

Linux Base Driver for 10 Gigabit PCI Express Intel(R) Network Connection

March 10, 2009

Contents

- In This Release
- Identifying Your Adapter
- Building and Installation
- Additional Configurations
- Support

In This Release

This file describes the ixgbe Linux Base Driver for the 10 Gigabit PCI Express Intel(R) Network Connection. This driver includes support for Itanium(R)2-based systems.

For questions related to hardware requirements, refer to the documentation supplied with your 10 Gigabit adapter. All hardware requirements listed apply to use with Linux.

The following features are available in this kernel:

- Native VLANs
- Channel Bonding (teaming)
- SNMP
- Generic Receive Offload
- Data Center Bridging

Channel Bonding documentation can be found in the Linux kernel source:
/Documentation/networking/bonding.txt

Ethtool, lspci, and ifconfig can be used to display device and driver specific information.

Identifying Your Adapter

This driver supports devices based on the 82598 controller and the 82599 controller.

For specific information on identifying which adapter you have, please visit:

<http://support.intel.com/support/network/sb/CS-008441.htm>

Building and Installation

ixgbe.txt

select m for "Intel(R) 10GbE PCI Express adapters support" located at:

Location:

-> Device Drivers

-> Network device support (NETDEVICES [=y])

-> Ethernet (10000 Mbit) (NETDEV_10000 [=y])

1. make modules & make modules_install

2. Load the module:

modprobe ixgbe

The insmod command can be used if the full path to the driver module is specified. For example:

```
insmod /lib/modules/<KERNEL VERSION>/kernel/drivers/net/ixgbe/ixgbe.ko
```

With 2.6 based kernels also make sure that older ixgbe drivers are removed from the kernel, before loading the new module:

```
rmmod ixgbe; modprobe ixgbe
```

3. Assign an IP address to the interface by entering the following, where x is the interface number:

```
ifconfig ethx <IP_address>
```

4. Verify that the interface works. Enter the following, where <IP_address> is the IP address for another machine on the same subnet as the interface that is being tested:

```
ping <IP_address>
```

Additional Configurations

Viewing Link Messages

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

```
dmesg -n 8
```

NOTE: This setting is not saved across reboots.

Jumbo Frames

The driver supports Jumbo Frames for all adapters. Jumbo Frames support is enabled by changing the MTU to a value larger than the default of 1500. The maximum value for the MTU is 16110. Use the ifconfig command to increase the MTU size. For example:

```
ifconfig ethx mtu 9000 up
```

The maximum MTU setting for Jumbo Frames is 16110. This value coincides with the maximum Jumbo Frames size of 16128.

Generic Receive Offload, aka GRO

The driver supports the in-kernel software implementation of GRO. GRO has shown that by coalescing Rx traffic into larger chunks of data, CPU utilization can be significantly reduced when under large Rx load. GRO is an evolution of the previously-used LRO interface. GRO is able to coalesce other protocols besides TCP. It's also safe to use with configurations that are problematic for LRO, namely bridging and iSCSI.

GRO is enabled by default in the driver. Future versions of ethtool will support disabling and re-enabling GRO on the fly.

Data Center Bridging, aka DCB

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

To enable DCB support in ixgbe, you must enable the DCB netlink layer to allow the userspace tools (see below) to communicate with the driver. This can be found in the kernel configuration here:

- > Networking support
- > Networking options
- > Data Center Bridging support

Once this is selected, DCB support must be selected for ixgbe. This can be found here:

- > Device Drivers
- > Network device support (NETDEVICES [=y])
- > Ethernet (10000 Mbit) (NETDEV_10000 [=y])
- > Intel(R) 10GbE PCI Express adapters support
- > Data Center Bridging (DCB) Support

After these options are selected, you must rebuild your kernel and your modules.

In order to use DCB, userspace tools must be downloaded and installed. The dcbd tools can be found at:

<http://e1000.sf.net>

Ethtool

ixgbe.txt

The driver utilizes the ethtool interface for driver configuration and diagnostics, as well as displaying statistical information. Ethtool version 3.0 or later is required for this functionality.

The latest release of ethtool can be found from
<http://sourceforge.net/projects/gkernel>.

NAPI

NAPI (Rx polling mode) is supported in the ixgbe driver. NAPI is enabled by default in the driver.

See www.cyberus.ca/~hadi/usenix-paper.tgz for more information on NAPI.

Support

=====

For general information, go to the Intel support website at:

<http://support.intel.com>

or the Intel Wired Networking project hosted by Sourceforge at:

<http://e1000.sourceforge.net>

If an issue is identified with the released source code on the supported kernel with a supported adapter, email the specific information related to the issue to e1000-devel@lists.sf.net