When setting up a target node to use Linux, you need to perform the following tasks:

1. Patch, compile, and install a Linux kernel with RTAI support and install RTAI.
2. Load the RTAI modules at system boot.
3. Modify the `/etc/hosts` file.
4. Change the attributes of certain files and directories.
5. Create a Virgo2000 user.
6. Install RTNet (optional).

You perform these tasks in a Linux shell, logged in as `root`, and require the use of a text editor such as VIM (vi enhanced). If you are not familiar with VIM, see the Performing basic tasks in the VIM (vi enhanced) editor section on page 3.

There are many versions of both the Linux kernel and RTAI that can be used on the target. In this example, we will be using version 2.2.16 of the kernel and version 1.7 of RTAI because they are known to be stable. However, keep in mind that this might change in the future. We recommend using **Linux Red Hat Linux 6.2** for compiling the kernel and developing because the compiler (egcs-2.91.66) included in this version of Linux is known to be stable.

To patch, compile, and install a Linux kernel with RTAI support and install RTAI

1. Get a clean source tree of the 2.2.16 Linux kernel and copy it in the `/usr/src` directory under the following name:
   `/usr/src/linux-2.2.16`
2. Get a clean source tree of RTAI-1.7 and also copy it in the `/usr/src` directory under the following name:
   `/usr/src/rtai-1.7`
3. Replace the following files with those provided in the `isagraf` directory of **ISaGRAF**'s Linux installation CD-ROM:
   `/usr/src/rtai-1.7/include/rtai_sched.h`
   `/usr/src/rtai-1.7/lxrt/lxrt.c`
   `/usr/src/rtai-1.7/lxrt/rtai_lxrt.h`
   `/usr/src/rtai-1.7/upscheduler/up_sched.c`
4. Edit the `/usr/src/rtai-1.7/README.QUICKINSTALL` file, then patch and compile the kernel by performing steps 1 to 12 from the file itself. In step 7, type `make xconfig` and make sure that the appropriate network drivers are enabled in the Ethernet (10 or 100Mbit) or Ethernet (1000Mbit) sub-section of the Network device support section. You enable network drivers by selecting `y` next to the driver(s). Next to the **Kernel module loader** sub-section of the **Loadable module support** section, make sure to select `n`. In the **Processor family** sub-section of the **Processor type and features** section, check that the correct processor has been selected. The remaining kernel parameters should be correct by default. Finally, in step 12, you don't need to do anything since we will be using the UP scheduler.
5. Make sure the links are correct in the `/usr/src/rtai/modules` directory:
   a) Type `cd /usr/src/rtai/modules`, then press **Enter**.
   b) Type `rm -f *`, then press **Enter**.
   c) Type `ln -s ../lxrt/lxrt.o lxrt.o` , then press **Enter**.
   d) Type `ln -s ../rtai/rtai_up.o rtai.o`, then press **Enter**.
   e) Type `ln -s ../newfifos/rtai_fifos.o rtai_fifos.o`, then press **Enter**.
   f) Type `ln -s ../upscheduler/rtai_sched_up.o rtai_sched.o`, then press **Enter**.
   g) Type `ln -s ../shmem/rtai_shm.o rtai_shm.o`, then press **Enter**.

6. Perform steps 13 and 14 of the `README.QUICKINSTALL` file located in the following directory:
   `/usr/src/rtai-1.7`

To load the RTAI modules at system boot

Loading the RTAI modules at system boot involves modifying the system startup script. The name of this file depends on the version of Linux you are using. For instance, on Red Hat Linux, this file is `/etc/rc.d/rc.sysinit` and on Slackware Linux, the file is `/etc/rc.d/rc.S`.

1. Edit the file corresponding to your Linux version by adding the following lines at the end of the file:
   ```
   /sbin/insmod /lib/modules/2.2.16-rthal3/misc/rtai.o

   /sbin/insmod /lib/modules/2.2.16-rthal3/misc/rtai_sched.o

   /sbin/insmod /lib/modules/2.2.16-rthal3/misc/lxrt.o
   ```
2. Save changes to the file and exit the editor.
3. Reboot your computer by typing `reboot`, then pressing **Enter**.
4. Log in as `root` and check that all your RTAI modules have been loaded by typing `lsmod`, then pressing **Enter**. You should at least see the following modules: **rtai.o**, **rtai_sched.o**, and **lxrt.o**.

To modify the /etc/hosts file

1. Edit the `/etc/hosts` file:
   - At the end of the line corresponding to your IP address, add the word `mystandardip`.
2. Save changes to the file and exit the editor.

To change attributes of certain files and directories

1. Change the mode of the reboot file:
   - Type `chmod +s /sbin/reboot`, then press **Enter**.
2. Change the mode of the :
   - Type `chmod -R 777 /etc/rc.d`, then press **Enter**.
3. Change the mode of the :
   - Type `chmod 777 /usr`, then press **Enter**.

To create the Virgo2000 user

1. Create the user by typing the following then pressing **Enter**:
   `useradd -d/usr/v2000 -m -g0 Virgo2000`
2. Type `passwd Virgo2000`, then press **Enter**. When prompted to provide a password, type `v2000`.
3. Change the user directory permissions by typing the following then pressing **Enter**:
   `chmod 777 /usr/v2000`

To install RTNet (optional)

The installation of the 0.9.0 version of RTNet (Real-time UDP networking) is required to use the failover feature of the product. For more details on installing RTNet, refer to the *Using RTNet Networks to Link Target Nodes* application note. In addition to what you will find in that application note, the following must be taken into account before installing RTNet:

1. When compiling the kernel, all your network drivers must be defined as modules; drivers must not be linked with the kernel. You load drivers as modules by selecting `m` next to the driver in step 7 of the `/usr/src/rtai-1.7/README.QUICKINSTALL` file.
2. Get a clean source tree of RTNet-0.9.0 and copy it in the `/usr/src` directory under the following name
   `/usr/src/rtnet-0.9.0`

3. Replace the `/usr/src/rtnet-0.9.0/rtnet/socket.c` file with the one provided in the `isagraf` directory of **ISaGRAF**'s Linux installation CD-ROM.

4. Edit the `/usr/src/rtnet-0.9.0/INSTALL` file and follow the instructions to compile and install the RTNet package.

5. To create the RTNet device, type `make dev`, then press **Enter**.

Performing basic tasks in the VIM (vi enhanced) editor

- Edit a file by typing `vi` *file_name*, then pressing **Enter**. The vi editor opens in command mode.
- Switch to edit mode by pressing INSERT. In edit mode, the vi editor behaves much like a standard editor.
- Save changes and exit a file, by pressing ESC to switch to command mode, then typing `:wq` and pressing ENTER.