

Network troubleshooting

Network adapter diagnostics and common issues

Lenovo

Network troubleshooting overview

When network issues occur, the following components should be checked:

- Network adapter
- Network switch
- SFP and cables
- Network configurations

The following section covers troubleshooting actions for the network adapter and SFP.

Note: Switch port status and other switch configuration issues might also cause the solution to break. If the switch utilized in the solution being serviced is a Lenovo switch, please refer to the corresponding switch course for problem determination and troubleshooting details.

Network adapter identification

All network interfaces are connected to the system board, so they can be identified in several ways.

- Use the `ncc hardware_info` command to identify network adapters.

[Click here to see an example.](#)

- Use the `lspci` command to check the PCI information on the host.

- AHV: `lspci | grep Ether` or `lspci -nk | grep -i kernel` (shows all kernel drivers)

[Click here to see an example.](#)

- ESXi: `lspci | grep vmnic` or `esxcli network nic list` or `esxconfig-nics -l`

[Click here to see an example.](#)

Network adapter identification

All network interfaces are connected to the system board, so they can be identified in several ways.

- Use the `ncc hardware_info` command to identify network adapters.

[Click here to see an example.](#)

- Use the `lspci` command to check the PCI information on the host.

- AHV: `lspci | grep Ether` or `lspci -nk | grep -i kernel` (shows all kernel drivers)

[Click here to see an example.](#)

- ESXi: `lspci | grep vmnic` or `esxcli network nic list` or `esxconfig-nics -l`

[Click here to see an example.](#)

```
#ncc hardware_info show_hardware_info | awk '/NIC/{f=1}'
+-----+
| Location | eth0 |
| Device name | eth0 |
| Driver name | igb |
| Firmware version | 1.63, 0x80000a05 |
| Mac address | 0C:C4:7A:64:EF:C0 |
| Manufacturer | Intel Corporation I350 Gigabit Network Connection (rev 01)|
| Driver Version | 5.3.0-k |
+-----+
```

Network adapter identification

All network interfaces are connected to the system board, so they can be identified in several ways.

- Use the `ncc hardware_info` command to identify network adapters.

[Click here to see an example.](#)

- Use the `lspci` command to check the PCI information on the host.

- AHV: `lspci | grep Ether` or `lspci -nk | grep -i kernel` (shows all kernel drivers)

[Click here to see an example.](#)

- ESXi: `lspci | grep vmnic` or `esxcli network nic list` or `esxconfig-nics -l`

[Click here to see an example.](#)

```
# lspci | grep Ether
03:00.0 Ethernet controller: Intel Corporation I350 Gigabit Network Connection (rev 01)
03:00.1 Ethernet controller: Intel Corporation I350 Gigabit Network Connection (rev 01)
81:00.0 Ethernet controller: Intel Corporation 82599ES 10-Gigabit SFI/SFP+ Network Connection (rev 01)
81:00.1 Ethernet controller: Intel Corporation 82599ES 10-Gigabit SFI/SFP+ Network Connection (rev 01)
```

OR

```
lspci -nk | grep -i kernel (shows all kernel drivers)
```

Network adapter identification

All network interfaces are connected to the system board, so they can be identified in several ways.

- Use the `ncc hardware_info` command to identify network adapters.

[Click here to see an example.](#)

- Use the `lspci` command to check the PCI information on the host.

- AHV: `lspci | grep Ether` or `lspci -nk | grep -i kernel` (shows all kernel drivers)

[Click here to see an example.](#)

- ESXi: `lspci | grep vmnic` or `esxcli network nic list` or `esxconfig-nics -l`

[Click here to see an example.](#)

Scroll down for more information.

```
[root@host:~] lspci | grep vmnic
0000:03:00.0 Network controller: Intel Corporation I350 Gigabit Network Connection [vmnic0]
0000:03:00.1 Network controller: Intel Corporation I350 Gigabit Network Connection [vmnic1]
0000:81:00.0 Network controller: Intel Corporation 82599 10 Gigabit Dual Port Network Connection [vmnic2]
0000:81:00.1 Network controller: Intel Corporation 82599 10 Gigabit Dual Port Network Connection [vmnic3]
```

OR

```
[root@host:~] esxcli network nic list
Name PCI Device Driver Admin Status Link Status Speed Duplex MAC Address MTU
```

Definition

Network adapter identification

All network interfaces are connected to the system board, so they can be identified in several ways.

- Use the `ncc hardware_info` command to identify network adapters.

[Click here to see an example.](#)

- Use the `lspci` command to check the PCI information on the host.

- AHV: `lspci | grep Ether` or `lspci -nk | grep -i kernel` (shows all kernel drivers)

[Click here to see an example.](#)

- ESXi: `lspci | grep vmnic` or `esxcli network nic list` or `esxconfig-nics -l`

[Click here to see an example.](#)

Scroll down for more information.

```
[root@host:~] esxcli network nic list
Name PCI Device Driver Admin Status Link Status Speed Duplex MAC Address MTU
Description
-----
vmnic0 0000:03:00.0 igbn Down Down 0 Half 0c:c4:7a:64:f0:ae 1500 Intel
Corporation I350 Gigabit Network Connection
vmnic1 0000:03:00.1 igbn Down Down 0 Half 0c:c4:7a:64:f0:af 1500 Intel
Corporation I350 Gigabit Network Connection
vmnic2 0000:81:00.0 ixgbe Up Up 10000 Full 0c:c4:7a:bb:11:12 9000 Intel(R) 82599
```


Network adapter identification

All network interfaces are connected to the system board, so they can be identified in several ways.

- Use the `ncc hardware_info` command to identify network adapters.

[Click here to see an example.](#)

- Use the `lspci` command to check the PCI information on the host.
 - AHV: `lspci | grep Ether` or `lspci -nk | grep -i kernel` (shows all kernel drivers)

[Click here to see an example.](#)

- ESXi: `lspci | grep vmnic` or `esxcli network nic list` or `esxcfg-nics -l`

[Click here to see an example.](#)

Scroll down for more information.

```
[root@host:~] esxcfg-nics -l
Name PCI Driver Link Speed Duplex MAC Address MTU Description
vmnic0 0000:03:00.0 igbn Down 0Mbps Half 0c:c4:7a:64:f0:ae 1500 Intel Corporation I350 Gigabit
Network Connection
vmnic1 0000:03:00.1 igbn Down 0Mbps Half 0c:c4:7a:64:f0:af 1500 Intel Corporation I350 Gigabit
Network Connection
vmnic2 0000:81:00.0 ixgbe Up 10000Mbps Full 0c:c4:7a:bb:11:12 9000 Intel(R) 82599 10 Gigabit Dual
Port Network Connection
vmnic3 0000:81:00.1 ixgbe Up 10000Mbps Full 0c:c4:7a:bb:11:13 9000 Intel(R) 82599 10 Gigabit Dual
Port Network Connection
```


Network adapter diagnostics

The primary tool for interacting with network adapters (NICs) on a node is the `ethtool` command. The command probes the kernel driver for the NIC and returns information on the driver as well as hardware state.

Example commands for NIC hardware diagnostics:

- `#ethtool <nic>` - Returns the connection state of the NIC. If the NIC accepts pluggable optics, it will report on the optic type. (Click [HERE](#) to see the command output.)
- `#ethtool -i <nic>` - Returns information on the NIC driver version. (Click [HERE](#) to see the command output.)
- `#ethtool -S <nic>` - Returns all the port statistics tracked by the kernel driver. These counters are queried by AOS to track activity statistics and connection health for each NIC. Counters such as **rx_missed_errors** and **rx_crc_errors** are tracked, and if their rate of change is too high, an alert will be generated in Prism. (Click [HERE](#) to see the command output.)

Note: Nutanix generates an alert when the percentage of errors seen on a NIC reaches 1% of the total packets in a one-hour period. The alert is displayed in the Prism Health tab as **NIC error rate is high**.



Output of #ethtool <nic> Diagnostics

```
#[root@host ~]# ethtool eth2
Settings for eth2:
    Supported ports: [ FIBRE ]
    Supported link modes: 10000baseT/Full
    Supported pause frame use: Symmetric
    Supports auto-negotiation: No
    Advertised link modes: 10000baseT/Full
    Advertised pause frame use: Symmetric
    Advertised auto-negotiation: No
    Speed: 10000Mb/s
    Duplex: Full
    Port: Direct Attach Copper
    PHYAD: 0
    Transceiver: external
    Auto-negotiation: off
    Supports Wake-on: d
    Wake-on: d
    Current message level: 0x00000007 (7)
    drv probe link
    Link detected: yes
```



Network adapter diagnostics

The primary tool for interacting with network adapters (NICs) on a node is the `ethtool` command. It returns the kernel driver for the NIC and returns information on the

Output of `#ethtool -i <nic>`

```
[root@host ~]# ethtool -i eth2
driver: ixgbe
version: 5.0.4
firmware-version: 0x800006da, 255.65535.255
bus-info: 0000:81:00.0
supports-statistics: yes
supports-test: yes
supports-eeprom-access: yes
supports-register-dump: yes
supports-priv-flags: yes
```

change is too high, an alert will be generated in Prism. (Click [HERE](#) to see the command output.)

Note: Nutanix generates an alert when the percentage of errors seen on a NIC reaches 1% of the total packets in a one-hour period. The alert is displayed in the Prism Health tab as **NIC error rate is high**.



Network adapter diagnostics

Output of `#ethtool -S <nic>`

h network adapters (NICs) on a node is the `ethtool`

```
[root@host ~]# ethtool -S eth2 | grep error
rx_errors: 7787
tx_errors: 0
rx_over_errors: 0
rx_crc_errors: 0
rx_frame_errors: 0
rx_fifo_errors: 0
rx_missed_errors: 603296349
tx_aborted_errors: 0
tx_carrier_errors: 0
tx_fifo_errors: 0
tx_heartbeat_errors: 0
rx_long_length_errors: 0
rx_short_length_errors: 0
rx_csum_offload_errors: 0
```

Keep monitoring these error messages.
An error will be generated in Prism if the
rate is high.

NOTE: Nutanix generates an alert when the percentage of errors seen on a NIC reaches 1% of the total packets in a one-hour period. The alert is displayed in the Prism Health tab as **NIC error rate is high**.

Network adapter common issues

NICs are named and numbered by the operating system. When a NIC is replaced, the ports on the new NIC will not necessarily be assigned the same names as those on the original. For example, “eth2” and “eth3” may be renamed as “eth4” and “eth5”. Refer to Nutanix support portal article [KB3261](#) and VMware article [KB209156](#) for details.

- RX errors: The most common errors occurring on physical NIC ports are receive errors, which indicate that the NIC is discarding inbound frames for some reason.
- CRC errors: Every Ethernet frame contains a 4-byte CRC field at the end of the frame that is used to detect if bit errors have occurred during transmission. When any Ethernet port (on a host, switch, or router) receives a frame and the CRC field does not match the value that it calculates for the frame, the frame is discarded. CRC errors are commonly caused by an issue at the physical layer: with a port, optic, or cable. If the network is made up of cut-through switches, which begin forwarding frames before the entire frame has been received, it's possible for CRC errors to propagate across the network. For more information about troubleshooting CRC errors, refer to the following links:
 - [NCC Health Check: host_nic_error_check](#)
 - [KB 1453 - Troubleshooting Node NIC Error Rate checks](#)

Network adapter rx_missed errors

The hypervisor OS sets aside an amount of node memory to allow for the buffering of inbound frames during periods of high network activity. If this buffer is not large enough, inbound frames will be dropped. The buffer size is defined per port and can be checked by using the `ethtool -g <nic>` command. (Click [HERE](#) to see details.)

The default setting for the receive buffer is 1024 frames, but it is suggested that users change that to the maximum of 4096. The maximum receive buffer size is dependent on the MTU of the interface. Enabling jumbo frames reduces the maximum buffer size. The amount of memory consumed is determined by the MTU and the number of frames configured to be buffered. Due to issues with accounting for the amount of memory consumed when the RX buffer size is changed, altering this value is not currently supported.

Note that the "dropped" counter in the `ifconfig` command is not an accurate representation of frame loss, as it accounts for loss at the kernel rather than hardware level.

Note: Contact Nutanix support for details of how to set the maximum receive buffer.



Network adapter rx_missed errors

Output of #ethtool -g <nic>

```
root@host ~# ethtool -g eth3
FIPS mode initialized
Nutanix AHV
Ring parameters for eth3:
Pre-set maximums:
RX: 4096
RX Mini: 0
RX Jumbo: 0
TX: 4096
Current hardware settings:
RX: 1024
RX Mini: 0
RX Jumbo: 0
TX: 512
```

Note: Contact Nutanix support for details of how to set the maximum receive buffer.

Network adapter SFP not recognized issues

Nutanix does not qualify specific SFP+ or QSFP+ modules. Switch vendors often put code in their products to restrict usage to their own branded optics. This forces the customer to buy only their models. Some vendors relax this constraint (for example, by using “non-standard mode”) to allow the use of other (generic) optics, but this is switch/vendor dependent. Compatibility problems occur when the SFP module is not compatible with the 10 GbE NIC or with the upstream switch.

- SFP Not Recognized:

- On ESXi, you might see that the NIC is down due to an unsupported SFP module. This can be observed in the vmkernel.log file on the node. (Click [HERE](#) to see an example output.)
- In AHV, these messages will appear in the `dmesg` log. (Click [HERE](#) to see an example output.)

To resolve incompatibility issues between the 10 Gig NIC and the SFP module, use a different brand of SFP module. As noted above, Nutanix does not recommend any particular brand.

If a compatibility issue between the 10 Gig NIC and SFP module is resolved by trying a different brand of SFP module, there might be an incompatibility between the SFP module and the upstream switch. Switch vendors can advise on the compatibility between SFP modules and these devices.



Network adapter SFP not recognized issues

Output of ESXi network failure

```
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.252 cpu2:4111)PCI: Announcing 0000:0f:00.0
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.252 cpu2:4111)socket buffer cache creation
succeeded for vmnic%d
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.296 cpu2:4111)<3>ixgbe: 0000:0f:00.0: ixgbe_probe:
failed to load because an unsupported SFP+ module type was detected.
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.296 cpu3:4111)PCI: Trying 0000:0f:00.1
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.296 cpu3:4111)PCI: Announcing 0000:0f:00.1
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.296 cpu3:4111)socket buffer cache creation
succeeded for vmnic%d
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.340 cpu3:4111)<3>ixgbe: 0000:0f:00.1: ixgbe_probe:
failed to load because an unsupported SFP+ module type was detected.
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.340 cpu2:4111)PCI: driver ixgbe claimed 0 device
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.340 cpu2:4111)Mod: 3003: Initialization for ixgbe
succeeded with module ID 30.
Jun 28 16:13:43 VMESX01 vmkernel: 0:00:00:26.340 cpu2:4111)ixgbe loaded successfully.
```

If a compatibility issue between the 10 Gig NIC and SFP module is resolved by trying a different brand of SFP module, there might be an incompatibility between the SFP module and the upstream switch. Switch vendors can advise on the compatibility between SFP modules and these devices.



Network a not recognized issues

Output of dmesg

```
File View Macros Tools Power Next Boot Virtual Media Help

[ 6.681529] ixgbe: module verification failed: signature and/or required key
missing - tainting kernel
[ 6.683876] Intel(R) 10GbE PCI Express Linux Network Driver - version 4.4.6
[ 6.683878] Copyright(c) 1999 - 2016 Intel Corporation.
[ 6.687798] ipmi message handler version 39.2
[ 6.618914] IPMI System Interface driver.
[ 6.618982] ipmi_si: probing via SMBIOS
[ 6.618984] ipmi_si: SMBIOS: io 0xca8 regsize 1 spacing 4 irq 18
[ 6.618986] ipmi_si: Adding SMBIOS-specified kcs state machine
[ 6.618989] ipmi_si: Trying SMBIOS-specified kcs state machine at i/o address
0xca8, slave address 0x28, irq 18
[ 6.638828] ixgbe 0000:01:00.0: failed to load because an unsupported SFP+ or
QSFP module type was detected.
[ 6.638823] ixgbe 0000:01:00.0: Reload the driver after installing a supporte
d module.
[ 6.659894] (NULL device *): The BMC does not support setting the rcv irq bi
t compensating, but the BMC needs to be fixed
[ 6.662238] ixgbe 0000:01:00.1: failed to load because an unsupported SFP+ or
QSFP module type was detected.
[ 6.662232] ixgbe 0000:01:00.1: Reload the driver after installing a supporte
d module.
[ 6.668941] ipmi_si ipmi_si.0: Using irq 18
[ 6.676262] mei_me 0000:00:16.0: Device doesn't have valid ME Interface
[ 6.699616] igb: Intel(R) Gigabit Ethernet Network Driver - version 5.3.0-k
:_
```