



T210/T110 - 10Gb Ethernet Protocol Engine and T204 - Quad Gigabit Ethernet Protocol Engine Installation and User's Guide for Linux.

Driver release 2.1.4, February 2006 (cxgbtoe-2.1.4), Document Rev 1.4

Copyright 2005 Chelsio Communications, Inc. All rights reserved.
370 San Aleso Ave, Suite 100, Sunnyvale, CA 94085

This document and related products are distributed under licenses restricting their use, copying, distribution, and reverse-engineering. No part of this document may be reproduced in any form or by any means without prior written permission by Chelsio Communications.

T210, T1104, and T204 are trademarks of Chelsio Communications, Inc.

THIS DOCUMENTATION IS PROVIDED "AS IS" AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, WITHOUT LIMITATION, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

THE USE OF THE SOFTWARE AND ANY ASSOCIATED MATERIALS (COLLECTIVELY THE "SOFTWARE") IS SUBJECT TO THE SOFTWARE LICENSE TERMS OF CHELSIO COMMUNICATIONS, INC.

Contents

Introduction	1
Hardware Requirements	2
Minimum System Requirements (for best performance)	2
Recommended System Requirements (for best performance)	2
Software Requirements	3
Linux OS Requirements.....	4
Recommended Distributions.....	4
Other Software Recommendations.....	4
Hardware Installation	5
Software Installation	6
Installing the Kernel Binary RPM Package	6
Patching the Driver into the Linux Kernel	8
Configuring the Linux Bootloader.....	14
TOE Configuration	15
Installing the chtoetool Utility	15
Using the chtoetool Utility	15
Setting the interface MTU	16
Performance Tuning for 10Gb	16
System Configuration	17
T204 Quad Gb Bonding Configuration	17
T110 Power Save	17
Errata	18
FAQ	19
Customer Support	22

Introduction

This document describes the Chelsio T210/T110 10Gb Ethernet Protocol Engine and T204 Quad Gigabit Ethernet Protocol drivers for Linux.

The Chelsio T210 10GbE Protocol Engine is a protocol-offload 10Gb Ethernet adapter for servers and storage systems. The T210 contains the second generation Terminator2 ASIC which offloads TCP/IP processing from the host system, providing higher performance and reduced latency across all connections. This offloading is commonly referred to as a TOE (TCP Offload Engine). The T210 is available with support for optical median (XFP-SR/LR) or copper (CX-4) media.

The T110 contains the original Terminator hardware with support for optical median or 10G copper (CX-4) media.

The Chelsio T204 Quad GbE Protocol Engine is a high-performance protocol-offload adapter for storage systems featuring four 1Gb Ethernet ports.

This driver provides 10Gb support in both offload (TOE) and non-offload (NIC) modes for T210/T110 and 1Gb support in both offload (TOE) and non-offload (NIC) modes for T204.

This driver also supports Chelsio's iSCSI Target software in offload (TOE) or non-offload (NIC) modes for any of Chelsio's Terminator or Terminator2 based adapters.

Hardware Requirements

The T210/T110/T204 driver is designed for the following architectures.

- All x86 (32-bit) and x86_64 (64-bit) architectures supporting PCI/PCI-X (133, 100, or 66Mhz) are supported.
- All Intel Itanium2 (ia64) architectures supporting PCI/PCI-X (133, 100, or 66Mhz) are supported.

* IBM PPC (ppc and ppc64) architectures have not been tested.

NOTE: The T210/T110 10Gb and T204 Ethernet adapter supports 3.3v PCI bus only. Running a 10Gb adapter on a PCI (32-bit) bus is not recommended as throughput performance will be significantly reduced by the limitations of PCI bus.

Minimum System Requirements (for best performance)

- One PCI-X (64-bit) or PCI (32-bit) which supports 3.3v devices.
- 1MB RAM with 400MHz memory bus.
- 1.8GHz AMD Opteron 144 or 2.4GHz Intel Xeon CPU

Recommended System Requirements (for best performance)

- One PCI-X (64-bit) running at 133Mhz and no other devices using same segment.
- 2.4GHz AMD Opteron 250 or 3.2GHz Intel Xeon EM64T CPU.
- 1GB DDR RAM per CPU with 800Mhz (or faster) memory bus.

Software Requirements

This driver supports the following Chelsio Ethernet adapters:

- T210 10Gb Ethernet Adapter with XFP (SR/LR) optics in TOE or NIC modes.
- T210-CX4 (10Gb Copper) in TOE or NIC mode.
- T204 Quad Gigabit Ethernet adapter in TOE or NIC mode.
- Legacy support for T110 and T110-CX4 10Gb Ethernet adapter.

The driver must modify the Linux kernel source. The current supported Linux kernels are:

- Linux-2.4.25
- Linux-2.6.5-7.139 (Suse SLES9 with SP1)
- Linux-2.6.5-7-191 (Suse SLES9 with SP2)
- Linux-2.6.9-5.EL (RedHat RHEL4)
- Linux-2.6.9-11.EL (RedHat RHEL4 with Update 1)
- Linux-2.6.9-22.EL (RedHat RHEL4 with Update 2)
- Linux-2.6.12 through Linux 2.6.12.6

The Linux kernel sources can be downloaded from <http://www.kernel.org>, with the exception of the distribution specific kernels (RedHat and Suse). These distribution kernel sources must be obtained through RedHat or Suse in the form of source RPMs.

The driver is also available in a pre-compiled Linux kernel for Suse SLES9 SP1/2 and RedHat RHEL4 (and Update 1/2). These are available as kernel RPMs which will include the Chelsio TOE driver pre-compiled in the kernel.

The driver should function on any Linux distribution, for as long as the distribution operates properly with the required kernel versions.

The driver has been tested with RedHat RHEL4 and Suse SLES9 Linux distributions.

Linux OS Requirements

If a new kernel is to be compiled for the target system, certain development software packages will be required. Additionally, the following packages will be required in order to compile the tools provided with the TOE driver. The minimum (and recommended) software versions are:

Gnu C Compiler (gcc)	2.95 (recommended 3.4.2)	4.0.0 is unsupported
Gnu make	3.79.1	
binutils	2.12	
util-linux	2.10o	
procps	3.2.0	
module-init-tools	3.0 (for linux-2.6 kernel)	

Depending on the options enabled within the Linux kernel, additional packages may be required. Check the requirements for the Linux kernel. These may be found in the Linux kernel source tree in the Documentation/Changes file.

Most or all of these packages may already be provided, depending on the Linux distribution. If the packages are missing, it may be due to not installing the development packages from the distribution CDROM.

It is important to note that GCC version 4.0+ is not supported by the driver. The driver will not properly compile when using this compiler version.

Recommended Distributions

The following Linux distributions are recommended for running with the Chelsio T210/T110/T204 Ethernet Protocol Engine adapter driver. These distributions provide the least complicated installation of 2.6 series kernels with the TOE driver and also provide the minimum required utilities for compiling the Linux kernel.

- Novell Suse Linux Enterprise Server 9
- RedHat Enterprise Linux 4

Other Software Recommendations

The following software utilities are required for certain network related operations. It is recommended that the target system includes these applications (and at least the minimum version, if stated) in order to perform network configuration.

Application/Utility	version	Comments
ethtool	1.8+	
ifconfig		
procfs		
netstat		

Hardware Installation

1. Halt and power off your system.
2. Power off all remaining peripherals attached to your system.
3. Unpack the Chelsio 10Gb/Quad-1Gb Ethernet adapter and place it on an anti-static surface.
4. Remove the system case cover according to the system manufacturer's instructions.
5. Remove the PCI filler plate from the slot where you will install the Chelsio adapter. For maximum performance, it is highly recommended to install the adapter into a PCI-X slot running at 133Mhz. Avoid installing more than one device per PCI segment as this will revert the bus to 100Mhz operation.
6. Holding the Chelsio adapter by the edges, align the edge connector with the PCI connector on the motherboard. Apply even pressure on both edges until the card is firmly seated. It may be necessary to remove any XFP (optics) modules prior to inserting the adapter.
7. Secure the Chelsio adapter with a screw, or other securing mechanism, as described by the system manufacturer's instructions. Replace the case cover.
8. Connect a fiber cable, multi-mode for short range (SR) optics or single-mode for long range (LR) optics, to the 10Gb Ethernet adapter. Connect CAT-5/CAT-5e/CAT-6 cables with RJ-45 connectors to the Quad-1Gb Ethernet adapter. Connect a CX-4 cable to the T210-CX4 adapter.
9. Power on the system and continue to the "Software Installation" section.

Software Installation

The Chelsio T210/T110/T204 driver and software may be downloaded via our support website at <http://www.chelsio.com>. The driver may also be distributed on CD-ROM.

The T210/T110/T204 Linux driver requires additions to the Linux kernel TCP stack code. The driver adds “hooks” into the TCP code in order to provide a TOE API for offloading capabilities.

With the TOE API in place, the Linux kernel is required to be compiled and installed. Generally, a Linux kernel is required to be patched, configured, compiled and installed, however, an alternative is to use a kernel binary RPM, provided by Chelsio, which is already configured and compiled with the TOE API and driver.

The kernel binary RPM is available for RedHat RHEL4 and Suse SLES9. The RPMs are kernel binaries from the respective distributions which are identical to the kernels installed on the system by default, except that they include the TOE driver.

Installing the Kernel Binary RPM Package

The Linux kernel binary RPM packages are available for x86 (32-bit) and x86_64 (64-bit) architectures. These binaries have been compiled with i386 and generic x86_64 optimizations, for both single-processor and multi-processor (SMP) systems.

See the “Software Requirements” sections for supported kernels and distributions.

These kernel RPMs are only available for RedHat RHEL4 and Suse SLES9 distributions! For other distributions and versions, it will be necessary to use a patch for one of the supported Linux kernels.

The following table describes the binary RPM package architecture naming convention.

kernel-[smp-]<linux version>cxgbtoe-<driver version>.<architecture>.rpm		
<linux version>	The base Linux kernel version.	
[smp]	The presence of “smp” indicates the kernel is compiled for symmetrical-multi-processor support (SMP). The absence indicates single-processor only.	
cxgbtoe-<driver version>	The version of the Chelsio 10Gb TOE driver.	
<architecture>	The following architectures are used when compiling the kernel binary RPM package:	
	i386	Supports all 32-bit x86 processors. This is the most compatible binary version of the Linux kernel; however, there are no optimizations for faster CPUs.
	x86_64	Supports all 64-bit x86 processors, including AMD Athlon/Hammer/K8/Opteron and Intel Xeon EM64T processors. This only has optimizations for generic 64-bit CPUs and will run on all x86_64 processors.

Use the above tables for determining the best kernel binary RPM for the target system. If an RPM package is not available, it will be necessary to configure and compile the Linux kernel

with the T210/T110/T204 driver patch. See the section “Patching the Driver into the Linux Kernel” for details.

The kernel binaries RPM packages have been compiled on a machine with the same target architecture using gcc version 3.4.2. The RPM packages have enabled all features for the T210/T110 and T204, including 10Gb support, 1Gb support, and bonding (for T204).

The Linux kernel binary RPM packages have been compiled with the same features and device drivers enabled as the distribution kernels. The kernel should function identically to the original kernel provide by the distribution, with the exception of including the Chelsio device driver.

There may be cases in which the kernel binary RPM does not contain a device driver required by the target system, in which case it would be necessary to compile a driver for the device and install it as a module on the target system. Please refer to the distribution documentation for details.

1. Login as root user and use the rpm utility to install the Linux kernel binary package.

```
[root@]# rpm -ivh kernel-2.6.9-11.ELcxgbtoe2.1.4.x86_64.rpm
Preparing... ##### [100%]
 1:kernel-2.6.9-11.EL ##### [100%]
```

2. After installing the kernel binary RPM package, the system bootloader will be configured to boot the new kernel image. Refer to section “Configuring the Linux Bootloader” for information on how to change the bootloader configuration.
3. After configuring the bootloader, reboot the system and select the new kernel when the bootloader starts. It may be necessary to run `depmod -a` to scan for the new modules which were installed on the system after the system is up.

Patching the Driver into the Linux Kernel

The T210/T110/T204 driver can be patched into a Linux kernel. The kernel needs to be free of any other patches or modifications prior to installing the TOE API and driver. Apply any additional patches or modifications after installing the driver.

A new Linux kernel can be downloaded from <ftp.kernel.org> or via their website at <http://www.kernel.org>. The following information is a guide for downloading a Linux kernel from the Kernel Dot Org Organization, Inc.

Use an ftp client application and make a connection to <ftp://ftp.kernel.org>. Log in as 'anonymous' user, no password is required. Use the following table for the path and filename of the kernel version which is to be installed.

Linux-2.6.12.6 ftp://ftp.kernel.org/pub/linux/kernel/v2.6/linux-2.6.12.6.tar.bz2
Linux-2.4.25 ftp://ftp.kernel.org/pub/linux/kernel/v2.4/linux-2.4.25.tar.bz2

Download the Linux kernel source tree to a location with enough space for building the kernel. This will be considered the working directory for building the kernel. It is recommended to use an area such as `/usr/src/` since the kernel source tree will reside in the build area permanently, so it can be accessible to other users for compiling kernel modules or linking other source code to the kernel headers.

Patch files for existing distribution based kernel are provided for RedHat RHEL4 and Suse SLES9 only. Before applying the patch to the RHEL4 kernel, it is important to note that RedHat no longer installs the complete kernel source tree with the distribution, even if "everything" is installed. RedHat only installs a skeleton tree with the required header files for compiling modules. Download the source RPM for the 2.6.9-x.EL kernel from RedHat's website and install the source tree according to the RedHat's documentation.

Depending on the features and drivers enabled, it could require up to 500MB of disk space. When installing the Linux kernel, be sure to use the root account on the system.

The following steps show the 2.6.12.6 version of the Linux kernel as an example. If installing the 2.4.25 kernel, substitute the version number with 2.4.25.

1. Uncompress the tar image to the `/usr/src` directory.

```
[root@]# tar xvj linux-2.6.12.6.tar.bz2 -C /usr/src
```

2. After downloading, uncompress the `cxgbtoe-2.1.4` driver package to a working directory.

```
[root@]# tar xvf cxgbtoe-2.1.4.tar.gz -C /root
```

3. Change directory into the new cxgbtoe-2.1.4 directory and run the install script. Pass in a parameter for the location of the Linux kernel source tree. Do not run the install.sh script from outside the cxgbtoe-2.1.4 directory.

```
[root@]# cd /root/cxgbtoe-2.1.4
[root@]# ./install.sh /usr/src/linux-2.6.12.6
```

The script will provide user prompts for installing the iSCSI patch (2.4 kernel only, 2.6 kernels do not need this patch) and compiling the chtoetool user application. Answer the questions by entering 'y' or 'n', or accept the defaults by hitting enter. The default is denoted by a capital 'Y' or 'N' within the parenthesis.

```
Do you want to install iSCSI support? (Y/n) y
Patching Linux kernel 2.6.12.6 with Chelsio's 10Gb TOE driver.
Do you want to add an EXTRAVERSION label to the kernel? (Y/n)
Enter new EXTRAVERSION label or use default [-cxgbtoe-2.1.4]

Do you want to install chtoetool? (Y/n) y
Compiling and installing chtoetool.

The Chelsio 10Gb TOE driver installation is not complete until
you build and install the Linux kernel. Follow the instructions
in the User/Installation Guide or continue with the following:
...
```

It is recommended to answer 'y' to adding an EXTRAVERSION label to the kernel. This will allow the customized kernel to be easily identified. It is suggested to use the default label provided. The kernel top-level Makefile will be modified with an updated kernel extraversion. The new version of the kernel will be displayed as **2.x.x-cxgbtoe2.1.4** when booting or when using the `uname` utility.

If answered 'y' to installing chtoetool, the script will compile the chtoetool utility and install it to `/sbin/chtoetool`. This tool may be needed for configuring certain features of the TOE adapter.

Performing a manual patch of the Linux kernel may be performed by using the patch files provided in the patches directory. Do not attempt to apply a patch to a kernel version for which a patch file has not been created.

When patching 2.6 kernels, there are two patches which need to be applied, first is the `toe_api` and second is the `toe_drv`.

When patching 2.4 kernels, and if iSCSI hardware support is required, there are three patches which need to be applied, first is the `toe_api`, second is the `toe_iscsi` and third is the `toe_drv`.

To patch the kernel manually, change to the Linux source tree root directory and use the `patch` command.

```
[root@]# cd /usr/src/linux-2.6.12.6
[root@]# patch -p1 < /root/cxgbtoe-2.1.4/patches/linux-2.6.12.6-toe_api.patch
[root@]# patch -p1 < /root/cxgbtoe-2.1.4/patches/linux-2.6.12.6-toe_drv.patch
```

4. After patching the Linux kernel with the driver, it is required to configure the kernel for the target system. It is necessary to have basic knowledge of the system hardware in order to configure the kernel.

It is not recommended to use a configuration from another version kernel, especially from kernels with different minor (patchlevel) versions.

Change directory to the Linux kernel source tree located in `/usr/src/linux-<version>`. If a kernel configuration file is already created for the target system, copy it to `.config` and run `make oldconfig`.

```
[root@]# cd /usr/src/linux-2.6.12.6
[root@]# cp config-2.6.12.6 .config
[root@]# make oldconfig
```

If not, use `make menuconfig` to configure the kernel.

```
[root@]# make menuconfig
```

Depending on the version of the kernel, the menu system of the config utility will have a different menu structure. Follow the T210/T110/T204 driver configuration requirements for the kernel version being installed.

The following tables list the T210/T110 driver configuration requirements for version 2.6 and 2.4 Linux kernels. The driver must be included in the Linux kernel as a “built-in” driver. **This driver does not support being built as a module at this time.**

Linux 2.6.12.6 configuration for T210/T110.

```
Device Drivers --->
  Networking support --->
    Ethernet (10000 Mbit) --->
      <*> Chelsio Communications T2/T1 support
      [*] TCP offload support
      [ ] ULP support
      [ ] Bonding support
      [ ] Chelsio gigabit Ethernet support
```

Linux 2.6.12.6 configuration for T204 (with bonding support).

```
Device Drivers --->
Networking support --->
  <M> Bonding driver support
  Ethernet (10000 Mbit) --->
    <*> Chelsio Communications T2/T1 support
    [*] TCP offload support
    [ ] ULP support
    [*] Bonding support
    [*] Chelsio gigabit Ethernet support
```

NOTE: The Linux bonding driver may be configured as a module or as built-in driver.

Linux 2.6.12.6 configuration for iSCSI support (enable ULP support).

```
Device Drivers --->
Networking support --->
  Ethernet (10000 Mbit) --->
    <*> Chelsio Communications T2/T1 support
    [*] TCP offload support
    [*] ULP support
    [ ] Bonding support
    [ ] Chelsio gigabit Ethernet support
```

Linux 2.4.25 configuration for T210/T110.

```
Network device support --->
  Ethernet (1000 Mbit) --->
    <*> Chelsio Communications T2/T1 support
    [*] TCP offload support
    [ ] ULP support
    [ ] Bonding support
    [ ] Chelsio gigabit Ethernet support
```

NOTE: The 2.4.25 version Linux kernel does not have a menu for 10000 Mbit (10Gb) Ethernet so the 1000 Mbit (1Gb) menu is used instead.

Linux 2.4.25 configuration for T204.

```
Network device support --->
  <M> Bonding driver support
  Ethernet (1000 Mbit) --->
    <*> Chelsio Communications T2/T1 support
    [*] TCP offload support
    [ ] ULP support
    [*] Bonding support
    [*] Chelsio gigabit Ethernet support
```

Linux 2.4.25 configuration for iSCSI support (enable ULP support).

```
Network device support --->
 Ethernet (1000 Mbit) --->
  <*> Chelsio Communications T2/T1 support
  [*]   TCP offload support
  [*]   ULP support
  [ ]   Bonding support
  [ ]   Chelsio gigabit Ethernet support
```

Continue through the kernel configuration and configure the kernel for the target system. Do not enable the Kernel Debugging features, as this will severely impact the performance of the system and slow down the high-speed networking.

Once finished with the configuration, exit and save the kernel config file. This file will be saved to the kernel source root directory as `.config`.

5. Build the Linux kernel. Linux 2.6.12.6 and 2.4.25 differ in the way the kernel is built. Follow the directions for the version of kernel being installed on the target system.

Building the Linux 2.6.12.6 kernel

```
[root@]# make bzImage
```

Building the Linux 2.6.12.6 kernel on Itanium systems (IA64)

```
[root@]# make
```

After running the make command on an Itanium architecture system, it is not necessary to run make modules, skip to step 7.

Building the Linux 2.4.25 kernel

```
*** End of Linux kernel configuration.
*** Check the top-level Makefile for additional configuration.
*** Next, you must run 'make dep'.

[root@]# make dep
    ...

[root@]# make bzImage
```

6. If the kernel is configured for modules, they will need to be compiled as well. This is performed the same way regardless of the Linux kernel version.

```
[root@]# make modules
```

7. Even if the target system was not configured for modules or the modules were not compiled, running `make modules_install` is required for the kernel.

This will setup a symlink to the kernel source tree in the `/lib/modules/<kernel version>` directory to be used by future applications or device drivers required to link to the kernel source tree.

```
[root@]# make modules_install
```

8. Install the kernel by copying the kernel image to the `/boot` directory. The location of the kernel image (`bzImage`) will vary depending on the target system architecture.

For all x86 architectures, `<ARCH>` will be `i386`.

For `x86_64` (64-bit) architectures, `<ARCH>` will be `x86_64`.

For Itanium (64-bit) architectures, the compressed kernel binary will be located in the Linux source root directory and named `vmlinuz.gz`. This file will need to be copied to the appropriate boot directory for the target system. Since Itanium systems vary in boot loaders (EFI), please refer to the distribution documentation or the documentation provided by the system manufacturer for details.

The `bzImage` file is located in `arch/<ARCH>/boot/bzImage`

Copy the `bzImage` to the `/boot` directory and rename it according to the kernel version number (2.6.12.6 or 2.4.25) and append the `-cxgbtoe2.1.4` name to the file.

```
[root@]# cp arch/x86_64/boot/bzImage /boot/vmlinuz-2.6.12.6-cxgbtoe2.1.4
```

NOTE: The new kernel will have an updated extraversion `-cxgbtoe2.1.4`. The appended version tag should not be altered; it helps in identifying the version of the T210/T110/T204 driver compiled into the Linux kernel.

Copy the `System.map` and `.config` file to the `/boot` directory. Rename the files with the proper kernel version and append the `-cxgbtoe2.1.4` version.

```
[root@]# cp System.map /boot/System.map-2.6.12.6-cxgbtoe2.1.4
[root@]# cp .config /boot/config-2.6.12.6-cxgbtoe2.1.4
```

9. Configure the bootloader for the installed kernel. Refer to section “Configuring the Linux Bootloader” for information on how to configure the bootloader.

Configuring the Linux Bootloader

The GRUB (Grand Unified Bootloader) is the standard bootloader for the latest Linux distributions. The Linux Loader (LILO) is an older bootloader which will not be covered in this document; however, the same principles of configuration are applicable. See the LILO documentation for details on configuring the LILO bootloader.

1. Determine the /root device of the target system. Some distributions use drive labels for identifying the /dev/<device> of the hard disk. Do not use a drive label unless the support has been enabled in the kernel and/or an initrd file has been created.

Use the mount command to find the / device. In this example, the / (root) filesystem is mounted from /dev/hda3.

```
[root@]# mount
/dev/hda3 on / type ext3 (rw)
```

2. Edit the GRUB configuration file located in /boot/grub/grub.conf. There may be symlinks to this file in other locations on the target system depending on the distribution.

Add an entry for the new kernel to the grub.conf file. The following table shows an example of a new entry in the grub.conf file for the 2.6.12.6-cxgbtoe2.1.4 kernel. Modify the default= variable with the number of the new kernel entry which was added. GRUB numbers the kernel entries starting with the number 0.

For best performance, add the kernel parameter "noirqbalance" to disable the kernel interrupt balancer, if it has been enabled in the kernel.

If using initrd, add an initrd image entry, see the example from the RHEL entry.

GRUB bootloader configuration file.

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this file
# NOTICE: You have a /boot partition. This means that
#           all kernel and initrd paths are relative to /boot/, eg.
#           root (hd0,0)
#           kernel /vmlinuz-version ro root=/dev/hda3
#           initrd /initrd-version.img
#boot=/dev/hda
default=2
timeout=5
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Enterprise Linux AS (2.6.9-11.ELsmp)
    root (hd0,0)
    kernel /vmlinuz-2.6.9-11.ELsmp ro root=LABEL=/1 rhgb quiet
    initrd /initrd-2.6.9-11.ELsmp.img
title Red Hat Enterprise Linux AS-up (2.6.9-11.EL)
    root (hd0,0)
    kernel /vmlinuz-2.6.9-11.EL ro root=LABEL=/1 rhgb quiet
    initrd /initrd-2.6.9-11.EL.img
title Linux 2.6.12.6-cxgbtoe2.1.4
    root (hd0,0)
    kernel /vmlinuz-2.6.12.6-cxgbtoe2.1.4 ro root=/dev/hda3 noirqbalance
```

3. Save and exit the grub.conf file. Reboot the system into the new kernel.

TOE Configuration

Installing the `chtoetool` Utility

The `chtoetool` utility is provided for enabling the TOE (TCP/IP Offload Engine) when the TOE mode has been disabled by default. The utility also provides control for setting interface MTU and read/write access to registers.

The utility may be compiled using `make` or by using the `install.sh` script provided by the driver package. It is necessary to perform only one of the two steps provided for installing the `chtoetool` utility. If the script was used to install the utility during the driver patch, do not continue with these steps.

1. Change to the `cxgbtoe-2.1.4` source directory and use the `install.sh` script to compile and install the `chtoetool` utility.

```
[root@]# cd cxgbtoe-2.1.4/
[root@]# ./install.sh

Do you want to patch the Linux kernel with the Chelsio 10Gb TOE
driver? (y/N) n

Do you want to install chtoetool? (Y/n) y
Compiling and installing chtoetool.
```

2. Change to the `cxgbtoe-2.1.4/tools/chtoetool` source directory and use `make` to compile and install the `chtoetool` utility.

```
[root@]# cd cxgbtoe-2.1.4/tools/chtoetool
[root@]# make install
```

Using the `chtoetool` Utility

The `chtoetool` utility is used to configure some TOE specific features or to read/write special registers of the Chelsio Terminator hardware.

The `chtoetool` utility requires at least two arguments for operation.

```
chtoetool <interface> <operation>
```

The `<interface>` is the current network interface of the TOE adapter. The `chtoetool` utility will only operate with Chelsio TOE adapters.

The TOE adapter can be configured to initialize in NIC mode by adding the `'toe_disable'` parameter (no quotes) to the kernel line in the bootloader. Use `chtoetool` to enable TOE mode. Once TOE mode is enabled, it cannot revert to NIC mode unless the system is rebooted with the `'toe_disable'` parameter.

The following table lists the available features of chtoetool.

mtus [mtu0 - mtu7]	<p>Reads or writes the MTU table. This should be used for T204 cards only!! Never adjust the MTU table while there are active connections!</p> <p>If writing the MTU table with a new entry, all MTU table values must be provided and in ascending order. Example, when changing the existing table 4000 value to 4096: <code>chtoetool eth0 mtu 68 508 576 1492 1500 2000 4096 9000</code></p>
up	Enable TOE mode on the adapter. This is used when the adapter is booted into NIC mode (using 'toe_disable' kernel parameter).
regdump	Dumps the contents of the Terminator ASIC registers.

Setting the interface MTU

The TOE needs to co-operate with the interface MTU settings. Using the `ifconfig` utility will set the MTU as on any other NIC.

The T210/T110 TOE adapters support any MTU value from 68 to 9580 bytes. If the interface MTU is changed (using `ifconfig`) during an active connection, the driver will compensate for the change. If the MTU is lowered, the driver will scale the currently active connection down to the best value. In some cases, this value will be lower than the set MTU size.

The T204 TOE adapter operates very differently from the other adapters, since it is a quad-port adapter, it must be able to handle different MTU values on each port. This is done by providing a "fixed" MTU table.

The default MTU table for the T204 adapter provides the MTU values as shown in the table.

68	508	576	1492	1500	2000	4000	9000
----	-----	-----	------	------	------	------	------

If an MTU value is used which is not in the MTU table, the TOE connections will transmit data on the next lowest MTU value. If a different MTU value is required, use the `chtoetool` utility to set a new table with the desired MTU. The table may be modified to include MTU values in the range of 68 to 9580 bytes. See "Using the chtoetool Utility" for details on how to change these values.

Performance Tuning for 10Gb

Please refer to the "Linux 10Gb Tuning Guide for Terminator2" document provided by Chelsio Communications at <http://www.chelsio.com>.

A performance tuning script 'perftune.sh' is provided in the driver package under the scripts directory. This script is an example of tuning for specific applications. It is recommended that the script be examined and understand what it is modifying. This script can be run for most throughput intensive applications, such as the netperf benchmark test.

System Configuration

The system network will need to be configured for the new Ethernet interface. Edit the interface configuration files for the target system to make the network configuration permanent. Each distribution has a different method of configuring the network interfaces. Refer to the distribution documentation on configuring the network.

By default, the T210/T110/T204 devices are in TOE (offload) mode. If it is necessary to enable the devices in “tunneled” (non-offload) mode, add the kernel parameter “toe_disable” to the bootloader configuration.

T204 Quad Gb Bonding Configuration

The 4-port Gigabit Ethernet adapter can be configured using the network utilities provided by the Linux distribution.

The T204 supports two bonding modes for TOE, 802.3ad link aggregation (mode 4) and active backup (mode 1). All other bonding modes will be run in NIC mode (non-offloaded).

NOTE: Bonding mode 6 is not supported by the T204.

For complete details on how to configure the system and use the Linux bonding driver, see the Linux kernel documentation in <LinuxSource>/Documentation/networking/bonding.txt.

T110 Power Save

By default, the T110 hardware is running in an “over-clocked” mode. This is the normal mode of operation for the T110 hardware. In very few systems with insufficient power capabilities, it has been noted that the power consumption may cause a reboot to occur during the system boot-up.

To enable the powersave feature of the software, add the kernel parameter `tlpowersave` to the bootloader.

```
kernel /vmlinuz-2.6.12.6-cxgbtoe2.1.4 ro root=/dev/hda3 noirqbalance tlpowersave
```

Errata

The following issues have been noted by Chelsio Communications for this driver release.

- The TOE driver uses larger memory buffers than conventional NICs. Under rare conditions, this could result in applications running out of buffer space. A solution is to increase the buffer size of the application. This can sometimes be accomplished by using `sysctl`. An example of a failure would be generating a ping packet with a very large packet size while having an MTU size between 1524 and 5459.

```
ping -s 60000 <interface>
```

This ping would fail under the conditions described.

- When using an IPv6 server as dual family server (IPv4/IPv6), any connections established on the IPv4 side of the server will not be offloaded. The Terminator2 ASIC does not support offloading IPv6 connections and the hardware sees a dual family server with IPv4 connections as an IPv6 server.

Do not use a dual family (IPv6/IPv4) server for offloaded connections.

- Using `sysctl` to change `tcp_timestamp`, `tcp_sack`, `tcp_fin_timeout`, and other related TCP parameters does not affect the packet options. The `sysctl` application does not return an error, but the TCP configuration is not applied to the offloaded packets.
- If running an application which reuses a 4-tuple too quickly, the application can fail to connect or transfer data on the network.

An example is when performing two `rsh` commands, within a script, it will be possible for the second `rsh` command to fail.

This occurs because the `rsh` command reuses the same 4-tuples in opening connections in both directions when executing a command.

- When a listening server is closed while a connection is still in the process of being established, or if a connection is reset by the peer (or the connection timed out) before the listening server was closed, the kernel warning message:

```
"Kernel: assertion (!lopt->qlen) failed at net/ipv4/tcp.c (625)" may be printed to the console.
```

This is normally a harmless message, but the resulting effect is in the TOE hardware. The hardware will reduce the number of total available connections by one. This means that instead of being capable of 31488 active connections, the TOE will now only handle 31487 active connections.

FAQ

Frequently asked questions regarding TOE and Chelsio's 10Gb Ethernet adapters.

Q. *During boot, my Linux 2.6.12 kernel is getting stuck at the RedHat init stage, I receive an error message stating "unable to open initial console".*

- A. RedHat Enterprise Linux 4 uses a filesystem called udev (Userspace dev filesystem) instead of the traditional dev filesystem. If using a kernel without an initrd image, it is probable that the kernel will not be able to obtain an initial console from /dev, since the udev filesystem has not yet started, and /dev is empty.

To avoid this problem, either build an initrd, OR, copy a few entries into the /dev directory.

Use the following steps to copy the necessary entries into /dev:

1. Boot into the stock RedHat 4 kernel.
2. Login as root.
3. Make a temporary directory for udev:
`mkdir /tmp/dev`
4. Temporarily relocate the udev filesystem:
`mount --move /dev /tmp/dev`
5. Copy the following files to the /dev directory:
`cp -a /tmp/dev/{console,null,zero} /dev/`

NOTE: Be sure to use the `-a` option, otherwise the files will be copied as regular files and not as special character/block devices!!

6. Relocate the udev filesystem:
`mount --move /tmp/dev /dev`
7. Reboot the system into the Linux-2.6.12 kernel.

It is also possible to use MAKEDEV to create new dev entries. See the MAKEDEV manpage on how to do this to create the above entries.

This fix provides the necessary character devices for the kernel to get an initial console. Once the RedHat INIT starts to boot, the udev filesystem will be mounted over the entries created in /dev. These entries will always be available to the kernel prior to udev mounting over them, so it is completely safe for these entries to co-exist with udev and the RedHat kernel.

Q. *Why does ethtool report a connection speed of unknown (10000) instead of 10Gb?*

- A. The ethtool utility, as of version 3 (Jan 27, 2005), does not provide full support of 10Gb network interfaces. There is a patch available, which has not yet made it into ethtool's source tree, which can be applied to the ethtool source. The ethtool utility will then need to be recompiled and installed on the system. See the "[1170813] Patch for correctly reporting 10Gb/s support" ID at <http://sourceforge.net/projects/gkernel/> for more information and to download the patch.

Q. *Why can't I enable TSO on the T204 card?*

- A. The T204 Quad 1Gb Protocol Engine does not support TSO when in TOE or NIC mode. The card is designed for full offload capability and when run in NIC, even though offload is disabled, the host stack TSO feature cannot be used.

Q. I have a Supermicro X6DH8-G2 motherboard and I cannot get 7.4Gbps performance with the T210? Why?

- A. The Supermicro X6DH8-G2 cannot deliver high performance because it connects the PCI-X 133 to the PXH. The PXH itself is connected to the MCH via PCI-E x4, which is not capable of delivering enough performance. The PCI-E x4 interface provides a peak of 8Gb/s in each direction. Assuming, however, a PCI-E maximum payload size of 128B, 64b addressing, and ECRC, there are 5DWs of overhead for each 128B of payload. This results in an 86% efficiency. So $8\text{Gb/s} * 0.86 = 6.88\text{Gb/s}$. This is the theoretical max BW, therefore, in each direction for the aggregate traffic on the PCI-X buses. Testing has shown that the typical goodput seen on this motherboard is ~5.8Gbps.

NOTE: The X6DH8-XB connects the PXH to the MCH via PCI-E x8 and thus provides the 7.4Gbps performance.

Q. How do I verify the Chelsio card is running at PCI-X 133MHz?

- A. Use the `dmesg` and `grep` for Chelsio to find the values.

```
# dmesg | grep Chelsio
Chelsio TOE Network Driver - version 2.1.1
eth1: Chelsio T204 4x100/1000BaseT TOE (rev 3), PCI-X 133MHz/64-bit
eth5: Chelsio T210 1x10GBaseX TOE (rev 3), PCI-X 133MHz/64-bit
```

Q. Why am I not able to get 7.4Gbps with T210?

- A. There can be many reasons for not getting the expected performance.
1. The processor must be powerful enough to handle high-performance networking. The following processors are recommended for high-performance applications:

AMD Opteron 250+ @ 2.2Ghz
Intel Xeon with EM64T @ 3.4Ghz

NOTE: Using a dual-processor 2Ghz Xeon, expected goodput is ~4.5Gbps. At least one CPU will be 100% utilized mainly doing the data copy between kernel and user application.

2. If using an Intel XEON processor, be sure to select a motherboard with an 800Mhz FSB (Front Side Bus), check the motherboard manufacturer's manual or website. Also, be sure to use the T210 in a PCI-X 133Mhz slot with no other devices sharing the same bus segment.

WARNING: Some new PCI Express motherboards are shipping that do not have full PCI-X 133HMz bus speed support. This is because the PCI-X backbone is really a 4x PCI Express. This is slower than PCI-X 133 and thus performance is limited. Check the motherboard manual for details.

NOTE: A 100MHz PCI-X bus speed is expected to deliver ~6Gbps goodput.

3. Use faster memory. One major impact to performance is the memory copy for userspace. By using the fastest possible memory speed, this decreases the memory copy time. Use DDR2-400+ or DDR-400+ memory. Check the motherboard for supported memory speeds.

4. The PCI Latency configuration settings should be set to 0xF800. This setting increases the number of cycles that the PCI device can hold on the bus and thus improves performance.

Use `lspci` to get the PCI device ID for the card:

```
# lspci -d 1425: | awk '{print $1}'
03:01.0
03:02.0
```

Use `setpci` to view the PCI latency value:

```
# setpci -s 03:02.0 0x0c.1
0000f800
```

Use `setpci` to set the PCI latency value for a device:

```
# setpci -s 03:02.0 0x0c.1=0x0000f800
```

5. The PCI burst configuration setting should be set to 0x4a0007. This setting increases the read burst amount on the PCI bus.

Use `setpci` to set the PCI burst configuration:

```
# setpci -s 03:02.0 0x60.1=0x004a0007
```

WARNING: If using an AMD-8131 HyperTransport PCI-X Tunnel chipset (rev12), the system could experience the "133-Mhz Mode Split Completion Data Corruption" bug identified by AMD while using a 133Mhz PCI-X card on the bus PCI-X bus. If experiencing this bug, AMD recommends setting the PCI data length to 1kB and the number of outstanding transactions to 2. This translates to a value of 0x00160007. This may significantly decrease performance throughput. AMD should have a fix for this bug in rev 13 of the chipset. Please refer to AMD 8131-HT/PCI-X Errata 26310 Rev 3.08 August 2004, section 56, "133-MHz Mode Split Completion Data Corruption" for more details with this bug.

6. If using a dual-processor (SMP) machine, set the CPU affinity mask for the Chelsio adapter interrupts to a single CPU. Each CPU is assigned a bit starting with bit 0 for CPU 0.

Identify the interrupt of the Chelsio device by using `ifconfig` or checking:

```
# cat /proc/interrupts
```

Write to the proc filesystem the mask for the IRQ number identified, in this case, IRQ 29:

```
echo 1 > /proc/irq/29/smp_affinity
```

7. Verify the TOE card is in offload mode:

```
# cat /proc/net/toe/devices
Device      Offload Module  Interfaces
eth0        Chelsio_T2     eth0
```

Customer Support

The Terminator2 ASIC is an extremely complex design with a large feature set which includes many tunable parameters. Some of these features may be of interest to advanced users who have a thorough understanding of the TCP standards. Chelsio may be able to provide advanced tools and documentation to these customers or OEM customers. Please contact Chelsio via email at support@chelsio.com to request additional information.

If you have problems with the software or hardware, please contact our customer support team via email at support@chelsio.com or check our website at <http://www.chelsio.com>.

Chelsio Communications, Inc.
370 San Aleso Ave.
Suite 100
Sunnyvale, CA 94085
<http://www.chelsio.com>

Copyright 2005 Chelsio Communications, Inc. All rights reserved.