Boot image           Option ROM only
  Link speed           Negotiated automatically
  Link-up delay time   5 seconds
  Banner delay time    2 seconds
  Boot skip delay time 5 seconds
  Boot type            Disabled
Physical Functions per port      4
  MSI-X interrupt limit 32
Number of Virtual Functions    0
  VF MSI-X interrupt limit 8
Firmware variant               full feature / virtualization
  Insecure filters      Disabled
  VLAN tags             None
Switch mode                   PFIOW
```

**2** A reboot of the server is required for the changes to be effective.

**3** Following reboot the PFs will be visible using the `ifconfig` or `ip` commands - each PF will have a unique MAC address. The `lspci` command will also identify the PFs:

```
# lspci -d 1924:
```

```
07:00.0 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.1 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.2 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.3 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.4 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.5 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.6 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
07:00.7 Ethernet controller: Solarflare Communications Device 0903 (rev 01)
```

## Identify PFs to Physical Port

The following command can be used to identify the physical port that a PF belongs to:

```
# cat /sys/class/net/enp4s0f?/device/physical_port
0
1
0
1
0
1
0
```

From lspci output above, the primary PF of each physical port can be identified as they have the PCIe function 0 or 1: e.g. 07:00.0 and 07:00.1.

## 6.5 General Configuration

### Enabling Physical Functions

Use the `sfboot` utility from the Solarflare Linux Utilities package to create PFs. Up to 16 PF and 16 MAC addresses are supported *per adapter*.

```
sfboot pf-count=<N>
```

PF VLAN tags can also be assigned using `sfboot`.

```
sfboot pf-count=4, pf-vlans=100,110,200,210
```

The first VLAN tag is assigned to the first function, thereafter the tags are applied to PFs in MAC address order.

### Enabling Virtual Functions

On RHEL6.5 and later versions, VF creation is controlled through `sysfs`. Use the following commands (example) to create and view created VFs.

```
echo 2 > /sys/class/net/eth8/device/sriov_numvfs
cat /sys/class/net/eth8/device/sriov_totalvfs
```

On kernels not having this control via `sysfs` the Solarflare net driver module option `max_vfs` can be used to enable VFs. The `max_vfs` value applies to all adapters and can be set to a single integer i.e. all adapter physical functions will have the same number of VFs, or can be set to a comma separated list to have different numbers of VFs per PF.

The driver module parameter should be enabled in a user-created file (e.g. `sfc.conf`) in the `/etc/modprobe.d` directory and the `sfc` driver must be reloaded following changes.

```
options sfc max_vfs=4
options sfc max_vfs=2,4,8
```

When specified as a comma separated list, the first VF count is assigned to the PF with the lowest index i.e. the lowest MAC address, then the PF with the next highest MAC address etc. If the sfc driver option is used to create VFs, reload the driver:

```
modprobe -r sfc
modprobe sfc
```

VLAN tags can be dynamically assigned to VFs using libvirt commands, or using the ip command:

```
ip link vf NUM [mac LLADDR] [vlan VLANID]
```

To ensure VLAN tags persist after reboot, these can be configured in the VM XML file.

## Using OpenOnload in a Virtual Machine

Onload users should refer to the Onload User Guide (SF-104474-CD) for further information about using Onload in a KVM.

When Onload and the sfc net driver have been installed in the guest, the sfc driver module option num\_vis is used to allocate the required number of virtual interfaces. One VI is needed for each Onload stack using a VF.

Driver module options should be enabled in a user created file (e.g. sfc.conf) in the /etc/modprobe.d directory.

```
options sfc num_vis=<num>
```

Reload the driver after setting/changing this value:

```
# onload_tool reload
```

## 6.6 Feature Summary

The table below summarizes the features.

**Table 40: Feature Summary**

	Default	SRIOV	Partitioning	Partitioning + SRIOV	PFIOV
Number of PFs (per adapter)	num ports	num ports	≥num ports ≤16	≥num ports ≤16	≥num ports ≤16
All PFs (per port) must be on unique VLANs	N/A	N/A	Yes	Yes	No
Num VFs (per adapter)	0	>0, ≤240	0	>0, ≤240	0
Mode suitable for PF PCIe passthrough	No	No	No	No	Yes
Mode suitable for VF PCIe passthrough	No	Yes	No	Yes	No

**Table 40: Feature Summary (continued)**

	Default	SRIOV	Partitioning	Partitioning + SRIOV	PFIOW
sfboot settings	switch-mode =default	switch-mode =sriov	switch-mode =partitioning	switch-mode =partitioning -with-sriov	switch-mode =pfiov
	pf-count=1	pf-count=1	pf-count>1	pf-count>1	pf-count>1
	vf-count=0	vf-count>0	vf-count=0	vf-count>0	vf-count=0
L2 switching between PF and associated VFs	N/A	Yes	N/A	Yes	N/A
L2 switching between PFs on the same physical port	N/A	N/A	No	No	Yes

## 6.7 Limitations

Users are advised to refer to the Solarflare net driver release notes for details of all limitations.

### Per Port Configuration

For initial releases, all PFs on a physical port have the same expansion ROM configuration where PXE/UEFI settings are stored. This means that all PFs will PXE boot or none will attempt to PXE boot. Users should ensure that a DHCP server responds to the first MAC address.

The PF (pf-count) configuration is a global setting and applies to all physical ports on an adapter. It is not currently possible to configure ports individually.

### PTP

PTP can only run on the primary physical function of each physical port and is not supported on VF interfaces.

# 7

## SR-IOV Virtualization Using ESXi

This chapter includes procedures for installation and configuration of Solarflare adapters for SR-IOV using VMware® ESXi. For details of installation and configuration on VMware® platforms refer to [Solarflare Adapters on VMware on page 173](#).

### Features Supported

On ESXi Solarflare adapters support the following deployments:

**Table 41: ESXi Virtualization Features**

Feature	Guest OS
VF Passthrough	Linux 6.5 to 7.x
PF Passthrough (DirectPath I/O)	Linux 6.5 to 7.x Windows Server 2012 R2

### Platform Compatibility

SR-IOV and DirectPath I/O are not supported on all server platforms and users are advised to check server compatibility.

DirectPath I/O - PF Passthrough does not require platform SR-IOV support.

- Check for SR-IOV support in the VMware compatibility web page: <http://www.vmware.com/resources/compatibility/search.php>
- Ensure the BIOS has all SR-IOV/Virtualization options enabled.
- On a server with SR-IOV correctly configured, identify if Virtual Functions (VF) can be exposed to the host OS. Refer to `sfboot` options below for the procedure to configure VFs on the Solarflare adapter.

### BIOS

To use SR-IOV modes, SR-IOV must be enabled in the platform BIOS where the actual BIOS setting can differ between machines, but may be identified as SR-IOV, IOMMU or VT-d and VT-x on an Intel platform.

There may be other BIOS options which should be enabled to support SR-IOV, for example on DELL servers the following BIOS option must also be enabled:

Integrated Devices, SR-IOV Global Enable

*Users are advised to consult the server vendor BIOS options documentation.*



## Supported Platform OS

### Host

VMware ESXi 5.5-6.7

### Guest VM

- Red Hat Enterprise Linux 6.5 to 7.x
- Windows Server 2012 R2

Acceleration of Virtual Machines (VM) running guest operating systems not listed above are not currently supported.

## Solarflare Driver/Firmware

SR-IOV is supported for all Solarflare adapters using either the

- Solarflare legacy driver ([Legacy Driver \(vmkernel API\) on page 174](#))
- Solarflare native driver ([Native ESXi Driver \(VMkernel API\) on page 174](#)).

To use utilities such as `sfupdate` and `sfboot_esxi`, the Solarflare supplied CIM-Provider package should also be installed on the ESXi host.

Features described in the chapter require the following (minimum) Solarflare driver and firmware versions.

### Native Driver and CIM for ESXi host:

*Supports the SFN8000 series and X2 series adapters.*

Part Number	Minimum version
SF-118824-LS	version 5, driver version 2.3.2.0000
SF-120055-LS	version 4, CIM-Provider version 2.1.0.24

### Legacy Driver and CIM for ESXi host:

*Supports the SFN5000, SFN6000, SFN7000 and SFN8000 series adapters.*

Part Number	Minimum version
SF-111981-LS	version 11, driver version 4.10.10.1001
SF-115711-LS	version 3, CIM-Provider version 2.0.0.3

## 7.1 Configuration Procedure - SR-IOV

Use the following procedure to configure the adapter and server for SR-IOV.

- [Install the Solarflare Driver on the ESXi host on page 252](#)
- [Solarflare Utilities for legacy driver on the ESXi host on page 252](#)
- [Install Solarflare Drivers in the Guest on page 252](#)
- [Configure VFs on the Host/Adapter on page 255](#)

## 7.2 Configuration Procedure - DirectPath I/O

Use the following procedure to configure the adapter and server for PF passthrough.

- [Install the Solarflare Driver on the ESXi host on page 252](#)
- [Solarflare Utilities for legacy driver on the ESXi host on page 252](#)
- [Install Solarflare Drivers in the Guest on page 252](#)

## 7.3 Install Solarflare Drivers in the Guest

For both VF and PF passthrough configurations, the Solarflare adapter driver must be installed in the virtual machine guest OS.

Drivers are available from the Solarflare download site for Linux and Windows guests: <https://support.solarflare.com/>.

Driver installation procedures on a guest are the same as installation for a host.

## 7.4 Install the Solarflare Driver on the ESXi host

Solarflare VMware ESXi drivers are available from: <https://support.solarflare.com/>.

Refer to [Solarflare Adapters on VMware on page 173](#) for instructions to install VIB driver packages through the CLI.

## 7.5 Solarflare Utilities for legacy driver on the ESXi host

Solarflare utilities - including sfboot, sfupdate and sfkey are distributed in the Solarflare Linux Utilities package (SF-107601-LS issue 36 or later).

These can be used on the ESXi host when the legacy driver is installed.



**NOTE:** The Solarflare driver must be installed before using sfboot or any of the utilities.

## sfboot - Configuration Options

The sfboot utility allows the user to configure:

- The number of PFs exposed per port to host and/or Virtual Machine (VM).
- The number VFs exposed per port to host and/or Virtual Machine (VM).
- The number of MSI-X interrupts assigned to each PF or VF.
- Firmware Variant and switch mode.

To check the current adapter configuration run the sfboot command:

```
# sfboot
```

```
Solarflare boot configuration utility [v4.7.0]
Copyright Solarflare Communications 2006-2015, Level 5 Networks 2002-2005
```

```
vmnic6:
  Boot image                               Disabled
  Physical Functions on this port         1
  PF MSI-X interrupt limit                 32
  Virtual Functions on each PF           4
  VF MSI-X interrupt limit                 16
  Port mode                                2x10G
  Firmware variant                       Full feature / virtualization
  Insecure filters                         Enabled
  MAC spoofing                             Disabled
  VLAN tags                                 None
  Switch mode                             SR-IOV
  RX descriptor cache size                 32
  TX descriptor cache size                 16
  Total number of VIs                      2048
  Rate limits                              None
  Event merge timeout                      8740 nanoseconds
```

A reboot of the server (power OFF/ON) is required for the changes to become effective.

## Firmware Variant

The firmware variant must be set to full-feature/virtualization.

```
# sfboot --adapter=vmnic6 firmware-variant=full-feature
```

## SR-IOV (VF Passthrough) sfboot Settings

The following example creates 4 VFs for each physical port.

```
# sfboot switch-mode=sriov pf-count=1 vf-count=4
```

When used without the --adapter option, the command applies to all adapters

## 7.6 Solarflare Utilities for native driver on the ESXi host

When the native driver is installed on the ESXi host, the user can use `sfupdate` from the `esxcli extensions` command line and `sfboot_esxi` from a remote Linux server to connect with the CIM-Provider installed on the ESXi host.

Refer to [Install CIM Provider on page 197](#) to install the CIM-Provider.

Refer to [Adapter Configuration - sfboot\\_esxi on page 201](#) to configure the adapter using `sfboot_esxi`.

Refer to [Firmware Images VIB on page 205](#) to upgrade adapter firmware.

### SR-IOV (VF Passthrough) sfboot\_esxi Settings

Use `sfboot_esxi` from a remote Linux server to connect to the CIM-Provider on the ESXi host.

```
sfboot_esxi -i <vmnicX> \  
            -a "https://<fully qualified server domain name>:5989" \  
            -u <user> \  
            -p <user password> \  
            pf-count=1 \  
            vf-count=<N> \  
            firmware-variant=full-feature \  
            switch-mode=sriov
```

For example:

```
sfboot_esxi -i vmnic4 \  
            -a "https://server1.mycompanycom.com:5989" \  
            -u root \  
            -p tester \  
            pf-count=1 \  
            vf-count=4 \  
            firmware-variant=full-feature \  
            switch-mode=sriov
```

The server must be rebooted (power ON/OFF) for the changes to become effective.

## 7.7 Configure VFs on the Host/Adapter

The following host procedure is used to expose VFs from the Solarflare adapter.

- 1 Set the sfc driver module parameter for the required number of VFs the driver is to support - this should be  $\geq$  the vf-count set on the SR-IOV adapter:

```
esxcli system module parameters set -m [sfc|sfvmk] -p max_vfs=4
esxcli system module parameters list -m [sfc|sfvmk]
```

- 2 List VFs exposed in the host:

```
# lspci | grep Solarflare
0000:04:00.0 Network controller: Solarflare SFC9120 [vmnic6]
0000:04:00.1 Network controller: Solarflare SFC9120 [vmnic7]

0000:04:00.2 Network controller: Solarflare [PF_0.4.0_VF_0]
0000:04:00.3 Network controller: Solarflare [PF_0.4.0_VF_1]
0000:04:00.4 Network controller: Solarflare [PF_0.4.0_VF_2]
0000:04:00.5 Network controller: Solarflare [PF_0.4.0_VF_3]

0000:04:00.6 Network controller: Solarflare [PF_0.4.1_VF_0]
0000:04:00.7 Network controller: Solarflare [PF_0.4.1_VF_1]
0000:04:01.0 Network controller: Solarflare [PF_0.4.1_VF_2]
0000:04:01.1 Network controller: Solarflare [PF_0.4.1_VF_3]
```

The example above is a dual-port adapter. Each physical port is exposed as 1 PF and 4 VFs (PFs are shown in bold text).

- 3 Use the following commands on the ESXi host to identify SR-IOV adapters and VF availability:

```
esxcli network sriovnic list

Name   PCI Device  Driver Link Speed Duplex MAC Address      MTU
vmnic4 0000:81:00.0 sfc    Up    10000 Full  00:0f:53:45:f3:30 1500

Description
Solarflare SFC9220

esxcli network sriovnic vf list -n vmnic4

VF ID  Active  PCI Address      Owner World ID
-----  -----  -----
0      true    00000:129:00.2  68953
1      true    00000:129:00.3  69105
```

## 7.8 Virtual Machine

To connect a VF to a Virtual Machine, the VM should be powered OFF. The PF from which the VF is created should be attached as the UPLINK on a vSwitch and the vSwitch associated with a port group.

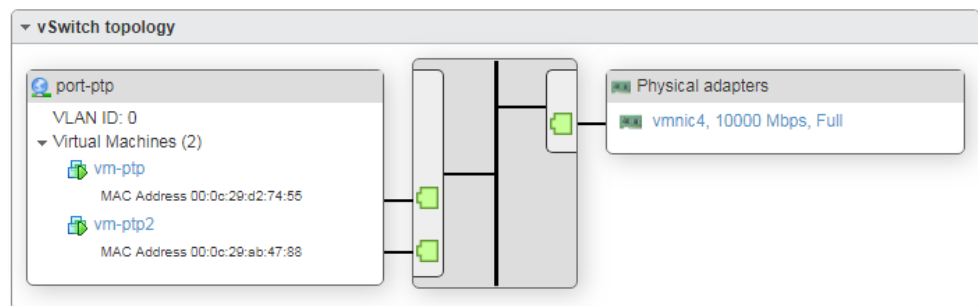
vSwitch and port group are created using vSphere web GUI. Refer to VMware documentation for details if necessary.

The Virtual Machine settings can then be edited to add the VF adapter as a new network adapter type “SRIOV passthrough” adapter before the VM is powered ON.

When powered ON, the VF interface will be available in to the VM.

Check network topology on the vSphere GUIE (vswitch and port group) for VM connectivity.

In the example below, VMNIC4 is the PF uplink and there are two VMs: *vm-ptp* and *vm-ptp2*. Each VM has its own VF from the physical adapter in the host:



**NOTE:** A VF does not use a port-group for sending/receiving traffic. The port-group is used only to apply networking properties such as VLAN ID.

# 8

## Solarflare Boot Manager

### 8.1 Introduction

Solarflare adapters support the following Preboot Execution Environment (PXE) options enabling diskless systems to boot from a remote target operating system:

- Solarflare Boot Manager (based on gPXE) - the default PXE client installed on all Solarflare adapters.
- UEFI network boot.
- iPXE - supported on Solarflare XtremeScale™ and X2 series adapters and the U25 adapter.

The Solarflare Boot Manager complies with the PXE 2.1 specification.

#### Boot Manager Exposed

Solarflare adapters are shipped with boot ROM support 'exposed', that is the Boot Manager runs during the machine bootup stage allowing the user to enter the setup screens (via *Ctrl+B*) and enable PXE support when this is required.

The Boot Manager can also be invoked using the Solarflare supplied *sfboot* utility. For instructions on the *sfboot* method refer to the *sfboot* commands in the relevant OS specific sections of this user guide.

Using the *sfboot* utility, the *boot-image* options identify which boot images are exposed on the adapter during boot time. The *boot-image=uefi* option allows the Solarflare UEFI driver to be loaded by the UEFI platform which can be configured to PXE boot from the Solarflare adapter.

The *boot-type* options allows the user to select PXE boot or to disable PXE boot. This is effective on the next reboot.



**NOTE:** If network booting is not required, startup time can be decreased when the *boot-image* option is 'disabled' so that the CTRL-B option is not exposed during system startup.

## PXE Enabled

Some Solarflare distributors are able to ship Solarflare adapters with PXE boot enabled. Customers should contact their distributor for further information.

Solarflare XtremeScale and X2 series adapters and the U25 adapter are shipped with the PXE boot enabled and set to PXE boot.



**NOTE:** PXE, UEFI network boot is not supported for Solarflare adapters on IBM System p servers.

## Firmware Upgrade - Recommended

Before configuring the Solarflare Boot Manager, it is recommended that servers are running the latest adapter firmware which can be updated as follows:

- From a Windows environment use the supplied Command Line Tool `sfupdate.exe`.
- From a Linux or VMware environment update the firmware via `sfupdate`. See OS specific sections of this document for `sfupdate` commands.

This section covers the following subjects.

- [Solarflare Boot Manager on page 258](#)
- [iPXE Support on page 259](#)
- [sfupdate Options for PXE upgrade/downgrade on page 259](#)
- [Starting PXE Boot on page 261](#)
- [iPXE Image Create on page 265](#)
- [Multiple PF - PXE Boot on page 267](#)

## 8.2 Solarflare Boot Manager

**The standard Solarflare Boot Manager, based on gPXE, is supported on all Solarflare adapters.**

The boot ROM agent, pre-programmed into the adapter's flash image, runs during the machine bootup stage and, if enabled, supports PXE booting the server.

The Boot Manager can be configured using its embedded setup screens (entered via `Ctrl+B` during system boot) or via the Solarflare-supplied `sfboot` utility.

The boot ROM agent firmware version can be upgraded using the Solarflare-supplied `sfupdate` utility. Refer to the OS specific sections of this document for details of `sfupdate` commands.

The use of the Solarflare Boot Manager is fully supported by Solarflare (including meeting any SLA agreements in place for prioritized and out-of-hours support).



## 8.3 iPXE Support

**iPXE boot is supported on Solarflare XtremeScale™ and X2 series adapters and the U25 adapter.**

An iPXE boot image can be programmed into the adapter's flash via the Solarflare-supplied sfupdate utility.

iPXE is an alternative open-source network boot firmware providing both PXE support and additional features such as HTTP and iSCSI boot. Solarflare have integrated, maintain and support iPXE drivers in the iPXE open source code base.

Users can use iPXE features not provided within the gPXE based Solarflare Boot ROM agent. However, iPXE is an open source project with its own development and test process not under the direct control of the Solarflare engineering team. Solarflare will monitor the iPXE development mailing lists and participate to ensure the iPXE driver for Solarflare adapters operates correctly.



**NOTE:** It is recommended that customers having support questions on the iPXE feature set work directly with the iPXE open source community.

## 8.4 sfupdate Options for PXE upgrade/downgrade

This section describes sfupdate when used to install/upgrade/downgrade PXE images. Refer to sfupdate in OS specific sections of this document for a complete list of sfupdate options.

Each version of sfupdate contains a firmware image and Solarflare Boot Manager image.

### Current Versions

Run the sfupdate command to identify current image versions:

```
enp5s0f0 - MAC: 00-0F-53-41-C7-00sfb
  Firmware version: v6.3.0
  Controller type: Solarflare SFC9200 family
  Controller version: v6.2.5.1000
  Boot ROM version: v5.0.3.1002
  UEFI ROM version: v2.4.3.1
```

When an iPXE image has been flashed over the Solarflare Boot Manager:

```
enp5s0f0 - MAC: 00-0F-53-41-C7-00
  Controller type: Solarflare SFC9200 family
  Controller version: v6.2.5.1000
  Boot ROM version: iPXE
  UEFI ROM version: v2.4.3.1
```



**NOTE:** sfupdate is not able to display version numbers for iPXE images.

## sfupdate - Solarflare Boot Manager image

- To install the firmware image and Solarflare, gPXE based, Boot Manager image:  
# sfupdate [--adapter=] --write [--force] [--backup]
- To reinstall firmware and Solarflare Boot Manager image from sfupdate:  
# sfupdate [--adapter=] --write --restore-bootrom
- To reinstall only a Solarflare Boot Manager image from backup:  
# sfupdate [--adapter=] --write --img=<image.dat>

Use the --force option when downgrading. Use the --backup option to create a backup image (.dat) file of the current firmware and Solarflare Boot Manager image.

## sfupdate - iPXE image

- To install the iPXE image, but keep current firmware:  
# sfupdate [--adapter=] --write [--backup] --ipxe-image=<image.mrom>

Use the --backup option to create a backup of the existing firmware and PXE boot ROM image.

- To upgrade firmware and retain the iPXE image:  
# sfupdate [--adapter=] --write [--force]

Using the --force option allows firmware to be downgraded but keeps the current iPXE image.



**CAUTION:** sfupdate does not do version checking for iPXE therefore it is possible to downgrade the image without any displayed warning and without using the --force option.

## 8.5 Starting PXE Boot

The Boot Manager can be configured using any of the following methods:

- On server startup, press *Ctrl+B* when prompted during the boot sequence. This starts the Solarflare Boot Manager GUI.
  - Use the supplied `sfboot` command line tool.
- From a Linux environment, you can use the `sfboot` utility. See [Configuring the Boot Manager with sfboot on page 78](#).

`sfboot` is a command line utility program from the Solarflare Linux Utilities RPM package (SF-107601-LS) available from [support@solarflare.com](mailto:support@solarflare.com).

PXE requires DHCP and TFTP Servers, the configuration of these servers depends on the deployment service used.

### Linux

See [Unattended Installations on page 272](#) for more details of unattended installation on Linux

## Configuring the Boot Manager for PXE

This section describes configuring the adapter via the *Ctrl+B* option during server startup.



**NOTE:** If the BIOS supports console redirection, and you enable it, then Solarflare recommends that you enable ANSI terminal emulation on both the BIOS and your terminal. Some BIOSs are known to not render the Solarflare Boot Manager properly when using vt100 terminal emulation.

- 1 On starting or re-starting the server, press **Ctrl+B** when prompted. The Solarflare Boot Manager is displayed.

```

Solarflare Boot Manager (v5.2.0.1004)
Select Adapter

Base MAC address  PCI      Boot image
| 00-0f-53-4c-e5-e0 04:00.0 OptionROM & UEFI |
| 00-0f-53-5c-ff-a0 05:00.0 OptionROM & UEFI |

Controller:      Solarflare Flareon Ultra 2000 Series 10/25G Adapter

Up,Down Arrow to select | SPACE,+,- to change | ESC to exit | F1=help

```

The initial **Select Adapter** page lists the available adapters. In the above example, that are two adapters, on PCI bus 04 and PCI bus 05.

- 2 Use the arrow keys to highlight the adapter you want to boot via PXE and press *Enter*. The **Adapter Menu** is displayed.

```

Solarflare Boot Manager (v5.2.0.1004)
Adapter Menu

> Global adapter Options ->
Reset to Defaults ->

PF MAC          PCI      Boot type
0 00-0f-53-4c-e5-e0 04:00.0 PXE
1 00-0f-53-4c-e5-e1 04:00.1 PXE

Up,Down Arrow to select | SPACE,+,- to change | ESC to exit | F1=help

```

- 3 Use the arrow keys to highlight the **Global adapter Options** and press *Enter*. The **Global Adapter Options** menu is displayed.

```

Solarflare Boot Manager (v5.2.0.1004)
Adapter Menu

> Global adapter Options ->
Reset to Defaults ->
PF MAC
0 00-0f-53
1 00-0f-53

Global Adapter Options
> Rescue Options ->
Use F1 for help on the fields below
Port Mode: Default
[ ] PF100 enabled
Firmware Variant: Auto
[ ] Allow insecure filters

Up,Down Arrow to select | SPACE,+, - to change | ESC to exit | F1=help

```

- 4 The Rescue Options Window

The default setting from the Rescue Menu is OptionROM & UEFI and it should not be necessary to change this.



**CAUTION:** This is not a standard PXE procedure. Customers with a PXE boot problem should contact [support@solarflare.com](mailto:support@solarflare.com)

- 5 Select the required boot image:
  - a) Use the arrow keys to highlight the **Boot Image**.
  - b) Use the space bar to choose the required image.
  - c) Press the *Esc* key to exit the **Global Adapter Options**.

The **Adapter Menu** is again displayed.

- 6 Use the arrow keys to highlight the PF you want to boot via PXE and press *Enter*.

```

Solarflare Boot Manager (v5.2.0.1004)
Adapter Menu

Global adapter Options ->
Reset to Defaults ->

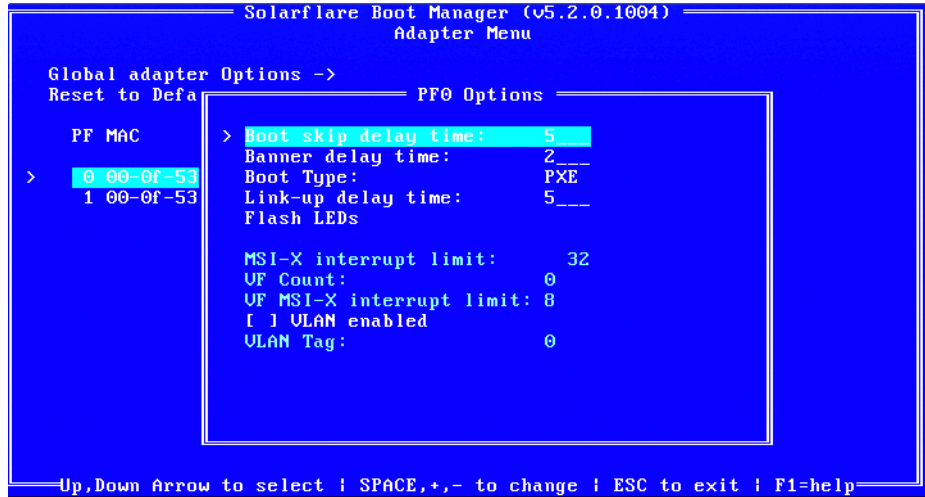
PF MAC          PCI      Boot type
> 0 00-0f-53-4c-e5-e0  04:00.0  PXE
  1 00-0f-53-4c-e5-e1  04:00.1  PXE

Up,Down Arrow to select | SPACE,+, - to change | ESC to exit | F1=help

```

The **PF Options** menu is displayed.

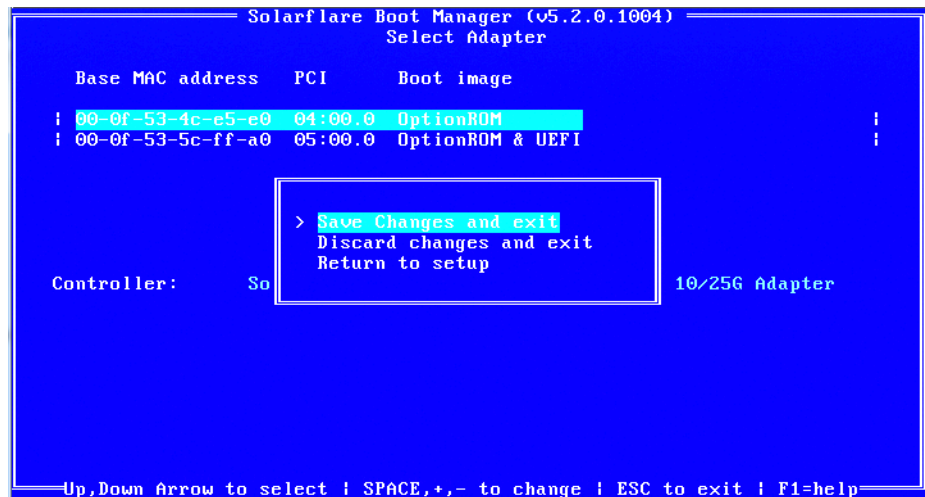
7 Set the PF to use PXE boot:



- a) Use the arrow keys to highlight the **Boot Type**.
- b) Use the space bar to select **PXE**.

Solarflare recommend leaving other settings at their default values. For details on the default values for the various adapter settings, see [Table 8.8 on page 270](#).

- 8 Press the *ESC* key repeatedly until the Solarflare Boot Manager exits.
- 9 Choose **Save Changes and exit**.



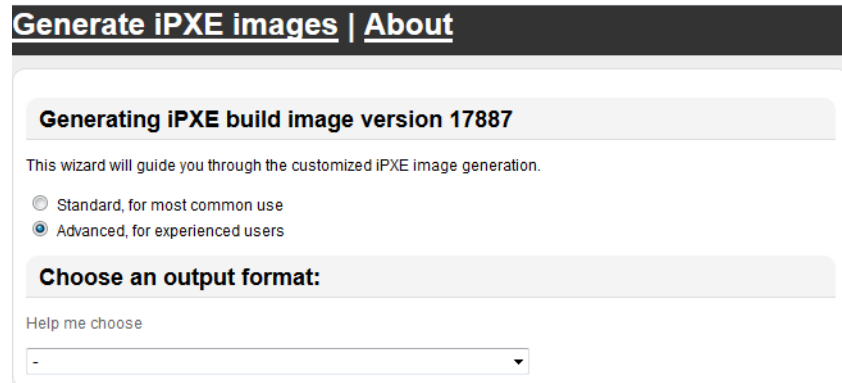
## 8.6 iPXE Image Create

Solarflare do not provide pre-built iPXE images.

The Solarflare iPXE boot ROM image can be generated from the *rom-o-matic* iPXE web builder wizard available from:

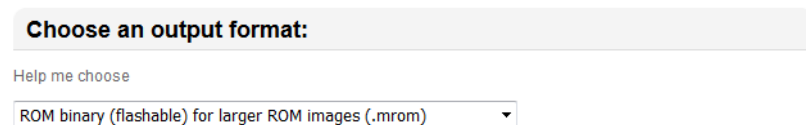
<https://rom-o-matic.eu/>

- 1 Select the Advanced option:



The screenshot shows a web interface titled "Generate iPXE images | About". Below the title is a section "Generating iPXE build image version 17887". A message states: "This wizard will guide you through the customized iPXE image generation." There are two radio button options: "Standard, for most common use" and "Advanced, for experienced users", with the "Advanced" option selected. Below this is a section "Choose an output format:" with a "Help me choose" label and a dropdown menu currently showing a hyphen (-).

- 2 Select an Output format:



This is a close-up of the "Choose an output format:" section. It includes the "Help me choose" label and a dropdown menu that has been expanded to show the option "ROM binary (flashable) for larger ROM images (.mrom)".

### 3 Enter NIC Details:

#### Generating iPXE build image version 17887

This wizard will guide you through the customized iPXE image generation.

Standard, for most common use  
 Advanced, for experienced users

#### Choose an output format:

Help me choose

ROM binary (flashable) for larger ROM images (.mrom) ▼

#### Enter NIC device details:

You have chosen Binary ROM image as your output format. To match the image to your NIC device, please enter its PCI VENDOR CODE and PCI DEVICE CODE.

Information on how to determine NIC PCI IDs:  
 PCI VENDOR CODE:  PCI DEVICE CODE:

iPXE does not support all possible PCI IDs for supported NICs.

#### Embedded script:

Read about embedded scripts

Paste your script:

```
#!ipxe
```

Or import your script:  
 No file selected.

Or drop your script:

Drop your script here

#### Which revision?

Read about GIT revision

Default master (recommended)  ▼

#### Ready to build?

- a) Enter the Solarflare PCI vendor identifier: 1924.
- b) Enter the adapter PCI Device Code: [0a03 | 0b03]<sup>1</sup>.
- c) Select the GIT version (master is recommended).

### 4 Generate the image file.

Click the Proceed button to start image generation then download the created image file to the target server.

### 5 Apply the image to the Solarflare adapter using sfupdate:

```
# sfupdate [--adapter=] --write --ipxe-image=<filename.mrom>
```

1. 0a03 - SFN8000 series adapter, 0b03 - X2 series adapter



## 8.7 Multiple PF - PXE Boot

Using the `sfboot` command line utility v4.5.0 (or later version) it is possible to PXE boot when multiple Physical Functions have been enabled. The primary function on each port (PF0/PF1) is a privileged function and can be selected for configuration. Other PFs inherit from their privileged function- so, for example, with two physical ports and 2 PFs per port:

- PF0 and PF2 will have the same boot-type
- PF1 and PF3 will have the same boot-type

Configuration of non-privileged functions is not currently supported.

In the following example 2 PFs (and 2 VFs) are enabled for each physical interface.

```
# sfboot
Solarflare boot configuration utility [v4.5.0]

eth2:
  Boot image                Option ROM only
  Link speed                Negotiated automatically
  Link-up delay time       5 seconds
  Banner delay time        2 seconds
  Boot skip delay time     5 seconds
  Boot type                 Disabled
  Physical Functions per port 2
  MSI-X interrupt limit    32
  Number of Virtual Functions 2
  VF MSI-X interrupt limit 8
  Firmware variant         full feature / virtualization
  Insecure filters         Disabled
  MAC spoofing             Disabled
  VLAN tags                 100,110
  Switch mode              Partitioning with SRIOV

eth3:
  Boot image                Option ROM only
  Link speed                Negotiated automatically
  Link-up delay time       5 seconds
  Banner delay time        2 seconds
  Boot skip delay time     5 seconds
  Boot type                 Disabled
  Physical Functions per port 2
  MSI-X interrupt limit    32
  Number of Virtual Functions 2
  VF MSI-X interrupt limit 8
  Firmware variant         full feature / virtualization
  Insecure filters         Disabled
  MAC spoofing             Disabled
  VLAN tags                 100,110
  Switch mode              Partitioning with SRIOV
```

eth4:

Interface-specific boot options are not available. Adapter-wide options are available via eth2 (00-0F-53-25-39-90).

eth5:

Interface-specific boot options are not available. Adapter-wide options are available via eth2 (00-0F-53-25-39-90).

## Using the Boot Manager

When multiple Physical Functions have been enabled, the Solarflare Boot Manager GUI utility (CTRL-B) will list them:

```

Solarflare Boot Manager (v4.5.2.1009)
Adapter Menu

> Global adapter Options ->
Reset to Defaults ->

PF MAC                PCI      Boot type  Link speed
0 00-0f-53-20-ff-80   04:00.0  Disabled   Auto
1 00-0f-53-20-ff-81   04:00.1  Disabled   Auto
2 00-0f-53-20-ff-82   04:00.2  Disabled   Auto
3 00-0f-53-20-ff-83   04:00.3  Disabled   Auto
4 00-0f-53-20-ff-84   04:00.4  Disabled   Auto
5 00-0f-53-20-ff-85   04:00.5  Disabled   Auto
6 00-0f-53-20-ff-86   04:00.6  Disabled   Auto
7 00-0f-53-20-ff-87   04:00.7  Disabled   Auto

Up,Down arrow to select | SPACE,+,- to change | ESC to exit | F1=help

```

Figure 22: Boot Manager lists multiple PFs

The settings for each PF are read-only, and the only supported action is to **Flash LEDs** on the port being used.

```

Solarflare Boot Manager (v4.5.2.1009)
Adapter Menu

Global adapter Options ->
Reset to Defaults ->

PF MAC                PF0 Options
> 0 00-0f-53          Boot skip delay time: 5
1 00-0f-53          Banner delay time: 1
2 00-0f-53          Boot Type: Disabled
3 00-0f-53          Link speed: Auto
4 00-0f-53          Link-up delay time: 5
5 00-0f-53          > Flash LEDs
6 00-0f-53          iSCSI Options ->
7 00-0f-53          MSI-X interrupt limit: 32
                   UF Count: 0
                   UF MSI-X interrupt limit: 8
                   [ ] ULAN enabled
                   ULAN Tag: 0

Up,Down arrow to select | SPACE,+,- to change | ESC to exit | F1=help

```

Figure 23: Read-only settings for multiple PFs

## Recovery from incorrect settings

Certain settings must be correct for successful PXE booting, such as:

- port mode
- VLAN tagging.

If these settings become incorrect, for example because a server is moved to a different part of the network. PXE booting will then fail.

To correct these settings, you must use the Solarflare Boot Manger GUI utility. (You cannot use the `sfbboot` command line utility, because there is no OS to host it.) This is possible only in single Physical Function mode.

If multiple Physical Functions have been enabled, the incorrect settings are read-only. In such cases, you must reset the adapter to its default settings (see [Default Adapter Settings on page 270](#)). This returns the adapter to a single Physical Function mode, and removes all VLAN tags. You can then use the Boot Manger to make the settings that you require.

## 8.8 Default Adapter Settings

Resetting an adapter does not change the boot ROM image. To reset an adapter to its default settings:

- 1 Start or re-start the server and when prompted, press **Ctrl+B**. The Solarflare Boot Manager is displayed.

```

Solarflare Boot Manager (v5.2.0.1004)
Select Adapter

Base MAC address  PCI  Boot image
| 00-0f-53-4c-e5-e0  04:00.0  OptionROM & UEFI |
| 00-0f-53-5c-ff-a0  05:00.0  OptionROM & UEFI |

Controller:      Solarflare Flareon Ultra 2000 Series 10/25G Adapter

Up,Down Arrow to select | SPACE,+,- to change | ESC to exit | F1=help

```

- 2 Use the arrow keys to highlight the adapter you want to restore and press *Enter*. The **Adapter Menu** is displayed.

```

Solarflare Boot Manager (v5.2.0.1004)
Adapter Menu

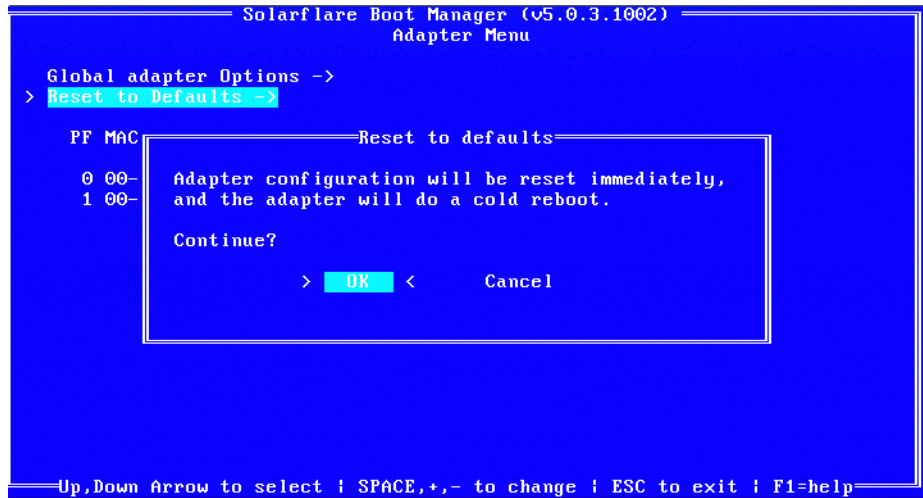
> Global adapter Options ->
Reset to Defaults ->

PF MAC          PCI  Boot type
0 00-0f-53-4c-e5-e0  04:00.0  PXE
1 00-0f-53-4c-e5-e1  04:00.1  PXE

Up,Down Arrow to select | SPACE,+,- to change | ESC to exit | F1=help

```

- Use the arrow keys to highlight **Reset to Defaults** and press *Enter*. The **Reset to Defaults** confirmation is displayed.



- Use the arrow keys to highlight **OK** and press *Enter*. The settings are reset to the defaults.

Table 42 lists the various adapter settings and their default values.

**Table 42: Default Adapter Settings**

Setting	Default Value
Boot Image	OptionROM & UEFI
Link up delay	5 seconds
Banner delay	2 seconds
Boot skip delay	5 seconds
Boot Type	PXE
MPIO attempts	
MSIX Limit	32

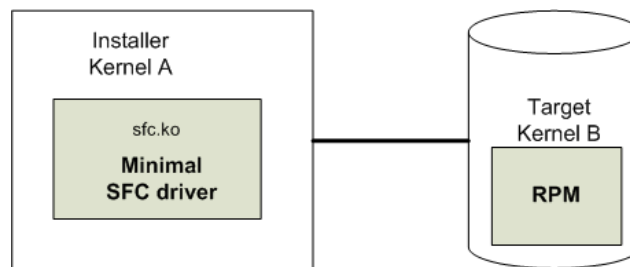
# 9

## Unattended Installations

### Building Drivers and RPMs for Unattended Installation

Linux unattended installation requires building two drivers:

- A minimal installation Solarflare driver that only provides networking support. This driver is used for network access during the installation process.
- An RPM that includes full driver support. This RPM is used to install drivers in the resultant Linux installation.



**Figure 24: Unattended Installation RPM**

Figure 24 shows how the unattended installation process works.

- 1 Build a minimal Solarflare driver needed for use in the installation kernel (Kernel A in the diagram above). This is achieved by defining “sfc\_minimal” to rpmbuild. This macro disables hardware monitoring, MTD support (used for access to the adapters flash), I2C and debugfs. This results in a driver with no dependencies on other modules and allows networking support from the driver during installation.

```
# as normal user
$ mkdir -p /tmp/rpm/BUILD
$ rpm -i sfc-<ver>-1.src.rpm
$ rpmbuild -bc -D 'sfc_minimal=1' -D 'kernel=<installer kernel>' \
  /tmp/rpm/SPECS/sfc.spec
```

- 2 The Solarflare minimal driver `sfc.ko` can be found in `/tmp/rpm/BUILD/sfc-<ver>/linux_net/sfc.ko`. Integrate this minimal driver into your installer kernel, either by creating a driver disk incorporating this minimal driver or by integrating this minimal driver into `initrd`.
- 3 Build a full binary RPM for your Target kernel and integrate this RPM into your Target (Kernel B).

## Driver Disks for Unattended Installations

Table 43 below identifies the various stages of an unattended installation process:

**Table 43: Installation Stages**

In Control	Stages of Boot	Setup needed
BIOS	PXE code on the adapter runs.	Adapter must be in PXE boot mode. See <a href="#">Solarflare Boot Manager on page 257</a> .
SF Boot ROM (PXE)	DHCP request from PXE (SF Boot ROM).	DHCP server filename and next-server options.
SF Boot ROM (PXE)	TFTP request for filename to next-server, e.g. pxelinux.0	TFTP server.
pxelinux	TFTP retrieval of pxelinux configuration.	pxelinux configuration on TFTP server.
pxelinux	TFTP menu retrieval of Linux kernel image initrd.	pxelinux configuration Kernel, kernel command, initrd
Linux kernel/installer	Installer retrieves kickstart configuration, e.g. via HTTP.	Kickstart/AutoYaST configuration.
Target Linux kernel	kernel reconfigures network adapters.	DHCP server.

## 9.1 Unattended Installation - Red Hat Enterprise Linux

Documentation for preparing for a Red Hat Enterprise Linux network installation can be found at:

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Installation\\_Guide/](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/)

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Installation\\_Guide/](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Installation_Guide/)

The prerequisites for a Network Kickstart installation are:

- Red Hat Enterprise Linux installation media.
- A Web server and/or FTP Server for delivery of the RPMs that are to be installed.
- A DHCP server for IP address assignments and to launch PXE Boot.
- A TFTP server for download of PXE Boot components to the machines being kickstarted.
- The BIOS on the computers to be Kickstarted must be configured to allow a network boot.
- A Boot CD-ROM or flash memory that contains the kickstart file or a network location where the kickstart file can be accessed.
- A Solarflare driver disk.

Unattended Red Hat Enterprise Linux installations are configured with Kickstart. The documentation for Kickstart can be found at:

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Installation\\_Guide/ch-kickstart2.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/ch-kickstart2.html)

[https://access.redhat.com/documentation/en-US/Red\\_Hat\\_Enterprise\\_Linux/7/html/Installation\\_Guide/chap-kickstart-installations.html](https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Installation_Guide/chap-kickstart-installations.html)

To install Red Hat Enterprise you need the following:

- 1** A modified `initrd.img` file with amended `modules.alias` and `modules.dep` which incorporates the Solarflare minimal driver for the installation kernel.

Find current aliases with the `modinfo` command:

```
modinfo sfc | grep alias
```

Then add the aliases found to the `modules.alias` file:

```
pci:v00001924d00001A03sv*sd*bc*sc*i*
pci:v00001924d00000A03sv*sd*bc*sc*i*
pci:v00001924d00001923sv*sd*bc*sc*i*
pci:v00001924d00000923sv*sd*bc*sc*i*
pci:v00001924d00001903sv*sd*bc*sc*i*
pci:v00001924d00000903sv*sd*bc*sc*i*
pci:v00001924d00000813sv*sd*bc*sc*i*
pci:v00001924d00000803sv*sd*bc*sc*i*
```



**2** Identify the driver dependencies using the modinfo command:

```
modinfo ./sfc.ko | grep depends
depends:    i2c-core,mii,hwmon,hwmon-vid,i2c-algo-bit mtdcore mtdpart
```

All modules listed as depends must be present in the initrd file image. In addition the user should be aware of further dependencies which can be resolved by adding the following lines to the modules.dep file:

```
sfc: i2c-core mii hwmon hwmon-vid i2c-algo-bit mtdcore mtdpart1
i2c-algo-bit: i2c-core
mtdpart: mtdcore
```

**3** A configured kickstart file with the Solarflare Driver RPM manually added to the %Post section. For example:

```
%post

/bin/mount -o ro <IP Address of Installation server>:<path to
location directory containing Solarflare RPM> /mnt
/bin/rpm -Uvh /mnt/<filename of Solarflare RPM>
/bin/umount /mnt
```

## 9.2 Unattended Installation - SUSE Linux Enterprise Server

Unattended SUSE Linux Enterprise Server installations are configured with AutoYaST. The documentation for AutoYaST can be found at:

[https://www.suse.com/documentation/sles11/book\\_automast/data/book\\_automast.html](https://www.suse.com/documentation/sles11/book_automast/data/book_automast.html)

[https://www.suse.com/documentation/sles-12/book\\_automast/data/book\\_automast.html](https://www.suse.com/documentation/sles-12/book_automast/data/book_automast.html)

The prerequisites for a Network AutoYaST installation are:

- SUSE Linux Enterprise installation media.
- A DHCP server for IP address assignments and to launch PXE Boot.
- A NFS or FTP server to provide the installation source.
- A TFTP server for the download of the kernel boot images needed to PXE Boot.
- A boot server on the same Ethernet segment.
- An install server with the SUSE Linux Enterprise Server OS.
- An AutoYaST configuration server that defines rules and profiles.
- A configured AutoYaST Profile (control file).

---

1. For Red Hat Enterprise Linux from version 5.5 add `mdio` to this line.

## Further Reading

- SUSE Linux Enterprise Server remote installation:  
[https://www.suse.com/documentation/sles11/book\\_sle\\_deployment/data/cha\\_deployment\\_remoteinst.html](https://www.suse.com/documentation/sles11/book_sle_deployment/data/cha_deployment_remoteinst.html)  
[https://www.suse.com/documentation/sles-12/book\\_sle\\_deployment/data/cha\\_deployment\\_remoteinst.html](https://www.suse.com/documentation/sles-12/book_sle_deployment/data/cha_deployment_remoteinst.html)
- SUSE install with PXE Boot:  
[https://www.suse.com/documentation/sles11/book\\_sle\\_deployment/data/sec\\_deployment\\_remoteinst\\_boot.html#sec\\_deployment\\_remoteinst\\_boot\\_pxe](https://www.suse.com/documentation/sles11/book_sle_deployment/data/sec_deployment_remoteinst_boot.html#sec_deployment_remoteinst_boot_pxe)  
[https://www.suse.com/documentation/sles-12/book\\_sle\\_deployment/data/sec\\_deployment\\_remoteinst\\_boot.html#sec\\_deployment\\_remoteinst\\_boot\\_pxe](https://www.suse.com/documentation/sles-12/book_sle_deployment/data/sec_deployment_remoteinst_boot.html#sec_deployment_remoteinst_boot_pxe)  
[http://www.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5.4/html/Deployment\\_Guide/s2-modules-bonding.html](http://www.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5.4/html/Deployment_Guide/s2-modules-bonding.html)
- RHEL6:  
[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/html/Deployment\\_Guide/s2-networkscripts-interfaces-chan.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/html/Deployment_Guide/s2-networkscripts-interfaces-chan.html)
- SLES:  
[http://www.novell.com/documentation/sles11/book\\_sle\\_admin/data/sec\\_basicnet\\_yast.html#sec\\_basicnet\\_yast\\_netcard\\_man](http://www.novell.com/documentation/sles11/book_sle_admin/data/sec_basicnet_yast.html#sec_basicnet_yast_netcard_man)