

Content Type Install & Setup

Article ID 000005480

Last Reviewed 11/01/2022

There are three Linux* base drivers for Intel® Gigabit Network Connections:

- igb-x.x.x.tar.gz driver: Supports all 82575/6, 82580, I350, I354, and I210/I211 based gigabit network connections.
- e1000e-x.x.x.x.tar.gz driver: Supports the Intel® PRO/1000 PCI-E (82563/6/7, 82571/2/3/4/7/8/9, or 82583) I217/I218/I219 based gigabit network adapters.
- e1000-x.x.x.tar.gz driver: Supports Intel® PRO/1000 PCI and PCI-X family of gigabit network connections.

Both the e1000e and e1000 drivers have changed to a kernel-only support model

. Thus, the latest $% \left\{ 1,2,\ldots,n\right\} =\left\{ 1,2,\ldots,n\right\}$

Note e1000e release is 3.8.7 and the latest for e1000 is 8.0.35. In brief, the kernel drivers (drivers included with the Operating System) will be the latest. Bug fixes and changes are made upstream in the Linux kernel

Select the proper version below for download and installation instructions.

Click on the topic for details:

Expand all

Linux* igb* base driver for Intel® Gigabit Ethernet Network Connection installation instructions

The Linux* igb driver supports all 82575, 82576, 82580, I350, I354, and I210/I211 based Intel® Gigabit Ethernet Network Connections.

For driver configuration details, see the Read Me file

in Download Center.

Overview

The Linux base drivers support the 2.4.x, 2.6.x and 3.x kernels. These drivers include support for Intel® Itanium® 2 Processor-based systems.

These drivers are only supported as a loadable module. We don't supply patches against the kernel source to allow for static linking of the drivers. For questions related to hardware requirements, see the documentation supplied with your Intel Gigabit adapter. All hardware requirements listed apply to use with Linux.

The following features are available in supported kernels:

Native VLANs

- · Channel Bonding (teaming)
- SNMP

Find channel bonding documentation in the Linux kernel source: /documentation/networking/bonding.txt

The igb driver supports IEEE 1588 time stamping for kernels 2.6.30 and above. For a basic tutorial for the technology, see NIST $\,$

The driver information previously displayed in the /proc file system isn't supported in this release. You can use ethtool (version 1.6 or later), lspci, and ifconfig to obtain the same information.

Building and installation

The steps below require elevated privileges.

Red Hat based platforms: CentOS, RHEL, or Fedora
yum install gcc make
yum install kernel kernel-devel

Prerequisites

Ubuntu and Debian based platforms

- apt-get install linux-headers-\$(uname -r)
- · apt-get install gcc make

You may need to perform a general update and restart before the next steps.

1. Download current igb package from Download Center

. Move the base driver tar file

to the directory of your choice.

For example, use /home/username/igb or /usr/local/src/igb.

2. **Untar/unzip** the archive, where <x.x.x> is the version number for the driver tar file:

tar zxf iqb-<x.x.x>.tar.qz

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

cd igb-<x.x.x>/src/

4. Compile the driver module:

make install

The binary will be installed as:

/lib/modules/<KERNEL VERSION>/kernel/drivers/net/igb/igb.ko
The install location listed above is the default location. This may differ for various Linux distributions.

5. Load the module using either the insmod or modprobe command:

modprobe igb insmod igb

For 2.6 kernels the insmod command can be used if the full path to the driver module is

Note specified. For example:

insmod /lib/modules/<KERNEL

VERSION>/kernel/drivers/net/igb/igb.ko

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

rmmod igb; modprobe igb

6. **Assign** an IP address to the interface by entering the following, where <x> is the interface number:

ifconfig eth<x> <IP address>

7. **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>

Some systems have trouble supporting MSI and/or MSI-X interrupts. If your system needs to disable this style of interrupt, the driver can be built and installed with the command:

Note

make CFLAGS_EXTRA=-DDISABLE_PCI_MSI install Normally the driver will generate an interrupt every two seconds. If you're no longer getting interrupts in cat /proc/interrupts for the ethX igb device, then this workaround may be necessary.

To build igb driver with DCA:

If your kernel supports DCA, the driver will build by default with DCA enabled.

Linux* e1000e base driver for Intel® Gigabit Ethernet Network Connection installation instructions

The Linux* e1000e driver supports the Intel® PRO/1000 PCI-E (82563/6/7, 82571/2/3/4/7/8/9, or 82583) I217/I218/I219 based gigabit network adapters.

For information concerning driver configuration details, refer to the Read Me file in the Download Center.

Overview

The Linux* base drivers support the 2.4.x and 2.6.x kernels. They include support for Intel® Itanium® 2 Processor-based systems.

The drivers are only supported as a loadable module. We don't supply patches against the kernel source to allow for static linking of the drivers. For questions related to hardware requirements, see the documentation supplied with your Intel® Gigabit Network Adapter. All hardware requirements listed apply for use with Linux.

Features now available in supported kernels:

- Native VLANs
- Channel Bonding (teaming)
- SNMP

Find Channel Bonding documentation in the Linux kernel source: /documentation/networking/bonding.txt

This release doesn't support the driver information previously displayed in the /proc file system. You can also use ethtool (version 1.6 or later), lspci, and ifconfig to get the same information.

Note

The Intel® 82562v 10/100 Network Connection only provides 10/100 support.

Building and installation

Steps below require elevated privileges.

Red Hat based platforms: CentOS, RHEL, or Fedora

• yum install gcc make

• yum install kernel kernel-devel

Prerequisites

Ubuntu and Debian based platforms

- apt-get install linux-headers-\$(uname -r)
- apt-get install gcc make

You may need to perform a general update and restart before the next steps.

- 1. Download current e1000e package from Download Center
 - . Move the base driver tar file

to the directory of your choice.

For example, use /home/<USERNAME>/e1000e or /usr/local/src/e1000e.

2. **Untar/unzip** the archive, where <x.x.x> is the version number for the driver tar file:

tar zxf e1000e-<x.x.x>.tar.qz

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

cd e1000e-<x.x.x>/src/

4. Compile the driver module:

make install

Note

The binary installs as:

/lib/modules/<KERNEL VERSION>/kernel/drivers/net/e1000e/e1000e.ko
The install location listed above is the default. Location may differ for various Linux* distributions.

5. Load the module using either the insmod or modprobe command:

modprobe e1000e insmod e1000e

You can use the insmod command for 2.6 kernels if you specify the full path to the driver module. For example:

insmod /lib/modules/<KERNEL

VERSION>/kernel/drivers/net/e1000e/e1000e.ko

With 2.6 based kernels, make sure that older e1000e drivers are removed from the kernel before you load the new module:

rmmod e1000e; modprobe e1000e

6. **Assign** an IP address to the interface by entering the following, where <x> is the interface number:

ifconfig eth<x> <IP address>

7. **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 you're testing:

ping <IP_address>

Some systems have trouble supporting MSI and/or MSI-X interrupts. If your system must disable this style of interrupt, build and install the driver with the command:

Note

make CFLAGS_EXTRA=-DDISABLE_PCI_MSI install Normally the driver generates an interrupt every two seconds. If you're no longer seeing interrupts in cat /proc/interrupts for the ethX e1000e device, then this workaround may be necessary.

 Linux* e1000 base driver for Intel® PCI, PCI-X Gigabit Network Connection installation instructions

The Linux* e1000 driver supports Legacy (PCI, PCI-X*) Gigabit Network Connections. For driver configuration details, see the ReadMe file in Download Center.

Note

The e1000 driver is no longer maintained as a standalone component. Request support from the maintainer of your Linux* distribution.

The Linux base drivers support the 2.4.x and 2.6.x kernels. These drivers include support for Intel® Itanium® 2 Processor-based systems.

These drivers are only supported as a loadable module. We don't supply patches against the kernel source to allow for static linking of the drivers. For questions related to hardware requirements, refer to the documentation supplied with your Intel® Gigabit Adapter. All hardware requirements listed apply to use with Linux.

The following features are now available in supported kernels:

- Native VLANs
- Channel Bonding (teaming)
- SNMP

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

The driver information previously displayed in the /proc file system isn't supported in this release. Alternatively, you can use ethtool (version 1.6 or newer), Ispci, and ifconfig to obtain the same information.

Note

The Intel® 82562V Fast Ethernet PHY 10/100 Network Connection only provides 10/100 support.

Building and installation

The steps below require elevated privileges.

Red Hat based platforms: CentOS, RHEL, or Fedora

• yum install gcc make
• yum install kernel kernel-devel

Ubuntu and Debian based platforms
• apt-get install linux-headers-\$(uname -r)
• apt-get install gcc make

You may need to perform a general update and restart before the next steps.

1. Download current e1000 package from Download Center

. Move the base

driver tar file to the directory of your choice. For example, use /home/username/e1000 or /usr/local/src/e1000.

2. **Untar/unzip** the archive, where <x.x.x> is the version number for the driver tar file:

tar zxf e1000-<x.x.x>.tar.gz

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

cd e1000-<x.x.x>/src/

4. Compile the driver module:

make install

Note

The binary is installed as:

/lib/modules/<KERNEL VERSION>/kernel/drivers/net/e1000/e1000.ko
The install location listed above is the default location. This may differ for various Linux distributions.

5. Load the module using either the insmod or modprobe command:

modprobe e1000 insmod e1000

For 2.6 kernels 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/e1000/e1000.ko

With 2.6 based, kernels also make sure that older e1000 drivers are removed from the kernel, before loading the new module:

rmmod e1000; modprobe e1000

6. **Assign** an IP address to the interface by entering the following, where <x> is the interface number:

ifconfig eth<x> <IP_address>

7. 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's being tested:

ping <IP_address>

	Related topic Linux* Operating System Vendor Websites
	Related Products
	This article applies to 26 products.
	Show all ♥
	Need more help?
	Contact support
	Give Feedback
· ·	Company Overview
(Contact Intel
	Newsroom
ı	nvestors
(Careers
(Corporate Responsibility
1	Diversity & Inclusion
ı	Public Policy
	f y in
	© Intel Corporation Terms of Use
*	Trademarks
	Cookies
	Privacy Supply Chain Transparency
	Site Map
F	Recycling
5	ntel technologies may require enabled hardware, software or service activation. // No product or component can be absolutely secure. // Your costs and results may vary. // Performance varies by use, configuration and other factors. // See our complete legal Notices and Disclaimers
<u> </u>	.// Intel is committed to respecting human rights and avoiding complicity in human rights abuses. See Intel's <u>Global Human Rights Principles</u> Intel's intel's orducts and software are intended only to be used in applications that do not cause or contribute to a violation of an internationally recognized human right.

