NVIDIA WinOF-2 Documentation v23.7

Exported on Oct/23/2023 03:45 PM

Table of Contents

Overview 4
Software Download 4
Document Revision History 4
Release Notes 5
General Support 5
WinOF-2 Package Contents 5
Tested Operating System 6
Upgrade/Downgrade Matrix 7
Certifications 7
Supported Network Adapter Cards and MFT Tools 8
Changes and New Features 9
Backward Compatibility 10
Bug Fixes in This Version 10
Known Issues 11
SR-IOV Support Limitations 17
User Manual 18
Intended Audience 18
Installation and Initialization 19
Downloading WinOF-2 Driver 19
Installing WinOF-2 Driver 19
Installation Results 27
Uninstalling WinOF-2 Driver 27
Extracting Files Without Running Installation 28
Firmware Upgrade 30
Booting Windows from an iSCSI Target or PXE 30
Features Overview and Configuration 34
BIOS Settings Configuration 35
General Capabilities 35
Ethernet Network 42
InfiniBand Network 81
Storage Protocols 86
Virtualization 87

Configuring the Driver Registry Keys 116
Network Direct Interface 133
Performance Tuning 136
Adapter Cards Counters 141
Resiliency 162
RDMA Capabilities 169
NVIDIA BlueField SmartNIC Mode 170
RShim Drivers and Usage 172
Utilities 179
Fabric Performance Utilities 179
Management Utilities 183
Snapshot Utility 192
Troubleshooting 193
General Related Troubleshooting 194
System Configuration Related Troubleshooting 194
Installation Related Troubleshooting 194
InfiniBand Related Troubleshooting 195
Ethernet Related Troubleshooting 196
Performance Related Troubleshooting 197
Virtualization Related Troubleshooting 198
Reported Driver Events 198
Extracting WPP Traces 205
Appendixes 205
Windows MPI (MS-MPI) 206
Document History 209
Release Notes History 209
Release Notes Change Log History 209
Bug Fixes History 218
User Manual Revision History 240

Overview

Windows OS Host controller driver for Cloud, Storage and High-Performance computing applications utilizing field-proven RDMA and Transport Offloads

NVIDIA® Windows distribution includes software for database clustering, Cloud, High Performance Computing, communications, and storage applications for servers and clients running different versions of Windows OS. This collection consists of drivers, protocols, and management in simple ready-to-install MSIs.

NVIDIA® WinOF-2 is the Windows driver for NVIDIA® ConnectX®-4 Lx and onwards adapter cards. It does not support earlier NVIDIA Networking adapter generations.

The documentation here relates to WinOF-2:

- Release Notes
- User Manual

Software Download

Please visit WinOF-2 webpage.

Document Revision History

A list of the changes made to the User Manual are provided in User Manual Revision History.

Release Notes

Release Notes Update History

Version	Date	Description
23.7.50000	August 10, 2023	Initial release of this Release Notes version. This version introduces Changes and New Features and Bug Fixes.

These are the release notes of NVIDIA® WinOF-2 Ethernet and InfiniBand drivers.

> ⚠ Please note that WinOF-2 driver supports NVIDIA® ConnectX-4 Lx onwards adapter cards only.

Release Notes contain the following sections:

- General Support
- Changes and New Features
- Bug Fixes in This Version
- Known Issues

General Support

WinOF-2 Package Contents

The WinOF-2 package contains the following components:

- Diagnostic Tools
- Documentation
- Management Tools
- Performance Tools
- Drivers

	Drivers	Version
Mlx5 Driver Package	- Mlx5.sys - Mlx5.inf - Mlx5.cat - Mlx5ui.dll	23.7.26138
mlx5 DevX Package	- mlx5devx.dll	23.7.26138
MUX Driver Package **Note:** Windows Client 10 onwards only.	- Mlx5mux.sys - Mlx5mux.dll - Mlx5mux.inf - Mlx5mux.cat - Mlx5muxp.inf - Mlx5muxp.cat	23.7.26138

	Drivers	Version
Bluefield Management Drivers **Note:** Windows Server 2016 onwards and Windows Client 10 only.	- Mlxrshimbus.sys - Mlxrshimbus.inf - Mlxrshimbus.cat - Mlxrshimeth.sys - Mlxrshimeth.inf - Mlxrshimcom.cat - Mlxrshimcom.sys - Mlxrshimcom.inf - Mlxrshimcom.cat	23.7.26138

Tested Operating System

The following describes the tested operating systems and their roles in a virtualization environment.

Virtualization Mode	Supported Host OS	Supported Guest OS
None	Windows Server 2012 R2* Windows Server 2016 Windows Server 2019 Windows Server 2022 Windows 8.1 Client (64 bit only) Windows 10 Client 1809 / 21H2 / 22H2 (64 bit only) Windows 11 Client / 22H2 (64 bit only)	N/A
VMQ	Windows Server 2012 R2 Windows Server 2016 Windows Server 2019 Windows Server 2022	Windows Server 2012 R2 Windows Server 2016 Windows Server 2019 Windows Server 2022 Windows 8.1 Client (64 bit only) Windows 10 Client 1809 (64 bit only) Windows 10 Client 21H2 (64 bit only) Windows 10 Client 22H2 (64 bit only) Windows 11 Client (64 bit only) Windows 11 Client 22H2 (64 bit only)

SR-IOV (Ethernet)	Windows Server 2016 Windows Server 2019 Windows Server 2022	Windows Server 2016 Windows Server 2019 Windows Server 2022 Windows 10 Client 1809 (64 bit only) Windows 10 Client 21H2 (64 bit only) Windows 10 Client 22H2 (64 bit only) Windows 11 Client (64 bit only) Windows 11 Client 22H2 (64 bit only) Ubuntu 18.04 Ubuntu 20.04 Ubuntu 22.04 SLES15.3 SP3 5.3.18-57-default CentOS/RHEL 7.9 Centos/RHEL 8.7 Centos/RHEL 9.1 RHEL8.4 upstream FreeBSD 13.0-STABLE FreeBSD 13.1-RELEASE FreeBSD 14-0-CURRENT-x64-15565e0a217-257277
SR-IOV (InfiniBand)	Windows Server 2016 Windows Server 2019 Windows Server 2022	Windows Server 2016 Windows Server 2019 Windows Server 2022 Windows 10 Client 1809 (64 bit only) Windows 10 Client 21H2 (64 bit only) Windows 10 Client 22H2 (64 bit only) Windows 11 Client (64 bit only) Windows 11 Client 22H2 (64 bit only) Ubuntu 20.04 5.13.0-1017-azure + OFED 23.07 Ubuntu 22.04 + OFED 23.07 SLES15.3 SP3 5.3.18-57-default + OFED 23.07 CentOS/RHEL 8.7 + OFED 23.07 Centos/RHEL 9.1 + OFED 23.07 SLES15.3 SP3 5.3.18-57-default + OFED 23.07

* Windows Server 2012 R2 will no longer be supported after WinOF-2 v23.7.50000.

Upgrade/Downgrade Matrix

This section reflects which versions were tested and verified for upgrade and downgrade.

Target Version	Versions Verified for Upgrade/ Downgrade	Release Type	Release Date
23.7.50000 GA (July 2023)	23.4.50020	GA	April 2023
	3.10.52010	GA-LTS	June 2023

Certifications

The following describes the driver's certification status per operating system.

Operating System	Logo Certification	SDDC Premium Certification
Windows Client 8.1	Certified	N/A
Windows 10 Client 1809	Certified	N/A
Windows 10 Client 21H2	Certified	N/A
Windows 10 Client 22H2	Certified	N/A
Windows 11 Client	Certified	N/A
Windows 11 Client 22H2	Certified	N/A
Windows Server 2012 R2	Certified	N/A
Windows Server 2016	Certified	Certified
Windows Server 2019	Certified	Certified

Operating System	Management AQ Certification	Storage AQ Certification	Compute AQ Certification
Windows Server 2022	Certified	Premium	Premium

⚠ This section is updated in accordance with the certifications obtainment.

⚠ The RSHIM drivers are certified only for Windows Server 2016 and above Operating Systems and Windows Client 10 / 1809 and above Operating Systems.

Supported Network Adapter Cards and MFT Tools

Supported Network Adapter Cards

NVIDIA® WinOF-2 supports the following NVIDIA® network adapter cards:

NICs	Supported Protocol	Supported Link Speed
NVIDIA® ConnectX®-7	InfiniBand	HDR, NDR200 and NDR
	Ethernet	1GbE, 10GbE, 25GbE, 40GbE, 50GbE, 100GbE, 200GbE
NVIDIA® BlueField®-3 SmartNIC	InfiniBand	EDR, HDR100, HDR, NDR, NDR200
	Ethernet	25, 40, 50, 100, 200 and 400GbE
NVIDIA® BlueField®-2 SmartNIC	InfiniBand	QDR, FDR, EDR, HDR100, HDR
	Ethernet	1, 10, 25, 40, 50, 100 and 200GbE
NVIDIA® ConnectX®-6 Lx	Ethernet	10, 25, and 50GbE
NVIDIA® ConnectX®-6 Dx	Ethernet	10, 25, 40, 50, 100 and 200GbE
NVIDIA® ConnectX®-6	Ethernet	10, 25, 40, 50, 100 and 200GbE
	InfiniBand	SDR, FDR, EDR and HDR
NVIDIA® ConnectX®-5/Ex	Ethernet	10, 25, 40, 50 and 100GbE
	InfiniBand	QDR, FDR and EDR
NVIDIA® ConnectX®-4 Lx	Ethernet	10, 25, 40, and 50GbE

Firmware Versions

WinOF-2 is tested with the following firmware for NVIDIA® NICs:

> ⚠ Firmware versions listed are the minimum supported versions.

NICs	Recommended Firmware Rev.	Additional Firmware Rev. Supported
NVIDIA ConnectX-7	28.38.1002	28.37.1014
BlueField-3	32.38.1002	32.37.1306
BlueField-2 integrated ConnectX-6 Dx Adapter	24.38.1002	24.37.1014
BlueField integrated ConnectX-5 Adapter	18.33.1048	18.33.1048
NVIDIA ConnectX-6 Lx	26.38.1002	26.37.1014
NVIDIA ConnectX-6 Dx	22.38.1002	22.37.1014
NVIDIA ConnectX-6	20.38.1002	20.37.1014
NVIDIA ConnectX-5 / ConnectX-5 Ex	16.35.2000	N/A
NVIDIA ConnectX-4 Lx	14.32.1010	N/A

NVIDIA MFT - Firmware Versions

WinOF-2 is compatible with the following MFT versions:

Product	Recommended Rev.	Additional Rev. Supported
MFT	4.25.0	4.24.0

Changes and New Features

> ⚠ Starting v23.4, WinOF-2 will adopt the following versioning naming scheme: YY.MM-x.x.x.x (i.e., Year.Month-x.x.x.x).
> The existing release branches will maintain the existing naming scheme.

Category	Description
Rev 23.7.50000 (DRV 23.7.26138)	
Installation Package	Windows Server 2012 R2 will no longer be supported after WinOF-2 v23.7.50000.
Mlx5Cmd: NIC Health Monitor	The NIC Health Monitor is an external tool used to check and monitor the health of the NIC by analyzing the firmware and the diagnostic counters previously collected by the user. For further information, see NIC Health Monitor.

Category	Description
Rev 23.7.50000 (DRV 23.7.26138)	
Mlx5Cmd: AutoLogger	The AutoLogger is a debuggability capability implemented as part of Mlx5Cmd, that automatically collects logs until it detects a trigger defined by the user. For further information, see AutoLogger.
Counters	Added new "Mellanox WinOF-2 Transmit Datapath Counters". For further information, see Adapter Cards Counters.
Registry Keys	Updated the default values of the following registry keys: • *PriorityVLANTag • DumpMeNowDumpMask • MaxCallsToNdisIndicate • RelaxedOrderingWrite • TxIntModeration For further information, see Configuring the Driver Registry Keys.
Bug Fixes	See Bug Fixes.

Backward Compatibility

This section will include all the features that their backward compatibility is broken at a certain release.

Category/Feature	Description
mlxndperf	As of WinOF-2 v23.4, the "`-estatLatencywas`" is no longer supported in the mlxndperf tool. This argument is now replaced by the "`-latency`" argument.
DriverVersion Utility	As of WinOF-2 v2.90, the "`mlx5cmd -driverversion`" command presents the OS build number + Server\Client information instead of presenting the OS name. Meaning, when querying the "`driverversion`" of the Virtual Function, the OS version format depends on the VF version. If the VF driver version is < 2.90, it will show the OS Name, otherwise it will show the OS build number + Server\Client information.

Bug Fixes in This Version

For a list of old fixes, please see Bug Fixes History.

Internal Ref.	Issue
3466737	**Description:** Fixed an issue that resulted in driver upgrade failure on systems with more than 3 devices. To resolve the issue, the mstdump generation process via the PCI configuration space when in teardown was stopped to prevent a very slow teardown which caused the upgrade timeout.
	Keywords: Upgrade, mstdump
	Detected in version: 23.4.50020
	Fixed in version: 23.7.50000

Internal Ref.	Issue
3478979	**Description:** Fixed an issue that resulted in missing section of the capabilities when querying the FwCaps of a VF from the Host using mlx5cmd -fwcaps.
	Keywords: FwCaps
	Detected in version: 23.4.50020
	Fixed in version: 23.7.50000
3464588	**Description:** Removed false error message which sometimes appeared in machines with several BlueField cards.
	Keywords: BlueField
	Detected in version: 23.4.50020
	Fixed in version: 23.7.50000
3483336	**Description:** Improved mlxndperf tool's latency tests results.
	Keywords: mlxndperf, performance tests
	Detected in version: 23.4.50020
	Fixed in version: 23.7.50000
3472624	**Description:** Fixed an issue that occasionally resulted in truncated printed DPU log.
	Keywords: RshimCmd, "DPU log"
	Detected in version: 23.4.50020
	Fixed in version: 23.7.50000

Known Issues

For the list of old Know Issues, please see the relevant Release Notes version.

Internal Ref.	Issue
3554731	**Description:** On Bluefield devices ,network adapter is disabled when performing cold boot right after restarting of the DPU.
	Workaround: To resolve the issue perform one of the following: • Wait for the DPU to load and link up before executing cold boot. • Cold boot and also restart the DPU so a DPU restart is not required in case of cold boot.
	Keywords: BlueField, cold boot, restart
	Detected in version: 23.7.50000
3178409	**Description:** Disabling of VF's SL-DiFF from the driver on Bluefield devices is not supported.
	Workaround: N/A
	Keywords: BlueField SL-DiFF
	Detected in version: 23.7.50000
3035275	**Description:** VMQoS statistics counter does not count RDMA traffic running on the PF.
	Workaround: N/A
	Keywords: VMQoS statistics counter, RDMA
	Detected in version: 3.20.50010
3316447	**Description:** When using a firmware version between xx.34.1000 and xx.36.1000, the driver reports the wrong number of SQs (adds 1).

Internal Ref.	Issue
	Workaround: Update the firmware version to xx.36.1000 or above.
	Keywords: VMQoS SR-IOV
	Detected in version: 3.20.50010
3240702	**Description:** Anti-spoofing counters are not supported on BlueField-2 in Smart mode.
	Workaround: Arm user can add anti-spoofing rule via Linux in DPU as Arm is the manager of the eSwitch in embedded model.
	Keywords: Anti-spoofing counters, Smart mode, Embedded mode, DPU
	Detected in version: 3.10.50000
3046630	**Description:** The device will stay associated with the currently installed driver when downgrading to a package that does not support the device.
	Workaround: Uninstall the device and scan for new hardware. The device will appear as a unknown device.
	Keywords: Installation, downgrade
	Detected in version: 3.10.50000
3150126	**Description:** On ConnectX-4 and ConnectX-4 Lx, when using Hardware QoS Offload revision 2 and running RDMA from the VF, only the RX inbound counters will be increased in the RDMA activity counters, and the bytes and frames counters will be the same. **Note:** There is no functional impact on the actual traffic, only wrong counter value.
	Workaround: N/A
	Keywords: Hardware QoS Offload, RDMA activity counters
	Detected in version: 3.0.50000
3040551	**Description:** The `Set/Get/Enable-NetAdapterEncapsulatedPacketTaskOffload` powersehll commands are not supported by default when working in NIC mode on NVIDIA Bluefield-2 DPU. These commands will fail in this mode because the encapsulation registry keys (`*EncapsulatedPacketTaskOffload,*EncapsulatedPacketTaskOffloadNvgre,*EncapsulatedPacketTaskOffloadVxlan`) are missing. However, as the encapsulation is still enabled by default, the user can configure encapsulation without these commands.
	Workaround: Manually add these keys, however, the keys must be removed or disabled when switching to Smart NIC mode. For instruction on how to add the keys please see Configuring the Driver Registry Keys.
	Keywords: NVIDIA BlueField-2, NIC Mode,NVGRE,VXLAN
	Detected in version: 2.90.50010
2891364	**Description:** When working with ConnectX-4 or ConnectX-4 Lx dual-port adapter cards, the value of the `"EnableVmQoSOffloadRev2"` registry key must be the same on both ports, otherwise one port will failed to load.
	Workaround: Set the same value for the `"EnableVmQoSOffloadRev2"` registry key on both ports.
	Keywords: EnableVmQoSOffloadRev2, VMQoS
	Detected in version: 2.80.50000
2854943	**Description:** When using Hardware QoS Offload Rev 2 when in VMQ mode and the VM traffic is mapped to TC != 0, the rate limit will be enforced only on NVIDIA ConnectX-4 Lx. For all other devices, the rate limit will be enforced only for TC = 0. **Note:** When in SR-IOV mode, it works for all devices as expected.
	Workaround: N/A
	Keywords: Hardware QoS Offload, VMQ

Internal Ref.	Issue
	Detected in version: 2.80.50000
2302247	**Description:** mlx5cmd exposes the system GUID information of a NVIDIA BlueField Virtual Function irrespective of its trusted state.
	Workaround: N/A
	Keywords: mlx5cmd, VF, NVIDIA BlueField, GUID
	Detected in version: 2.80.50000
2491846	**Description:** As oversubscription of QP parameters (entries and depth) is allowed, it could cause run-time failure when running out of resources.
	Workaround: N/A
	Keywords: QP creation
	Detected in version: 2.70.50000
2380684	**Description:** Although the IPOIB failover team gets the correct DHCP address when first created, if the team is disabled and then enabled, Windows requests and rejects the DHCP address as BAD_ADDRESS.
	Workaround: When the issue is seen, restart the secondary member(s) of the team.
	Keywords: IPOIB teaming, DHCP
	Detected in version: 2.70.50000
2603423	**Description:** When in ETH mode, setting the MTU (JumboPacket) lower than 1514, results in Received Packets Error counters not being increased when receiving packets with larger frame size but less or equal to 1518 bytes (Like ping with data size of 1476).
	Workaround: N/A
	Keywords: MTU, traffic, counters
	Detected in version: 2.70.50000
2374101	**Description:** After upgrade, *PtpHardwareTimestamp remains enabled. When *PtpHardwareTimestamp is enabled, UDP performance feature (URO) wil be automatically disabled. This is an OS limitation, if you do not use the HW time stamp feature, it is recommended to disable this feature by setting *PtpHardwareTimestamp to 0.
	Workaround: Disable HW timestamping. by setting *PtpHardwareTimestamp to 0.
	Keywords: *PtpHardwareTimestamp, UDP performance feature ,URO
	Detected in version: 2.60.50000
2306807	**Description:** When the Decouple VmSwitch protocol is enabled, VM's friendly given name is not displayed when running the `"Get-NetAdapterSriovVf"` and `"mlnx5hpccmd -DriverVersion"` commands.
	Workaround: N/A
	Keywords: HPC, SR-IOV
	Detected in version: 2.60.50000
2205722	**Description:** WinOF-2 driver does not support IB MTU lower than 614.
	Workaround: N/A
	Keywords: IB MTU
	Detected in version: 2.60.50000
2180714	**Description:** In case the user configs TCP to priority 0 with no VlanID, the packets will be sent without a VLAN header since the miniport cannot distinguish between priority 0 with VlanId 0 and no Vlan tag.
	Workaround: N/A
	Keywords: TCP QOS

Internal Ref.	Issue
	Detected in version: 2.50.50000
2216232	**Description:** As ConnectX-5 adapter cards do not create counters for RX PACKET MARKED PCIe BUFFERS, its value will be 0.
	Workaround: N/A
	Keywords: ECN Marking
	Detected in version: 2.50.50000
2243909	**Description:** The driver to sends a wrong CNP priority counter while running RDMA.
	Workaround: Change the CNP priority using mlxconfig.
	Keywords: RDMA, CNP
	Detected in version: 2.50.50000
2118837	**Description:** Performance degradation might be experienced during UDP traffic when using a container networking and the UDP message size is larger than the MTU size .
	Workaround: N/A
	Keywords: Nested Virtualization, container networking
	Detected in version: 2.50.50000
2137585	**Description:** While working in IPoIB mode and *JumboPacket is set in the range of [256, 614], the driver issues a warning event log message (Event ID: 25). This is a false alarm and could be ignored.
	Workaround: N/A
	Keywords: JumboPacket
	Detected in version: 2.50.50000
2148077	**Description:** Explicitly disabling the *NetworkDirect key when using the HyperV mode, disables NDSPI as well as the NDK.
	Workaround: Enable NetworkDirect (ND).
	Keywords: ND, HyperV
	Detected in version: 2.50.50000
2117964	**Description:** A delay in connection establishment might be experienced when the ND application is started immediately after restarting the adapter card. This scenario occurs because the ND application requires the ARP table to find the destination MAC and generate the ARP request.
	Workaround: Use static ARP. Ping the system before starting the ND application.
	Keywords: ND, RDMA
	Detected in version: 2.40.51000
2117636	**Description:** On a native setup, when setting JumboPacket to be less than 1514, the Large Receive Offload (LRO) feature might be disabled, and all its counters will not be valid.
	Workaround: N/A
	Keywords: LRO, RSC
	Detected in version: 2.40.51000
2083686	**Description:** As PCIe Write Relaxed Ordering is enabled by default, some older Intel processors might observe up to 5% packet loss in high packet rate and small packets. (https://lore.kernel.org/patchwork/patch/820922/)
	Workaround: Disable the Relaxed Ordering Write option by setting the RelaxedOrderingWrite registry key to 0 and restart the adapter.
	Keywords: PCIe Write Relaxed Ordering
	Detected in version: 2.40.50000

Internal Ref.	Issue
1763379	**Description:** On Windows Server 19H1, running "netstat -axn" when RDMA is enabled and a vNIC is present, results in RDMA being disabled on the port with the VMswitch.
	Workaround: N/A
	Keywords: VMSwitch, RDMA, Windows Server 2019
	Detected in version: 2.40.50000
1908862	**Description:** When running RoCE traffic with a different RoceFrameSize configuration, and the fabric (jumbo packet size) is large enough, the MTU will be taken from the initiator even when it supports larger size than the server.
	Workaround: N/A
	Keywords: RoCE, MTU
	Detected in version: 2.40.50000
1846356	**Description:** The driver ignores the value set by the "*NumVfs" key. The maximal number of VFs is the maximal number of VFs supported by the hardware.
	Workaround: N/A
	Keywords: SR-IOV NUMVFs
	Detected in version: 2.30.50000
1598716	**Description:** Issues with the OS' "SR-IOV PF/VF Backchannel Communication" mechanism in Windows Server 2019 Hyper-V, effect VF-Counters functionality as well.
	Workaround: N/A
	Keywords: Mellanox WinOF-2 VF Port Traffic, VF-Counters
	Detected in version: 2.30.50000
1702662	**Description:** On WIndows Server 2019, the physical media type of the IPoIB NIC will be 802.3 and not InfiniBand.
	Workaround: Use the mlx5cmd tool (`"mlx5cmd -stat"`) which is part of the driver package to display the lin_layer type.
	Keywords: Windows Server 2019, IPoIB NdisPhysicalMedium
	Detected in version: 2.20
1718201	**Description:** Heavy traffic causes Sniffer' limit file to be the same as the buffer size (100M by default).
	Workaround: N/A
	Keywords: Sniffer, heavy traffic
	Detected in version: 2.20
1576283	**Description:** When working with SR-IOV in Windows Server 2019, the vNIC that is working in SR-IOV mode status will be displayed as "Degraded (SR-IOV not operational)" although the SR-IOV VF is fully operational. The message can be safely ignored.
	Workaround: N/A
	Keywords: SR-IOV IB, Windows Server 2019
	Detected in version: 2.10
1580985	**Description:** iSCSI boot over IPoIB is currently not supported.
	Workaround: N/A
	Keywords: iSCSI Boot, IPoIB
	Detected in version: 2.10
1536971	**Description:** The RscIPv4 and RscIPv6 keys' values are set to 0 for the host in Windows Server 2019. As the values for those keys are already written by the Inbox Driver in Windows Server 2019, they will not be changed when upgrading.

Internal Ref.	Issue
	Workaround: N/A
	Keywords: RscIPv4, RscIPv6, Windows Server 2019
	Detected in version: 2.10
1419597	**Description:** On servers with large number of VMs, (typically more than 40), after restarting the NIC on the host, VMs' IPv6 global address is not retrieved back from the DHCP.
	Workaround: Restart the NIC inside the VM.
	Keywords: VMQ, SR-IOV
	Detected in version: 2.10
1419597	**Description:** On servers with a large number of VMs (typically > 40) - after a NIC restart on the host, VMs' IPv6 global address cannot be retrieved from DHCP.
	Workaround: Restart Microsoft NIC inside the VM.
	Keywords: VM, IPv6 address, DHCP
	Detected in version: 2.0
1336097	**Description:** Due to an OID timeout, the miniport reset is executed.
	Workaround: Increase the timeout value in such way that 2 * CheckForHangTOInSeconds > Max OID time. For further information, refer to section General Registry Keys in the User Manual.
	Keywords: Resiliency
	Detected in version: 1.90
1310086	**Description:** Multicast packets are passed via to the VM the Hyper-V (even in SR-IOV VMs). As such, the Hyper-V can decide to drop the packets based on its specific policy. **Note:** This issue is only related to FreeBSD OSes.
	Workaround: N/A
	Keywords: Hyper-V OS
	Detected in version: 1.90
1154447	**Description:** Adding diagnostic counters to performance monitor might cause counters to get cleared every several seconds.
	Workaround: Change the time period between samples to more than 1 second.
	Keywords: Diagnostic Counters
	Detected in version: 1.90
1074589	**Description:** When PXE boot is using Flexboot, the IPoIB interface is not receiving the reserved address from the DHCP using GUID reservation.
	Workaround: To obtain the reserved address, use a 6-byte MAC address instead of the 8-byte client ID.
	Keywords: PXE boot, IPoIB, Flexboot, DHCP
	Detected in version: 1.80
917747	**Description:** VF driver initialization fails in case of bad MSIX mapping when running Windows Server 2012 R2 Hypervisor with Windows Server 2016 VM with more than a single core CPU. As a result, performance desegregation might occur.
	Workaround: Run either with one CPU core, or run with different Operating Systems.
	Keywords: SR-IOV
	Detected in version: 1.80
1170780	**Description:** The driver must be restarted in order to switch from RSS to NonRSS mode. Therefore, if a PowerShell command is used on a specific VM to an enabled/disabled VMMQ without restarting the driver, the RSS counters will keep increasing in Perfmon.

Internal Ref.	Issue
	Workaround: Restart the driver to switch to NonRSS mode.
	Keywords: RSS, NonRSS, VMMQ
	Detected in version: 1.80
1149961	**Description:** In RoCE, the maximum MTU of WinOF-2 (4k) is greater than the maximum MTU of WinOF (2k). As a result, when working with MTU greater than 2k, WinOF and WinOF-2 cannot operate together.
	Workaround: N/A
	Keywords: RoCE, MTU
	Detected in version: 1.80
1145421	**Description:** In IPoIB SR-IOV setup, in the Hyper-V Manager, the address appears as "SR-IOV enabled" instead of "SR-IOV active". This does not influence any activity or functionality.
	Workaround: N/A
	Keywords: IPoIB SR-IOV setup, Hyper-V
	Detected in version: 1.80
1145421	**Description:** In the "Network Connections" panel of Virtual Function (VF) in IPoIB SR-IOV setup, the Microsoft adapter may appear in addition to the NVIDIA® adapter. This does not influence any activity or functionality.
	Workaround: N/A
	Keywords: Network Connections, VF, IPoIB SR-IOV
	Detected in version: 1.80

SR-IOV Support Limitations

The below table summarizes the SR-IOV working limitations, and the driver's expected behavior in unsupported configurations.

WinOF-2 Version	NVIDIA® ConnectX®-4 Firmware Ver.	Adapter Mode		
		InfiniBand		Ethernet
		SR-IOV On	SR-IOV Off	SR-IOV On/Off
Earlier versions	Up to 12.16.1020	Driver will fail to load and show "Yellow Bang" in the device manager.		No limitations
1.50 and 1.60	Between 1x.16.1020 and 1x.19.2002 (IPoIB supported)	"Yellow Bang" unsupported mode - disable SR-IOV via mlxconfig	OK	No limitations
1.70 and onwards	1x.19.2002 and onwards (IPoIB supported)	OK	OK	No limitations

For further information on how to enable/disable SR-IOV, please refer to section Single Root I/O Virtualization (SR-IOV).

User Manual

This User Manual describes installation, configuration and operation of NVIDIA® WinOF-2 driver. features, performance, diagnostic tools, content and configuration. Additionally, this document provides information on various performance tools supplied with this version.

NVIDIA® WinOF-2 is composed of several software modules that contain InfiniBand and Ethernet drivers. It supports 10, 25, 40, 50 or 100 Gb/s Ethernet, and DDR, FDR, EDR, HDR, or NDR InfiniBand network ports. The port type and speed are determined upon boot based on card capabilities and user settings.

The NVIDIA® WinOF-2 driver release introduces the following capabilities:

- General capabilities:
  - Support for Single and Dual port Adapters
  - Receive Side Scaling (RSS)
  - Hardware Tx/Rx checksum offload
  - Large Send Offload (LSO)
  - UDP Segmentation Offload (USO)
  - Dynamic Interrupt Moderation
  - Support for MSI-X interrupts
  - Network Direct Kernel (NDK) with support for SMBDirect
  - Virtual Machine Queue (VMQ) for Hyper-V
  - Single Root I/O Virtualization (SR-IOV)
  - Receive Side Scaling
  - Checksum Offloads
  - Quality of Service (QoS)
    - Support for global flow control and Priority Flow Control (PFC)
    - Enhanced Transmission Selection (ETS)
- Ethernet capabilities:
  - Receive Side Coalescing (RSC)
  - Hardware VLAN filtering
  - RDMA over Converged Ethernet
    - RoCE MAC Based (v1)
    - RoCE over UDP (v2)
  - VXLAN
  - NDKPI v2.0, v3.0
  - VMMQ
  - PacketDirect Provider Interface (PDPI)
  - NVGRE hardware encapsulation task offload

For the complete list of Ethernet and InfiniBand Known Issues and Limitations, refer to the latest Release Notes.

Intended Audience

This manual is intended for system administrators responsible for the installation, configuration, management and maintenance of the software and hardware of Ethernet and InfiniBand adapter cards. It is also intended for application developers.

Installation and Initialization

This chapter describes WinOF-2 driver installation and initialization process.

The chapter contains the following sections:

- Downloading WinOF-2 Driver
- Installing WinOF-2 Driver
- Installation Results
- Uninstalling WinOF-2 Driver
- Extracting Files Without Running Installation
- Firmware Upgrade
- Booting Windows from an iSCSI Target or PXE

Downloading WinOF-2 Driver

To download the .exe file according to your Operating System, please follow the steps below:

1. Obtain the machine architecture.
   a. To go to the Start menu, position your mouse in the bottom-right corner of the Remote Desktop of your screen.
   b. Open a CMD console (Click Task Manager-->File --> Run new task and enter CMD).
   c. Enter the following command.

   ```
   echo %PROCESSOR_ARCHITECTURE%
   ```

   On an x64 (64-bit) machine, the output will be “AMD64”.
2. Go to the WinOF-2 web page at: https://www.nvidia.com/en-us/networking/ > Products > Software > InfiniBand Drivers (Learn More) > Nvidia WinOF-2.
3. Download the .exe image according to the architecture of your machine (see Step 1).
   The name of the .exe is in the following format: MLNX_WinOF2-<version>_<arch>.exe.

   Installing the incorrect .exe file is prohibited. If you do so, an error message will be displayed.
   For example, if you install a 64-bit .exe on a 32-bit machine, the wizard will display the following (or a similar) error message: “The installation package is not supported by this processor type. Contact your vendor”

Installing WinOF-2 Driver

The snapshots in the following sections are for illustration purposes only. The installation interface may slightly vary, depending on the used operating system.

> NVIDIA® WinOF-2 supports adapter cards based on NVIDIA® ConnectX®-4 Lx family and newer adapter IC devices only. If you have NVIDIA® ConnectX®-3 and NVIDIA® ConnectX®-3 Pro on your server, you will need to install WinOF driver.
> For details on how to install WinOF driver, please refer to WinOF User Manual.

This section provides instructions for two types of installation procedures, and both require administrator privileges:

- Attended Installation
  An installation procedure that requires frequent user intervention.
- Unattended Installation
  An automated installation procedure that requires no user intervention.

Attended Installation

The following is an example of an installation session.

1. Double click the .exe and follow the GUI instructions to install MLNX_WinOF2.
2. [Optional] Manually configure your setup to contain the logs option (replace “LogFile” with the relevant directory).

   ```
   MLNX_WinOF2_<revision_version>_All_Arch.exe /v"/l*vx [LogFile]"
   ```

   Example:

   ```
   MLNX_WinOF2-2_10_50000_All_x64.exe /v"/l*vx [LogFile]"
   ```

3. [Optional] If you do not want to upgrade your firmware version (i.e., MT_SKIPFWUPGRD default value is False).

   ```
   MLNX_WinOF2_<revision_version>_All_Arch.exe /v" MT_SKIPFWUPGRD=1"
   ```

4. [Optional] If you do not want to install the Rshim driver, run.

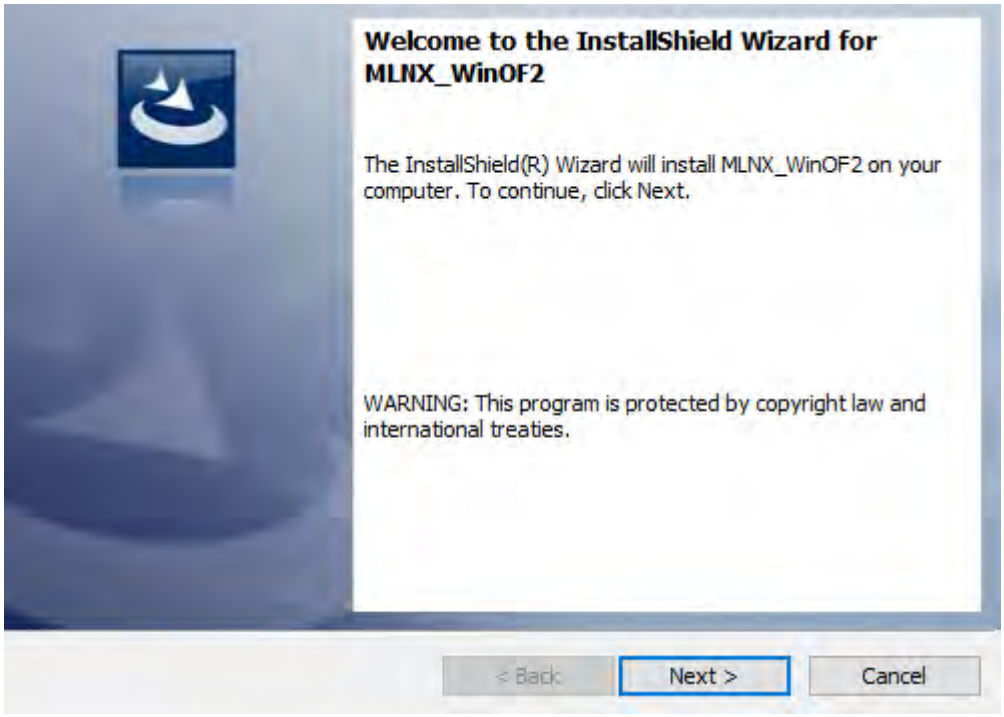
   ```
   MLNX_WinOF2_<revision_version>_All_Arch.exe /v" MT_DISABLE_RSHIM_INSTALL=1"
   ```

   > The Rshim driver installanion will fail if a prior Rshim driver is already installed. The following fail message will be displayed in the log:
   > "ERROR!!! Installation failed due to following errors: MlxRshim drivers installation disabled and MlxRshim drivers Installed, Please remove the following oem inf files from driver store: <oem inf list>"

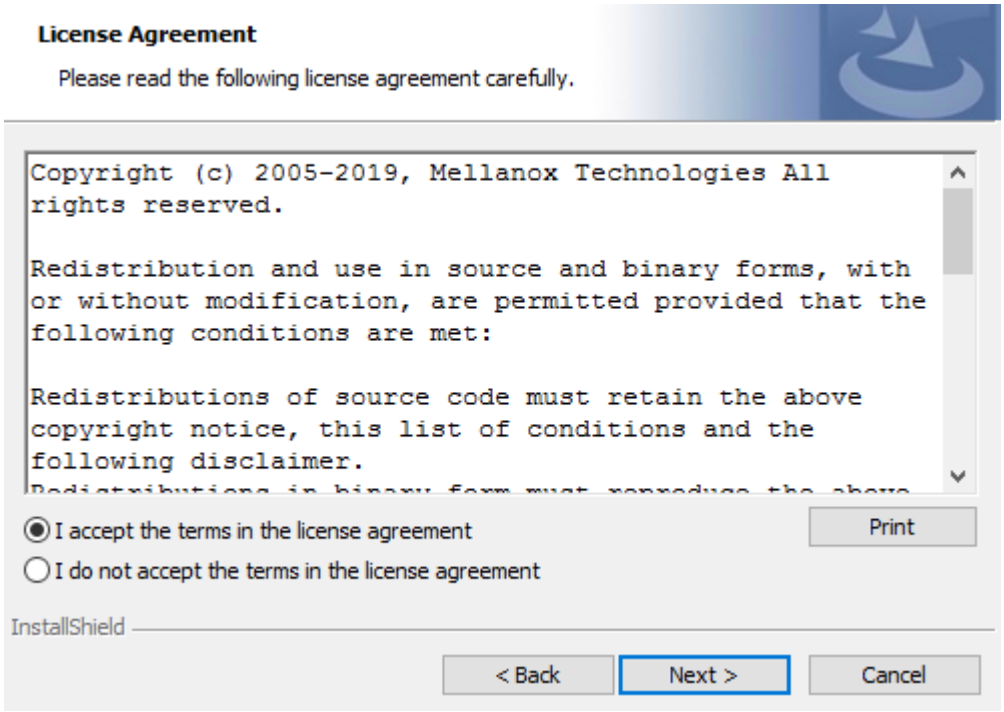
5. [Optional] If you want to skip the check for unsupported devices, run.

```
MLNX_WinOF2_<revision_version>_All_Arch.exe /v" SKIPUNSUPPORTEDDEVCHECK=1"
```

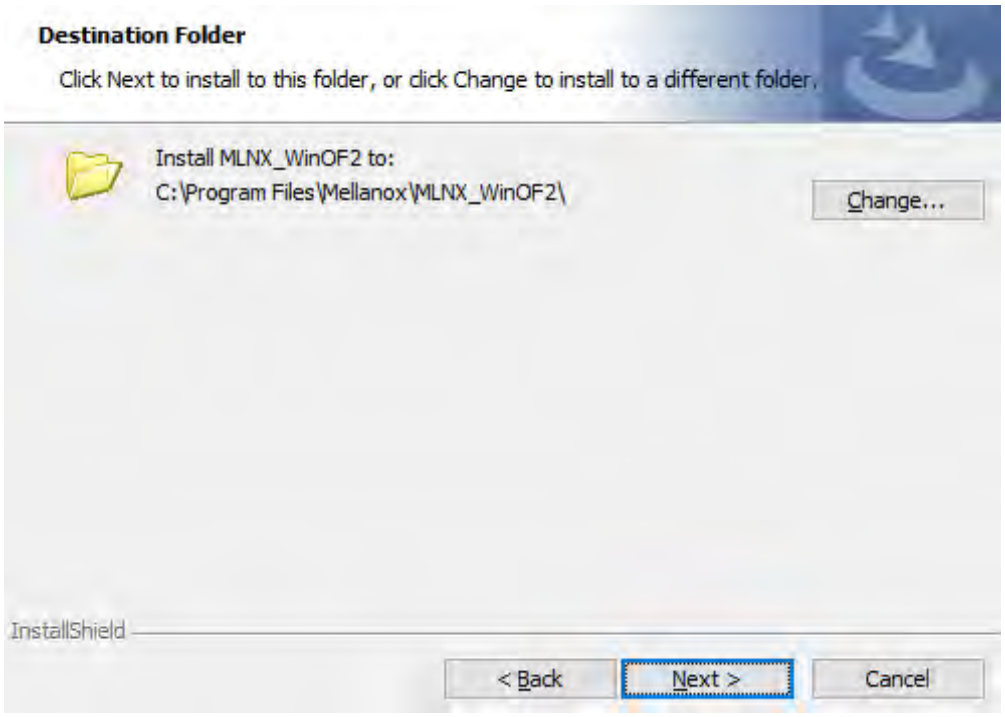
6. Click Next in the Welcome screen.



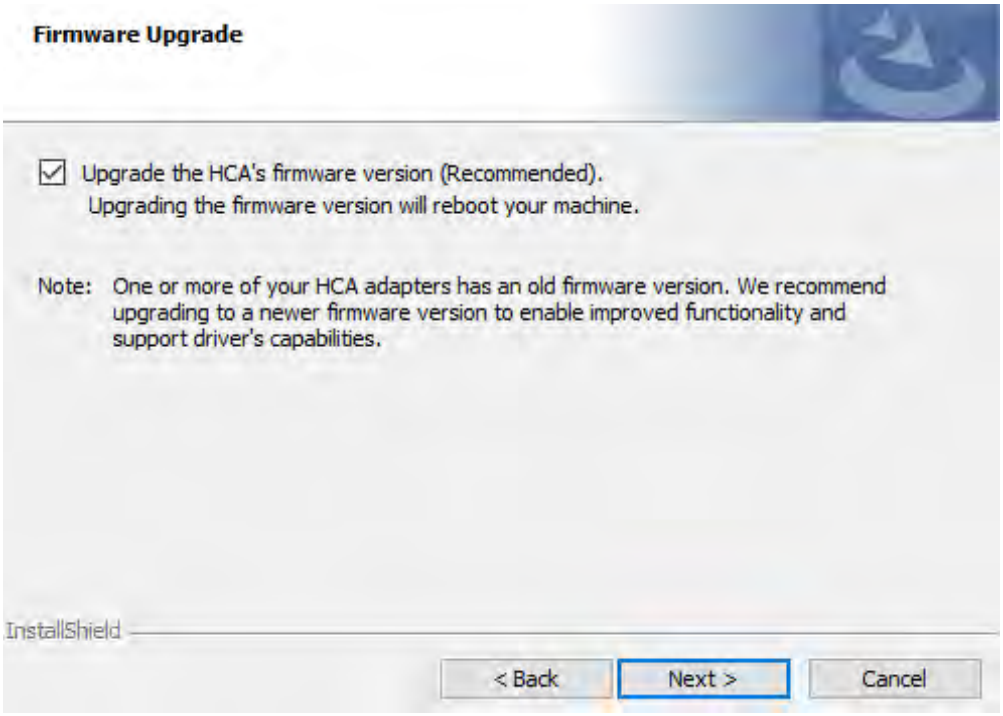
7. Read and accept the license agreement and click Next.



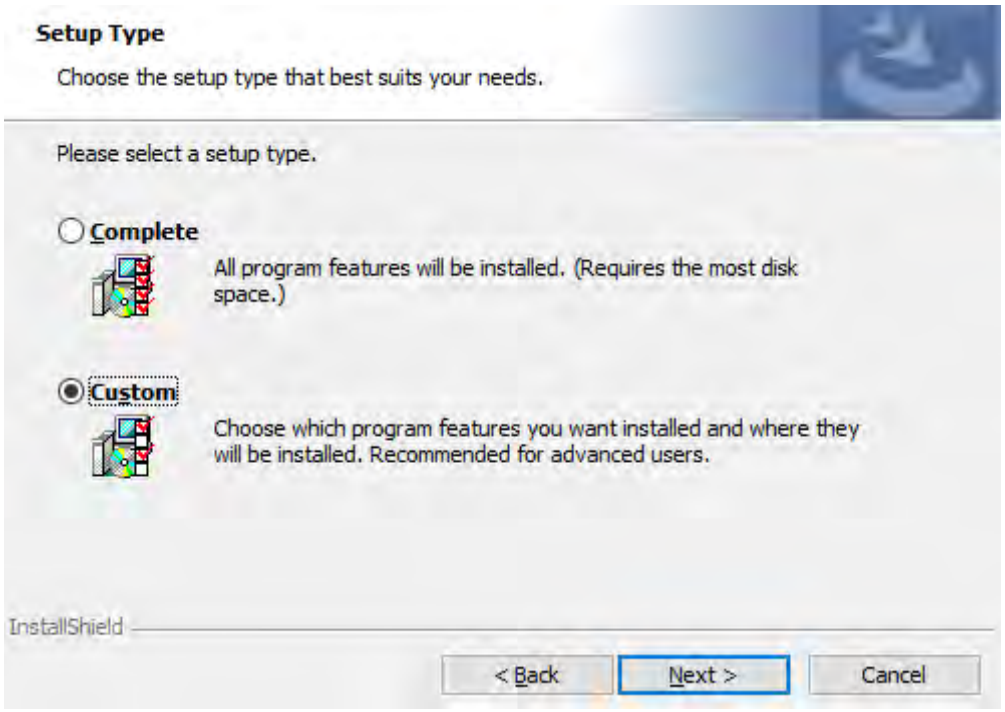
8. Select the target folder for the installation.



9. The firmware upgrade screen will be displayed in the following cases:
   - If the user has an OEM card. In this case, the firmware will not be displayed.
   - If the user has a standard NVIDIA® card with an older firmware version, the firmware will be updated accordingly. However, if the user has both an OEM card and a NVIDIA® card, only the NVIDIA® card will be updated.

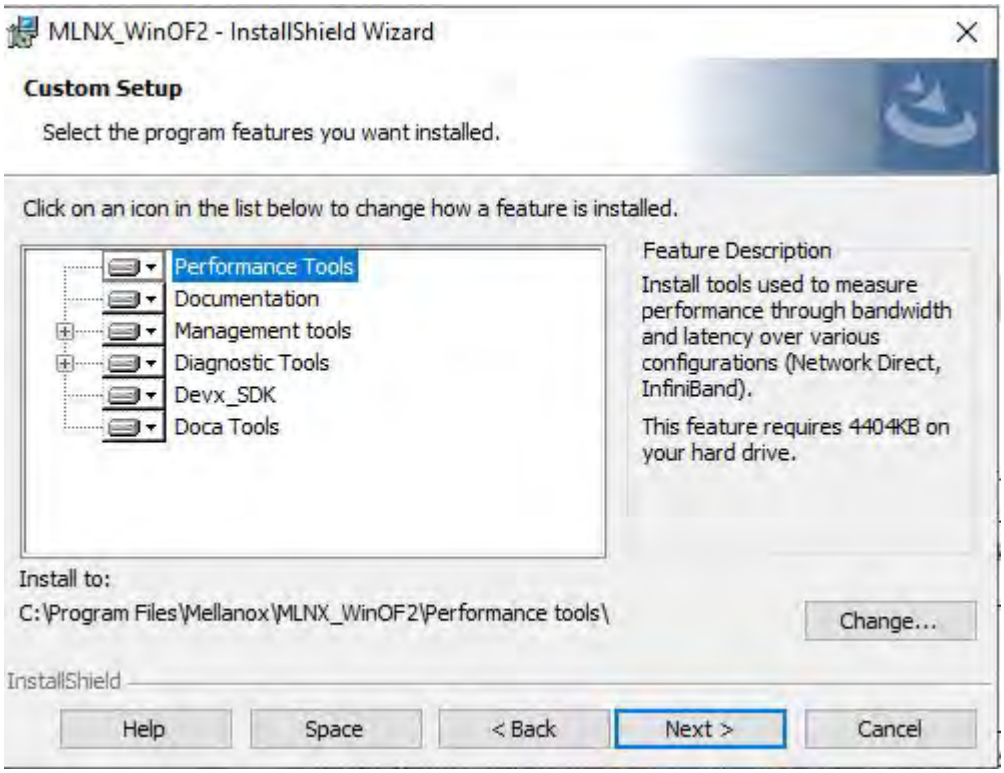


10. Select a Complete or Custom installation, follow Step a onward.

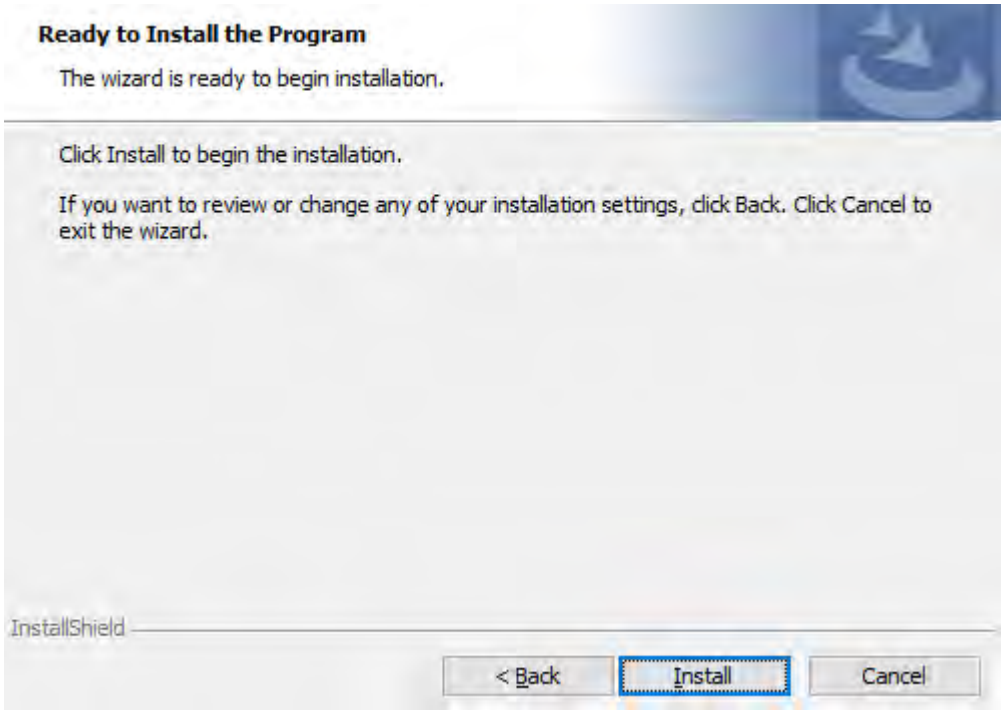


a. Select the desired feature to install:
   - Performances tools - install the performance tools that are used to measure performance in user environment
   - Documentation - contains the User Manual and Release Notes
   - Management tools - installation tools used for management, such as mlxstat
   - Diagnostic Tools - installation tools used for diagnostics, such as mlx5cmd

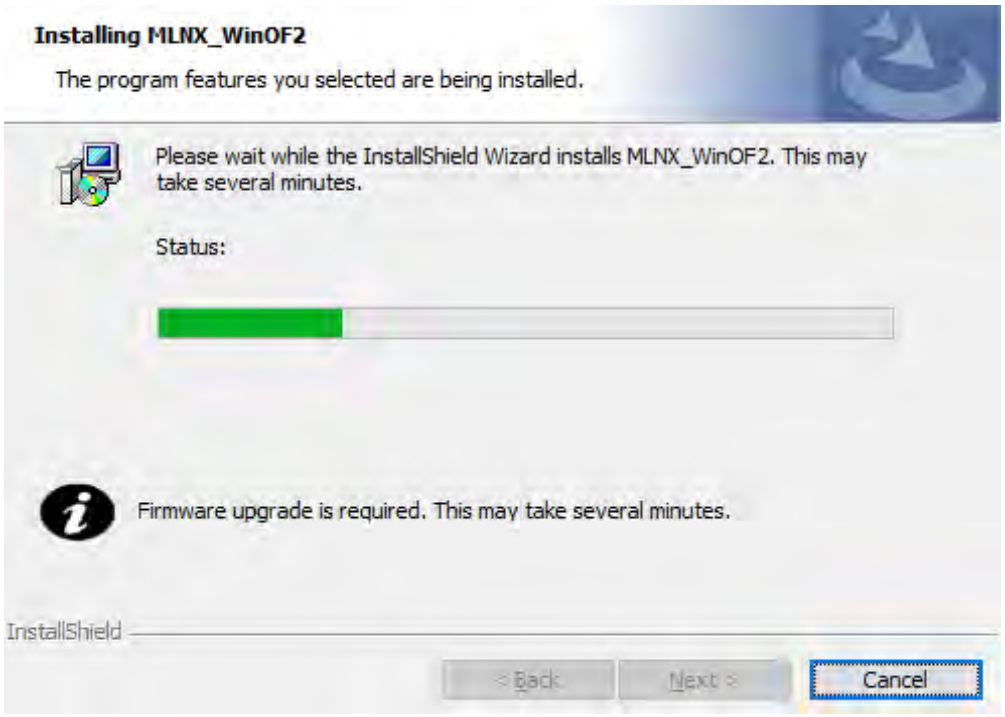
b. Click Next to install the desired tools.



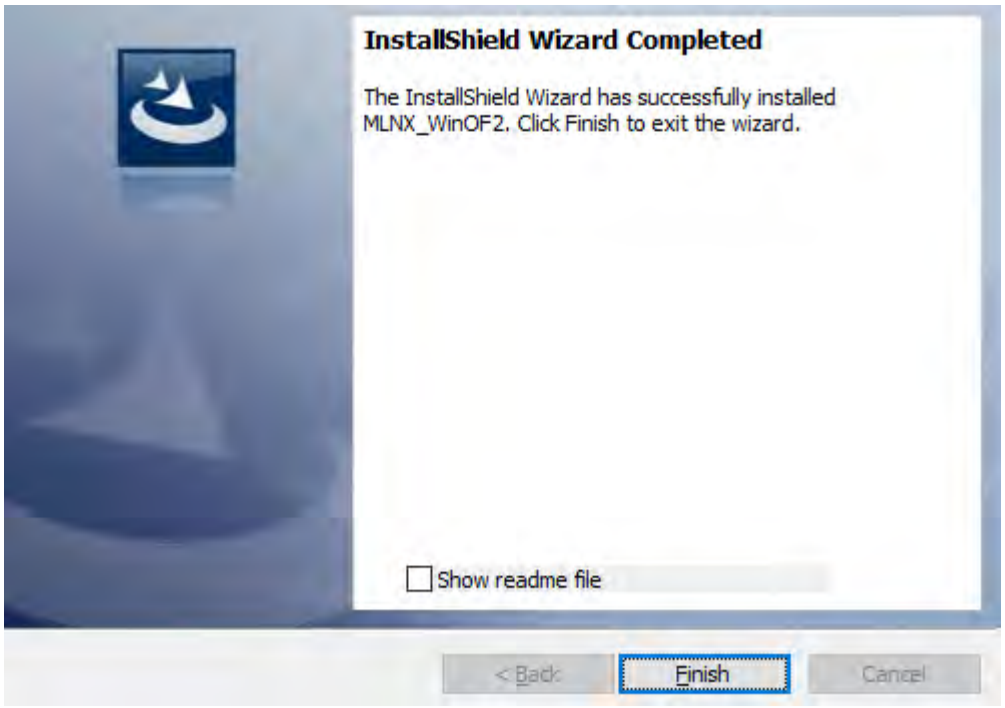
11. Click Install to start the installation.



12. In case firmware upgrade option was checked in Step 7, you will be notified if a firmware upgrade is required (see).



13. Click Finish to complete the installation.



Unattended Installation

> If no reboot options are specified, the installer restarts the computer whenever necessary without displaying any prompt or warning to the user.
> To control the reboots, use the */norestart* or */forcerestart* standard command-line options.

The following is an example of an unattended installation session.

1. Open a CMD console-> Click Start-> Task Manager File-> Run new task-> and enter CMD.
2. Install the driver. Run:

```
MLNX_WinOF2-[Driver/Version]_<revision_version>_All_-Arch.exe /S /v/qn
```

3. [Optional] Manually configure your setup to contain the logs option:

```
MLNX_WinOF2-[Driver/Version]_<revision_version>_All_-Arch.exe /S /v/qn /v"/l*vx [LogFile]"
```

4. [Optional] if you wish to control whether to install ND provider or not (i.e., *MT_NDPROPERTY default value is True*).

```
MLNX_WinOF2-[Driver/Version]_<revision_version>_All_Arch.exe /vMT_NDPROPERTY=1
```

5. [Optional] If you do not wish to upgrade your firmware version (i.e.,*MT_SKIPFWUPGRD default value is False*).

```
MLNX_WinOF2-[Driver/Version]_<revision_version>_All_Arch.exe /vMT_SKIPFWUPGRD=1
```

6. [Optional] If you do not want to install the Rshim driver, run.

```
MLNX_WinOF2_<revision_version>_All_Arch.exe /v" MT_DISABLE_RSHIM_INSTALL=1"
```

> The Rshim driver installanion will fail if a prior Rshim driver is already installed. The following fail message will be displayed in the log:
> "ERROR!!! Installation failed due to following errors: MlxRshim drivers installation disabled and MlxRshim drivers Installed, Please remove the following oem inf files from driver store: <oem inf list>"

7. [Optional] If you want to enable the default configuration for Rivermax, run.

```
MLNX_WinOF2_<revision_version>_All_Arch.exe /v"MT_RIVERMAX=1 /l*vx C:\Users\<user>\log.txt "
```

8. [Optional] If you want to skip the check for unsupported devices, run/

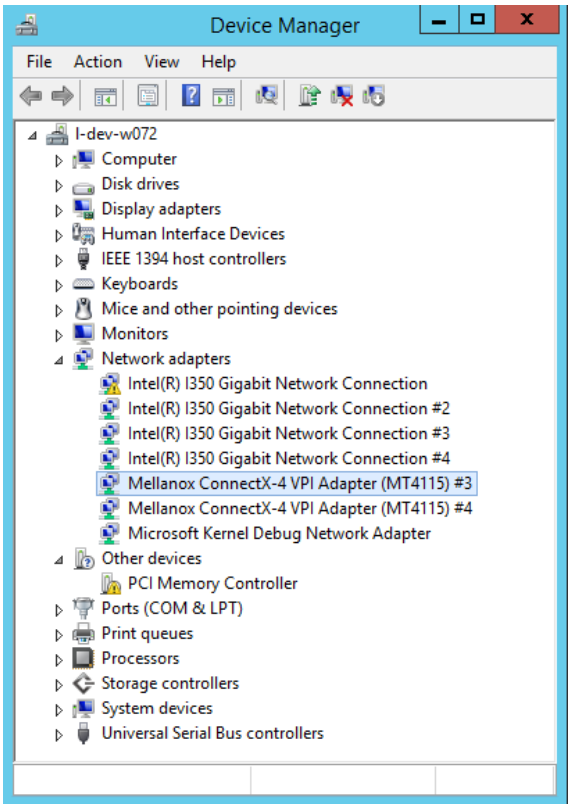
```
MLNX_WinOF2_<revision_version>_All_Arch.exe /v" SKIPUNSUPPORTEDDEVCHECK=1"
```

Installation Results

Upon installation completion, you can verify the successful addition of the network card(s) through the Device Manager. The inf files can be located at:

```
%ProgramFiles%\Mellanox\MLNX_WinOF2\Drivers\
```

To see the NVIDIA® network adapters, display the Device Manager and pull down the “Network adapters” menu.



Uninstalling WinOF-2 Driver

Attended Uninstallation

To uninstall MLNX_WinOF2 on a single node you need elevated administrator privileges.

Click Start-> Control Panel-> Programs and Features-> MLNX_WinOF2 -> Uninstall.

Unattended Uninstallation

To uninstall MLNX_WinOF2 using the unattended mode:

1. Open a CMD console-> Click Start-> Task Manager-> File-> Run new task-> and enter CMD.
2. Uninstall the driver. Run:

```
MLNX_WinOF2-<revision_version>_All_x64.exe /S /x /v"/qn"
```

Extracting Files Without Running Installation

To extract the files without running installation, perform the following steps:

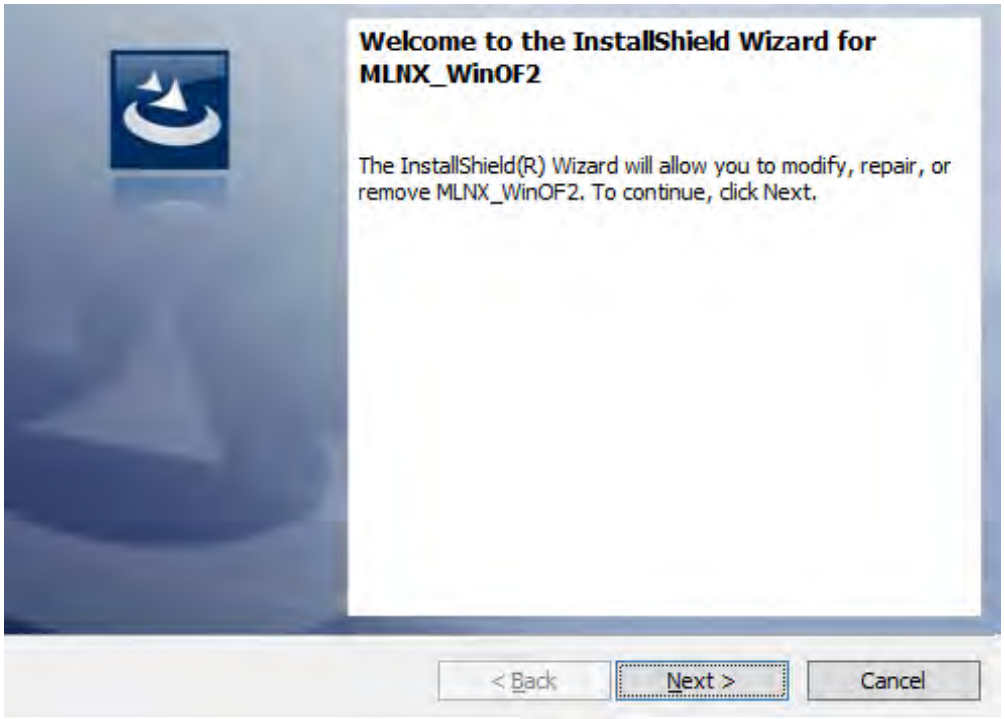
1. Open a CMD console-> Click Start-> Task Manager-> File-> Run new task-> and enter CMD.
2. Extract the driver and the tools:

```
MLNX_WinOF2-2_0_<revision_version>_All_x64.exe /a
```

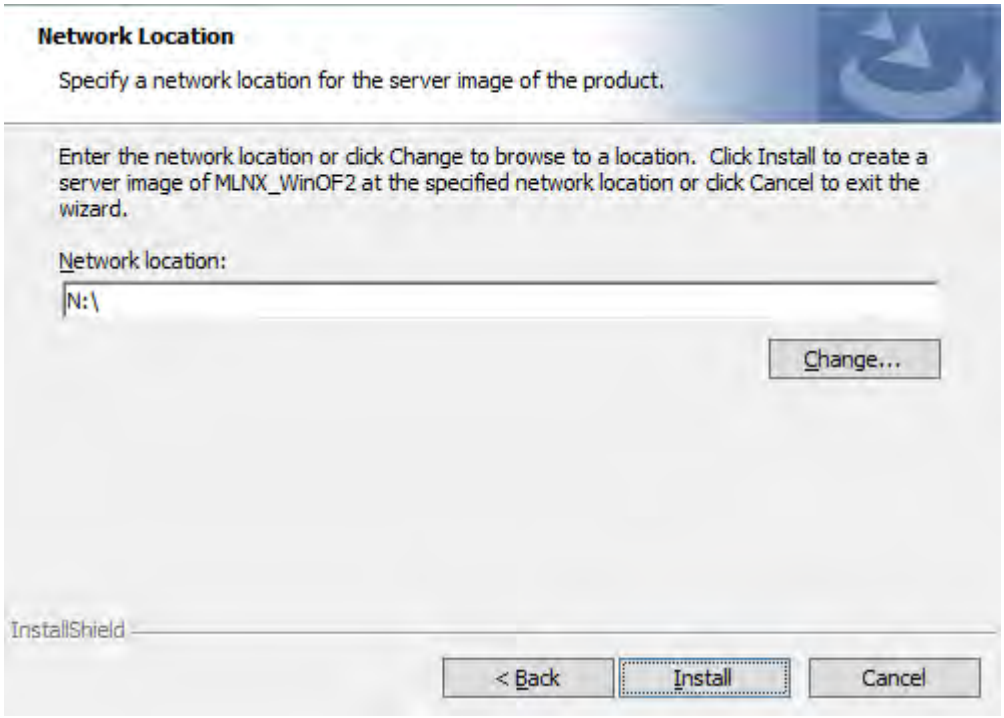
To extract only the driver file

```
MLNX_WinOF2-2_0_<revision_version>_All_x64.exe /a /vMT_DRIVERS_ONLY=1
```

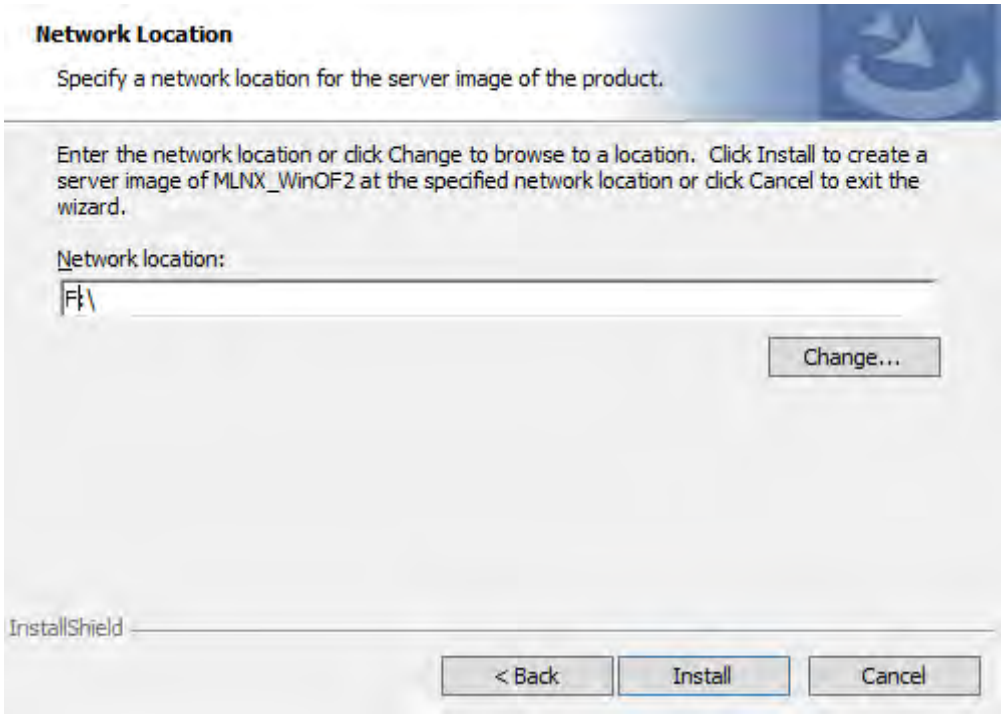
3. Click Next to create a server image.



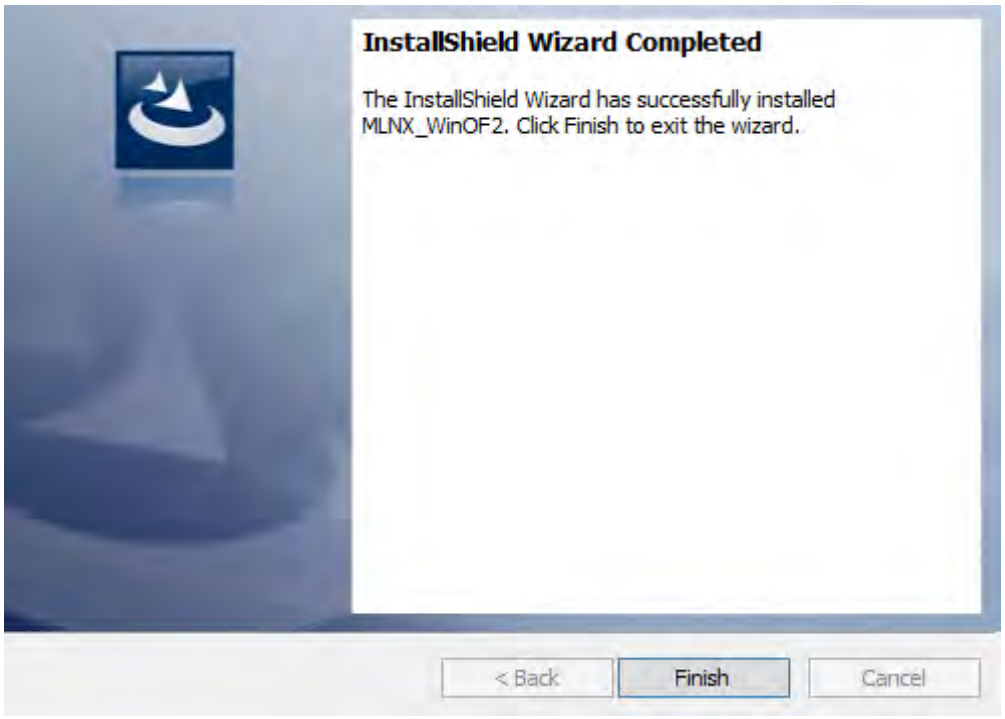
4. Click Change and specify the location in which the files are extracted to.



5. Click Install to extract this folder, or click Change to install to a different folder.



6. To complete the extraction, click Finish.



Firmware Upgrade

If the machine has a standard NVIDIA® card with an older firmware version, the firmware will be automatically updated as part of the NVIDIA® WinOF-2 package installation. For information on how to upgrade firmware manually, please refer to MFT User Manual.

If the machine has a DDA (pass through) facility, firmware update is supported only in the Host. Therefore, to update the firmware, the following must be performed:

1. Return the network adapters to the Host.
2. Update the firmware according to the steps in the MFT User Manual.
3. Attach the adapters back to VM with the DDA tools

Booting Windows from an iSCSI Target or PXE

⚠ SAN network boot is not supported.

Configuring the WDS, DHCP and iSCSI Servers

Configuring the WDS Server

1. Install the WDS server.

2. Extract the drivers to a local directory using the '-a' parameter.
   Example:

```
Mellanox.msi.exe -a
```

3. Add the driver to boot.wim (i.e., Use ‘index:2’ for Windows setup and ‘index:1’for WinPE).

```
dism /Mount-Wim /WimFile:boot.wim /index:2 /MountDir:mnt
dism /Image:mnt /Add-Driver /Driver:drivers /recurse
dism /Unmount-Wim /MountDir:mnt /commit
```

4. Add the NVIDIA® driver to install.wim (i.e., When adding the NVIDIA® driver to install.wim, verify you are using the appropriate index for your OS flavor. To check the OS run ‘imagex /info install.win’).

```
dism /Mount-Wim /WimFile:install.wim /index:4 /MountDir:mnt
dism /Image:mnt /Add-Driver /Driver:drivers /recurse
dism /Unmount-Wim /MountDir:mnt /commit
```

5. Add the new boot and install images to WDS.

For additional details on WDS, please refer to: http://technet.microsoft.com/en-us/library/jj648426.aspx

Configuring iSCSI Target

1. Install iSCSI Target (e.g StartWind).
2. Add to the iSCSI target initiators the IP addresses of the iSCSI clients.

Configuring the DHCP Server

1. Install a DHCP server.
2. Add to IPv4 a new scope.
3. Add boot client identifier (MAC/GUID) to the DHCP reservation.
4. Add to the reserved IP address the following options if DHCP and WDS are deployed on the same server:

Option	Name	Value
017	Root Path	iscsi:11.4.12.65::::iqn:2011-01:iscsiboot Assuming the iSCSI target IP is: 11.4.12.65 and the Target Name: iqn:2011-01:iscsiboot
060	PXEClient	PXEClient
066	Boot Server Host Name	WDS server IP address
067	Boot File Name	boot\x86\wdsnbp.com

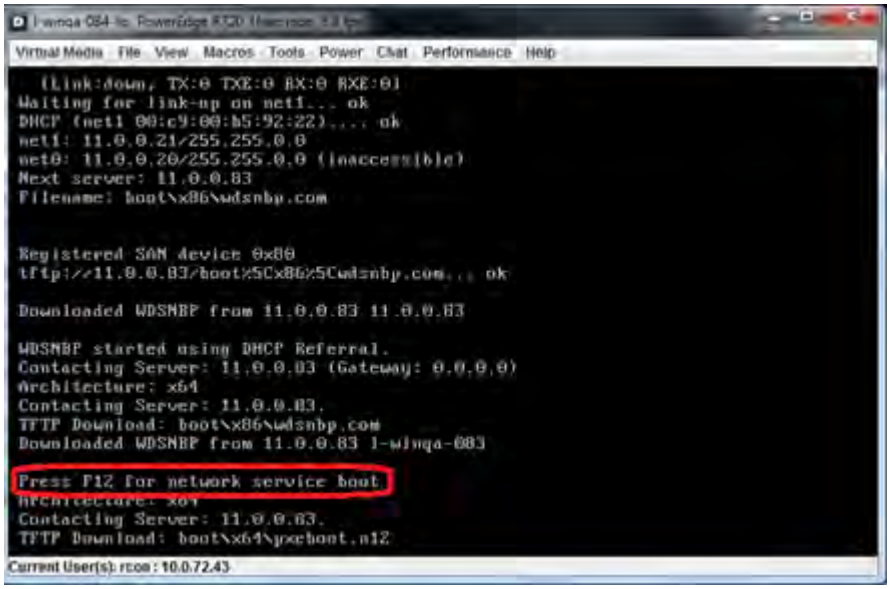
⚠ When DHCP and WDS are NOT deployed on the same server, DHCP options (60, 66, 67) should be empty, and the WDS option 60 must be configured.

Configuring the Client Machine

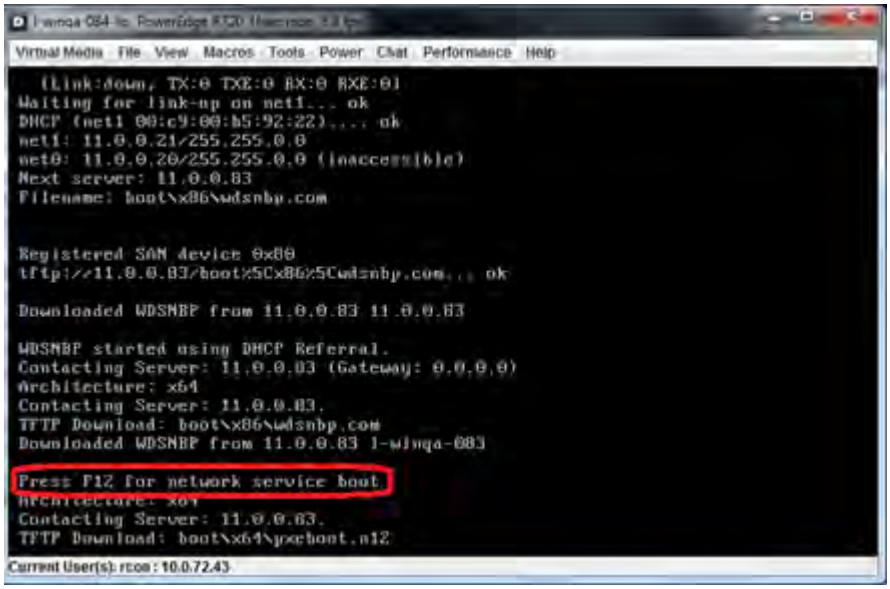
To configure your client, set the “Mellanox Adapter Card” as the first boot device in the BIOS settings boot order.

Installing the Operating System

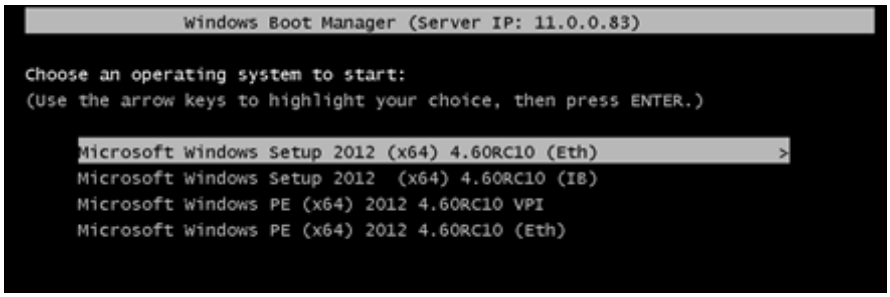
1. Reboot your client.
2. Press F12 when asked to proceed to network boot.
   Network Service Boot in ISCSi



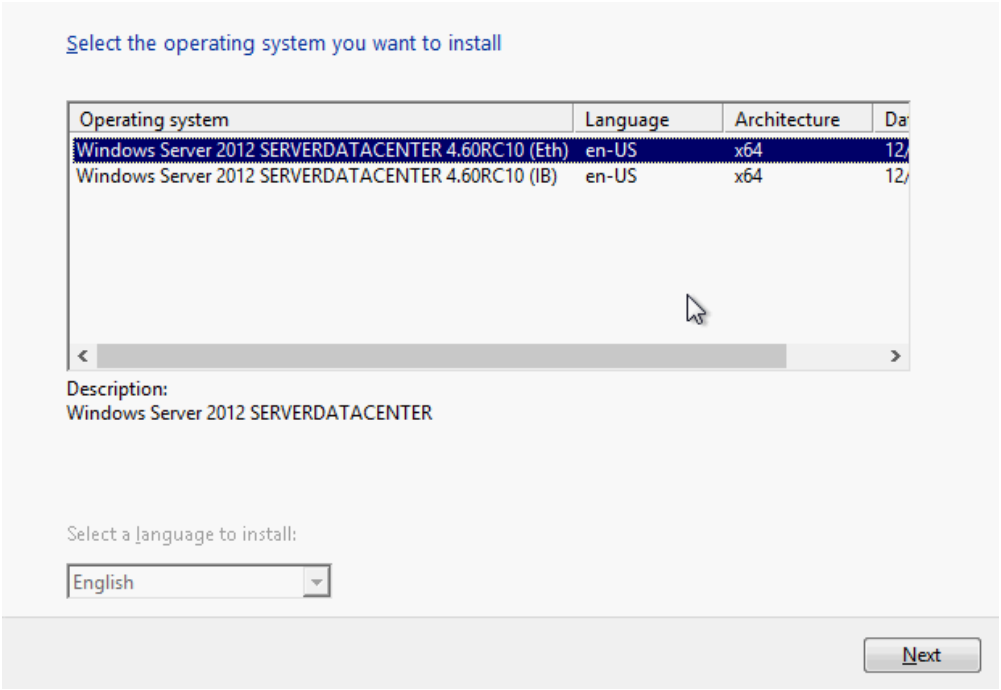
Network Service Boot in PXE



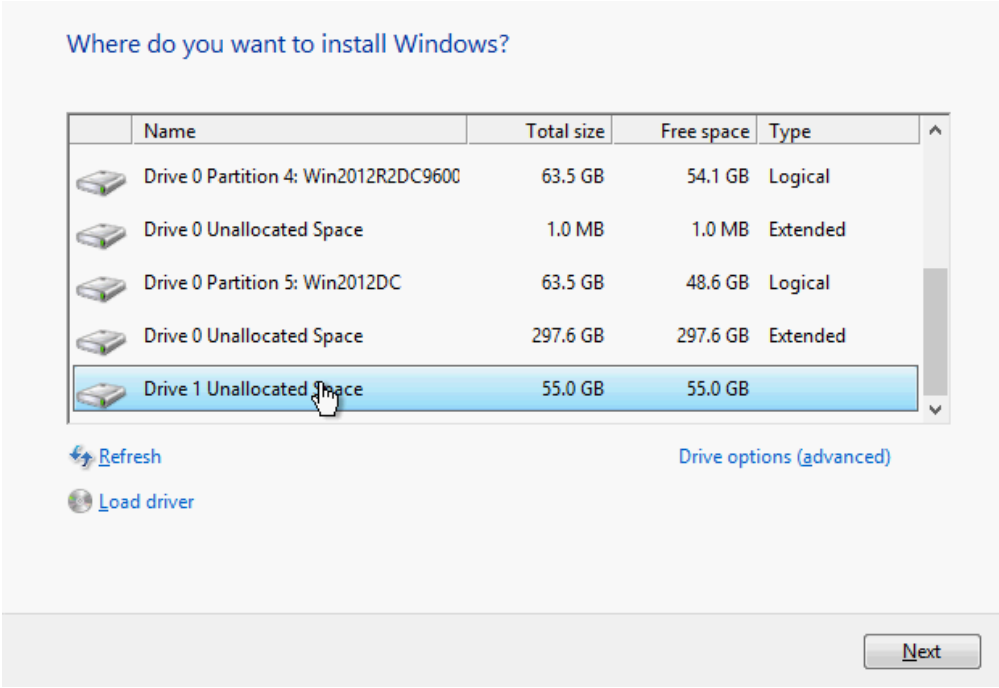
3. Choose the relevant boot image from the list of all available boot images presented.



4. Choose the Operating System you wish to install.



5. Run the Windows Setup Wizard.
6. Choose target drive to install Windows and follow the instructions presented by the installation Wizard.



Installation process will start once completing all the required steps in the Wizard, the Client will reboot and will boot from the iSCSI target.

Features Overview and Configuration

Once you have installed NVIDIA WinOF-2 package, you can perform various modifications to your driver to make it suitable for your system's requirements.

Changes made to the Windows registry take effect immediately, and no backup is automatically made.
Do *not* edit the Windows registry unless you are confident regarding the changes.

The chapter contains the following sections:

- BIOS Settings Configuration
- General Capabilities
- Ethernet Network
- InfiniBand Network
- Storage Protocols
- Virtualization
- Configuring the Driver Registry Keys
- Network Direct Interface
- Performance Tuning
- Adapter Cards Counters

- Resiliency
- RDMA Capabilities
- NVIDIA BlueField SmartNIC Mode
- RShim Drivers and Usage

BIOS Settings Configuration

> ⚠ It is recommended to enable the “above 4G decoding” BIOS setting for features that require large amount of PCIe resources.
>
> Such features are: SR-IOV with numerous VFs, PCIe Emulated Switch, and Large BAR Requests.

General Capabilities

General supported capabilities:

- Port Management
- Assigning Port IP After Installation
- Modifying Driver’s Configuration
- Receive Side Scaling (RSS)
- Displaying Adapter Related Information
    - DSCP Sanity Testing
- Live Firmware Patch Update

> ⚠ The capabilities described below are applicable to both Ethernet and InfiniBand networks.

Port Management

For retrieving the port types, perform one of the following:

- Run mlx5cmd -stat from the "Command Prompt", and check the link_layer from the output.
- See the Port Type in the Information tab in the Device Manager window (see Displaying Adapter Related Information)

To configure the port types to Ethernet/InfiniBand mode on a device, use the mlxconfig.exe utility, which is a part of the MFT package for Windows, and is available at https://network.nvidia.com/products/adapter-software/firmware-tools/.

1. Install the WinMFT package.
2. Retrieve the device name:
    a. In command prompt, run "mst status -v":

```
mst status -v
MST devices:
------------
```

```
mt4099_pci_cr0 bus:dev.fn=04:00.0
mt4099_pciconf0 bus:dev.fn=04:00.0
mt4103_pci_cr0 bus:dev.fn=21:00.0
mt4103_pciconf0 bus:dev.fn=21:00.0

mt4115_pciconf0 bus:dev.fn=24:00.0
```

b. Identify the desired device by its "bus:dev.fn" PCIe address.

3. Configure the port type to either InfiniBand or Ethernet:

   a. Ethernet, execute the following command with the appropriate device name:

   ```
   mlxconfig -d mt4115_pciconf0 set LINK_TYPE_P1=2
   ```

   b. InfiniBand, execute the following command with the appropriate device name:

   ```
   mlxconfig -d mt4115_pciconf0 set LINK_TYPE_P1=1
   ```

   ⚠ To set the type of the second port, set the parameter LINK_TYPE_P2.

4. Reboot the system.

⚠ Changing the port type will change some of the registry keys to the default values of the new port type.

For further information, please refer to the MFT User Manual.

Assigning Port IP After Installation

By default, your machine is configured to obtain an automatic IP address via a DHCP server. In some cases, the DHCP server may require the MAC address of the network adapter installed in your machine.

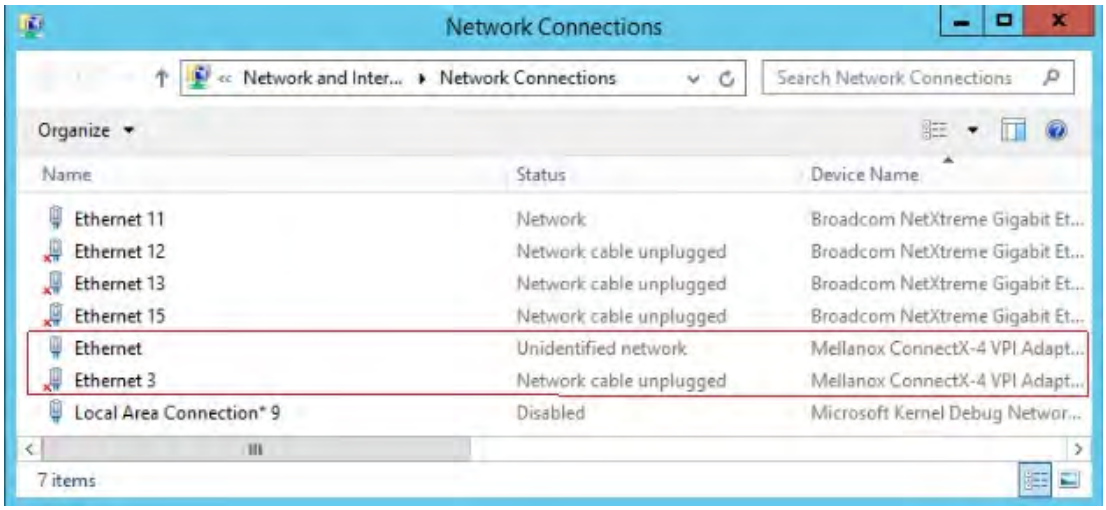
➢ *To obtain the MAC address:*

1. Open a CMD console-> Click Start-> Task Manager-> File-> Run new task-> and enter CMD.
2. Display the MAC address as “Physical Address”.

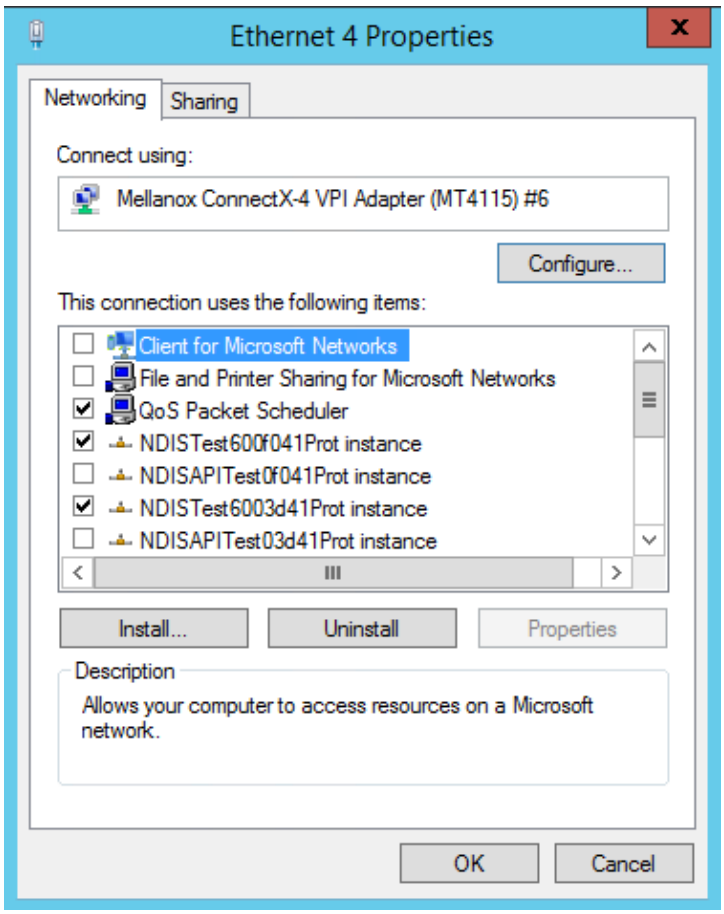
```
ipconfig /all
```

➢ *To assign a static IP address to a network port after installation:*

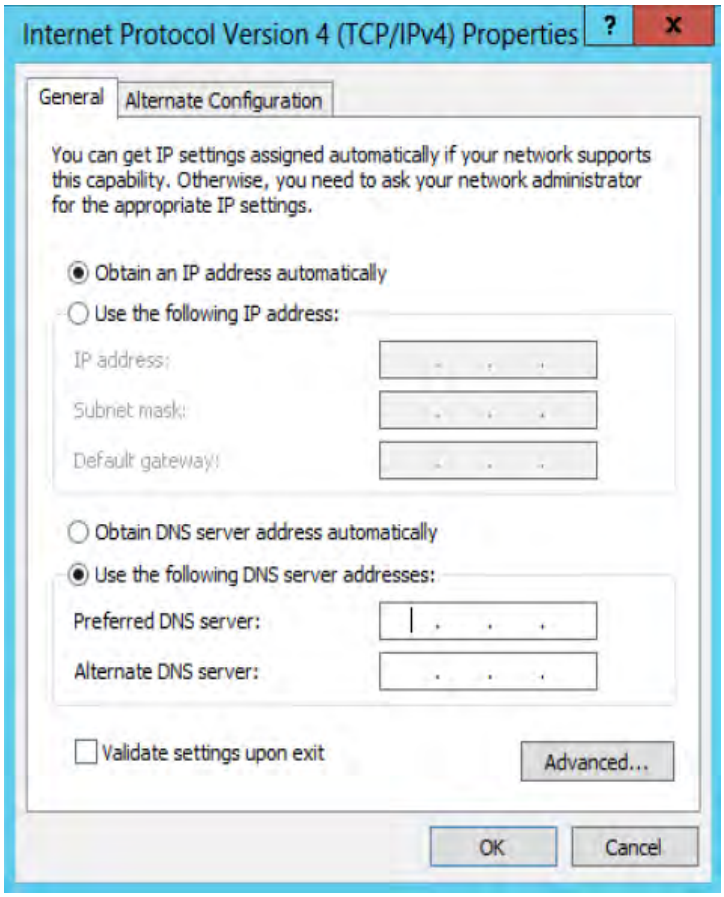
1. Open the Network Connections window. Locate Local Area Connections with NVIDIA® devices.



2. Right-click a NVIDIA® Local Area Connection and left-click Properties.



3. Select Internet Protocol Version 4 (TCP/IPv4) from the scroll list and click Properties.
4. Select the “Use the following IP address:” radio button and enter the desired IP information.



5. Click OK.
6. Close the Local Area Connection dialog.
7. Verify the IP configuration by running ‘ipconfig’ from a CMD console.

```
ipconfig
...
Ethernet adapter Local Area Connection 4:

Connection-specific DNS Suffix . :
IP Address. . . . . . . . . . . . : 11.4.12.63
Subnet Mask . . . . . . . . . . . : 255.255.0.0
Default Gateway . . . . . . . . . :
...
```

Modifying Driver’s Configuration

➢ *To modify the driver’s configuration after installation, perform the following steps:*

1. Open Device Manager and expand Network Adapters in the device display pane.
2. Right-click the NVIDIA® ConnectX adapter entry and left-click Properties.
3. Click the Advanced tab and modify the desired properties.

> ⚠ The IPoIB network interface is automatically restarted once you finish modifying IPoIB parameters. Consequently, it might affect any running traffic.

Important Notes:
- For help on a specific parameter/option, check the help button at the bottom of the dialog.
- If you select one of the entries Offload Options, Performance Options, or Flow Control Options, you'll need to click the Properties button to modify parameters via a pop-up dialog.

Receive Side Scaling (RSS)

RSS settings can be set per individual adapters as well as globally using the Registry Keys below.

> ⚠ It is recommended that the RSS base processor is core #1 and above as usually processor 0 is very utilized.

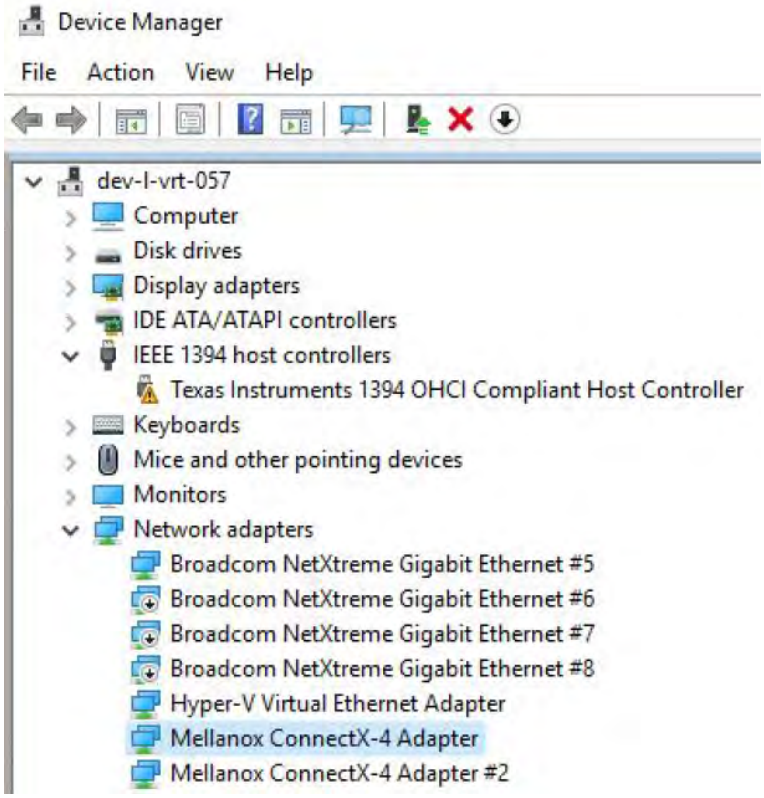
For instructions on how to find interface index in registry <nn>, please refer to section Finding the Index Value of the Network Interface.

Sub-key	Description
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>*MaxRSSProcessors	Maximum number of CPUs allotted. Sets the desired maximum number of processors for each interface. The number can be different for each interface. **Note:** Restart the network adapter after you change this registry key.
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>*RssBaseProcNumber	Base CPU number. Sets the desired base CPU number for each interface. The number can be different for each interface. This allows partitioning of CPUs across network adapters. **Note:** Restart the network adapter when you change this registry key.
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>*NumaNodeID	NUMA node affinitization
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>*RssBaseProcGroup	Sets the RSS base processor group for systems with more than 64 processors.
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>\RssV2	Enables the RSS V2 feature.
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>\ValidateRssV2	Enable strict argument validation for upper layer testing. Set along with RssV2 key to enable the RSSv2 feature.

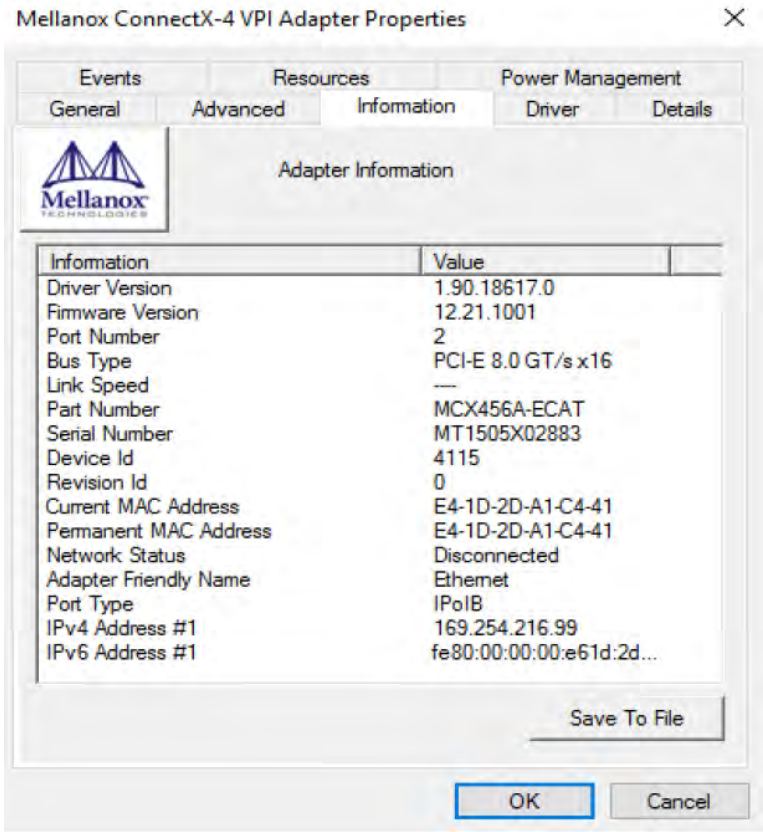
Displaying Adapter Related Information

➢ *To display a summary of network adapter software, firmware and hardware related information, perform the following steps:*

1. Display the Device Manager.



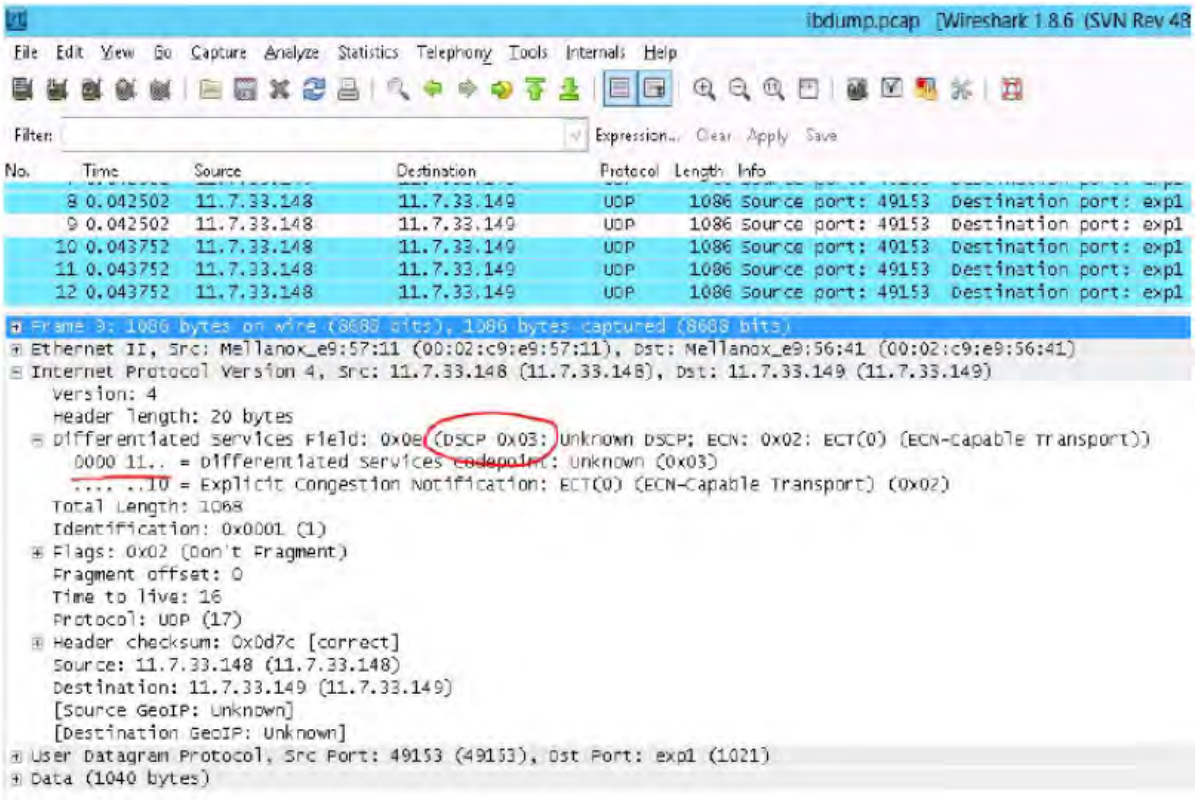
2. Select the Information tab from the Properties sheet.



> Click Save to File and provide the output file name to save this information for debug purposes,

DSCP Sanity Testing

To verify that all QoS and DSCP settings are correct, you can capture the incoming and outgoing traffic by using the mlx5cmd sniffer tool. The tool allows you to see the DSCP value in the captured packets, as displayed in the figure below.



Live Firmware Patch Update

Live Firmware Patch allows ConnectX adapter cards family firmware update (upgrade or downgrade) while the driver, the network ports, and the PCI link remain functional. It is supported only between two firmware versions that support Live Firmware Patch.

To check this capability is available in both firmware versions, burn a new firmware (e.g. using mlxburn) and use “mlxfwreset -d <device> q” command to check if Live Firmware Patch is supported. If Live Firmware Patch feature is supported between two firmware versions, it will be the default reset option.

For example:

```
mlxfwreset -d <device> r
```

This command will perform Live Firmware Patch if it is possible.

Upon a successful Live Firmware Patch update, the following Event Log message will be generated:

```
<Adapter name>: Firmware version was updated from version %3 to version %4 as a result of the firmware live patch
update.
Log Name: System
Source: mlx5
Event ID: 400
Level: Warning
```

Ethernet Network

Ethernet supported capabilities:

- Packet Burst Handling
- Dropless Mode
- RDMA over Converged Ethernet (RoCE)
- RoCEv2 Congestion Management (RCM)
- Enhanced Connection Establishment
- Zero Touch RoCE
- Teaming and VLAN
- Command Line Based Teaming Configuration
- Configuring Quality of Service (QoS)
- Differentiated Services Code Point (DSCP)
- Receive Segment Coalescing (RSC)
- Wake-on-LAN (WoL)
- Data Center Bridging Exchange (DCBX)
- Receive Path Activity Monitoring
- Head of Queue Lifetime Limit
- VXLAN
- Threaded DPC
- UDP Segmentation Offload (USO)
- Hardware Timestamping
- Striding RQ
- Additional MAC Addresses for the Network Adapter
- Explicit Congestion Notification (ECN) Hint in CQE
- NDIS Poll Mode
- GPUDirect
- Hardware QoS Offload
- Multi Prio Send Queue

Packet Burst Handling

This feature allows packet burst handling, while avoiding packet drops that may occur when a large amount of packets is sent in a short period of time. For the feature's registry keys, see section Performance Registry Keys.

By default, the feature is disabled, and the AsyncReceiveIndicate registry key is set to 0. To enable the feature, choose one of the following options:

- To enable packet burst buffering using threaded DPC (recommended), set the AsyncReceiveIndicate registry key to 1.
- To enable packet burst buffering using polling, set the AsyncReceiveIndicate to 2.

To control the number of reserved receive packets, set the RfdReservationFactor registry key:

Default	150
Recommended	10,000
Maximum	5,000,000

The memory consumption will increase in accordance with the "RfdReservationFactor" registry key value.

Dropless Mode

This feature helps avoid dropping packets when the driver is not posting receive descriptors fast enough to the device (e.g. in cases of high CPU utilization).

Enabling/Disabling the Feature

There are two ways to enable/disable this feature:

- Send down an OID to the driver. The following is the information buffer format:

```
Typedef struct _DROPLESS_MODE
{
UINT32 signature; UINT8 dropless_mode;
} DROPLESS_MODE, *PDROPLESS_MODE;
```

OID code	0xFFA0C932
Signature value	(ULONG) 0x0C1EA2
Dropless_mode value	1- Enables the feature 2- Disables the feature

  The driver sets a default timeout value of 5 milliseconds.
- Add the "DelayDropTimeout" registry key, set the value to one of the following options, and reload the adapter:

DelayDropTimeout	“50" (recommended value to set the timeout to is 5 milliseconds) "0" to disable **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

The registry key should be added to
`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<IndexValue>`
To find the IndexValue, refer to section Finding the Index Value of the Network Interface.

Status Query

The status of the feature can be queried by sending down the same OID code (0xFFA0C932). If enabled, the driver will fill up the information buffer in the following format

```
DROPLESS_MODE *answer = (DROPLESS_MODE *)InformationBuffer;
answer->signature = MLX_OID_BUFFER_SIGNATURE;
answer->dropless_mode = 1;
```

The Dropless_mode value will be set to 0 if disabled.

Timeout Values and Timeout Notification

The feature's timeout values are defined as follows:

Registry value units	100usec
Default driver value	50 (5 milliseconds)
Accepted values	0 (disabled) to100 (10 milliseconds)

When the feature is enabled and a packet is received for an RQ with no receive WQEs, the packet processing is delayed, waiting for receive WQEs to be posted. The feature allows the flow control mechanism to take over, thus avoiding packet loss. During this period, the timer starts ticking, and if receive WQEs are not posted before the timer expires, the packet is dropped, and the feature is disabled.

The driver notifies of the timer's expiration by generating an event log with event ID 75 and the following message:

"Delay drop timer timed out for RQ Index [RqId]. Dropless mode feature is now disabled".

The feature can be re-enabled by sending down an OID call again with a non-zero timeout value. Every time the feature is enabled by the user, the driver logs an event with event ID 77 and the following message:
"Dropless mode entered. For more details, please refer to the user manual document"

Similarly, every time the feature is disabled by the user, the driver logs an event with event ID 78 and the following message:
"Dropless mode exited. For more details, please refer to the user manual document."

RDMA over Converged Ethernet (RoCE)

Remote Direct Memory Access (RDMA) is the remote memory management capability that allows server to server data movement directly between application memory without any CPU involvement. RDMA over Converged Ethernet (RoCE) is a mechanism to provide this efficient data transfer with very low latencies on lossless Ethernet networks. With advances in data center convergence over reliable Ethernet, ConnectX® EN with RoCE uses the proven and efficient RDMA

transport to provide the platform for deploying RDMA technology in mainstream data center application at 10GigE and 40GigE link-speed. ConnectX® EN with its hardware offload support takes advantage of this efficient RDMA transport (InfiniBand) services over Ethernet to deliver ultra-low latency for performance-critical and transaction intensive applications such as financial, database, storage, and content delivery networks. RoCE encapsulates IB transport and GRH headers in Ethernet packets bearing a dedicated ether type. While the use of GRH is optional within InfiniBand subnets, it is mandatory when using RoCE. Applications written over IB verbs should work seamlessly, but they require provisioning of GRH information when creating address vectors. The library and driver are modified to provide mapping from GID to MAC addresses required by the hardware.

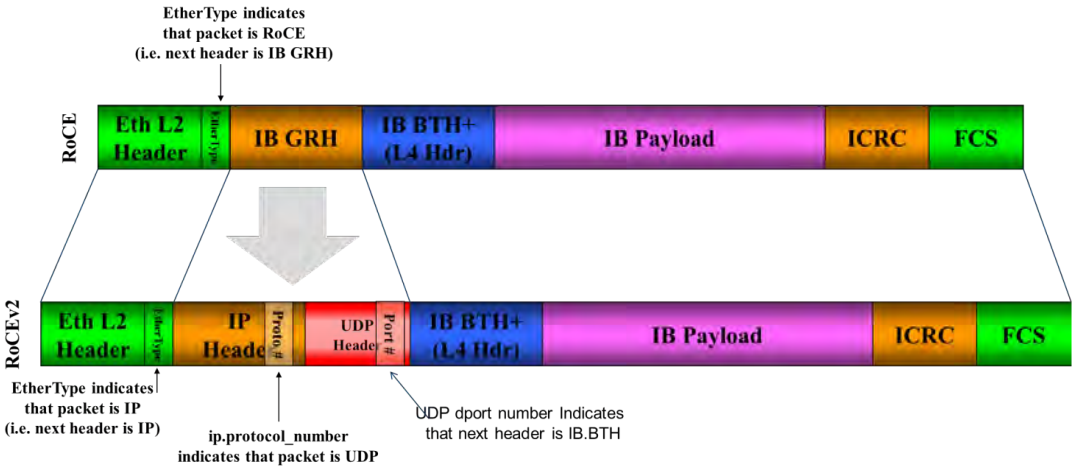
IP Routable (RoCEv2)

RoCE has two addressing modes: MAC based GIDs, and IP address based GIDs. In RoCE IP based, if the IP address changes while the system is running, the GID for the port will automatically be updated with the new IP address, using either IPv4 or IPv6.

RoCE IP based allows RoCE traffic between Windows and Linux systems, which use IP based GIDs by default.

A straightforward extension of the RoCE protocol enables traffic to operate in layer 3 environments. This capability is obtained via a simple modification of the RoCE packet format. Instead of the GRH used in RoCE, routable RoCE packets carry an IP header which allows traversal of IP L3 Routers and a UDP header that serves as a stateless encapsulation layer for the RDMA Transport Protocol Packets over IP.

Figure 1: RoCE & RoCE v2 Differences

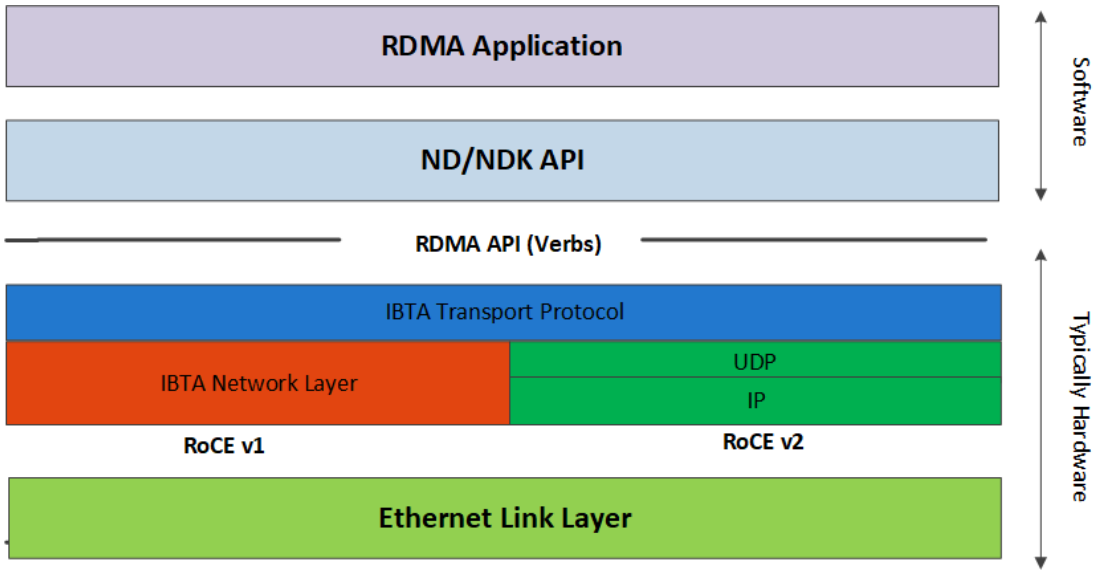


The proposed RoCEv2 packets use a well-known UDP destination port value that unequivocally distinguishes the datagram. Similar to other protocols that use UDP encapsulation, the UDP source port field is used to carry an opaque flow-identifier that allows network devices to implement packet forwarding optimizations (e.g. ECMP) while staying agnostic to the specifics of the protocol header format.

The UDP source port is calculated as follows: *UDP.SrcPort = (SrcPort XOR DstPort) OR 0xC000* , where SrcPort and DstPort are the ports used to establish the connection.
For example, in a Network Direct application, when connecting to a remote peer, the destination IP

address and the destination port must be provided as they are used in the calculation above. The source port provision is optional.

Furthermore, since this change exclusively affects the packet format on the wire, and due to the fact that with RDMA semantics packets are generated and consumed below the AP applications can seamlessly operate over any form of RDMA service (including the routable version of RoCE as shown in the RoCE and RoCE v2 Frame Format Differences diagram), in a completely transparent way (Standard RDMA APIs are IP based already for all existing RDMA technologies).



The fabric must use the same protocol stack in order for nodes to communicate.

In earlier versions, the default value of RoCE mode was RoCE v1. As of WinOF-2 v1.30, the default value of RoCE mode will be RoCEv2.

Upgrading from earlier versions to version 1.30 or above will save the old default value (RoCEv1).

RoCE Configuration

In order to function reliably, RoCE requires a form of flow control. While it is possible to use global flow control, this is normally undesirable, for performance reasons.

The normal and optimal way to use RoCE is to use Priority Flow Control (PFC). To use PFC, it must be enabled on all endpoints and switches in the flow path.

In the following section we present instructions to configure PFC on NVIDIA® ConnectX® family cards. There are multiple configuration steps required, all of which may be performed via PowerShell. Therefore, although we present each step individually, you may ultimately choose to write a PowerShell script to do them all in one step. Note that administrator privileges are required for these steps.

The NIC is configured by default to enable RoCE. If the switch is not configured to enable ECN and/or PFC, this will cause performance degradation. Thus, it is recommended to enable ECN on the switch or disable the *NetworkDirect registry key.

For more information on how to enable ECN and PFC on the switch, refer to the https://enterprise-support.nvidia.com/docs/DOC-2855 community page.

Configuring Windows Host

Since PFC is responsible for flow controlling at the granularity of traffic priority, it is necessary to assign different priorities to different types of network traffic.

As per RoCE configuration, all ND/NDK traffic is assigned to one or more chosen priorities, where PFC is enabled on those priorities.

Configuring Windows host requires configuring QoS. To configure QoS, please follow the procedure described in Configuring Quality of Service (QoS)

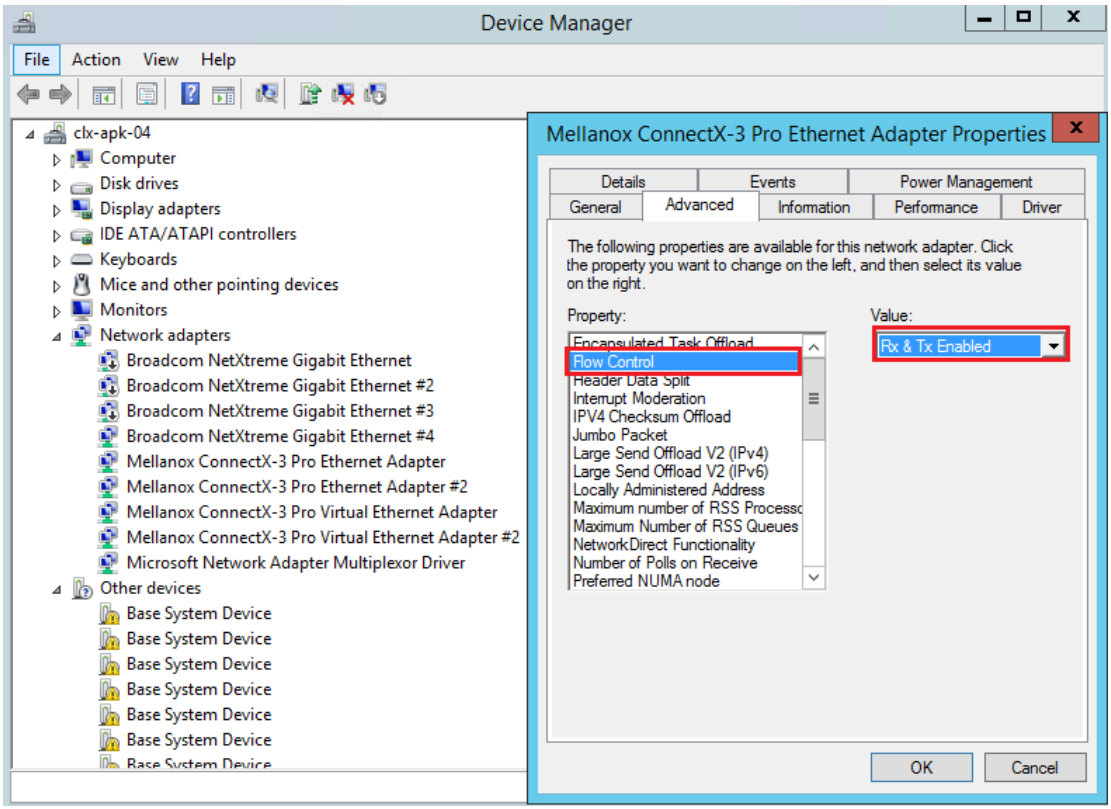
Global Pause (Flow Control)

To use Global Pause (Flow Control) mode, disable QoS and Priority:

```
PS $ Disable-NetQosFlowControl
PS $ Disable-NetAdapterQos <interface name>
```

To confirm flow control is enabled in adapter parameters:

Go to: Device manager --> Network adapters --> Mellanox ConnectX-4/ConnectX-5 Ethernet Adapter --> Properties -->Advanced tab



Configuring Arista Switch

1. Set the ports that face the hosts as trunk.

```
(config)# interface et10
(config-if-Et10)# switchport mode trunk
```

2. Set VID allowed on trunk port to match the host VID.

```
(config-if-Et10)# switchport trunk allowed vlan 100
```

3. Set the ports that face the network as trunk.

```
(config)# interface et20
(config-if-Et20)# switchport mode trunk
```

4. Assign the relevant ports to LAG.

```
(config)# interface et10
(config-if-Et10)# dcbx mode ieee
(config-if-Et10)# speed forced 40gfull
(config-if-Et10)# channel-group 11 mode active
```

5. Enable PFC on ports that face the network.

```
(config)# interface et20
(config-if-Et20)# load-interval 5
(config-if-Et20)# speed forced 40gfull
(config-if-Et20)# switchport trunk native vlan tag
(config-if-Et20)# switchport trunk allowed vlan 11
```

```
(config-if-Et20)# switchport mode trunk
(config-if-Et20)# dcbx mode ieee
(config-if-Et20)# priority-flow-control mode on
(config-if-Et20)# priority-flow-control priority 3 no-drop
```

Using Global Pause (Flow Control)

To enable Global Pause on ports that face the hosts, perform the following:

```
(config)# interface et10
(config-if-Et10)# flowcontrol receive on
(config-if-Et10)# flowcontrol send on
```

Using Priority Flow Control (PFC)

To enable PFC on ports that face the hosts, perform the following:

```
(config)# interface et10
(config-if-Et10)# dcbx mode ieee
(config-if-Et10)# priority-flow-control mode on
(config-if-Et10)# priority-flow-control priority 3 no-drop
```

Configuring Router (PFC only)

The router uses L3's DSCP value to mark the egress traffic of L2 PCP. The required mapping, maps the three most significant bits of the DSCP into the PCP. This is the default behavior, and no additional configuration is required.

Copying Port Control Protocol (PCP) between Subnets

The captured PCP option from the Ethernet header of the incoming packet can be used to set the PCP bits on the outgoing Ethernet header.

Configuring the RoCE Mode

RoCE mode is configured per adapter or per driver. If RoCE mode key is set for the adapter, then it will be used. Otherwise, it will be configured by the per-driver key. The per-driver key is shared between all devices in the system.

The supported RoCE modes depend on the firmware installed. If the firmware does not support the needed mode, the fallback mode would be the maximum supported RoCE mode of the installed NIC.

RoCE is enabled by default. Configuring or disabling the RoCE mode can be done via the registry key.

To update it for a specific adapter using the registry key, set the roce_mode as follows:

1. Find the registry key index value of the adapter according to section Finding the Index Value of the Network Interface.

2. Set the roce_mode in the following path:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-
bfc1-08002be10318}\<IndexValue>
```

To update it for all the devices using the registry key, set the roce_mode as follows:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\mlx5\Parameters\Roce
```

For changes to take effect, please restart the network adapter after changing this registry key.

Registry Key Parameters

The following are per-driver and will apply to all available adapters.

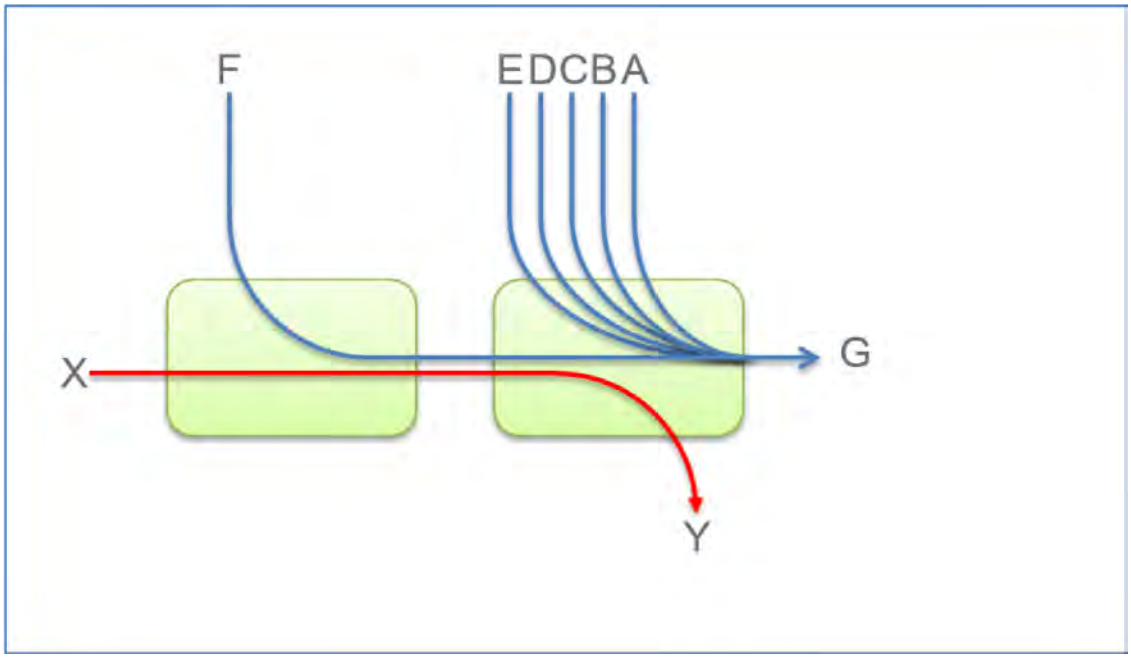
Parameters Name	Parameter type	Description	Allowed and Default Values
roce_mode	DWORD	Sets the RoCE mode. The following are the possible RoCE modes: • RoCE MAC Based • RoCE v2 • No RoCE	• RoCE MAC Based = 0 • **[Default]** RoCE v2 = 2 • No RoCE = 4

RoCEv2 Congestion Management (RCM)

Network Congestion occurs when the number of packets being transmitted through the network approaches the packet handling the capacity of the network. A congested network will suffer from throughput deterioration manifested by increasing time delays and high latency.

In lossy environments, this leads to a packet loss. In lossless environments, it leads to “victim flows” (streams of data which are affected by the congestion, caused by other data flows that pass through the same network).

The figure below demonstrates a victim flow scenario. In the absence of congestion control, flow X'Y suffers from reduced bandwidth due to flow F'G, which experiences congestion. To address this, Congestion Control methods and protocols were defined.



This chapter describes (in High-Level), RoCEv2 Congestion Management (RCM), and provides a guide on how to configure it in Windows environment.

RoCEv2 Congestion Management (RCM) provides the capability to avoid congestion hot spots and optimize the throughput of the fabric.

With RCM, congestion in the fabric is reported back to the “sources” of traffic. The sources, in turn, react by throttling down their injection rates, thus preventing the negative effects of fabric buffer saturation and increased queuing delays.

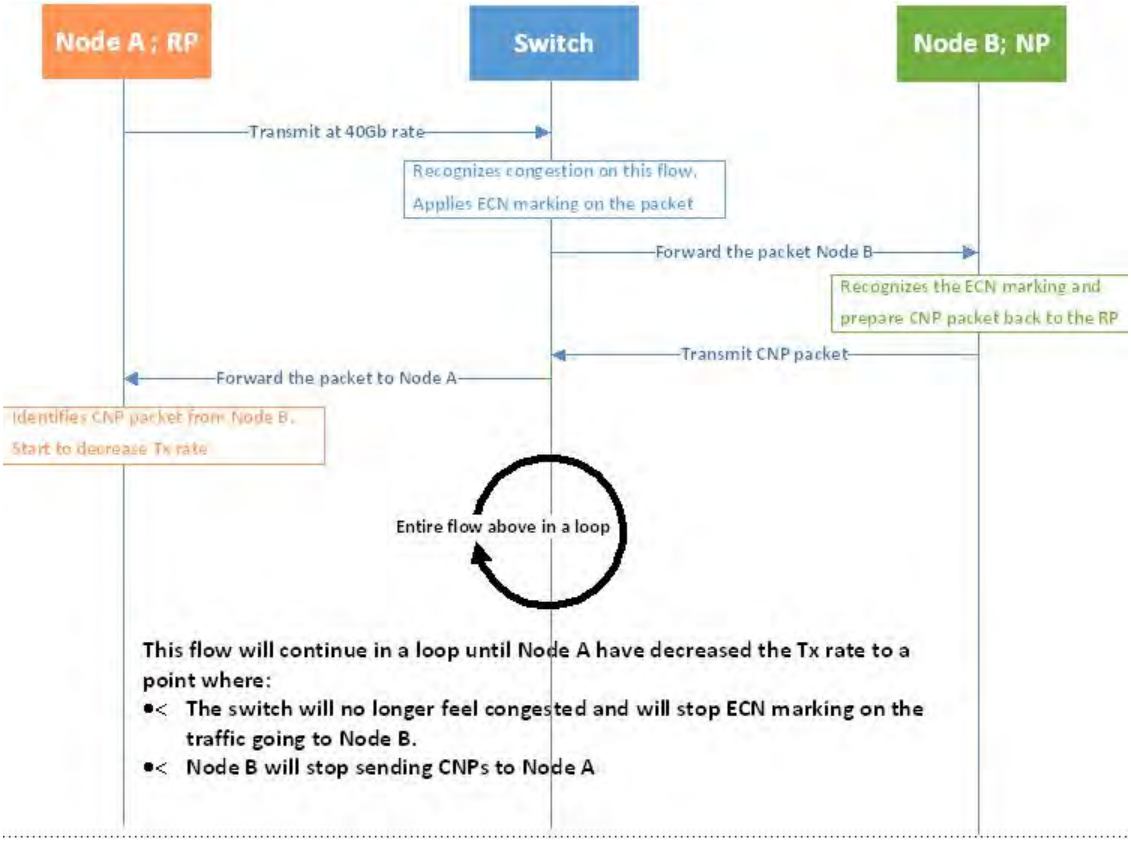
For signaling of congestion, RCM relies on the mechanism defined in RFC3168, also known as DCQCN.

The source node and destination node can be considered as a “closed-loop control” system. Starting from the trigger, when the destination node reflects the congestion alert to the source node, the source node reacts by decreasing, and later on increasing, the Tx rates according to the feedback provided. The source node keeps increasing the Tx rates until the system reaches a steady state of non-congested flow with traffic as high rate as possible.

The RoCEv2 Congestion Management feature is composed of the following points:

- Congestion Point (CP) - detects congestion and marks packets using the DCQCN bits
- Notification Point (NP) (receiving end node) - reacts to the DCQCN marked packets by sending congestion notification packets (CNPs)
- Reaction Point (RP) (transmitting end node) - reduces the transmission rate according to the received CNPs

These components can be seen in the High-Level sequence diagram below:



For further details, please refer to the IBTA RoCeV2 Spec, Annex A-17.

Restrictions and Limitations

	Restrictions and Limitations
General	• In order for RCM to function properly, the elements in the communication path must support and be configured for RCM (nodes) and DCQCN marking (Switches, Routers). • ConnectX®-4 and ConnectX®-4 Lx support congestion control only with RoCEv2. • RCM does not remove/replace the need for flow control. In order for RoCEv2 to work properly, flow control must be configured. It is not recommended to configure RCM without PFC or global pauses.
NVIDIA®	• Minimal firmware version - 2.30 • Minimal driver version - 1.35 • NVIDIA® switch support as of “NVIDIA® Spectrum®” based switch systems • RCM is supported only when using a physical adapter

RCM Configuration

RCM configuration to NVIDIA® adapter is done via mlx5cmd tool.

➢ *To view the current status of RCM on the adapter, run the following command:*

```
Mlx5Cmd.exe -Qosconfig -Dcqcn -Name <Network Adapter Name> -Get
```

Example of RCM being disabled:

```
PS C:\Users\admin\Desktop> Mlx5Cmd.exe -Qosconfig -Dcqcn -Name "Ethernet" -Get
DCQCN RP attributes for adapter "EthernetcnRPEnablePrio0: 0
    DcqcnRPEnablePrio1: 0
    DcqcnRPEnablePrio2: 0
    DcqcnRPEnablePrio3: 0
    DcqcnRPEnablePrio4: 0
    DcqcnRPEnablePrio5: 0
    DcqcnRPEnablePrio6: 0
    DcqcnRPEnablePrio7: 0
    DcqcnClampTgtRate: 0
    DcqcnClampTgtRateAfterTimeInc: 1
    DcqcnRpgTimeReset: 100
    DcqcnRpgByteReset: 400
    DcqcnRpgThreshold: 5
    DcqcnRpgAiRate: 10
    DcqcnRpgHaiRate: 100
    DcqcnAlphaToRateShift: 11
    DcqcnRpgMinDecFac: 50
    DcqcnRpgMinRate: 1
    DcqcnRateToSetOnFirstCnp: 3000
    DcqcnDceTcpG: 32
    DcqcnDceTcpRtt: 4
    DcqcnRateReduceMonitorPeriod: 32
    DcqcnInitialAlphaValue: 0

DCQCN NP attributes for adapter "Ethernet":
    DcqcnNPEnablePrio0: 0
    DcqcnNPEnablePrio1: 0
    DcqcnNPEnablePrio2: 0
    DcqcnNPEnablePrio3: 0
    DcqcnNPEnablePrio4: 0
    DcqcnNPEnablePrio5: 0
    DcqcnNPEnablePrio6: 0
    DcqcnNPEnablePrio7: 0
    DcqcnCnpDscp: 0
    DcqcnCnp802pPrio: 7
    DcqcnCnpPrioMode: 1
The command was executed successfully
```

➢ *To enable/disable DCQCN on the adapter, run the following command:*

```
Mlx5Cmd.exe -Qosconfig -Dcqcn -Name <Network Adapter Name> -Enable/Disable
```

This can be used on all priorities or on a specific priority.

```
PS C:\Users\admin\Desktop> Mlx5Cmd.exe -Qosconfig -Dcqcn -Name "Ethernet" -Enable

PS C:\Users\admin\Desktop> Mlx5Cmd.exe -Qosconfig -Dcqcn -Name "Ethernet" -Get
DCQCN RP attributes for adapter "Ethernet":
    DcqcnRPEnablePrio0: 1
    DcqcnRPEnablePrio1: 1
    DcqcnRPEnablePrio2: 1
    DcqcnRPEnablePrio3: 1
    DcqcnRPEnablePrio4: 1
    DcqcnRPEnablePrio5: 1
    DcqcnRPEnablePrio6: 1
    DcqcnRPEnablePrio7: 1
    DcqcnClampTgtRate: 0
    DcqcnClampTgtRateAfterTimeInc: 1
    DcqcnRpgTimeReset: 100
    DcqcnRpgByteReset: 400
    DcqcnRpgThreshold: 5
    DcqcnRpgAiRate: 10
    DcqcnRpgHaiRate: 100
    DcqcnAlphaToRateShift: 11
    DcqcnRpgMinDecFac: 50
    DcqcnRpgMinRate: 1
    DcqcnRateToSetOnFirstCnp: 3000
    DcqcnDceTcpG: 32
    DcqcnDceTcpRtt: 4
    DcqcnRateReduceMonitorPeriod: 32
    DcqcnInitialAlphaValue: 0

DCQCN NP attributes for adapter "Ethernet":
    DcqcnNPEnablePrio0: 1
    DcqcnNPEnablePrio1: 1
    DcqcnNPEnablePrio2: 1
    DcqcnNPEnablePrio3: 1
    DcqcnNPEnablePrio4: 1
    DcqcnNPEnablePrio5: 1
    DcqcnNPEnablePrio6: 1
    DcqcnNPEnablePrio7: 1
    DcqcnCnpDscp: 0
    DcqcnCnp802pPrio: 7
    DcqcnCnpPrioMode: 1
The command was executed successfully
```

RCM Parameters

The table below lists the parameters that can be configured, their description and allowed values.

Parameter (Type)	Allowed Values
DcqcnEnablePrio0 (BOOLEAN)	0/1
DcqcnEnablePrio1 (BOOLEAN)	0/1
DcqcnEnablePrio2 (BOOLEAN)	0/1
DcqcnEnablePrio3 (BOOLEAN)	0/1
DcqcnEnablePrio4 (BOOLEAN)	0/1
DcqcnEnablePrio5 (BOOLEAN)	0/1
DcqcnEnablePrio6 (BOOLEAN)	0/1
DcqcnEnablePrio7 (BOOLEAN)	0/1
DcqcnClampTgtRate (1 bit)	0/1
DcqcnClampTgtRateAfterTimeInc (1 bit)	0/1
DcqcnCnpDscp (6 bits)	0 - 63
DcqcnCnp802pPrio (3 bits)	0 - 7
DcqcnCnpPrioMode(1 bit)	0/1
DcqcnRpgTimeReset (uint32)	0 - 131071 [uSec]
DcqcnRpgByteReset (uint32)	0 - 32767 [64 bytes]
DcqcnRpgThreshold (uint32)	1 - 31
DcqcnRpgAiRate (uint32)	1 - line rate [Mbit/sec]
DcqcnRpgHaiRate (uint32)	1 - line rate [Mbit/sec]
DcqcnAlphaToRateShift (uint32)	0 - 11
DcqcnRpgMinDecFac (uint32)	0 - 100
DcqcnRpgMinRate (uint32)	0 - line rate
DcqcnRateToSetOnFirstCnp (uint32)	0 - line rate [Mbit/sec]
DcqcnDceTcpG (uint32)	0 - 1023 (fixed point fraction of 1024)
DcqcnDceTcpRtt (uint32)	0 - 131071 [uSec]
DcqcnRateReduceMonitorPeriod (uint32)	0 - UINT32-1 [uSec]
DcqcnInitialAlphaValue (uint32)	0 - 1023 (fixed point fraction of 1024)

An attempt to set a greater value than the parameter's maximum "line rate" value (if exists), will fail. The maximum "line rate" value will be set instead.

RCM Default Parameters

Every parameter has a default value assigned to it. The default value was set for optimal congestion control by NVIDIA®. In order to view the default parameters on the adapter, run the following command:

```
Mlx5Cmd .exe -Qosconfig -Dcqcn -Name <Network Adapter Name> -Defaults
```

RCM with Untagged Traffic

Congestion control for untagged traffic is configured with the port default priority that is used for untagged frames.

The port default priority configuration is done via Mlx5Cmd tool.

Parameter (Type)	Allowed and Default Values	Note
DefaultUntaggedPriority	0 - 7 Default: 0	As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

➢ *To view the current default priority on the adapter, run the following command:*

```
Mlx5Cmd .exe -QoSConfig -DefaultUntaggedPriority -Name -Get
```

➢ *To set the default priority to a specific priority on the adapter, run the following command:*

```
Mlx5Cmd .exe -QoSConfig -DefaultUntaggedPriority -Name -Set
```

Congestion Control Behavior when Changing the Parameters

⚠ Changing the values of the parameters may strongly affect the congestion control efficiency.
Please make sure you fully understand the parameter usage, value and expected results before changing its default value.

CNP Priority

Parameter	Description
fCnpDscp	This parameter changes the priority value on IP level that can be set for CNPs.
DcqcnCnpPrioMode	If this parameter is set to '0', then use DcqcnCnp802pPrio as the priority value (802.1p) on the Ethernet header of generated CNPs. Otherwise, the priority value of CNPs will be taken from received packets that were marked as DCQCN packets.
DcqcnCnp802pPrio	This parameter changes the priority value (802.1p) on the Ethernet header of generated CNPs. Set DcqcnCnpPrioMode to '0' in order to use this priority value.

alpha -"α" = Rate Reduction Factor

The device maintains an "alpha" value per QP. This alpha value estimates the current congestion severity in the fabric.

Parameter	Description
DcqcnInitialAlphaValue	This parameter sets the initial value of alpha that should be used when receiving the first CNP for a flow (expressed in a fixed point fraction of 2^10). The value of alpha is updated once every DcqcnDceTcpRtt, regardless of the reception of a CNP. If a CNP is received during this time frame, alpha value will increase. If no CNP reception happens, alpha value will decrease.
DcqcnDceTcpG/ DcqcnDceTcpRtt	These two parameters maintain alpha. • If a CNP is received on the RP - alpha is increased: *(1-DcqcnDecTcpG)*a + DcqcnDecTcpG* • If no CNP is received for a duration of DcqcnDceTcpRtt microseconds, alpha is decreased: *(1- DcqcnDecTcpG) * alpha*

"RP" Decrease

Changing the DcqcnRateToSetOnFirstCnp parameter determines the Current Rate (CR) that will be set once the first CNP is received.

The rate is updated only once every DcqcnRateReduceMonitorPeriod microseconds (multiple CNPs received during this time frame will not affect the rate) by using the following two formulas:

- $Cr1_{(new)}$ = (1- (α /(2^DcqcnAlphaToRateShift)))*$Cr_{(old)}$
- $Cr2_{(new)}$ = Cr(old)/DcqcnRpgMinDecFac

The maximal reduced rate will be chosen from these two formulas.

The target rate will be updated to the previous current rate according to the behavior stated in section Increase on the "RP".

Parameter	Description
DcqcnRpgMinDecFac	This parameter defines the maximal ratio of decrease in a single step (Denominator: ! zero. Please see formula above).
DcqcnAlphaToRateShift	This parameter defines the decrease rate for a given alpha (see formula above)
DcqcnRpgMinRate	In addition to the DcqcnRpgMinDecFac , the DcqcnRpgMinRate parameter defines the minimal rate value for the entire single flow. **Note:** Setting it to a line rate will disable Congestion Control.

"RP" Increase

RP increases its sending rate using a timer and a byte counter. The byte counter increases rate for every DcqcnRpgByteResetx64 bytes (mark it as B), while the timer increases rate every DcqcnRpgTimeReset time units (mark it as T). Every successful increase due to bytes transmitted/time passing is counted in a variable called rpByteStage and rpTimeStage (respectively).

The DcqcnRpgThreshold parameter defines the number of successive increase iteration (mark it as Th). The increase flow is divided into 3 types of phases, which are actually states in the "RP Rate Control State Machine". The transition between the steps is decided according to DcqcnRpgThreshold parameter.

- Fast Recovery
  If MAX (rpByteStage, rpTimeStage) < Th.
  No change to Target Rate (Tr)
- Additive Increase
  If MAX (rpByteStage, rpTimeStage) > Th. && MIN (rpByteStage, rpTimeStage) < Th.
  DcqcnRpgAiRate value is used to increase Tr
- Hyper Additive Increase
  If MAX (rpByteStage, rpTimeStage) > Th. && MIN (rpByteStage, rpTimeStage) > Th.
  DcqcnRpgHaiRate value is used to increase Tr

For further details, please refer to 802.1Qau standard, sections 32.11-32.15.

Parameter	Description
DcqcnClampTgtRateAfterTimeInc	When receiving a CNP, the target rate should be updated if the transmission rate was increased due to the timer, and not only due to the byte counter.
DcqcnClampTgtRate	If set, whenever a CNP is processed, the target rate is updated to be the current rate.

NVIDIA® Commands and Examples

For a full description of Congestion Control commands please refer to section MlxCmd Utilities.

Set a value for one or more parameters:	
Command	`Mlx5Cmd.exe -Qosconfig -Dcqcn -Name <Network Adapter Name> -Set -Arg1 <value> -Arg2 <value>`
Example	`PS C:\Users\admin\Desktop> Mlx5Cmd .exe -Qosconfig -Dcqcn -Name "Ethernet" -Set -DcqcnClampTgtRate 1 -DcqcnCnpDscp 3`
Enable/Disable DCQCN for a specific priority:	
Command	`Mlx5Cmd.exe -Qosconfig -Dcqcn -Name <Network Adapter Name> -Enable <prio>`
Example	`PS C:\Users\admin\Desktop> Mlx5Cmd .exe -Qosconfig -Dcqcn -Name "Ethernet" -Enable/Disable 3`
Enable/Disable DCQCN for all priorities:	
Command	`Mlx5Cmd.exe -Qosconfig -Dcqcn -Name <Network Adapter Name> -Enable`
Example	`PS C:\Users\admin\Desktop> Mlx5Cmd .exe -Qosconfig -Dcqcn -Name "Ethernet" -Enable/Disable`
Set port default priority for a specific priority:	
Command	`Mlx5Cmd.exe -Qosconfig -DefaultUntaggedPriority -Name <Network Adapter Name> -Set <prio>`
Example	`PS C:\Users\admin\Desktop> Mlx5Cmd .exe -Qosconfig -DefaultUntaggedPriority -Name "Ethernet" -Set 3`
Restore the default settings of DCQCN the are defined by NVIDIA®:	
Command	`Mlx5Cmd.exe -Dcqcn -Name <Network Adapter Name> -Restore`
Example	`PS C:\Users\admin\Desktop> Mlx5Cmd .exe -Dcqcn -Name "Ethernet" -Restore`

For information on the RCM counters, please refer to section WinOF-2 Congestion Control.

Enhanced Connection Establishment

Enhanced Connection Establishment (ECE) is a new negotiation scheme introduced in IBTA v1.4 to exchange extra information about nodes capabilities and later negotiate them at the connection establishment phase. ECE is intended for RDMA connection, i.e., it works in ND and NDK connections.

This capability is supported in ConnectX-6 Dx and above adapter cards.

ECE is used by the driver by default if ECE is supported by the firmware. The feature is enabled/disabled by setting the "EceSupportEnabled".

The registry key should be added to `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11cebfc1-08002be10318}\<IndexValue>`

To find the IndexValue, refer to section Finding the Index Value of the Network Interface.

Value Name	Value	Description
EceSupportEnabled	0 - Disabled 1 - Enabled (Default)	Enables/Disables ECE support in the driver.

To check if the feature is supported by the firmware, run:

```
mlx5cmd -dbg -FwCaps -name "Ethernet 8" -Local | grep "ece "
```

This capability can be disabled by the NV configuration as well.

To check whether Selective Repeat algorithm is supported, run:

```
mlxconfig -d mt4125_pciconf0 | grep RDMA_SELECTIVE_REPEAT_EN
```

To enable ECE algorithm using the NV configuration:

```
mlxconfig -d mt4125_pciconf0 -y s ROCE_CC_LEGACY_DCQCN=0
mlxconfig -d mt4125_pciconf0 -y s USER_PROGRAMMABLE_CC=1 RDMA_SELECTIVE_REPEAT_EN=1
```

To check the feature's status, run:

```
mlx5cmd -Features
```

The command prints information on ECE feature: either "Enabled", "Disabled" or "Reason". The "Reason" can be either the firmware does not support this capability or ECE is disabled using the registry key.

To check ECE's state in the existing ND/NDK connections, run:

```
mlx5cmd -ndkstat -ece
mlx5cmd -ndstat -ece
```

ECE value is printed on every connection (at the end of the line).

Zero Touch RoCE

Zero touch RoCE enables RoCE to operate on fabrics where no PFC nor ECN are configured. This makes RoCE configuration a breeze while still maintaining its superior high performance.
Zero touch RoCE enables:

- Packet loss minimization by:
  - Developing a congestion handling mechanism which is better adjusted to a lossy environment
  - Moving more of the congestion handling mechanism to the hardware and to the dedicated microcode
  - Moderating traffic bursts by tuning of transmission window and slow restart of transmission
- Protocol packet loss handling improvement by:
  - ConnectX-4: Repeating transmission from a lost segment of a IB retransmission protocol
  - ConnectX-5 and above: Improving the response to the packet loss by using hardware re-transmission
  - ConnectX-6 Dx: Using a proprietary selective repeat protocol

Facilities

Zero touch RoCE contains the following facilities, used to enable the above algorithms.

- SlowStart: Start a re-transmission with low bandwidth and gradually increase it
- AdpRetrans: Adjust re-transmission parameters according to network behavior
- TxWindow: Automatic tuning of the transmission window size

The facilities can be independently enabled or disabled. The change is persistent, i.e. the configuration does not change after the driver restart. By default, all the facilities are enabled.

Restrictions and Limitations

- Currently, Zero touch RoCE is supported only for the Ethernet ports, supporting RoCE
- The required firmware versions are: 1x.25.xxxx and above.
- ConnectX-4/ConnectX-4 Lx, supports only the following facilities: SlowStart and AdpRetrans

Configuring Zero touch RoCE

Zero touch RoCE is configured using the mlx5cmd tool.

- To view the status of the Zero touch RoCE on the adapter.

```
Mlx5Cmd.exe -ZtRoce -Name <Network Adapter Name> -Get
```

The output below shows the current state, which is limited by the firmware capabilities and the last state set.

```
Current configuration for Adapter 'Ethernet':
AdpRetrans Disabled
TxWindow Disabled
SlowStart Enabled
```

- To view the firmware capabilities regarding Zero touch RoCE.

```
Mlx5Cmd.exe -ZtRoce -Name <Network Adapter Name> -Caps
```

The output below is characteristic to ConnectX-4 adapter cards where only two facilities are supported:

```
FW capabilities for Adapter 'Ethernet':
AdpRetrans Enabled
TxWindow Disabled
SlowStart Enabled
```

- To view the software default settings.

```
Mlx5Cmd.exe -ZtRoce -Name <Network Adapter Name> -Defaults
```

The output below shows Zero touch RoCE default settings.

```
Default configuration for Adapter 'Ethernet':
AdpRetrans Enabled
TxWindow Enabled
SlowStart Enabled
```

Configuring Zero touch RoCE Facilities

The facilities states can be enabled or disabled using the following format:

```
Mlx5Cmd -ZtRoce -Name <Network Adapter Name> -Set [-AdpRetrans 0 | 1 ] [-TxWindow 0 | 1 ] [-SlowStart 0 | 1]
```

The example below shows how you can enable Slow Restart and Transmission Window facilities and disable the Adaptive Re-transmission.

```
Mlx5Cmd -ZtRoce -Name "Ethernet 3" -Set -AdpRetrans 0 -TxWindow 1 -SlowStart 1
```

- To disable all the facilities.

```
Mlx5Cmd -ZtRoce -Name <Network Adapter Name> -Disable
```

- To enable all the facilities.

```
Mlx5Cmd -ZtRoce -Name <Network Adapter Name> -Enable
```

- To restore the default values.

```
Mlx5Cmd -ZtRoce -Name <Network Adapter Name> -Restore
```

Facilities cannot be enabled if the firmware does not support this feature.

For further information, refer to the feature help page: *Mlx5Cmd -ZtRoce -h*

Teaming and VLAN

Windows Server 2012 and above supports Teaming as part of the operating system. Please refer to Microsoft guide "NIC Teaming in Windows Server 2012".

Note that the Microsoft teaming mechanism is only available on Windows Server distributions.

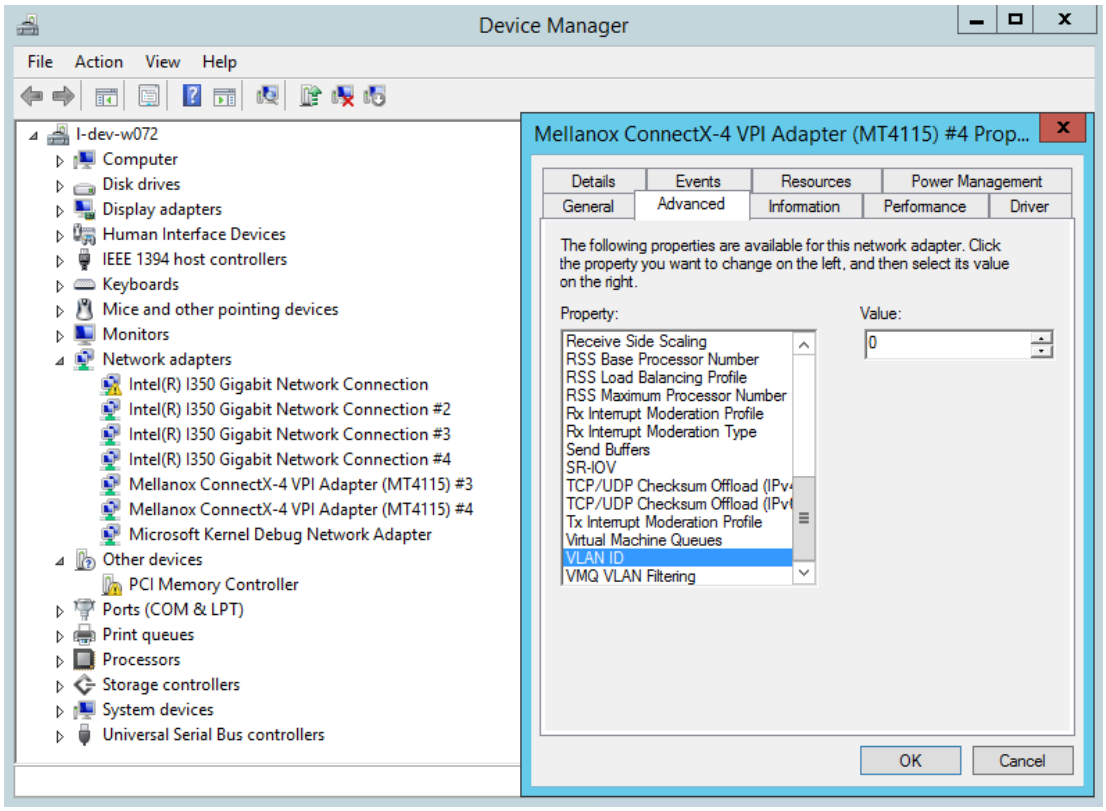
Configuring a Network to Work with VLAN in Windows Server 2012 and Above

In this procedure you DO NOT create a VLAN, rather use an existing VLAN ID.

To configure a port to work with VLAN using the Device Manager:

1. Open the Device Manager.
2. Go to the Network adapters.
3. Go to the properties of NVIDIA® ConnectX®-4 Ethernet Adapter card.
4. Go to the Advanced tab.
5. Choose the VLAN ID in the Property window.

6. Set its value in the Value window.



Command Line Based Teaming Configuration

NIC Teaming

NIC Teaming allows you to group between one and 32 physical Ethernet/IPoIB network adapters into one or more software-based virtual network adapters. These virtual network adapters provide fast performance and fault tolerance in the event of a network adapter failure.
On Windows Server edition, there is a built-in module that supports teaming and VLAN Ethernet. For more information see here.

One of the existing limitations with Windows OS support is that it does not support NIC teaming for Windows client editions.

> ⚠ A team can either have only IPoIB members or only Ethernet members.

To overcome these limitations, WinOF-2 driver provides a teaming solution. The supported modes are:

- Static Link aggregate mode – In this mode, all the team members can send and receive traffic. The underlying adapter to which packet to post is forwarded is based on a hash value obtained from the NET_BUFFER_LIST structure, module number of the underlying adapters.

- Active-Standby mode - In this mode, the user can pick the primary adapter responsible for sending traffic. In the event of a link failure, a failover happens and the standby adapter takes over. User can also tell us to not failback to primary in case of fail-over followed by primary adapter has link up. For this mode, set the failover mode in the mlx5MuxTool.

Please refer to content below on how to configure using custom teaming solution with RSS functionality.

Prerequisites

- Operating Systems: Windows 10 and above

- Adapter Cards: All supported devices supported by WinOF-2 driver.
- For using the Mlx5muxtool, users that do not install the full package (HKEY_LOCAL_MACHINE\SOFTWARE\Mellanox\MLNX_WinOF2\InstalledPath) must point this key (InstalledPath) to the location of the mux drivers as the tool searches for the mux drivers files in a folder called "mux" in the folder that define by this key(InstalledPath).

Configuring Command Line Based Teaming

1. Show the help menu. The following command prints out all supported modes and functionalities:

```
mlx5muxtool.exe --help
[TEAMING]
To list all adapters including teams, use:
       mlx5muxtool showlist
To create a team use:
       mlx5muxtool create team <Type> <Name> [NoFailBackToPrimary] [IPoIB]
          Type is one of the following: Aggregate | Failover
          For IPoIB team, only type 'Failover' is supported

To add adapter to the team use:
       mlx5muxtool attach team <TeamName> {<Adapter-GUID>} [primary] [SetTeamMacAddress]
To remove an adapter from the team use:
       mlx5muxtool detach team <TeamName> {<Adapter-GUID>}
To delete a team use:
       mlx5muxtool removeteam <TeamName>
To query an existing team, use:
       mlx5muxtool queryteam <TeamName>
```

Example:

```
mlx5muxtool create team Aggregate MyTeam
mlx5muxtool attach team MyTeam {2E9C1992-98B5-43C3-97A0-9993AEAC7F80}
mlx5muxtool attach team MyTeam {8D05C52B-BCD6-4FCE-8235-1E90BD334519}
```

2. Show all the adapter cards (including all created teams already).

```
mlx5muxtool.exe showlist
{90F5F52D-4384-4263-BD12-4588CA5CE80A}          Mellanox ConnectX-5 Adapter #2 (IPoIB)
{62B9661A-17C4-4AF3-AAA1-2B3337FD02E0}          Mellanox ConnectX-5 Adapter (IPoIB)
{136A1E6F-1168-48D4-B9CC-55EE563D427B}          Mellanox ConnectX-6 Adapter (IPoIB)
{87B55F92-D573-471B-882C-379773296A6D}          Mellanox ConnectX-6 Adapter #2 (IPoIB)
```

3. Create an empty Ethernet team.

```
mlx5muxtool.exe create team aggregate MyTeam
Adding team MyTeam
Team created with Guid = AC956713-F772-4C6B-AB13-6178BB0E3BDC
```

4. Attach members to the team.

```
mlx5muxtool.exe attach team MyTeam {90F5F52D-4384-4263-BD12-4588CA5CE80A} primary
Attaching adapter {90F5F52D-4384-4263-BD12-4588CA5CE80A} to team MyTeam
```

5. Query the team.

```
mlx5muxtool.exe queryteam MyTeam

Found 1 team(s)

Name               : MyTeam
GUID               : {FED1925F-F88F-4970-B4C3-38AA030874DF}
PortType           : IPoIB
TeamType           : Failover
MemberCount        : 2
Member[0]          : {62B9661A-17C4-4AF3-AAA1-2B3337FD02E0} (SLOT 5 Port 2)
Member[1]          : {90F5F52D-4384-4263-BD12-4588CA5CE80A} (Primary) (SLOT 5 Port 1)
```

6. Detach members from the team.

```
mlx5muxtool.exe detach team MyTeam {62B9661A-17C4-4AF3-AAA1-2B3337FD02E0}
Dettaching adapter {62B9661A-17C4-4AF3-AAA1-2B3337FD02E0} from team MyTeam
```

7. Remove an entire team.

```
mlx5muxtool.exe removeteam MyTeam
Delete team MyTeam
Deleting member {90F5F52D-4384-4263-BD12-4588CA5CE80A}
```

VLAN Support

WinOF-2 v2.30 supports configuring only a single VLAN to a team interface. VLAN tagging is disabled by default.

To tag all the outgoing packets with VLAN, “VlanID” registry key should be set to a non-zero value.

- Find the registry key index value of the team (virtual adapter) according to section Finding the Index Value of the Network Interface.
- Set the VlanID key in the following path HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<IndexValue>

Parameter Name	Parameter Type	Description	Allowed values and Defaults
VlanID	DWORD	The tag that transmits the packets with the value in the registry.	• 0: No Tag (Default) • 1 - 4094 : VLAN ID to be inserted by the underlying miniport hardware.
VlanPrio	DWORD	The priority field of the VLAN header to be inserted.	• 0 - 7. 0 is the default value.

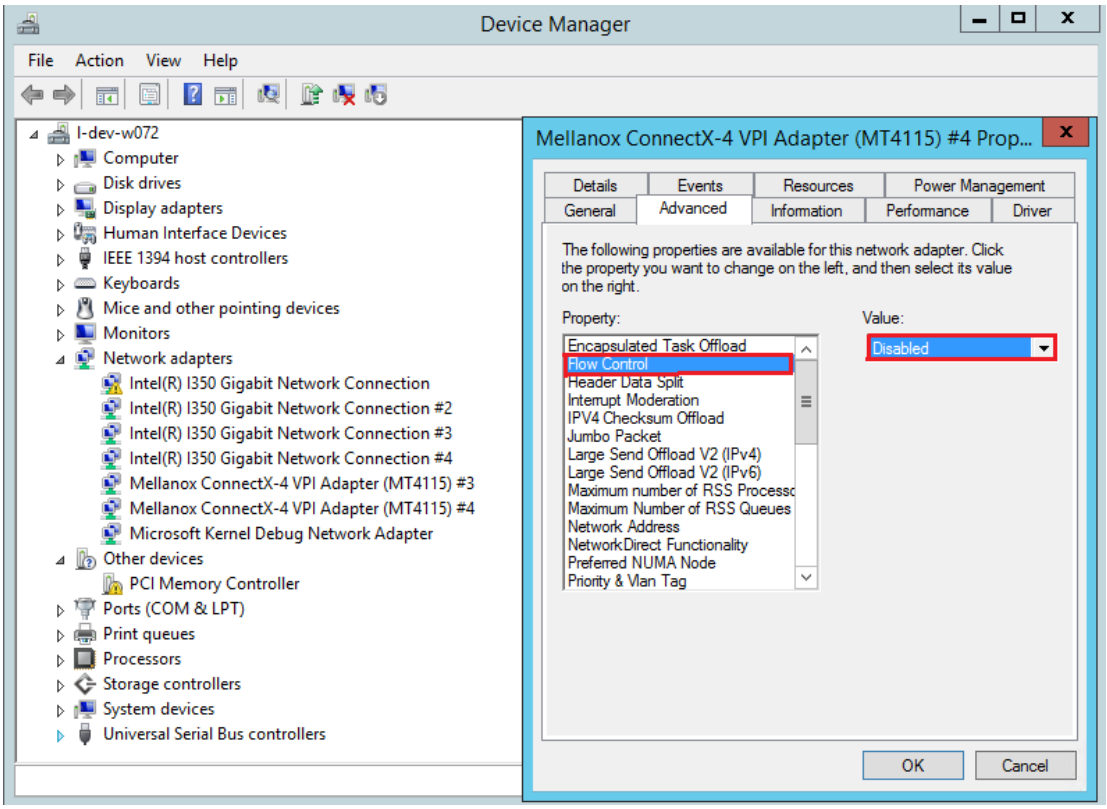
Configuring Quality of Service (QoS)

QoS Configuration

Prior to configuring Quality of Service, you must install Data Center Bridging using one of the following methods:

Disabling Flow Control Configuration

Device manager->Network adapters->Mellanox ConnectX-4/ConnectX-5 Ethernet Adapter->Properties->Advanced tab



Installing the Data Center Bridging using the Server Manager

1. Open the 'Server Manager'.
2. Select 'Add Roles and Features'.
3. Click Next.
4. Select 'Features' on the left panel.
5. Check the 'Data Center Bridging' checkbox.
6. Click 'Install'.

Installing the Data Center Bridging using PowerShell

Enable Data Center Bridging (DCB).

```
PS $ Install-WindowsFeature Data-Center-Bridging
```

Configuring QoS on the Host

> ⚠ The procedure below is not saved after you reboot your system. Hence, we recommend you create a script using the steps below and run it on the startup of the local machine. Please see the procedure below on how to add the script to the local machine startup scripts.

1. Change the Windows PowerShell execution policy.

   ```
   PS $ Set-ExecutionPolicy AllSigned
   ```

2. Remove the entire previous QoS configuration.

   ```
   PS $ Remove-NetQosTrafficClass
   PS $ Remove-NetQosPolicy -Confirm:$False
   ```

3. Create a Quality of Service (QoS) policy and tag each type of traffic with the relevant priority.
   In this example, TCP/UDP use priority 1, SMB over TCP use priority 3.

   ```
   PS $ New-NetQosPolicy "DEFAULT" -store Activestore -Default -PriorityValue8021Action 3
   PS $ New-NetQosPolicy "TCP" -store Activestore -IPProtocolMatchCondition TCP -PriorityValue8021Action 1
   PS $ New-NetQosPolicy "UDP" -store Activestore -IPProtocolMatchCondition UDP -PriorityValue8021Action 1
   New-NetQosPolicy “SMB” -SMB -PriorityValue8021Action 3
   ```

4. Create a QoS policy for SMB over SMB Direct traffic on Network Direct port 445.

   ```
   PS $ New-NetQosPolicy "SMBDirect" -store Activestore -NetDirectPortMatchCondition 445
    -PriorityValue8021Action 3
   ```

5. [Optional] If VLANs are used, mark the egress traffic with the relevant VlanID.
   The NIC is referred as *"Ethernet 4”* in the examples below.

   ```
   PS $ Set-NetAdapterAdvancedProperty -Name "Ethernet 4" -RegistryKeyword "VlanID" -RegistryValue "55"
   ```

6. [Optional] Configure the IP address for the NIC.
   If DHCP is used, the IP address will be assigned automatically.

   ```
   PS $ Set-NetIPInterface -InterfaceAlias “Ethernet 4” -DHCP Disabled
   PS $ Remove-NetIPAddress -InterfaceAlias “Ethernet 4” -AddressFamily IPv4 -Confirm:$false
   PS $ New-NetIPAddress -InterfaceAlias “Ethernet 4” -IPAddress 192.168.1.10 -PrefixLength 24 -Type Unicast
   ```

7. [Optional] Set the DNS server (assuming its IP address is 192.168.1.2).

   ```
   PS $ Set-DnsClientServerAddress -InterfaceAlias “Ethernet 4” -ServerAddresses 192.168.1.2
   ```

   > ⚠ After establishing the priorities of ND/NDK traffic, the priorities must have PFC enabled on them.

8. Disable Priority Flow Control (PFC) for all other priorities except for 3.

```
PS $ Disable-NetQosFlowControl 0,1,2,4,5,6,7
```

9. Enable QoS on the relevant interface.

```
PS $ Enable-NetAdapterQos -InterfaceAlias "Ethernet 4"
```

10. Enable PFC on priority 3.

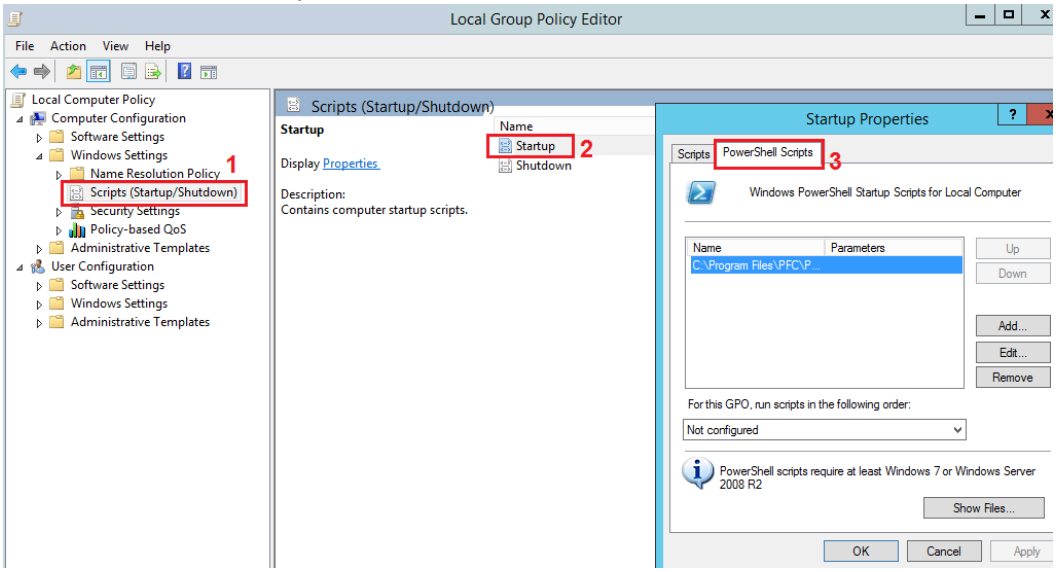
```
PS $ Enable-NetQosFlowControl -Priority 3
```

Adding the Script to the Local Machine Startup Scripts

1. From the PowerShell invoke.

```
gpedit.msc
```

2. In the pop-up window, under the 'Computer Configuration' section, perform the following:
   a. Select Windows Settings.
   b. Select Scripts (Startup/Shutdown).
   c. Double click Startup to open the Startup Properties.
   d. Move to “PowerShell Scripts” tab.



   e. Click Add.
   The script should include only the following commands:

```
PS $ Remove-NetQosTrafficClass
PS $ Remove-NetQosPolicy -Confirm:$False
PS $ set-NetQosDcbxSetting -Willing 0
PS $ New-NetQosPolicy "SMB" -Policystore Activestore -NetDirectPortMatchCondition 445
 -PriorityValue8021Action 3
PS $ New-NetQosPolicy "DEFAULT" -Policystore Activestore -Default -PriorityValue8021Action 3
PS $ New-NetQosPolicy "TCP" -Policystore Activestore -IPProtocolMatchCondition TCP
-PriorityValue8021Action 1
PS $ New-NetQosPolicy "UDP" -Policystore Activestore -IPProtocolMatchCondition UDP
-PriorityValue8021Action 1
PS $ Disable-NetQosFlowControl 0,1,2,4,5,6,7
PS $ Enable-NetAdapterQos -InterfaceAlias "port1"
PS $ Enable-NetAdapterQos -InterfaceAlias "port2"
PS $ Enable-NetQosFlowControl -Priority 3
PS $ New-NetQosTrafficClass -name "SMB class" -priority 3 -bandwidthPercentage 50 -Algorithm ETS
```

   f. Browse for the script's location.

g. Click OK
h. To confirm the settings applied after boot run:

```
PS $ get-netqospolicy -policystore activestore
```

Enhanced Transmission Selection (ETS)

Enhanced Transmission Selection (ETS) provides a common management framework for assignment of bandwidth to frame priorities as described in the IEEE 802.1Qaz specification.

For further details on configuring ETS on Windows™ Server, please refer to: http://technet.microsoft.com/en-us/library/hh967440.aspx

Differentiated Services Code Point (DSCP)

DSCP is a mechanism used for classifying network traffic on IP networks. It uses the 6-bit Differentiated Services Field (DS or DSCP field) in the IP header for packet classification purposes. Using Layer 3 classification enables you to maintain the same classification semantics beyond local network, across routers.

Every transmitted packet holds the information allowing network devices to map the packet to the appropriate 802.1Qbb CoS. For DSCP based PFC or ETS, the packet is marked with a DSCP value in the Differentiated Services (DS) field of the IP header. In case DSCP is enabled, QoS traffic counters are incremented based on the DSCP mapping described in section Receive Trust State.

System Requirements	
Operating Systems:	Windows Server 2012 and onward
Firmware version:	12/14/16.18.1000 and higher

Setting the DSCP in the IP Header

Marking the DSCP value in the IP header is done differently for IP packets constructed by the NIC (e.g. RDMA traffic) and for packets constructed by the IP stack (e.g. TCP traffic).

- For IP packets generated by the IP stack, the DSCP value is provided by the IP stack. The NIC does not validate the match between DSCP and Class of Service (CoS) values. CoS and DSCP values are expected to be set through standard tools, such as PowerShell command New-NetQosPolicy using PriorityValue8021Action and DSCPAction flags respectively.
- For IP packets generated by the NIC (RDMA), the DSCP value is generated according to the CoS value programmed for the interface. CoS value is set through standard tools, such as PowerShell command New-NetQosPolicy using PriorityValue8021Action flag. The NIC uses a mapping table between the CoS value and the DSCP value configured through the RroceDscpMarkPriorityFlow- Control[0-7] Registry keys

Configuring Quality of Service for TCP and RDMA Traffic

1. Verify that DCB is installed and enabled (is not installed by default).

```
PS $ Install-WindowsFeature Data-Center-Bridging
```

2. Import the PowerShell modules that are required to configure DCB.

```
PS $ import-module NetQos
PS $ import-module DcbQos
PS $ import-module NetAdapter
```

3. Enable Network Adapter QoS.

```
PS $ Set-NetAdapterQos -Name "CX4_P1" -Enabled 1
```

4. Enable Priority Flow Control (PFC) on the specific priority 3,5.

```
PS $ Enable-NetQosFlowControl 3,5
```

Configuring DSCP to Control PFC for TCP Traffic

Create a QoS policy to tag All TCP/UDP traffic with CoS value 3 and DSCP value 9.

```
PS $ New-NetQosPolicy "DEFAULT" -Default -PriorityValue8021Action 3 -DSCPAction 9
```

DSCP can also be configured per protocol.

```
PS $ New-NetQosPolicy "TCP" -IPProtocolMatchCondition TCP -PriorityValue8021Action 3 -DSCPAction 16
PS $ New-NetQosPolicy "UDP" -IPProtocolMatchCondition UDP -PriorityValue8021Action 3 -DSCPAction 32
```

Configuring DSCP to Control ETS for TCP Traffic

- Create a QoS policy to tag All TCP/UDP traffic with CoS value 0 and DSCP value 8.

```
PS $ New-NetQosPolicy "DEFAULT" -Default -PriorityValue8021Action 0 -DSCPAction 8 -PolicyStore activestore
```

- Configure DSCP with value 16 for TCP/IP connections with a range of ports.

```
PS $ New-NetQosPolicy "TCP1" -DSCPAction 16 -IPDstPortStartMatchCondition 31000 -IPDstPortEndMatchCondition
31999 -IPProtocol TCP -PriorityValue8021Action 0 -PolicyStore activestore
```

- Configure DSCP with value 24 for TCP/IP connections with another range of ports.

```
PS $ New-NetQosPolicy "TCP2" -DSCPAction 24 -IPDstPortStartMatchCondition 21000 -IPDstPortEndMatchCondition
31999 -IPProtocol TCP -PriorityValue8021Action 0 -PolicyStore activestore
```

- Configure two Traffic Classes with bandwidths of 16% and 80%.

```
PS $ New-NetQosTrafficClass -name "TCP1" -priority 3 -bandwidthPercentage 16 -Algorithm ETS
PS $ New-NetQosTrafficClass -name "TCP2" -priority 5 -bandwidthPercentage 80 -Algorithm ETS
```

Configuring DSCP to Control PFC for RDMA Traffic

Create a QoS policy to tag the ND traffic for port 10000 with CoS value 3.

```
PS $ New-NetQosPolicy "ND10000" -NetDirectPortMatchCondition 10000 - PriorityValue8021Action 3
```

Related Commands	
Get-NetAdapterQos	Gets the QoS properties of the network adapter
Get-NetQosPolicy	Retrieves network QoS policies
Get-NetQosFlowControl	Gets QoS status per priority

Receive Trust State

Received packets Quality of Service classification can be done according to the DSCP value, instead of PCP, using the RxTrustedState registry key. The mapping between wire DSCP values to the OS priority (PCP) is static, as follows:

DSCP Value	Priority
0-7	0
8-15	1
16-23	2
24-31	3
32-39	4
40-47	5
48-55	6
56-63	7

When using this feature, it is expected that the transmit DSCP to Priority mapping (the PriorityToDscpMappingTable _* registry key) will match the above table to create a consistent mapping on both directions.

DSCP Based QoS

DSCP Based QoS can be enable by setting the following registry keys and mapping the DscpToPriorityMappingTable keys according to the table below.

- DscpBasedEtsEnabled = 1
- RxTrustedState = 2

Registry name	DSCP Default Value	Mapped to Priority
DscpToPriorityMappingTable_0	0	0
DscpToPriorityMappingTable_1	1	0
DscpToPriorityMappingTable_2	2	0
DscpToPriorityMappingTable_3	3	3
DscpToPriorityMappingTable_4	4	4
DscpToPriorityMappingTable_5	5	0
DscpToPriorityMappingTable_6	6	0
DscpToPriorityMappingTable_7	7	0
DscpToPriorityMappingTable_8 … DscpToPriorityMappingTable_15	8 to 15 respectively	1

Registry name	DSCP Default Value	Mapped to Priority
DscpToPriorityMappingTable_16 … DscpToPriorityMappingTable_ 23	16 to 23 respectively	2
DscpToPriorityMappingTable_24 … DscpToPriorityMappingTable _31	24 to 31 respectively	3
DscpToPriorityMappingTable _32 … DscpToPriorityMappingTable_39	32 to 39 respectively	4
DscpToPriorityMappingTable _40 … DscpToPriorityMappingTable _47	40 to 47 respectively	5
DscpToPriorityMappingTable _48 … DscpToPriorityMappingTable _55	48 to 55 respectively	6
DscpToPriorityMappingTable _56 … DscpToPriorityMappingTable_63	56 to 63 respectively	7

DSCP Based QoS: Ethernet (in DSCP Trust Mode)

The following is the DSCP Based QoS Ethernet behavior:

- On the Receive side, the hardware will look at the DSCP value of the packet and map it to correct priority based on the DscpToPriorityMappingTable programmed. e.g.: DSCP of 26 is mapped to priority 3
- On the Transmit side, the driver will read the DSCP value and choose the ring/priority based on the DscpToPriorityMappingTable. e.g.: DSCP value of 20 is mapped to priority 2

DSCP Based QoS: RDMA (in DSCP Trust Mode)

The following is the DSCP Based QoS RDMA behavior:

- Transmit: When QoS at user level is not configured, and no priority exists in the packet, the driver will insert the default DSCP value (26 today) for the packet to go out with. The default DSCP value is controlled by the DscpForGlobalFlowControl registry key. Hardware will perform a lookup of DSCP 26 in the DscpToPriorityMappingTable we programmed and send it out on priority 3.
- Transmit: When QoS at user level is configured, and priority exists in the packet, the driver will perform the lookup in PriorityToDscpMappingTable to insert the mapped DSCP value. Packets will go out with this mapped DSCP value instead of the default DSCP value. e.g.: If a packet arrives with priority 3, the driver will insert a DSCP value of 3 before it goes into the wire.
- Receive: The hardware on the receive side will look at the DSCP value of the packet and map it to the correct priority based on the mapping above.

Registry Settings

The following attributes must be set manually and will be added to the miniport registry.

For more information on configuring registry keys, see section Configuring the Driver Registry Keys.

Registry Key	Description
TxUntagPriorityTag	If 0x1, do not add 802.1Q tag to transmitted packets which are assigned 802.1p priority, but are not assigned a non-zero VLAN ID (i.e. priority-tagged). Default: 0x0. For DSCP based PFC set to 0x1. **Note:** These packets will count on the original priority, even if the registry is on.
RxUntaggedMapToLossless	If 0x1, all untagged traffic is mapped to the lossless receive queue. Default 0x0, for DSCP based PFC set to 0x1. **Note:** This key is only relevant when in PCP mode. **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
PriorityToDscpMappingTable_<ID>	A value to mark DSCP for RoCE packets assigned to CoS=ID, when priority flow control is enabled. The valid values range is from 0 to 63, Default is ID value, e.g. PriorityToDscpMappingTable_3 is 3. ID values range from 0 to 7.
DscpToPriorityMappingTable_<ID>	DscpToPriorityMappingTable_0 to DscpToPriorityMappingTable_63 are 64 registry keys used to set DSCP Based QoS priorities according to the mapping specified in DSCP Based QoS. The user can change this by creating a registry key and overwriting the value.
DscpBasedEtsEnabled	If 0x1 - all DSCP based ETS feature is enabled, if 0x0 - disabled. Default 0x0.
DscpBasedpfcEnabled	If set, the DSCP value on the ROCE packet will be based according to the priority set.
DscpForGlobalFlowControl	Default DSCP value for flow control. Default 0x1a.
RxTrustedState	Default using host priority (PCP) is 1 Default using DSCP value is 2

⚠ For changes to take effect, restart the network adapter after changing any of the above registry keys.

Default Settings

When DSCP configuration registry keys are missing in the miniport registry, the following defaults are assigned:

Registry Key	Default Value
TxUntagPriorityTag	0
RxUntaggedMapToLossles	0
PriorityToDscpMappingTable_0	0
PriorityToDscpMappingTable_1	1
PriorityToDscpMappingTable_2	2
PriorityToDscpMappingTable_3	3
PriorityToDscpMappingTable_4	4
PriorityToDscpMappingTable_5	5
PriorityToDscpMappingTable_6	6
PriorityToDscpMappingTable_7	7
DscpBasedEtsEnabled	eth:0
DscpForGlobalFlowControl	26

Receive Segment Coalescing (RSC)

RSC allows reduction of CPU utilization when dealing with large TCP message size. It allows the driver to indicate to the Operating System once, per-message and not per-MTU that Packet Offload can be disabled for IPv4 or IPv6 traffic in the Advanced tab of the driver proprieties.

RSC provides diagnostic counters documented at : Receive Segment Coalescing (RSC)

Wake-on-LAN (WoL)

Wake-on-LAN is a technology that allows a network admin to remotely power on a system or to wake it up from sleep mode by a network message. WoL is enabled by default.

To check whether or not WoL is supported by adapter card:

1. Check if mlxconfig recognizes the feature.

```
mlxconfig -d /dev/mst/mt4117_pciconf0 show_confs
```

2. Check if the firmware used in your system supports WoL.

```
mlxconfig -d /dev/mst/mt4117_pciconf0 query
```

Data Center Bridging Exchange (DCBX)

Data Center Bridging Exchange (DCBX) protocol is an LLDP based protocol which manages and negotiates host and switch configuration. The WinOF-2 driver supports the following:

- PFC - Priority Flow Control
- ETS - Enhanced Transmission Selection
- Application priority

The protocol is widely used to assure lossless path when running multiple protocols at the same time. DCBX is functional as part of configuring QoS mentioned in section Configuring Quality of Service (QoS). Users should make sure the willing bit on the host is enabled, using PowerShell if needed:

```
set-NetQosDcbxSetting -Willing 1
```

This is required to allow negotiating and accepting peer configurations. Willing bit is set to 1 by default by the operating system. The new settings can be queried by calling the following command in PowerShell.

```
Get-NetAdapterQos
```

The below configuration was received from the switch in the below example.

The output would look like the following:

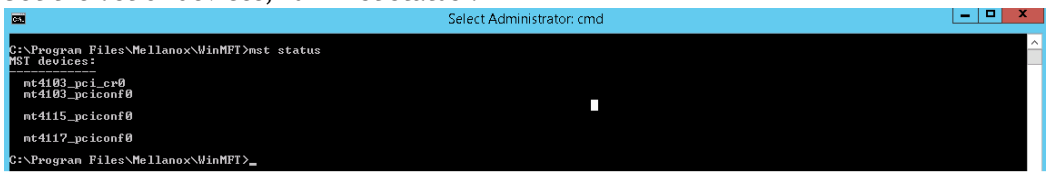
```
PS C:\Users\Administrator> get-netadapterqos


Name    : Ethernet 9
Enabled : True

Name    : Ethernet 10
Enabled : True

Name                        : Ethernet 7
Enabled                     : True
Capabilities                :                          Hardware     Current
                                                       --------     -------
                              MacSecBypass             : NotSupported NotSupported
                              DcbxSupport              : IEEE         IEEE
                              NumTCs(Max/ETS/PFC)      : 8/8/8        8/8/8

OperationalTrafficClasses   : TC TSA    Bandwidth Priorities
                              -- ---    --------- ----------
                               0 ETS    25%       0-1
                               1 ETS    25%       2-3
                               2 ETS    25%       4-5
                               3 ETS    25%       6-7

OperationalFlowControl      : Priorities 0-4 Enabled
OperationalClassifications  : Not Available
RemoteTrafficClasses        : TC TSA    Bandwidth Priorities
                              -- ---    --------- ----------
                               0 ETS    25%       0-1
                               1 ETS    25%       2-3
                               2 ETS    25%       4-5
                               3 ETS    25%       6-7

RemoteFlowControl           : Priorities 0-4 Enabled
RemoteClassifications       : Not Available
```

In a scenario where both peers are set to Willing, the adapter with a lower MAC address takes the settings of the peer.

DCBX is disabled in the driver by default and in the some firmware versions as well.

➢ *To use DCBX:*

1. Query and enable DCBX in the firmware.
    a. Download the WinMFT package.
    b. Install WinMFT package and go to *\Program Files\Mellanox\WinMFT*
    c. Get the list of devices, run "mst status".



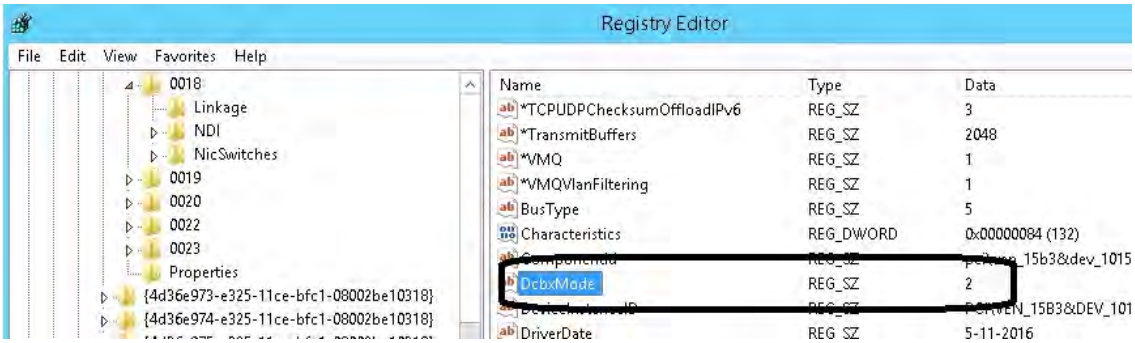
d. Verify is the DCBX is enabled or disabled, run *"mlxconfig.exe -d mt4117_pciconf0 query"* .

```
RATE_REDUCE_MONITOR_PERIOD_P2          4
INITIAL_ALPHA_VALUE_P2                 0
MIN_TIME_BETWEEN_CNPS_P2               0
CNP_DSCP_P2                            7
CNP_802P_PRIO_P2                       0
PORT_OWNER                             True(1)
ALLOW_RD_COUNTERS                      True(1)
IP_VER                                 IPv4(0)
NUM_OF_TC_P1                           8_TCS(0)
NUM_OF_VL_P1                           4_VLS(3)
NUM_OF_TC_P2                           8_TCS(0)
LLDP_NB_RX_MODE_P1                     2
LLDP_NB_TX_MODE_P1                     2
LLDP_NB_DCBX_P1                        True(1)
LLDP_NB_RX_MODE_P2                     0
LLDP_NB_TX_MODE_P2                     0
LLDP_NB_DCBX_P2                        True(1)
DCBX_CEE_P1                            True(1)
```

e. If disabled, run the following commands for a dual-port card.

```
mlxconfig -d mt4117_pciconf0 set LLDP_NB_RX_MODE_P1=2
mlxconfig -d mt4117_pciconf0 set LLDP_NB_TX_MODE_P1=2
mlxconfig -d mt4117_pciconf0 set LLDP_NB_DCBX_P1=1
mlxconfig -d mt4117_pciconf0 set LLDP_NB_RX_MODE_P2=2
mlxconfig -d mt4117_pciconf0 set LLDP_NB_TX_MODE_P2=2
mlxconfig -d mt4117_pciconf0 set LLDP_NB_DCBX_P2=1
```

2. Add the *"DcbxMode"* registry key, set the value to "2" and reload the adapter.
   The registry key should be added to *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<IndexValue>*
   To find the IndexValue, refer to section Finding the Index Value of the Network Interface.



Receive Path Activity Monitoring

In the event where the device or the Operating System unexpectedly becomes unresponsive for a long period of time, the Flow Control mechanism may send pause frames, which will cause congestion spreading to the entire network.

To prevent this scenario, the device monitors its status continuously, attempting to detect when the receive pipeline is stalled. When the device detects a stall for a period longer than a pre-configured timeout, the Flow Control mechanisms (Global Pause and PFC) are automatically disabled.

If the PFC is in use, and one or more priorities are stalled, the PFC will be disabled on all priorities. When the device detects that the stall has ceased, the flow control mechanism will resume with its previously configured behavior.

Head of Queue Lifetime Limit

This feature enables the system to drop the packets that have been awaiting transmission for a long period of time, preventing the system from hanging. The implementation of the feature complies with the Head of Queue Lifetime Limit (HLL) definition in the InfiniBand™ Architecture Specification.

The HLL has three registry keys for configuration:

TCHeadOfQueueLifeTimeLimit, TCStallCount and TCHeadOfQueueLifeTimeLimitEnable (see section Ethernet Registry Keys).

VXLAN

VXLAN technology provides scalability and security challenges solutions. It requires extension of the traditional stateless offloads to avoid performance drop. ConnectX®-4 and onwards adapter cards offer stateless offloads for a VXLAN packet, similar to the ones offered to non-encapsulated packets. VXLAN protocol encapsulates its packets using outer UDP header.

ConnectX®-4 and onwards support offloading of tasks related to VXLAN packet processing, such as TCP header checksum and VMQ (i.e.: directing incoming VXLAN packets to the appropriate VM queue).

VXLAN Offloading is a global configuration for the adapter. As such, on a dual-port adapter, any modification to one port will apply to the other port as well. Due to a hardware limitation, this will not be shown when querying different ports (e.g. if Port A is modified, this will show up when querying Port A but not Port B.). As such, it is recommended that any modification on one port be applied to the other port using Mlx5Cmd.

VXLAN can be configured using the standardized * *VxlanUDPPortNumber* and **EncapsulatedPacketTaskOffloadVxlan* keys.

Threaded DPC

A threaded DPC is a DPC that the system executes at IRQL = PASSIVE_LEVEL. An ordinary DPC preempts the execution of all threads, and cannot be preempted by a thread or by another DPC. If the system has a large number of ordinary DPCs queued, or if one of those DPCs runs for a long period time, every thread will remain paused for an arbitrarily long period of time. Thus, each ordinary DPC increases the system latency, which can damage the performance of time-sensitive applications, such as audio or video playback.

Conversely, a threaded DPC can be preempted by an ordinary DPC, but not by other threads. Therefore, the user should use threaded DPCs rather than ordinary DPCs, unless a particular DPC must not be preempted, even by another DPC.

For more information, please refer to Introduction to Threaded DPCs.

UDP Segmentation Offload (USO)

[Windows Client 10 18908 (20H1) and later] UDP Segmentation Offload (USO) enables network cards to offload the UDP datagrams' segmentation that are larger than the MTU on the network medium. It is enabled/disabled using standardized registry keys (UsoIPv4 & UsoIPv6) as described in Offload Registry Keys.

> ⚠ UDP Segmentation Offload (USO) is currently supported in ConnectX-4/ConnectX-4 Lx/ ConnectX-5 adapter cards only.

Hardware Timestamping

Hardware Timestamping is used to implement time-stamping functionality directly into the hardware of the Ethernet physical layer (PHY) using Precision Time Protocol (PTP). Time stamping is performed in the PTP stack when receiving packets from the Ethernet buffer queue.

This feature can be disabled, if desired, through a registry key. Registry key location:

```
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1- 08002be10318}\<nn>
```

For more information on how to find a device index nn, refer to to section Finding the Index Value of the Network Interface..

Key Name	Key Type	Values	Description
*PtpHardwareTimestamp	REG_DWORD	• 0 - Disabled • 1 - Enabled	Enables or disables the hardware timestamp feature.

> ⚠ Hardware Timestamping is supported in Windows Server 2019 and above.

Striding RQ

> ⚠ This feature is supported in Ethernet protocol and in ConnectX-5 and above adapter cards.

> ⚠ This feature is supported only when LRO is enabled. LRO minimum value is 16KB and maximum value is 64KB.

Receive buffers size is set to the maximum possible size of incoming messages. Every incoming message that is smaller than the maximum possible size, leaves a unutilized memory in order to increase the memory utilization. Receive buffers are segmented into fixed size strides and each incoming packet (or an LRO aggregate) consumes a buffer of its size (rather than the maximum possible incoming message size.)

Additional MAC Addresses for the Network Adapter

This feature allows the user to configure additional MAC addresses for the network adapter without setting the adapter to promiscuous mode. Registering MAC addresses for a network adapter will allow the adapter to accept packets with the registered MAC address.

> ⚠ This feature is supported in Ethernet protocol and native mode only.

Configuring Additional MAC Addresses:

The additional MAC addresses are configured using the mlx5cmd tool.

- **To view the adapter's current configuration:**
  `mlx5cmd -MacAddressList -Name <Adapter name> -Query`
- **To add additional MAC addresses (three in the example below):** `mlx5cmd -MacAddressList -Name <Adapter name> -Add -Entries 3 AB-CD-EF-AB-CD-E0 AB-CD-EF-AB-CD-E1 AB-CD-EF-AB-CD-E2`
- **To delete more than 1 MAC addresses (two in the example below):** `mlx5cmd -MacAddressList -Name "Ethernet" -Delete -Entries 2 AB-CD-EF-AC-CD-E1 AB-CD-EF-AC-CD-E2`

Explicit Congestion Notification (ECN) Hint in CQE

In a multi-host system, a single receive buffer is used for all hosts. If one or more host(s) are being congested, the congested host(s) can exhaust the device's receive buffer and cause service degradation for the other host(s). In order to manage this situation, the device can mark the ECN (Explicit Congestion Notification) bits in the IP header for the congested hosts. When ECN is enabled on the host, the host will sense the ECN marking and will reduce the TCP traffic and by that will throttle the traffic.

For the ECN related software counters refer to WinOF-2 Receive Datapath and WinOF-2 PCI Device Diagnostic.

> ⚠ The feature is supported only for lossy traffic, single port adapter cards, and TCP traffic.

Registry keys:

Name	Description	
CongestionMonitoringEnable	Driver will read CQE hint to mark ECN in the packet. Registry key is dynamic.	• 0 - Disabled (Default) • 1 - Enabled
CongestionAction	When overflow encountered by hardware for lossy traffic, packets will either be dropped or marked for driver to get hint in CQE. Values can be changed only when CongestionMonitoringEnable is set to 1. Registry key is dynamic.	• 0 - Disabled • 1 - Drop • 2 - Mark (default)

Name	Description	
CongestionMode	Programs hardware to be in aggressive mode where traffic is dropped/marked in an aggressive way, or in dynamic mode where the drop/mark in more relaxed. Values can be changed only when CongestionMonitoringEnable is set to 1. Registry key is dynamic.	• 0 – Aggressive • 1 - Dynamic (default)

NDIS Poll Mode

Windows introduced a new poll mode feature starting NDIS 6.85 onwards. The poll API handles Datapath processing for both TX and/or RX side. When the feature is enabled, the driver registers with NDIS for call backs to poll RX and/or TX data.

Enabling/Disabling NDIS Poll Mode

The registry keys used to enable/disable this capability are not dynamic. At this time, the registry keys are not exposed in the INF as the operating system is not GA as yet.

Registry Name	Value	Comments
RecvCompletionMethod	Set to 4 to register and use Ndis Poll Mode	Default is 1 (Adaptive)
SendCompletionMethod	Set to 2 to register and use Ndis Poll Mode	Default is 1 (Interrupt)

Limitations

When enabled on RX side, the following capabilities are not be supported:

- AsyncReceiveIndicate
- Receive side Threaded DPC
- Force low resource indication

When enabled on TX side, the following capabilities are not be supported:

- Transmit side Threaded DPC
- TxMaxPostSendsCoalescing is limited to 32

GPUDirect

Peer-to-Peer data transfers allows direct data transfer between PCIe devices without the need to use system main memory as a temporary storage or use of the CPU for moving data. When one peer is a NIC while the other peer is a GPU, it allows the NIC to have direct access to the GPU memory and transfer data through the network, bypassing the CPU and reducing memory copy operations.

This feature can be enabled or disabled by setting registry key below:

Key Name	Key Type	Values	Descriptions
EnableGpuDirect	REG_DWORD	[0, 1] Default: 0	This registry key enables or disables this feature. **Note**: Restart the network adapter after you change this registry key.

The registry key should be added to:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<IndexValue>
```

To find the IndexValue, refer to section Finding the Index Value of the Network Interface.

Feature Requirements

- RDMA support: To enable RDMA, please see section RDMA over Converged Ethernet (RoCE).
- DevX Interface support: To enable DevX Interface, please see section DevX Interface.
- NVIDIA Turing Architecture Quadro GPUs and later, e.g. Quadro RTX 4000.
- NVIDIA GPU driver support: R470 and later.

Feature Limitations

- The current NIC and GPU kernel driver implementations do not support memory region registration on GPU memory buffer larger than 4 GB.

Hardware QoS Offload

From NDIS 6.84+, Windows supports Hardware QoS Offload for the VMQoS capability. This feature allows moving egress bandwidth management entirely into the hardware. It allocates explicit Schedule Queues (SQs) on the physical NIC with bandwidth limits and guarantees reservations on a per-Traffic class basis.

To get full functionally of the feature, the users must set the "SendQueuePerPrio" key to 1 on the hypervisor and on the Virtual Machines.

⚠ Currently, the driver only supports enabling all TCs or none for an SQ.

⚠ For ConnectX-5 and above adapter cards, only TC0 traffic will be limited over the synthetic path.

This feature can be enabled or disabled by setting the registry key below:

Key Name	Key Type	Values	Descriptions
*QoSOffload	REG_SZ	[0, 1] Default: 1	This registry key enables or disables HwQoS. **Note:** Restart the network adapter after updating the registry key.

Key Name	Key Type	Values	Descriptions
*QosOffloadSupportedTCs	REG_SZ	[0, 0xFF] Default 0xFF	This registry key indicates which traffic classes HwQoS should be applied to. Each set bit indicates that traffic class should be supported. **Note:** Restart the network adapter after updating the registry key.
SendQueuePerPrio	REG_SZ	[0, 1] Default: 0	This key controls how the driver will open a single queue for each priority when the key is disbaled. 1 queue will be used for all prioritties. **Note:** Enabling this feature increases the alocated resources darmatically and therefore should be enbaled only when using Hardware QoS.

To check the status of this feature, run the following command:

```
mlx5cmd -features -name <adapter name>
```

Multi Prio Send Queue

The "MultiPrioSq" controls the SL-Diff feature in which the firmware modifies the priority (SL - Service Level) of the HW send-queue to match the one of the sent packet (QoS). When the SQ is flooded with packets of random priories, than the SL-Diff capability will be triggered rapidly impacting the SQ performance. The miniport is using a single SQ per NDIS TX-ring by default hence the SL-Diff is applicable.

Feature Configuration

This feature can be enabled or disabled by setting the registry key below:

Key Name	Key Type	Values	Descriptions
MultiPrioSq	REG_DWORD	• 0 - sldiff disabled • 1 - sldiff enabled (Default) • 2 - sldiff disabled if "SendQueuePerPrio"=0	Added a "MultiPrioSq" registry key to enable and disable the MultiPrioSq feature.

InfiniBand Network

General supported capabilities:

- Supported/Unsupported IPoIB Capabilities
- Default and non-default PKeys
- Teaming

Supported/Unsupported IPoIB Capabilities

Supported Capabilities	Unsupported Capabilities
• Port Management • Assigning Port IP After Installation • Modifying Driver's Configuration • Receive Side Scaling (RSS) • Displaying Adapter Related Information • Default and non-default PKeys	• VXLAN • NVGRE • Receive Side Coalescing (RSC) • VLAN • Multiple and non-default PKyes • Head of Queue Lifetime Limit

Default and non-default PKeys

Partition Keys (PKeys) are used to partition IPoIB communication by mapping a non-default full-membership PKey to index 0, and mapping the default PKey to an index other than zero. Driver's over-end-points communicate via the PKey is set in index 0. Their communication with the Subnet Agent is done via the default PKey that is not necessarily set in index 0. To enable such behavior, the PKey in index 0 must be in full state.

PKey is a four-digit hexadecimal number specifying the InfiniBand partition key. It can be specified by the user when a non-default PKey is used.

The default PKey (0x7fff) is inserted in block 0 index 0, by default. PKey's valid values are 0x1 - 0x7fff.

System Requirements	
Firmware version:	14/16.23.1020 and higher

The feature is firmware dependent. If an earlier firmware version is used, traffic may fail as the feature is unsupported and the following event will be displayed in the Event Viewer:

Event ID: 0x0034:

Event message: <Adapter name>: Non-default PKey is not supported by FW.

PKey Membership Types

The following are the available PKey's membership types:

- Full (default): Members with full membership may communicate with all hosts (members) within the network/partition.
- Limited/partial: Members with limited membership cannot communicate with other members with limited membership. However, they can communicate between every other combination of membership types (e.g., full + limited, limited + full).

Changing PKey membership	
Setting a full membership	`ib partition <partition name> member all type full`
Setting a limited membership	`ib partition <partition name> member all type limited`
Changing PKey membership using UFM	`ib partition management defmember <limited/full>`

Changing the PKey Index

PKey index can be changed using one of the following methods:

- Subnet Manager (SM) in the Switch
    a. Obtain the partition.conf file.
    - In the switch, the file is located at: *vtmp/infiniband-default/var/opensm/partitions.conf*
    - In the Linux host, the file is located at: */etc/opensm/partitions.conf*
    b. Add ",indx0" for a non-default PKey. If if already exists the default PKey, remove it. For example: *non-deafult=0x3,ipoib: ALL=full; --> non-deafult=0x3,indx0,ipoib: ALL=full;*
    c. Load/save the partition.conf file.
- UFM
  Add a new full membership PKey with indx0. The newly added PKey will replace the default PKey.

Creating, Deleting or Configuring PKey

PKey can be created, deleted or configured using one of the following methods:

- **Subnet Manager (SM)** in the Switch
  Note: To perform any of the actions below, you need to access the switch's configuration.
    - To create a new PKey, run:

      ```
      ib partition <partition name> pkey <pkey number>
      ```

    - To delete a PKey, run:

      ```
      no ib partition <partition name>
      ```

    - To configure the PKey to be IPoIB, run:

      ```
      ib partition <partition name> ipoib force
      ```

- UFM
  PKey can be created, deleted or configured using UFM by adding an extension to the partitions.conf file that is generated by the UFM. The new extension can be added by editing the /opt/ufm/files/conf/partitions.conf.user_ext file according to the desired action (create/delete/configure). The content of this extension file is added to the partitions.conf

file upon file synchronization done by the UFM on every logical model change. Synchronization can also be triggered manually by running the /opt/ufm/scripts/sync_partitions_conf.sh script. The script merges the /opt/ufm/files/conf/partitions.conf.user_ext file into the /opt/ufm/files/conf/opensm/partitions.conf file and starts the heavy sweep on the SM.

Teaming

NIC Teaming allows you to group between one and 32 physical Ethernet/IPoIB network adapters into one or more software-based virtual network adapters. These virtual network adapters provide fast performance and fault tolerance in the event of a network adapter failure.
On Windows Server edition, there is a built-in module that supports teaming and VLAN Ethernet. For more information see here.

One of the existing limitations with Windows OS support is that it does not support NIC teaming solution for IPoIB devices.

> ⚠ A team can either have only IPoIB members or only Ethernet members.

To overcome these limitations, WinOF-2 driver provides a teaming solution. The supported modes are:

- Active-Standby mode - In this mode, the user can pick the primary adapter responsible for sending traffic. In the event of a link failure, a failover happens and the standby adapter takes over. User can also tell us to not failback to primary in case of fail-over followed by primary adapter has link up. For this mode, set the failover mode in the mlx5MuxTool.

Please refer to content below on how to configure using custom teaming solution with RSS functionality.

Prerequisites

> ⚠ IPoIB teaming solution is supported ONLY when using the WinOF-2 mlx5mux driver.

- Operating Systems: Windows 10 and above

- Adapter Cards: All supported devices supported by WinOF-2 driver.
- For using the Mlx5muxtool, users that do not install the full package (HKEY_LOCAL_MACHINE\SOFTWARE\Mellanox\MLNX_WinOF2\InstalledPath) must point this key (InstalledPath) to the location of the mux drivers as the tool searches for the mux drivers files in a folder called "mux" in the folder that define by this key(InstalledPath).

Configuring Command Line Based Teaming

1. Show the help menu. The following command prints out all supported modes and functionalities:

```
mlx5muxtool.exe --help
[TEAMING]
To list all adapters including teams, use:
       mlx5muxtool showlist
To create a team use:
       mlx5muxtool create team <Type> <Name> [NoFailBackToPrimary] [IPoIB]
          Type is one of the following: Aggregate | Failover
          For IPoIB team, only type 'Failover' is supported

To add adapter to the team use:
       mlx5muxtool attach team <TeamName> {<Adapter-GUID>} [primary] [SetTeamMacAddress]
To remove an adapter from the team use:
       mlx5muxtool detach team <TeamName> {<Adapter-GUID>}
To delete a team use:
       mlx5muxtool removeteam <TeamName>
To query an existing team, use:
       mlx5muxtool queryteam <TeamName>
```

Example:

```
mlx5muxtool create team Aggregate MyTeam
mlx5muxtool attach team MyTeam {2E9C1992-98B5-43C3-97A0-9993AEAC7F80}
mlx5muxtool attach team MyTeam {8D05C52B-BCD6-4FCE-8235-1E90BD334519}
```

2. Show all the adapter cards (including all created teams already).

```
mlx5muxtool.exe showlist
{90F5F52D-4384-4263-BD12-4588CA5CE80A}          Mellanox ConnectX-5 Adapter #2 (IPoIB)
{62B9661A-17C4-4AF3-AAA1-2B3337FD02E0}          Mellanox ConnectX-5 Adapter (IPoIB)
{136A1E6F-1168-48D4-B9CC-55EE563D427B}          Mellanox ConnectX-6 Adapter (IPoIB)
{87B55F92-D573-471B-882C-379773296A6D}          Mellanox ConnectX-6 Adapter #2 (IPoIB)
```

3. Create an empty IPoIB team.

```
mlx5muxtool.exe create team failover MyTeam IPoIB
Adding team MyTeam
Team created {FED1925F-F88F-4970-B4C3-38AA030874DF}
```

4. Attach members to the team.

```
mlx5muxtool.exe attach team MyTeam {90F5F52D-4384-4263-BD12-4588CA5CE80A} primary
Attaching adapter {90F5F52D-4384-4263-BD12-4588CA5CE80A} to team MyTeam
```

5. Query the team.

```
mlx5muxtool.exe queryteam MyTeam

Found 1 team(s)

Name               : MyTeam
GUID               : {FED1925F-F88F-4970-B4C3-38AA030874DF}
PortType           : IPoIB
TeamType           : Failover
MemberCount        : 2
Member[0]          : {62B9661A-17C4-4AF3-AAA1-2B3337FD02E0} (SLOT 5 Port 2)
Member[1]          : {90F5F52D-4384-4263-BD12-4588CA5CE80A} (Primary) (SLOT 5 Port 1)
```

6. Detach members from the team.

```
mlx5muxtool.exe detach team MyTeam {62B9661A-17C4-4AF3-AAA1-2B3337FD02E0}
Dettaching adapter {62B9661A-17C4-4AF3-AAA1-2B3337FD02E0} from team MyTeam
```

7. Remove an entire team.

```
mlx5muxtool.exe removeteam MyTeam
Delete team MyTeam
Deleting member {90F5F52D-4384-4263-BD12-4588CA5CE80A}
```

Storage Protocols

Deploying SMB Direct

The Server Message Block (SMB) protocol is a network file sharing protocol implemented in Microsoft Windows. The set of message packets that defines a particular version of the protocol is called a dialect.

The Microsoft SMB protocol is a client-server implementation and consists of a set of data packets, each containing a request sent by the client or a response sent by the server.

SMB protocol is used on top of the TCP/IP protocol or other network protocols. Using the SMB protocol allows applications to access files or other resources on a remote server, to read, create, and update them. In addition, it enables communication with any server program that is set up to receive an SMB client request.

SMB Configuration Verification

Verifying Network Adapter Configuration

Use the following PowerShell cmdlets to verify Network Direct is globally enabled and that you have NICs with the RDMA capability. The command must be ran on both the SMB server and the SMB client.

```
PS $ Get-NetOffloadGlobalSetting | Select NetworkDirect
PS $ Get-NetAdapterRDMA
PS $ Get-NetAdapterHardwareInfo
```

Verifying SMB Configuration

Use the following PowerShell cmdlets to verify SMB Multichannel is enabled, confirm the adapters are recognized by SMB and that their RDMA capability is properly identified.

- On the SMB client, run the following PowerShell cmdlets:

  ```
  PS $ Get-SmbClientConfiguration | Select EnableMultichannel
  PS $ Get-SmbClientNetworkInterface
  ```

- On the SMB server, run the following PowerShell cmdlets:
  Note: The NETSTAT command confirms if the File Server is listening on the RDMA interfaces.

  ```
  PS $ Get-SmbServerConfiguration | Select EnableMultichannel
  PS $ Get-SmbServerNetworkInterface
  PS $ netstat.exe -xan | ? {$_ -match "445"}
  ```

Verifying SMB Connection

To verify the SMB connection on the SMB client:

1. Copy the large file to create a new session with the SMB Server.
2. Open a PowerShell window while the copy is ongoing.
3. Verify the SMB Direct is working properly and that the correct SMB dialect is used.

```
PS $ Get-SmbConnection
PS $ Get-SmbMultichannelConnection
PS $ netstat.exe -xan | ? {$_ -match "445"}
```

> If you have no activity while you run the commands above, you might get an empty list due to session expiration and absence current connections.

Verifying SMB Events that Confirm RDMA Connection

To confirm RDMA connection, verify the SMB events:

1. Open a PowerShell window on the SMB client.
2. Run the following cmdlets.
   Note: Any RDMA-related connection errors will be displayed as well.

```
PS $ Get-WinEvent -LogName Microsoft-Windows-SMBClient/Operational | ? Message -match "RDMA"
```

Virtualization

Hyper-V with VMQ

System Requirements	
Operating Systems:	Windows Server 2012 and above

Using Hyper-V with VMQ

NVIDIA® WinOF-2 driver includes a Virtual Machine Queue (VMQ) interface to support Microsoft Hyper-V network performance improvements and security enhancement.

VMQ interface supports:

- Classification of received packets by using the destination MAC address to route the packets to different receive queues
- NIC ability to use DMA to transfer packets directly to a Hyper-V child-partition shared memory
- Scaling to multiple processors, by processing packets for different virtual machines on different processors.

To enable Hyper-V with VMQ using UI:

1. Open Hyper-V Manager.
2. Right-click the desired Virtual Machine (VM), and left-click Settings in the pop-up menu.
3. In the Settings window, under the relevant network adapter, select “Hardware Acceleration”.
4. Check/uncheck the box “Enable virtual machine queue” to enable/disable VMQ on that specific network adapter.

To enable Hyper-V with VMQ using PowerShell:

1. Enable VMQ on a specific VM: Set-VMNetworkAdapter <VM Name> -VmqWeight 100
2. Disable VMQ on a specific VM: Set-VMNetworkAdapter <VM Name> -VmqWeight 0

Network Virtualization using Generic Routing Encapsulation (NVGRE)

VF CPU MonitorNetwork Virtualization using Generic Routing Encapsulation (NVGRE) offload is currently supported in Windows Server 2012 R2 with the latest updates for Microsoft.

For further information, please refer to the Microsoft’s “Network Virtualization using Generic Routing Encapsulation (NVGRE) Task Offload” document.

Enabling/Disabling NVGRE Offloading

To leverage NVGRE to virtualize heavy network IO workloads, the NVIDIA® ConnectX®-4 network NIC provides hardware support for GRE offload within the network NICs by default.

To enable/disable NVGRE offloading:

1. Open the Device Manager.
2. Go to the Network adapters.
3. Right click ‘Properties’ on Mellanox ConnectX®-4 Ethernet Adapter card.
4. Go to Advanced tab.
5. Choose the ‘Encapsulate Task Offload’ option.
6. Set one of the following values:
    a. Enable - GRE offloading is Enabled by default
    b. Disabled - When disabled the Hyper-V host will still be able to transfer NVGRE traffic, but TCP and inner IP checksums will be calculated by software that significantly reduces performance.

If both the NVGRE and VXLAN are disabled, the driver configures the inner rule to be ready in case encapsulation will be enabled (e.g., OID_RECEIVE_FILTER_SET_FILTER with flag NDIS_RECEIVE_FILTER_PACKET_ENCAPSULATION is received). That means while working without non-tunneled rule, no traffic will match this filter until encapsulation is enabled.

When using NVGRE and VXLAN when in NIC mode in NVIDIA BlueField-2 DPU, please note known issues 3040551 in the Release Notes.

Configuring NVGRE using PowerShell

Hyper-V Network Virtualization policies can be centrally configured using PowerShell 3.0 and PowerShell Remoting.

For further information of now to configure NVGRE using PowerShell, please refer to Microsoft's "Step-by-Step: Hyper-V Network Virtualization" blog.

Once the configuration using PowerShell is completed, verifying that packets are indeed encapsulated as configured is possible through any packet capturing utility. If configured correctly, an encapsulated packet should appear as a packet consisting of the following headers:

- Outer ETH Header
- Outer IP
- GRE Header
- Inner ETH Header
- Original Ethernet Payload

Single Root I/O Virtualization (SR-IOV)

Single Root I/O Virtualization (SR-IOV) is a technology that allows a physical PCIe device to present itself multiple times through the PCIe bus. This technology enables multiple virtual instances of the device with separate resources. NVIDIA® adapters are capable of exposing up to 127 virtual instances called Virtual Functions (VFs) per port. These virtual functions can then be provisioned separately. Each VF can be seen as an addition device connected to the Physical Function. It also shares resources with the Physical Function.

SR-IOV is commonly used in conjunction with an SR-IOV enabled hypervisor to provide virtual machines direct hardware access to network resources hence increasing its performance.

This guide demonstrates the setup and configuration of SR-IOV, using NVIDIA® adapter cards family. SR-IOV VF is a single port device.

SR-IOV over Hyper-V

System Requirements	
Server and BIOS	A server and BIOS with SR-IOV support. **Note:** BIOS settings may require an update to enable virtualization support and SR-IOV support.
Hypervisor OS:	• Ethernet: Windows Server 2012 R2 and above • IPoIB: Windows Server 2016 and above
Virtual Machine (VM) OS:	Windows Server 2012 and above
Adapter cards	NVIDIA® ConnectX®-4 onward adapter cards
SR-IOV supported driver version:	• SR-IOV Ethernet over Hyper-V: WinOF-2 v1.20 or higher • SR-IOV IPoIB over Hyper-V and the guest: WinOF-2 v1.80 or higher and on Windows Server 2016

Feature Limitations

- SR-IOV in IPoIB node:
  - LID based IPoIB is supported with the following limitations:
    - It does not support routers in the fabric
    - It supports up to 2^15-1 LIDs
- No synthetic path: The SR-IOV path that goes thru the WinOF-2 driver
  Although both the NVIDIA® adapter - Virtual Function (VF) and the NetVSC will be presented in the VM, it is recommended to use only the NVIDIA® interface.

Configuring SR-IOV Host Machines

The sections below describe the required flows for configuring the host machines:

Enabling SR-IOV in BIOS

Depending on your system, perform the steps below to set up your BIOS. The figures used in this section are for illustration purposes only.

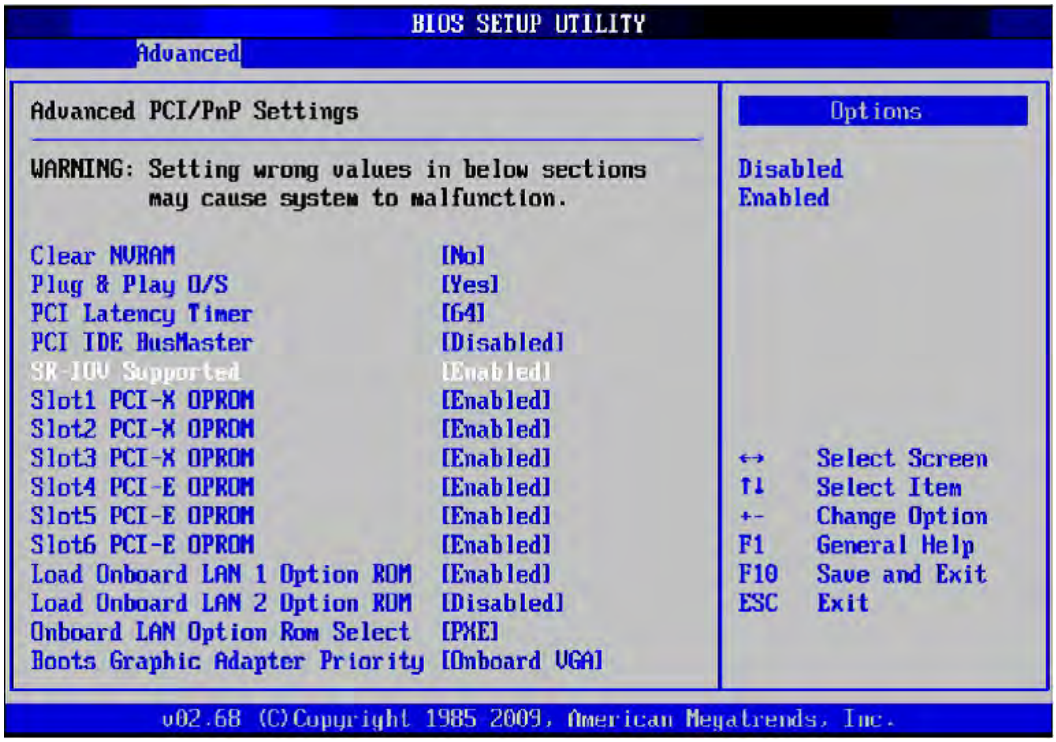
For further information, please refer to the appropriate BIOS User Manual.

> ⚠ It is recommended to enable the “above 4G decoding” BIOS setting for features that require large amount of PCIe resources.
>
> Such features are: SR-IOV with numerous VFs, PCIe Emulated Switch, and Large BAR Requests.

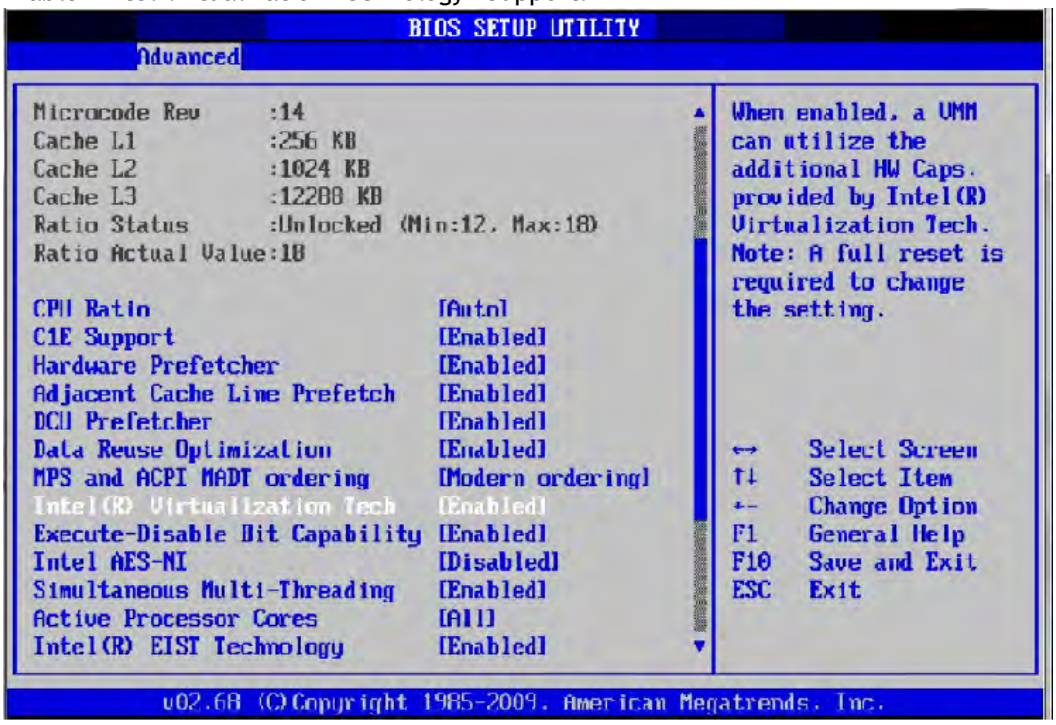
To enable SR-IOV in BIOS:

1. Make sure the machine’s BIOS supports SR-IOV.
   Please, consult BIOS vendor website for SR-IOV supported BIOS versions list. Update the BIOS version if necessary.
2. Enable SR-IOV according to the BIOS vendor guidelines.
   For example:

a. Enable SR-IOV.



b. Enable “Intel Virtualization Technology” Support.



For further details, please refer to the vendor's website.

Installing Hypervisor Operating System

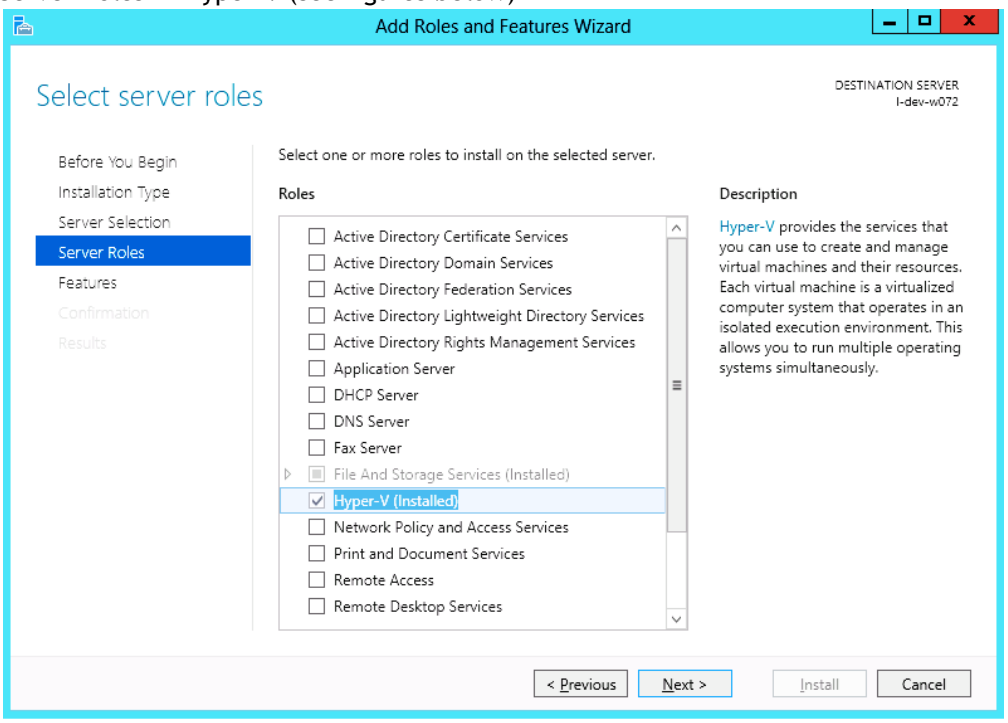
➢ *To install Hypervisor Operating System:*

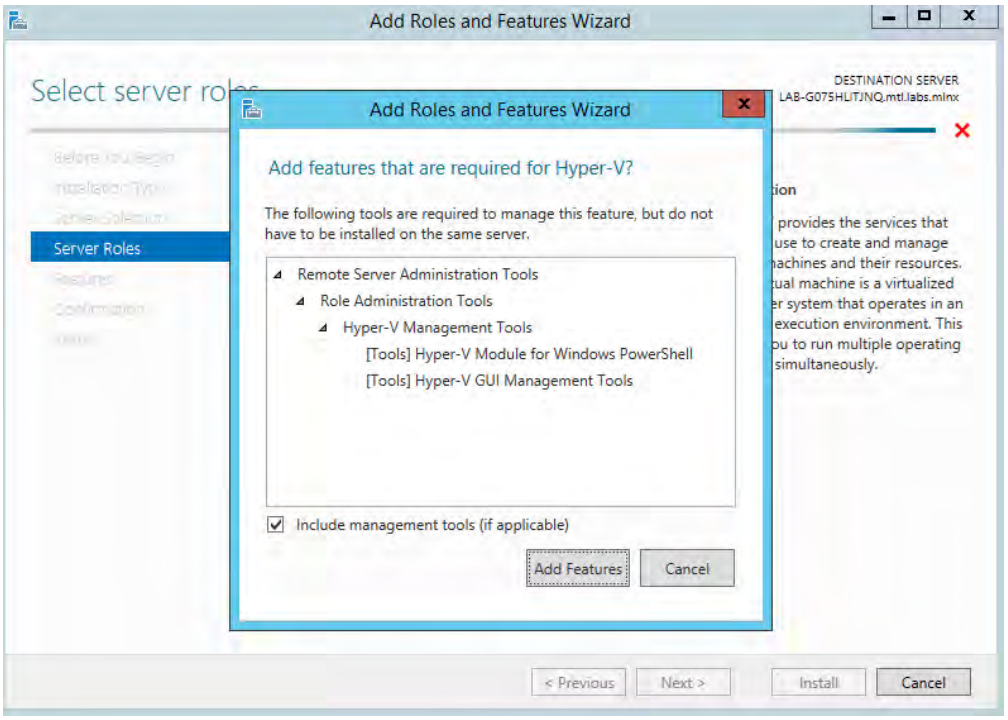
1. Install Windows Server 2012 R2
2. Install Hyper-V role:
    - Go to: Server Manager -> Manage -> Add Roles and Features and set the following:

    a. Installation Type -> Role-based or Feature-based Installation

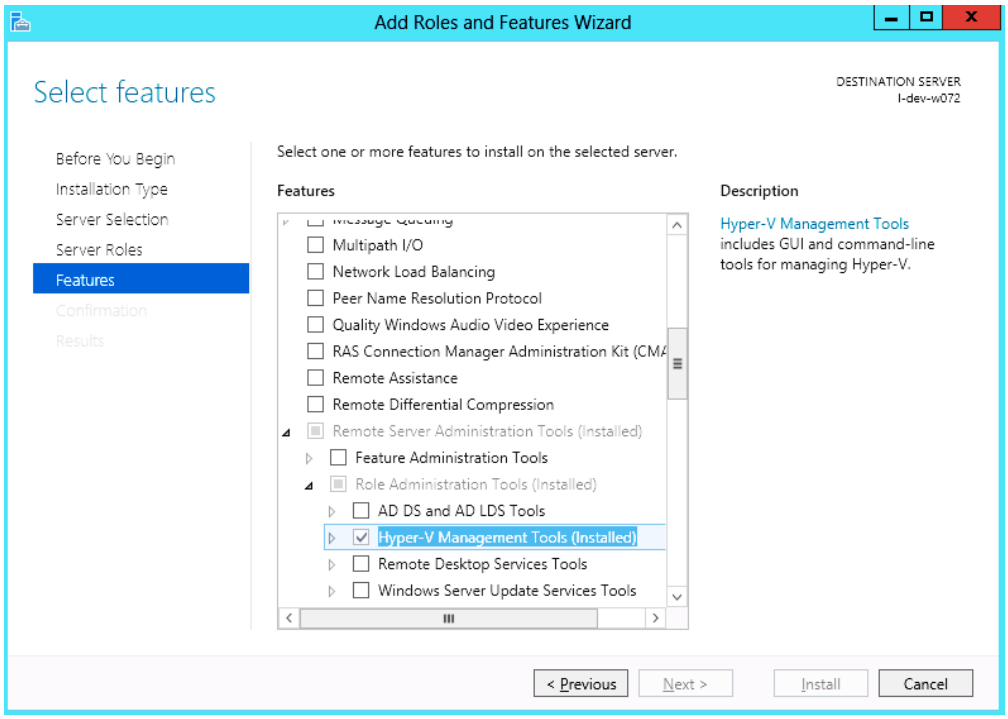
    b. Server Selection -> Select a server from the server pool

    c. Server Roles -> Hyper-V (see figures below)

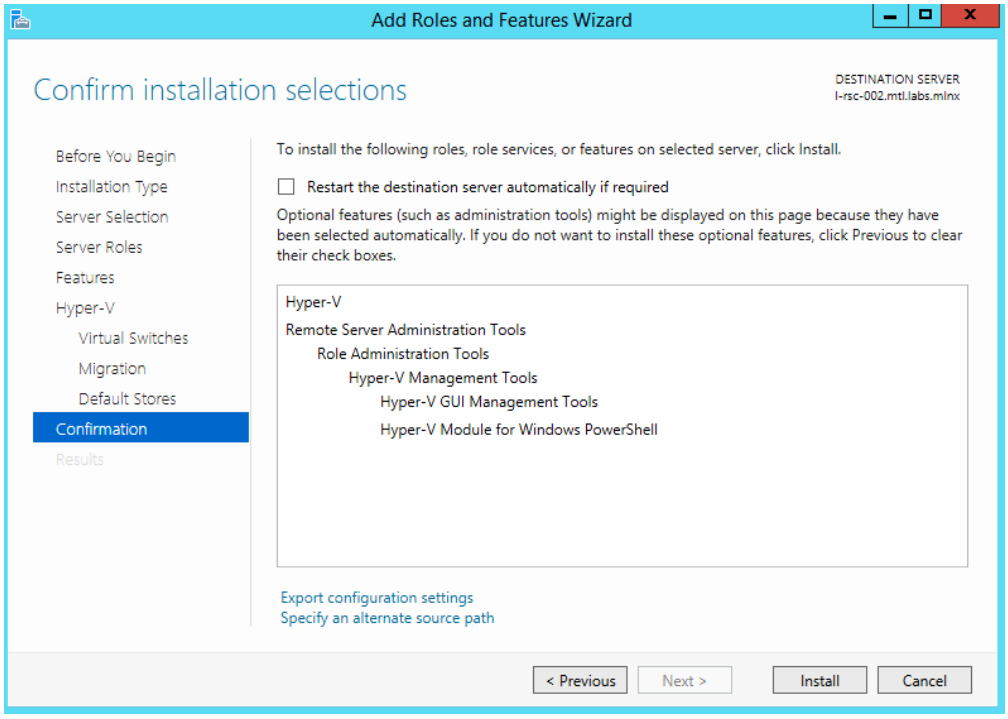




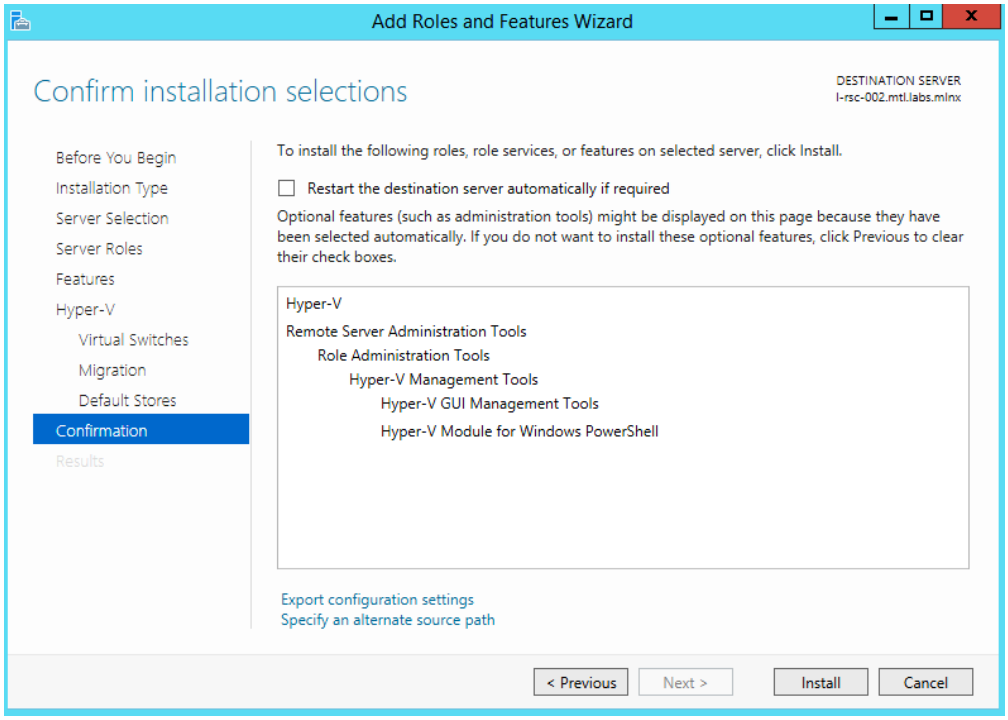
3. Install Hyper-V Management Tools.
   Features - > Remote Server Administration Tools -> Role Administration Tools ->Hyper-V Administration Tool.



4. Confirm installation selection.



5. Click Install.



6. Reboot the system.

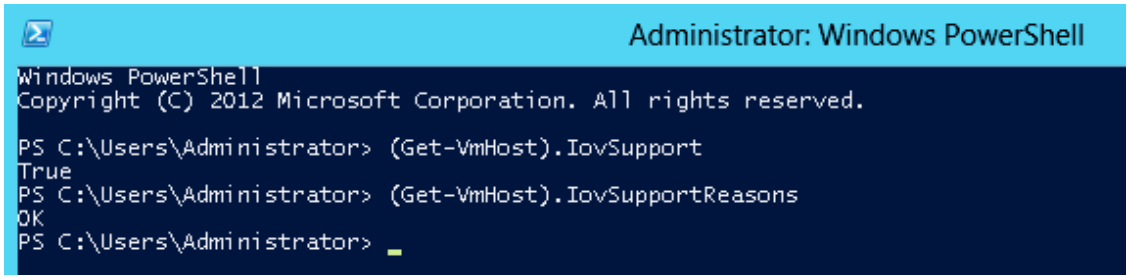
Verifying SR-IOV Support within the Host Operating System

To verify that the system is properly configured for SR-IOV:

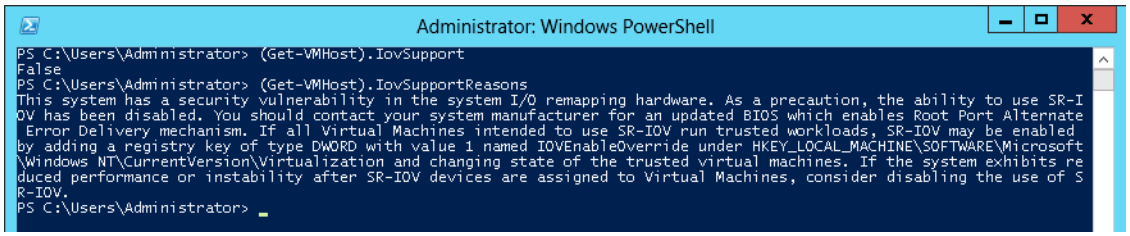
1. Go to: Start-> Windows Powershell.
2. Run the following PowerShell commands.

```
PS $ (Get-VmHost).IovSupport
PS $ (Get-VmHost).IovSupportReasons
```

In case that SR-IOV is supported by the OS, the output in the PowerShell is as in the figure below.



> ⚠ If the BIOS was updated according to BIOS vendor instructions and you see the message displayed in the figure below, update the registry configuration as described in the (Get-VmHost).IovSupportReasons message.



3. Reboot
4. Verify the system is configured correctly for SR-IOV as described in Steps 1/2.

Verifying Sufficient Resources are Available in the Adapter to Enable SR-IOV VFs

To verify resources sufficiency in the adapter to enable SR-IOV VFs:

1. Go to: Start-> Windows Powershell.
2. Run the following PowerShell commands.

```
PS C:\Windows\system32> Get-NetAdapterSriov
```

Example:

```
Name : SLOT 4 Port 1
InterfaceDescription : Mellanox ConnectX-4 Adapter
Enabled : True
SriovSupport : NoVfBarSpace
SwitchName : "Default Switch"
NumVFs : 32
```

> ⚠ If the *"SriovSupport"* field value shows *"NoVfBarSpace"* , SR-IOV cannot be used on this network adapter as there are not enough PCI Express BAR resources available. To use SR-IOV, you need to reduce the number of VFs to the number supported by the OS.

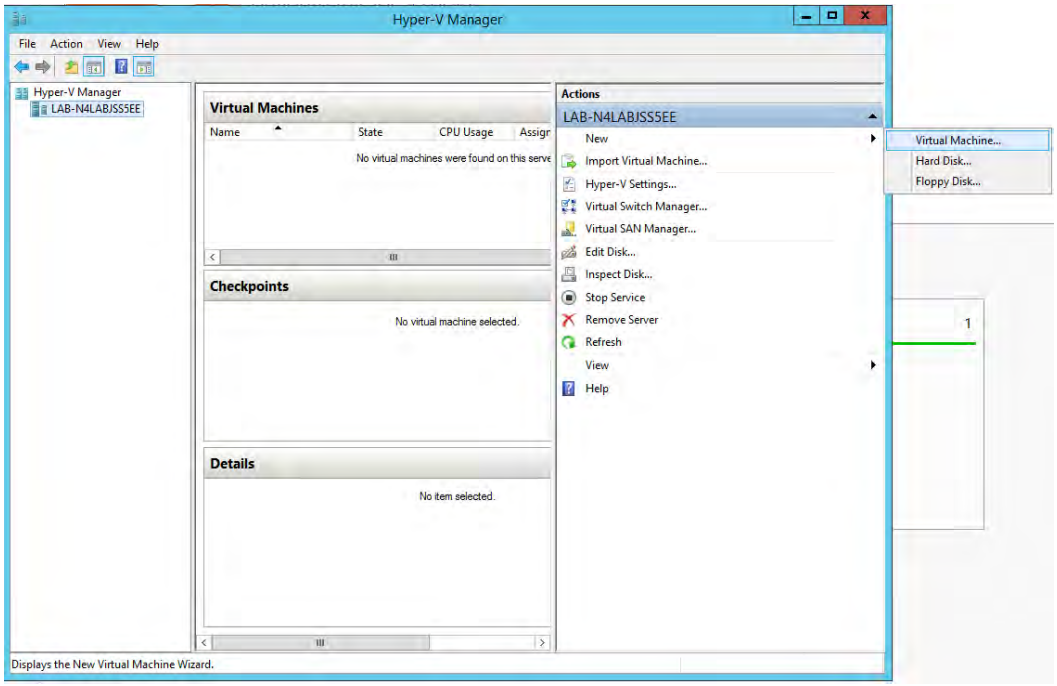
For further information, see https://technet.microsoft.com/en-us/library/jj130915(v=wps.630).aspx

Creating a Virtual Machine

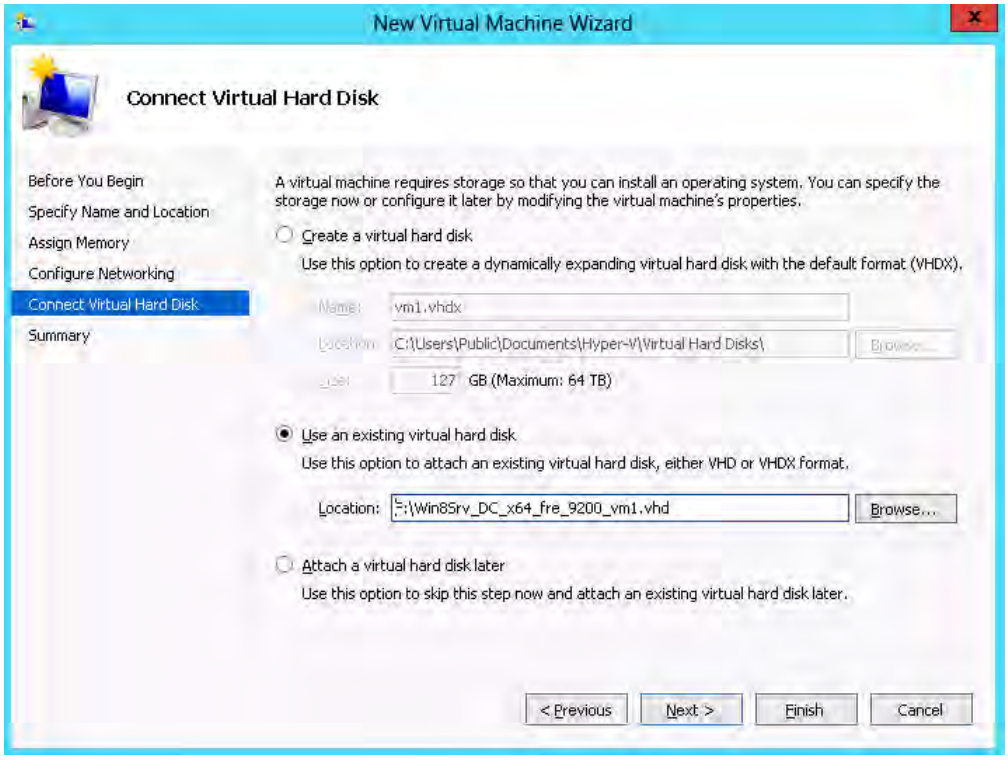
To create a Virtual Machine:

1. Go to: Server Manager -> Tools -> Hyper-V Manager.
2. Go to: New->Virtual Machine and set the following:
   - Name: <name>
   - Startup memory: 4096 MB

- Connection: Not Connected



3. Connect the virtual hard disk in the New Virtual Machine Wizard.
4. Go to: Connect Virtual Hard Disk -> Use an existing virtual hard disk.
5. Select the location of the VHD file.



Configuring Host Memory Limit per VF

In SR-IOV mode, the host allocates memory resources per the adapter`s needs for each VF. It is important to limit the amount of memory that the VF can receive from the host, in order to ensure the host's stability. To prevent excessive allocation, the MaxFWPagesUsagePerVF registry key must be configured to the maximum number of 4KB pages that the host could allocate for VFs resources. In case of attempting to use more pages then configured, an error will be printed to the system event log. For more information, see See SR-IOV Options.

Configuring NVIDIA® Network Adapter for SR-IOV

The sections below describe the required flows for configuring the NVIDIA® Network Adapter for SR-IOV:

Enabling SR-IOV in Firmware

> ⚠ For non-NVIDIA® (OEM) branded cards you may need to download and install the new firmware.

To enable SR-IOV using mlxconfig:

> ⚠ mlxconfig is part of MFT tools used to simplify firmware configuration. The tool is available using MFT v3.6.0 or higher.

1. Download MFT for Windows.
2. Get the device ID (look for the "_pciconf" string in the output).

   ```
   mst status
   ```

   Example:

   ```
   MST devices:
   ------------
   mt4115_pciconf0
   ```

3. Check the current SR-IOV configuration.

   ```
   mlxconfig -d mt4115_pciconf0 q
   ```

   Example:

   ```
   Device #1:
   ----------

   Device type: ConnectX4
   PCI device: mt4115_pciconf0
   Configurations: Current
   SRIOV_EN N/A
   NUM_OF_VFS N/A
   ```

```
WOL_MAGIC_EN_P2 N/A
LINK_TYPE_P1 N/A
LINK_TYPE_P2 N/A
```

4. Enable SR-IOV with 16 VFs.

```
mlxconfig -d mt4115_pciconf0 s SRIOV_EN=1 NUM_OF_VFS=16
```

> ⚠ All servers are guaranteed to support 16 VFs. Increasing the number of VFs can lead to exceeding the BIOS limit of MMIO available address space.

> ⚠ OS limits the maximum number of VFs to 32 per Network Adapter.
> To increase the number of VFs, the following PowerShell command should be used:
> *Set-NetAdapterSRIOV - name <AdapterName> -NumVFs <Required number of VFs>*

Example:

```
Device #1:
----------

Device type: ConnectX4
PCI device: mt4115_pciconf0

Configurations: Current New
SRIOV_EN N/A 1
NUM_OF_VFS N/A 16
WOL_MAGIC_EN_P2 N/A N/A
LINK_TYPE_P1 N/A N/A
LINK_TYPE_P2 N/A N/A

Apply new Configuration? ? (y/n) [n] : y
Applying... Done!
-I- Please reboot machine to load new configurations.
```

Configuring IPoIB in SR-IOV

Subnet Manager (SM) Configuration

The SM should be up in the fabric in order to work with IPoIB, and can be run on a switch or on a Linux host.

- Switch SM Configuration

1. Install the SM image that supports virtualization (3.6.4640 version and above). For more details, please refer to the switch operating system User Manual.
2. Enter the config mode.

```
switch > enable
switch # config terminal
```

3. Enable the SM (to disable the SM, type: no ib sm).

```
ib sm
```

4. Enable virtualization.

```
ib sm virt enable
```

5. Save the configuration.

```
configuration write
```

6. Restart the switch.

```
reload
```

7. Validate the Subnet Manager is enabled.

```
show ib sm
```

8. Validate Virtualization is enabled.

```
show ib sm virt
```

For more details, please refer to the Subnet Manager (SM) section in the MLNX-OS® User Manual for VPI.

- Linux Host SM Configuration

1. Enable the virtualization by setting the virt_enable field to 2 on the /etc/opensm/opensm.conf file.
2. Start OpenSM and bind it to a specific port.

```
opensm -e -B -g <Port GUID>
```

OpenSM may be bound to one port at a time. If the given GUID is 0, the OpenSM displays a list of possible port GUIDs and awaits user input. Without “-g”, the OpenSM attempts to use the default port.

Firmware Configuration

1. Get the device name.

```
mst status
```

2. Show device configurations.

```
mlxconfig -d <device name> q
```

3. Enable SR-IOV: (1 = Enable).

```
mlxconfig -d <device name> set SRIOV_EN=1
```

4. Set max VFs count.

```
mlxconfig -d <device name> set NUM_OF_VFS=<Count>
```

5. Configure device to work in IB mode (1=IB).

```
mlxconfig -d <device name> set LINK_TYPE_P1=1 set LINK_TYPE_P2=1
```

6. Enable LID based IPoIB.

```
mlxconfig -d <Device name> set SRIOV_IB_ROUTING_MODE_P1=1
mlxconfig -d <Device name> set SRIOV_IB_ROUTING_MODE_P2=1
```

7. Restart the firmware.

```
mlxfwreset -d <Device name> r --yes
```

> ⚠ The mlxconfig and mlxfwreset tools are a part of the WinMFT package. For more details, please refer to the MFT User Manual.

> ⚠ To enable IPoIB LID base by mlxconfig, install MFT v4.8.0-25, and above.

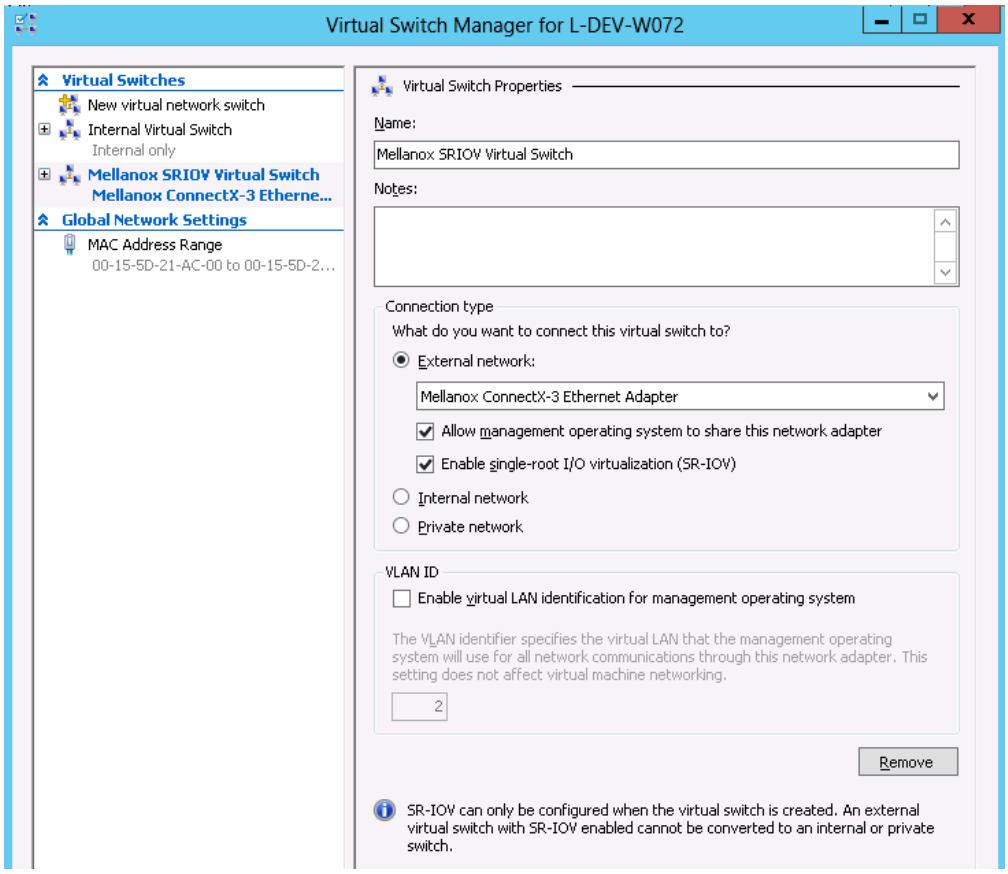
Configuring Virtual Machine Networking (InfiniBand SR-IOV Only)

For further details on enabling/configuring SR-IOV on KVM, please refer to section *"Single Root IO Virtualization (SR-IOV)"* in MLNX_OFED for Linux User Manual.

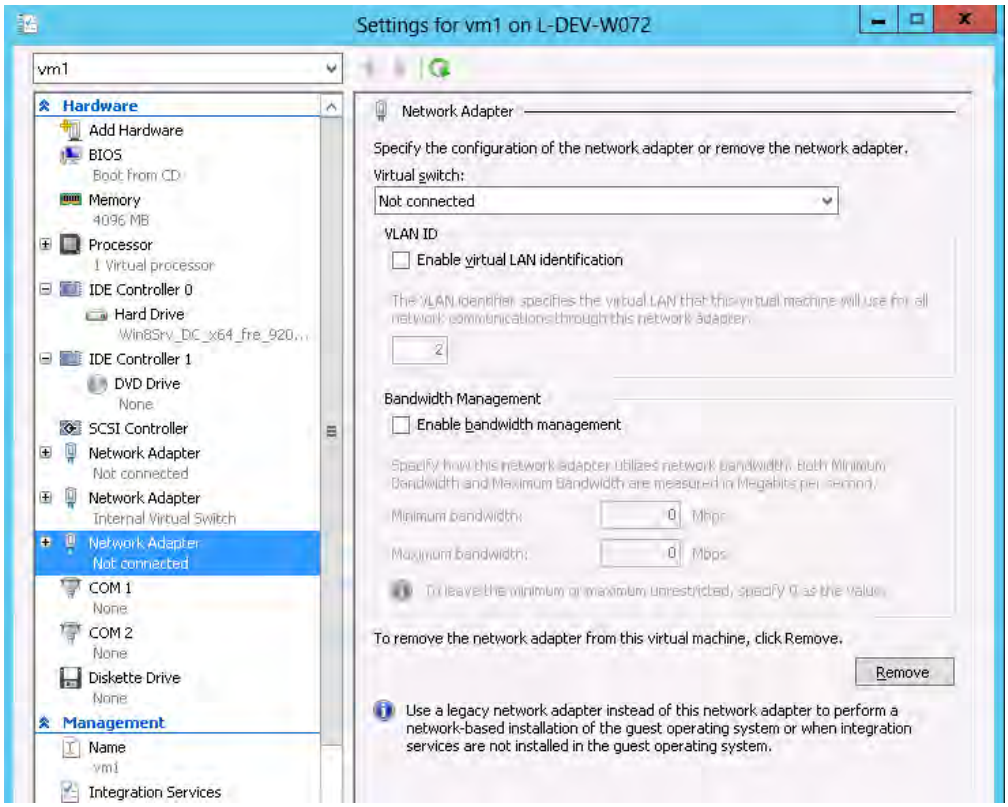
Configuring Virtual Machine Networking

To configure Virtual Machine networking:

1. Create an SR-IOV-enabled Virtual Switch over NVIDIA® Ethernet Adapter.
   - Go to: Start -> Server Manager -> Tools -> Hyper-V Manager
   - Hyper-V Manager: Actions -> Virtual SwitchManager -> External-> Create Virtual Switch
2. Set the following:
   - Name:
   - External network:
   - Enable single-root I/O virtualization (SR-IOV)

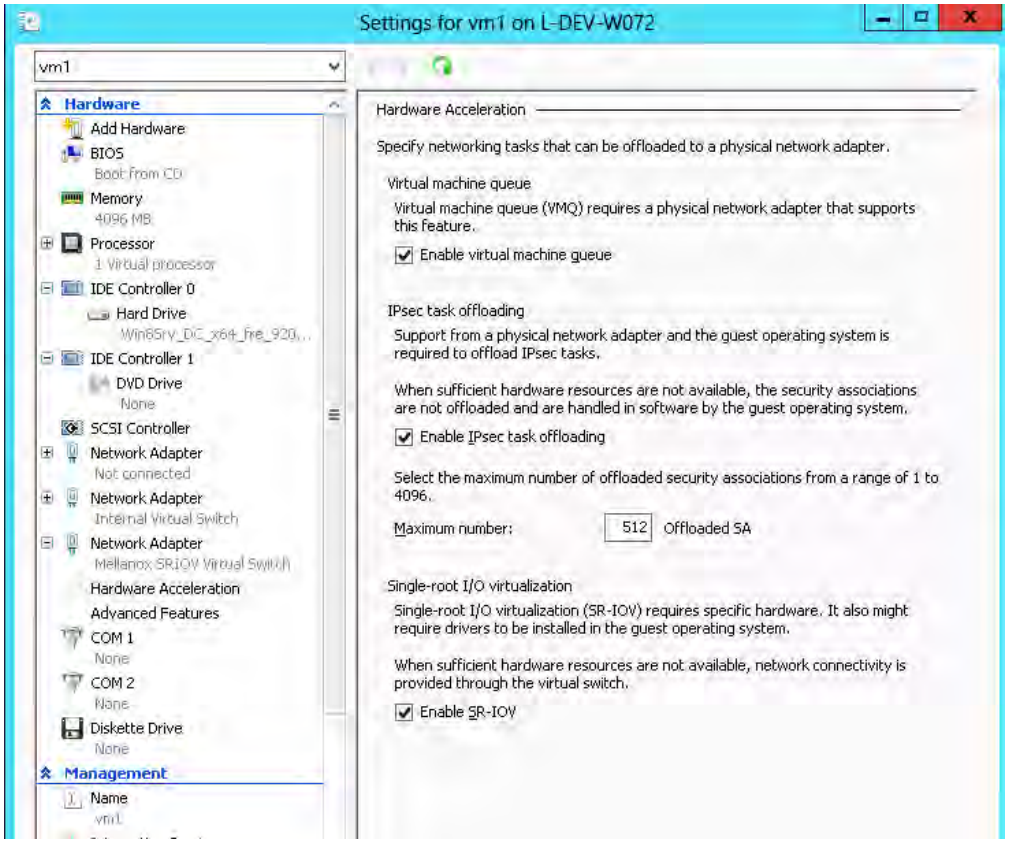


3. Click Apply.
4. Click OK.
5. Add a VMNIC connected to a NVIDIA® vSwitch in the VM hardware settings:
   - Under Actions, go to Settings -> Add New Hardware-> Network Adapter-> OK
   - In “Virtual Switch” dropdown box, choose Mellanox SR-IOV Virtual Switch

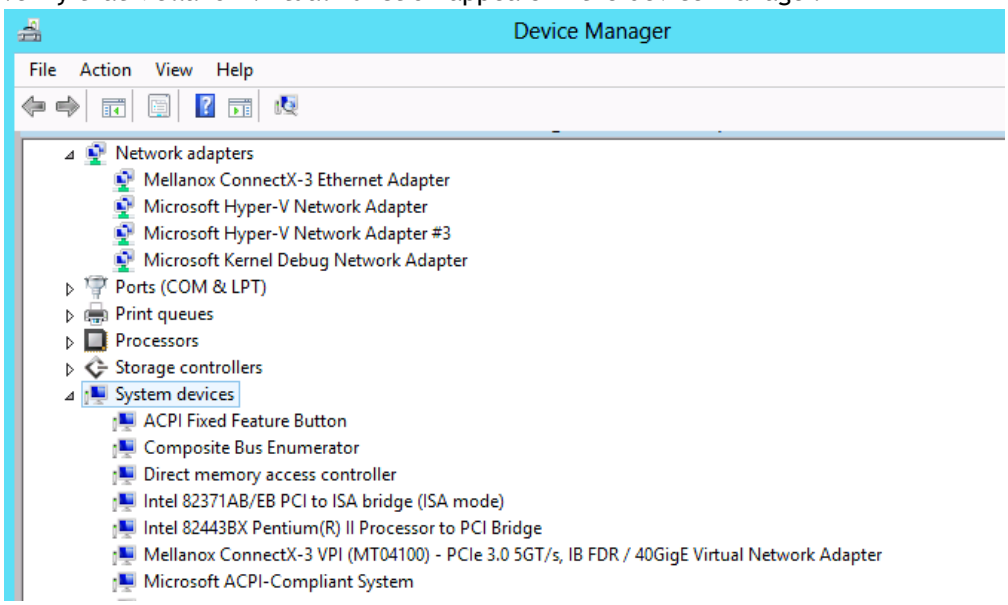


6. Enable the SR-IOV for Mellanox VMNIC:
    a. Open VM settings Wizard.
    b. Open the Network Adapter and choose Hardware Acceleration.
    c. Tick the “Enable SR-IOV” option.

d. Click OK.



7. Start and connect to the Virtual Machine:
8. Select the newly created Virtual Machine and go to: Actions panel-> Connect.
   In the virtual machine window go to: Actions-> Start
9. Copy the WinOF driver package to the VM using Mellanox VMNIC IP address.
10. Install WinOF driver package on the VM.
11. Reboot the VM at the end of installation.
12. Verify that Mellanox Virtual Function appears in the device manager.



To achieve best performance on SR-IOV VF, please run the following powershell commands on the host:

- For 10Gbe:
  PS $ Set-VMNetworkAdapter -Name "Network Adapter" -VMName vm1 -IovQueuePairsRequested 4
- For 40Gbe and 56Gbe:
  PS $ Set-VMNetworkAdapter -Name "Network Adapter" -VMName vm1 -IovQueuePairsRequested 8

VF Spoof Protection

WinOF-2 supports two levels of spoof protection:

- Hypervisor sets VF's MAC address and only packets with that MAC can be transmitted by the VF
- Hypervisor can control allowed Ethertypes that the VF can transmit

If a VF attempts to transmit packets with undesired source MAC or Ethertype, the packets will be dropped by an internal e-Switch.

By default, the anti-spoof filter is enabled with the following Ethertypes:

- Internet Protocol version 4 (IPv4) (0x0800)
- Internet Protocol Version 6 (IPv6) (0x86DD)
- Address Resolution Protocol (ARP) (0x0806)

The hypervisor can configure an Ethertype table for VFs, which includes a set of allowed Ethertypes values for transmission via the NIC registry. The registry keys are as follows:

Key Name	Key Type	Values	Description
VFAllowedTxEtherTypeListEnable	REG_SZ	0 = Disabled 1 = Enabled (default)	Enables/disables the feature
VFAllowedTxEtherType0	REG_DWORD	Ethertype value	The first Ethertype to allow VF to transmit
VFAllowedTxEtherType1	REG_DWORD	Ethertype value	The second Ethertype to allow VF to transmit
VFAllowedTxEtherType2	REG_DWORD	Ethertype value	The third Ethertype to allow VF to transmit
VFAllowedTxEtherType3	REG_DWORD	Ethertype value	The fourth Ethertype to allow VF to transmit
VFAllowedTxEtherType4	REG_DWORD	Ethertype value	The fifth Ethertype to allow VF to transmit
VFAllowedTxEtherType5	REG_DWORD	Ethertype value	The sixth Ethertype to allow VF to transmit
VFAllowedTxEtherType6	REG_DWORD	Ethertype value	The seventh Ethertype to allow VF to transmit
VFAllowedTxEtherType7	REG_DWORD	Ethertype value	The eighth Ethertype to allow VF to transmit

- By default, the feature is enabled and uses the default Ethertype table.

- The Source MAC protection cannot be disabled, and the Ethertype protection can be disabled by setting the VFAllowedTxEtherTypeListEnable key to 0.
- When the feature is disabled, only the Ethernet flow control protocol (0x8808) is restricted to be transmitted by the VF.
- Configuring at least one Ethertype in the registry will override the default table of the Ethertypes mentioned above.

VF's DHCP Redirections

This feature forces every received\sent DHCP packet to be redirected to PF, including DHCP packets sent or received for VFs. The detection of a packet as a DHCP is done by checking UDP-Ports 67 and 68.

> ⚠ When using devices older than ConnectX-5 (i.e. ConnectX-4 and ConnectX-4 Lx) and when this capability is set to 'on', the VF's version must be higher than WinOF-2 v2.50.

To enable this new capability, the steps below are required:

1. Set the PF to work on promiscuous mode to enable PF to receive DHCP packet from various ethernet addresses.
2. Add to the NIC a new registry named "RedirectVfDHCPToPF" and set this registry to '1'.

Key Name	Key Type	Values	Description
RedirectVfDHCPToPF	REG_SZ	0 = Disabled (default) 1 = Enabled	Enables/disables the feature. **Note:** After changing the registry key''s value, driver restart is required.

VF Monitoring

> ⚠ The feature is not supported in VMs, only in Hyper-V Host.

VF CPU Monitor capability allows the user to check two characteristics of VFs, namely - VF `'FwCpuUsage'` and `'Errors2FW'` counters. If the values of these counters are too high, warnings will be presented in the Event Log. The warnings come from the Host driver, which reads the `'FwCpuUsage'` and `'Errors2FW'` counters automatically once in `VfCpuMonBatchPeriodSec` seconds, compares the results with the previous reading, and issues warnings if the difference in values is greater than `VfCpuMonFwCpuUsageMax` and `VfCpuMonErrors2FwMax` thresholds correspondently.

The Event message format is as follows:

```
VF <vf_id> used too many resources over the last %4 seconds: FwCpuUsage %5%, Errors2Fw %6.
```

Note that `<vf_id>` can theoretically be incorrect if the reported VF was de-attached and another new VF was assigned its number.

For further information on detach/attach events see Microsoft Event Log file %SystemRoot% \System32\Winevt\Logs\Microsoft-Windows-Hyper-V-Worker-Admin.evtx.

Customer Facilities

- Batch Request: The driver reads the VF ‘FwCpuUsage’ and ‘Errors2FW’ counters using new FW “batch request” which allows reading one counter from all VFs in a single command. The result of this command is a resource dump. The user can perform the batch request, using mlx5cmd.exe (see more in Resource Dump section).
  The below are examples of how to read the counters:
  - To read the VF ‘FwCpuUsage’ counter from VF0 to VF31:

    ```
    mlx5cmd.exe -dbg -ResourceDump -Dump -Segment 0x5000 -Index1 1 -NumOfObj1 32 -Index2 1 -Depth 1
    ```

  - To read the VF ‘Errors2FW’ counter from VF1 to VF8:

    ```
    mlx5cmd.exe -dbg -ResourceDump -Dump -Segment 0x5000 -Index1 2 -NumOfObj1 8 -Index2 2 -Depth 1
    ```

  The tool will print the name of the folder with the result, written into a file.

> ⚠ The ‘-NumOfObj1’ special values 0xffff and 0xfffe are not supported for the segment 0x5000.

- Feature state" The user can check the state of the feature by running the ‘mlx5cmd -Features’ command. If the feature is not supported by the firmware or was disabled by default or by the user, the tool prints
  State: Disabled
  Otherwise, the tool prints
  State: Enabled
  - To disable/enable the feature, change the value of the VfCpuMonEnable parameter.
  - To print the configuration parameters, run:

    ```
    mlx5cmd -RegKeys -DynamicKeys | grep VfCpu
    ```

  For further information, see “VF Monitoring Registry Keys”.

Virtual Machine Multiple Queue (VMMQ)

> ⚠ VMMQ is supported in Windows Server 2016 and above only, when using Ethernet mode (No IPoIB).

Virtual Machine Multiple Queues (VMMQ), formerly known as Hardware vRSS, is a NIC offload technology that provides scalability for processing network traffic of a VPort in the host (root partition) of a virtualized node. In essence, VMMQ extends the native RSS feature to the VPorts that are associated with the physical function (PF) of a NIC including the default VPort.

VMMQ is available for the VPorts exposed in the host (root partition) regardless of whether the NIC is operating in SR-IOV or VMQ mode.

System Requirements	
Operating System(s):	Windows Server 2016
Adapter Cards	NVIDIA® ConnectX-4/ConnectX-4 Lx/ConnectX-5 adapter card family

SR-IOV Support Limitations

The below table summarizes the SR-IOV working limitations, and the driver's expected behavior in unsupported configurations.

WinOF-2 Version	ConnectX-4 Version	Adapter Mode		
		InfiniBand		Ethernet
		SR-IOV On	SR-IOV Off	SR-IOV On/ Off
Earlier versions	Up to 12.16.1020	Driver will fail to load and show "Yellow Bang" in the device manager.		No limitations
1.50 onwards	12.17.2020 onwards (IPoIB supported)	"Yellow Bang" unsupported mode - disable SR-IOV via mlxConfig	OK	No limitations

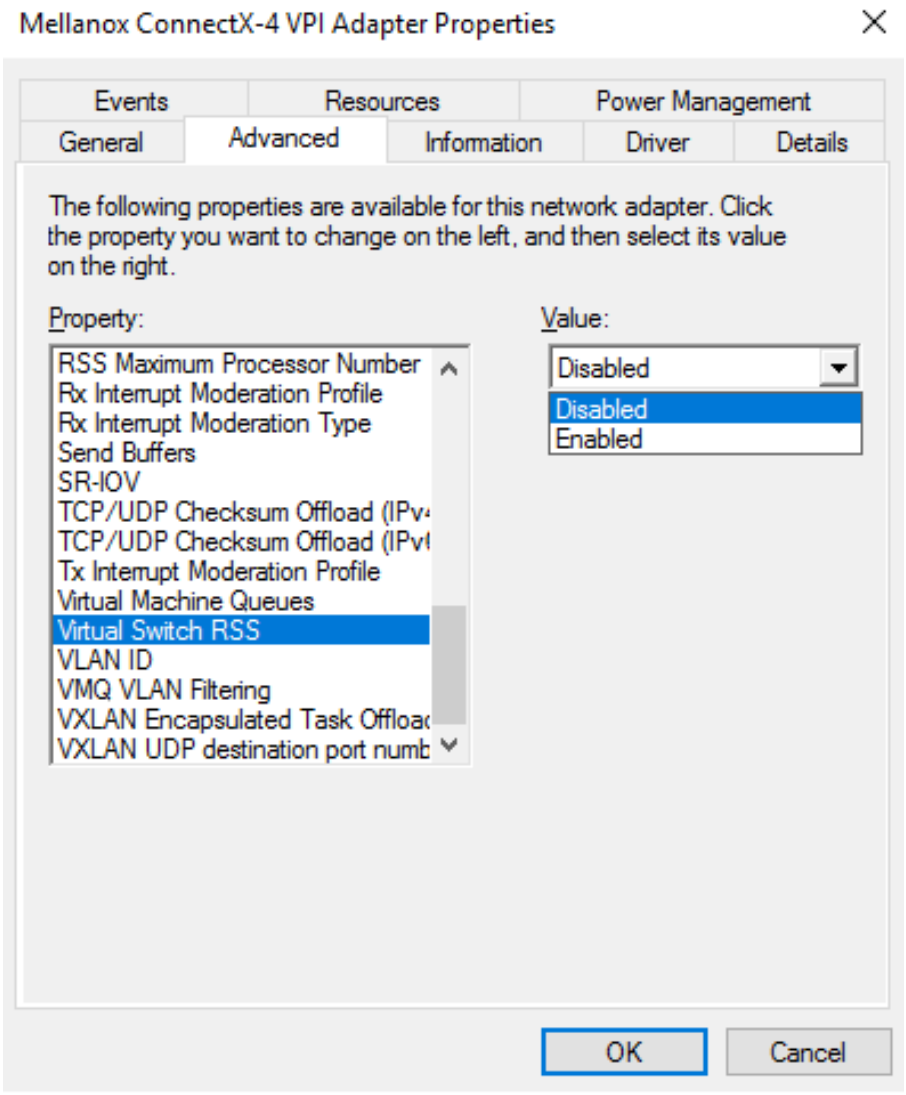
For further information on how to enable/disable SR-IOV, please refer to section Single Root I/O Virtualization (SR-IOV).

Enabling/Disabling VMMQ

- On the Driver Level

  To enable/disable VMMQ:

a. Go to: Display Manager-> Network adapters->Mellanox ConnectX-4/ConnectX-5 Ethernet Adapter->Properties-> advanced tab->Virtual Switch Rss



b. Select Enabled or Disabled

➢ *To enable/disable VMMQ using a Registry Key:*

Set the *RssOnHostVPorts* registry key in the following path to either 1 (enabled) or 0 (disabled).

```
HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>\* RssOnHostVPorts
```

- On a VPort

  ➢ *To enable VMMQ on a* VPort:

  ```
  PS $ Set-VMNetworkAdapter -Name "Virtual Adapter Name" -VmmqEnabled $true
  ```

  ➢ *To disable VMMQ on a VPort:*

  ```
  PS $ Set-VMNetworkAdapter -Name "Virtual Adapter Name" -VmmqEnabled $false
  ```

Since the VMMQ is an offload feature for vRss, vRss must be enabled prior to enabling VMMQ.

Controlling the Number of Queues Allocated for a vPort

The requested number of queues for a virtual network adapter (vPort) can be set by invoking this PS cmdlet:

```
PS $ Set-VMNetworkAdapter -VMName "VM Name" -name "Virtual Adapter Name" -VmmqQueuePairs <number>
```

The number provided to this cmdlet is the requested number of queues per vPort. However, the OS might decide to not fulfill the request due to some resources and other factors considerations.

Network Direct Kernel Provider Interface

As of v1.45, WinOF-2 supports NDIS Network Direct Kernel Provider Interface version 2. The Network Direct Kernel Provider Interface (NDKPI) is an extension to NDIS that allows IHVs to provide kernel-mode Remote Direct Memory Access (RDMA) support in a network adapter.

System Requirement	
Operating System:	Windows Server 2012 R2 and above (Without NDK from/to a VM) and Windows Client 10 and above.

Configuring NDK

General Configurations

1. Make sure the port is configured as Ethernet.
2. Make sure the RoCE mode is configured the same on both ends, run “Mlx5Cmd -stat” from the "Command Prompt". ROCE v2 is the default mode.

Configuring NDK for Virtual NICs

1. Create a VMSwitch.

```
PS $ New-VMSwitch -Name <vSwitchName> -NetAdapterName <EthInterfaceName> -AllowManagementOS $False
```

2. Create the virtual network adapters.

```
PS $ Add-VMNetworkAdapter -SwitchName <vSwitchName> -Name <EthInterfaceName> -ManagementOS
```

3. Enable the "Network Direct (RDMA)" on the new virtual network adapters.

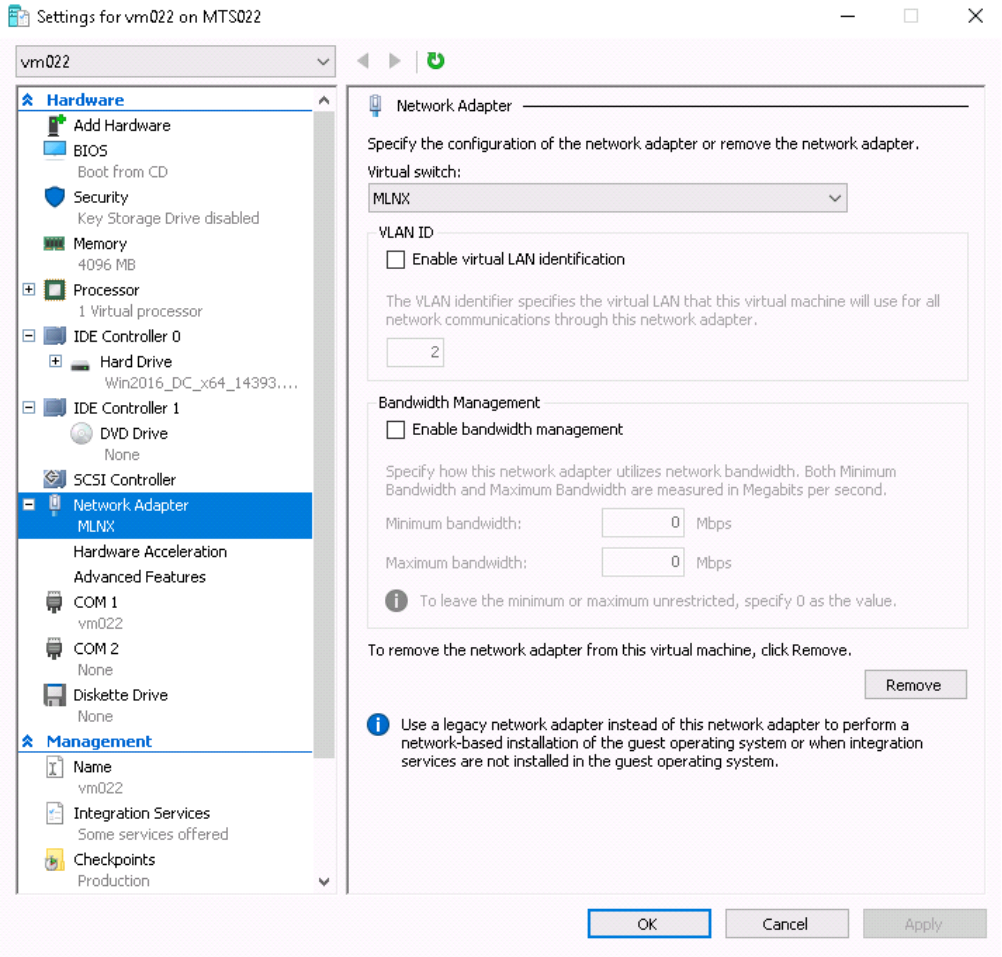
```
PS $ Enable-NetAdapterRdma <EthInterfaceName>
```

Configuring the VM

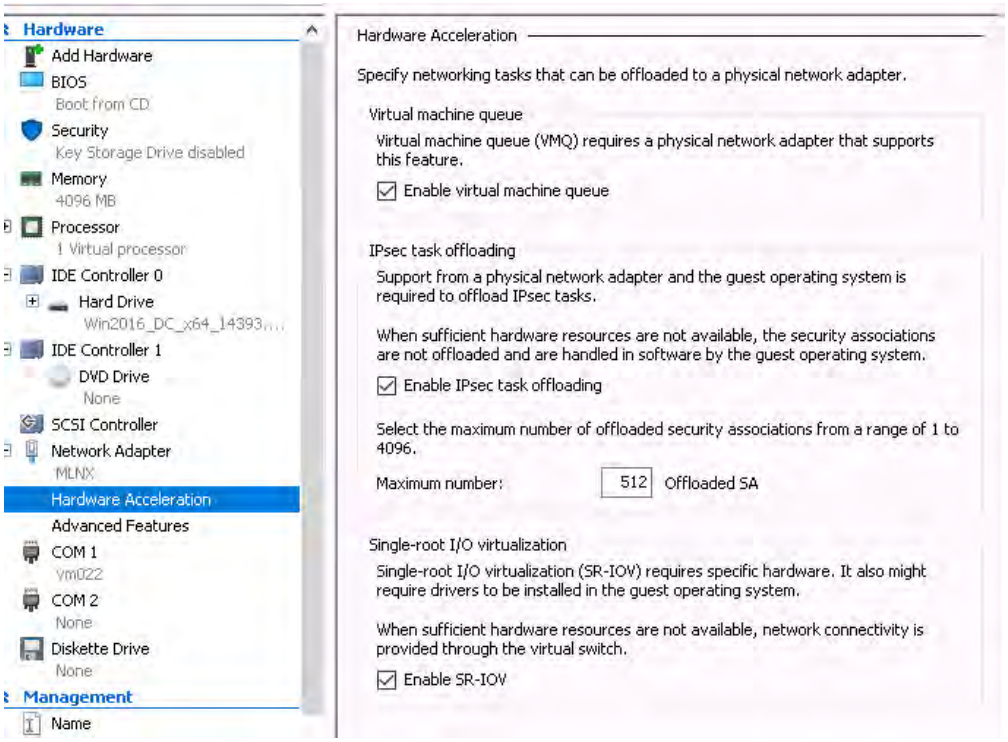
1. Make sure your machine supports SR-IOV.
2. Create a VM (make sure the VM is running the same OS as host)
3. Create an SR-IOV enabled VMSwitch.

```
PS $ New-VMSwitch -Name <vSwitchName> -NetAdapterName <EthInterfaceName> -EnableIov $True
-AllowManagementOS $True
```

4. Add a Network Adapter to the VM in the Hyper-V Manager, and choose the VMSwitch just created.



5. Check the "Enable SR-IOV" option on the "Hardware Acceleration" under the Network Adapter.



> ⚠ If you turn ON the VM at this time in the VM Device Manager, you should see Mellanox Virtual Adapter under the Network adapters.

6. Install the NVIDIA® Driver in the VM.
   Use the same package you installed on the host.
7. Enable RDMA on the corresponding network adapter in the VM (Run the command in the VM).

```
PS $ Enable-NetAdapterRdma <EthInterfaceName>
```

Configuring Guest RDMA for Windows Server 2016

> ⚠ The following is applicable to Windows Server 2016 and above.

Before attending to the below steps, accomplish the configuration detailed in section Configuring the VM.

1. Configure the Guest RDMA, keep the VM up and running, and run the following command on the host:

```
Set-VMNetworkAdapter -VMName <VM name> -IovWeight 0
Set-VMNetworkAdapterRdma -VMName <VM name> -RdmaWeight <0 | 100>
Set-VMNetworkAdapter -VMName <VM name> -IovWeight 100
```

   Options:

Value	Usage
IovWeight	VF allocation
0	Detach the VF
100	Attach the VF
RdmaWeight	RDMA capability
0	Disable RDMA for this specific VM
100	Enable RDMA for this specific VM

2. Query whether a specific VM has RDMA capability, run the following command:

```
Get-VMNetworkAdapterRdma -VMName <VM name>
```

> ⚠ Any non-zero value for the RdmaWeight field indicates that RDMA capability is true for this VM.

Utility to Run and Monitor NDK

Running NDK

Since SMB is NDK's client, it should be used to generate traffic. To generate traffic, do a big copy from one machine to the other.

For instance, use "xcopy" to recursively copy the entire c:\Windows directory or from a "Command Prompt" window, run:

```
xcopy /s c:\Windows \\<remote machine ip>\<remote machine directory for receiving>
```

Example:

```
xcopy /s c:\Windows \\11.0.0.5\c$\tmp
```

Validating NDK

During the run time of NDK test (xcopy), with "RDMA Activity" in the perfmon. Use the Mlx5Cmd sniffer to see the protocol information of the traffic packet.

PacketDirect Provider Interface

> ⚠ PacketDirect is supported on Ethernet ports only (no IPoIB).

As of v1.45, WinOF-2 supports NDIS PacketDirect Provider Interface. PacketDirect extends NDIS with an accelerated I/O model, which can increase the number of packets processed per second by an

order of magnitude and significantly decrease jitter when compared to the traditional NDIS I/O path.

System Requirements	
Hypervisor OS:	Windows Server 2012 R2 and above, and Windows Client 10 and above
Virtual Machine (VM) OS:	Windows Server 2012 and above
Adapter Cards:	NVIDIA® ConnectX-4/ConnectX-4 Lx/ConnectX-5/ConnectX-5 Ex
Driver:	NVIDIA® WinOF-2 1.45 or higher
Firmware version:	12.16.1020/14.16.1020 or higher

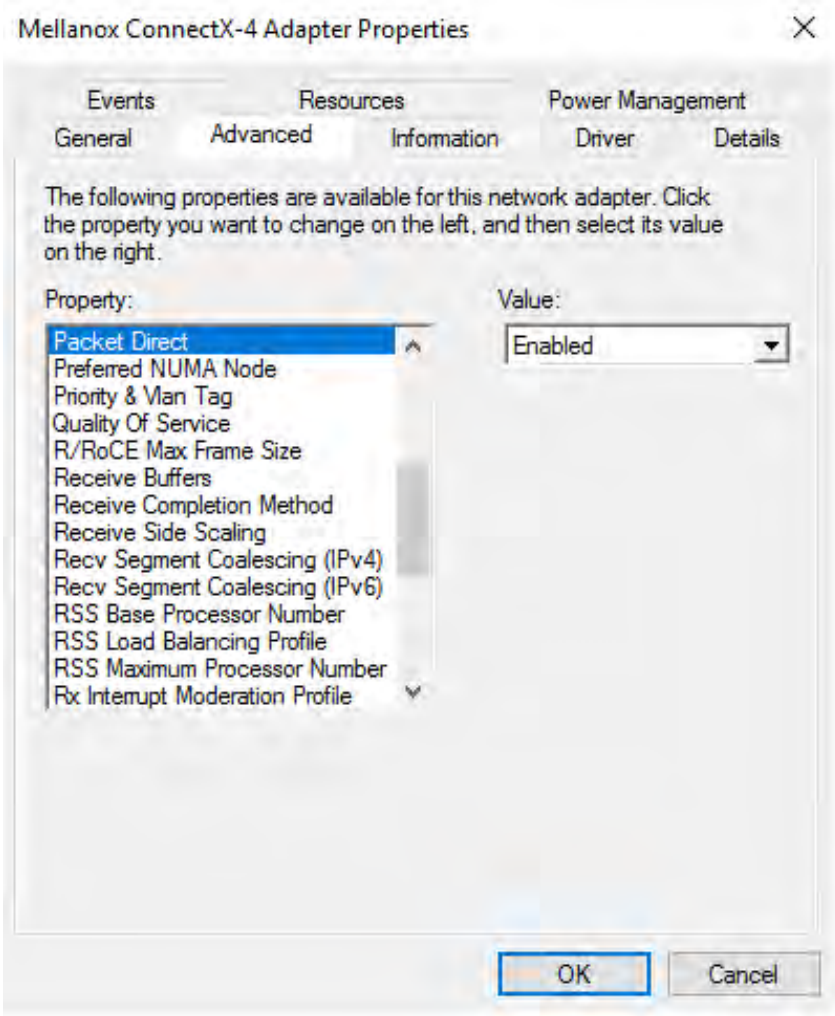
Using PacketDirect for VM

To allow a VM to send/receive traffic in PacketDirect mode:

1. Enable PacketDirect:
    - On the Ethernet adapter.

```
PS $ Enable-NetAdapterPacketDirect -Name <EthInterfaceName>
```

- In the Device Manager.



2. Create a vSwitch with PacketDirect enabled.

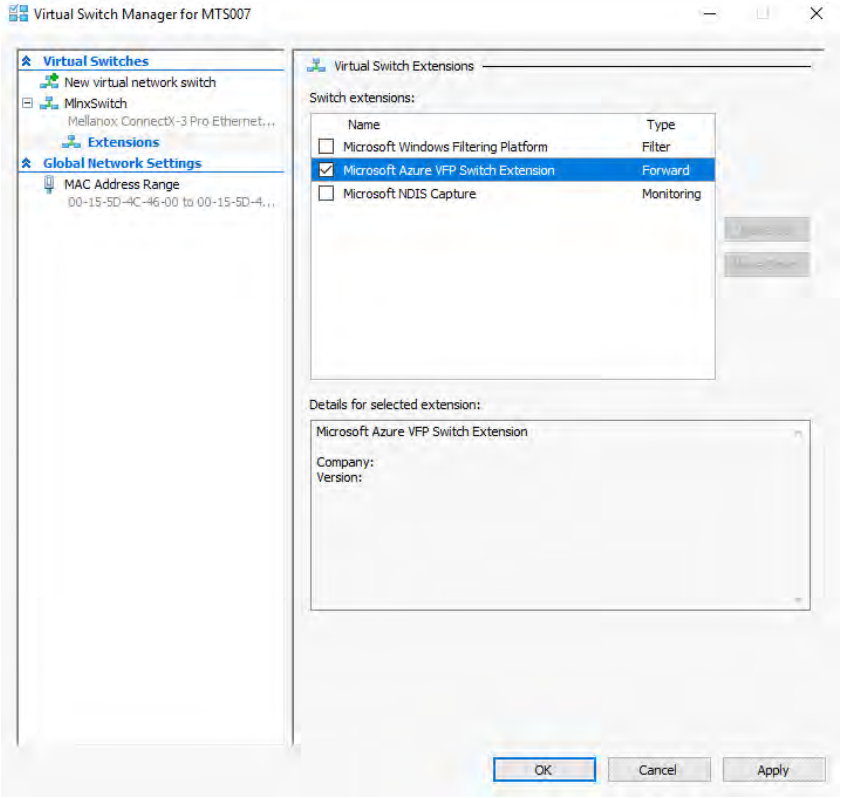
```
PS $ New-VMSwitch <vSwitchName> -NetAdapterName <EthInterfaceName> -EnablePacketDirect $true
 -AllowManagementOS $true
```

3. Enable VFP extension:
   - On the vSwitch

```
PS $ Enable-VMSwitchExtension -VmSwitchName <vSwitchName> -Name "Windows Azure VFP Switch
Extension"
```

> ⚠ Starting from Windows Server 2016, to enable the VFP extension, use the following command instead:
> *Enable-VMSwitchExtension -VmSwitchName <vSwitchName> -Name "Microsoft Azure VFP Switch Extension"*

- In the Hyper-V Manager: Action->Virtual Switch Manager...



4. Shut down the VM.

```
PS $ Stop-VM -Name <VMName> -Force -Confirm
```

5. Add a virtual network adapter for the VM.

```
PS $ Add-VMNetworkAdapter -VMName <VMName> -SwitchName <vSwitchName> -StaticMacAddress <StaticMAC Address>
```

6. Start the VM.

```
PS $ Start-VM -Name <VMName>
```

   Since VFP is enabled, without any forwarding rules, it will block all traffic going through the VM.

7. Unblock the traffic, find the port name for the VM.

```
CMD > vfpctrl /list-vmswitch-port
......
Port name : E431C413-D31F-40EB-AD96-0B2D45FE34AA
Port Friendly name :
Switch name : 8B288106-9DB6-4720-B144-6CC32D53E0EC
Switch Friendly name : MlnxSwitch
PortId : 3
VMQ Usage : 0
SR-IOV Usage : 0
Port type : Synthetic
Port is Initialized.
MAC Learning is Disabled.
NIC name : bd65960d-4215-4a4f-bddc-962a5d0e2fa0--e7199a49-6cca-4d3c-a4cd-22907592527e
NIC Friendly name : testnic
```

```
MTU : 1500
MAC address : 00-15-5D-4C-46-00
VM name : vm
.....
Command list-vmswitch-port succeeded!
```

8. Disable the port to allow traffic.

```
CMD > vfpctrl /disable-port /port <PortName>
Command disable-port succeeded!
```

The port should be disabled after each reboot of the VM to allow traffic.

Data Plane Development Kit (DPDK)

DPDK is a set of libraries and optimized NIC drivers for fast packet processing in user space. It provides a framework and common API for high speed networking applications.

The WinOF driver supports running DPDK from an SR-IOV virtual machine, see Single Root I/O Virtualization (SR-IOV).

For further information, see NVIDIA®'s DPDK documentation:

- DPDK Quick Start Guide
- NVIDIA® DPDK Release Notes

Flows Prerequisites

- The DPDK flows must have a valid source MAC.
- The flows' VLAN is determined by the Hyper-V.

Configuring the Driver Registry Keys

NVIDIA® IPoIB and Ethernet drivers use registry keys to control the NIC operations. The registry keys receive default values during the installation of the NVIDIA® adapters. Most of the parameters are visible in the registry by default, however, certain parameters must be created in order to modify the default behavior of the NVIDIA® driver.
The adapter can be configured either from the User Interface (Device Manager -> Mellanox Adapter -> Right click -> Properties) or by setting the registry directly.
All NVIDIA® adapter parameters are located in the registry under the following registry key:

```
HKEY_LOCAL_MACHINE
\SYSTEM
\CurrentControlSet
\ Control
\ Class
\{4D36E972-E325-11CE-BFC1-08002bE10318}
\<Index>
```

The registry key can be divided into 4 different groups:

Group	Description
Basic	Contains the basic configuration.
Offload Options	Controls the offloading operation that the NIC supports.

Group	Description
Performance Options	Controls the NIC operation in different environments and scenarios.
Flow Control Options	Controls the TCP/IP traffic.

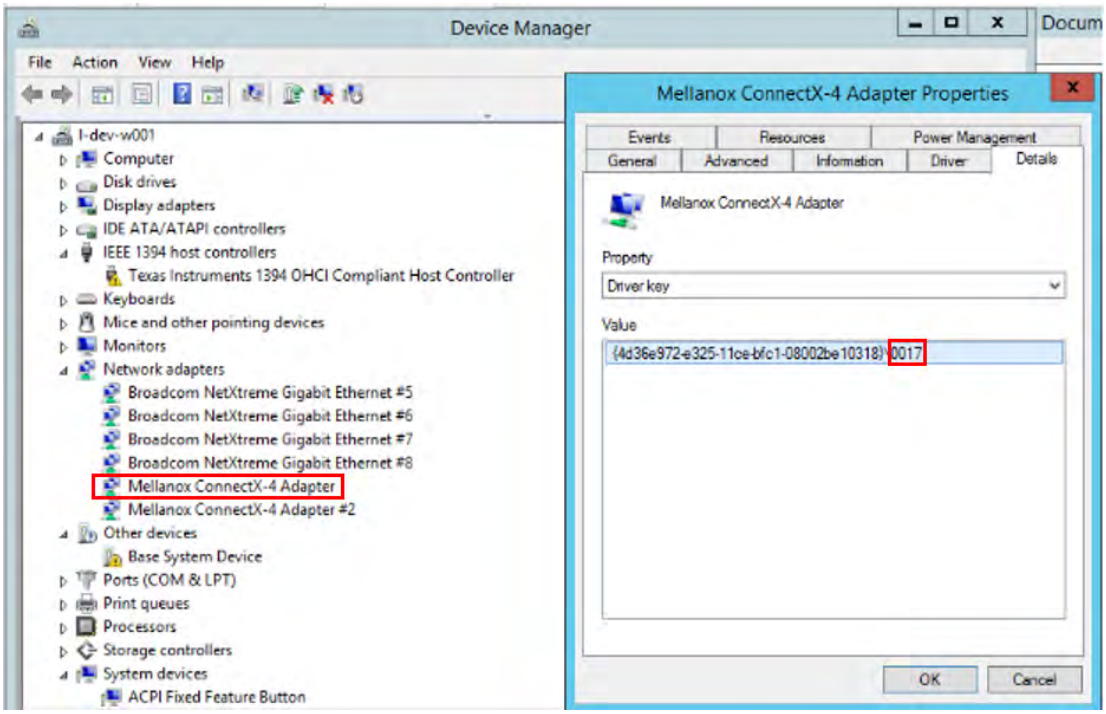
Any registry key that starts with an asterisk ("*") is a well-known registry key. For more details regarding the registries, please refer to:
http://msdn.microsoft.com/en-us/library/ff570865(v=VS.85).aspx

Finding the Index Value of the Network Interface

➢ *To find the index value of your Network Interface from the Device Manager please perform the following steps:*

1. Open Device Manager, and go to Network Adapters.
2. Right click ->Properties on Mellanox Connect-X® Ethernet Adapter.
3. Go to Details tab.
4. Select the Driver key, and obtain the nn number.

In the below example, the index equals 0010.



All registry keys added for driver configuration should be of string type (REG_SZ).

After setting a registry key and re-loading the driver, you may use the *mlx5cmd -regkeys* command to assure that the value was read by the driver.

Basic Registry Keys

This group contains the registry keys that control the basic operations of the NIC

Value Name	Default Value	Description
*JumboPacket	ETH: 1514 IPoIB: 4092	The maximum size of a frame (or a packet) that can be sent over the wire. This is also known as the maximum transmission unit (MTU). The MTU may have a significant impact on the network's performance as a large packet can cause high latency. However, it can also reduce the CPU utilization and improve the wire efficiency. The standard Ethernet frame size is 1514 bytes, but NVIDIA® drivers support wide range of packet sizes. The valid values are: • Ethernet: 614 up to 9614 • IPoIB: 600 up to 4092 **Note**: All the devices across the network (switches and routers) should support the same frame size. Be aware that different network devices calculate the frame size differently. Some devices include the header, i.e. information in the frame size, while others do not. NVIDIA® adapters do include Ethernet header information in the frame size. (i.e when setting *JumboPacket to 1514, the actual payload size is 1500 bytes).
*ReceiveBuffers	512	The number of packets each ring receives. This parameter affects the memory consumption and the performance. Increasing this value can enhance receive performance, but also consumes more system memory. In case of lack of received buffers (dropped packets or out of order received packets), you can increase the number of received buffers. The valid values are 256 up to 4096.
*TransmitBuffers	2048	The number of packets each ring sends. Increasing this value can enhance transmission performance, but also consumes system memory. The valid values are 256 up to 4096.
*NetworkDirect	1	The *NetworkDirect keyword determines whether the miniport driver's NDK functionality can be enabled. If this keyword value is set to 1 ("Enabled"), NDK functionality can be enabled. If it is set to 0 ("Disabled"), NDK functionality cannot be enabled. **Note:** This key is enabled by default, thus NDK will be used. It is important to set the switch to enable ECN and/or PFC otherwise the system will experience performance degradation. **Note:** This key affects NDK functionality and not Userspace ND (Network Direct). For further details, see: https://msdn.microsoft.com/en-us/windows/hardware/drivers/ network/enabling-and-disabling-ndk-functionality
*NetworkDirectTechnology	0	The *NetworkDirectTechnology keyword determines the technology used for the device. The valid values are: • 0 - Device Default • 3 - RoCE • 4 - RoCE v2 For further details, see: https://docs.microsoft.com/en-us/windows-hardware/drivers/network/inf-requirements-for-ndkpi

General Registry Keys

Key Name	Key Type	Values	Description
ThreadedDpcEnable	DWORD	• 0 - Disabled • 1 - Enabled	Controls the threaded DPC mode enablement for Rx traffic completion processing.
TxThreadedDpcEnable	DWORD	• 0 - Disabled • 1 - Enabled	Controls the threaded DPC mode enablement for Tx traffic completion processing.
CheckForHangTOInSeconds	REG_DWORD	[0 - MAX_ULONG] Default: 4	The interval in seconds for the Check- for-Hang mechanism **Note:** This registry key is available only when using WinOF-2 v2.0 and later. **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

Offload Registry Keys

This group of registry keys allows the administrator to specify which TCP/IP offload settings are handled by the adapter rather than by the operating system.

Enabling offloading services increases transmission performance. Due to offload tasks (such as checksum calculations) performed by adapter hardware rather than by the operating system (and, therefore, with lower latency). In addition, CPU resources become more available for other tasks.

Value Name	Default Value	Description
*LsoV2IPv4	1	Large Send Offload Version 2 (IPv4). The valid values are: • 0: disable • 1: enable
*LsoV2IPv6	1	Large Send Offload Version 2 (IPv6). The valid values are: • 0: disable • 1: enable
LSOSize	64000	The maximum number of bytes that the TCP/IP stack can pass to an adapter in a single packet. This value affects the memory consumption and the NIC performance. The valid values are MTU+1024 up to 64000. **Note:** This registry key is not exposed to the user via the UI. If LSOSize is smaller than MTU+1024, LSO will be disabled.
LSOMinSegment	2	The minimum number of segments that a large TCP packet must be divisible by, before the transport can offload it to a NIC for segmentation. The valid values are 2 up to 32. **Note**: This registry key is not exposed to the user via the UI.

Value Name	Default Value	Description
LSOTcpOptions	1	Enables that the miniport driver to segment a large TCP packet whose TCP header contains TCP options. The valid values are: • 0: disable • 1: enable **Note**: This registry key is not exposed to the user via the UI.
LSOIpOptions	1	Enables its NIC to segment a large TCP packet whose IP header contains IP options. The valid values are: • 0: disable • 1: enable **Note**: This registry key is not exposed to the user via the UI.
*IPChecksumOffloadIPv4	3	Specifies whether the device performs the calculation of IPv4 checksums. The valid values are: • 0: (disable) • 1: (Tx Enable) • 2: (Rx Enable) • 3: (Tx and Rx enable)
*TCPUDPChecksumOffl oadIPv4	3	Specifies whether the device performs the calculation of TCP or UDP checksum over IPv4. The valid values are: • 0: (disable) • 1: (Tx Enable) • 2: (Rx Enable) • 3: (Tx and Rx enable)
*TCPUDPChecksumOffl oadIPv6	3	Specifies whether the device performs the calculation of TCP or UDP checksum over IPv6. The valid values are: • 0: (disable) • 1: (Tx Enable) • 2: (Rx Enable) • 3: (Tx and Rx enable)
*RssOnHostVPorts	1	Virtual Machine Multiple Queue (VMMQ) HW Offload The valid values are: • 0: disable • 1: enable
SwParsing	N/A	Specifies whether the device performs the calculation of TCP checksum over IP-in-IP encapsulated IPv4/6 sent packets. The valid values are: • 0: (disable) • 1: (Tx Enable)
UsoIPv4	1	UDP Segmentation Offload (IPv4). The valid values are: • 0: (Disable) • 1: (Enable)
UsoIPv6	1	UDP Segmentation Offload (IPv6). The valid values are: • 0: (Disable) • 1: (Enable)

Performance Registry Keys

This group of registry keys configures parameters that can improve adapter performance.

Value Name	Default Value	Description
TxIntModerationProfile	1	Enables the assignment of different interrupt moderation profiles for send completions. Interrupt moderation can have great effect on optimizing network throughput and CPU utilization. The valid values are: • 0: Low Latency Implies higher rate of interrupts to achieve better latency, or to handle scenarios where only a small number of streams are used. • 1: Moderate Interrupt moderation is set to midrange defaults to allow maximum throughput at minimum CPU utilization for common scenarios. • 2: Aggressive Interrupt moderation is set to maximal values to allow maximum throughput at minimum CPU utilization for more intensive, multi-stream scenarios. • 3: Dynamic Improve existing system performance by changing interrupt moderation dynamically while also decreasing latency and CPU usage **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
RxIntModerationProfile	1	Enables the assignment of different interrupt moderation profiles for receive completions. Interrupt moderation can have a great effect on optimizing network throughput and CPU utilization. The valid values are: • 0: Low Latency Implies higher rate of interrupts to achieve better latency, or to handle scenarios where only a small number of streams are used. • 1: Moderate Interrupt moderation is set to midrange defaults to allow maximum throughput at minimum CPU utilization for common scenarios. • 2: Aggressive Interrupt moderation is set to maximal values to allow maximum throughput at minimum CPU utilization, for more intensive, multi-stream scenarios. • 3: Dynamic Improve existing system performance by changing interrupt moderation dynamically while also decreasing latency and CPU usage **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

Value Name	Default Value	Description
RecvCompletionMethod	1	Sets the completion methods of the receive packets, and it affects network throughput and CPU utilization. The supported methods are: • Polling - increases the CPU utilization, because the system polls the received rings for incoming packets; however, it may increase the network bandwidth since the incoming packet is handled faster. • Adaptive - combines the interrupt and polling methods dynamically, depending on traffic type and network usage. The valid values are: • 0: polling • 1: adaptive
*InterruptModeration	1	Sets the rate at which the controller moderates or delays the generation of interrupts, making it possible to optimize network throughput and CPU utilization. When disabled, the interrupt moderation of the system generates an interrupt when the packet is received. In this mode, the CPU utilization is increased at higher data rates, because the system must handle a larger number of interrupts. However, the latency is decreased, since that packet is processed more quickly. When interrupt moderation is enabled, the system accumulates interrupts and sends a single interrupt rather than a series of interrupts. The valid values are: • 0: disable • 1: enable **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
RxIntModeration	2	Sets the rate at which the controller moderates or delays the generation of interrupts, making it possible to optimize network throughput and CPU utilization. The default setting (Adaptive) adjusts the interrupt rates dynamically, depending on traffic type and network usage. Choosing a different setting may improve network and system performance in certain configurations. The valid values are: • 0: off • 1: static • 2: adaptive • 3: dynamic The interrupt moderation count and time are configured dynamically, based on traffic types and rate.

Value Name	Default Value	Description
TxIntModeration	4 - Default	Sets the rate at which the controller moderates or delays the generation of interrupts, making it possible to optimize network throughput and CPU utilization. The default setting (Adaptive) adjusts the interrupt rates dynamically, depending on traffic type and network usage. Choosing a different setting may improve network and system performance in certain configurations. The valid values are: • 0: off • 1: static • 2: adaptive • 3: dynamic • 4: default The interrupt moderation count and time are configured dynamically, based on traffic types and rate. Default mode (4) will set Adaptive (2) for RSS mode setup configuration, and OFF (0) in all other cases.
*RSS	1	Sets the driver to use Receive Side Scaling (RSS) mode to improve the performance of handling incoming packets. This mode allows the adapter port to utilize the multiple CPUs in a multi-core system for receiving incoming packets and steering them to their destination. RSS can significantly improve the number of transactions per second, the number of connections per second, and the network throughput. This parameter can be set to one of two values: • 1: enable (default) Sets RSS Mode. • 0: disable The hardware is configured once to use the Toeplitz hash function and the indirection table is never changed.
ThreadPoll	3000	The number of cycles that should be passed without receiving any packet before the polling mechanism stops when using polling completion method for receiving. Afterwards, receiving new packets will generate an interrupt that reschedules the polling mechanism. The valid values are 0 up to 200000. **Note**: This registry value is not exposed via the UI. **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
VlanId	ETH: 0	Enables packets with VlanId. It is used when no team intermediate driver is used. The valid values are: • 0: disable No VLAN Id is passed. • 1-4095 Valid VLAN ID that will be passed. **Note**: This registry value is only valid for Ethernet.
*NumRSSQueues	8	The maximum number of the RSS queues that the device should use. **Note**: This registry key is only in Windows Server 2012 and above.

Value Name	Default Value	Description
BlueFlame	1	The latency-critical Send WQEs to the device. When a BlueFlame is used, the WQEs are written directly to the PCI BAR of the device (in addition to memory), so that the device may handle them without having to access memory, thus shortening the execution latency. For best performance, it is recommended to use the BlueFlame when the HCA is lightly loaded. For high- bandwidth scenarios, it is recommended to use regular posting (without BlueFlame). The valid values are: • 0: disable • 1: enable **Note**: This registry value is not exposed via the UI.
*MaxRSSProcessors	8	The maximum number of RSS processors. **Note**: This registry key is only in Windows Server 2012 and above.
AsyncReceiveIndicate	0	Disabled default
	1	Enables packet burst buffering using threaded DPC
	2	Enables packet burst buffering using polling
RfdReservationFactor	150	Controls the number of reserved receive packets,
*RscIPv4	1	Enable or disable support for RSC for the IPv4 datagram version.
*RscIPv6	1	Enable or disable support for RSC for the IPv6 datagram version.
MaxCallsToNdisIndicate	5	Maximum number of times chained packets can be indicated before packets processing is stop processing is stopped.
RssV2	0	Enables the RSS v2 feature which improves the Receive Side Scaling by offering dynamic, per-VPort spreading of queues. It reduces the time to update the indirection table. **Note:** RSSv2 is only supported by NDIS 6.80 and later versions.
ValidateRssV2	0	Enables strict argument validation for upper layer testing. Set along with the RssV2 key to enable the RSSv2 feature.
StridingRqEnabled	0	When set, enables the Striding RQ feature. The receive buffers are segmented into fixed size strides and each incoming packet (or an LRO aggregate) consumes a buffer of its size.

Value Name	Default Value	Description

NumberOfStrides	16	Relevant when Striding RQ feature is enabled. The value can be power of two in the range 8-256,. This value will determine the number of segments of receive buffer.
		In General, Receive buffer size is determined by the maximum between RscMaxPacketSize and *JumboPacket. (for this value we might add headers or additional alignments required by HW).
		The buffer size is divided into NumberOfStrides segments. Each segment size can be of range 64-8192.
		In case of inconsistency with those values, the following Event log message will be displayed:
		`MLX_EVENT_LOG_ILLEGAL_STRIDE_RQ_PARAM will appear and Receive buffers will not be segmented.`
		`All values can be seen via tool using command: mlx5Cmd -Stat -Verbose`

*RscIPv4	unset	When set to '1' LRO is enabled.
*RscIPv6	unset	When set to '1' LRO is enabled.
RscMaxPacketSize	unset	In this configuration, this value should be from 16KB up to 64KB (64*1024).
EnableZtt	0x0 (Disable)	ZTT register enables users to configure the device zero touch tuning algorithm. 0x0: Disable (Default) 0x1: Enable

Ethernet Registry Keys

The following section describes the registry keys that are only relevant to Ethernet driver.

Value Name	Default Value	Description
RoceFrameSize	Unset (Will be derived from JumboPacket)	The maximum size of a frame (or a packet) that can be sent by the RoCE protocol (a.k.a Maximum Transmission Unit (MTU). Using larger RoCE MTU will improve the performance; however, one must ensure that the entire system, including switches, supports the defined MTU. Ethernet packet uses the general MTU value, whereas the RoCE packet uses the RoCE MTU. When defining the RoCE MTU, the size of the JumboPacket should be taken into consideration. The value must be set according to the following formula: **JumboPacket >= RoCE_MTU + Header** The valid values are: • 256 • 512 • 1024 • 2048 • 4096 **Note**: This registry key is supported only in Ethernet drivers.

Value Name	Default Value	Description
*PriorityVLANTag	**3**: Packet Priority & VLAN Enabled	Enables sending and receiving IEEE 802.3ac tagged frames, which include: • 802.1p QoS (Quality of Service) tags for priority-tagged packets. • 802.1Q tags for VLANs. When this feature is enabled, the NVIDIA® driver supports sending and receiving a packet with VLAN and QoS tag.
DeviceRxStallTimeout	1000	The maximum period for a single received packet processing. If the packet was not processed during this time, the device will be declared as stalled and will increase the "Critical Stall Watermark Reached" counter. The value is given in mSec. The maximum period is 8000 mSec. The special value of 0, indicates that the DeviceRxStallTimeout is active. **Range:** 0x0050 (80)- 0x1F40 (8000) **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
DeviceRxStallWatermark	0	The maximum period for a single received packet processing. If the packet was not processed during this time, the device will increase a diagnostic counter called "Minor Stall Watermark Reached". The value is given in mSec. The maximum period is 8000 mSec. The special value of 0 indicates that the DeviceRxStallWatermark is active **Range:** 0x0050 (80)- 0x1F40 (8000) **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
TCHeadOfQueueLifeTimeLimit	0-20 Default: 19	The time a packet can live at the head of a TC queue before it is discarded. The timeout value is defined by 4,096us multiplied by 2^TCHeadOfQueueLifetimeLimit. **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
TCHeadOfQueueLifeTimeLimitEnable	0-255 Default: 255	Enables the TCHeadOfQueueLifeTimeLimit. **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

Value Name	Default Value	Description
TCStallCount	1-7 Default: 1	The number of sequential packets dropped due to Head Of Queue Lifetime Limit, in order for the port to enter the TCStalled state. All packets for the TC are discarded in this state for a period of 8 times the timeout defined by TCHeadOfQueueLifetimeLimit. **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
TCStallEnable	0 - Disabled 1 - Enabled (Default)	Enables/Disables the TCStalled state. **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
TCHeadOfQueueLifeTimeLimitEnable	0	The TCs for which Head Of Queue Lifetime Limit is enabled. Bit 0 represents TC0, bit 1 represents TC1 and so on. The valid values are: • 0-255 • 0: disabled **Note:** As of WinOF-2 v2.20, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
RelaxedOrderingWrite	0 - Disabled 1 - Enabled Default: Auto Detect	When this register is set, a PCIe cycle is issued with "relaxed ordering" attribute (allows write after write bypassing) for writes. **Note:** This register is supported only in Ethernet flows and not RDMA. For additional information on the PCIe relaxed ordering feature please refer to the PCI Express® Base Specification section on Transaction Ordering Rules. Default value is Auto Detect, meaning the feature is enabled always unless the CPU family is Haswell or Broadwell where the feature will be disabled as a performance degradation is expected.
VFAllowedRelaxedOrdering	0 - No Relaxed Ordering will be supported for new VFs 1 - Only Relaxed Ordering Write will be supported for new VFs 2 - Only Relaxed Ordering Read will be supported for new VFs 3 - Both Relaxed Ordering types will be supported for new VFs (Default)	Limits the PCIe relaxed ordering feature for VFs. **Note:** When set to 0, limitation is disabled. Although the key is dynamic, changes will take effect after VFs are created. For additional information on the PCIe relaxed ordering feature please refer to the PCI Express® Base Specification section on Transaction Ordering Rules. **Note:** This registry key cannot be changed in Bluefield 2 SmartNIC mode, the value in this setup will be 3.
DisableLocalLoopbackFlags	0 - Do not disable any local loopback (Default) 1 - Disable Multicast 2 - Disable Unicast 3 - Disable Unicast and Multicast	This key controls whether or not to disable any local Loopback.

Flow Control Options

This group of registry keys allows the administrator to control the TCP/IP traffic by pausing frame transmitting and/or receiving operations. By enabling the Flow Control mechanism, the adapters can overcome any TCP/IP issues and eliminate the risk of data loss.

Value Name	Default Value	Description
*FlowControl	3	When Rx Pause is enabled, the receiving adapter generates a flow control frame when its received queue reaches a pre-defined limit. The flow control frame is sent to the sending adapter. When TX Pause is enabled, the sending adapter pauses the transmission if it receives a flow control frame from a link partner. The valid values are: • 0: Flow control is disabled • 1: Tx Flow control is Enabled • 2: Rx Flow control is enabled • 3: Rx & Tx Flow control is enabled
DeviceRxStallTimeout	1000 mSec	When the device is in stall state (congestion mode), after the configured period of having the device in such state expires (the maximum period is 8 sec), the device will disable the Flow Control mechanism. The valid values are: • Minimum: 0 • Maximum: 8000
DeviceRxStallWatermark	0 mSec	When the device is in "stall state" (congestion mode), after the configured period of having the device in such state expires (the maximum period is 8 sec), the device will declare the driver as stalled. The valid values are: • Minimum: 0 • Maximum: 8000

VMQ Options

This section describes the registry keys that are used to control the NDIS Virtual Machine Queue (VMQ). VMQ is supported by WinOF-2 and allows a performance boost for Hyper-V VMs.

For more details about VMQ please refer to Microsoft web site, http://msdn.microsoft.com/en-us/library/windows/hardware/ff571034(v=vs.85).aspx

Value Name	Default Value	Description
*VMQ	1	The support for the virtual machine queue (VMQ) features of the network adapter. The valid values are: • 1: enable • 0: disable
*RssOrVmqPreference	0	Specifies whether VMQ capabilities should be enabled instead of receive- side scaling (RSS) capabilities. The valid values are: • 0: Report RSS capabilities • 1: Report VMQ capabilities **Note**: This registry value is not exposed via the UI.

Value Name	Default Value	Description
*VMQVlanFiltering	1	Specifies whether the device enables or disables the ability to filter network packets by using the VLAN identifier in the media access control (MAC) header. The valid values are: • 0: disable • 1: enable

RoCE Options

This section describes the registry keys that are used to control RoCE mode.

Value Name	Default Value	Description
roce_mode	0 - RoCE	The RoCE mode. The valid values are: • 0 - RoCE • 4 - No RoCE **Note**: The default value depends on the WinOF package used.

SR-IOV Options

This section describes the registry keys that are used to control the NDIS Single Root I/O Virtualization (SR-IOV). The SR-IOV is supported by WinOF-2 and allows a performance boost for Hyper-V VMs.

For more details about the VMQ, please see Single Root I/O Virtualization (SR-IOV) on Microsoft website.

Value Name	Default Value	Description
*SRIOV	1	The support for the SR-IOV features of the network adapter. The valid values are: • 1: enable • 0: disable
*SriovPreferred	N/A (hidden)	A value that defines whether SR-IOV capabilities should be enabled instead of the virtual machine queue (VMQ), or receive side scaling (RSS) capabilities.
MaxFWPagesUsagePerVF	250000	This key sets the limitation for the maximum number of 4KB pages that the host could allocate for VFs resources. When set to 0, limitation is disabled. The minimum valid value (when it is not 0) is 17000. When a smaller value (and larger than 0) is configured, the driver will use 17000 instead of the configured value. **Note:** This key can be changed dynamically.

RDMA Registry Keys

The following section describes the registry keys that are only relevant to RDMA.

Value Name	Default Value	Description
EnableGuestRdma	1: Enabled	Able to prevent RDMA in the VF from the host. This feature is enabled by default in IPoIB. **Note:** This registry key cannot be changed in Bluefield 2 SmartNIC mode, the selected mode in this setup will be enabled.
MaxCMRetries	15	Maximum number of times that either party can re-send a REQ, REP, or DREQ message. After re-sending for the maximum number of times without a response, the sending party should then terminate the protocol by sending a REJ message indicating that it timed out.
RemoteCMResponseTimeout	16	Expressed as 4.096 microSec * 2 ^ cm_response_timewait, within which the CM message recipient shall transmit a response to the sender. **Valid values are:** 3-25
NetworkDirectAdminOnly	0	In case this key is set 1, only an Admin user can use the ND - NetworkDirect application. **Max value:** 1 This registry key can be found at: *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\mlx5\Parameters*
NdkFmrDedicatedQp	0: Disabled	Controls whether or not a separated QP is used for NDK fast-register operations.
EnableCmAntiSpoofing	0	In case this key is set 1, the CM will not accept connection requests which the source IP-in-IP header is different from the source IP-in-CM private date (if there is any difference, the connection will be refused). **Max value:** 1

Diagnostics Registry Keys

Dump Me Now (DMN) Registry Keys

The registry keys for the DMN feature are located at: *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn*

For further information on how to find the registry keys, refer to section Finding the Index Value of the Network Interface.

The following section describes the registry keys that configure the Dump Me Now feature (see section Dump Me Now (DMN)).

Value Name	Key Type	Description
DumpMeNowDirectory	REG_SZ	Path to the root directory in which the DMN places its dumps. The path should be provided in a kernel path style, which means prefixing the drive name with "\??\" (e.g. \??\C:\DMN_DIR). BDF will be added to specified name. (e.g. if specified directory name is \??\C:\DMN_DIR, then directory \??\C:\DMN_DIR-<b>-<d>-<f> will be created for Host and \??\C:\DMN_DIR-<b>-<d> for VF) **Default Value:** • Host: \Systemroot\temp\Mlx5_Dump_Me_Now-<b>-<d>-<f> • VF: \Systemroot\temp\Mlx5_Dump_Me_Now-<b>-<d>-0

Value Name	Key Type	Description
DumpMeNowTotalCount	REG_DWORD	The maximum number of allowed DMN dumps. Newer dumps beyond this number will override old ones. **Values:** [0,512] **Default Value:** 128 **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
DumpMeNowPreservedCount	REG_DWORD	Specifies the number of DMN dumps that will be reserved, and will never be overridden by newer DMN dump. **Values:** [0,512] **Default Value:** 8 **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
DumpMeNowDumpMask	0xFFFF	Mask that controls the allowed dumps by DumpMeNow (If applicable). • MST_DUMP = 1 << 0 • CORE_DUMP = 1 << 1 • ADAPTER_DUMP = 1 << 2 • PDDR_DUMP = 1 << 3 • MP_STATS_DUMP = 1 << 4 • MP_RESOURCE_DUMP = 1 << 5 • REGISTRY_DUMP = 1 << 6 • QoS_DUMP = 1 << 7 • IPoIB_DUMP = 1 << 8 • VMQoS_DUMP = 1 << 9 • FULL_DUMP = 0xFFFF Values: [0,0xFFFF] **Note:** This key can be changed dynamically.

⚠ Setting *DumpMeNowTotalCount* and *DumpMeNowPreservedCount* to *"0"* will disable the DMN feature.

ResourceDump Registry Keys

The following section describes the registry keys that configure the ResourceDump feature (see section Resource Dump).

Value Name	Key Type	Description
ResourceDumpEnable	REG_DWORD	• 0 - ResourceDump notifications are disabled • 1 - ResourceDump notifications are enabled Values: [0,1] Default Value: 0 **Note:** This key can be changed dynamically.
ResourceDumpQuotaTimeLimit	REG_DWORD	This key is used to manage the quota time in **seconds**, when the time passes this value, the quota count will be reset. This mechanism is to control how many events per the **"Key Value"** in seconds are allowed. Values: [1, 1048575] Default value: 3600 (1 hour) **Note:** This key can be changed dynamically.
ResourceDumpQuotaCount	REG_DWORD	Quota Count in the period of QuotaTimeLimit are allowed. Values: [1, 100] Default Value: 5 **Note:** This key can be changed dynamically

FwTrace Registry Keys

The following section describes the registry keys that configure the FwTrace feature (see section FwTrace).

Value Name	Key Type	Description
FwTracerEnabled	REG_DWORD	• 0 - FwTrace is disabled • 1 - FwTrace is enabled **Values:** [0,1] **Default Value:** 1 **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.
FwTracerBufferSize	REG_DWORD	FwTracer Buffer Size in Bytes. This value is rounded up to be equal to 2^N * 4096 bytes. **Values:** [0x2000, 0x200000] **Default Value:** 0x10000 **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

DevX Registry Keys

The following section describes the registry keys that configure the DevX feature (see section DevX Interface).

Value Name	Key Type	Description
DevxEnabled	REG_DWORD	Enables Devx support. • 0 - disabled • 1 - enabled **Default Value:** 0
DevxFsRules	REG_DWORD	Devx steering rules support (mask value). • 0 - Only the default IPV4/UDP DevX steering rule is supported • 8 - Add support for Unicast MAC DevX steering rule • 16 - Add support for IPV4/UDP with CVLAN DevX steering rule • 32 - Add support for promiscuous mode • 64 - Add support for IPv6 multicast • 128 - Add support for IPv6 and L4 protocol • 256 - Add support for Ethernet MAC and L4 protocol • 512 - Add support for IPv6 multicast and any IP version • 1024 - Add support for Ethernet MAC and any IP version • 2048 - Add support for L4 protocol • 4096 - Add support for any IP protocol • 8192 - Add support for IPv4 and TCP port with cvlan • 16384 - Add support for Ethernet MAC and VLAN • 32768 - Add support for IPv6 multicast and VLAN • 65536 - Add support for Ethernet MAC, VLAN and IP • 131072 - Add support for IPv6 multicast, VLAN and IP • 262144 - Add support for all multicast

Value Name	Key Type	Description
AllowPromiscVport	REG_DWORD	Allows promiscuous mode enablement for vPorts. • 0 - Not allowed • 1 - Allowed **Default Value:** 0 **Note:** This capability is not supported in BlueField DPU mode, VPORTs are controlled by the DPU side.

VF Monitoring Registry Keys

The following section describes the registry keys that configure the VF Monitoring feature (see section "VF Monitoring").

The keys are located in the driver key of the adapter, and they are all dynamic.

Value Name	Key Type	Description
VfCpuMonEnable	REG_DWORD	The wanted state of the feature. **Values:** [0,1] **Default Value:** 0 **Note:** The feature cannot be enabled if the firmware does not support it.
VfCpuMonBatchPeriodSec	REG_DWORD	The frequency of issuing of automatic batch request in seconds. **Values:** [0, 86400] **Default value:** 60 (i.e., once in minute) **Note:** The value '0' means "Stop issuing automatic requests".
VfCpuMonFwCpuUsageMax	REG_DWORD	The threshold for the `'FwCpuUsage'` counter, showing the VF CPU usage in percent. **Values:** [0, 100] **Default value:** 60
VfCpuMonErrors2FwMax	REG_DWORD	The threshold for 'Errors2FW' counter, showing the number of errors, handled by FW. **Values:** [0, 0xffffffff] **Default value:** 1000

Network Direct Interface

Network Direct is a user-mode programming interface specification for Remote Direct Memory Access (RDMA). RDMA is provided by RDMA-enabled network adapters. Because Network Direct is fabric agnostic, it can be used on InfiniBand, iWARP, and RoCE. Network Direct allows RDMA-enabled network interface card manufacturers to expose the RDMA functionality of their network adapters in Windows.

RDMA is a kernel bypass technique which makes it possible to transfer large amounts of data quite rapidly. Because the transfer is performed by the DMA engine on the network adapter, the CPU is not used for the memory movement, which frees the CPU to perform other work.

Network Direct is widely used for High-Performance Computing (HPC) applications in which computational workloads are distributed to large numbers of servers for parallel processing. In addition, various financial markets trading workloads also require extremely low latency and extremely high message rates, which RDMA can provide.

The Network Direct Interface (NDI) architecture provides application developers with a networking interface that enables zero-copy data transfers between applications, kernel-bypass I/O generation and completion processing, and one-sided data transfer operations. NDI is supported by Microsoft and is the recommended method to write an RDMA application. NDI exposes the advanced capabilities of the NVIDIA® networking devices and allows applications to leverage advances of RDMA. Both RoCE and InfiniBand (IB) can implement NDI.

For further information please refer to: http://msdn.microsoft.com/en-us/library/cc904397(v=vs.85).aspx

Test Running

To run the test, follow the steps below:

1. Connect two servers to NVIDIA® adapters.
2. Verify ping between the two servers.
3. Configure the RoCE version to be:
   - Linux side - V2
   - Windows side - V2
   - Verify that ROCE udp_port is the same on the two servers. For the registry key, refer to RoCE Options section.
4. Select the server side and the client side, and run accordingly:
   - Server:

```
nd_rping/rping -s [-v -V -d] [-S size] [-C count] [-a addr] [-p port]
```

   - Client:

```
nd_rping/rping -c [-v -V -d] [-S size] [-C count] -a addr [-p port]
```

Executable Options:

Letter	Description
-s	Server side
-P	Persistent server mode allowing multiple connections
-c	Client side
-a	Address
-p	Port

Debug Extensions:

Letter	Description
-v	Displays ping data to stdout every test cycle
-V	Validates ping data every test cycle
-d	Shows debug prints to stdout
-S	Indicates ping data size - must be < (64*1024)
-C	Indicates the number of ping cycles to perform

Example:
Linux server:

```
rping -v -s -a <IP address> -C 10
```

Windows client:

```
nd_rping -v -c -a <same IP as above> -C 10
```

Using Network Direct with NVIDIA® Adapters

In order to use Network Direct with NVIDIA® adapters, NVIDIA® ND Provider should be installed in Windows. The tool can be used to remove, install and list OFA NetworkDirect providers.

Usage:

```
> ndinstall -h
```

where:

[-[i|r] [provider]]	Install/remove the specified/default providers. Provider must be one of the following names: • mlx4nd • mlx4nd2 • mlx5nd • mlx5nd2 • <blank> use the default ND providers
[-l]	List OFA ND providers
[-h]	This text

- Run "ndinstall -i" to install all available NVIDIA® ND Providers.

```
Installing mlx5nd provider: successful
Installing mlx5nd2 provider: successful

Current providers:
        0000001001 - Hyper-V RAW
        0000001006 - MSAFD Tcpip [TCP/IP]
        0000001007 - MSAFD Tcpip [UDP/IP]
        0000001008 - MSAFD Tcpip [RAW/IP]
        0000001009 - MSAFD Tcpip [TCP/IPv6]
        0000001010 - MSAFD Tcpip [UDP/IPv6]
        0000001011 - MSAFD Tcpip [RAW/IPv6]
        0000001016 - RSVP TCPv6 Service Provider
        0000001017 - RSVP TCP Service Provider
        0000001018 - RSVP UDPv6 Service Provider
        0000001019 - RSVP UDP Service Provider
        0000001055 - NDv1 Provider for Mellanox WinOF-2
        0000001056 - NDv2 Provider for Mellanox WinOF-2
```

- Run "ndinstall -l" to see a list of installed ND Providers:

```
Current providers:
        0000001001 - Hyper-V RAW
        0000001006 - MSAFD Tcpip [TCP/IP]
        0000001007 - MSAFD Tcpip [UDP/IP]
        0000001008 - MSAFD Tcpip [RAW/IP]
        0000001009 - MSAFD Tcpip [TCP/IPv6]
        0000001010 - MSAFD Tcpip [UDP/IPv6]
        0000001011 - MSAFD Tcpip [RAW/IPv6]
        0000001016 - RSVP TCPv6 Service Provider
        0000001017 - RSVP TCP Service Provider
        0000001018 - RSVP UDPv6 Service Provider
        0000001019 - RSVP UDP Service Provider
```

```
0000001055 - NDv1 Provider for Mellanox WinOF-2
0000001056 - NDv2 Provider for Mellanox WinOF-2
```

In the example above you can see that NDv1 and NDv2 NVIDIA® Providers are installed.

Performance Tuning

This section describes how to modify Windows registry parameters in order to improve performance.

> ⚠ Modifying the registry incorrectly might lead to serious problems, including the loss of data, system hang, and you may need to reinstall Windows. As such it is recommended to backup the registry on your system before implementing recommendations included in this section. If the modifications you apply lead to serious problems, you will be able to restore the original registry state. For more details about backing up and restoring the registry, please visit www.microsoft.com.

General Performance Optimization and Tuning

To achieve the best performance for Windows, you may need to modify some of the Windows registries.

Registry Tuning

The registry entries that may be added/changed by this "General Tuning" procedure:

- Under *HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters*:
  - Disable TCP selective acks option for better CPU utilization:

Registry Key	Type	Value
SackOpts	REG_DWORD	0

- Under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\AFD\Parameters:
  - Enable fast datagram sending for UDP traffic:

Registry Key	Type	Value
FastSendDatagramThreshold	REG_DWORD	64K

- Under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Ndis\Parameters:
  - Set RSS parameters:

Registry Key	Type	Value
RssBaseCpu	REG_DWORD	1

Enable RSS

Enabling Receive Side Scaling (RSS) is performed by running the following command:

```
“netsh int tcp set global rss = enabled”
```

Improving Live Migration

In order to improve live migration over SMB direct performance, please set the following registry key to 0 and reboot the machine:

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\LanmanServer\Parameters\RequireSecuritySignature
```

Application Specific Optimization and Tuning

Ethernet Performance Tuning

The user can configure the Ethernet adapter by setting some registry keys. The registry keys may affect Ethernet performance.

To improve performance, activate the performance tuning tool as follows:

1. Start the "Device Manager" (open a command line window and enter: devmgmt.msc).
2. Open "Network Adapters".
3. Right click the relevant Ethernet adapter and select Properties.
4. Select the "Advanced" tab
5. Modify performance parameters (properties) as desired.

Performance Known Issues

- On Intel I/OAT supported systems, it is highly recommended to install and enable the latest I/OAT driver (download from www.intel.com).
- With I/OAT enabled, sending 256-byte messages or larger will activate I/OAT. This will cause a significant latency increase due to I/OAT algorithms. On the other hand, throughput will increase significantly when using I/OAT.

Ethernet Bandwidth Improvements

To improve Ethernet Bandwidth:

1. Check you are running on the closest NUMA.

a. In the PowerShell run: Get-NetAdapterRss -Name "adapter name"

```
PS C:\Users\herod> Get-NetAdapterrss -Name "ethernet 6"


Name                                        : Ethernet 6
InterfaceDescription                        : Mellanox ConnectX-4 Lx Ethernet Adapter
Enabled                                     : True
NumberOfReceiveQueues                       : 8
Profile                                     : Closest
BaseProcessor: [Group:Number]               : 0:0
MaxProcessor: [Group:Number]                : 0:27
MaxProcessors                               : 8
RssProcessorArray: [Group:Number/NUMA Distance] : 0:0/0  0:1/0  0:2/0  0:3/0  0:4/0  0:5/0  0:6/0  0:7/0
                                                  0:8/0  0:9/0  0:10/0  0:11/0  0:12/0  0:13/0  0:14/32767  0:15/32767
                                                  0:16/32767  0:17/32767  0:18/32767  0:19/32767  0:20/32767
                                                  0:21/32767  0:22/32767  0:23/32767
                                                  0:24/32767  0:25/32767  0:26/32767  0:27/32767
IndirectionTable: [Group:Number]            : 0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
                                              0:0    0:1    0:2    0:3    0:4    0:5    0:6    0:7
```

b. Validate that the *IndirectionTable* CPUs are located at the closest NUMA.
As illustrated in the figure above, the CPUs are 0:0 - 0:7, CPU 0 -7 and the distance from the NUMA is 0, 0:0/0 - 0:7/0, unlike CPU 14-27/32767.

c. If the CPUs are not close to the NUMEA, change the *"RSS Base Processor Number"* and *"RSS Max Processor Number"* settings under the Advance tab to point to the closest CPUs.

> ⚠ For high performance, it is recommended to work with at least 8 processors.

2. Check the Ethernet bandwidth, run ntttcp.exe.

- Server side: ntttcp -r -m 32,*,server_ip
- Client side: ntttcp -s -m 32,*,server_ip

IPoIB Performance Tuning

The user can configure the IPoIB adapter by setting some registry keys. The registry keys may affect IPoIB performance.

➢ *To improve performance, activate the performance tuning tool as follows:*

1. Start the "Device Manager" (open a command line window and enter: devmgmt.msc).
2. Open "Network Adapters".
3. Right click the relevant IPoIB adapter and select Properties.
4. Select the "Advanced" tab
5. Modify performance parameters (properties) as desired.

Tunable Performance Parameters

The following is a list of key parameters for performance tuning.

Parameter	Description	Additional Options
Jumbo Packet	The maximum available size of the transfer unit, also known as the Maximum Transmission Unit (MTU). The MTU of a network can have a substantial impact on performance. A 4K MTU size improves performance for short messages, since it allows the OS to coalesce many small messages into a large one. Valid MTU values range for an Ethernet driver is between 614 and 9614. **Note:** All devices on the same physical network, or on the same logical network, must have the same MTU. This is applicable to the SoC MTU when using BlueField devices as well.	-
Receive Buffers	The number of receive buffers (default 512).	-
Send Buffers	The number of sent buffers (default 2048).	-
Performance Options	Configures parameters that can improve adapter performance.	**Interrupt Moderation** Moderates or delays the interrupts' generation. Hence, optimizes network throughput and CPU utilization (default Enabled). • When the interrupt moderation is enabled, the system accumulates interrupts and sends a single interrupt rather than a series of interrupts. An interrupt is generated after receiving 5 packets or after 10ms from the first packet received. It improves performance and reduces CPU load however, it increases latency. • When the interrupt moderation is disabled, the system generates an interrupt each time a packet is received or sent. In this mode, the CPU utilization data rates increase, as the system handles a larger number of interrupts. However, the latency decreases as the packet is handled faster.

Parameter	Description	Additional Options
		Receive Side Scaling (RSS Mode) Improves incoming packet processing performance. RSS enables the adapter port to utilize the multiple CPUs in a multi-core system for receiving incoming packets and steering them to the designated destination. RSS can significantly improve the number of transactions, the number of connections per second, and the network throughput. This parameter can be set to one of the following values: • Enabled (default): Set RSS Mode • Disabled: The hardware is configured once to use the Toeplitz hash function, and the indirection table is never changed. **Note:** I/OAT is not used while in RSS mode.
		Receive Completion Method Sets the completion methods of the received packets, and can affect network throughput and CPU utilization. • Polling Method Increases the CPU utilization as the system polls the received rings for the incoming packets. However, it may increase the network performance as the incoming packet is handled faster. • Adaptive (Default Settings) A combination of the interrupt and polling methods dynamically, depending on traffic type and network usage. Choosing a different setting may improve network and/or system performance in certain configurations.
		Rx Interrupt Moderation Type Sets the rate at which the controller moderates or delays the generation of interrupts making it possible to optimize network throughput and CPU utilization. The default setting (Adaptive) adjusts the interrupt rates dynamically depending on the traffic type and network usage. Choosing a different setting may improve network and system performance in certain configurations.
		Send Completion Method Sets the completion methods of the Send packets and it may affect network throughput and CPU utilization.

Parameter	Description	Additional Options
Offload Options	Allows you to specify which TCP/IP offload settings are handled by the adapter rather than the operating system. Enabling offloading services increases transmission performance as the offload tasks are performed by the adapter hardware rather than the operating system. Thus, freeing CPU resources to work on other tasks.	**IPv4 Checksums Offload** Enables the adapter to compute IPv4 checksum upon transmit and/or receive instead of the CPU (default Enabled).
		TCP/UDP Checksum Offload for IPv4 packets Enables the adapter to compute TCP/UDP checksum over IPv4 packets upon transmit and/or receive instead of the CPU (default Enabled).
		TCP/UDP Checksum Offload for IPv6 packets Enables the adapter to compute TCP/UDP checksum over IPv6 packets upon transmit and/or receive instead of the CPU (default Enabled).
		Large Send Offload (LSO) Allows the TCP/UDP stack to build a TCP/UDP message up to 64KB long and sends it in one call down the stack. The adapter then re-segments the message into multiple TCP/UDP packets for transmission on the wire with each pack sized according to the MTU. This option offloads a large amount of kernel processing time from the host CPU to the adapter.

Adapter Cards Counters

Adapter cards counters are used to provide information on Operating System, application, service or the drivers' performance. Counters can be used for different system debugging purposes, help to determine system bottlenecks and fine-tune system and application performance. The Operating System, network, and devices provide counter data that the application can consume to provide users with a graphical view of the system’s performance quality.

WinOF-2 counters hold the standard Windows CounterSet API that includes:

- Network Interface
- RDMA activity
- SMB Direct Connection

Mellanox WinOF-2 Port Traffic

Mellanox WinOF-2 Port Traffic counters set consists of counters that measure the rates at which bytes and packets are sent and received over a port network connection. It includes counters that monitor connection errors.

Mellanox WinOF-2 Port Traffic	Description
Bytes/Packets IN	
Bytes Received	Shows the number of bytes received by network adapter. The counted bytes include framing characters.

KBytes Received/Sec	Shows the rate at which kilobytes are received by a network adapter. The counted kilobytes include framing characters.
Packets Received	Shows the number of packets received by a network interface.
Packets Received/Sec	Shows the rate at which packets are received by a network interface.
Packets Received Frame too long Error	The number of received packets on a physical port dropped due to a large MTU size.
Packets Received Unsupported opcode Error	The number of MAC control packets received on a physical port with unsupported opcode.
Packets Received Frame undersize Error	The number of received packets on a physical port dropped due to the length of the packet being shorter than 64 bytes.
Packets Received Fragments Error	The number of received packets on a physical port dropped due to the length of the packet being shorter than 64 bytes and have FCS error.
Packets Received jabbers Error	The number of received packets on a physical port dropped due to the length of the packet being longer than 64 bytes and have FCS error.
Bytes/Packets OUT	
Bytes Sent	Shows the number of bytes sent by a network adapter. The counted bytes include framing.
KBytes Sent/Sec	Shows the rate at which kilobytes are sent by a network adapter. The counted kilobytes include framing characters.
Packets Sent	Shows the number of packets sent by a network interface.
Packets Sent/Sec	Shows the rate at which packets are sent by a network interface.
Bytes Total	Shows the total of bytes handled by a network adapter. The counted bytes include framing characters.
KBytes Total/Sec	Shows the total rate of kilobytes that are sent and received by a network adapter. The counted kilobytes include framing characters.
Packets Total	Shows the total of packets handled by a network interface.
Packets Total/Sec	Shows the rate at which packets are sent and received by a network interface.
Control Packets	The total number of successfully received control frames. **Note:** This counter is relevant only for ETH ports
ERRORS, DISCARDED	
Packets Received Frame too long Error	The number of received packets on a physical port dropped due to a large MTU size. **Note:** This counter is relevant only for ETH ports
Packets Received Unsupported opcode Error	The number of MAC control packets received on a physical port with unsupported opcode. **Note:** This counter is relevant only for ETH ports
Packets Received Frame undersize Error	The number of received packets on a physical port dropped due to the length of the packet being shorter than 64 bytes. **Note:** This counter is relevant only for ETH ports
Packets Received Fragments Error	The number of received packets on a physical port dropped due to the length of the packet being shorter than 64 bytes and have FCS error. **Note:** This counter is relevant only for ETH ports
Packets Received jabbers Error	The number of received packets on a physical port dropped due to the length of the packet being longer than 64 bytes and have FCS error. **Note:** This counter is relevant only for ETH ports
Packets Outbound Errors	Shows the number of outbound packets that could not be transmitted because of errors found in the physical layer.

Packets Outbound Discarded	Shows the number of outbound packets to be discarded in the physical layer, even though no errors had been detected to prevent transmission. One possible reason for discarding packets could be to free up buffer space.
Packets Received Errors	Shows the number of inbound packets that contained errors in the physical layer, preventing them from being deliverable.
Packets Received Frame Length Error	Shows the number of inbound packets that contained error where the frame has length error. Packets received with frame length error are a subset of packets received errors. **Note:** This counter is relevant only for ETH ports
Packets Received Symbol Error	Shows the number of inbound packets that contained symbol error or an invalid block. Packets received with symbol error are a subset of packets received errors.
Packets Received Bad CRC Error	Shows the number of inbound packets that contained bad CRC error. Packets received with bad CRC error are a subset of packets received errors.
Packets Received Discarded	No Receive WQEs - Packets discarded due to no receive descriptors posted by driver or software.
RSC Aborts	Number of RSC abort events. That is, the number of exceptions other than the IP datagram length being exceeded. This includes the cases where a packet is not coalesced because of insufficient hard-ware resources. **Note:** This counter is relevant only for ETH ports
RSC Coalesced Events	Number of RSC Coalesced events. That is, the total number of packets that were formed from coalescing packets. **Note:** This counter is relevant only for ETH ports
RSC Coalesced Octets	Number of RSC Coalesced bytes. **Note:** This counter is relevant only for ETH ports
RSC Coalesced Packets	Number of RSC Coalesced Packets. **Note:** This counter is relevant only for ETH ports
RSC Average Packet Size	RSC Average Packet Size is the average size in bytes of received packets across all TCP connections. **Note:** This counter is relevant only for ETH ports

Mellanox WinOF-2 VF Port Traffic

⚠ Mellanox WinOF2 VF Port Traffic counters exist per each VF and are created according to the adapter's configurations. These counters are created upon VFs configuration even if the VFs are not up.

Mellanox WinOF-2 VF Port Traffic counters set consists of counters that measure the rates at which bytes and packets are sent and received over a virtual port network connection that is bound to a virtual PCI function. It includes counters that monitor connection errors.

This set is available only on hypervisors and not on virtual network adapters.

⚠ These counters set is relevant only for ETH ports.

Mellanox WinOF-2 VF Port Traffic	Description
Bytes/Packets IN	
Bytes Received/Sec	Shows the rate at which bytes are received over each network VPort. The counted bytes include framing characters.
Bytes Received Unicast/Sec	Shows the rate at which subnet-unicast bytes are delivered to a higher-layer protocol.
Bytes Received Broadcast/ Sec	Shows the rate at which subnet-broadcast bytes are delivered to a higher-layer protocol.
Bytes Received Multicast/Sec	Shows the rate at which subnet-multicast bytes are delivered to a higher-layer protocol.
Packets Received Unicast/ Sec	Shows the rate at which subnet-unicast packets are delivered to a higher-layer protocol.
Packets Received Broadcast/ Sec	Shows the rate at which subnet-broadcast packets are delivered to a higher-layer protocol.
Packets Received Multicast/ Sec	Shows the rate at which subnet-multicast packets are delivered to a higher-layer protocol.
Bytes/Packets OUT	
Bytes Sent/Sec	Shows the rate at which bytes are sent over each network VPort. The counted bytes include framing characters.
Bytes Sent Unicast/Sec	Shows the rate at which bytes are requested to be transmitted to subnet-unicast addresses by higher-level protocols. The rate includes the bytes that were discarded or not sent.
Bytes Sent Broadcast/Sec	Shows the rate at which bytes are requested to be transmitted to subnet-broadcast addresses by higher-level protocols. The rate includes the bytes that were discarded or not sent.
Bytes Sent Multicast/Sec	Shows the rate at which bytes are requested to be transmitted to subnet-multicast addresses by higher-level protocols. The rate includes the bytes that were discarded or not sent.
Packets Sent Unicast/Sec	Shows the rate at which packets are requested to be transmitted to subnet-unicast addresses by higher-level protocols. The rate includes the packets that were discarded or not sent.
Packets Sent Broadcast/Sec	Shows the rate at which packets are requested to be transmitted to subnet-broadcast addresses by higher-level protocols. The rate includes the packets that were discarded or not sent.
Packets Sent Multicast/Sec	Shows the rate at which packets are requested to be transmitted to subnet-multicast addresses by higher-level protocols. The rate includes the packets that were discarded or not sent.
ERRORS, DISCARDED	
Packets Outbound Discarded	Shows the number of outbound packets to be discarded even though no errors had been detected to prevent transmission. One possible reason for discarding a packet could be to free up buffer space.
Packets Outbound Errors	Shows the number of outbound packets that could not be transmitted because of errors.
Packets Received Discarded	Shows the number of inbound packets that were chosen to be discarded even though no errors had been detected to prevent their being deliverable to a higher-layer protocol. One possible reason for discarding such a packet could be to free up buffer space.
Packets Received Errors	Shows the number of inbound packets that contained errors preventing them from being deliverable to a higher-layer protocol.

Mac Anti-Spoofing Packets Discarded	Shows the number of packets discarded due to illegal mac address usage.
Mac Anti-Spoofing Bytes Discarded	Shows the number of bytes discarded due to illegal mac address usage.
Vlan Anti-Spoofing Packets Discarded	Shows the number of packets discarded due to illegal vlan usage.
Vlan Anti-Spoofing Bytes Discarded	Shows the number of bytes discarded due to illegal vlan usage.
Allowed EthType Anti-Spoofing Packets Discarded	Shows the number of packets discarded due to unallowed ether type usage.
Allowed EthType Anti-Spoofing Bytes Discarded	Shows the number of Bytes discarded due to unallowed ether type usage.
RDMA Bytes/Packets IN	
Rdma Packets Received Unicast/Sec	Shows the rate at which subnet-unicast rdma packets are delivered to a higher-layer protocol.
Rdma Packets Received Multicast/Sec	Shows the rate at which subnet-multicast rdma packets are delivered to a higher-layer protocol.
Rdma Bytes Received Unicast/Sec	Shows the rate at which subnet-unicast rdma bytes are delivered to a higher-layer protocol.
Rdma Bytes Received Multicast/Sec	Shows the rate at which subnet-multicast rdma bytes are delivered to a higher-layer protocol.
RDMA Bytes/Packets OUT	
Rdma Packets Sent Unicast/Sec	Shows the rate at which subnet-unicast rdma packets are sent by a higher-layer protocol.
Rdma Packets Sent Multicast/Sec	Shows the rate at which subnet-multicast rdma packets are sent by a higher-layer protocol.
Rdma Bytes Sent Unicast/Sec	Shows the rate at which subnet-unicast rdma bytes are sent by a higher-layer protocol.
Rdma Bytes Sent Multicast/Sec	Shows the rate at which subnet-multicast rdma bytes are sent by a higher-layer protocol.

Mellanox WinOF-2 Port QoS

Mellanox WinOF-2 Port QoS counters set consists of flow statistics per (VLAN) priority. Each QoS policy is associated with a priority. The counter presents the priority's traffic, pause statistic.

These counters set is relevant only for ETH ports.

Mellanox WinOF-2 QoS	Description
Bytes/Packets IN	
Bytes Received	The number of bytes received that are covered by this priority. The counted bytes include framing characters (modulo 2^64).
KBytes Received/Sec	The number of kilobytes received per second that are covered by this priority. The counted kilobytes include framing characters.

Packets Received	The number of packets received that are covered by this priority (modulo 2^64).
Packets Received/Sec	The number of packets received per second that are covered by this priority.
Packets Received Discarded	The number of outbound packets to be discarded in the physical layer even though no errors have been detected to prevent transmission. A possible reason for discarding packets could be to free up buffer space.
Bytes/Packets OUT	
Bytes Sent	The number of bytes sent that are covered by this priority. The counted bytes include framing characters (modulo 2^64).
KBytes Sent/Sec	The number of kilobytes sent per second that are covered by this priority. The counted kilobytes include framing characters.
Packets Sent	The number of packets sent that are covered by this priority (modulo 2^64).
Packets Sent/Sec	The number of packets sent per second that are covered by this priority.
Bytes and Packets Total	
Bytes Total	The total number of bytes that are covered by this priority. The counted bytes include framing characters (modulo 2^64).
KBytes Total/Sec	The total number of kilobytes per second that are covered by this priority. The counted kilobytes include framing characters.
Packets Total	The total number of packets that are covered by this priority (modulo 2^64).
Packets Total/Sec	The total number of packets per second that are covered by this priority.
PAUSE INDICATION	
Sent Pause Duration	The total time in microseconds that the peer port has been requested to pause.
Sent Pause Frames	The number of pause packets transmitted on priority p on a physical port. If this counter is increasing, it implies that the adapter is congested and cannot absorb the traffic coming from the network.
Received Pause Frames	The number of pause packets received with priority p on a physical port. If this counter is increasing, it implies that the network is congested and cannot absorb the traffic coming from the adapter.
Received Pause Duration	The total time in microseconds that the transmission of packets to the peer port have been paused.

RDMA Activity

RDMA Activity counters set consists of NDK performance counters. These performance counters allow you to track Network Direct Kernel (RDMA) activity, including traffic rates, errors, and control plane activity.

RDMA Activity	Description
RDMA Accepted Connections	The number of inbound RDMA connections established.
RDMA Active Connections	The number of active RDMA connections.
RDMA Completion Queue Errors	This counter is not supported, and always is set to zero.
RDMA Connection Errors	The number of established connections with an error before a consumer disconnected the connection.
RDMA Failed Connection Attempts	The number of inbound and outbound RDMA connection attempts that failed.

RDMA Activity	Description
RDMA Inbound Bytes/sec	The number of bytes for all incoming RDMA traffic. This includes additional layer two protocol overhead.
RDMA Inbound Frames/sec	The number, in frames, of layer two frames that carry incoming RDMA traffic.
RDMA Initiated Connections	The number of outbound connections established.
RDMA Outbound Bytes/sec	The number of bytes for all outgoing RDMA traffic. This includes additional layer two protocol overhead.
RDMA Outbound Frames/sec	The number, in frames, of layer two frames that carry outgoing RDMA traffic.

Mellanox WinOF-2 Congestion Control

Mellanox WinOF-2 Congestion Control counters set consists of counters that measure the DCQCN statistics over the network adapter.

These counters set is relevant only for ETH ports.

Mellanox WinOF-2 Congestion Control	Description
Notification Point	
Notification Point - CNPs Sent Successfully	Number of congestion notification packets (CNPs) successfully sent by the notification point.
Notification Point - RoCEv2 DCQCN Marked Packets	Number of RoCEv2 packets that were marked as congestion encountered.
Reaction Point	
Reaction Point - Current Number of Flows	Current number of Rate Limited Flows due to RoCEv2 Congestion Control.
Reaction Point - Ignored CNP Packets	Number of ignored congestion notification packets (CNPs).
Reaction Point - Successfully Handled CNP Packets	Number of congestion notification packets (CNPs) received and handled successfully.

Mellanox WinOF-2 Diagnostics

Mellanox WinOF-2 Diagnostics counters set consists of the following counters:

Mellanox WinOF-2 Diagnostics	Description
Reset Requests	Number of resets requested by NDIS.
Link State Change Events	Number of link status updates received from the hardware.
Link State Change Down Events	Number of events received from the hardware, where the link state was changed to down.
Minor Stall Watermark Reached	Number of times the device detected a stalled state for a period longer than device_stall_minor_watermark. **Note:** This counter is relevant only for ETH ports
Critical Stall Watermark Reached	Number of times the port detected a stalled state for a period longer than device_stall_critical_watermark. **Note:** This counter is relevant only for ETH ports

Mellanox WinOF-2 Diagnostics	Description
Head of Queue timeout Packet discarded	Number of packets discarded by the transmitter due to Head-Of-Queue Lifetime Limit timeout. **Note:** This counter is relevant only for ETH ports
Stalled State Packet discarded	Number of packets discarded by the transmitter due to TC in Stalled state. **Note:** This counter is relevant only for ETH ports
Requester CQEs flushed with error	Number of requester CQEs flushed with error flowing queue transition to error state.
Send queues priority	The total number of QP/SQ priority/SL update events.
Async EQ Overrun	The number of times an EQ mapped to Async events queue encountered overrun queue.
Completion EQ Overrun	The number of times an EQ mapped to Completion events queue encountered overrun queue.
Current Queues Under Processor Handle	The current number of queues that are handled by the processor due to an Async error (e.g. retry exceeded) or due to a CMD error (e.g. 2eer_qp cmd).
Total Queues Under Processor Handle	The total number of queues that are handled by the processor due to an Async error (e.g. retry exceeded) or due to a CMD error (e.g. 2eer_qp cmd),
Queued Send Packets	Number of send packets pending transmission due to hardware queues overflow.
Send Completions in Passive/Sec	Number of send completion events handled in passive mode per second.
Receive Completions in Passive/Sec	Number of receive completion events handled in passive mode per second.
Packets Received dropped due to Steering	Number of packets that completed the NIC Receive FlowTable steering and were discarded due to lack of match rule in Flow Table.
Copied Send Packets	Number of send packets that were copied in slow path.
Correct Checksum Packets In Slow Path	Number of receive packets that required the driver to perform the checksum calculation and resulted in success.
Bad Checksum Packets In Slow Path	Number of receive packets that required the driver to perform checksum calculation and resulted in failure.
Undetermined Checksum Packets In Slow Path	Number of receive packets with undetermined checksum result.
Watch Dog Expired/Sec	Number of watch dogs expired per second.
Requester timeout received	Number of timeout received when the local machine generates outbound traffic.
Requester out of order sequence NAK	Number of Out of Sequence NAK received when the local machine generates outbound traffic, i.e. the number of times the local machine received NAKs indicating OOS on the receiving side.
Requester RNR NAK	Number of RNR (Receiver Not Ready) NAKs received when the local machine generates outbound traffic.
Responder RNR NAK	Number of RNR (Receiver Not Ready) NAKs sent when the local machine receives inbound traffic.
Responder out of order sequence received	Number of Out of Sequence packets received when the local machine receives inbound traffic, i.e. the number of times the local machine received messages that are not consecutive.
Responder duplicate request received	Number of duplicate requests received when the local machine receives inbound traffic.

Mellanox WinOF-2 Diagnostics	Description
Requester RNR NAK retries exceeded errors	Number of RNR (Receiver Not Ready) NAKs retries exceeded errors when the local machine generates outbound traffic.
Responder Local Length Errors	Number of times the responder detected local length errors
Requester Local Length Errors	Number of times the requester detected local length errors
Responder Local QP Operation Errors	Number of times the responder detected local QP operation errors
Local Operation Errors (a.k.a Requester Local QP Operation Errors)	Number of times the requester detected local QP operation errors
Responder Local Protection Errors	Number of times the responder detected memory protection error in its local memory subsystem
Requester Local Protection Errors	Number of times the requester detected a memory protection error in its local memory subsystem
Responder CQEs with Error	Number of times the responder flow reported a completion with error
Requester CQEs with Error	Number of times the requester flow reported a completion with error
Responder CQEs Flushed with Error	Number of times the responder flow completed a work request as flushed with error
Requester CQEs Flushed with Error	Number of times the requester completed a work request as flushed with error
Requester Memory Window Binding Errors	Number of times the requester detected memory window binding error
Requester Bad Response	Number of times an unexpected transport layer opcode was returned by the responder
Requester Remote Invalid Request Errors	Number of times the requester detected remote invalid request error
Responder Remote Invalid Request Errors	Number of times the responder detected remote invalid request error
Requester Remote Access Errors	Number of times the requester detected remote access error
Responder Remote Access Errors	Number of times the responder detected remote access error
Requester Remote Operation Errors	Number of times the requester detected remote operation error
Requester Retry Exceeded Errors	Number of times the requester detected transport retries exceed error
CQ Overflow	Counts the QPs attached to a CQ with overflow condition
Received RDMA Write requests	Number of RDMA write requests received
Received RDMA Read requests	Number of RDMA read requests received
Implied NAK Sequence Errors	Number of times the Requester detected an ACK with a PSN larger than the expected PSN for an RDMA READ or ATOMIC response. The QP retry limit was not exceeded
Dropless Mode Entries	The number of times entered dropless mode.
Dropless Mode Exits	The number of times exited dropless mode.
Transmission Engine Hang Events	The number of sx execution engine hang events.
MTT Entries Used For QP	Number of Memory Translation Table (MTT) entries used for QPs.
MTT Entries Used For CQ	Number of Memory Translation Table (MTT) entries used for CQs.
MTT Entries Used For EQ	Number of Memory Translation Table (MTT) entries used for EQs.
MTT Entries Used For MR	Number of Memory Translation Table (MTT) entries used for MRs.
CPU MEM-Pages (4K) Mapped By TPT For QP	Total number of CPU memory pages (4K) mapped by TPT for QPs.
CPU MEM-Pages (4K) Mapped By TPT For CQ	Total number of CPU memory pages (4K) mapped by TPT for CQs.

Mellanox WinOF-2 Diagnostics	Description
CPU MEM-Pages (4K) Mapped By TPT For EQ	Total number of CPU memory pages (4K) mapped by TPT for EQs.
CPU MEM-Pages (4K) Mapped By TPT For MR	Total number of CPU memory pages (4K) mapped by TPT for MRs.
Quota Exceeded Command	Number of commands issued by the VF and failed due to quota being exceeded.
Send Queue Priority Update Flow	The total number of QP/SQ priority/SL update events.
CQ Overrun	Number of times a CQ entered an error state due to overflow. Overflow occurs when the device tries to post a CQE into a full CQ buffer.

Mellanox WinOF-2 Diagnostics Ext 1

Mellanox WinOF-2 Diagnostics Ext 1 counters set consists of the following counters:

Mellanox WinOf-2 Diagnostics Ext 1	Description
RoCE Adaptive Retransmission	The number of adaptive retransmissions for RoCE traffic.
RoCE adaptive retransmission timeouts	The number of times RoCE traffic reached timeout due to adaptive retransmission.
RoCE Slow Restart	The number of times RoCE slow restart option was used.
RoCE Slow Restart CNPs	The number of times RoCE slow restart generated CNP packets.
RoCE Slow Restart Transmission	The number of times RoCE slow restart changed its state to slow restart.
Checksum calculated by SW/Packet	The number of times SW has calculated the checksum.
CQ Overrun	Number of times a CQ entered an error state due to overflow. Overflow occurs when the device tries to post a CQE into a full CQ buffer.
CM DREQ	RDMA disconnect by peer
Generated Packets dropped due to steering failure	Number of packets generated by the VNIC experiencing an unexpected steering failure (at any point in steering flow)
Handled Packets dropped due to steering failure	Number of packets handled by the VNIC experiencing an unexpected steering failure (at any point in steering flow owned by the VNIC, including the FDB for the eSwitch owner)

Mellanox WinOf-2 SW Backchannel Diagnostics

Mellanox WinOF-2 SW Backchannel Diagnostics counters set consists of the following counters:

Mellanox WinOf-2 SW Backchannel Diagnostics	Description
Supported Capabilities Bitmask	Bitmask of capabilities supported by VF
Currently Active Capabilities Bitmask	Bitmask of capabilities currently activated for VF
Read Config Block OIDs/Sec	The number of OID_SRIOV_READ_VF_CONFIG_BLOCK received per second
Write Config Block OIDs/Sec	The number of OID_SRIOV_WRITE_VF_CONFIG_BLOCK received per second
Illegal Or Unsupported Read Config Block OIDs	The number of OID_SRIOV_READ_VF_CONFIG_BLOCK detected as illegal or unsupported
Illegal Or Unsupported Write Config Block OIDs	The number of OID_SRIOV_WRITE_VF_CONFIG_BLOCK detected as illegal or unsupported
Read Config Block OIDs Failed To Apply	The number of OID_SRIOV_READ_VF_CONFIG_BLOCK returned with fail status **Note:** It does not necessary indicates error.
Write Config Block OIDs Failed To Apply	The number of OID_SRIOV_WRITE_VF_CONFIG_BLOCK returned with fail status. **Note:** It does not necessary indicates error

Mellanox WinOF-2 Device Diagnostic

Mellanox WinOF-2 Device Diagnostic counters are global for the device used. Therefore, all the adapter cards associated with the device will have the same counters' values.

Mellanox WinOF-2 Device Diagnostic counters set consists of the following counters:.

Mellanox WinOF-2 Device Diagnostics	Description
L0 MTT miss	The number of access to L0 MTT that were missed
L0 MTT miss/Sec	The rate of access to L0 MTT that were missed
L0 MTT hit	The number of access to L0 MTT that were hit
L0 MTT hit/Sec	The rate of access to L0 MTT that were hit
L1 MTT miss	The number of access to L1 MTT that were missed
L1 MTT miss/Sec	The rate of access to L1 MTT that were missed
L1 MTT hit	The number of access to L1 MTT that were hit
L1 MTT hit/Sec	The rate of access to L1 MTT that were hit
L0 MPT miss	The number of access to L0 MKey that were missed
L0 MPT miss/Sec	The rate of access to L0 MKey that were missed
L0 MPT hit	The number of access to L0 MKey that were hit
L0 MPT hit/Sec	The rate of access to L0 MKey that were hit
L1 MPT miss	The number of access to L1 MKey that were missed
L1 MPT miss/Sec	The rate of access to L1 MKey that were missed
L1 MPT hit	The number of access to L1 MKey that were hit
L1 MPT hit/Sec	The rate of access to L1 MKey that were hit

Mellanox WinOF-2 Device Diagnostics	Description
RXS no slow path credits	No room in RXS for slow path packets
RXS no fast path credits	No room in RXS for fast path packets
RXT no slow path credits	No room in RXT for slow path packets
RXT no fast path credits	No room in RXT for fast path packets
Slow path packets slice load	Number of slow path packets loaded to HCA as slices from the network
Fast path packets slice load	Number of fast path packets loaded to HCA as slices from the network
Steering pipe 0 processing time	Number of clocks that steering pipe 0 worked
Steering pipe 1 processing time	Number of clocks that steering pipe 1 worked
WQE address translation back-pressure	No credits between RXW and TPT
Receive WQE cache miss	Number of packets that got miss in RWqe buffer L0 cache
Receive WQE cache hit	Number of packets that got hit in RWqe buffer L0 cache
Slow packets miss in LDB L1 cache	Number of slow packet that got missed in LDB L1 cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Slow packets hit in LDB L1 cache	Number of slow packet that got hit in LDB L1 cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Fast packets miss in LDB L1 cache	Number of fast packet that got missed in LDB L1 cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Fast packets hit in LDB L1 cache	Number of fast packet that got hit in LDB L1 cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Packets miss in LDB L2 cache	Number of packet that got missed in LDB L2 cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Packets hit in LDB L2 cache	Number of packet that got hit in LDB L2 cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Slow packets miss in REQSL L1	Number of slow packet that got missed in REQSL L1 fast cache
Slow packets hit in REQSL L1	Number of slow packet that got hit in REQSL L1 fast cache
Fast packets miss in REQSL L1	Number of fast packet that got missed in REQSL L1 fast cache
Fast packets hit in REQSL L1	Number of fast packet that got hit in REQSL L1 fast cache
Packets miss in REQSL L2	Number of packet that got missed in REQSL L2 fast cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
Packets hit in REQSL L2	Number of packet that got hit in REQSL L2 fast cache **Note:** Supported only in ConnectX-4 Lx adapter cards.
No PXT credits time	Number of clocks in which there were no PXT credits
EQ slices busy time	Number of clocks where all EQ slices were busy
CQ slices busy time	Number of clocks where all CQ slices were busy
MSIX slices busy time	Number of clocks where all MSIX slices were busy
QP done due to VL limited	Number of QP done scheduling due to VL limited (e.g. lack of VL credits)
QP done due to desched	Number of QP done scheduling due to de-scheduling (Tx full burst size)
QP done due to work done	Number of QP done scheduling due to work done (Tx all QP data)
QP done due to limited	Number of QP done scheduling due to limited rate (e.g. max read)
QP done due to E2E credits	Number of QP done scheduling due to e2e credits (other peer credits)
Packets sent by SXW to SXP	Number of packets that were authorized to send by SXW (to SXP)
Steering hit	Number of steering lookups that were hit **Note:** Supported only in ConnectX-4 Lx adapter cards.

Mellanox WinOF-2 Device Diagnostics	Description
Steering miss	Number of steering lookups that were miss **Note:** Supported only in ConnectX-4 Lx adapter cards.
Steering processing time	Number of clocks that steering pipe worked **Note:** Supported only in ConnectX-4 Lx adapter cards.
No send credits for scheduling time	The number of clocks that were no credits for scheduling (Tx)
No slow path send credits for scheduling time	The number of clocks that were no credits for scheduling (Tx) for slow path **Note:** Supported only in ConnectX-4 Lx adapter cards.
TPT indirect memory key access	The number of indirect mkey accesses
Internal RQ out of buffer	Number of times the device that owned the queue had insufficient number of buffers allocated
Nic temperature in Celsius degrees unit	The temperature of the NIC in Celsius degrees unit

Mellanox WinOF-2 PCI Device Diagnostic

Mellanox WinOF-2 Device Diagnostic counters are global for the device used. Therefore, all the adapter cards associated with the device will have the same counters' values.

Mellanox WinOF-2 PCI Device Diagnostic counters set consists of the following counters:

Mellanox WinOF-2 PCI Device Diagnostic	Description
PCI back-pressure cycles	The number of clocks where BP was received from the PCI, while trying to send a packet to the host.
PCI back-pressure cycles/Sec	The rate of clocks where BP was received from the PCI, while trying to send a packet to the host.
PCI write back-pressure cycles	The number of clocks where there was lack of posted outbound credits from the PCI, while trying to send a packet to the host.
PCI write back-pressure cycles/ Sec	The rate of clocks where there was lack of posted outbound credits from the PCI, while trying to send a packet to the host.
PCI read back-pressure cycles	The number of clocks where there was lack of non-posted outbound credits from the PCI, while trying to send a packet to the host.
PCI read back-pressure cycles/ Sec	The rate of clocks where there was lack of non-posted outbound credits from the PCI, while trying to send a packet to the host.
PCI read stuck no receive buffer	The number of clocks where there was lack in global byte credits for non-posted outbound from the PCI, while trying to send a packet to the host. **Note:** Supported only in ConnectX-4 Lx adapter cards.
Available PCI BW/Sec	The number (per seconds) of 128 bytes that are available by the host.
Used PCI BW//Sec	The number (per seconds) of 128 bytes that were received from the host.
Available PCI BW	**[Deprecated]** The number of 128 bytes that are available by the host.
Used PCI BW	**[Deprecated]** The number of 128 bytes that were received from the host.

Mellanox WinOF-2 PCI Device Diagnostic	Description
RX PCI errors	The number of physical layer PCIe signal integrity errors. The number of transitions to recovery due to Framing errors and CRC (dlp and tlp). If the counter is advancing, try to change the PCIe slot in use. Note: Only a continues increment of the counter value is considered an error.
TX PCI errors	The number of physical layer PCIe signal integrity errors. The number of transition to recovery initiated by the other side (moving to Recovery due to getting TS/EIEOS). If the counter is advancing, try to change the PCIe slot in use. Note: transitions to recovery can happen during initial machine boot. The counter should not increment after boot. Note: Only a continues increment of the counter value is considered an error.
TX PCI non-fatal errors	The number of PCI transport layer Non-Fatal error msg sent. If the counter is advancing, try to change the PCIe slot in use.
TX PCI fatal errors	The number of PCIe transport layer fatal error msg sent. If the counter is advancing, try to change the PCIe slot in use.
PCI link width the current width of PCIe link	In order to get the overall PCIe bandwidth, the PCI link width should be multiply by PCI link speed.
PCI link speed the current speed of PCIe link	In order to get the overall PCIe bandwidth, the PCI link speed should be multiply by PCI link width.
RX Packet Drops PCIe Buffers	Number of packets dropped by Weighted Random Early Detection (WRED) function.
RX Packet Marked PCIe Buffers	Number of packets marked as ECN.

Mellanox WinOF-2 VF Diagnostics

Mellanox WinOF2 VF Diagnostics counters exist per each VF and are created according to the adapter's configurations. These counters are created upon VFs configuration even if the VFs are not up.

Mellanox WinOF2 VF Diagnostics counters set consists of VF diagnostic and debug counters. This set is available only on the hypervisors and not on the virtual network adapters:

Mellanox WinOF-2 VF Diagnostics	Description
Async EQ Overrun	The number of times an EQ mapped to Async events queue encountered overrun queue.
Completion EQ Overrun	The number of times an EQ mapped to Completion events queue encountered overrun queue.
Current Queues Under Processor Handle	The current number of queues that are handled by the processor due to an Async error (e.g. retry exceeded) or due to a CMD error (e.g. 2eer_qp cmd).
Total Queues Under Processor Handle	The total number of queues that are handled by the processor due to an Async error (e.g. retry exceeded) or due to a CMD error (e.g. 2eer_qp cmd).
Packets Received dropped due to Steering	Number of packets that completed the NIC Receive FlowTable steering and were discarded due to lack of match rule in Flow Table.
Packets Received dropped due to VPort Down	Number of packets that were steered to a VPort, and discarded because the VPort was not in a state to receive packets
Packets Transmitted dropped due to VPort Down	Number of packets that were transmitted by a vNIC, and discarded because the VPort was not in a state to transmit packets.
Invalid Commands	Number of commands issued by the VF and failed.
Quota Exceeded Command	Number of commands issued by the VF and failed due to quota exceeded.
Send Queue Priority Update Flow	The total number of QP/SQ priority/SL update events.
Packets Received WQE too small	The number of packets that reached the Ethernet RQ but cannot fit into the WQE due to their large size
CQ Overrun	Number of times CQs entered an error state due to overflow
Packets Received dropped due to lack of receive WQEs	Number of dropped packets due to lack of receive WQEs for an internal device RQs
Generated Packets dropped due to steering failure	Number of packets generated by the VNIC experiencing an unexpected steering failure (at any point in steering flow)
Handled Packets dropped due to steering failure	Number of packets handled by the VNIC experiencing an unexpected steering failure (at any point in steering flow owned by the VNIC, including the FDB for the eSwitch owner)

Mellanox WinOF-2 VF Internal Traffic

> ⚠ Mellanox WinOF-2 VF Internal Traffic Counters are relevant for Physical Functions ONLY.

Mellanox WinOF-2 VF Internal Traffic Counters set consists of counters that measure the rates at which bytes and packets are sent and received over each core of a virtual port that is bound to a virtual PCI function.

This set is available only on hypervisors, and each virtual network adapter should be allowed to update its counters by using the mlx5cmd tool.

> ⚠ The virtual network adapter driver should support internal traffic counter set exposure, to make it available on hypervisor.

These counters are relevant only for ETH ports.

Mellanox WinOF-2 VF Internal Traffic	Description
Receive Packets	The number of packets received by this virtual adapter at specific core.
Receive Octets	The number of bytes received by this virtual adapter at specific core. The counted bytes don't include framing characters (modulo 2^64)
Transmit Packets	The number of packets sent by this virtual adapter at specific core.
Transmit Octets	The number of bytes sent by this virtual adapter at specific core. The counted bytes don't include framing characters (modulo 2^64)

Controlling VF Internal Traffic

VF Internal Traffic Counters can be controlled using the mlx5cmd.exe tool. The tool enables the user to make the virtual network adapter's traffic counters per core available or unavailable for performance monitoring consumers.

Usage:	mlx5cmd.exe -VfStats -name <adapter> -vf <virtual function ID> [-register -rate <in 100 mSec.> | -unregister]
Detailed usage:	mlx5cmd.exe -VfStats -hh

Mellanox WinOF-2 Rss

These counters set is relevant only for ETH ports.

Mellanox WinOF-2 Rss counters may have performance impact when they are active.

Mellanox WinOF-2 Rss Counters set provides monitoring for hardware RSS behavior. These counters are accumulative and collect packets per type (IPv4 or IPv6 only, IPv4/6 TCP or UDP), for tunneled and non-tunneled traffic separately, and when the hardware RSS is functional or dysfunctional.

The counters are activated upon first addition into perfmon, and are stopped upon removal.

Setting "RssCountersActivatedAtStartup" registry key to 1 in the NIC properties will cause the Rss counters to collect data from the startup of the device.

All Rss counters are provided under the counter set “Mellanox Adapter Rss Counters”.

Each Ethernet adapter provides multiple instances:

- Instance per vPort per CPU in HwRSS mode is formatted: <NetworkAdapter> + vPort_<id> CPU_<cpu>
- Instance per network adapter per CPU in native Rss per CPU is formatted: <NetworkAdapter> CPU_<cpu>

Mellanox WinOF-2 Rss	Description
Number of interrupts	Number of interrupts generated to process RX completions.
Rss IPv4 Only	Shows the number of received packets that have RSS hash calculated on IPv4 header only
Rss IPv4/TCP	Shows the number of received packets that have RSS hash calculated on IPv4 and TCP headers
Rss IPv4/UDP	Shows the number of received packets that have RSS hash calculated on IPv4 and UDP headers
Rss IPv6 only	Shows the number of received packets that have RSS hash calculated on IPv6 header only
Rss IPv6/TCP	Shows the number of received packets that have RSS hash calculated on IPv6 and TCP headers
Rss IPv6/UDP	Shows the number of received packets that have RSS hash calculated on IPv6 and UDP headers
Encapsulated Rss IPv4 Only	Shows the number of received encapsulated packets that have RSS hash calculated on IPv4 header only
Encapsulated Rss IPv4/ TCP	Shows the number of received encapsulated packets that have RSS hash calculated on IPv4 and TCP headers
Encapsulated Rss IPv4/ UDP	Shows the number of received encapsulated packets that have RSS hash calculated on IPv4 and UDP headers
Encapsulated Rss IPv6 Only	Shows the number of received encapsulated packets that have RSS hash calculated on IPv6 header only
Encapsulated Rss IPv6/ TCP	Shows the number of received encapsulated packets that have RSS hash calculated on IPv6 and TCP headers
Encapsulated Rss IPv6/ UDP	Shows the number of received encapsulated packets that have RSS hash calculated on IPv6 and UDP headers
NonRss IPv4 Only	Shows the number of IPv4 packets that have no RSS hash calculated by the hardware
NonRss IPv4/TCP	Shows the number of IPv4 TCP packets that have no RSS hash calculated by the hardware
NonRss IPv4/UDP	Shows the number of IPv4 UDP packets that have no RSS hash calculated by the hardware
NonRss IPv6 Only	Shows the number of IPv6 packets that have no RSS hash calculated by the hardware
NonRss IPv6/TCP	Shows the number of IPv6 TCP packets that have no RSS hash calculated by the hardware
NonRss IPv6/UDP	Shows the number of IPv6 UDP packets that have no RSS hash calculated by the hardware
Encapsulated NonRss IPv4 Only	Shows the number of encapsulated IPv4 packets that have no RSS hash calculated by the hardware
Encapsulated NonRss IPv4/TCP	Shows the number of encapsulated IPv4 TCP packets that have no RSS hash calculated by the hardware
Encapsulated NonRss IPv4/UDP	Shows the number of encapsulated IPv4 UDP packets that have no RSS hash calculated by the hardware
Encapsulated NonRss IPv6 Only	Shows the number of encapsulated IPv6 packets that have no RSS hash calculated by the hardware
Encapsulated NonRss IPv6/TCP	Shows the number of encapsulated IPv6 TCP packets that have no RSS hash calculated by the hardware
Encapsulated NonRss IPv6/UDP	Shows the number of encapsulated IPv6 UDP packets that have no RSS hash calculated by the hardware

Mellanox WinOF-2 Rss	Description
Rss Misc	Shows the number of received packets that have RSS hash calculated with unknown RSS hash type
Encapsulated Rss Misc	Shows the number of received encapsulated packets that have RSS hash calculated with unknown RSS hash type
NonRss Misc	Shows the number of packets that have no RSS hash calculated by the hardware for no apparent reason
Encapsulated NonRss Misc	Shows the number of encapsulated packets that have no RSS hash calculated by the hardware for no apparent reason

Mellanox WinOF-2 Receive Datapath

Mellanox WinOF-2 Receive Datapath counters set provides queue counters per receive. These counters are available in Native, VMQ and SR-IOV mode. These counters provide visibility into the driver when running traffic. Each Ethernet adapter provides multiple instances. An instance per vPort per queue number is formatted as one of the below depending on the mode set (Native or VMQ/SR-IOV):

- <NetworkAdapter> + RqNum_<num>
- <NetworkAdapter> + vPort_<id> + RqNum_<num>

These counters set is relevant only for ETH ports.

Mellanox WinOF-2 Receive Datapath	Description
Cpu Number	The CPU where the driver process the queue completions.
Drops due to invalid packet size	Advanced when a packet is received with <A> size that is larger than the maximum MTU size allowed, which is the max size HW supports. The value can be checked using the NDIS miniport adapter general attributes struct in the field MTuSize.
Number of receive buffers posted	When this counter is not advancing, the SW/HW might be stuck. Meaning, either the SW is not processing the receive requests or the HW is not using the post receives. To check the state of WQ/CQ, check the error events log messages.
Average packet count per indicate	The average of the handled send packets per indicate calls to NDIS. The average is the number of packets completed /number of indicates to NDIS.
Packets in low resource mode	When a forced low resource (Regestry ForceLowResourcesIndication is 1, when the default is 0) or the number of outstanding post receive is lower than the minimum number of RFDs configured (Regestry is NicMinRfds).
Packets processed in interrupt mode	The number of packets indicated to NDIS during interrupt. The counter progresses as the argument “NumberOfNetBufferLists” in the function "NdisMIndicateReceiveNetBufferLists" progresses when it is called during interrupt handling.
Packets processed in polling mode	The number of packets indicated to NDIS while in polling mode.

Mellanox WinOF-2 Receive Datapath	Description
Consumed max receives	Number of times the driver processed the number of packets that is higher than the maximum calls to NDIS Indicate (the value shown in REgestry MaxCallsToNdisIndicate). When this counter progresses, the driver stops processing any more packets. **Note:** The counters “Packets processed in polling mode” and “Packets processed in interrupt mode” also progress accordingly.
Number of traffic profile transitions	Number of times the core’s Receive Queue changed traffic Latency/ Throughput.
DpcWatchDog (SingleDpc) Starvation	The number of times the driver had watchdog starvation during DPC and re-submitted a DPC. When this counter progresses, DPC does not process any packets, meaning counters 6-10 will not progress.
DpcWatchDog (TotalDpc) Starvation	The number of times the driver had watchdog starvation during DPC and moved to. When this counter progresses, DPC does not process any packets, meaning counters 6-10 will not progress.
Drops due to completion queue errors	The number of Receive Drops Due To Cqe Errors.
Interrupts on incorrect cpu	The number of received interrupts on a wrong CPU. In this case, the driver re-submits a DPC on the correct CPU.
Number of interrupts	Number of Receive Datapath interrupts.
Strided Wqes	The number of Wqes that its strides are consumed by the HW. They should progress only if StridingRQ feature is enabled (check in Regestry StridingRqEnabled). Counters: For every N > 0 packets received, the packetsCounter should be incremented by N. The wqe counter can be incremented by [upper bound(N/ number of strides in wqe) ,N].
Ecn Marked Packets (Ipv4)	The number of times the driver marked an IPv4 packet with ECN.
Ecn Marked Packets (Ipv6)	The number of times the driver marked an IPv6 packet with ECN.
Packets processed in NDIS poll mode	When the feature is enabled, counter for “Packets processed in Interrupt mode” or “Packets processed in poll mode” are not counters incremented.

Mellanox WinOF-2 Transmit Datapath

Mellanox WinOF-2 Transmit Datapath counters set provides queue counters per transmit. These counters are available in Native, VMQ and SR-IOV mode. These counters provide visibility into the driver when running traffic. Each Ethernet adapter provides multiple instances. An instance per vPort per queue number is formatted as one of the below depending on mode (Native or VMQ/SR-IOV):

- <NetworkAdapter> + SqNum_<num>
- <NetworkAdapter> + vPort_<id> + SqNum_<num>

⚠ These counters set is relevant only for ETH ports.

Mellanox WinOF-2 Transmit Datapath	Description
Cpu Number	The CPU where the driver process the queue completions.

Mellanox WinOF-2 Transmit Datapath	Description
Transmit ring is full	Counts the time the transmit ring was full during sends.
Transmit copy packets	Counts the number of times a packet should be copied during sends. This could happen in case a packet has a size larger than supported by the HW.
Number of packets posted	The number of send requests that have been forwarded to the HW, (packets that are pending aren't counted).
Number of packets completed	Counts the number of processed and completed sends, when it progress, the resources allocated to the sent packet is freed.
OS call to build SGL failed	The LSO header size cannot be received if SKB allocation fails or the packet has an invalid size.
Drops due to invalid packet size	Number of packets with invalid size, "OS call to build SGL failed" counter should also progress in this case.
Number of packets posted in bypass mode	Number of packets detected by driver as forwarded.
Average packet count per indicate	The average of the handled send packets per indicate calls to NDIS. The average is the number of packets completed /number of indicates to NDIS.
Interrupts on incorrect cpu	The number of times the TX received a completion on the wrong CPU. In such case, the driver re-submits a DCP on the correct CPU.
CQ Overrun	Number of times a CQ entered an error state due to overflow. Overflow occurs when the device tries to post a CQE into a full CQ buffer.
Drops due to completion queue errors	Drops due to completion queue errors
Number of traffic profile transitions	Number of traffic profile transitions
Packets processed in interrupt mode	Packets processed in interrupt mode
Packets processed in polling mode	Packets processed in polling mode
Packets processed in NDIS poll mode	Packets processed in NDIS poll mode

Mellanox WinOF-2 Port Diagnostics

Mellanox WinOF-2 Port Diagnostics counters set contains physical layer statistical counters. This set exists for every adapter in the PF, it is not supported in the VF.

Mellanox WinOF-2 Port Diagnostics	Description
RX Error Lane0 phy	The number error bits on lane 0
RX Error Lane0 phy/Sec	The rate of changing of the lane 0 counter
RX Error Lane1 phy	The number error bits on lane 1
RX Error Lane1 phy/Sec	The rate of changing of the lane 1 counter
RX Error Lane2 phy	The number error bits on lane 2
RX Error Lane2 phy/Sec	The rate of changing of the lane 2 counter
RX Error Lane3 phy	The number error bits on lane 3
RX Error Lane3 phy/Sec	The rate of changing of the lane 3 counter

Mellanox WinOF-2 Port Diagnostics	Description
RX Kbits phy	The total amount of traffic that could have been received on the port
RX Kbits phy/Sec	The rate of changing of the above counter
RX PCS Corrected Bits phy	The number of symbol errors that wasn't corrected by FEC correction algorithm or that FEC algorithm was not active on this interface
RX PCS Corrected Bits phy/ Sec	The rate of changing of the above counter
RX PCS Symbol Error phy	The number of corrected bits on this port according to active FEC (RS/FC). If this counter is increasing, it implies that the link between the NIC and the network is suffering from high BER
RX PCS Symbol Error phy/Sec	The rate of changing of the above counter

Mellanox WinOF-2 ICMC Diag Counters Ext1

Mellanox WinOF-2 Device Diagnostic counters are global for the device used. Therefore, all the adapter cards associated with the device will have the same counters' values.

Mellanox WinOF-2 ICMC Diag Counters Ext counters set contains information on interconnect contexts memory cache (ICMC).

Mellanox WinOF-2 ICMC Diag Counters Ext1	Description
ICMC QP Send Hit	Number of Internal cache hits for Send Queue Pair contexts
ICMC QP Send Miss	Number of Internal cache misses for Send Queue Pair contexts
ICMC QP Receive Hit	Number of Internal cache hits for Receive Queue Pair contexts
ICMC QP Receive Miss	Number of Internal cache misses for Receive Queue Pair contexts
ICMC SRQ Hit	Number of Internal cache hits for Shared Receive Queue contexts
ICMC SRQ Miss	Number of Internal cache misses for Shared Receive Queue contexts
ICMC CQ Hit	Number of Internal cache hits for Completion Queue contexts
ICMC CQ Miss	Number of Internal cache misses for Completion Queue contexts

Resiliency

Dump Me Now (DMN)

DMN generates dumps and traces from various components, including hardware, firmware and software, upon user requests, upon internally detected issues (by the resiliency sensors) and ND application requests via the extended NVIDIA® ND API.

DMN dumps are crucial for offline debugging. Once an issue is hit, the dumps can provide useful information about the NIC's state at the time of the failure. This includes hardware state dumps, firmware traces and various driver component state and resource dumps.

For information on the relevant registry keys for this feature, please refer to Dump Me Now (DMN) Registry Keys.

DMN Triggers and APIs

DMN supports three triggering APIs:

1. mlx5Cmd.exe can be used to trigger DMN by running the *-Dmn sub* command:

```
Mlx5Cmd -Dmn -hh | -Name <adapter name>
Submit dump-me-now request
```

Options:

-hh	Show this help screen
-Name <adapter name>	Network adapter name
-NoMstDump	Run DMN without mst dump
-CoreDumpQP<QP number>	Run DMN with QP Core Dump

2. ND SPI NVIDIA® extension (defined in ndspi_ext_mlx.h):
   a. API function to generate a general DMN dump from an ND application:

```
HRESULT
Nd2AdapterControlDumpMeNow(
__in IND2AdapterControl* pCtrl,
__in HANDLE hOverlappedFile,
__inout OVERLAPPED* pOverlapped
);
```

   b. API function to generate a QP based DMN dump from an ND application. The function generates a dump that might include more information about the queue pair specified by its number.

```
HRESULT
Nd2AdapterControlDumpQpNow(
__in IND2AdapterControl* pCtrl,
__in HANDLE hOverlappedFile,
__in ULONG Qpn,
__inout OVERLAPPED* pOverlapped
);
```

   c. An internal API between different driver components, in order to support generating DMN upon self-detected errors and failures (by the resiliency feature).

Dumps and Incident Folders

DMN generates a directory per incident, where it places all of the needed NIC dump files. There is a mechanism to limit the number of created Incident Directories. For further information, see Cyclic DMN Mechanism.

The DMN incident directory name includes a timestamp, dump type, DMN event source and reason. It uses the following directory naming scheme: *dmn-<type of DMN>-<source of DMN trigger>-<reason>-<timestamp>*

Example:

```
dmn-GN-USR-NA-4.13.2017-07.49.02.747
```

In this example:

- GN: The dump type is "General"
- USR: The DMN was triggered by mlx5Cmd (user)
- NA: In this version of the driver, the cause for the dump is not available in case of mlx5Cmd triggering
- The dump was created on April 13th, 2017 at 747 milliseconds after 7:49:02 AM

In this version of the driver, the DMN generates the following dump files upon a DMN event:

- IPoIB: The adapter's IPoIB state
- PDDR: The port diagnostics database
- General
- mst files
- Registry

DMN incident dumps are created under the DMN root directory, which can be controlled via the registry. The root directory will include the port identification in its name.

The default is:

- Host: *"\Systemroot\temp\Mlx5_Dump_Me_Now-<b>-<d>-<f>"*
- VF: *"\Systemroot\temp\Mlx5_Dump_Me_Now-<b>-<d>"*. See section Dump Me Now (DMN) Registry Keys.

State Dumping (via Dump Me Now)

Upon several types of events, the drivers can produce a set of files reflecting the current state of the adapter.

Automatic state dumps via DMN are done upon the following events:

Event Type	Description	Provider	Default	Tag
CMD_FAILED	Command failure	Mlx5	On	FAILED
CMD_TIMEOUT	Timeout reached on a command	Mlx5	On	TOUT
RESILIENCY	Resiliency sensor was activated	Mlx5	OFF	RES

Event Type	Description	Provider	Default	Tag
EQ_STUCK	Driver decided that an event queue is stuck	Mlx5	On	EQ
TXCQ_STUCK	Driver decided that a transmit completion queue is stuck	Mlx5	On	TXCQ
RXCQ_STUCK	Driver decided that a receive completion queue is stuck	Mlx5	On	RXCQ
PORT_STATE	Adapter passed to “port up” state, “port down” state or “port unknown” state.	Mlx5	On	PORT
USER	User application asked to generate dump files	Mlx5	N/A	USR

where

Provider	The driver creating the set of files.
Default	Whether or not the state dumps are created by default upon this event.
Tag	Part of the file name, used to identify the event that has triggered the state dump.

Dump events can be enabled/disabled by adding DWORD32 parameters into HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1- 08002be10318}\<nn> as follows:

- Dump events can be disabled by adding MstDumpMode parameter as follows:

```
MstDumpMode 0
```

- PORT_STATE events can be disabled by adding EnableDumpOnUnknownLink and EnableDumpOnPortDown parameters as follows:

```
EnableDumpOnUnknownLink 0
EnableDumpOnPortDown 0
EnableDumpOnPortUp 0
```

> ⚠ As of WinOF-2 v2.10, the registry keys above can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used.

- EQ_STUCK, TXCQ_STUCK and RXCQ_STUCK events can be disabled by adding DisableDumpOnEqStuck, DisableDumpOnTxCqStuck and DisableDumpOnRxCqStuck parameters as follows:

```
DisableDumpOnEqStuck 1
DisableDumpOnTxCqStuck 1
DisableDumpOnRxCqStuck 1
```

The set consists of 2 consecutive mstdump files. These files are created in the same directory as the DMN, and should be sent to NVIDIA® Support for analysis when debugging WinOF2 driver problems.

Their names have the following format: <event_name>-<dump_mode>_<file_index>.txt

<event_name>

Event name	Description
poll-tout-<OPCODE>	Timeout reached on command with polling mode, OPCODE is the command opcode in the driver.

Event name	Description
wait-tout-<OPCODE>	Timeout reached on command while waiting, OPCODE is the command opcode in the driver.
poll-failed-<OPCODE>	Command with polling mode failed, OPCODE is the command opcode in the driver.
wait-failed-<OPCODE>	Command failed, OPCODE is the command opcode in the driver.
eth-eq-<EQN >-<EQ_IDX>	EQ stuck, EQN: EQ number, EQ_IDX: EQ index
eth-txcq-<CQN>	TXCQ is stuck, CQN is the CQ number
eth-rxcq-<CQN>	RXCQ is stuck, CQN is the CQ number
eth-<STATE>	PORT change event, STATE: [“up”, “down”, “none”]
oid	User application asked the dump
BugCheck	Bug check event
resiliency	When resiliency flow is triggered

<dump_mode>: The mode of collecting the mstdump: “crspcae”, “fast-crspace”

<file_index>: The file number of this type in the set

Example:

```
Name: wait-failed-936-fast-crspace_1.txt
```

The default number of sets of files for each event is 20. The other dump files have the filename of: <DumpType>.log

DumpType can be: PDDR, Registry, General, IPoIB, MiniportProfiling

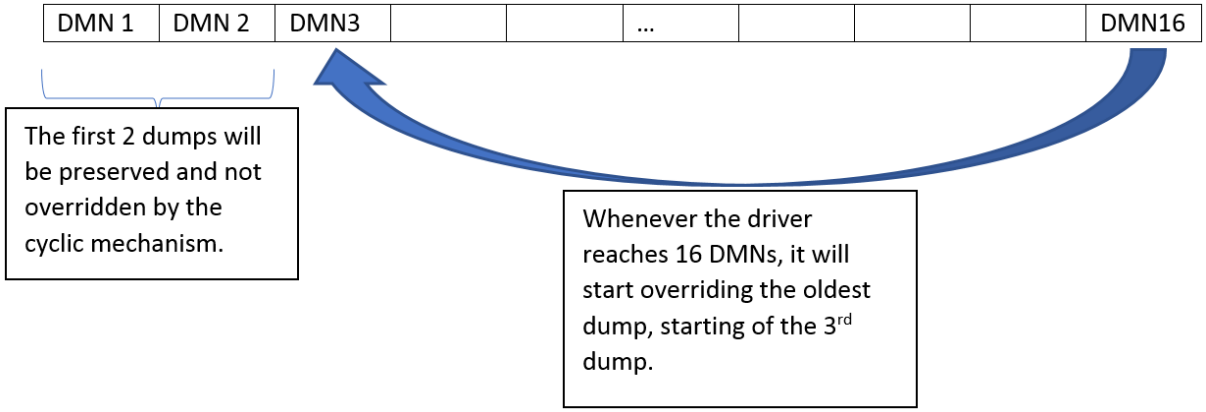
Cyclic DMN Mechanism

The driver manages the DMN incident dumps in a cyclic fashion, in order to limit the amount of disk space used for saving DMN dumps, and avoid low disk space conditions that can be caused from creating the dumps.

Rather than using a simple cyclic override scheme by replacing the oldest DMN incident folder every time it generates a new one, the driver allows the user to determine whether the first N incident folders should be preserved or not. This means that the driver will maintain a cyclic overriding scheme starting from a given index.

The two registry keys used to control this behavior are DumpMeNowTotalCount, which specifies the maximum number of allowed dumps under the DMN root folder, and DumpMeNowPreservedCount, which specifies the number of reserved incident folders that will not be overridden by the cyclic algorithm.

The following diagram illustrates the cyclic scheme’s work, assuming DumpMeNowPreservedCount=2 and DumpMeNowTotalCount=16:



Configuring DMN-IOV

The DMN-IOV detail level can be configured by the "DmnIovMode" value that is located in device parameters registry key. The default value is 2. The acceptable values are 0-4:

Values	Description
0	The feature is disabled
1	Major IOV objects and their state will be listed
2	All VF hardware resources and their state will be listed in the dump (QPs, CQs, MTTs, etc.)
3	All QP-to-Ring mapping will be added (the huge dump)
4	All IOV objects and their state will be list

Dump PDDR Information

The DMN-PDDR can configured by the "EnableDumpOnPortUp" and "EnableDumpOnPortDown" values that are located in device parameters registry keys.

The default values of the keys are follow:

- EnableDumpOnPortUp = 0 [capability disabled]
- EnableDumpOnPortDown = 1 [capability enabled]

Event Logs

DMN generates an event to the system event log upon the success or failure of the dump file generation.

Reported Driver Event Severity: Error

Event ID	Message
0x101	<device name>: Failed to create a full dump me now. Dump me now root directory: <path to root DMN folder> Failure: <Failure description> Status: <status code>

Reported Driver Event Severity: Warning

For a list of the DMN Warning events, see Reported Driver Events.

FwTrace

FwTrace feature allows firmware traces to be logged Online into the WPP tracing without any NVIDIA® specific tools' requirements. It provides an easy way to debug and diagnose issues at production without the need to reproduce the issue. Both the firmware and the driver traces are displayed at the same file. Additionally, FwTrace is also used as a platform for core_dump.

System Requirements	
Firmware versions:	• NVIDIA® ConnectX®-4 v12.22.1002 • NVIDIA® ConnectX®-4 Lx v14.22.1002 • NVIDIA® ConnectX®-5 v16.22.4020

Configuring FwTrace

FwTrace uses Registry Keys for its configuration. For more information see section FwTrace Registry Keys.

FwTrace feature could be enabled/disabled dynamically (without requiring an adapter restart) using the FwTracerEnabled registry key.

FwTrace uses a cyclic buffer. The size of the buffer could be configured using the dynamic registry key FwTracerBufferSize. To change buffer size, set the desired value to FwTracerBufferSize and then restart FwTrace using FwTracerEnabled registry key or adapter restart.

NIC Health Monitor

The NIC Health Monitor is an external tool used to check and monitor the health of the NIC by analyzing the firmware and the diagnostic counters previously collected by the user.

This capability can be used using the following command and its parameters:

```
Mlx5Cmd -Dbg -NicHealthMonitor -hh | -Input <CSV file> [-Type N] [-FullName] [-Desc] [-Format TXT | CSV]
```

where:

-hh	Show this help screen

-Input <CSV file>	File, containing the names and values of counters to be checked.
	Note: This is a mandatory parameter, containing counters for analysis. This file can be produced using the typeperf utility. For example:
	`typeperf -qx | findstr "Mell*" > c:\Counters.txt`
	`typeperf -cf c:\Counters.txt -o c:\CounterData.csv -sc 200 -si 2`
	The first command creates the list of counters to collect.
	The second one collects 200 sets of the above counters, one probe in 2 seconds.

-Type N	Bit field, containing types of results to be shown: • 1 errors • 2 warnings • 3 errors + warnings **(default)** • 4/8 good/unchecked counters The tool makes its analysis based on the internal list of counters that can show issues in the NIC health. There are four possible results of a counter analysis: • ERROR - the value of the counter is regarded as error. • WARN - the value of the counter is suspicious. • GOOD - the value of the counter is OK. • ABSNT - the counter is not in the internal list and was not analyzed.
-FullName	Print full counter names. The full name of the counter is quite long. Its format is: `\\node_name\counter_set_name(adapter_instance(es))\counter_name` By default, the tool prints only counter_name.
-Desc	Print description of the counter. The name of the counters is often not enough to understand its purpose. The tool can print the description of the counter to give more information.
-Format TXT | CSV	Output format; default - TXT (plain text). The tool prints the results to the stdout. It can produce the output in two formats: Plain Text (default) or CSV.

-CfgFile <CfgFile>	Some of the checked counters have two configuration values: threshold and time unit.
	To see the default values of these parameters, run: -List
	`Mlx5Cmd.exe –Dbg – NicHealthMonitor –List`
	The output is a plane text that can be easily edit by changing threshold values and then check counters with the new thresholds:
	`Mlx5Cmd –Dbg – NicHealthMonitor –Check –Input <CSV file> –CfgFile config.log`

The following are a few examples of how to run the command:

- To print only error counters in default format:

```
Mlx5Cmd.exe -Dbg -NicHealthMonitor -input c:\tmp\CounterData.csv -type 1
```

- To print only error and warning counters with full name of counters:

```
Mlx5Cmd.exe -Dbg -NicHealthMonitor -input c:\tmp\CounterData.csv -type 3 -FullName
```

- To print conclusions on all counters, found in the input file, with maximum info and in CSV format:

```
Mlx5Cmd.exe -Dbg -NicHealthMonitor -input c:\tmp\CounterData.csv -type 15 -FullName -Desc -Format CSV >
output.csv
```

Resource Dump

Resource Dump is a debuggability utility that extracts and prints data segments generated by the firmware/hardware. The driver will register to all the supported types of resources (Segments) and will listen on the events sent by the firmware to initiate a collect resource dump request and export it to the filesystem (using Dump-Me-Now mechanism).

For further information, see ResourceDump Registry Keys and Resource Dump Utility.

> ⚠ As Resource Dump depends on DMN, its enablement is coupled with the DMN enablement.

RDMA Capabilities

Shutting Down RDMA QPs with Excessive Retransmissions

> ⚠ This capability is supported in RoCE (Ethernet) only.

The driver offers a mechanism to detect excessive retransmissions for an RC connection, and to close the connection in response to it. If the number of retransmissions due to a Local Ack Timeout, NAK-Sequence Error, or Implied NAK, during a specified period, exceeds the specified threshold, the QP will be handled as if the IB spec defined Retry Count was exceeded.

Setting this limit for all RC QPs is done by setting the EXT_QP_MAX_RETRY_PERIOD registry as a measurement period, and the EXT_QP_MAX_RETRY_LIMIT registry as a retries threshold. If any of these registries is set to 0x0, the feature is disabled.

> ⚠ When the threshold is exceeded during the measurement period, the following will occur:
> - The QP will be transitioned to an Error (ERR) state
> - The "Requester QP Transport Retries Exceeded Errors" counter will be incremented. See Mellanox WinOF-2 Diagnostics.

The Shutdown RDMA QPs feature is controlled per adapter, using registry keys.

Registry keys location: *HKLM\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\<nn>*

For more information on how to find a device index nn, please refer to Finding the Index Value of the Network Interface.

Key Name	Key Type	Values	Description
EXT_QP_MAX_RETRY_LIMIT	REG_DWORD	[0-0xFFFF] Default = 50	The number of retransmissions during EXT_QP_MAX_RETRY_PERIOD for which the QP will be closed due to a faulty connection. The 0x0 value indicates that the feature is disabled.**Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used. **Note:** If the *EXT_QP_MAX_RETRY_LIMIT* value is set to 0, the *EXT_QP_MAX_RETRY_PERIOD* value must be set to 0 as well. **Note:** *EXT_QP_MAX_RETRY_LIMIT* and *EXT_QP_MAX_RETRY_PERIOD* registry keys are supported only if the firmware supports this capability. If these keys are used, but not supported by the firmware, the following message is displayed to the user: *"<adapter name>: Shutting Down RDMA QPs with Excessive Retransmissions feature is not supported by FW <FW version>"*.
EXT_QP_MAX_RETRY_PERIOD	REG_DWORD	[0-0xFFFF] Default = 1	The period for measuring the number of retransmissions to declare the connection as faulty and close the QP. The value is given in seconds. The 0x0 value indicates that the feature is disabled. **Note:** As of WinOF-2 v2.10, this key can be changed dynamically. In any case of an illegal input, the value will fall back to the default value and not to the last value used. **Note:** If the EXT_QP_MAX_RETRY_PERIOD value is set to 0, the EXT_QP_MAX_RETRY_LIMIT value must be set to 0 as well. **Note:** *EXT_QP_MAX_RETRY_LIMIT* and *EXT_QP_MAX_RETRY_PERIOD* registry keys are supported only if the firmware supports this capability. If these keys are used, but not supported by the firmware, the following message is displayed to the user: *"<adapter name>: Shutting Down RDMA QPs with Excessive Retransmissions feature is not supported by FW <FW version>"*.

NVIDIA BlueField SmartNIC Mode

The NVIDIA® BlueField® family of (Data Processing Unit) DPU devices combines an array of Arm processors coupled with the NVIDIA® ConnectX® interconnect. Standard Linux distributions run on the Arm cores allowing common open source development tools to be used. The SoC can be accessed via USB (external cable) or PCIe driven by our RShim drivers. RShim drivers provides functionalities like resetting the Arm cores, pushing a bootstream image, networking functionality and console functionality.

For further information see RShim Drivers and Usage.

When the adapter is in SmartNIC mode, the following features are controlled from System-On-Chip (SoC) side. For more information on NVIDIA® BlueField® and functionality, please refer to NVIDIA BlueField Family Documentation → BlueField Software Overview.

- Encapsulation/Decapsulation - VXLAN/GRE packet encapsulation/decapsulation is done on the SoC side. Please refer to NVIDIA BlueField Family Documentation → Virtual Switch on BlueField SmartNIC
- Rate limiting of host PF and VF - For example, users may limit the transmit rate of the PF in the host to 1000mbps and VF to 500 mbps. Please refer to NVIDIA BlueField Family Documentation → QoS Configuration
- Offloading VLANs - The OVS can add VLAN tag to all packets sent by network interface running on host PF or VF. Please refer to NVIDIA BlueField Family Documentation → Virtual Switch on BlueField SmartNIC
- Bluefield Link Aggregation - configure network bonding on the Arm side in a manner transparent to the host. Under such configuration, the host would only see a single PF. Please refer to NVIDIA BlueField Family Documentation → BlueField Link Aggregation
- Setting Host PF and VF Default MAC Address. Please refer to NVIDIA BlueField Family Documentation → Controlling Host PF and VF Parameters
- DCQCN and DSCP based congestion control for RoCE. Please refer to: https://support.mellanox.com/s/article/mlnx-qos
- QoS - Host settings can be honored or ignored based on settings (changeable using mstpriv tool). Please refer to https://support.mellanox.com/s/article/mlnx-qos
- Link speed cannot be changed using user space (mlx5cmd). Please refer to NVIDIA BlueField Family Documentation → Controlling Host PF and VF Parameters

Limitations

- Performance counters - Cannot query Mellanox WinOF-2 PCI Device Diagnostics
- Cannot query/modify VF capabilities from Windows host
- Dropless mode query/set is not supported from the host side
- When performing MlxFwReset (one of our MFT tools), need to disable host network adapters manually and wait until SoC is up before enabling them

Open-vSwitch Limitation and Windows Certification Workaround

- Open vSwitch (OVS) running on the Arm cores allows Virtual Machines (VMs) to communicate with each other and with the outside world. For more details on OVS, please refer to NVIDIA BlueField Family Documentation.
- OpenvSwitch (OVS) running on the Arm cores supports two modes:
    - hardware offload mode enabled - With Hardware offload enabled (default mode), the first few packets are processed by the OVS for learning and rule injection which can be processed in parallel thus, test fails because packets go out-of-order to the host (windows driver).
    - hardware offload mode disabled - With Hardware Offload disabled, all packets go through the Arm core and cannot keep up with heavy network traffic. To overcome this limitation, and to make it easy for customers who want to run certification, we provide two scripts under `/opt/mellanox/hlk`.
      Please execute `"/opt/mellanox/hlk/mlnx-pre-hlk"` from the SoC before starting the

HLK tests and after done, execute `"/opt/mellanox/hlk/mlnx-post-hlk"` to enable OVS and delete manually programmed rules.

- NDIS6.0/6.5 of Windows HLK tests use IPX/SPX protocol for send/receive in quite a few cases. There is no handshake or retransmits. Test keeps track of packet count and ordering.

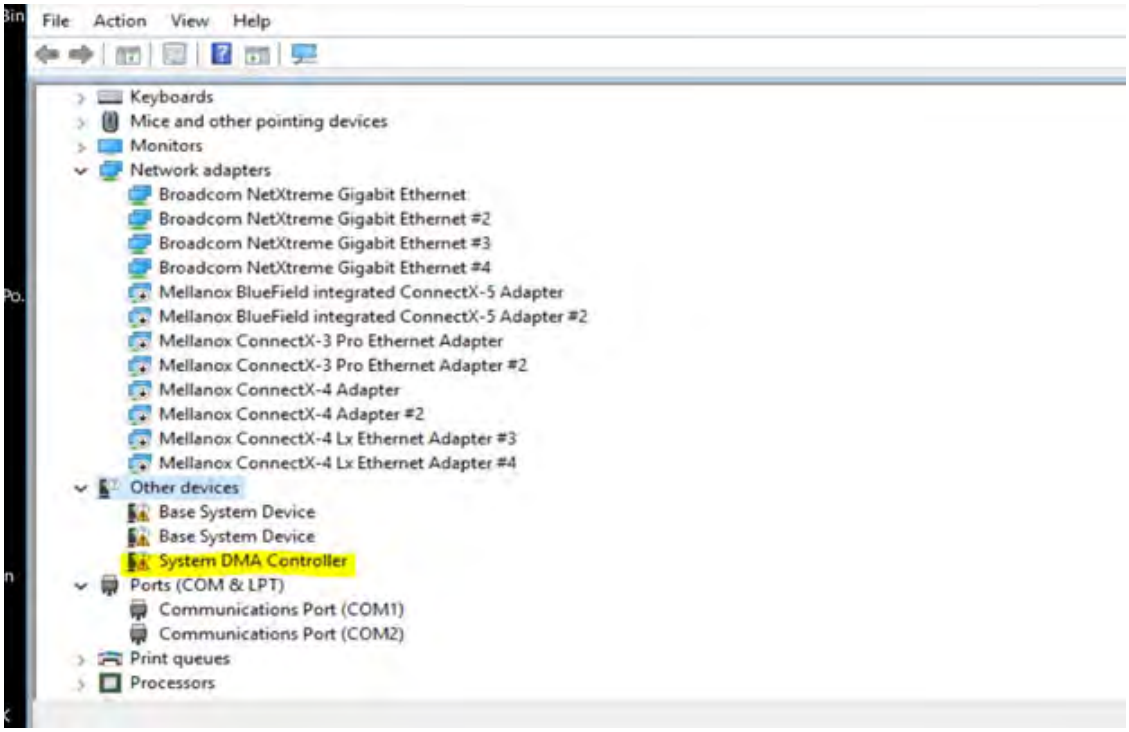
RShim Drivers and Usage

This section of the user manual describes installation and operation of NVIDIA® RShim drivers.

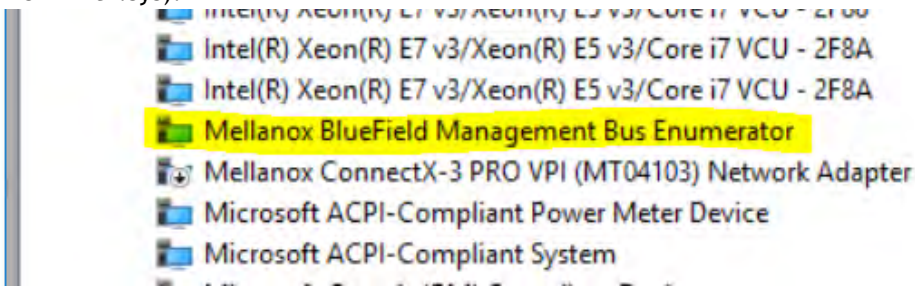
> ⚠ The Rshim drivers will be installed only on Windows Server 2016 and above or Windows Client 10 Operating Systems.

Verifying RShim Drivers Installation

1. Open the Device Manager when no drivers are installed to make sure a new PCIe device is available as below.



2. Run the installer to install all 3 drivers (MlxRshimBus.sys, MlxRshimCom.sys and MlxRshimEth.sys).



3. Make sure the Bus driver created 2 child devices after the installation (Com port and the Ethernet adapter).

At this time, PuTTY application or any other network utility can be used to communicate with DPU via Virtual Com Port or Virtual Ethernet Adapter (ssh). The Com Port can be used using the 9600 baud-rate and default settings.

> ⚠ RShim drivers can be connect via PCIe (the drivers we are providing) or via USB (external connection) but not both at the same time. So when the bus driver detects that an external USB is already attached, it will not create the child virtual devices for data access. Access via PCIe is available once the USB connection is removed.

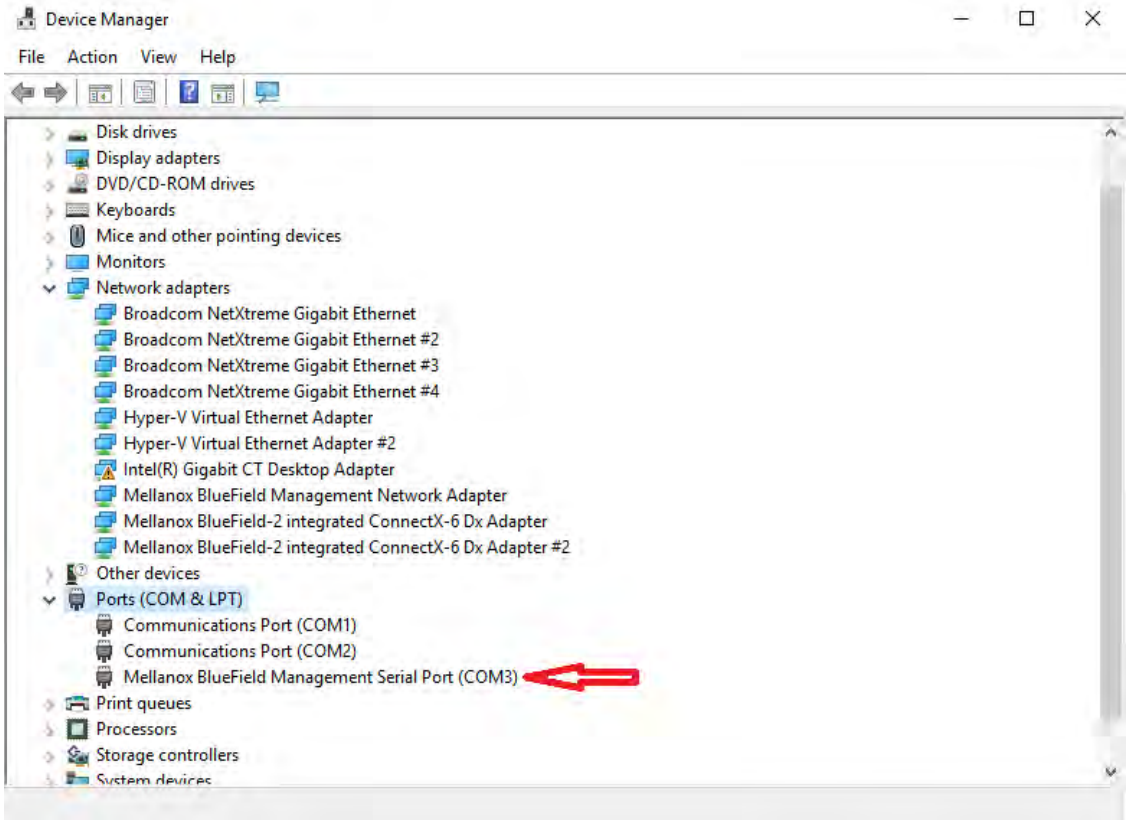
Accessing BlueField DPU From Host

The BlueField DPU can be accessed via PuTTY or any other network utility application to communicate via virtual COM or virtual Ethernet adapter. To use COM:

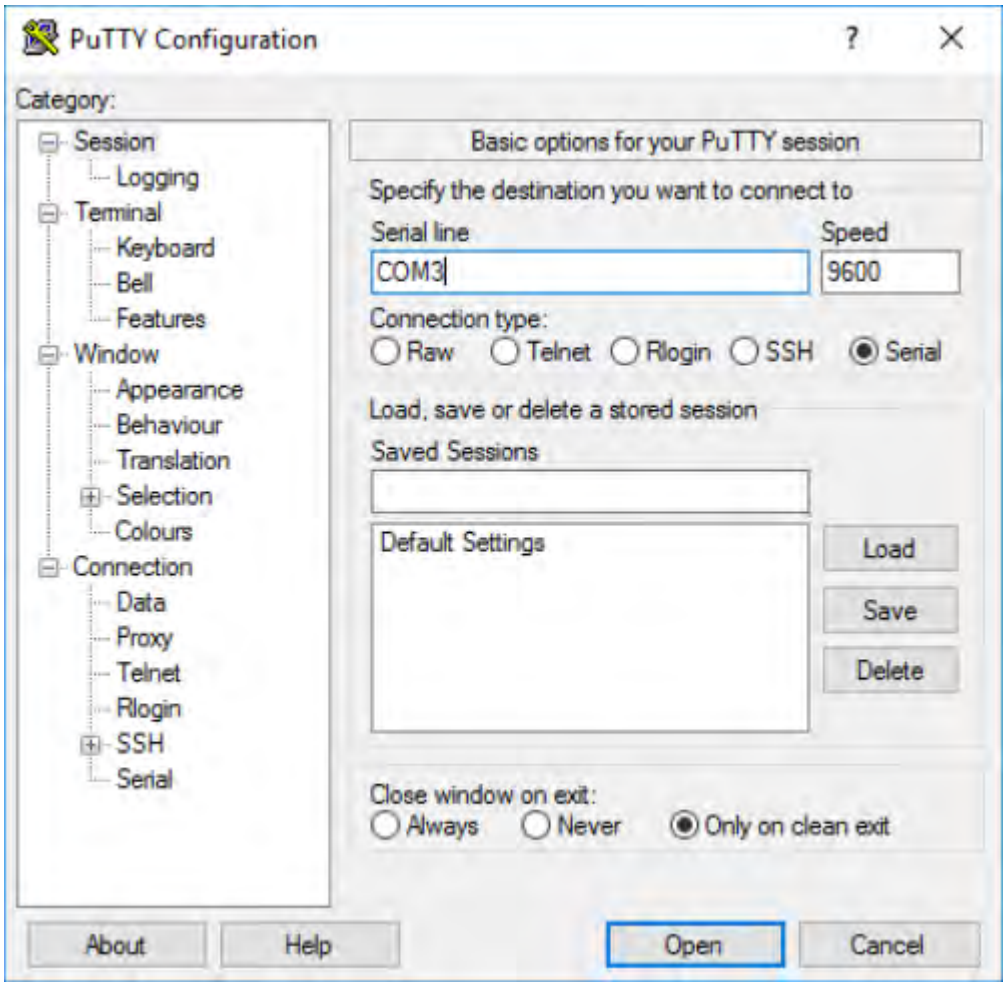
1. Open Putty.
2. Change connection type to Serial.
3. Run the following command in order to know what to set the "Serial line" field to:

```
C:\Users\username\Desktop> reg query HKLM\HARDWARE\DEVICEMAP\SERIALCOMM | findstr MlxRshim
    \MlxRshim\COM3      REG-SZ      COM3
```

In this case use COM3. This name can also be found via Device Manager under "Ports (Com & LPT)".



4. Press Open and hit Enter.



To access via BlueField management network adapter, configure an IP address as shown in the example below and run a ping test to confirm configuration.

RShim Ethernet Driver

Registry parameters

The device does not support any type of statefull or stateless offloads. This is indicated to the Operating System accordingly when the driver loads. The MAC address is a pre-defined MAC address (CA-FE-01-CA-FE-02). The following registry keys can be used to change basic settings such as MAC address.

Registry Name	Description	Valid Values
HKLM\SYSTEM\CurrentControlSet\Control\	The size, in bytes, of the largest supported	1514 (default) - 2048
Class\{4d36e972-e325-11ce-	Jumbo Packet (an Ethernet frame that is	
bfc1-08002be10318}\<nn>*JumboPacket	greater than 1514 bytes) that the	
	hardware can support.	

Registry Name	Description	Valid Values
HKLM\SYSTEM\CurrentControlSet\Control\	The network address of the device. The	CA-FE-01-CA-FE-02
Class\{4d36e972-e325-11ce-	format for a MAC address is: XX-XX-XX-XX-	(default)
bfc1-08002be10318}\<nn>*NetworkAddress	XX-XX.	
HKLM\SYSTEM\CurrentControlSet\Control\	The number of receive descriptors used by	16 – 64 (Default)
Class\{4d36e972-e325-11ce-	the miniport adapter.	
bfc1-08002be10318}\<nn>\ReceiveBuffers		

For instructions on how to find interface index in registry <nn>, please refer to section Finding the Index Value of the Network Interface.

Support of Multiple Bluefield Cards

When the server contains several BlueField cards, several instances of Rshim bus driver will run and each one will create two virtual adapters for COM and network interface. In this case, a unique MAC and IP addresses must be set for the virtual network adapters.

MAC address can be set either using the above registry key or the RshimCmd tool's commands below, whereas the IP address should be set manually, and it should allow communication with DPU's `tmfifo_net0` interface. In this case, there is a need to set unique MAC and IP addresses also for the DPUs. The IP address is changed with the help of bf.cfg file.

Rshim Com Driver

To check the progress of the DPU reboot, use the COM interface by connecting to the DPU before pushing the new BFB image or resetting of DPU.

To change the way the driver boots, enter the UEFI or GRUB menus after the corresponding boot process prompts.

RShim Bus Driver

This driver does all the read/write work to the hardware registers. RShimCmd tool sends IOCTLs to the bus driver to restart the system on chip, to push a new BlueField boot stream image or to perform other actions.

RShimCmd Tool

RShimCmd is a command line tool that provides user with several possibilities.

Functionality	Command	Description
Device Enumeration	`RshimCmd -EnumDevices`	Prints the bus numbers of all BlueField device, found on the machine. All the next numbers contain the ‘bus number’ as a parameter.

Verbosity Level	`RshimCmd -SetDisplayLevel [ 0 | 1] -Busnum N`	Sets the verbosity level of the Rshim bus drivers replies.
Boot Mode	`RshimCmd -SetBootMode <new_boot_mode> -Busnum N`	Sets the next boot mode in DPU.
Timeout	`RshimCmd -SetBootTimeout <timeout_in_secs> -Busnum N`	Sets the value of timeout in -PushImage command.

Print of the Driver's and DPU's Variables	`RshimCmd -PrintVars -Busnum N`	Prints a few of the driver's and DPU's variables
		• When the verbosity is 0 (default), the command prints only the driver variables:
		`DISPLAY_LEVEL 1 (0:basic, 1:advanced, 2:log)`
		• When the verbosity is 1, the command prints in additional the following DPU variables:
		`BOOT_MODE 1 (0:rshim, 1:emmc, 3:emmc-boot-swap)`
		`BOOT_TIMEOUT 120 (seconds)`
		`DEV_INFO BlueField-2 (Rev 1)`
		`PEER_MAC 00:1a:ca:ff:ff:07 (rw)`

Change the MAC Address of the `tmfifo_net0` Interface in DPU	`RshimCmd -SetPeerMac xx:xx:xx:xx:xx:xx -Busnum N`	Sets new MAC address to `tmfifo_net0` interface in DPU. Reading the MAC address back with `''RshimCmd -PrintVars'` can print either the new or the old value, depending on the OS in the DPU. **Note:** In both cases, DPU must be reset for it to accept the new MAC address.
Restart the DPU	`RshimCmd -RestartSmartNic 1 -BusNum 11`	Restarts the DPU. **Note:** This command should be used after changing the MAC address of `tmfifo_net0` interface in the DPU.

Push the BFB Image	`RshimCmd -PushImage c:\bin\MlnxBootImage.bfb -BusNum 11`	A BlueField Boot file (.bfb) is a generated BlueField boot stream file that contains Linux operating system image intended to be run on the DPU. BFB files can be downloaded from the NVIDIA DOCA SDK webpage
		IP address of the RShim Ethernet component (called `tmfifo_net0` on the BlueField side) is 192.168.100.2/30 by default. Please set the IP address on the Windows side accordingly to be able to communicate via SSH. For example, 192.168.100.1/30.
		Once the BFB file is downloaded, the tool starts to print the DPU Log file showing the process happening in the DPU. The prints are usually as follows:
		`----------------------------------------`
		`Log Messages`
		`----------------------------------------`
		`INFO[BL2]: start`
		`INFO[BL2]: DDR POST passed`
		`INFO[BL2]: UEFI loaded`
		`INFO[BL31]: start`
		`INFO[BL31]: runtime`
		`The printing of Log is performing maximum for 720 seconds.`
		`One can stop it with Ctrl-C.`

Tuning of BFB image before Pushing / BlueField UEFI System Boot Customizations during Installation	N/A	The BFB file can be tuned with the help of the `bf.cfg` configuration file. This file is fully described in BlueField documentation.
		To include the `bf.cfg` file into the BFB installation, you must append the file to the BFB file as follow:
		1. Copy BFB file to a local folder.
		`Copy <path>\DOCA_<version>_BSP_<version>_Ubuntu_20.04-5.20220707.bfb c:\bf\MlnxBootImage.bfb`
		2. Append the bf.cfg file to the BFB file.
		`Cd c:\bf`
		`Copy /b MlnxBootImage.bfb + bf.cfg MlnxBootImage_with_bf_cfg.bfb`
		3. Download the BFB image.
		`RshimCmd -PushImage c:\bf\MlnxBootImage_with_bf_cfg.bfb -BusNum 11`
		Pay attention, that the `bf.cfg` file is intended for Linux, so it should be created according to Linux rules. For example, the lines of this text file should end in LF and not in CR/LF as is accepted in Windows.
		Also, the syntax should be as the accepting OS expects. For example, there should be no spaces in the middle of set statements:
		`NET_RSHIM_MAC=00:1a:ca:ff:ff:05`

Printing of DPU Log	`RshimCmd -PrintLog -Busnum N`	Prints the above DPU Log.

EventLogs and Driver Logging

All driver logging is part of the Mellanox-WinOF2-Kernel trace session that comes with the network drivers installation. The default location to the trace is at `%SystemRoot%\system32\LogFiles\Mlnx\Mellanox-WinOF2-System.etl`.

The following are the Event logs RShim drivers generate:

RShim Bus Driver

Event ID	Severity	Message
2	Informational	RShim Bus driver loaded successfully.
3	Informational	Device successfully stopped.
4	Error	The SmartNic adapter card seems to be stuck as the boot fifo data is not being drained.
5	Error	Driver startup failed due to failure in creation of the child device.
6	Error	Smartnic is in a bad state. Please restart smartnic and reload bus drivers. Please refer to user manual on how to restart smartnic.
7	Warning	Smartnic is in LiveFish mode.
8	Warning	Failed creating child virtual devices as a backend USB device is attached and accessing RShim FIFO. Please refer to user manual for more details.

RShim Serial Driver

Event ID	Severity	Message
2	Informational	RShim Serial driver loaded successfully.
3	Informational	device successfully stopped.

RShim Ethernet Driver

Event ID	Severity	Message
2	Error	MAC Address read from registry is not supported. Please set valid unicast address.
3	Informational	device is successfully stopped.
4	Warning	value read from registry is invalid. Therefore use the default value.
5	Error	SmartNic seems stuck as transmit packets are not being drained.
6	Informational	RShim Ethernet driver loaded successfully.

Utilities

This chapter describes various utilities used in the WinOF-2 driver to manage device's performances.

The chapter contains the following sections:

- Fabric Performance Utilities
- Management Utilities
- Snapshot Utility

Fabric Performance Utilities

The performance utilities described in this chapter are intended to be used as a performance micro-benchmark. They support both InfiniBand and RoCE.

⚠ For further information on the following tools, please refer to the help text of the tool by running the --help command line parameter.

⚠ The performance utilities described in the table below will be deprecated as of the next release.

Utility	Description
nd_write_bw	This test is used for performance measuring of RDMA-Write requests in Microsoft Windows Operating Systems. nd_write_bw is performance oriented for RDMA-Write with maximum throughput, and runs over Microsoft's NetworkDirect standard. The level of customizing for the user is relatively high. User may choose to run with a customized message size, customized number of iterations, or alternatively, customized test duration time. nd_write_bw runs with all message sizes from 1B to 4MB (powers of 2), message inlining, CQ moderation.
nd_write_lat	This test is used for performance measuring of RDMA-Write requests in Microsoft Windows Operating Systems. nd_write_lat is performance oriented for RDMA-Write with minimum latency, and runs over Microsoft's NetworkDirect standard. The level of customizing for the user is relatively high. User may choose to run with a customized message size, customized number of iterations, or alternatively, customized test duration time. nd_write_lat runs with all message sizes from 1B to 4MB (powers of 2), message inlining, CQ moderation.
nd_read_bw	This test is used for performance measuring of RDMA-Read requests in Microsoft Windows Operating Systems. nd_read_bw is performance oriented for RDMA-Read with maximum throughput, and runs over Microsoft's NetworkDirect standard. The level of customizing for the user is relatively high. User may choose to run with a customized message size, customized number of iterations, or alternatively, customized test duration time. nd_read_bw runs with all message sizes from 1B to 4MB (powers of 2), message inlining, CQ moderation.
nd_read_lat	This test is used for performance measuring of RDMA-Read requests in Microsoft Windows Operating Systems. nd_read_lat is performance oriented for RDMA-Read with minimum latency, and runs over Microsoft's NetworkDirect standard. The level of customizing for the user is relatively high. User may choose to run with a customized message size, customized number of iterations, or alternatively, customized test duration time. nd_read_lat runs with all message sizes from 1B to 4MB (powers of 2), message inlining, CQ moderation.
nd_send_bw	This test is used for performance measuring of Send requests in Microsoft Windows Operating Systems. nd_send_bw is performance oriented for Send with maximum throughput, and runs over Microsoft's NetworkDirect standard. The level of customizing for the user is relatively high. User may choose to run with a customized message size, customized number of iterations, or alternatively, customized test duration time. nd_send_bw runs with all message sizes from 1B to 4MB (powers of 2), message inlining, CQ moderation.
nd_send_lat	This test is used for performance measuring of Send requests in Microsoft Windows Operating Systems. nd_send_lat is performance oriented for Send with minimum latency, and runs over Microsoft's NetworkDirect standard. The level of customizing for the user is relatively high. User may choose to run with a customized message size, customized number of iterations, or alternatively, customized test duration time. nd_send_lat runs with all message sizes from 1B to 4MB (powers of 2), message inlining, CQ moderation.

MlxNdPerf Utility

MlxNdPerf is a new tool that replaces all older Network Direct applications from older drivers (e.g nd_write_bw, nd_read_bw, nd_send_bw, nd_*_lat). The tool is used to determine the maximum performance with various parameters and what is the current available RDMA Read\Write\Send Performance between two endpoints.

The following are the commands used by the tool to perform various operations:

Client or Server Role

The role of Client or Server determines if this side is an RDMA requestor or responder (Client → Requestor, Server → Responder).

Usage	MlxNdPerf.exe -Server\ -Client

RDMA Operation

Determines the RDMA operation to be performed, a single option per time.

Usage	MlxNdPerf -Read\-Write\-Send

Source/Destination IP

Determines the Source IP (The local IP) and the Destination IP (Remote IP).

Usage	MlxNdPerf -SrcIP\ -DestIP

Number of Threads

Determines the number of threads to be executed, a single QP per thread.

Usage	MlxNdPerf -NumOfThreads

Port Number

Determines the port number used.

Usage	MlxNdPerf -PortNumber

Number of Scatter

Determines the number of scatter gather entries per post Send\Write\Read.

Usage	MlxNdPerf -SgeNumber

Buffer Size

Determines the number of bytes to be transmitted by a single post Send\Write\Read.

Usage	MlxNdPerf -BufferSize

Queue Depth

Determines the number of entries in the QP and the CQ.

Usage	MlxNdPerf -QueueDepth

Number of Iteration

Determines the number of iteration for post Send\Write\Read. Is ignored when in Duration mode.

Usage	MlxNdPerf -Iterations

Duration Mode

Duration mode – for how long the test executes in seconds.

Usage	MlxNdPerf -Duration

Event Notification Mode

Use event Notification mode for the CQ, it does not poll the CQ.

Usage	MlxNdPerf -UseEvents

Resiliency

Registering to the adapter's status changes callbacks and listening for any adapter status changes. In this mode the application will not exit unless the test is completed successfully.

Note: This Mode is not available for the Server side when in Send Mode.

Usage	MlxNdPerf -Resilient

Latency

Latency can be measured using the "-Latency" parameter. This parameter should be added to one of the operation `- Write, Read` or `Send`.

Usage	MlxNdPerf -Read\-Write\-Send -Latency

The following are a few limitations related to the latency tests:

- Parameters `'-Latency'` and `'-BufferSize'` should be coded from both sides.
- Parameters `'-Resilient', '-NumOfThreads', '-UseEvents'` and `'-QueueDepth'` are not supported with latency tests.

Verbose

Enables extra information prints.

Usage	MlxNdPerf -Verbose

Example1 Measure the bandwidth on operation IB Read with traffic from 2 threads, running for 30 seconds

- Server side: MlxNdPerf.exe -Server -Read -SrcIp 11.137.58.1 -DestIp 11.137.58.1 -NumOfThreads 2 -Duration 30
- Client side: MlxNdPerf.exe -Client -Read -SrcIp 11.137.57.1 -DestIp 11.137.58.1 -NumOfThreads 2 -Duration 30

Example2 Measure the latency on operation IB Write with Ipv6 addresses (if Estat presents)

- Server side: MlxNdPerf.exe -Server -Write -SrcIp fe80::ee0d:9aff:fe42:e8e8 -DestIp fe80::ee0d:9aff:fe42:e8e8 -Latency -BufferSize 8
- Client side: MlxNdPerf.exe -Client -Write -SrcIp fe80::ee0d:9aff:fe42:e8e4 -DestIp fe80::ee0d:9aff:fe42:e8e8 -Latency -BufferSize 8

Win-Linux nd_rping Test

The purpose of this test is to check interoperability between Linux and Windows via an RDMA ping. The Windows *nd_rping* was ported from Linux's RDMACM example: *rping.c*

- Windows
  - To use the built-in nd_rping.exe tool, go to: *C:\Program Files\Mellanox\MLNX_WinOF2\Performance Tools*
  - To build the *nd_rping.exe* from scratch, use the SDK example: choose the machine's OS in the configuration manager of the solution, and build the *nd_rping.exe* .
- Linux
  - Installing the MLNX_OFED on a Linux server will also provide the "rping" application.

Management Utilities

The management utilities described in this chapter are used to manage device's performance, NIC attributes information and traceability.

The following are the supported management utilities:

- mlx5cmd Utilities
  - Performance Tuning Utility
  - Information Utility
  - DriverVersion Utility
  - Trace Utility
  - QoS Configuration Utility
    - Quick RoCE Configuration (One-Click RoCE)
  - Registry Keys Utility
  - Non-RSS Traffic Capture Utility
  - Sniffer Utility
  - Link Speed Utility
  - Link FEC Configuration Utility
  - NdStat Utility
  - NdkStat Utility
  - Debug Utility
    - VF Resources
    - Features Status Utility
    - Firmware Capabilities
    - Port Diagnostic Database Register (PDDR)
    - Software Reset for Adapter Command
    - Resource Dump

        - Packet Pacing Capabilities
    - Temperature Utility
    - Get-NetView Utility
    - Display RSS Information
    - smpquery Utility
    - Configuration Validator
    - VXLAN Offloading Configuration Utility
- AutoLogger
- DevX Utility

mlx5cmd Utilities

mlx5cmd is a general management utility used for configuring the adapter, retrieving its information and collecting its WPP trace.

Usage	`mlx5Cmd.exe <tool-name> <tool-arguments>`

Performance Tuning Utility

This utility is used mostly for IP forwarding tests to optimize the driver's configuration to achieve maximum performance when running in IP router mode.

Usage	`mlx5cmd.exe -PerfTuning <tool-arguments>`

Information Utility

This utility displays information of NVIDIA® NIC attributes. It is the equivalent utility to ibstat and vstat utilities in WinOF.

Usage	`mlx5cmd.exe -Stat <tool-arguments>`

DriverVersion Utility

The utility can display both the PF's and the VF's driver version.

Usage	`mlx5cmd -DriverVersion -hh | -Name <adapter name> | [-PF] | [-VF] <VF number>`

The VF's driver version format naming is different when the VM runs on a Windows or a Linux OS.If the VF number is not set, then all the driver's VFs' versions will be printed.

- In a VM that runs on Windows OS, the naming format is: Os version,Driver Name,Driver version (e.g., Windows2012R2,WinOF2,2.000.019684)
- In a VM that runs on Linux OS, the naming format is: OS,Driver,Driver version

- (e.g., Linux Driver: Linux,mlx5_core,4.003.030211; Linux Inbox Driver: Linux,mlx5_core,3.0-1)

Trace Utility

The utility saves the ETW WPP tracing of the driver.

Usage	`mlx5cmd.exe -Trace <tool-arguments>`

QoS Configuration Utility

The utility configures Quality of Service (QoS) settings.

Usage	`mlx5cmd.exe -QoSConfig -Name <Network Adapter Name> <-DefaultUntaggedPriority | -Dcqcn | -SetupRoceQosConfig>`

For further information about the parameters, you may refer to RCM Configuration.

Quick RoCE Configuration (One-Click RoCE)

This utility provides a quick RoCE configuration method using the mlx5cmd tool. It enables the user to set different QoS RoCE configuration without any pre-requirements.

To set the desired RoCE configuration, run the *-Configure <Configuration name>* command.

The following are the types of configuration currently support:

- Lossy fabric
- Lossy fabric with QoS
- Lossless fabric

Once set, RoCE will be configured with DSCP priority 26 by default, if the *-Priority* or *-Dscp* flags are not specified.

When configuring the interface to work in a "Lossy fabric" state, the configuration is returned to its default (out-of-box) settings and the *-Dscp* and *-Priority* flags are ignored.

To check the current configuration, run the *-Query* command.

Detailed usage	`mlx5cmd.exe -QosConfig -SetupRoceQosConfig -h`

Registry Keys Utility

This utility shows the registry keys that were set in the registry and are read by the driver. The PCI information can be queried from the "General" properties tab under "Location".

Usage	`mlx5cmd.exe -RegKeys [-bdf <pci-bus#> <pci-device#> <pci-function#>]`
Example	If the "Location" is "PCI Slot 3 (PCI bus 8, device 0, function 0)" `mlx5cmd.exe -RegKeys -bdf 8.0.0`

Non-RSS Traffic Capture Utility

The RssSniffer utility provides sampling of packets that did not pass through the RSS engine, whether it is non-RSS traffic, or in any other case that the hardware determines to avoid RSS hashing.Non-RSS Traffic Capture Utility

The tool generates a packet dump file in a .pcap format. The RSS sampling is performed globally in native RSS mode, or per vPort in virtualization mode, when the hardware vRSS mode is active.

Detailed usage	`mlx5cmd.exe -RssSniffer -hh`

> Note that the tool can be configured to capture only a part of the packet, as well as specific packets in a sequence (N-th).

Sniffer Utility

Sniffer utility provides the user the ability to capture Ethernet, RoCE and IB traffic that flows to and from the NVIDIA® NIC's ports. The tool generates a packet dump file in *.pcap* format. This file can be read using the Wireshark tool (www.wireshark.org) for graphical traffic analysis. The .pcap file generated by the Sniffer Utility will be limited by default to 10M. Users can change or cancel the limit size per their demand. In order to force the file limit, the oldest captures will be saved in fileNamePrev.pcap and will be deleted when the limit is reached.

> In Bluefield 2 SmartNIC mode, sniffer cannot capture VF to VF traffic.

Detailed usage	`mlx5cmd.exe -sniffer -help`

> When using the sniffer utility in IPoIB in loopback mode, between VMs and hosts on the same network port, packets are seen twice in the pcap file: once for transmitting and once for receiving.
>
> For multicast packets, packets are seen once for each direction and not for each destination.

> The Ethernet Sniffer utility when in SR-IOV mode, on ConnectX-5 and above adapter cards, sniffs only the PF’s traffic and not its VF’s traffic.

Link Speed Utility

This utility provides the ability to query supported link speeds by the adapter. Additionally, it enables the user to force set a particular link speed that the adapter can support.

> When using this utility, setting the link speed to 56GbE is not supported.

Usage	`mlx5cmd.exe -LinkSpeed -Name <Network Adapter Name> -Query`
Example	`mlx5cmd.exe -LinkSpeed -Name <Network Adapter Name> -Set 1`
Detailed usage	`mlx5cmd.exe -LinkSpeed -hh`

Link FEC Configuration Utility

Forward Error Correction (FEC) is an algorithm for finding and fixing errors in data transmission on physical link. The NIC can support several algorithms for every link speed. There is an internal register called PPLM, which contains information on FEC algorithms for every link speed.

PPLM register contains two fields for every link speed - 'cap' and 'admin'.

- 'cap' – means 'capability' – is a bitmask field, showing several FEC algorithms, supported for this link speed.
- 'admin' – means 'configured' – contains the above 'cap' field where only one bit is set. It defines the FEC algorithm which is currently configured.

The Link FEC Configuration utility provides the ability to query supported link FEC modes by the adapter for the current link speed and for all supported link speeds.

Additionally, the utility enables the user to change the default FEC algorithm to one of the FEC modes, that the adapter supports.

Usage	`mlx5cmd.exe -Dbg -LinkSpeed -Name <Network Adapter Name> -Query | -QueryPplm | -Set <value>`
Example	`mlx5cmd.exe -Dbg -LinkSpeed -Name <Network Adapter Name> -Set RS`
Detailed usage	`mlx5cmd.exe -Dbg -LinkSpeed -hh`

NdStat Utility

This utility enumerates open ND connections. Connections can be filtered by adapter IP or Process ID.

Usage	`mlx5cmd -NdStat -hh | [-a <IP address>] [-p <Process Id>] [-e] [-n <count>] [-t <time>]`
Example	`mlx5cmd -NdStat`
Detailed usage	`mlx5cmd -NdStat -hh`

NdkStat Utility

This utility enumerates open NDK connections. Connections can be filtered by adapter IP or Process ID.

Usage	`mlx5cmd -NdkStat -hh | [-a <IP address>] [-e] [-n <count>] [-t <time>]`
Example:	`mlx5cmd -NdkStat`
Detailed usage	mlx5cmd -NdkStat -hh `mlx5cmd -NdkStat -hh`

Debug Utility

This utility exposes driver's debug information.

Usage	`mlx5cmd -Dbg <-PddrInfo | -SwReset> | -hh`
Detailed usage	`mlx5cmd -Dbg -hh`

VF Resources

This tool queries VF MSI-X and EQ count.

> ⚠ This tool is not supported in BlueField 2 SmartNIC mode.

Usage	`mlx5cmd -Dbg -VfResources -Name <adapter name>` `mlx5cmd -Dbg -VfResources -Name <adapter name> -Vf <vf id>`
Detailed usage	`mlx5cmd -Dbg -VfResources -hh`

Features Status Utility

The utility displays the status of driver features.

Usage	`mlx5cmd -Features -hh | -Name <adapter name> [-Json] [-Indentation <count>]`
Detailed usage	`mlx5cmd -Features -hh`

Firmware Capabilities

This tool queries firmware capabilities.

> ⚠ This tool is not supported in BlueField 2 SmartNIC mode.

Usage	`mlx5cmd –Dbg –FwCaps –Name <adapter name>`
	`mlx5cmd –Dbg –FwCaps –Name <adapter name> –Vf <vf id>`
	`mlx5cmd –Dbg –FwCaps –Name <adapter name> –Vf <vf id> –DumpAll`

Detailed usage	`mlx5cmd -FwCaps -hh`

Port Diagnostic Database Register (PDDR)

The tool provides troubleshooting and operational information that can assist in debugging physical layer link related issues.

Usage	`mlx5cmd -Dbg -PddrInfo [-bdf <pci-bus#> <pci-device#> <pci-function#>] | [-Name <adapter name>] | -hh`
Detailed usage	`mlx5cmd -Dbg -PddrInfo -hh`

Software Reset for Adapter Command

The tool enables the user to execute a software reset on the adapter.

Usage	`mlx5cmd -Dbg -SwReset -Name <adapter name>`
Detailed usage	`mlx5cmd -Dbg -SwReset -hh`

Resource Dump

Resource Dump is used to:

- query a menu segments mode:

Usage	`mlx5cmd -Dbg -ResourceDump -Menu -hh | -Name <adapter name>`
Detailed usage	`mlx5cmd -Dbg -ResourceDump -Menu -hh`

Example	**Two menu segment records:**
	`mlx5cmd -Dbg -ResourceDump -Menu -Name "Ethernet"`
	`......`
	`......`
	`_____`
	
	`Segment Type - 0x1301 (EQ_BUFF)`
	
	`Dump Params  Applicability  Special Values`
	`----------  ----------  ----------`
	`index1 -> EQN  Mandatory  N/A`
	`num_of_obj1  N/A  N/A`
	`index2 -> EQE  Optional  N/A`
	`num_of_obj2  Optional  All`
	`_____`
	`_____`
	
	`Segment Type - 0x3000 (SX_SLICE)`
	
	`Dump Params  Applicability  Special Values`
	`----------  ----------  ----------`
	`index1 -> SLICE  Mandatory  N/A`
	`num_of_obj1  N/A  N/A`
	`index2 -> N/A  N/A  N/A`
	`num_of_obj2  N/A  N/A`
	`_____`
	`_____`
	`......`
	`......`

- dump a segments mode:

Usage	`mlx5cmd -Dbg -ResourceDump -Menu -hh | -Name <adapter name>`
Detailed usage	`mlx5cmd -Dbg -ResourceDump -Menu -hh`

Example	mlx5cmd -Dbg -ResourceDump -Dump -Name "Ethernet" -Segment 0x1310 -Index1 1 Output file generated at C:\Windows\temp\Mlx5_Dump_Me_Now-7-0-0\PF\dmn-GN-OID-RESDUMP-2020.6.17-19.18.16-Gen6

> ⚠ The tool does not validate any segment parameters, therefore if any of parameter is missing, the tool will recognize it as zero value. In the case of dump failure, the output file will contain an error message. Hence, we recommend using the menu mode before using this command.

The tool will generate a text file at the printed path, (in our case: “ResourceDump_SegType_0x1310.txt”), and the output text file will contain unparsed text-hex values:

```
0x0004fffe 0x00000000 0x00000000 0x101b0fb4
0x0005fffa 0x13100000 0x00000001 0x00000000
0x00000000 0x0001fffb
```

> ⚠ Since the Resource Dump feature is used in DMN to generate a directory, DMN uses a mechanism that limits the number of created directories. For further information, see Cyclic DMN Mechanism.

Packet Pacing Capabilities

This tools query allocated Packet Pacing objects

Usage	mlx5cmd -Dbg -FWPacketPacing -Name <adapter name> mlx5cmd -Dbg -FWPacketPacing -Name <adapter name> -Index <index id> mlx5cmd -Dbg -FWPacketPacing -Name <adapter name> -UID <uid>
Detailed usage	mlx5cmd -FWPacketPacing -hh

Temperature Utility

The tool queries the external ASIC temperature sensor to get temperature readings. It displays the highest temperature among the ASIC diodes on the adapter in Celsius units.

Usage	mlx5cmd -Temperature -hh | [-Name <adapter name>]
Detailed usage	mlx5cmd -Temperature -hh

Get-NetView Utility

This utility allows the user to collect data on system and network configurations for troubleshooting purposes.

> ⚠ The utility is only supported on Windows Server 2016 and above. For more information, please refer to the Microsoft SDN repository documentation.

Usage	The script is available publicly as part of the Microsoft repository at 'https://github.com/Microsoft/SDN/blob/master/Diagnostics/Get-NetView.PS1'. To execute the script, simply run the script from PowerShell. Once the script has completed, it will display the output location.

Display RSS Information

RSS information is now displayed from the driver. On the Hyper-V it will also display Vport's VMMQ configurations.

Usage	`mlx5cmd -Dbg -RssInfo -Name <adapter name> [-Json <file_name.json>]| -hh`

smpquery Utility

smpquery allows querying of various information about the InfiniBand network.

Usage	`mlx5cmd -ib -SmpQuery -help`

Configuration Validator

This tool validates the configuration of registry keys provided in the configuration file.

Usage	`mlx5cmd -ConfigValidator | -Name <Adapter Name> | [-Template] | [-ConfigCompare] | -File <File Name> | -hh`
Detailed usage	`mlx5cmd -ConfigValidator -hh`

Example	Print a Template file:
	`mlx5cmd -ConfigValidator -Name cx4 -Template -File .\at.json`
	Compare driver registry configuration with the one in the file:
	`mlx5cmd -Dbg -ConfigValidator -Name cx4 -ConfigCompare -File .\at.json`

VXLAN Offloading Configuration Utility

This tool will allow the user to configure additional ports for VXLAN offloading. The user can also query the VXLAN ports offload configuration of the adapter.

Usage	`mlx5cmd -Vxlan -hh | -Name <adapter name> [-add_port <port_num> | -del_port <port_num> | -query]`
Detailed usage	`mlx5cmd -Vxlan -hh`
Notes	• VXLAN offloading is a global hardware configuration, therefore any modification applies to all adapter ports. • VXLAN offloading is always configured on the IANA standard VXLAN port, regardless of OS configuration.

AutoLogger

The AutoLogger is a debuggability capability implemented as part of Mlx5Cmd, that automatically collects logs until it detects a trigger defined by the user.

Usage	`mlx5cmd -AutoLogger -hh | [-Name <adapter name>] -TriggerType <type>`
Detailed usage	`mlx5cmd -AutoLogger -hh`

DevX Utility

> ⚠ This feature is supported in NVIDIA® BlueField®-2 devices only. Using the feature on other devices with REAL_TIME timestamping will result in wrong PTP clock.

This feature creates a PTP like ability to let the user sync the clock by getting a PTP similar clock from a DevX commands. To update the system's time, use the value from the "`devx_ptp_query_time`" option. This feature must be used when the REAL_TIME timestamping is enabled.
The DevX API used for this utility is `devx_ptp_create(__deref_out struct devx_ptp_context** ppPtpCtx, __in devx_device_ctx* pDevxCtx, __in uint32_t flags)`.

To enable the feature:

1. Create the PTP context (devx_ptp_create).
2. Query the PTP clock (devx_ptp_query_time).
3. Repeat the process as many times as needed.
4. Delete the devx PTP context (devx_ptp_destroy).

Snapshot Utility

The snapshot tool scans the machine and provides information on the current settings of the operating system, networking and hardware.

> ⚠ It is highly recommended to add this report when you contact the support team.

The snapshot tool can be found at: <installation_directory>\Diagnostic Tools\MLNX_System_Snapshot.exe

The user can set the report location.

To generate the snapshot report:

1. [Optional] Change the location of the generated file by setting the full path of the file to be generated, or by pressing "Set target file" and choosing the directory that will hold the generated file and its name.

2. Click on Generate HTML button.

Once the report is ready, the folder which contains the report will open automatically.

Troubleshooting

You may be able to easily resolve the issues described in this section. If a problem persists and you are unable to resolve it, please contact your NVIDIA® representative or Support.

The chapter contains the following sections:

- General Related Troubleshooting
- System Configuration Related Troubleshooting
- Installation Related Troubleshooting
- InfiniBand Related Troubleshooting
- Ethernet Related Troubleshooting
- Performance Related Troubleshooting
- Virtualization Related Troubleshooting
- Reported Driver Events
- Extracting WPP Traces

General Related Troubleshooting

Issue	Cause	Solution

Link down	The link might be down due to one of the following issues:
	• cable issues,
	• unsupported speeds,
	• configuration issues
		Run mlx5cmd -dbg -pddrinfo and check the following lines in the output presented:
		• Troubleshooting Info:
		  • **Messages:** Indicates the issue that requires attention.
		• Operational Info:
		  • **Active link speed:** The active speed displayed in bit mask. The list of bits are stated below.
		  • **Supported speeds:** The supported speed displayed in bit mask. The list of bits are stated below.
		Ethernet:
		• 100GB: Bits 23-20
		• 56GB: Bit 8
		• 50GB: Bits 31-30; 19-18
		• 40GB: Bits 16-15; 7
		• 25GB: Bits 29-28
		• 20GB: Bit 5
		• 10GB: Bits 14-12; 4-2
		• 1G: Bit 1
		InfiniBand:
		• Bit 0: SDR
		• Bit 1: DDR
		• Bit 2: QDR
		• Bit 4: FDR

System Configuration Related Troubleshooting

Issue	Cause	Solution
Duplicated node GUIDs on two or more machines	Burning the same node GUID on different servers on the same cluster/VLAN.	Make sure that each machine has a unique GUID set.

Installation Related Troubleshooting

Issue	Cause	Solution
The installation of WinOF-2 fails with the following error message: “This installation package is not supported by this processor type. Contact your product vendor".	An incorrect driver version might have been installed, e.g., you are trying to install a 64-bit driver on a 32-bit machine (or vice versa).	Use the correct driver package according to the CPU architecture.
The installation of WinOF-2 fails with the following error message: “This package release does not support ConnectX-4 devices, please see Release notes for versions that support these devices.”	Only ConnectX-4 devices were found on your machine, those devices are not supported by WinOF-2 v3.10 installation package.	Remove ConnectX-4 devices from your machine or install an older version of the driver that supports those devices.

Installation Error Codes and Troubleshooting

Error Code	Description	Troubleshooting
Setup Return Codes		
1603	Fatal error during installation	Contact support
1633	The installation package is not supported on this platform.	Make sure you are installing the right package for your platform For additional details on Windows installer return codes, please refer to: http://support.microsoft.com/kb/229683
Firmware Burning Warning Codes		
1004	Failed to open the device	Contact support
1005	Could not find an image for at least one device	The firmware for your device was not found. Please try to manually burn the firmware.
1006	Found one device that has multiple images	Burn the firmware manually and select the image you want to burn.
1007	Found one device for which force update is required	Burn the firmware manually with the force flag.
1008	Found one device that has mixed versions	The firmware version or the expansion rom version does not match.
Restore Configuration Warnings		
3	Failed to restore the configuration	Please see log for more details and contact the support team

InfiniBand Related Troubleshooting

Issue	Cause	Solution
No link over NVIDIA® ConnectX®-6 IB VF.	Old OpenSM version.	Use UFM Appliance version 4.0 and above as it automatically installs OpenSM v5.4.0. For further information on how to add support for additional devices, please refer to UFM Appliance User Manual.
The InfiniBand interfaces are not up after the first reboot after the installation process is completed.	Port status might be PORT_DOWN: Switch port state might be “disabled” or cable is disconnected.	Enable switch admin or connect cable.
	Port status might be PORT_INITIALIZED: SM might not be running on the fabric.	Run the SM on the fabric.
	Port status might be PORT_ARMED: Firmware issue.	Please contact Support.
	SR-IOV might be enabled with firmware that does not support SR-IOV and IPoIB simultaneously. In this case, the driver will report an error message stating that IPoIB is not supported by the firmware.	Use the mlxconfig tool to disable SR-IOV. Consult the MFT User Manual for further details.

Ethernet Related Troubleshooting

Issue	Cause	Solution
Low performance caused by insufficient number of MSI-X vectors.	The number of MSI-X vectors required by the driver equals the NumberOfCpuCores + 3. In cases where the default number of MSI-X vectors for a PF is 64, but there are more than 64 CPU cores, the driver will generate an event log.	Use mlxconfig tool to increase MSI-X vector allocation (NUM_PF_MSIX) for a PF to avoid sharing of resources (fewer MSI-X vectors would mean sharing of resources). **Note:** mlxconfig is contained in the MFT package.
Low performance	Non-optimal system configuration might have occurred.	See section “Performance Tuning” to take advantage of NVIDIA® 10/40/56 GBit NIC performance.
The driver fails to start.	There might have been an RSS configuration mismatch between the TCP stack and the NVIDIA® adapter.	1. Open the event log and look under "System" for the "mlx5" source. 2. If found, enable RSS, run: "netsh int tcp set global rss = enabled" or a less recommended suggestion (as it will cause low performance): Disable RSS on the adapter, run: *"netsh int tcp set global rss = no dynamic balancing*".
The driver fails to start and a yellow sign appears near the "Mellanox ConnectX- 4/ConnectX-5 Adapter <X>" in the Device Manager display. (Code 10)	Look into the Event Viewer to view the error.	• If the failure occurred due to unsupported mode type, refer section Port Management for the solution. • If the solution isn't mentioned in event viewer, disable and re-enable "Mellanox ConnectX-4/ConnectX-5 Adapter <X>" from the Device Manager display. If the failure resumes, please refer to Support.
No connectivity to a Fault Tolerance team while using network capture tools (e.g., Wireshark).	The network capture tool might have captured the network traffic of the non-active adapter in the team. This is not allowed since the tool sets the packet filter to "promiscuous", thus causing traffic to be transferred on multiple interfaces.	Close the network capture tool on the physical adapter card, and set it on the team interface instead.
No Ethernet connectivity on 10Gb adapters after activating Performance Tuning (part of the installation).	A TcpWindowSize registry value might have been added.	• Remove the value key under HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters\TcpWindowSize or • Set its value to 0xFFFF.
Packets are being lost.	The port MTU might have been set to a value higher than the maximum MTU supported by the switch.	Change the MTU according to the maximum MTU supported by the switch.
NVGRE changes done on a running VM, are not propagated to the VM.	The configuration changes might not have taken effect until the OS is restarted.	Stop the VM and afterwards perform any NVGRE configuration changes on the VM connected to the virtual switch.

Performance Related Troubleshooting

Issue	Cause	Solution
Low performance issues	The OS profile might not be configured for maximum performance.	1. Go to "Power Options" in the "Control Panel". Make sure "Maximum Performance" is set as the power scheme 2. Reboot the machine.
Low SMBDirect performance	The NetworkDirect registry is enabled by default in the NIC but the ECN and/or PFC is not enabled in the switch.	Either enable ECN/PFC in the switch or set NetworkDirect to zero.

General Diagnostic

1. Go to “Device Manager”, locate the Mellanox adapter that you are debugging, right- click and choose “Properties” and go to the “Information” tab:
   - PCI Gen 1: should appear as "PCI-E 2.5 GT/s"
   - PCI Gen 2: should appear as "PCI-E 5.0 GT/s"
   - PCI Gen 3: should appear as "PCI-E 8.0 GT/s"
   - Link Speed: 56.0 Gbps / 40.0Gbps / 10.0Gbps / 100 Gbps
2. To determine if the Mellanox NIC and PCI bus can achieve their maximum speed, it's best to run nd_send_bw in a loopback. On the same machine:
   a. Run "start /b /affinity 0x1 nd_send_bw -S <IP_host>" where <IP_host> is the local IP.
   b. Run "start /b /affinity 0x2 nd_send_bw -C <IP_host>"
   c. Repeat for port 2 with the appropriate IP.
      On PCI Gen3 the expected result is around 5700MB/s
      On PCI Gen2 the expected result is around 3300MB/s
      Any number lower than that points to bad configuration or installation on the wrong PCI slot. Malfunctioning QoS settings and Flow Control can be the cause as well.
3. To determine the maximum speed between the two sides with the most basic test:
   a. Run "nd_send_bw -S <IP_host1>" on machine 1 where <IP_host1> is the local IP.
   b. Run "nd_send_bw -C <IP_host1>" on machine 2.
   c. Results appear in Gb/s (Gigabits 2^30), and reflect the actual data that was transferred, excluding headers.
   d. If these results are not as expected, the problem is most probably with one or more of the following:
      - Old Firmware version.
      - Misconfigured Flow-control: Global pause or PFC is configured wrong on the hosts, routers and switches.
      - CPU/power options are not set to "Maximum Performance".

Virtualization Related Troubleshooting

Issue	Cause	Solution
When enabling the VMQ, in case NVGRE offload is enabled, and a teaming of two virtual ports is performed, no ping is detected between the VMs and/or ping is detected but no establishing of TCP connection is possible.	Might be missing critical Microsoft updates.	Please refer to: http://support.microsoft.com/kb/2975719 "August 2014 update rollup for Windows server RT 8.1, Windows server 8.1, and Windows server 2012 R2" - specifically, fixes.
When running the system from an SR-IOV, The operation of several hardware resources might fail.	Low resources for VF	1. Run the mlxconfig tool, according to the instructions in the "MFT User Manual" that is available via the MFT downloader under https://network.nvidia.com/products/adapter-software/firmware-tools/. 2. Extract the device name from "mst status", select the appropriate size (> 0, 2,4,8), and run the following command: mlxconfig -[device name] set VF_LOG_BAR_SIZE=size

Reported Driver Events

The driver records events in the system log of the Windows server event system which can be used to identify, diagnose, and predict sources of system problems.

To see the log of events, open System Event Viewer as follows:

Right click on My Computer, click Manage, and then click Event Viewer.

or

1. Click start-->Run and enter "eventvwr.exe".
2. In Event Viewer, select the system log. The following events are recorded:

Reported Driver Event Severity: Error

Event ID	Message
0x0002	<Adapter name>: Adapter failed to initialize due to FW initialization timeout.
0x0004	<Adapter name>: device has been configured to use RSS while Windows' TCP RSS is disabled. This configuration prevents the initialization and enabling of the port. You need to either enable Windows' TCP RSS, or configure the adapter's port to disable RSS. For further details, see the README file under the documentation folder.
0x0006	<Adapter name>: Maximum MTU supported by FW <L>.<Y>.<Z>(<q>) is smaller than the minimum value <K>.
0x0008	<Adapter name>: Adapter failed to complete FLR.
0x000C	<Adapter name>: device startup fails due to less than minimum MSI-X vectors available.
0x0042	<Adapter name>: FW health report - ver <Y>, hw 0x<Z>, rfr 0x<K>, callra 0x<L>, var[1] 0x<L>, synd <M>, ext_synd 0x<R>, exit_ptr 0x<G>.

Event ID	Message
0x0045	<Adapter name>: Driver startup fails because minimal IPoIB driver requirements are not supported by FW <Y><Z><F> FW reported: IPoIB enhanced offloads are not supported Please burn a firmware that supports the requirements and restart the ConnectX device.
0x0046	<Adapter name>: Driver startup fails because IPoIB driver is not supported <Y><Z> IPoIB mode is supported only on physical adapter with RSS mode
0x0047	<Adapter name>: Driver startup fails because RDMA device initialization failed, failure <Y>.
0x004C	<Adapter name>: VF #<Y> reached the maximum number of allocated 4KB pages (<Z>). You could extend this limit by configuring the registry key "MaxFWPagesUsagePerVF". For more details, please refer to the user manual document.
0x008a	<Adapter name>: Resiliency - Ignores error that was reported by sensor <Y>(0x<Z>) as a result of reaching the limit (<F>) of resets.Please clear the counters of the Resiliency feature. For more details, please refer to WinOF-2 User Manual.
0x0095	Restart <Adapter name> as a result of error that was reported by sensor <Y>(0x<Z>) Resiliency state: • Restarts count: <F> • Max restarts count: <L>
0x0096	Restart <Adapter name> as a result of error that was reported by sensor <Y>(0x<Z>) Resiliency state: • Restarts count: <F>
0x010b	<Adapter name>: QUERY_HCA_CAP command fails with error <Y>. The adapter card is dysfunctional. Most likely a FW problem. Please burn the last FW and restart the ConnectX device.
0x010c	<Adapter name>: QUERY_ADAPTER command fails with error <Y>. The adapter card is dysfunctional. Most likely a FW problem. Please burn the last FW and restart the ConnectX device.
0x0130	<Adapter name>: FW command fails. op 0x<Y>, status 0x<Z>, errno <F>, syndrome 0x<L>.
0x0133	<Adapter name>: Execution of FW command fails. op 0x<Y>, errno <Z>.
0x014f	<Adapter name>: Driver startup fails because an insufficient number of Event Queues (EQs) is available. (<Y> are required, <Z> are recommended, <M> are available)
0x0153	<Adapter name>: Driver startup has failed due to unsupported port type=<Y> configured on the device. The driver supports Ethernet mode only. Refer to the relevant section in this manual for instructions on how to configure the correct mode.
0x0154	<Adapter name>: Driver startup fails because minimal driver requirements are not supported by FW <Y>.<Z>.<L>. FW reported: • rss_ind_tbl_cap <Q> • vlan_cap <M> • max_rqs <F> • max_sqs <N> • max_tirs <O> Please burn a firmware that supports the requirements and restart the ConnectX device.
0x0155	<Adapter name>: Driver startup fails because maximum flow table size that is supported by FW <Y>.<Z>.<L> is too small (<K> entries). Please burn a firmware that supports a greater flow table size and restart the ConnectX device.
0x0156	<Adapter name>: Driver startup fails because required receive WQE size is greater than the maximum WQEs size supported by FW <Y>.<Z>.<M>. (<F> are required, <O> are supported)

Event ID	Message
0x0157	<Adapter name>: Driver startup fails because maximum WQE size that is supported by FW <Y>.<L>.<M> is too small (<K>). Please burn a firmware that supports a greater WQE size and restart the ConnectX device.
0x0163	NDIS initiated reset on device <Adapter name> has failed.
0x0164	<Adapter name>: FW reported receive engine hang event.
0x0165	<Adapter name>: FW reported transmit engine hang event: vhca_id <Y>, transmit_engine_id <Z>, qpn 0x<F>.
0x016b	Restart <Adapter name> as a result of error that was reported by sensors <Y>(0x<Z>)
0x016e	<Adapter name>: Failed to open Channel Adapter.

Reported Driver Event Severity: Warning

Event ID	Message
0x0003	<Adapter name>: device has been requested for <Y> Virtual Functions (VFs), while it only supports <Z> VFs. Therefore, only <L> VFs will be allowed.
0x0005	<Adapter name>: Jumbo packet value read from registry (<Y>) is greater than the value supported by FW (<Z>). Therefore use the maximum value supported by FW(<q>).
0x0009	<Adapter name>: Jumbo packet value read from registry(<Y>) is invalid. Therefore use the default value (<Z>).
0x000A	<Adapter name>: Q_Key 0x<Y> is not supported. Only default Q_Key(0x<Z>) is supported by FW. **Note:** The adapter will continue to work with the default Q_Key.
0x000F	<Adapter name>: device configures not to use RSS. This configuration may significantly affect the network performance.
0x0010	<Adapter name>: device reports an "Error event" on CQn #<Y>. Since the event type is:<Z>, the NIC will be reset. (The issue is reported in Function <K>).
0x0012	<Adapter name>: Resiliency - The current firmware does not support hardware reset. For more details, please refer to the user manual document.
0x0013	<Adapter name>: device reports a send=<Y> "CQE error" on cqn #<Z> qpn #<M> cqe_error->syndrome <L>, cqe_error->vendor_error_syndrome <N>, Opcode <O> Therefore, the NIC might be reset. (The issue is reported in Function <P>). For more information refer to details.
0x0014	<Adapter name>: device reports an "EQ stuck" on EQn <Y>. Attempting recovery.
0x0015	<Adapter name>: device reports a send completion handling timeout on TxQueue 0x<Y>. Attempting recovery.
0x0016	<Adapter name>: device reports a receive completion handling timeout on RxQueue 0x<Y>. Attempting recovery.
0x0017	<Adapter name>: detected that Head-of-Queue life limit value (<Y>) does not correspond with the Resiliency feature configuration - CheckForHangCQMaxNoProgress = <Z>, SHCheckForHangTimeInSeconds =<F>. CheckForHangCQMaxNoProgress value is increased to <L>. For more details, please refer to WinOF-2 User Manual.
0x0018	<Adapter name>: detected that Head of Queue feature is disabled. It is recommended to enable it in order to prevent the system from hanging. For more details, please refer to WinOF-2 User Manual.
0x0019	<Adapter name>: <Y> value read from registry(<Z>) is invalid. Therefore use the default value (<F>).

Event ID	Message
0x001A	For more details, please refer to the user manual document.
0x001B	<Adapter name>: Shutting Down RDMA QPs with Excessive Retransmissions feature is not supported by FW <Y>. For more details, please refer to the user manual document.
0x00020	Flow control on the device <Adapter name> was not enabled. Therefore, RoCE cannot function properly. To resolve this issue, please make sure that flow control is configured on both the hosts and switches in your network. For more details, please refer to the user manual.
0x00022	<Adapter name>: Setting QoS port default priority is not allowed on a virtual device. This adapter will use the default priority <Y>.
0x00023	<Adapter name>: failed to set port default priority to <Y>. This adapter will use the default priority <Z>.
0x00024	<Adapter name>: DCQCN is not allowed on a virtual device.
0x00025	Dcqcn was enabled for adapter <Adapter name> but FW <Y>.<Z>.<W> does not support it. Dcqcn congestion control will not be enabled for this adapter. Please burn a newer firmware. For more details, please refer to the user manual document.
0x0026	<Adapter name>: failed to set Dcqcn RP/NP congestion control parameters. This adapter will use default Dcqcn RP/NP congestion control values. Please verify the Dcqcn configuration and then restart the adapter.
0x0027	<Adapter name>: device is configured with a MAC address designated as a multicast address: <Y>. Please configure the registry value NetworkAddress with another address, then restart the driver.
0x0029	<Adapter name>: failed to enable Dcqcn RP/NP congestion control for priority <Y>. This adapter will continue without Dcqcn <Y> congestion control for this priority. Please verify the Dcqcn configuration and then restart the adapter.
0x002C	The miniport driver initiates reset on device <Adapter name>.
0x002D	NDIS initiates reset on device <Adapter name>.
0x0034	<Adapter name>: Non-default PKey is not supported by FW. For more details, please refer to the user manual document.
0x0035	<Adapter name>: According to the configuration under the "Jumbo Packets" advanced property, the MTU configured is <Y>. The effective MTU is the supplied value + 4 bytes (for the IPoIB header). This configuration exceeds the MTU reported by OpenSM, which is <Z>. This inconsistency may result in communication failures. Please change the MTU of IPoIB or OpenSM, and restart the driver.
0x0036	<Adapter name>: GRH-based is configured but IPoIB in Virtual Function (VF) is supported only with LID-based. The link will stay down until LID-based is configured.
0x0043	<Adapter name>: RDMA device initialization failure <Y>. This adapter will continue running in Ethernet only mode.
0x0048	<Adapter name>: Dcbx is not supported by FW. For more details, please refer to the User Manual document.
0x0049	<Adapter name>: Head of queue Feature is not supported by the installed Firmware
0x004A	<Adapter name>: "RxUntaggedMapToLossless" registry key was enabled but the device is not configured for lossless traffic. please enable PFC or global pauses.
0x004B	<Adapter name>: Delay drop timer timed out for RQ Index 0x<Y>. Dropless mode feature is now disabled.
0x004D	<Adapter name>: Dropless mode entered. For more details, please refer to the User Manual document.

Event ID	Message
0x004E	<Adapter name>: Dropless mode exited. For more details, please refer to the User Manual document.
0x004F	<Adapter name>: RxUntaggedMapToLossless is enabled. Default priority changed from <Y> to <Z> in order to map traffic to lossless.
0x0050	<Adapter name>: Skipping device (bdf=<Y>:<Z>.<F>), Looks like it's a leftover from KDNET dedicated PF.
0x0051	<Adapter name>: (module <Y>) detects that the link is down. Bad cable was detected, error: <Z>. Please replace the cable to continue working.
0x0052	<Adapter name>: (module <Y>) detects that the link is down. Cable is unplugged. Please connect the cable to continue working.
0x0053	<Adapter name>: (module <Y>) detected high temperature. Error: <Z>.
0x0054	<Adapter name>: (module <Y>) detects that the link is down. Cable is unsupported. Please connect a supported cable to continue working.
0x0055	<Adapter name>: (module <Y>) detected bad/unreaddable EEPROM.
0x0056	<Adapter name>: (module <Y>) detected an unknown error type.
0x0080	<Adapter name>: RDMA is disabled as a part of the healing policy. For more details, please refer to the Resiliency section in the WinOF-2 User Manual.
0x0097	<Adapter name>: Failed to initialize Resiliency mechanism as a result of <Y> failure, error <Z>.
0x0107	<Adapter name>: Firmware version <Y>.<Z>.<F> is below the minimum FW version recommended for this driver. Minimum recommended Firmware version for this driver: <Y>.<Z>.<F> It is recommended to upgrade the FW, for more details, please refer to WinOF-2 User Manual.
0x0132	Too many IPs in-use for RRoCE. <Adapter name>: RRoCE supports only <Y> IPs per port. Please reduce the number of IPs to use the new IPs.
0x0158	<Adapter name>: CQ moderation is not supported by FW <Y>.<Z>.<L>.
0x0159	<Adapter name>: CQ to EQ remap is not supported by FW <Y>.<Z>.<L>.
0x015a	<Adapter name>: PCIe slot power capability was not advertised. Please make sure to use a PCIe slot that is capable of supplying the required power.
0x015b	<Adapter name>: Detected insufficient power on the PCIe slot (<n>W). Please make sure to use a PCIe slot that is capable of supplying the required power.
0x0160	<Adapter name>: VPort counters are not supported by FW <Y>.<Z>.<L>.
0x0161	<Adapter name>: LSO is not supported by FW <Y>.<Z>.<L>.
0x0162	<Adapter name>: Checksum offload is not supported by FW <Y>.<Z>.<L>.
0x0166	<Adapter name>: FW tracer is not supported.
0x0167	<Adapter name>: FW doesn't support trusted VFs, update FW to get more secured VFs.
0x0169	<Adapter name>: Failed to create full dump me now. Dump me now root directory: <Y>, Failure: <Z>, Status: <F>
0x016f	<Adapter name>: Failed to enable NDK with status <Y>.
0x0170	<Adapter name>: Failed to disable NDK with status <Y>.
0x0171	<Adapter name>: RoCE is disabled for the Virtual Functions (VFs) as the FW doesn't support it. For more details, please refer to the User Manual.
0x0173	<Adapter name>: Configuration value cannot be updated for value <Y>.

Event ID	Message
0x0174	<Adapter name>: Registry key DumpMeNowTotalCount must be greater than registry key DumpMeNowPreservedCount, setting new values: [DumpMeNowTotalCount: <Y> - DumpMeNowPreservedCount: <Z>].
0x0175	<Adapter name>: One or more network ports have been powered down due to insufficient/unadvertised power on the PCIe slot. Please refer to the card's user manual for power specifications or contact NVIDIA Networking support.
0x0176	<Adapter name>: (module <Y>) detects that Cable is plugged but the link is down.
0x0178	<Adapter name>: Device dynamic Registry configuration: < > invalid value, refer to user manual for acceptable values.
0x0181	<Adapter name>: Reducing the advertised MaxNumQueuePairs for vPorts to a power of two. Requested: <Y> Set: <Z>.
0x0182	<Adapter name>: Device reports a Send completion handling timeout on TxQueue 0x<Y> of VMQ <Z> . Attempting recovery.
0x0183	<Adapter name>: Device reports a Receive completion handling timeout on RxQueue 0x<Y> Rss table index <Z>VMQ <L> . Attempting recovery.
0x0184	<Adapter name> Firmware does not support the dynamic MSI-X allocation feature.
0x0186	<Adapter name>: DCQCN <X> values read from registry are invalid. Therefore use the default values.
0x0189	<Adapter name>: DCQCN <X> parameter was requested but FW <L>.<Y>.<Z> does not support it. Please burn a newer firmware. For more details, please refer to the user manual document.
0x018a	<Adapter name>: <X>: QP attached to priority <Y>, which is lossy. Why lossy: Configured neither PFC nor Global Pause. Peer: <L>:<M> Local: <N>:<O> More: peer_qpn <P>, local_qpn <Q>
0x018b	<Adapter name>: <X>: QP attached to priority <Y>, which is lossy. Why lossy: Configured PFC with no priorities. Peer: <L>:<M> Local: <N>:<O> More: peer_qpn <P>, local_qpn <Q>
0x018c	<Adapter name>: <X>: QP attached to priority <Y>, which is lossy. Why lossy: Configured PFC with wrong priority. Peer: <L>:<M> Local: <N>:<O> More: peer_qpn <P>, local_qpn <Q>
0x018e	<Adapter name>: Striding RQ parameters are illegal. Striding RQ will be disabled. Bytes per stride should be between 64-8192. Number of strides is: <X>. Receive buffer size is: <Y>.
0x0191	<Adapter name>: PCIe width/speed doesn't match expected value. Expected speed: < > actual speed: < >. Expected width: < > actual width: < >.
0x0192	<Adapter name>: An attempt was made to enable Relaxed Ordering <Read/Write> for Ethernet but the firmware/adapter card does not support this feature or the feature was turned off by the host. Please upgrade the relevant component or contact the host administrator if you are using an SRI-OV VF to enable this capability. To stop seeing this message in the future, disable it in the Windows Registry.
0x01A1	<Adapter name>: The firmware used does not support the "WQE too small" capability. Please update the firmware to enable it.

Event ID	Message
0x0193	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source USER. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x0194	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source RESILIENCY. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x0195	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source PORT. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x0196	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source EQ STUCK. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x0197	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source TX CQ STUCK. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x0198	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source RX CQ STUCK. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x0199	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source CMD TIMEOUT. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x019A	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source CMD FAILED. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.

Event ID	Message
0x019B	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source RESOURCE DUMP. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x019C	<device name>: The dump was created at folder (DMN folder name), due to dump-me-now request with source MP STATS. Dump-me-now dumps are placed by default in folder %SystemRoot%\temp\Mlx5_Dump_Me_Now or a folder that was set by the registry keyword HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Class\{4d36e972-e325-11ce-bfc1-08002be10318}\nnnn\DumpMeNowDirectory.
0x019D	<Adapter name>: Failed to add VXLAN UDP port <X> with status <Y>.
0x019E	<Adapter name>: dump-me-now is triggered due to request with source <X>. Files were not generated since they were not required (Config dump mask=<Y>, Source dump mask=<Z>)
0x01A0	<Adapter name>: DecoupleVmSwitch feature cannot be enabled. Driver: <X>, Port Type: <Y>, FW supports SRIOV: <Z>.

Extracting WPP Traces

WinOF-2 driver automatically dumps trace messages that can be used by the driver developers for debugging issues that have recently occurred on the machine.

The default location for the trace file is:

```
%SystemRoot%\system32\LogFiles\Mlnx\Mellanox-WinOF2-System.etl
```

The automatic trace session is called Mellanox-WinOF2-Kernel.

- To view the session:

  ```
  logman query Melloanox-WinOF2-Kernel -ets
  ```

- To stop the session:

  ```
  logman stop Mellanox-WinOF2-Kernel -ets
  ```

⚠ When opening a support ticket, it is advised to attach the file to the ticket.

Appendixes

The document contains the following appendixes:

- Windows MPI (MS-MPI)

Windows MPI (MS-MPI)

Message Passing Interface (MPI) provides virtual topology, synchronization, and communication functionality between a set of processes. MPI enables running one process on several hosts. With MPI you can run one process on several hosts.

- Windows MPI run over the following protocols:
  - Sockets (Ethernet or IPoIB)
  - Network Direct (ND) Ethernet and InfiniBand

System Requirements

- Install HPC (Build: 4.0.3906.0).
- Validate traffic (ping) between the whole MPI Hosts.
- Every MPI client need to run smpd process which open the mpi channel.
- MPI Initiator Server need to run: mpiexec. If the initiator is also a client, it should also run smpd.

Running MPI

1. Run the following command on each mpi client.

```
start smpd -d -p <port>
```

2. Install ND provider on each MPI client in MPI ND.

```
mpiexec.exe -p <smpd_port> -hosts <num_of_hosts> <hosts_ip_list> -env MPICH_NETMASK <network_ip/subnet>
-env MPICH_ND_ZCOPY_THRESHOLD -1 -env MPICH_DISABLE_ND <0/1> -env MPICH_DISABLE_SOCK <0/1> -affinity
<process>
```

3. Run the following command on MPI server.

Directing MSMPI Traffic

Directing MPI traffic to a specific QoS priority may delayed due to:

- Except for NetDirectPortMatchCondition, the QoS powershell CmdLet for NetworkDirect traffic does not support port range. Therefore, NetwrokDirect traffic cannot be directed to ports 1-65536.
- The MSMPI directive to control the port range (namely: MPICH_PORT_RANGE 3000,3030) is not working for ND, and MSMPI chose a random port.

Running MSMPI on the Desired Priority

Set the default QoS policy to be the desired priority (Note: this prio should be lossless all the way in the switches*)

1. Set SMB policy to a desired priority only if SMD Traffic running.

2. [Recommended] Direct ALL TCP/UDP traffic to a lossy priority by using the “IPProtocolMatchCondition”.

> ⚠ TCP is being used for MPI control channel (smpd), while UDP is being used for other services such as remote-desktop.

Arista switches forwards the pcp bits (e.g. 802.1p priority within the vlan tag) from ingress to egress to enable any two End-Nodes in the fabric as to maintain the priority along the route.

In this case the packet from the sender goes out with priority X and reaches the far end-node with the same priority X.

> ⚠ The priority should be lossless in the switches

To force MSMPI to work over ND and not over sockets, add the following in mpiexec command:

```
-env MPICH_DISABLE_ND 0 -env MPICH_DISABLE_SOCK 1
```

Configuring MPI

Configure all the hosts in the cluster with identical PFC (see the PFC example below).

1. Run the WHCK ND based traffic tests to Check PFC (ndrping, ndping, ndrpingpong, ndpingpong).
2. Validate PFC counters, during the run-time of ND tests, with “Mellanox Adapter QoS Counters” in the perfmon.
3. Install the same version of HPC Pack in the entire cluster.
4. NOTE: Version mismatch in HPC Pack 2012 can cause MPI to hung.
5. Validate the MPI base infrastructure with simple commands, such as “hostname”.

PFC Example

In the example below, ND and NDK go to priority 3 that configures no-drop in the switches. The TCP/UDP traffic directs ALL traffic to priority 1.

- Install DCBX.

```
Install-WindowsFeature Data-Center-Bridging
```

- Remove the entire previous settings.

```
Remove-NetQosTrafficClass
Remove-NetQosPolicy -Confirm:$False
```

- Set the DCBX Willing parameter to false as Mellanox drivers do not support this feature.

```
Set-NetQosDcbxSetting -Willing 0
```

- Create a Quality of Service (QoS) policy and tag each type of traffic with the relevant priority.
  In this example we used TCP/UDP priority 1, ND/NDK priority 3.

```
New-NetQosPolicy “SMB" -NetDirectPortMatchCondition 445 -PriorityValue8021Action 3
New-NetQosPolicy “DEFAULT" -Default -PriorityValue8021Action 3
New-NetQosPolicy “TCP" -IPProtocolMatchCondition TCP -PriorityValue8021Action1
New-NetQosPolicy “UDP" -IPProtocolMatchCondition UDP -PriorityValue8021Action 1
```

- Enable PFC on priority 3.

```
Enable-NetQosFlowControl 3
```

- Disable Priority Flow Control (PFC) for all other priorities except for 3.

```
Disable-NetQosFlowControl 0,1,2,4,5,6,7
```

- Enable QoS on the relevant interface.

```
Enable-netadapterqos -Name
```

Running MPI Command Examples

- Running MPI pallas test over ND.

```
> mpiexec.exe -p 19020 -hosts 4 11.11.146.101 11.21.147.101 11.21.147.51
11.11.145.101 -env MPICH_NETMASK 11.0.0.0/
255.0.0.0 -env MPICH_ND_ZCOPY_THRESHOLD -1 -env MPICH_DISABLE_ND 0 -env
MPICH_DISABLE_SOCK 1 -affinity c:\\test1.exe
```

- Running MPI pallas test over ETH.

```
> exempiexec.exe -p 19020 -hosts 4 11.11.146.101 11.21.147.101 11.21.147.51
11.11.145.101 -env MPICH_NETMASK 11.0.0.0/
255.0.0.0 -env MPICH_ND_ZCOPY_THRESHOLD -1 -env MPICH_DISABLE_ND 1 -env
MPICH_DISABLE_SOCK 0 -affinity c:\\test1.exe
```

Document History

Release Notes History

Release Notes Change Log History

Feature/Change	Description
Rev 23.4.50020 (DRV 23.4.26054)	
Multi Prio Sent Queue	Added a "MultiPrioSq" registry key to enable and disable the MultiPrioSq feature. The "MultiPrioSq" controls the SL-Diff feature in which the firmware modifies the priority (SL - Service Level) of the HW send-queue to match the one of the sent packet (QoS). For further information, refer to Multi Prio Send Queue.
Counters	Added two new error counters (`Generated Packets dropped due to steering failure` & `Handled Packets dropped due to steering failure`) to "Mellanox WinOF-2 Diagnostics Ext 1“ and "Mellanox WinOF-2 VF Diagnostics“ counter sets. For further information, refer to Adapter Cards Counters.
NVIDIA BlueField-3 DPU	Added support for NVIDIA BlueField-3 DPU devices.
RSHIM	The RShimCmd Tool for supports 2 additional capabilities: • Boot Mode: Sets the next boot mode in DPU. • Timeout: Sets the value of timeout in -PushImage command. Additionally, updated the P*rint of the Driver's and DPU's Variables* when the verbosity is 1. For further information, refer to RShim Drivers and Usage.
Bug Fixes	See Bug Fixes.
Rev 3.20.50010 (DRV 3.20.25915)	
DevxFsRules DPDK	Enables the creation of flow rules with patterns with the exact match on both the destination and the source ports for UDP and TCP. This new functionality is available via the new bit added to the Regkey DevxFsRules: bit 19 - MLX5_DEVX_FS_RULE_DST_PORTS. **Note:** Both the destination and source ports must be both specified.
Debuggability	Added support for callback of type `KbCallbackTriageDumpData` to collect mst dump as part of live dump, and in case of bugcheck. **Note:** This new capability is supported in Windows Client 10 version 1903 and Windows Server 2022 and above.
RSHIM	RSHIM host driver alignment for all the drivers (Windows/Linux/Arm). For further information, see RShim Drivers and Usage.
Rev 3.10.51000 (DRV 3.10.25798)	
General	Updated the MFT and firmware versions. For the updated version see Supported Network Adapter Cards and MFT Tools.
Rev 3.10.50000 (DRV 3.10.25798)	
Adapter Cards	ConnectX-4 adapter card will no longer be supported as of WinOF-2 v3.10.

Installation Package	As of WinOF-2 v3.10, Windows Server 2012 R2 and Windows 8.1 Client will have a separate installation package from other supported OS.
NDK	Added support for NDK 2.1 (NDIS 6.85) AcceptEx and CompleteConnectEx.
BlueField UEFI System Boot Customizations during Installation	Bluefield's UEFI system boot options and more can be customized during the BFB Installation through the use of configuration parameters in the bf.cfg file. For further information on the bf.cfg file, refer to the BlueField Documentation. For further information, see sections "BlueField UEFI System Boot Customizations during Installation".
Hibernation: ConnectX-6 Dx Active Cooled Cards	Added support for hibernation for ConnectX-6 Dx active cooled card in Windows workstations.
Counters	Added new performance counters: RDMA Connection Errors & CM DREQ. For further information see, Adapter Cards Counters.
Counters	Added new counters to query ICMC counters. For further information see, Mellanox WinOF-2 ICMC Diag Counters Ext1.
ZTT Register	Added support for setting a ZTT operation flag using a dynamic registry key. **Note:** Before sending the get/set request for the ZTT registry [mlx5cmd], make sure ZTT is supported by the firmware. For further information on the registry key, see "EnableZtt" in section Performance Registry Keys.
Bug Fixes	See Bug Fixes
Rev 3.0.50000 (DRV 3.0.25668)	
FwWaWqeTooSmallMode	Added a new mode to FwWaWqeTooSmallMode. In this mode the firmware will not generate a WQE completion and will discard the arrived packet (Discard Wqe No Cqe) when VF WQE is too small . By exposing the WqeTooSmall counter in the VM, the new mode enables the counter to count the number of times this action is successfully performed by the firmware.
NVIDIA BlueField-2 NIC Mode	The driver now supports NVIDIA BlueField-2 devices over IPoB running in NIC mode. In this mode, the DPU behaves exactly like an adapter card from the perspective of the external host. For further information, see section *"NVIDIA BlueField-2 DPU NIC Operation Mode"* in the NVIDIA BlueField-2 Firmware Release Notes.
Port Traffic Counters	Added new counters to "Mellanox WinOF-2 VF Port Traffic". For further information see section Adapter Cards Counters.
Diagnostic Counters	Added a new counter `"Packets Received dropped due to lack of receive WQEs"` to the "Mellanox WinOF-2 VF Diagnostics" counters set. For further information see section Adapter Cards Counters.
DOCA	As of WinOF-2 v3.0, the DOCA module will be published on a separate package. DOCA comm channel and socket relay applications were moved to the new DOCA, thus they are removed from the WinOF-2 package. **Note:** A DOCA developer should install the DOCA SDK package and not the DevX SDK package. For more information please see the DOCA documentation.
Bug Fixes	See Bug Fixes.
Rev 2.90.50010 (DRV 2.90.25506)	
Adapter Cards	Added support for NVIDIA ConnectX-7 adapter cards.

NVIDIA BlueField-2 NIC Mode	The driver now supports NVIDIA BlueField-2 devices running in NIC mode. In this mode, the DPU behaves exactly like an adapter card from the perspective of the external host. For further information, see section *"NVIDIA BlueField-2 DPU NIC Operation Mode"* in the NVIDIA BlueField-2 Firmware Release Notes.
DOCA Socket Relay	Added support for an AF_UNIX connection between applications that run on the Windows host and services that run inside the the DPU. For further information, see DOCA Socket Relay.
DOCA Communication Channel API	DOCA communication channel API in NVIDIA BlueField-2 SmartNIC adapter cards is now at GA level. For further information, see DOCA Communication Channel API.
DPKD DevX	Added new interfaces for DevX library to set: • the promiscuous mode with the two modes: ALL, MC • MTU (limited to Host case, and port MTU >= 1522) For further information, see Offload Capabilities for Windows DPDK.
NVIDIA BlueField-2 DevX	Added support for flex parser to the DevX steering rule.
Enhanced Connection Establishment	Enhanced Connection Establishment (ECE) is a new negotiation scheme introduced in IBTA v1.4 to exchange extra information about nodes capabilities and later negotiate them at the connection establishment phase. ECE is intended for RDMA connection, i.e., it works in ND and NDK connections. For further information, see Enhanced Connection Establishment.
CM Packets	This new capability provides the option of ignoring RoCE connections that have differences in the source IP address. Now the user can decide whether or not to allow differences between the IB header source IP and the private data source IP. To activate this option, the 'EnableCmAntiSpoofing' key must be set to 1 (default value is 0). For further information, see "RDMA Registry Keys" in section Configuring the Driver Registry Keys.
DriverVersion Utility	Changed the output of mlx5cmd -driverVersion command. Instead of presenting the OS name, now the tool will present the OS build number + Server\Client information.
Rivermax	Enabled multiple Rivermax applications to listen on the same stream.
Bug Fixes	See Bug Fixes
Rev 2.80.50000 (DRV 2.80.25134)	
Operating Systems	Removed support for Windows Server 2012.
DOCA Communication Channel API	[Alpha Level] Added support for DOCA communication channel API in NVIDIA BlueField-2 SmartNIC adapter cards. DOCA Communication Channel API allows developers to write client applications running on Windows native hosts or Windows Virtual Machines to exchange messages with service applications running on BlueField-2 DPU. For further information see DOCA Communication Channel API.
GPU Memory Registration	Added support for large GPU memory registration through `ibv_reg_mr()` and `ibv_reg_mr_iova2()`.
DPU Time Service	Added support for "PTP like" for BlueField-2 devices when using the REAL_TIME timestamping ability. For further information, see DevX Utility.
Hardware QoS Offload	Added Hardware QoS Offload support to allow egress bandwidth management entirely in the hardware. For further information, see Hardware QoS Offload.

Reduced RoCE Latency for SMBDirect using Two Queues: One for FRWR and the other for Send	Added a new configuration value (`'NdkFmrDedicatedQp'`) to control whether or not a separated QP is used for NDK fast-register operations. The dedicated QP will improve latency on systems that phase latency issues. **Warning:** The feature is useful **only** for SMBDirect and can be harmful for other applications using NDK. **Note:** This capability is supported in ConnectX-4 and ConnectX-4-Lx adapter cards when in Ethernet (RoCE) mode. **Note:** This value is OFF by default. For further information, see Configuring the Driver Registry Keys.
Asymmetric Number of VFs per PF	This new capability allows the user to set an asymmetric Number of VFs per PF using IOCTL. When using IOCTL to open a second PF on the same port, the new PF will be opened with '0' VFs. The number of VFs can be modified by using the mlxconfig tool, as long as the total number of VFs (on all PFs together) will not exceed the maximum number of VFs allowed.
GPUDirect for Windows	GPUDirect support in Windows is now at GA level. For further information, see GPUDirect.
Bug Fixes	See Bug Fixes History
Rev 2.70.53000 (DRV 2.70.24739)	
Bug Fixes	See Bug Fixes History
Rev 2.70.51000 (DRV 2.70.24728)	
Bug Fixes	See Bug Fixes History
Rev 2.70.50000 (DRV 2.70.24708)	
GPUDirect for Windows	**[Beta]** Added GPUDirect support in Windows to allow the NIC direct access to the GPU memory. For further information, see GPUDirect.
Operating System	Added support for Windows Server 2022 Operating System.
ND, MlxNdPerf	Added a new performance benchmark tool called MlxNdPerf that replaces all old ND performance tools (e.g. nd_send_bw...). For further information see MlxNdPerf Utility.
Real Time PTP	Added support for fetching the real_time value from the init_segment.
VXLAN Offloading	VXLAN offloading is now supported on multiple ports. Port configuration can be done through the use of "Mlx5Cmd -Vxlan" command and you can configure up to 4 ports for offloading. For further information see VXLAN Offloading Configuration Utility.
Vport, Promiscuous mode, DPDK	Added support for promiscuous mode enablement on a Vport when using DPDK. **Note:** This capability is support in PF only.
PDDR Cable Information	Added support for cable information in ConnectX-6 / ConnectX-6 Dx / ConnectX-6 Lx and Bluefield-2 adapter cards.
DSFP Connector	Added support for DSFP connector.
Bug Fixes	See Bug Fixes History
Rev 2.60.51000 (DRV 2.60.23957)	
Bug Fixes	See Bug Fixes History
Rev 2.60.50000 (DRV 2.60.23957)	
Adapter Cards	Added support for NVIDIA® BlueField SmartNIC at GA level.
Hardware vPort Context	Added the option to dump hardware vPort context using mlx5cmd.

Configuration Validator	This tool validates the configuration of registry keys provided in the configuration file. For further information see Configuration Validator
Link FEC Configuration Utility	The Link FEC Configuration utility provides the ability to query supported link FEC modes by the adapter for the current link speed and for all supported link speeds. For further information see Link FEC Configuration Utility
Packet Pacing Capabilities	This tools query allocated Packet Pacing objects. For further information see Packet Pacing Capabilities
DevX Registry Keys	Added new registry keys that configure the DevX feature. For further information see DevX Registry Keys
NDIS Poll Mode	Windows introduced a new poll mode feature starting NDIS 6.85 onwards. The poll API handles Datapath processing for both TX and/or RX side. When the feature is enabled, the driver registers with NDIS for call backs to poll RX and/or TX data. For further information see NDIS Poll Mode.
smpquery Utility	smpquery allows querying of various information about the InfiniBand network. For further information see smpquery Utility.
Counters	Added the following new counters: • Packets processed in NDIS poll mode • CQ Overrun For further information see Mellanox WinOF-2 Receive Datapath & Mellanox WinOF-2 Transmit Datapath / Mellanox WinOF-2 PCI Device Diagnostic & Mellanox WinOF-2 Diagnostics Extension 1
Non-encapsulated Packets Steering	Non-encapsulated packet handling enables the user to facilitate the following main flows: • Matching by the inner header only (non-encapsulated packets dropped or indicated on default vPort in Promiscuous mode). • Matching encapsulated packets by inner header and non-encapsulated packets when registry GreEnableDualTunneling is configured. • Matching encapsulated packets by outer header and non-encapsulated packets. For further information, see Non-encapsulated Packets Steering.
Driver Events	The following event logs severity status was changed from *"Error"* to *"Warning"* as they are not fatal errors: • MLX_EVENT_LOG_IPOIB_ILLEGAL_Q_KEY (0x000A) • MLX_EVENT_LOG_ILLEGAL_MAC_ADDRESS (0x0027) • MLX_EVENT_LOG_SM_MTU_MISMATCH (0x0035) • MLX_EVENT_ERROR_RESILIENCY_INIT_FAIL (0x0097) • MLX_EVENT_ERROR_DUMP_ME_NOW (0x0169) • EVENT_NDK_FAILED_TO_BE_ENABLED (0x016f) • EVENT_NDK_FAILED_TO_BE_DISABLED (0x0170)
Registry Keys	Added new registry keys to control moving to DPC mode once the maximum RX/TX packet processing limit is reached. For further information, see Performance Registry Keys.
Counters	Removed "Mellanox WinOF-2 VF Internal Traffic Counters" from Virtual Functions. **Note:** Mellanox WinOF-2 VF Internal Traffic Counters are relevant for Physical Functions ONLY.
PCIe Transfer Speed	Added PCIe transfer speed units for event MLX_PCIE_LINK. For further information, see event 0x0191 in Reported Driver Events.
IPoIB Teaming	Added support for IPoIB Teaming in failover mode.
Bug Fixes	See Bug Fixes History

Rev 2.50.50000 (DRV 2.50.23282)	
Adapter Cards	Added support for NVIDIA® Mellanox® ConnectX®-6 Lx adapter card.
Adapter Cards	Added support for NVIDIA® Mellanox® BlueField SmartNIC mode. For further information see NVIDIA BlueField SmartNIC Mode.
Adapter Cards	**[Beta]** Added support for NVIDIA® Mellanox® BlueField-2 SmartNIC adapter card. For further information see NVIDIA BlueField SmartNIC Mode.
Bluefield Settings	When using Bluefield in SmartNIC mode, encapsulation offload is done by the SoC, therefore, the encapsulation offload registry keys does not have any impact . For this reason, the following encapsulation offload registry keys for Bluefield device are not advertised. • *EncapsulatedPacketTaskOffloadNvgre • EncapsulatedPacketTaskOffloadVxlan • *VxlanUDPPortNumber • *EncapOverhead • *EncapsulatedPacketTaskOffload

INF Modifications	As part of disabling the offloads for Bluefield devices, the following changes we applied to the inf:
	• Added new section:
	  • Encap_keys.reg and encap_keys10.0.reg
	• Removed the virt10.0.reg section. The keys that were in this section and moved to encap_keys10.0.reg section are:
	  • *EncapsulatedPacketTaskOffloadNvgre
	  • EncapsulatedPacketTaskOffloadVxlan
	  • *VxlanUDPPortNumber
	  • *EncapOverhead
	• *EncapsulatedPacketTaskOffload key was moved from virt.reg section to Encap_keys.reg

Windows Mode Sleep/ Hibernation Detection	**[Beta]** Added support for device removal while the system is in sleep/ hibernation mode.
VF's DHCP Redirections	This feature forces every received\sent DHCP packet to be redirected to PF, including DHCP packets sent or received for VFs. The detection of a packet as a DHCP is done by checking UDP-Ports 67 and 68. For further information see VF's DHCP Redirections.
Mellanox WinOF-2 PCI Device Diagnostic	The "Available PCI BW" and "Used PCI BW" counters are now deprecated and replaced by "Available PCI BW/Sec" and "Used PCI BW/Sec". For further information see Mellanox WinOF-2 PCI Device Diagnostic.
DevX SDK	Added a DevX SDK executable file to develop code on top of DevX. The capability requires manual installation. For further information see DevX Interface.
Features Status Utility	The utility displays the status of driver features. For further information see Features Status Utility.
Temperature Utility	Mlx5Cmd can now query the external ASIC temperature sensor to get temperature readings using the `"Mlx5Cmd -Temperature"` command. For further information see Temperature Utility.
DSCP Based QoS	DSCP Based QoS enables the user to map DSCP to certain priority. For further information see DSCP Based QoS.
ECN Marking	The driver now updates the IP header with ECN bits based on hints received from the hardware. Additionally, software counters were also added for number of such packets marked and updated by the driver.

Explicit Congestion Notification (ECN) Hint in CQE	In a multi-host system, a single receive buffer is used for all hosts. If one or more host(s) are being congested, the congested host(s) can exhaust the device's receive buffer and cause service degradation for the other host(s). In order to manage this situation, the device can mark the ECN (Explicit Congestion Notification) bits in the IP header for the congested hosts. When ECN is enabled on the host, the host will sense the ECN marking and will reduce the TCP traffic and by that will throttle the traffic. Additionally, software counters were also added for number of such packets marked and updated by the driver. The device is capable to either drop or mark the packet (ECN) based on the Aggressive Mode or Dynamic Mode. For further information see Explicit Congestion Notification (ECN) Hint in CQE.
Additional MAC Addresses in RSS (Native Mode)	This feature allows the user to configure additional MAC addresses for the network adapter without setting the adapter to promiscuous mode. Registering MAC addresses for a network adapter will allow the adapter to accept packets with the registered MAC address. For further information see Additional MAC Addresses for the Network Adapter.
Limiting the Event Log from Flooding the Event Viewer	This new capability prevents the event log from flooding the event viewer in case of an unrecoverable error. This will limit printing similar events up to a defined amount in a defined time range. The users can enable/disable it per their needs. In addition, users can also configure the time range and threshold events count.
Resource Dump (Debuggability)	Resource Dump is a debuggability utility that extracts and prints data segments generated by the firmware/hardware. For further information see Resource Dump.
vPort RSS Configuration	RSS information is now displayed from the driver. On the Hyper-V it will also display Vport's VMMQ configurations. For further information see Display RSS Information.
DMN Offline Debug Improvements (Debuggability)	Improved the driver's debug information by the following means: Dump-Me-Now collected information, driver traces and mlx5cmd commands.
SR-IOV ATS	Added ATS support on Virtual Functions.
Rshim Drivers	Added support for NVIDIA® Mellanox® RShim drivers. The NVIDIA® Mellanox® BlueField® family of SoC devices combines an array of Arm processors coupled with the Mellanox ConnectX® interconnect. Standard Linux distributions run on the Arm cores allowing common open source development tools to be used. The SoC can be accessed via USB (external cable) or PCIe driven by our RShim drivers. RShim drivers provides functionalities like resetting the Arm cores, pushing a bootstream image, networking functionality and console functionality. For further information see .RShim Drivers and Usage v2.80.
Dump Me Now	Added a new registry key "DumpMeNowDumpMask" to control the DMN dumps. For further information see Dump Me Now (DMN) Registry Keys.
Bug Fixes	See Bug Fixes.
Rev 2.40.51000 (DRV 2.40.22533)	
Link Speed	Added support for InfiniBand link with 2x lanes width.
Software Parsing	Software Parsing is now enabled by default. Meaning, the TCP checksum over IP-in-IP encapsulation of IPv4/6 sent packets calculation is now enabled by default.
Bug Fixes	See Bug Fixes.

Rev 2.40.50000 (DRV 2.40.22511)	
Adapter Cards	Added support for ConnectX-6 Dx adapter card.
VM Supported Enhancement	Allows the user to to use a single UAR resource for all Tx priorities queues on the PCI bar, to allow using up to 8 times more RSS queues.
VF Commands Failures Debuggability	The driver generates a detailed (including the command's failure details) Dump-Me-Now file in case it receives error commands from the VF.
Striding RQ	Striding RQ handles the burst of received packets with few PCI access. For further information, see Striding RQ.
Live Firmware Patch Update (LFWP)	Firmware can be patched with critical bugs fixes live with minimal serviceability impact. The patching can be down only within the same major branch. For further information, see Live Firmware Patch.
Relaxed Ordering (Read/Write)	Enables the Relaxed Ordering capability for Window hosts and SR-IOV VFs using a new registry key. Additionally, a second registry key allows the host to control the exposure of the relaxed ordering feature (read and write) for VFs. For further information, see Ethernet Registry Keys.
Nested Virtualization	Added support for Mellanox VF in a VM with Hyper-V enabled
VF BlueFlame	Added support to the ND to execute BlueFlame packet from the VF to achieve low latency.
VF Counters (Anti Spoofing Counters)	Added 3 new counters (see Mellanox WinOF-2 VF Port Traffic) per Virtual Function that count the dropped packets: • Counter for dropped packet dues to invalid source MAC Address • Counter for dropped packet dues to invalid source VLAN • Counter for dropped packet dues to unallowed Ether type For further information, see .Adapter Cards Counters v2.70.
PCI Device Diagnostic Counters	Added the counter below that report the number of PCI width and lane speed as part of PCI diagnostic counter. Additionally, in case of unexpected speed/width, an event log will generated. Mellanox WinOF-2 PCI Device Diagnostic: • PCI link width the current width of PCIe link • PCI link speed the current speed of PCIe link For further information, see .Adapter Cards Counters v2.70.

Silently Dropped Counters	Added the following counters that will count all dropped packets:
	• Mellanox WinOF-2 Port QoS:
	  • Packets received discarded
	• Mellanox WinOF-2 Device Diagnostics:
	  • Internal RQ out of buffer
	• Mellanox WinOF-2 VF Diagnostics:
	  • Quota exceeded command
	  • Send queue priority update flow
	  • Mellanox WinOF-2 Port Traffic:
	  • Packets Received jabbers Error
	  • Packets Received Frame undersize error
	  • Packets Received unsupported opcode error
	  • Packets Received Frame too long Error
	  • Packets Received fragments Error
	For further information, see .Adapter Cards Counters v2.70.

Lossy QPs Reporting	Upon creating of a new ND or NDK QP, the driver will send a message to System Event Log if the assigned QP's priority is lossy (=disabled). The rate of the messages can be limited using the registry parameter `RoCEOnLossyPrioEvtRate`. For further information, see Lossy QPs Reporting
Bug Fixes	See Bug Fixes History
Rev 2.30.51000 (DRV 2.30.21713)	

Windows Client 10	See Bug Fixes History
Rev 2.30.50000 (DRV 2.30.21713)	
NIC Teaming	NIC Teaming allows you to group between one and 32 physical Ethernet network adapters into one or more software-based virtual network adapters. Note: This capability is supported only on Windows 10 Client. For further information, refer to section NIC Teaming.
Zero Touch RoCE	Zero Touch RoCE is at GA level. For further information see Zero Touch RoCE.
Zero Touch RoCE	Added new Diagnostic Counters. For further information, refer to section Mellanox WinOF-2 Diagnostics Extension 1.
Zero Touch RoCE	Added a new parameter to configure Zero Touch RoCE Facilities. For further information, refer to section Zero Touch RoCE Parameters.
ETL	Changed the maximum size of the ETL files (MaxFileSize) to 500MB.
Counters	Added a counter for packets that their checksum was calculated by the software.
Ethernet Registry Keys	Renamed the `RoceMaxFrameSize` registry keys to `RoceFrameSize` and updated its values. For further information, refer to section Ethernet Registry Keys.
Performance Registry Keys	Added a new performance registry key (MaxCallsToNdisIndicate). For further information, refer to section Performance Registry Keys.
Adapter Cards: Link Speed	**[Alpha]** Added support for ConnectX-6 200GbE link speed only when in Force mode (non-Auto-Negotiation).
Adapter Cards	Updated several ConnectX-6 adapter cards description displayed in the Device Manager window.
Bug Fixes	See Bug Fixes History
Rev 2.20.50000 (DRV2.20.21096)	
Zero Touch RoCE	**[Beta]** Zero Touch RoCE enables RoCE to operate on fabrics where no PFC nor ECN are configured. This makes RoCE configuration a breeze while still maintaining its superior high performance. **Note:** This capability is disabled by default. For further information see Zero Touch RoCE.
Driver Events (PCIe Power)	Added 3 new events that indicate the PCIe power status. For further information see the events below in section .Reported Driver Events v2.60. • 0x015a • 0x015b • 0x015c
Hardware Timestamping	Hardware Timestamping is used to implement time-stamping functionality directly into the hardware of the Ethernet physical layer (PHY) using Precision Time Protocol (PTP). Time stamping is performed in the PTP stack when receiving packets from the Ethernet buffer queue. For further information see Hardware Timestamping.
Troubleshooting (GetNetView Tool)	Added support to the mlx5cmd utility for the Microsoft SDN diagnostic script Get-NetView.
mlx5cmd (Registry Keys)	The "mlx5cmd -regkey" command displays now the minimal and maximal optional values for each registry key.
Receive Context Data (RFD)	Increased the minimum size of ReceiveBuffers to 64 to avoid extra barrier at the DataPath.

Event Log	Added new warning event log on the recommended firmware version. If the firmware version running on the system is less than the defined firmware version it will prompt a warning event stating the minimal recommended firmware version. **Note:** The driver supports older firmware version, thus updating the firmware version is not mandatory.
eSwitch Manager (BlueField)	Added support in driver to enable Arm when using BlueField adapter cards to be the eSwitch Manager (eswitch_manager).
Bug Fixes	See Bug Fixes History

Bug Fixes History

⚠ This section includes history of bug fixes of 3 major releases back. For older releases history, please refer to the relevant firmware versions.

Internal Ref.	Issue
3443006	**Description:** Updated the ports' number information in the Device Manager. Now the Information Pane of Properties of NIC adapter displays information of more than 2 ports.
	Keywords: Port Number, NIC
	Detected in version: 3.20.51000
	Fixed in version: 23.4.50020
3418355	**Description:** Fixed a continuous memory allocation issues in NDK.
	Keywords: NDK, continuous memory allocation
	Detected in version: 3.20.51000
	Fixed in version: 23.4.50020
3240588	**Description:** Fixed the results of the latency tests.
	Keywords: ND, performance
	Detected in version: 3.20.51000
	Fixed in version: 23.4.50020
3438140	**Description:** Modified the NDK send operation return sync status. Now upon QP closure (e.g. peer disconnect), the APIs will instead uniformly return async STATUS_ABORT via NdkGetCqResults CQEs (flush error).
	Keywords: NDK, SMB
	Detected in version: 3.20.51000
	Fixed in version: 23.4.50020
3361916	**Description:** Fixed a BSOD that occurred on a client OS after the driver returned from the sleep mode.
	Keywords: Client OS, Sleep, wake-up
	Detected in version: 3.20.51000
	Fixed in version: 23.4.50020
3363420	**Description:** Fixed an issue that caused the `'mlx5cmd -dbg -FwCaps'` command to fail on ConnectX-4 Lx adapter cards.
	Keywords: mlx5cmd, command tool, FW capabilities
	Detected in version: 3.20.51000

Internal Ref.	Issue
	Fixed in version: 23.4.50020
3298557	**Description:** Fixed the status of ZTT feature reported by the `"mlx5cmd -features"` command after driver restart when the feature is supported by the firmware and enabled by the registry key EnableZtt.
	Keywords: ZTT, mlx5cmd
	Detected in version: 3.10.50000
	Fixed in version: 3.20.50010
3318605	**Description:** Auto-negotiation is always enabled even when the user selects a specific speed to support a case where multiple options of the same speed are available.
	Keywords: AN, auto-negotiation, link speed
	Detected in version: 3.10.50000
	Fixed in version: 3.20.50010
3233876	**Description:** Disabled the mlx5cmd option to mention 'Zero Touch Roce' in a VM as this feature is supported only on the Host.
	Keywords: ZeroTouchRoCE ,ZTR, Virtual Machine
	Detected in version: 3.10.50000
	Fixed in version: 3.20.50010
3215011	**Description:** If SR-IOV or the number of VFs is set to 0, the "mlx5cmd -feature" shows VMQOS' status as enabled and rev2 as disabled although VMQOS is actually disabled.
	Keywords: VMQOS mlx5cmd
	Detected in version: 3.10.50000
	Fixed in version: 3.20.50010
2864037	**Description:** When using mlx5cmd "-vportmapping" on dual port devices, occasionally the header is not presented.
	Keywords: VportMapping, mlx5cmd
	Detected in version: 2.80.50000
	Fixed in version: 3.20.50010
3259399 / 3258418	**Description:** Fixed an issue where mlxndperf tool failed with error `0x8000001a` when setting the queue depth to 1 or 2 and on "send" mode.
	Keywords: mlxndperf
	Detected in version: 3.10.50000
	Fixed in version: 3.20.50010
3159828	**Description:** Fixed an issue that caused the driver to report it supports 257 scheduled queues when it actually supports only 256. The issue occurred when: • the firmware version was xx.34.1000 and above • the device supported more than 255 VF • VMQos revision 1
	Keywords: VMQOS Rev 1, Max SQ
	Detected in version: 3.10.50000
	Fixed in version: 3.20.50010
2868062	**Description:** Notification on service side disconnection is not supported.
	Keywords: DOCA
	Detected in version: 2.80.50000
	Fixed in version: 3.10.50000

Internal Ref.	Issue
3215358	**Description:** Fixed an issue that occasionally caused mlxndperf to display low bandwidth when using the Send operation.
	Keywords: mlxndperf
	Detected in version: 3.0.50000
	Fixed in version: 3.10.50000
3216318	**Description:** Fixed an issue that caused the driver to crash if it received `OID_NIC_SWITCH_VPORT_PARAMETERS` with vPort ID greater than the maximum supported vPorts.
	Keywords: OID_NIC_SWITCH_VPORT_PARAMETERS
	Detected in version: 3.0.50000
	Fixed in version: 3.10.50000
2970608	**Description:** Fixed an issue in "mlx5cmd -linkspeed" where the command returned an error although the link was up. This happened when link up time exceeded 5 seconds.
	Keywords: "mlx5cmd -linkspeed"
	Detected in version: 2.90.50010
	Fixed in version: 3.10.50000
3135949	**Description:** Fixed an issue that caused the mlxndperf tool to show low bandwidth results.
	Keywords: mlxndperf
	Detected in version: 3.0.50000
	Fixed in version: 3.10.50000
3158851	**Description:** Mlxndperf.exe improvements: the DestIp parameter is no longer allowed to be run together with the Server flag as the destination address is redundant for the server.
	Keywords: Mlxndperf
	Detected in version: 3.0.50000
	Fixed in version: 3.10.50000
3140361	**Description:** Fixed an issue that prevented the RSHim ethernet driver from reaching 10Mbs.
	Keywords: MLXRSHIM , Ethernet, Performance
	Detected in version: 3.0.50000
	Fixed in version: 3.10.50000
3004352	**Description:** Added missing support for LRO on ConnectX-7.
	Keywords: LRO, ConnectX-7
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
3123107	**Description:** Fixed an issue that allowed using wrong IPv4 DHCP ports for IPv6 DHCP.
	Keywords: DHCP redirect
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
3129686	**Description:** Fixed an issue that displayed the VF ID in the event ID 76 (MLX_EVENT_LOG_VF_REACHED_MAX_PAGES) as the firmware VF ID instead of the Operating System VF ID.
	Keywords: MaxFWPagesUsagePerVF
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000

Internal Ref.	Issue
2769660	**Description:** Fixed an issue that showed the ingress traffic for IB ports in the system counter-sets like "Network Interface" and "network Adapter".
	Keywords: Counter
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
3070631	**Description:** Removed unnecessary bandwidth prints after a connection error in mlxndperf.exe tool
	Keywords: mlxndperf.exe tool
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
3070632	**Description:** Added a new input parameter `'-Delay'` to define the optional delay (in msec) when in "Client in Resilience" mode after driver restart.
	Keywords: mlxndperf.exe tool
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
3049119	**Description:** Number of VFs is limited to 64 when working with VmQos revision 2.
	Keywords: SR-IOV, VMQOS
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
3060792	**Description:** Teaming-over-IPoB in Windows Client over Ethernet in ConnectX-7 adapter card is not supported, thus, the mlx5mux driver does not work over ConnectX-7 adapter cards.
	Keywords: Teaming-over-IPoB, MUX, ConnectX-7
	Detected in version: 2.90.50010
	Fixed in version: 3.0.50000
303781	**Description:** Fixed an issue that resulted in failure to apply QoS parameters on some ConnectX-4/ConnectX-4 Lx single port devices.
	Keywords: QoS
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
3040366	**Description:** Shorten the device name **from** "ConnectX Family mlx5Gen Virtual Function" **to** "ConnectX 5Gen vfunc" to avoid cases of messages being cut event-id 25 where the message was cut to "ctX Family mlx5Gen Virtual Function".
	Keywords: event-viewer
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2781020	**Description:** mlx5cmd "-vportmapping" capability is not supported when using the embedded mode in NVIDIA BlueField devices.
	Keywords: mlx5cmd "-vportmapping", NVIDIA BlueField
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2861814	**Description:** When using `OID_QOS_OFFLOAD_SQ_STATS` to retrieve statistics on an SQ connected to a vPort representing the PF (i.e. the vPort with the physical mac-address), it may count all the traffic on TC0, so the non-TC0 TCs counter will be '0'.
	Keywords: VMqOS statistics

Internal Ref.	Issue
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2864037	**Description:** When using mlx5cmd "-vportmapping" on dual port devices, occasionally the header is not presented.
	Keywords: VportMapping, mlx5cmd
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2870173	**Description:** Querying the SQ stat using the `vfctrl get-queue-info` command on ConnectX-4 and ConnectX-4 Lx devices may cause a BSOD as SQ stat is not supported on these devices.
	Keywords: SQ stat, BSOD, ConnectX-4, ConnectX-4 Lx, HWQOS
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2859027	**Description:** `mlx5cmd -b Smpquery` is not supported on NVIDIA BlueField-2 when in Smart-NIC mode from the host.
	Keywords: NVIDIA BlueField-2, InfiniBand, Smpquery
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2952890	**Description:** Fixed an issue related to DSCP for adapter cards older than ConnectX-6 Dx, that caused counters retrieved by `OID_QOS_OFFLOAD_SQ_STATS` to be with the wrong value.
	Keywords: VMqOS
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2876756	**Description:** Fixed the wrong "Mmps" value printed in the mlxNdPerf tool.
	Keywords: mlxNdPerf tool
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2889930	**Description:** Increased the timeout of loading multiple VF simultaneously to avoid cased of VFs failing to load.
	Keywords: Virtual Function, loading failure
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2899514	**Description:** Fixed an issue that resulted in dynamic changes applied to the registry key "TrustedVFs" not to be applied without performing a driver restart.
	Keywords: Registry key "TrustedVFs"
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010
2951413	**Description:** Increased the default size of resource dump into 2 pages instead of one.
	Keywords: Resource dump
	Detected in version: 2.80.50000
	Fixed in version: 2.90.50010

Internal Ref.	Issue
2727039	**Description:** WinOF-2 installation package will not automatically update the firmware on devices that are using secured firmware.
	Keywords: Firmware upgrade, secure firmware
	Detected in version: 2.70.51000
	Fixed in version: 2.80.50000
2724780	**Description:** On very rare cases a DevX call to create a native MKEY will fail due to fragmented memory in the allocated UMEM causing the UMEM page offset and the mkey page offset to misalign.
	Keywords: DevX, MKEY
	Detected in version: 2.70.51000
	Fixed in version: 2.80.50000
2793039	**Description:** The operation of updating an SQ, when working with VMQoSv2 and more than 100 vPorts attached, might take up to a 1 minute.
	Keywords: SQ, VMQoSv2, vPorts
	Detected in version: 2.70.51000
	Fixed in version: 2.80.50000
2841375	**Description:** Fixed an issue that caused a system crash due to a race between the miniport halt and the link state change event.
	Keywords: Race condition, system crash, IPoIB
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2849359	**Description:** Modified the driver's behaviour to only access secure hardware registers.
	Keywords: Rshim driver
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2755744	**Description:** Removed the global lock option (by default now it is removed) when in blue-flame mode (adapter NDIS config 'BlueFlame'), to prevent cases of heavy contention during concurrent RDMA send/read/write operations.
	Keywords: RDMA ND performance
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000

2752300	**Description:** Shorten the adapter cards name in the Event Viewer to overcome an OS limitation related to long names.
	The following is an example of the new naming format:
	• Old:
	  • “Mellanox ConnectX-4 Adapter #7”
	  • "BlueField ConnectX-6 Dx integrated virtual adapter #4"
	  • “Mellanox ConnectX-6 Lx Adapter”
	• New:
	  • “ConnectX-4 #7”
	  • "BlueField-2 CX6DX #4"
	  • “ConnectX-6 Lx”

	Keywords: Event Viewer: Adapter Cards Names
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000

Internal Ref.	Issue
2731484	**Description:** Fixed a possible system crash when deleting vPort under Rx traffic.
	Keywords: Virtualization, VMQ, VMMQ
	Detected in version: 2.62.50010
	Fixed in version: 2.80.50000
2722843	**Description:** Fixed an issue that caused traffic lose and connection closure when TCP Timestamp option (ts-val) is present and the MSB is set together with RSC.
	Keywords: RSC, ts-val
	Detected in version: 2.20
	Fixed in version: 2.80.50000
2783155	**Description:** Fixed an issue that allowed the installation process to be completed successfully even though one of the drivers was not updated.
	Keywords: Installation
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2700237	**Description:** Added support for large system memory registration through ibv_reg_mr() and ibv_reg_mr_iova2().
	Keywords: System memory registration
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2754300	**Description:** Updated the NDIS version of the Rshim driver to 6.85.
	Keywords: NDIS, Rshim driver
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2770294	**Description:** Changed the default value of "*RSSProfile" to 4 to be aligned with the MSDN requirements. On Windows Server 2019 and above, the new value will not overwrite the inbox driver setting due to the OS limitation.
	Keywords: RSS Profile
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2698839	**Description:** Removed the local IP address in the event massage from the following events: • EVENT_CREATED_LOSSY_QP_NO_CFG(394) • EVENT_CREATED_LOSSY_QP_PFC_NO_CFG(395) • EVENT_CREATED_LOSSY_QP_PFC_WRONG_CFG(396)
	Keywords: Local IP, events
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2690993	**Description:** Fixed a system crash that occurred upon printing information on fatal HW error while using on Arm64 platform.
	Keywords: Arm64, fatal HW error
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2703759	**Description:** Fixed inconsistent values between NDIS counters and NVIDIA WinOF-2 counters when traffic is going through the DevX created resources.
	Keywords: Counters

Internal Ref.	Issue
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2690140	**Description:** Requests of QPs with a string of values set to "max" (e.g., Max Queue Depth + Max SGE counter + Max inline Data size) cannot be processed by the driver as their accumulative size overcomes the WQ maximum size.
	Keywords: ND QP creation
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2701735	**Description:** Disabling one of the GPUs while the application is running could lead to system crash.
	Keywords: GPU
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2683075	**Description:** MPReset handler may be triggered by the OS when using Windows Server 2022 due to some OIDs (e.g. OID_NIC_SWITCH_DELETE_VPORT) that can take a very long time to be completed.
	Keywords: MiniportReset
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2710916	**Description:** Wrong values on the VF-counters are exposed on the Hypervisor. "Packets Received Discarded" and "Packets Received Errors" of the counter-set "Mellanox WinOF-2 VF Port Traffic" represent values taken from the global-device or the PF specific.
	Keywords: Counters
	Detected in version: 2.70.50000
	Fixed in version: 2.80.50000
2727039	**Description:** WinOF-2 installation package will not automatically update the firmware on devices that are using secured firmware.
	Keywords: Firmware upgrade, secure firmware
	Detected in version: 2.70.51000
	Fixed in version: 2.80.50000
2827584	**Description:** Fixed a rare issue that caused the DPDK Windows applications to fail to load due to wrong memory registration by the mlx5.sys driver.
	Keywords: DPDK
	Detected in version: 2.70.50000
	Fixed in version: 2.70.53000
2791350	**Description:** Fixed an issue that caused traffic lose and connection closure when TCP Timestamp option (ts-val) is present and the MSB is set. The aggregated TCP packet created by the RSC used clearing the MSB resulting in loose due to invalid timestamp.
	Keywords: RSC
	Detected in version: 2.70.50000
	Fixed in version: 2.70.53000
2735248	**Description:** Modified the Rshim BUS driver behavior to allow the "bfb push" option even when the driver detected an external USB cable connected that did not expose the virtual ETH and COM devices.
	Keywords: Rshim driver, bfb push

Internal Ref.	Issue
	Detected in version: 2.70.50000
	Fixed in version: 2.70.51000
2722843	**Description:** Fixed an issue that caused the TCP connection to drop when working with RSC and TCP timestamp options.
	Keywords: RSC, TCP timestamp option
	Detected in version: 2.70.50000
	Fixed in version: 2.70.51000
2284224	**Description:** UFM/SM reports a wrong node description.
	Keywords: IPoIB
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2673499	**Description:** Changed the NumaNodeID NDI definition **from** enum **to** min/max to be aligned with MSDN requirements.
	Keywords: NumdNodeID, MSDN
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2673503	**Description:** Changed the default value of "NumRSSQueues" to 16 to be aligned with MSDN requirements on Windows Server 2019 and above. The new value will not overwrite the inbox driver setting due to the OS limitation.
	Keywords: NumRSSQueues, Windows Server 2019 and above
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2680300	**Description:** Fixed a wrong rate limitation (120 Gbps) when using 200GbE adapter cards with port_type of IPoIB.
	Keywords: Performance
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2671192	**Description:** Changed the default value of *FlowControl" to 0 on Windows Server 2022 and above. Now the new value will not overwrite the inbox driver setting due to the OS limitation.
	Keywords: FlowControl, Windows Server 2022 and above
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2627088	**Description:** Updated the maximum value of the DevXFSRules registry key to 0xfffffffe.
	Keywords: DevXFSRules
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2673498	**Description:** Changed the default value of MaxRssProcessors to 16 to be aligned with MSDN requirements. On Windows Server 2019 and above, the new value will not overwrite the inbox driver setting due to an OS limitation.
	Keywords: MaxRssProcessors
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000

Internal Ref.	Issue
2677430	**Description:** Changed the maximal value for VlanID to 4094, 4095 is reserved and should not be used.
	Keywords: VlanID
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2672442	**Description:** Fixed an issue that prevented the package from returning a reboot error code when the MUX driver required reboot.
	Keywords: MUX driver
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2350785	**Description:** Updated the handling of PDDR operational info table to report valid link speed for all devices. The updated registry has 3 mode: • new pcam cap bit enable • new pcam cap bit disable on ConnectX-6 onwards adapter cards • new pcam cap bit disable on ConnectX-5 and older adapter cards
	Keywords: PDDR, pcam cap bit
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2459728	**Description:** Fixed an issue that prevented RshimCmd from enumerating more than 1 device on a system with > 1 DPU.
	Keywords: RshimCmd
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2482298	**Description:** Fixed an issue that caused the NDIS to crash when using version 20282 and PollMode feature. The latest 2022 OS does not have this issue.
	Keywords: NDIS Poll Mode
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2483060	**Description:** Fixed a race condition in the ND filter as a result of a closed connector failure since the connector was asynchronously accessed by the CM disconnect request that handled the QP's flush.
	Keywords: ND connector
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2559765	**Description:** Fixed an issue that caused the RshimCmd tool to crash when incorrect inputs were provided.
	Keywords: RshimCmd
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2579834	**Description:** Fixed the reporting of the OS version that a VF is running on when using `"mlx5cmd -driverversion"`.
	Keywords: DriverVersion Utility
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000

Internal Ref.	Issue
2622264	**Description:** Added a new cable identifier information QSA (QSFP to SFP) to get a more accurate information about the cable from the driver side.
	Keywords: Cable info
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2634546	**Description:** Added support for a new FW cap `log_min_stride_wqe_sz` and initialized an init failure process when the WQE size is too small to avoid HW issue. Now when using striding RQ with a WQE that is too small, the initialization process will fail and a Yellow Bang will appear.
	Keywords: log_min_stride_wqe_sz, striding rq
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2397425	**Description:** Fixed an issue that resulted in the adapter's name being trimmed in the Event log messages when the message size was larger than the Event log message limit size (240 characters).
	Keywords: Event log message size
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2501105	**Description:** Fixed an issue that prevented the package downgrade from replacing mlxdevx.dll in the system folder.
	Keywords: mlxdevx.dll, package downgrade
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
2583088	**Description:** Fixed an incorrect report related to the FwTracer feature on the VF.
	Keywords: FwTracer, Mlx5cmd
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000
1601551	**Description:** Added support for cable information in ConnectX-6 / ConnectX-6 Dx / ConnectX-6 Lx and Bluefield-2 adapter cards.
	Keywords: PDDR Info, ConnectX-6, ConnectX-6 Dx, ConnectX-6 Lx, Bluefield-2
	Detected in version: 2.20
	Fixed in version: 2.70.50000
2347181	**Description:** Although the driver allows attaching HCAs to VM as a physical device using Windows' pass-through facility (Discrete Device Assignment (DDA)), the management tool `mlx5cmd.exe` is partially supported in a VM with passed-through HCAs.
	Keywords: Discrete Device Assignment (DDA), pass-through facility, management tool `mlx5cmd.exe`
	Detected in version: 2.50.50000
	Fixed in version: 2.70.50000
2385017	**Description:** SmpQuery is not functional on dual ports VPI devices when the second port is using Ethernet and RoCE is enabled on that port.
	Keywords: SmpQuery
	Detected in version: 2.60.50000
	Fixed in version: 2.70.50000

Internal Ref.	Issue
2403578	**Description:** Fixed incorrect timestamp in the PCAP file.
	Keywords: mlx5cmd.exe -Sniffer
	Detected in version: 2.50.50000
	Fixed in version: 2.70.50000
2604448	**Description:** Fixed and issue that resulted in Virtual Function of a device with revisionID != 0 failed to load when running over an Operating System other than Windows.
	Keywords: VF
	Detected in version: 2.60.50000
	Fixed in version: 2.60.51000
2368632	**Description:** Fixed an issue that caused SR-IOV to fail when using Windows Server 2012 R2 and WinOF-2 v2.50 driver.
	Keywords: SR-IOV
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
1805972	**Description:** Fixed an issue that caused the SmartNIC and the network adapters to be restarted, and consequently the driver to fail from loading, when the fwreset command was used.
	Keywords: BlueField, MlxFwReset
	Detected in version: 2.40.50000
	Fixed in version: 2.60.50000
2384297	**Description:** Added a protection mechanism against multiple NIC-switch creation requests being sent to the same adapter.
	Keywords: NIC-switch creation
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2078012	**Description:** If the Resource dump is re-enabled, and the VFs executes an error command, and the feature is supported by the firmware, a DMN folder might be created containing the VF failure command data. The unrelated DMN folder can be ignored.
	Keywords: ResourceDump, VF CMD FAIL
	Detected in version: 2.40.50000
	Fixed in version: 2.60.50000
2265031	**Description:** Fixed the minimum and maximum values reported for "EnableRss" registry key.
	Keywords: EnableRss
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2281548	**Description:** Added new counters ("Packets processed in interrupt mode" and "Packets processed in polling mode") to the Transmit DataPath counters.
	Keywords: Counters
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2321629	**Description:** Removed the "modifyteam" option from the from mlx5muxtool. **Note:** The user will have to delete the team and recreate it if its name or mode needs to be changed.
	Keywords: "modifyteam", mlx5muxtool

Internal Ref.	Issue
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2329258	**Description:** Fixed an issue that caused an infinite loop in VF initializing process when getting bad PCI header data.
	Keywords: VF, PCI
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2356474	**Description:** Changed the default value of *PtpHardwareTimestamp to 0, **Note:** The new default value will not overwrite the existing value, the user must change it manually. For more information on the impact of keeping HW timestamp enabled see known issue 2374101.
	Keywords: PtpHardwareTimestamp
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2355210	**Description:** Fixed the version check capability that prevented the MTU from being activated on older WinOF-2 versions such as 1.90.
	Keywords: WqeTooSmallWa
	Detected in version: 2.20
	Fixed in version: 2.60.50000
2356917	**Description:** Mlx5Cmd -RssSniffer now displays the file's location that data is being written to when starting and stopping the sniffer.
	Keywords: Mlx5Cmd -RssSniffer
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2362900	**Description:** Modified the Miniport driver behaviour. Now it sets a queue ID on all NBLs in a chain before notifying NDIS.
	Keywords: Miniport driver, NDIS
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2363760	**Description:** Added support for WinPE basic commands to "Mlx5Cmd".
	Keywords: Mlx5Cmd, WinPE
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2370458	**Description:** Modified the "Mlx5Cmd -RssSniffer" behaviour when the RssSniffer is already running. Now the command will fail and will also return a failure if it is stopped when the RssSniffer is not running.
	Keywords: Mlx5Cmd -RssSniffer
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000
2233169	**Description: [Windows Server 2019 build 19041 Onward]** Fixed an installation failure that occurred when the same driver already exists on the device.
	Keywords: Driver installation
	Detected in version: 2.50.50000
	Fixed in version: 2.60.50000

Internal Ref.	Issue
2327695	**Description:** Removed the lscpi tool from the system snapshot tool.
	Keywords: Snapshot tool, lscpi
	Detected in version: 2.50.50000
	Fixed in version: 2.50.51000
2294165	**Description:** Fixed an issue that resulted in VF corruption when multiple VFs were revoked concurrently from the PF.
	Keywords: VF
	Detected in version: 2.50.50000
	Fixed in version: 2.50.51000
2143037	**Description:** Fixed an issue that caused the QoS priority counter to increase by 1 when TxUntagPriorityTag was enabled.
	Keywords: QoS counters
	Detected in version: 2.20
	Fixed in version: 2.50.51000
2281985 / 2294163	**Description:** Added protection to enable/disable multiple network adapters simultaneously.
	Keywords: [mlx5] CDriver
	Detected in version: 2.50.50000
	Fixed in version: 2.50.51000
2247958	**Description:** The descriptions of the Mellanox WinOf-2 Diagnostics Ext 1 counters are inaccurate, For the updated description, refer to the User Manual --> Adapter Cards Counters --> Mellanox WinOf-2 Diagnostics Ext 1.
	Keywords: TCP QOS
	Detected in version: 2.40.50000
	Fixed in version: 2.50.51000
2081797	**Description:** Fixed a potential performance degradation when both transmit and receive processing occurred on same core when running bidirectional traffic.
	Keywords: RFC2544
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2233067	**Description:** Fixed an issue that caused a BSOD when running "mlx5cmd -ndstat" while the ND connection was closing.
	Keywords: ND, BSOD, mlx5cmd
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2196387	**Description:** Fixed the following inaccurate event log message that appeared when the NIC was installed on an old GEN PCIe slot: `Event ID 0x191 PCIe width/speed doesn't match expected value.`
	Keywords: Even log
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2235059	**Description:** Fixed a crash that occurred due to a race between the SM disconnect action and a multicast join/leave handling action.
	Keywords: IPoIB, Multicast,race, crash

Internal Ref.	Issue
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
1977489	**Description:** The “Available PCI BW" and "Used PCI BW" counters display wrong information.
	Keywords: Counters
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2091921	**Description:** Running NonRss Sniffer with Packet Direct while toggling the RSS On/Off can cause BSOD in Disabling the device when PdRssOn failed due to Device in Error state.
	Keywords: RSS
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2096149	**Description:** Mellanox Device Diagnostics counters do not function properly when using a NIC with two adapters in the following flow: 1. Enable adapter 1 2. Enable adapter 2 3. Query Mellanox Device Diagnostics counters continuously 4. Disable adapter 1 The counters for adapter #2 will stop working and the following error message will be shown in the event log for command failure: `<adapter name>: FW command fails. op 0x821, status 0x4, errno -5, syndrome 0x993ca6.`
	Keywords: Mellanox Device Diagnostics counters
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2172748	**Description:** Added support for SFP Module info reports.
	Keywords: Module info reports
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2203675	**Description:** Fixed an issue that resulted in driver load failure in Windows containers. The driver INF file included the machine.inf and not the pci.inf. Note: machine.inf is available only in desktop.
	Keywords: Driver load failure
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2208551	**Description:** As the driver does not expect zero sized MDLs, its behavior was modified to skip zero sized MDLs.
	Keywords: FastSge
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2178395	**Description:** Fixed an issue that occasionally caused the system to crash due to a race between the DMN and the shutdown process.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2107824	**Description:** Fixed an issue that caused a stuck EQ that used the cmd interface to generate an EQ stuck event to the event log.
	Keywords: cmd, EQ

Internal Ref.	Issue
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2193380	**Description:** Fixed an issue that caused unset dynamic registry keys displaying the value "Unset" instead of a number when using Mlx5Cmd -RegKeys.
	Keywords: Dynamic Registry Keys
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2186275	**Description:** Changed the default of supported DMN Masks such that it ignores VMQoS.
	Keywords: DMN, VMQoS
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2182044	**Description:** Modified the driver's behavior to restore to default values when invalid values are configured dynamically.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2178567	**Description:** Fixed an issue that caused mlx5cmd to display incorrect values for DMN keys, and incorrect event log message on dynamic keys changes, when setting invalid value for DMN registry keys dynamically.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2175583	**Description:** Removed the option to print Cable module information to the VF.
	Keywords: Cables, VF
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2166067	**Description:** Fixed an issue that created empty folders when Dump-Me-Now was enabled but DumpMeNowDumpMask was set to 0.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2164801	**Description:** Fixed an issue that generated empty folders when only Core dump was configured.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2145645	**Description:** Updated the minimum MTU size to 600 for IPoIB.
	Keywords: MTU, IPoIB
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2136172	**Description:** Fixed an issue that caused PDDR Operational Info to show "Enabled manager link width" and "Enabled core to PHY link width" as Unknown.
	Keywords: PDDR

Internal Ref.	Issue
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2110618	**Description:** Updated Mlx5Cmd Sniffer behavior to display the location/name of the pcap file.
	Keywords: Mlx5Cmd Sniffer
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2102267	**Description:** Added a unit for "PCI link speed" under the "Mellanox WinOF-2 PCI Device Diagnostics" performance counter.
	Keywords: Performance counter
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2098237	**Description:** Fixed a wrong output DMN path name. The `"Mlx5Cmd.exe -Dmn"` command showed the default path name like as `"\SystemRoot\temp.."` instead of `"%SystemRoot%\temp\.."`.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2064337	**Description:** Fixed an issue where the actual file size of pcap files was greater than the intended file size specified by the user. Removed the buffer_size argument as a user input.
	Keywords: pcap files
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
1859854	**Description:** Fixed an issue that limited the number of VMs to 124 VMs when working in VMQ mode.
	Keywords: Virtualization
	Detected in version: 2.10
	Fixed in version:
1978788	**Description:** Due to memory allocation issue, an issue with the MST dump might occur.
	Keywords: MST dump memory allocation
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2088202	**Description:** The mlx5cmd FwCaps dumps the MAXIMUM values that can be used by the VF and the CURRENT values set in the PF/SR-IOV VM instead of the CURRENT capabilities of the VF from the host as well and the MAXIMUM values of the host.
	Keywords: FwCaps
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2091921	**Description:** Running NonRss Sniffer with Packet Direct while toggling the RSS On/Off can cause a BSOD during the adapter disabling.
	Keywords: RSS
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000

Internal Ref.	Issue
2092544 / 2090352	**Description:** Incorrect *JumboPacket values (Min, Max and Default) for IPoIB when runningmlx5cmd"-regkeys".
	Keywords: JumboPacket
	Detected in version: 2.40.50000
	Fixed in version: 2.50.50000
2112047	**Description:** A race might occur between the delete vPort and the DMN execution that occasionally may lead to BSOD when resiliency is enabled.
	Keywords: vPort, Dump-Me-Now (DMN), resiliency
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2120059	**Description:** IP-in-IP Checksum offload is not functional when working with VLANs.
	Keywords: IP-in-IP Checksum offload, VLAN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2164141	**Description:** Fixed an issue in DMN, that occasionally caused a system crash when driver startup failed, due to double free.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2177323	**Description:** Improved stuck transmit queue detection. Now it does not report stuck queues in case there is a TX DPC queued.
	Keywords: Resiliency
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2175147	**Description:** Fixed the "mlx5cmd -DMN" command return value to display a "Not supported" status when DMN is disabled.
	Keywords: DMN
	Detected in version: 2.40.51000
	Fixed in version: 2.50.50000
2096149	**Description:** Fixed the Device Diagnostic counters to function properly in the following flow: 1. Enable adapter 1 2. Enable adapter 2 3. Query Mellanox Device Diagnostics for both adapters continuously 4. Disable adapter 1
	Keywords: Device Diagnostic counters
	Detected in version: 2.40.50000
	Fixed in version: 2.40.51000
2081797	**Description:** Fixed a potential performance degradation when both transmit and receive processing occurred on same core when running bidirectional traffic.
	Keywords: RFC2544
	Detected in version: 2.40.50000
	Fixed in version: 2.40.51000
2113541	**Description:** Fixed a race between adding the vPort and the DMN execution, that occasionally led to BSOD when resiliency was enabled.

Internal Ref.	Issue
	Keywords: vPort, Dump-Me-Now (DMN), resiliency
	Detected in version: 2.40.50000
	Fixed in version: 2.40.51000
1859854	**Description:** When working in VMQ mode, the number of VMs is limited to 124 VMs. Whereas when in SRI-OV, the driver supports up to 200 vPorts in ConnectX-4 Lx and 254 vPorts in ConnectX-5.
	Keywords: Virtualization
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1997898	**Description:** Resiliency flow takes more than a minute when using CMD_TOUT as the writing of the file is done only in the mstdump destruction (Teardown), a process that can take time depending on the setup configuration and state.
	Keywords: Resiliency
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2001908	**Description:** Changed the default name of the trace file. The default name will be the driver's service name + Trace.etl. For example: mlnx5Trace.etl, mlx5Trace.etl, etc.
	Keywords: Trace File
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2001954	**Description:** Modified the information printed by the PDDR tool to differentiate between the supported cable speeds and the adapter supported speeds.
	Keywords: Pddrinfo
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2074477	**Description:** Fixed the value of configured MTU showed in event log 53.
	Keywords: MTU
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2077195	**Description:** Numbered the DMN folder, to allow the driver to see 2 or more events from the same source at the same second. As the DMN folder name consists of: hour-min-sec, it prevented the driver for seeing all the errors that were created at the same second under the DMN folder.
	Keywords: DMN
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1817808	**Description:** Fixed an issue that caused memory corruption in case the OS provided continues memory across multiple pages that did not start with offset zero on aligned memory address.
	Keywords: ND
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000

Internal Ref.	Issue
2072336	**Description:** Fixed an issue that prevented the device from being updated with the new driver because the driver was already in the driver store.
	Keywords: Driver installation
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1920768	**Description:** Fixed an issue in Windows Server 2019 that occasionally prevented the VF counters from being displayed correctly in perfmon.
	Keywords: VF counters, perfmon, Windows Server 2019
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1974372	**Description:** Fixed a wrong usage prints in Mlx5CmdDbg, mlx5CmdRegKeys, Mlx5CmdOidStat, and Mlx5CmdMstDump. From `"-bdf <pci-bus#> <pci-device#> <pci-function#>"` to `"-bdf <pci-bus#>.<pci-device#>.<pci-function#>"`
	Keywords: mlx5Cmd
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1979255	**Description:** Fixed an issue in the first ND connection that occurred as a result of a driver restart failure.
	Keywords: ND connection, restart
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2001945	**Description:** Fixed an issue that prevented"mlx5cmd -stat" from showing the current link speed when the speed was not supported or setting the link speed if it was supported by both the device and the cable used.
	Keywords: Link speed
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2002667	**Description:** Fixed an issue that prevented the driver from printing an informative message when ran "mlx5cmd -dbg -swreset" and the resiliency capability was not supported.
	Keywords: Resiliency
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2004368	**Description:** Fixed an issue that displayed the Sniffer default file name as "mlx5sniffer.pcap", and not according to the driver name.
	Keywords: Sniffer
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2060026	**Description:** Fixed a memory leak issue in error flow during driver initialization.
	Keywords: Memory leak
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000

Internal Ref.	Issue
2063808	**Description:** Fixed an issue with the "mlx5cmd -linkspeed -query" command that showed empty link speeds supported by the cable when the link was down and a proper message was shown when the link was down.
	Keywords: mlx5cmd
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2065395	**Description:** Fixed an issue that disabled other trusted VF capabilities whenever a host used a ConnectX-5 adapter card that supports Dynamic MSIX capability.
	Keywords: Dynamic MSIX, ConnectX-5, VFs
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
2059845	**Description:** Fixed an issue caused the driver to report zero link speeds supported when working with firmware older than 1x.18.0240.
	Keywords: Link speed
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1910180	**Description:** ndinstall fails to run when using the `"-d"` or `"-q"` parameters.
	Keywords: ndinstall
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1913056	**Description:** ndinstall does not handle invalid parameters correctly and can return incorrect status.
	Keywords: ndinstall
	Detected in version: 2.30.50000
	Fixed in version: 2.40.50000
1805026	**Description:** When working in IPoIB mode, after restarting the device the following warning may appear in the event log: • `Source: mlx5` • `EventID: 72` • `Warning description: Mellanox ConnectX-5 Adapter: Dcbx is not supported by FW. For more details, please refer to the user manual document.` The warning can be safely ignored, it is relevant only when working in Ethernet mode.
	Keywords: IB ,DCBX
	Detected in version: 2.30.50000
	Fixed in Release: 2.30.51000
-	**Description:** Fixed a corrupted file in the MUX driver package.
	Keywords: MUX driver
	Detected in version: 2.30.50000
	Fixed in Release: 2.30.51000
1890207/1813295	**Description:** Fixed an issue where the number of MSI-X reported by the command `"mlx5cmd -vfresource"` on a Virtual Machine was wrong.
	Keywords: MSI-x, Virtual Function,VF, mlx5cmd
	Detected in version: 2.20
	Fixed in Release: 2.30.50000

Internal Ref.	Issue
1912810	**Description:** Fixed an issue where the values of registry keys that can be changed dynamically were not included in get-netview report.
	Keywords: get-netview
	Detected in version: 2.20
	Fixed in Release: 2.30.50000
1887870	**Description:** Fixed an issue that prevented the driver from enabling the VLAN or changing the VLAN ID when a VF was available on Windows Server 2019.
	Keywords: VLAN, SR-IOV
	Detected in version: 2.20
	Fixed in Release: 2.30.50000
1893083	**Description: [Windows Server 2019]** Fixed an issue that set the WPP session ETL max file size according to the Inbox Driver WMI Autologger session registry key "MaxFileSize" value, when installed the driver on a clean image.
	Keywords: Driver Installation
	Detected in version: 2.20
	Fixed in Release: 2.30.50000
1870769/1829088	**Description:** Disabled the option to issue an event log message when NDK is not supported by the OS.
	Keywords: NDK
	Detected in version: 2.20
	Fixed in Release: 2.30.50000
1805026	**Description:** When working in IPoIB mode, after restarting the driver the following warning will be shown in the event log: • `Source: mlx5` • `EventID: 37` • `Warning description: DCQCN was enabled for adapter Mellanox ConnectX-5 Adapter but FW 16.25.6000 does not support it. DCQCN congestion control will not be enabled for this adapter. Please burn a newer firmware.` The warning can be safely ignored, no action is required.
	Keywords: IB ,DCQCN
	Detected in version: 2.20
	Fixed in Release: 2.30.50000
1629926 (Microsoft Servicing bug ID: 20741009)	**Description:** In some cases, when upgrading the driver, the user will see in the Device Manager a message that requires rebooting the driver. The issue happens due to a change in Windows Server 2019. The issue occurs due to the fact that after the driver is installed, the devices are restarted and then checked to ensure they are ready to be used. If they are not ready, they are marked as needing reboot. In Windows Server 2019, this function was overhauled a bit and it is now executed too quickly, although the driver installation is completed, the networking stack is still finishing up its configuration of the device. This causes PNP to think the device did not start up properly and tags it as needing reboot.
	Keywords: Windows Server 2019, Driver upgrade, reboot
	Detected in version: 2.10
	Fixed in Release: 2.30.50000

User Manual Revision History

Date	Revision	Section	Description
July 31, 2023	23.7	AutoLogger	New section
		NIC Health Monitor	New section
		Adapter Cards Counters	Updated the "Mellanox WinOF-2 Transmit Datapath Counters" table.
April 30, 2023	23.4	Multi Prio Send Queue	New section
		Adapter Cards Counters	Added the following new error counters: • Generated Packets dropped due to steering failure • Handled Packets dropped due to steering failure to "Mellanox WinOF-2 Diagnostics Ext 1“ and "Mellanox WinOF-2 VF Diagnostics“ counter sets.
		RShim Drivers and Usage	Updated "RShimCmd Tool", added: • *Boot Mode* • *Timeout* and updated: *Print of the Driver's and DPU's Variables*
		Fabric Performance Utilities	Added "Latency" capability
September 30, 2022	3.1	VF Monitoring	New section
		VF Monitoring Registry Keys	New section
		Mellanox WinOF-2 ICMC Diag Counters Ext1	New section
		Mellanox WinOF-2 Diagnostics Ext 1	Added the "CM DREQ" counter.
August 02, 2022	3.0	Mellanox WinOF-2 VF Diagnostics	Added the following counters: • Packets Received WQE too small • CQ Overrun • Packets Received dropped due to lack of receive WQEs
		Mellanox WinOF-2 VF Port Traffic	Updated the "RDMA Bytes/Packets IN/ RDMA Bytes/Packets OUT" content.
		DOCA	Removed the section.
April 30, 2022	2.90	Enhanced Connection Establishment	New section
		DOCA Socket Relay	New section
		Offload Capabilities for Windows DPDK	New section
		Installing WinOF-2 Driver	Updated the Custom Setup screenshot to include the new DOCA Tools
November 30, 2021	2.80	Hardware QoS Offload	New section
		DevX Utility	New section
		DOCA Communication Channel API	New section
		GPUDirect	Added feature limitation.
		Configuring the Driver Registry Keys	Added registry key "NdkFmrDedicatedQp" to the *RDMA Registry Keys* section.
October 28, 2021	2.70.53000	All	No changes to the User Manual

Date	Revision	Section	Description
July 13, 2021	2.70.51000	All	No changes to the User Manual
June 30, 2021	2.70	MlxNdPerf Utility	New section
		VXLAN Offloading Configuration Utility	New section
		GPUDirect	New section
		DevX Registry Keys	Updated the `DevxFsRules` registry key's values.
		Mellanox WinOF-2 Port QoS	Updated the description of the following counters: • Sent Pause Frames • Sent Pause Duration • Received Pause Frames • Received Pause Duration
January 04, 2021	2.60	Accessing DPU From Host	New section
		Configuration Validator	New section
		Link FEC Configuration Utility	New section
		Packet Pacing Capabilities	New section
		DevX Registry Keys	New section
		NDIS Poll Mode	New section
		smpquery Utility	New section
		Command Line Based Teaming Configuration	Updated section
		Ethernet Registry Keys	Added DisableLocalLoopbackFlags key
		Mellanox WinOF-2 Receive Datapath & Mellanox WinOF-2 Transmit Datapath / Mellanox WinOF-2 PCI Device Diagnostic & Mellanox WinOF-2 Diagnostics Extension 1	Added the following new counters: • Packets processed in NDIS poll mode • CQ Overrun
		Reported Driver Events	Changed the events below severity status from Error to Warnings: • MLX_EVENT_LOG_IPOIB_ILLEGAL_Q_KEY (0x000A) • MLX_EVENT_LOG_ILLEGAL_MAC_ADDRESS (0x0027) • MLX_EVENT_LOG_SM_MTU_MISMATCH (0x0035) • MLX_EVENT_ERROR_RESILIENCY_INIT_FAIL (0x0097) • MLX_EVENT_ERROR_DUMP_ME_NOW (0x0169) • EVENT_NDK_FAILED_TO_BE_ENABLED (0x016f) • EVENT_NDK_FAILED_TO_BE_DISABLED (0x0170)

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. Neither NVIDIA Corporation nor any of its direct or indirect subsidiaries and affiliates (collectively: “NVIDIA”) make any representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice. Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer (“Terms of Sale”). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer’s own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer’s product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA, the NVIDIA logo, and Mellanox are trademarks and/or registered trademarks of NVIDIA Corporation and/or Mellanox Technologies Ltd. in the U.S. and in other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright
© 2023 NVIDIA Corporation & affiliates. All Rights Reserved.