Red Hat Linux 7.2

The Official Red Hat Linux Customization Guide

ISBN: N/A

Red Hat, Inc.

2600 Meridian Parkway
Durham, NC 27713 USA
+1 919 547 0012 (Voice)
+1 919 547 0024 (FAX)
888 733 4281 (Voice)
P.O. Box 13588
Research Triangle Park, NC 27709 USA

rhl-cg(EN)-7.2-Print-RHI (2001-08-30T14:29-0400)

Printed in Canada, Ireland, and Japan

This manual is dedicated to Carole Williams, a valuable contributor to the Red Hat documentation team. Carole, we wish you all the best in your future endeavors. We miss your wisdom, superior editing skills, ability to write humor into just about any topic, and jokes that made each day a joy to work with you. Every time we eat a piece of chocolate we will think of you!

Contents

Red Hat Linux 7.2

Introduction

Welcome to the *Official Red Hat Linux Customization Guide*.

The *Official Red Hat Linux Customization Guide* contains information on how to customize your Red Hat Linux system to fit your needs. If you are looking for step-by-step, task-oriented guides for configuring and customizing your system, this is the guide for you. This manual discusses many intermediate topics such as the following:

- Setting up a network interface card (NIC)
- Performing a Kickstart installation
- Configuring Samba shares
- Managing your software with RPM
- Upgrading your kernel

This manual is divided into the following main categories:

- Installation-Related Reference
- Network-Related Reference
- System Configuration
- Package Management

This guide assumes you have a basic understanding of your Red Hat Linux system. If you need reference material which covers more basic issues, please refer to the *Official Red Hat Linux Getting Started Guide*. If you need more advanced documentation, please refer to the *Official Red Hat Linux Reference Guide*.

HTML and PDF versions of all the Official Red Hat Linux manuals are available online at http://www.redhat.com/support/manuals/.

Document Conventions

When you read this manual, you will see that certain words are represented in different fonts, typefaces, sizes and weights. This highlighting is systematic; different words are represented in the same style to indicate their inclusion in a specific category. The types of words that are represented this way include the following:

`command`

Linux commands (and other operating system commands, when used) are represented this way. This style should indicate to you that you can type in the word or phrase on the command line and press [Enter] to invoke a command. Sometimes a command contains words that would be displayed in a different style on their own (e.g., filenames). In these cases, they are considered to be part of the command, so the entire phrase will be displayed as a command. For example:

Use the `cat testfile` command to view the contents of a file, named `testfile`, in the current working directory.

`filename`

Filenames, directory names, paths and RPM package names are represented this way. This style should indicate that a particular file or directory exists by that name on your Red Hat Linux system. Examples:

The `.bashrc` file in your home directory contains bash shell definitions and aliases for your own use.

The `/etc/fstab` file contains information about different system devices and filesystems.

The `/usr/share/doc` directory contains documentation for various programs.

Install the `webalizer` RPM if you want to use a Web server log file analysis program.

application

This style should indicate to you that the program named is an end-user application (as opposed to system software). For example:

Use Netscape Navigator to browse the Web.

[key]

A key on the keyboard is shown in this style. For example:

To use [Tab] completion, type in a character and then press the [Tab] key. Your terminal will display the list of files in the directory that start with that letter.

[key]-[combination]

A combination of keystrokes is represented in this way. For example:

The [Ctrl]-[Alt]-[Backspace] key combination will restart the X Window System.

text found on a GUI interface

A title, word or phrase found on a GUI interface screen or window will be shown in this style. When you see text shown in this style, it is being used to identify a particular GUI screen or an element on a GUI screen (e.g., text associated with a checkbox or field). Examples:

On the GNOME **Control Center** screen, you can customize your GNOME window manager.

Select the **Require Password** checkbox if you would like your screensaver to require a password before stopping.

top level of a menu on a GUI screen or window

When you see a word in this style, it indicates that the word is the top level of a pulldown menu. If you click on the word on the GUI screen, the rest of the menu should appear. For example:

Under **Settings** on a GNOME terminal, you will see the following menu items: **Preferences**, **Reset Terminal**, **Reset and Clear**, and **Color selector**.

If you need to type in a sequence of commands from a GUI menu, they will be shown like the following example:

Click on **Programs**=>**Applications**=>**Emacs** to start the Emacs text editor.

button on a GUI screen or window

This style indicates that the text will be found on a clickable button on a GUI screen. For example:

Click on the **Back** button to return to the Web page you last viewed.

`computer output`

When you see text in this style, it indicates text displayed by the computer on the command line. You will see responses to commands you typed in, error messages and interactive prompts for your input during scripts or programs shown this way. For example:

Use the `ls` to display the contents of a directory:

```
$ ls
Desktop                axhome         logs         paulwesterberg.gif
Mail                   backupfiles    mail         reports
```

The output returned in response to the command (in this case, the contents of the directory) is shown in this style.

`prompt`

A prompt, which is a computer’s way of signifying that it is ready for you to input something, will be shown in this style. Examples:

`$`

`#`

`[stephen@maturin stephen]$`

`leopard login:`

`user input`

Text that the user has to type, either on the command line, or into a text box on a GUI screen, is displayed in this style. In the following example, **`text`** is displayed in this style:

To boot your system into the text based installation program, you will need to type in the **`text`** command at the `boot:` prompt.

Another example, with the word **`root`** displayed as something the user needs to type in:

If you need to log in as root when you first log into your system, and you are using the graphical login screen, at the `Login` prompt, type **`root`**. At the `Password` prompt, type in the root password.

glossary entry

A word that appears in the glossary will be shown in the body of the document in this style. For example:

The lpd **daemon** handles printing requests.

In this case, the style of the word **daemon** should indicate to you that a definition of the term is available in the glossary.

Additionally, we use several different strategies to draw your attention to certain pieces of information. In order of how critical the information is to your system, these items will be marked as a note, a caution or a warning. For example:

Note

Remember that Linux is case sensitive. In other words, a rose is not a ROSE is not a rOsE.

CAUTION

Do not do routine tasks as root — use a regular user account unless you need to use the root account to administer your system.

WARNING

If you choose not to partition manually, a server installation will remove all existing partitions on all installed hard drives. Do not choose this installation class unless you are sure you have no data you need to save.

More to Come

The *Official Red Hat Linux Customization Guide* is part of Red Hat's growing commitment to provide useful and timely support to Red Hat Linux users. As new tools and applications are released, this guide will be expanded to include them.

Send in Your Feedback

If you spot a typo in the *Official Red Hat Linux Customization Guide*, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla (http://www.redhat.com/bugzilla) against the component rhl-cg.

Be sure to mention the manual's identifier:

```
rhl-cg(EN)-7.2-Print-RHI (2001-08-30T14:29-0400)
```

If you mention this manual's identifier, we will know exactly which version of the guide you have.

If you have a suggestion for improving the documentation, try to be as specific as possible. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Sign Up for Support

If you have an official edition of Red Hat Linux 7.2, please remember to sign up for the benefits you are entitled to as a Red Hat customer.

You will be entitled to any or all of the following benefits, depending upon the Official Red Hat Linux product you purchased:

- Official Red Hat support — Get help with your installation questions from Red Hat, Inc.'s support team.
- Red Hat Network — Easily update your packages and receive security notices that are customized for your system. Go to http://rhn.redhat.com for more details.

- *Under the Brim: The Official Red Hat E-Newsletter* — Every month, get the latest news and product information directly from Red Hat.

To sign up, go to http://www.redhat.com/apps/activate/. You will find your Product ID on a black, red, and white card in your Official Red Hat Linux box.

To read more about technical support for Official Red Hat Linux, refer to the *Getting Technical Support* Appendix in the *Official Red Hat Linux Installation Guide*.

Good luck, and thank you for choosing Red Hat Linux!

The Red Hat Documentation Team

Part I Installation-Related Reference

1 Kickstart Installations

1.1 What are Kickstart Installations?

Many system administrators would prefer to use an automated installation method to install Red Hat Linux on their machines. To answer this need, Red Hat created the kickstart installation method. Using kickstart, a system administrator can create a single file containing the answers to all the questions that would normally be asked during a typical Red Hat Linux installation.

Kickstart files can be kept on single server system, and read by individual computers during the installation. This installation method can support the use of a single kickstart file to install Red Hat Linux on multiple machines, making it ideal for network and system administrators.

Kickstart lets you automate most of a Red Hat Linux installation, including:

- Language selection
- Mouse configuration
- Keyboard selection
- Boot loader installation
- Disk partitioning
- Network configuration
- NIS, LDAP, Kerberos, Hesiod, and Samba authentication
- Firewall configuration
- Package selection
- X Window System configuration

1.2 How Do You Perform a Kickstart Installation?

Kickstart installations can be performed using a local CD-ROM, a local hard drive, or via NFS, FTP or HTTP.

To use kickstart mode, you must first create a kickstart file (`ks.cfg`), and make it available to the Red Hat Linux installation program.

1.2.1 Where to Put A Kickstart File

A kickstart file must be placed in one of two locations:

- On a boot disk
- On a network

Normally a kickstart file is copied to the boot disk, or made available on the network. The network-based approach is most commonly used, as most kickstart installations tend to be performed on networked computers.

Let us take a more in-depth look at where the kickstart file may be placed.

To perform a diskette-based kickstart installation, the kickstart file must be named `ks.cfg` and must be located in the boot disk's top-level directory. Note that the Red Hat Linux boot disks are in MS-DOS format, so it is easy to copy the kickstart file under Linux using the `mcopy` command:

```
mcopy ks.cfg a:
```

Alternatively, you can use Windows to copy the file. You can also mount the MS-DOS boot disk and `cp` the file over. Although there's no technological requirement for it, most diskette-based kickstart installations install Red Hat Linux from a local CD-ROM.

Network installations using kickstart are quite common, because system administrators can easily automate the installation on many networked computers quickly and painlessly. In general, the approach most commonly used is for the administrator to have both a BOOTP/DHCP server and an NFS server on the local network. The BOOTP/DHCP server is used to give the client system its networking information, while the actual files used during the installation are served by the NFS server. Often, these two servers run on the same physical machine, but they are not required to.

To perform a network-based kickstart installation, you must have a BOOTP/DHCP server on your network, and it must include configuration information for the machine on which you are attempting to install Red Hat Linux. The BOOTP/DHCP server will provide the client with its networking information as well as the location of the kickstart file.

If a kickstart file is specified by the BOOTP/DHCP server, the client system will attempt an NFS mount of the file's path, and will copy the specified file to the client, using it as the kickstart file. The exact settings required vary depending on the BOOTP/DHCP server you use.

Here's an example of a line from the `dhcpd.conf` file for the DHCP server shipped with Red Hat Linux:

```
filename "/usr/new-machine/kickstart/";
next-server blarg.redhat.com;
```

Note that you should replace the value after `filename` with the name of the kickstart file (or the directory in which the kickstart file resides) and the value after `next-server` with the NFS server name.

If the filename returned by the BOOTP/DHCP server ends with a slash ("/"), then it is interpreted as a path only. In this case, the client system mounts that path using NFS, and searches for a particular file. The filename the client searches for is:

```
<ip-addr>-kickstart
```

The `<ip-addr>` section of the filename should be replaced with the client's IP address in dotted decimal notation. For example, the filename for a computer with an IP address of 10.10.0.1 would be `10.10.0.1-kickstart`.

Note that if you don't specify a server name, then the client system will attempt to use the server that answered the BOOTP/DHCP request as its NFS server. If you don't specify a path or filename, the client system will try to mount `/kickstart` from the BOOTP/DHCP server, and will try to find the kickstart file using the same `<ip-addr>-kickstart` filename as described above.

1.3 Starting a Kickstart Installation

To begin a kickstart installation, you must boot the system from a Red Hat Linux boot diskette or the CD-ROM and enter a special boot command at the boot prompt. If the kickstart file is located on a boot diskette that was created from the `boot.img` or `bootnet.img` image file, the correct boot command would be:

```
boot: linux ks=floppy
```

The **`linux ks=floppy`** command also works if the `ks.cfg` file is located on a vfat filesystem on a floppy diskette and you boot from the Red Hat Linux CD-ROM.

An alternate boot command for booting off the Red Hat Linux CD-ROM and having the kickstart file on a vfat filesystem on a floppy diskette is:

```
boot: linux ks=hd:fd0/ks.cfg
```

If you need to use a driver disk with kickstart, you can still have the kickstart file on a floppy disk:

```
boot: linux ks=floppy dd
```

The Red Hat Linux installation program looks for a kickstart file if the `ks` command line argument is passed to the kernel. The command line argument can take a number of forms:

`ks=nfs:`*`<server>:/<path>`*

The installation program will look for the kickstart file on the NFS server <*server*>, as file <*path*>. The installation program will use DHCP to configure the Ethernet card. For example, if your NFS server is server.example.com and the kickstart file is in the NFS share /mydir/ks.cfg, the correct boot command would be `ks=nfs:server.example.com:/mydir/ks.cfg`.

`ks=http:`*`<server>:/<path>`*

The installation program will look for the kickstart file on the HTTP server <*server*>*:*, as file <*path*>. The installation program will use DHCP to configure the Ethernet card. For example, if your HTTP server is server.example.com and the kickstart file is in the HTTP directory /my-dir/ks.cfg, the correct boot command would be `ks=http:server.example.com:/my-dir/ks.cfg`.

`ks=floppy`

The installation program looks for the file `ks.cfg` on a vfat filesystem on the floppy in drive /dev/fd0.

`ks=hd:`*`<device>/<file>`*

The installation program will mount the filesystem on <*device*> (which must be vfat or ext2), and look for the kickstart configuration file as <*file*> in that filesystem (for example, `ks=hd:sda3/mydir/ks.cfg`).

`ks=file:/`*`<file>`*

The installation program will try to read the file <*file*> from the filesystem; no mounts will be done. This is normally used if the kickstart file is already on the `initrd` image.

`ks=cdrom:/`*`<path>`*

The installation program will look for the kickstart file on CD-ROM, as file <*path*>.

`ks`

If `ks` is used alone, the installation program will configure the Ethernet card in the system using DHCP. The system will use the "bootServer" from the DHCP response as an NFS server to read the kickstart file from (by default, this is the same as the DHCP server). The name of the kickstart file is one of the following:

- If DHCP is specified and the bootfile begins with a `/`, the bootfile provided by DHCP is looked for on the NFS server.
- If DHCP is specified and the bootfile begins with something other then a `/`, the bootfile provided by DHCP is looked for in the `/kickstart` directory on the NFS server.
- If DHCP did not specify a bootfile, then the installation program tries to read the file `/kickstart/1.2.3.4-kickstart`, where *1.2.3.4* is the numeric IP address of the machine being installed.

`ksdevice=`*`<device>`*

The installation program will use this network device to connect to the network. For example, to start a kickstart installation with the kickstart file on an nfs server that is connected to the system through the eth1 device, use the command `ks=nfs:<server:>/<path> ksdevice=eth1` at the `boot:` prompt.

1.4 The Kickstart File

Now that you have some background information on kickstart installations, let’s take a look at the kickstart file itself. The kickstart file is a simple text file, containing a list of items, each identified by a keyword. You can create it by editing a copy of the `sample.ks` file found in the `RH-DOCS` directory of the Red Hat Linux Documentation CD-ROM, using the Kickstart Configurator application, or writing it from scratch. You should be able to edit it with any text editor or word processor that can save files as ASCII text.

First, be aware of the following issues when you are creating your kickstart file:

- Items must be specified *in order*. That order is:
  - Command section — Refer to Section 1.5, *Kickstart Options* for a list of kickstart options. You must include the required options.
  - The `%packages` section — Refer to Section 1.5.29, *`%packages` — Package Selection* for details.
  - The `%pre` and `%post` sections — These two sections can be in any order and are not required. Refer to Section 1.5.30, *`%pre` — Pre-Installation Configuration Section* and Section 1.5.31, *`%post` — Post-Installation Configuration Section* for details.
- Items that are not required can be omitted.
- Omitting any required item will result in the installation program prompting the user for an answer to the related item, just as the user would be prompted during a typical installation. Once the answer is given, the installation will continue unattended (unless it finds another missing item).
- Lines starting with a pound sign ("#") are treated as comments, and are ignored.
- For kickstart *upgrades*, the following items are required:
  - Language
  - Installation method
  - Device specification (if device is needed to perform installation)
  - Keyboard setup
  - The `upgrade` keyword
  - LILO configuration

If any other items are specified for an upgrade, those items will be ignored (note that this includes package selection).

1.5 Kickstart Options

The following options can be placed in a kickstart file. If you prefer to use a graphical interface for creating your kickstart file, you can use the Kickstart Configurator application. Refer to Chapter 2, *Kickstart Configurator* for details.

1.5.1 `autostep`

`autostep` (optional) [1]

Similar to `interactive` except it goes to the next screen for you. It is used mostly for debugging.

1.5.2 `auth` — Authentication Options

`auth` or `authconfig` (required)

Sets up the authentication options for the system. It's similar to the `authconfig` command, which can be run after the install. By default, passwords are normally encrypted and are not shadowed.

`--enablemd5`

Use md5 encryption for user passwords.

`--enablenis`

Turns on NIS support. By default, `--enablenis` uses whatever domain it finds on the network. A domain should almost always be set by hand (via `--nisdomain`).

`--nisdomain`

NIS domain name to use for NIS services.

`--nisserver`

Server to use for NIS services (broadcasts by default).

`--useshadow` or `--enableshadow`

Use shadow passwords.

`--enableldap`

Turns on LDAP support in `/etc/nsswitch.conf`, allowing your system to retrieve information about users (UIDs, home directories, shells, etc.) from an LDAP directory.

To use this option, you must have the `nss_ldap` package installed. You must also specify a server and a base DN.

`--enableldapauth`

Use LDAP as an authentication method. This enables the `pam_ldap` module for authentication and changing passwords, using an LDAP directory. To use this option, you must have the `nss_ldap` package installed. You must also specify a server and a base DN.

`--ldapserver=`

If you specified either `--enableldap` or `--enableldapauth`, the name of the LDAP server to use. This option is set in the `/etc/ldap.conf` file.

`--ldapbasedn=`

The DN (distinguished name) in your LDAP directory tree under which user information is stored. This option is set in the `/etc/ldap.conf` file.

`--enableldaptls`

Use TLS (Transport Layer Security) lookups. This option allows LDAP to send encrypted usernames and passwords to an LDAP server before authentication.

`--enablekrb5`

Use Kerberos 5 for authenticating users. Kerberos itself does not know about home directories, UIDs, or shells. So if you enable Kerberos you will need to make users' accounts known to this workstation by enabling LDAP, NIS, or Hesiod or by using the `/usr/sbin/useradd` command to make their accounts known to this workstation. If you use this option, you must have the `pam_krb5` package installed.

`--krb5realm`

The Kerberos 5 realm to which your workstation belongs.

`--krb5kdc`

The KDC (or KDCs) that serve requests for the realm. If you have multiple KDCs in your realm, separate their names with commas (,).

`--krb5adminserver`

The KDC in your realm that is also running kadmind. This server handles password changing and other administrative requests. This server must be run on the master KDC if you have more than one KDC.

`--enablehesiod`

Enable Hesiod support for looking up user home directories, UIDs, and shells. More information on setting up and using Hesiod on your network is in `/usr/share/doc/glibc-2.x.x/README.hesiod`, which is included in the `glibc` package. Hesiod is an extension of DNS that uses DNS records to store information about users, groups, and various other items.

`--hesiodlhs`

The Hesiod LHS ("left-hand side") option, set in `/etc/hesiod.conf`. This option is used by the Hesiod library to determine the name to search DNS for when looking up information, similar to LDAP's use of a base DN.

`--hesiodrhs`

The Hesiod RHS ("right-hand side") option, set in `/etc/hesiod.conf`. This option is used by the Hesiod library to determine the name to search DNS for when looking up information, similar to LDAP's use of a base DN.

Tip

To look up user information for "jim", the Hesiod library looks up *jim.passwd<LHS><RHS>*, which should resolve to a TXT record that looks like what his passwd entry would look like (`jim:*:501:501:Jungle Jim:/home/jim:/bin/bash`). For groups, the situation is identical, except *jim.group<LHS><RHS>* would be used.

Looking up users and groups by number is handled by making "501.uid" a CNAME for "jim.passwd", and "501.gid" a CNAME for "jim.group". Note that the LHS and RHS do not have periods [.] put in front of them when the library determines the name for which to search, so the LHS and RHS usually begin with periods.

`--enablesmbauth` [1]

Enables authentication of users against an SMB server (typically a Samba or Windows server). SMB authentication support does not know about home directories, UIDs, or shells. So if you enable it you will need to make users' accounts known to the workstation by enabling LDAP, NIS, or Hesiod or by using the `/usr/sbin/useradd` command to make their accounts known to the workstation. To use this option, you must have the `pam_smb` package installed.

`--smbservers=` [1]

The name of the server(s) to use for SMB authentication. To specify more than one server, separate the names with commas (,).

`--smbworkgroup=` [1]

The name of the workgroup for the SMB servers.

`--enablecache` [1]

Enables the `nscd` service. The `nscd` service caches information about users, groups, and various other types of information. Caching is especially helpful if you choose to distribute information about users and groups over your network using NIS, LDAP, or hesiod.

1.5.3 `bootloader`

`bootloader` (required) [1]

Specifies how the boot loader should be installed and whether the bootloader should be LILO or GRUB.

`--append`

Specifies kernel parameters.

`--location=`

Specifies where the boot record is written. Valid values are the following: **`mbr`** (the default), **`partition`** (installs the boot loader on the first sector of the partition containing the kernel), or **`none`** (do not install the boot loader).

`--password=`*`mypassword`*

If using GRUB, sets the GRUB bootloader password to *mypassword*. This should be used to restrict access to the GRUB shell where arbitrary kernel options can be passed.

`--md5pass=`*`mypassword`*

If using GRUB, similar to `--password` except *mypassword* should be the password already encrypted.

`--useLilo`

Use LILO instead of GRUB as the boot loader.

`--linear`

If using LILO, use the `linear` LILO option; this is only for backwards compatibility (and linear is now used by default).

--nolinear

If using LILO, use the `nolinear` LILO option; linear is the default.

--lba32

If using LILO, force use of lba32 mode instead of autodetecting.

1.5.4 `clearpart` — Removing Partitions Based On Partition Type

`clearpart` (optional)

Removes partitions from the system, prior to creation of new partitions. By default, no partitions are removed.

--linux

Erases all Linux partitions.

--all

Erases all partitions from the system.

--drives [1]

Specifies which drives to clear partitions from.

--initlabel [1]

Initializes the disk label to the default for your architecture (`msdos` for x86 and `gpt` for Itanium). It is useful so that the installation program does not ask if it should initialize the disk label if installing to a brand new hard drive.

1.5.5 device

`device` (optional)

On most PCI systems, the installation program will autoprobe for Ethernet and SCSI cards properly. On older systems and some PCI systems, however, kickstart needs a hint to find the proper devices. The device command, which tells Anaconda to install extra modules, is in this format:

```
device <type> <moduleName> --opts <options>
```

<type> should be one of "scsi" or "eth", and *<moduleName>* is the name of the kernel module which should be installed.

--opts

Options to pass to the kernel module. Note that multiple options may be passed if they are put in quotes. For example:

```
--opts "aic152x=0x340 io=11"
```

1.5.6 deviceprobe

deviceprobe (optional)

Forces a probe of the PCI bus and loads modules for all the deviced found if a module is available.

1.5.7 driverdisk

driverdisk (optional)

Driver disks can be used during kickstart installations. You will need to copy the driver disk's contents to the root directory of a partition on the system's hard drive. Then you will need to use the `driverdisk` command to tell the installation program where to look for the driver disk.

```
driverdisk <partition> [--type <fstype>]
```

<partition> is the partition containing the driver disk.

--type

Filesystem type (for example, vfat, ext2, or ext3).

1.5.8 firewall

firewall (optional)

Firewall options can be configured in kickstart. This configuration corresponds to the **Firewall Configuration** screen in the installation program.

```
firewall [--high | --medium | --disabled] [--trust
<device>] [--dhcp] [--ssh] [--telnet] [--smtp] [--http]
[--ftp] [--port <portspec>]
```

Levels of security

Choose one of the following levels of security:

- `--high`
- `--medium`
- `--disabled`

`--trust` ***`<device>`***

Listing a device here, such as eth0, allows all traffic coming from that device to go through the firewall. To list more than one device, use `--trust eth0 --trust eth1`. Do NOT use a comma-separated format such as `--trust eth0, eth1`.

Allow incoming

Enabling these options allow the specified services to pass through the firewall.

- `--dhcp`
- `--ssh`
- `--telnet`
- `--smtp`
- `--http`
- `--ftp`

`--port` *<portspec>*

You can specify that ports be allowed through the firewall using the port:protocol format. For example, if you wanted to allow IMAP access through your firewall, you can specify `imap:tcp`. You can also specify numeric ports explicitly; for example, to allow UDP packets on port 1234 through, specify `1234:udp`. To specify multiple ports, separate them by commas.

1.5.9 `install`

`install` (optional)

Tells the system to install a fresh system rather than upgrade an existing system. This is the default mode.

1.5.10 Installation Methods

You must use one of these four commands to specify what type of kickstart installation is being performed:

`nfs`

Install from the NFS server specified.

- `--server` *`<server>`*

  Server from which to install (hostname or IP).

- `--dir` *`<dir>`*

  Directory containing the Red Hat installation tree.

For example:

```
nfs --server <server> --dir <dir>
```

`cdrom`

Install from the first CD-ROM drive on the system.

For example:

```
cdrom
```

`harddrive`

Install from a Red Hat installation tree on a local drive, which must be either vfat or ext2.

- `--partition` *`<partition>`*

  Partition to install from (such as, sdb2).

- `--dir` *`<dir>`*

  Directory containing the Red Hat installation tree.

For example:

```
harddrive --partition <partition> --dir <dir>
```

`url`

Install from a Red Hat installation tree on a remote server via FTP or HTTP.

For example:

```
url --url http://<server>/<dir>
url --url ftp://<username>:<password>@<server>/<dir>
```

1.5.11 interactive

`interactive` (optional) [1]

Uses the information provided in the kickstart file during the installation, but allow for inspection and modification of the values given. You will be presented with each screen of the installation program with the values from the kickstart file. Either accept the values by clicking **Next** or change the values and click **Next** to continue. See also Section 1.5.1, *autostep*.

1.5.12 keyboard

keyboard (required)

Sets system keyboard type. Here's the list of available keyboards on i386 and Alpha machines:

```
ANSI-dvorak, azerty, be-latin1, be2-latin1, bg, br-abnt2, cf,
croat, cz, cz-lat2, cz-lat2-prog, cz-us-qwertz, de, de-latin1,
de-latin1-nodeadkeys, defkeymap, defkeymap_V1.0, dk, dk-latin1,
dvorak, dvorak-l, dvorak-r, emacs, emacs2, es, es-cp850, et,
et-nodeadkeys, fi, fi-latin1, fr, fr-latin0, fr-latin1, fr-pc,
fr_CH, fr_CH-latin1, gr, gr-pc, hebrew, hu, hu101, is-latin1,
it, it-ibm, it2, jp106, la-latin1, lt, lt.l4, lv-latin4,
lv-latin7,mk, nl, nl-latin1, nl-latin1-nodeadkeys, no, no-latin1,
pc-dvorak-latin1, pc110, pl, pl1, pt-latin1, pt-old, ro, ru,
ru-cp1251, ru-ms, ru-yawerty, ru1, ru2, ru3, ru4, ru_win,
se-latin1, sg, sg-latin1, sg-latin1-lk450, sk-prog, sk-prog-qwerty,
sk-prog-qwerty, sk-qwerty, sk-qwertz, slovene, sr, sr, tr_f-latin5,
tr_q-latin5, tralt, trf, trq, ua, ua-utf, ua-utf-ws, ua-ws, uaw,
uaw_uni, uk, us, us-latin1, wangbe
```

Here's the list for SPARC machines:

```
sun-pl-altgraph, sun-pl, sundvorak, sunkeymap, sunt4-es,
sunt4-no-latin1, sunt5-cz-us, sunt5-de-latin1, sunt5-es,
sunt5-fi-latin1, sunt5-fr-latin1, sunt5-ru, sunt5-uk, sunt5-us-cz
```

1.5.13 lang

lang (required)

Sets the language to use during installation. For example, to set the language to English, the kickstart file should contain the following line:

```
lang en_US
```

Valid language codes are the following (please note that these are subject to change at any time):

```
cs_CZ, da_DK, en_US, fr_FR, de_DE, hu_HU, is_IS, it_IT,
ja_JP.eucJP, no_NO, ro_RO, sk_SK, sl_SI, sr_YU, es_ES,
ru_RU.KOI8-R, uk_UA.KOI8-U, sv_SE, tr_TR
```

1.5.14 langsupport

langsupport

Sets the language(s) to install on the system. The same language codes used with `lang` can be used with `langsupport`.

`--default` [1]

Sets the default language to use for any language-specific aspect of the installed system.

An example to install English and French and use English as the default language:

```
languagesupport --default en_US fr_FR
```

1.5.15 `lilo`

`lilo` (replaced by `bootloader`)

WARNING

This option has been replaced by `bootloader` and is only available for backwards compatibility. Refer to Section 1.5.3, *`bootloader`*.

Specifies how the boot loader should be installed on the system. By default, LILO installs on the MBR of the first disk, and installs a dual-boot system if a DOS partition is found (the DOS/Windows system will boot if the user types **`dos`** at the `LILO:` prompt).

`--append` <***params***>

Specifies kernel parameters.

`--linear`

Use the `linear` LILO option; this is only for backwards compatibility (and linear is now used by default).

`--nolinear`

Use the `nolinear` LILO option; linear is now used by default.

`--location`

Specifies where the LILO boot record is written. Valid values are the following: **`mbr`** (the default) or **`partition`** (installs the boot loader on the first sector of the partition containing the kernel). If no location is specified, LILO is not installed.

`--lba32` [1]

Forces the use of lba32 mode instead of autodetecting.

1.5.16 `lilocheck`

`lilocheck` (optional)

If `lilocheck` is present, the installation program checks for LILO on the MBR of the first hard drive, and reboots the system if it is found — in this case, no installation is performed. This can prevent kickstart from reinstalling an already installed system.

1.5.17 `mouse`

`mouse` (required)

Configures the mouse for the system, both in GUI and text modes. Options are:

`--device` ***<dev>***

Device the mouse is on (such as --device ttyS0).

`--emulthree`

If present, simultaneous clicks on the left and right mouse buttons will be recognized as the middle mouse button by the X Window System. This option should be used if you have a two button mouse.

After options, the mouse type may be specified as one of the following:

```
alpsps/2, ascii, asciips/2, atibm, generic, generic3,
genericps/2, generic3ps/2, genericusb, generic3usb,
geniusnm, geniusnmps/2,geniusprops/2, geniusscrollps/2,
thinking, thinkingps/2, logitech, logitechcc, logibm,
logimman, logimmanps/2, logimman+, logimman+ps/2,
logimmusb, microsoft, msnew, msintelli, msintellips/2,
msintelliusb, msbm, mousesystems, mmseries, mmhittab,
sun, none
```

If the mouse command is given without any arguments, or it is omitted, the installation program will attempt to autodetect the mouse. This procedure works for most modern mice.

1.5.18 `network`

`network` (optional)

Configures network information for the system. If the kickstart installation does not require networking (in other words, it is not installed over NFS, HTTP, or FTP), networking is not configured for the system. If the installation does require networking and network information is not provided in the kickstart file, the Red Hat Linux installation program assumes that the installation should be done over eth0 via a dynamic IP address (BOOTP/DHCP), and configures the final, installed system to determine its IP address dynamically. The `network` option configures networking information for kickstart installations via a network as well as for the installed system.

`--bootproto`

One of **`dhcp`**, **`bootp`**, or **`static`** (defaults to DHCP, and **`dhcp`** and **`bootp`** are treated the same). Must be **`static`** for static IP information to be used.

`--device` ***<device>***

Used to select a specific Ethernet device for installation. Note that using `--device` *<device>* will not be effective unless the kickstart file is a local file (such as ks=floppy), since the installation program will configure the network to find the kickstart file. Example:

```
network --bootproto dhcp --device eth0
```

`--ip`

IP address for the machine to be installed.

`--gateway`

Default gateway as an IP address.

`--nameserver`

Primary nameserver, as an IP address.

`--netmask`

Netmask for the installed system.

`--hostname`

Hostname for the installed system.

There are three different methods of network configuration:

- DHCP
- BOOTP
- static

The DHCP method uses a DHCP server system to obtain its networking configuration. As you might guess, the BOOTP method is similar, requiring a BOOTP server to supply the networking configuration.

The static method requires that you enter all the required networking information in the kickstart file. As the name implies, this information is static, and will be used during the installation, and after the installation as well.

To direct a system to use DHCP to obtain its networking configuration, use the following line:

```
network --bootproto dhcp
```

To direct a machine to use BOOTP to obtain its networking configuration, use the following line in the kickstart file:

```
network --bootproto bootp
```

The line for static networking is more complex, as you must include all network configuration information on one line. You'll need to specify:

- IP address
- Netmask
- Gateway IP address
- Nameserver IP address

Here's an example static line:

```
network –bootproto static –ip 10.0.2.15 –netmask 255.255.255.0 –gateway 10.0.2.254 –nameserver 10.0.2.1
```

If you use the static method, be aware of the following two restrictions:

- All static networking configuration information must be specified on *one* line; you cannot wrap lines using a backslash, for example.
- You can only specify one nameserver here. However, you can use the kickstart file's `%post` section (described in Section 1.5.31, *`%post` — Post-Installation Configuration Section*) to add more name servers, if needed.

1.5.19 part

`part` or `partition` (required for installs, ignored for upgrades)

Creates a partition on the system.

The <*mntpoint*> is where the partition will be mounted and must be of one of the following forms:

`/<mntpoint>`

For example, **`/`**, **`/usr`**, **`/home`**

swap

The partition will be used as swap space.

raid.<*id*>

The partition will be used for software RAID (see the Section 1.5.20, *`raid`* below).

`--size` ***`<size>`***

The minimum partition size in megabytes. Specify an integer value here such as 500. Do not append the number with MB.

`--grow`

Tells the partition to grow to fill available space (if any), or up to the maximum size setting.

`--maxsize` ***`<size>`***

The maximum partition size in megabytes when the partition is set to grow. Specify an integer value here, and do not append the number with MB.

`--noformat`

Tells the installation program not to format the partition, for use with the `--onpart` command.

`--onpart` ***`<part>`*** **or** **`--usepart`** ***`<part>`***

Tells the installation program to put the partition on the *already existing* device <*part*>. For example, `partition /home --onpart hda1` will put `/home` on `/dev/hda1`, which must already exist.

`--ondisk` ***`<disk>`*** **or** **`--ondrive`** ***`<drive>`***

Forces the partition to be created on a particular disk. For example, `--ondisk sdb` will put the partition on the second disk on the system.

`--onprimary` ***`<N>`***

Forces the partition to be created on the primary partition <*N*> or fail. <*N*> can be 1 through 4. For example, **`--onprimary=1`** specifies that the partition is to be created on the first primary partition.

`--asprimary`

Forces automatic allocation of the partition as a primary partition or the partitioning will fail.

`--bytes-per-inode=`*`<N>`*

<N> represents the number of bytes per inode on the filesystem when it is created. It must be given in decimal format. This option is useful for applications where you want to increase the number of inodes on the filesystem.

`--type=`*`<X>`* (replaced by `fstype`)

This option is no longer available. Use `fstype`.

`--fstype` 1

Sets the filesystem type for the partition. Valid values are ext2, ext3, swap, vfat.

`--start` 1

Specifies the starting cylinder for the partition. It requires that a drive be specified with `--ondisk` or `ondrive`. It also requires that the ending cylinder be specified with `--end` or the partition size be specified with `--size`.

`--end` 1

Specifies the ending cylinder for the partition. It requires that the starting cylinder be specified with `--start`.

`--badblocks` 1

Specifies that the partition should be checked for bad sectors.

All partitions created will be formatted as part of the installation process unless `--noformat` and `--onpart` are used.

Note

If `--clearpart` is used in the `ks.cfg` file, then `--onpart` cannot be used on a logical partition.

Note

If partitioning fails for any reason, diagnostic messages will appear on virtual console 3.

1.5.20 `raid`

`raid` (optional)

Assembles a software RAID device. This command is of the form:

```
raid <mntpoint> --level <level> --device
<mddevice><partitions*>
```

The <*mntpoint*> is the location where the RAID filesystem is mounted. If it is `/`, the RAID level must be 1 unless a boot partition (`/boot`) is present. If a boot partition is present, the `/boot` partition must be level 1 and the root (`/`) partition can be any of the available types. The <*partitions**> (which denotes that multiple partitions can be listed) lists the RAID identifiers to add to the RAID array.

`--level <level>`

RAID level to use (0, 1, or 5).

`--device <mddevice>`

Name of the RAID device to use (such as md0 or md1). RAID devices range from md0 to md7, and each may only be used once.

`--spares=N` [1]

Specifies that there should be N spare drives allocated for the RAID array. Spare drives are used to rebuild the array in case of drive failure.

`--fstype` [1]

Sets the filesystem type for the RAID array. Valid values are ext2, ext3, swap, and vfat.

`--noformat` [1]

Do not format the RAID array.

The following example shows how to create a RAID level 1 partition for `/`, and a RAID level 5 for `/usr`, assuming there are three SCSI disks on the system. It also creates three swap partitions, one on each drive.

```
part raid.01 --size 60 --ondisk sda
part raid.02 --size 60 --ondisk sdb
part raid.03 --size 60 --ondisk sdc

part swap --size 128 --ondisk sda
part swap --size 128 --ondisk sdb
part swap --size 128 --ondisk sdc

part raid.11 --size 1 --grow --ondisk sda
part raid.12 --size 1 --grow --ondisk sdb
part raid.13 --size 1 --grow --ondisk sdc

raid / --level 1 --device md0 raid.01 raid.02 raid.03
raid /usr --level 5 --device md1 raid.11 raid.12 raid.13
```

1.5.21 `reboot`

`reboot` (optional)

Reboot after the installation is complete (no arguments). Normally, kickstart displays a message and waits for the user to press a key before rebooting.

1.5.22 `rootpw`

`rootpw` (required)

```
rootpw [--iscrypted] <password>
```

Sets the system's root password to the <*password*> argument.

`--iscrypted`

If this is present, the password argument is assumed to already be encrypted.

1.5.23 `skipx`

`skipx` (optional)

If present, X is not configured on the installed system.

1.5.24 `text`

`text` (optional) [1]

Perform the kickstart installation in text mode. Kickstart installations are performed in graphical mode by default.

1.5.25 `timezone`

`timezone` (required)

```
timezone [--utc] <timezone>
```

Sets the system time zone to <*timezone*> which may be any of the time zones listed by `timeconfig`.

`--utc`

If present, the system assumes the hardware clock is set to UTC (Greenwich Mean) time.

1.5.26 upgrade

upgrade (optional)

Tells the system to upgrade an existing system rather than install a fresh system.

1.5.27 xconfig

xconfig (optional)

Configures the X Window System. If this option is not given, the user will need to configure X manually during the installation, if X was installed; this option should not be used if X is not installed on the final system.

--noprobe

Do not probe the monitor.

--card *<card>*

Use card <*card*>; this card name should be from the list of cards in Xconfigurator. If this argument is not provided, Anaconda will probe the PCI bus for the card. Since AGP is part of the PCI bus, AGP cards will be detected if supported. The probe order is determined by the PCI scan order of the motherboard.

--videoram *<vram>* [1]

Specify the amount of video RAM the video card has.

--monitor *<mon>*

Use monitor <*mon*>; this monitor name should be from the list of monitors in Xconfigurator. This is ignored if **--hsync** or **--vsync** is provided. If no monitor information is provided, the installation program tries to probe for it automatically.

--hsync *<sync>*

Specifies the horizontal sync frequency of the monitor.

--vsync *<sync>*

Specifies the vertical sync frequency of the monitor.

--defaultdesktop=GNOME or --defaultdesktop=KDE

Sets the default desktop to either GNOME or KDE (and assumes that GNOME and/or KDE has been installed through `%packages`).

--startxonboot

Use a graphical login on the installed system.

--resolution *<res>* [1]

Specify the default resolution for the X Window System on the installed system. Valid values are 640x480, 800x600, 1024x768, 1152x864, 1280x1024, 1400x1050, 1600x1200. Be sure to specify a resolution that is compatible with the video card and monitor.

--depth *<cdepth>* [1]

Specify the default color depth for the X Window System on the installed system. Valid values are 8, 16, 24, and 32. Be sure to specify a color depth that is compatible with the video card and monitor.

1.5.28 `zerombr` — Partition Table Initialization

`zerombr` (optional)

If `zerombr` is specified, and `yes` is its sole argument, any invalid partition tables found on disks are initialized. This will destroy all of the contents of disks with invalid partition tables. This command should be in the following format:

```
zerombr yes
```

No other format is effective.

1.5.29 `%packages` — Package Selection

Use the `%packages` command to begin a kickstart file section that lists the packages you'd like to install (this is for installations only, as package selection during upgrades is not supported).

Packages can be specified by component or by individual package name. The installation program defines several components that group together related packages. See the `RedHat/base/comps` file on any Red Hat Linux CD-ROM for a list of components. The components are defined by the lines that begin with a number followed by a space and then the component name. Each package in that component is then listed, line-by-line. Individual packages lack the leading number found in front of component lines.

Additionally, there are three other types of lines in the `comps` file:

Architecture specific (i386:, ia64:, alpha:, and sparc64:)

If a package name begins with an architecture type, you only need to type in the package name, not the architecture name. For example:

For `i386:   apmd` you only need to use the `apmd` part for that specific package to be installed.

[1] This option is new to Red Hat Linux 7.2.

Lines beginning with ?

Lines that begin with a ? are used by the installation program and should not be altered.

Lines beginning with --hide

If a package name begins with --hide, you only need to type in the package name, without the --hide. For example:

For --hide Network Server you only need to use the Network Server part for that specific package to be installed.

In most cases, it's only necessary to list the desired components and not individual packages. Note that the Base component is always selected by default, so it's not necessary to specify it in the %packages section.

Here's an example %packages selection:

```
%packages
@ Network Managed Workstation
@ Development
@ Web Server
@ X Window System
xgammon
```

As you can see, components are specified, one to a line, starting with an @ symbol, a space, and then the full component name as given in the comps file. Specify individual packages with no additional characters (the xgammon line in the example above is an individual package).

Note

You can also direct the kickstart installation to install the default packages for a workstation (KDE or GNOME) or server installation (or choose an everything installation to install all packages). To do this, simply add *one* of the following lines to the %packages section:

```
@ GNOME
@ KDE
@ Server
@ Everything
```

1.5.30 %pre — Pre-Installation Configuration Section

You can add commands to run on the system immediately after the ks.cfg has been parsed. This section must be at the end of the kickstart file (after the commands) and must start with the %pre command. Note that you can access the network in the %pre section; however, **name service** has not been configured at this point, so only IP addresses will work. Here's an example %pre section:

```
%pre

# add comment to /etc/motd
echo "Kickstart-installed Red Hat Linux ‘/bin/date‘" > /etc/motd

# add another nameserver
echo "nameserver 10.10.0.2" >> /etc/resolv.conf
```

This section creates a message-of-the-day file containing the date the kickstart installation took place. It also gets around the network command's limitation of only one name server by adding another nameserver to /etc/resolv.conf.

Note

Note that the pre-install script is not run in the change root environment.

1.5.31 %post — Post-Installation Configuration Section

You have the option of adding commands to run on the system once the installation is complete. This section must be at the end of the kickstart file and must start with the %post command.

Note

If you configured the network with static IP information, including a nameserver, you can access the network and resolve IP addresses in the %post section. If you configured the network for DHCP, the /etc/resolv.conf file has not been completed when the installation executes the %post section. You can access the network, but you can not resolve IP addresses. Thus, if you are using DHCP, you must specify IP addresses in the %post section.

Here's an example %post section that creates a message of the day file containing the date that the kickstart installation took place, and gets around the network command's limitation of one nameserver only by adding another nameserver to /etc/resolv.conf.

```
%post

# add comment to /etc/motd
echo "Kickstart-installed Red Hat Linux ‘/bin/date‘" > /etc/motd
```

```
# add another nameserver
echo "nameserver 10.10.0.2" >> /etc/resolv.conf
```

Note

The post-install script is run in a chroot environment; therefore, performing tasks such as copying scripts or RPMs from the installation media will not work.

--nochroot

Allows you to specify commands that you would like to run outside of the chroot environment.

The following example copies the file /etc/resolv.conf to the filesystem that was just installed.

```
%post --nochroot
cp /etc/resolv.conf /mnt/sysimage/etc/resolv.conf
```

--interpreter */usr/bin/perl*

Allows you to specify a different scripting language, such as Perl. Replace */usr/bin/perl* with the scripting language of your choice.

The following example uses a Perl script to replace /etc/HOSTNAME.

```
%post --interpreter /usr/bin/perl

# replace /etc/HOSTNAME
open(HN, ">HOSTNAME");
print HN "1.2.3.4 an.ip.address\n";
```

More examples of post-installation scripts can be found in Section 2.11, *Post-Installation Script*.

2 Kickstart Configurator

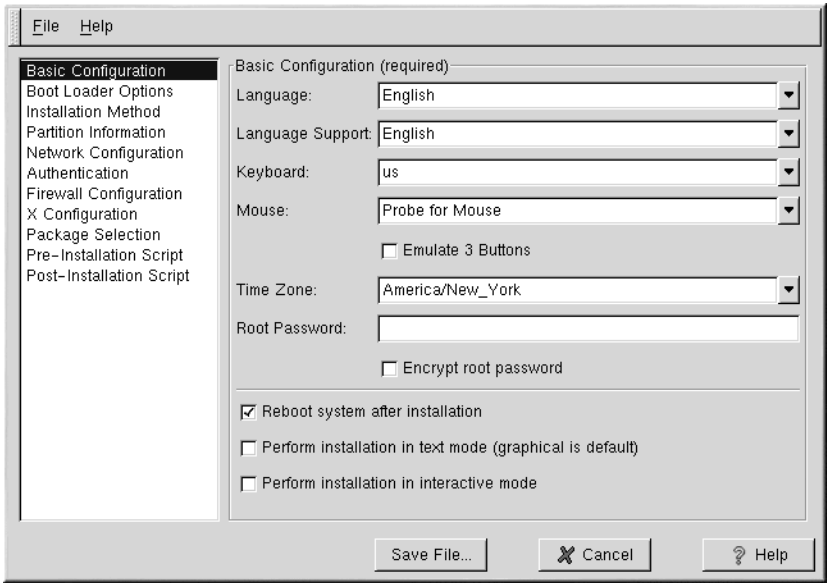
Kickstart Configurator allows you to create a kickstart file using a graphical user interface, so that you do not have to remember the correct syntax of the file. After choosing the kickstart options, click the **Save File** button, verify the options you have chosen, and save the kickstart file to a desired location.

To use Kickstart Configurator, you must by running the X Window System. To start Kickstart Configurator, use one of the following methods:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Kickstart Configurator**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Kickstart Configurator**.
- Type the command `ksconfig` at a shell prompt (for example, in an XTerm or GNOME terminal).

2.1 Basic Configuration

Figure 2–1 Basic Configuration



Basic Configuration (required)

Language: English

Language Support: English

Keyboard: us

Mouse: Probe for Mouse

□ Emulate 3 Buttons

Time Zone: America/New_York

Root Password:

□ Encrypt root password

☑ Reboot system after installation

□ Perform installation in text mode (graphical is default)

□ Perform installation in interactive mode

Choose the language to use during the installation from the **Language** menu. Choose the language to use after installation from the **Language Support** menu. Select the system keyboard type from the **Keyboard** menu.

Choose the mouse for the system from the **Mouse** menu. If you choose **No Mouse**, no mouse will be configured. If you choose **Probe for Mouse** the installation program will try to autodetect the mouse. Probing works for most modern mice.

If you have a two-button button mouse, you can emulate a three-button mouse by selecting **Emulate 3 Buttons**. If this option is selected, simultaneously clicking the left and right mouse buttons will be recognized as a middle mouse button click.

From the **Time Zone** menu, choose the time zone to use for the system.

Enter the desired root password for the system in the **Root Password** text entry box. If you want to save the password as an encrypted password in the file, select **Encrypt root password**. When the file is saved, the plaintext password that you typed will be encrypted and written to the kickstart file. Do not type an already encrypted password and select to encrypt it.

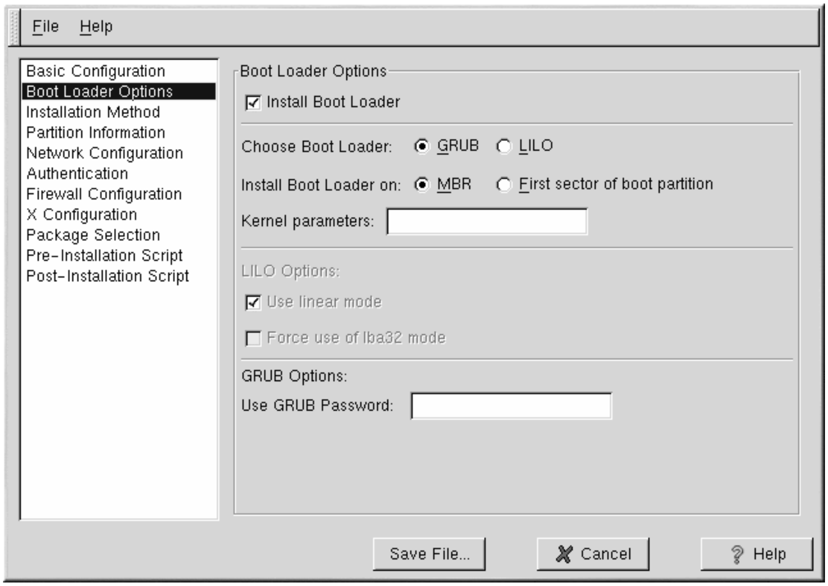
Choosing **Reboot system after installation** will reboot your system automatically after the installation is finished.

Kickstart installations are performed in graphical mode by default. To override this default and use text mode instead, check the **Perform installation in text mode** button.

You can perform a kickstart installation in interactive mode. This means that the installation program will use all the options pre-configured in the kickstart file, but it will allow you to preview the options in each screen before you can continue to the next screen. To continue to the next screen, click the **Next** button after you have approved the settings. If you are not satisfied with the pre-configured options, you can change them before continuing the installation. If you prefer this type of installation, check the **Perform installation in interactive mode** button.

2.2 Boot Loader Options

Figure 2–2 Boot Loader Options



Boot Loader Options

☑ Install Boot Loader

Choose Boot Loader: ● GRUB ○ LILO

Install Boot Loader on: ● MBR ○ First sector of boot partition

Kernel parameters:

LILO Options:

☑ Use linear mode

□ Force use of lba32 mode

GRUB Options:

Use GRUB Password:

You have the option of installing GRUB or LILO as the boot loader. If you do not want to install a boot loader, uncheck the **Install Boot Loader** checkbutton. If you choose not to install a boot loader, make sure you create a boot disk or have another way to boot (such as a third-party boot loader) your Red Hat Linux system.

If you choose to install a boot loader, you must also choose which boot loader to install (GRUB or LILO) and where to to install the boot loader (the Master Boot Record or the first sector of the `/boot` partition). Install the boot loader on the MBR if you plan to use it as your boot loader. If you are using a different boot loader, install LILO or GRUB on the first sector of the `/boot` partition and configure the other boot loader to boot Red Hat Linux.

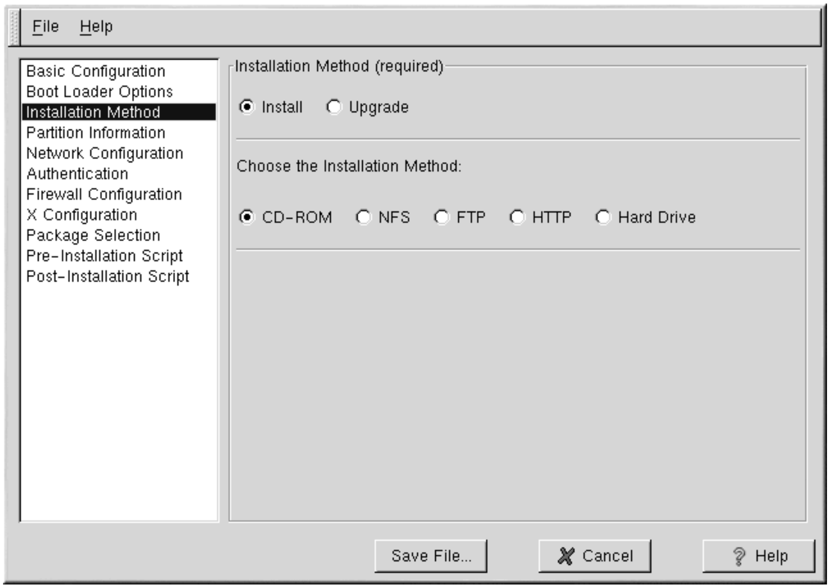
If you need to pass any special parameters to the kernel to be used when the system boots, enter them in the **Kernel parameters** text field. For example, if you have an IDE CD-ROM burner, you can tell the kernel to use the SCSI emulation driver that must be loaded before using cdrecord by typing **`hdd=ide-scsi`** as a kernel parameter (where **hdd** is the CD-ROM device).

If you choose LILO as the boot loader, choose whether you want to use linear mode and whether you want to force the use of lba32 mode.

If you choose GRUB as the boot loader, you can password protect it by configuring a GRUB password. Enter a password in the **Use GRUB Password** text entry area.

2.3 Installation Method

Figure 2–3 Installation Method



Installation Method (required)

● Install ○ Upgrade

Choose the Installation Method:

● CD-ROM ○ NFS ○ FTP ○ HTTP ○ Hard Drive

The **Installation Method** page allows you to choose whether you want to perform a full installation or an upgrade. If you choose upgrade, the **Partition Information** and **Package Selection** pages will be disabled. They are not supported for kickstart upgrades.

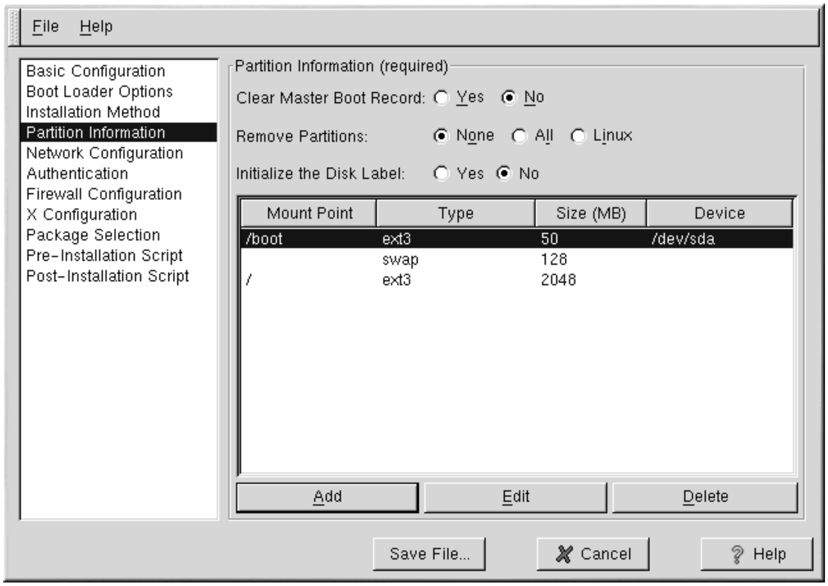
Also choose the type of kickstart installation to perform from this page. You can choose from the following options:

- **CD-ROM** — Choose this option if you wish to install Red Hat Linux from the Red Hat Linux CD-ROMs.
- **NFS** — Choose this option if you wish to install Red Hat Linux from an NFS shared directory. Two text entry boxes for the NFS server and NFS directory will appear. Enter the fully-qualified domain name or IP address of the NFS server. For the NFS directory, enter the name of the NFS directory that contains the `RedHat` directory. For example, if your NFS server contains the directory `/mirrors/redhat/i386/RedHat`, enter `/mirrors/redhat/i386` for the NFS directory.

- **FTP** — Choose this option if you wish to install Red Hat Linux from an FTP server. Two text entry boxes for the FTP server and FTP directory will appear. Enter the fully-qualified domain name or IP address of the FTP server. For the FTP directory, enter the name of the FTP directory that contains the `RedHat` directory. For example, if your FTP server contains the directory `/mirrors/redhat/i386/RedHat`, enter `/mirrors/redhat/i386` for the FTP directory.
- **HTTP** — Choose this option if you wish to install Red Hat Linux from an HTTP server. Two text entry boxes for the HTTP server and HTTP directory will appear. Enter the fully-qualified domain name or IP address of the HTTP server. For the HTTP directory, enter the name of the HTTP directory that contains the `RedHat` directory. For example, if your HTTP server contains the directory `/mirrors/redhat/i386/RedHat`, enter `/mirrors/redhat/i386` for the HTTP directory.
- **Hard Drive** — Choose this option if you wish to install Red Hat Linux from a hard drive. Two text entry boxes for hard drive partition and hard drive directory will appear. Hard drive installations require the use of ISO (or CD-ROM) images. Be sure to verify that the ISO images are intact before you start the installation. To verify them, use an `md5sum` program. Enter the hard drive partition that contains the ISO images (for example, `/dev/hda1`) in the **Hard Drive Partition** text box, and enter the directory that contains the ISO images in the **Hard Drive Directory** text box.

2.4 Partition Information

Figure 2–4 Partition Information



Partition Information (required)

Clear Master Boot Record: ○ Yes ● No

Remove Partitions: ● None ○ All ○ Linux

Initialize the Disk Label: ○ Yes ● No

Mount Point	Type	Size (MB)	Device
/boot	ext3	50	/dev/sda
	swap	128	
/	ext3	2048	

To clear the Master Boot Record, select **Yes** beside the option on the top of the page. You can choose to keep the existing partitions, remove all the existing partitions, or remove all the existing Linux partitions by selecting **None**, **All**, or **Linux**, respectively, next to **Remove Partitions**.

You can initialize the disk label to the default for the architecture of the system (`msdos` for x86 and `gpt` for Itanium). Choose **Yes** if you are installing on a brand new hard drive.

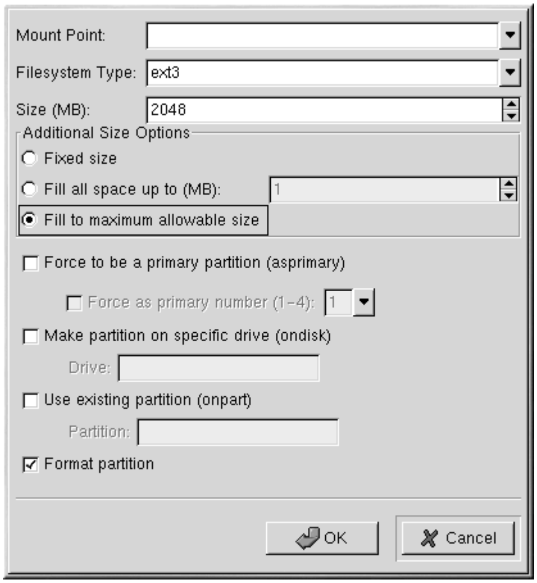
2.4.1 Creating Partitions

To create a partition, click the **Add** button. The **Partition Options** window shown in Section 2.4.1, *Creating Partitions* will appear. Choose mount point, filesystem type, and partition size for the new partition. Optionally, you can also choose from the following:

- Additional Size Options — Choose to make the partition a fixed size, up to a chosen size, or fill the remaining space on the hard drive.
- Force the partition to be created as a primary partition.
- Create the partition on a specific hard drive.

- Use an existing partition.
- Format the partition as the chosen filesystem type.

Figure 2–5 Creating Partitions



Mount Point:

Filesystem Type: ext3

Size (MB): 2048

Additional Size Options

○ Fixed size

○ Fill all space up to (MB): 1

● Fill to maximum allowable size

□ Force to be a primary partition (asprimary)

□ Force as primary number (1–4): 1

□ Make partition on specific drive (ondisk)

Drive:

□ Use existing partition (onpart)

Partition:

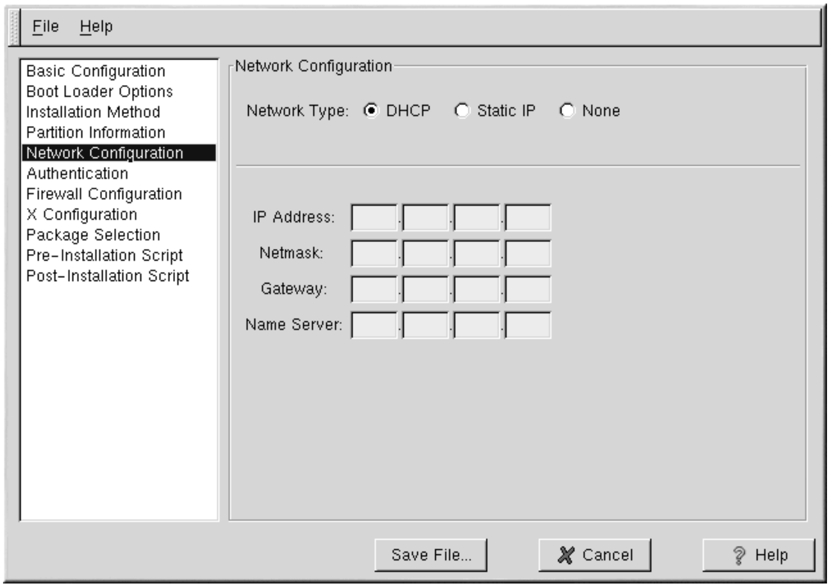
☑ Format partition

To edit an existing partition, select the partition from the list and click the **Edit** button. The same **Partitions Options** window that appears when you add a partition appears, except it contains the values for the selected partition. Modify the partition options and click **OK**.

To delete an existing partition, select the partition from the list and click the **Delete** button.

2.5 Network Configuration

Figure 2–6 Network Configuration



Network Type: ● DHCP ○ Static IP ○ None

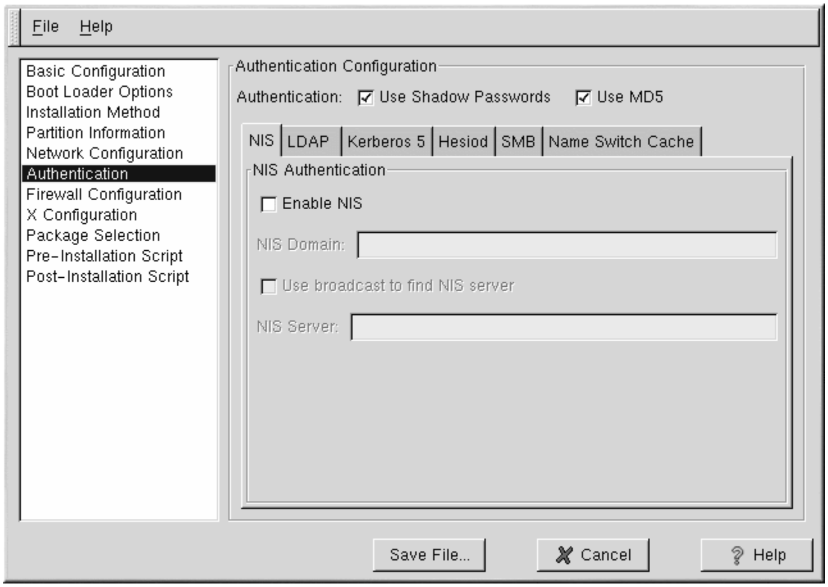
There are three network configuration options: **DHCP**, **Static IP**, and **None**. If there is not an ethernet card in the system, choose **None**.

Networking is only required if you choose a networking-type installation method (NFS or FTP). If you are unsure which to choose, choose **None**. Networking can always be configured after installation with Network Configurator (`redhat-config-network`).

If you select **Static IP**, you must provide additional networking information in the table below the network types.

2.6 Authentication

Figure 2–7 Authentication



Authentication: ☑ Use Shadow Passwords ☑ Use MD5

□ Enable NIS

NIS Domain:

□ Use broadcast to find NIS server

NIS Server:

In the **Authentication** section, select whether to use shadow passwords and md5 encryption for user passwords. These options are highly recommended and chosen by default.

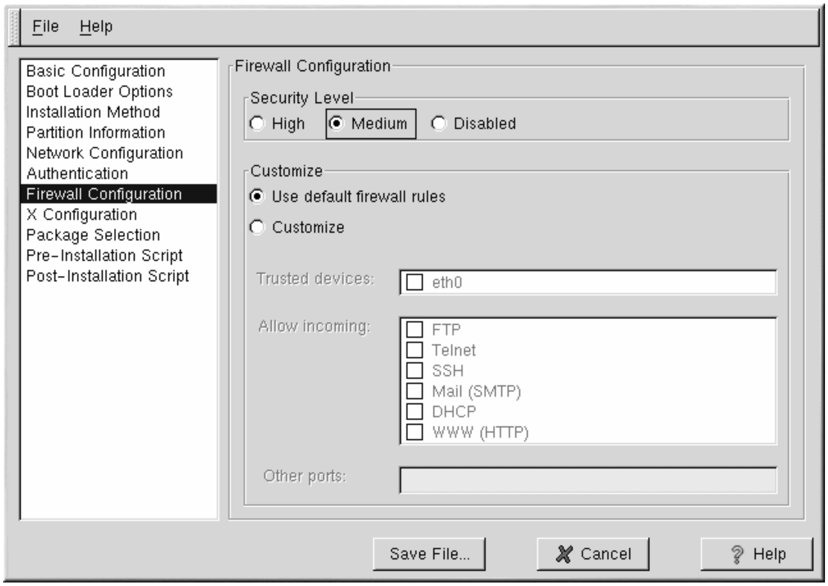
The **Authentication Configuration** page allows you to configure the following methods of authentication:

- NIS
- LDAP
- Kerberos 5
- Hesiod
- SMB
- Name Switch Cache

They are not enabled by default. To enable one or more of these methods, click the appropriate tab, click the checkbutton next to **Enable**, and enter the appropriate information for the authentication method.

2.7 Firewall Configuration

Figure 2–8 Firewall Configuration



Firewall Configuration

Security Level

○ High ● Medium ○ Disabled

Customize

● Use default firewall rules

○ Customize

Trusted devices: □ eth0

Allow incoming:

□ FTP
□ Telnet
□ SSH
□ Mail (SMTP)
□ DHCP
□ WWW (HTTP)

Other ports:

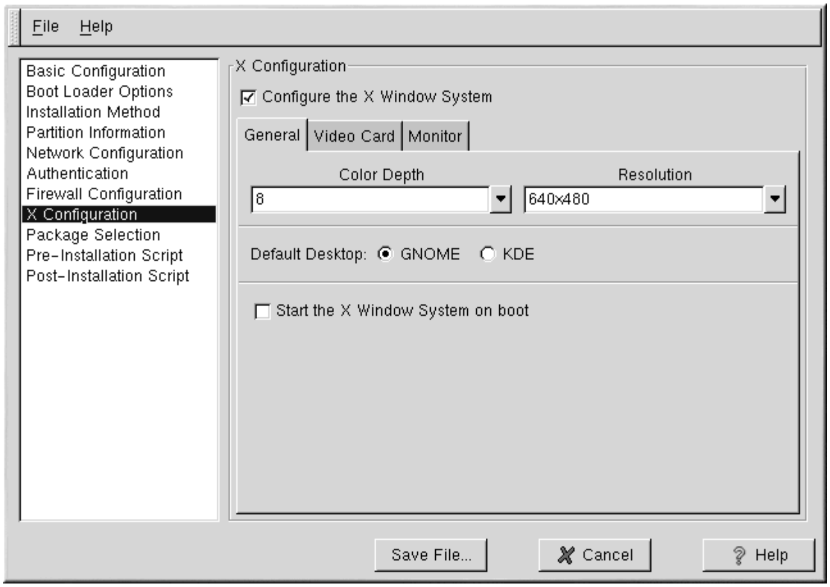
The **Firewall Configuration** page is identical to the screen in the Red Hat Linux installation program and provides the same functionality. Choose between **High**, **Medium**, and **Disabled** security levels. Refer to the *Official Red Hat Linux Installation Guide* for detailed information about these security levels.

2.8 X Configuration

If you are installing the X Window System, you can configure it during the kickstart installation by checking the **Configure the X Window System** button on the **X Configuration** page as shown in Figure 2–9, *X Configuration - General*. If this option is not chosen, the X configuration options will be disabled and the `skipx` option will be written to the kickstart file.

2.8.1 General

Figure 2–9 X Configuration - General



☑ Configure the X Window System

Color Depth: 8

Resolution: 640x480

Default Desktop: ● GNOME ○ KDE

□ Start the X Window System on boot

The first step in configuring X is to choose the default color depth and resolution. Select them from their respective pulldown menus. Be sure to specify a color depth and resolution that is compatible with the video card and monitor for the system.

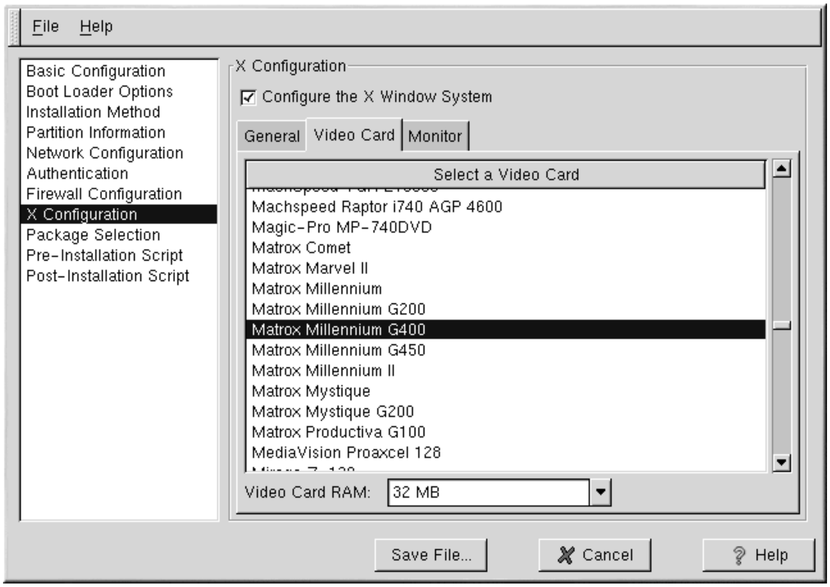
If you are installing both the GNOME and KDE desktops, you need to choose which desktop you want to be the default. If you are just installing one desktop, be sure to choose it. Once the system is installed, users can choose which desktop they want to be their default. For more information about GNOME and KDE, refer to the *Official Red Hat Linux Installation Guide* and the *Official Red Hat Linux Getting Started Guide*.

Next, choose whether to start the X Window System when the system is booted. This option will start the system in runlevel 5 with the graphical login screen. After the system is installed, this can be changed by modifying the `/etc/inittab` configuration file.

2.8.2 Video Card

Select the video card from the list on the **Video Card** tab as shown in Figure 2–10, *X Configuration - Video Card*. Also select the amount of video RAM the selected video card has from the **Video Card RAM** pulldown menu.

Figure 2–10 X Configuration - Video Card



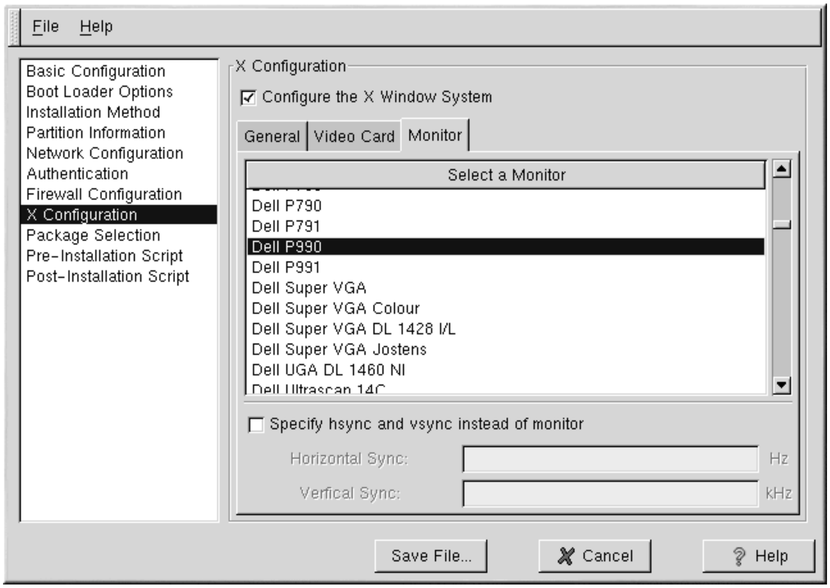
☑ Configure the X Window System

Video Card RAM: 32 MB

2.8.3 Monitor

After configuring the video card, click on the **Monitor** tab shown in Figure 2–11, *X Configuration - Monitor* and select the monitor for the system. You can specify the horizontal and vertical sync rates instead of specifying a monitor by checking the **Specify hysnc and vsync instead of monitor** option. This option is useful if the monitor for the system is not listed. Notice that when this option is enabled, the monitor list is disabled.

Figure 2–11 X Configuration - Monitor



X Configuration

☑ Configure the X Window System

Select a Monitor

Dell P790
Dell P791
Dell P990
Dell P991
Dell Super VGA
Dell Super VGA Colour
Dell Super VGA DL 1428 I/L
Dell Super VGA Jostens
Dell UGA DL 1460 NI
Dell Ultrascan 14C

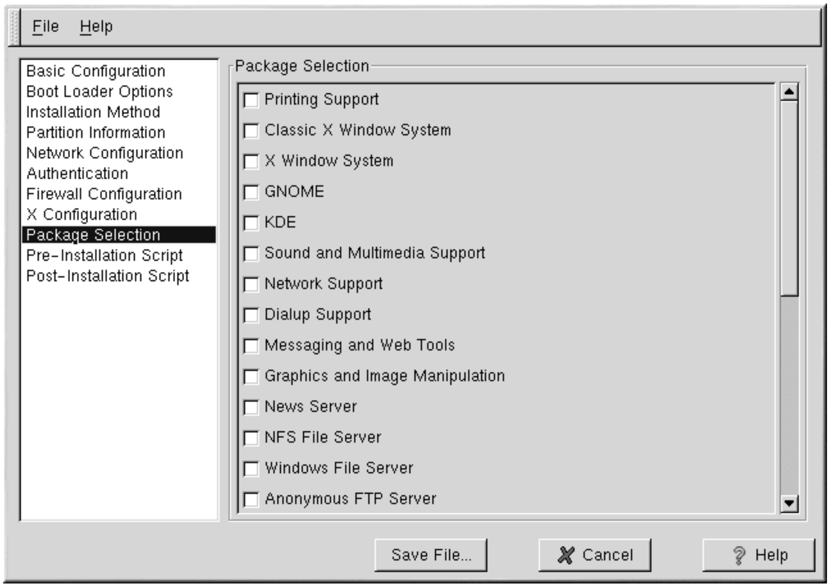
☐ Specify hsync and vsync instead of monitor

Horizontal Sync: Hz

Vertical Sync: kHz

2.9 Package Selection

Figure 2–12 Package Selection

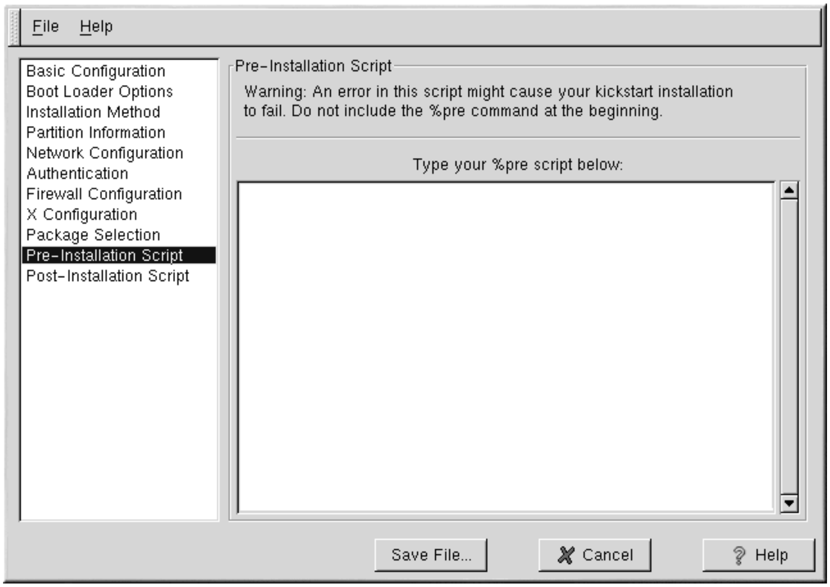


☐ Printing Support
☐ Classic X Window System
☐ X Window System
☐ GNOME
☐ KDE
☐ Sound and Multimedia Support
☐ Network Support
☐ Dialup Support
☐ Messaging and Web Tools
☐ Graphics and Image Manipulation
☐ News Server
☐ NFS File Server
☐ Windows File Server
☐ Anonymous FTP Server

The **Package Selection** page allows you to choose which package categories to install. Currently, Kickstart Configurator does not allow you to select individual packages. To install individual packages, modify the `%packages` section of the kickstart file after you save it.

2.10 Pre-Installation Script

Figure 2–13 Pre-Installation Script



Warning: An error in this script might cause your kickstart installation to fail. Do not include the %pre command at the beginning.

Type your %pre script below:

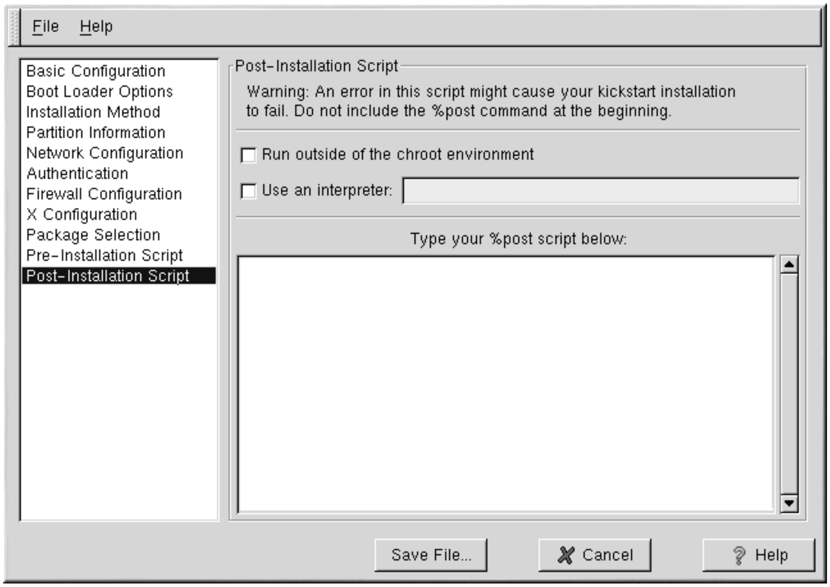
You can add commands to run on the system immediately after the kickstart file has been parsed and before the installation begins. If you have configured the network in the kickstart file, the network is enabled before this section is processed. If you would like to include a pre-installation script, type it in the text area.

CAUTION

Do not include the `%pre` command. It will be added for you.

2.11 Post-Installation Script

Figure 2–14 Post-Installation Script



Post-Installation Script

Warning: An error in this script might cause your kickstart installation to fail. Do not include the %post command at the beginning.

□ Run outside of the chroot environment

□ Use an interpreter:

Type your %post script below:

You can also add commands to execute on the system after the installation is completed. If you have properly configured the network in the kickstart file, the network is enabled. If you would like to include a post-installation script, type it in the text area.

CAUTION

Do not include the `%post` command. It will be added for you.

For example, to change the message of the day for the newly installed system, add the following command to the `%post` section:

```
echo "Hackers will be punished!" > /etc/motd
```

2.11.1 Chroot Environment

If you want your post-installation script to run outside of the chroot environment, click the checkbutton next to this option on the top of the **Post-Installation** page. This is equivalent to the using the `--nochroot` option in the `%post` section.

Tip

If you want to make any changes to the newly installed filesystem in the post-installation section outside of the chroot environment, you need to append the directory name with `/mnt/sysimage`.

For example, if you check the **Run outside of the chroot environment** button, the previous example needs to be changed to the following:

```
echo "Hackers will be punished!" > /mnt/sysimage/etc/motd
```

2.11.2 Use an Interpreter

If you want to specify a scripting language to use to execute your script, click the **Use an interpreter** button and enter the interpreter in the text box beside the button. For example, **`/usr/bin/perl`** can be specified for a Perl script. This option corresponds to using `%post --interpreter` *`/usr/bin/perl`* in your kickstart file.

2.11.3 Examples

The post-installation script can be used to perform any useful functions such as the following examples.

Turn services on and off:

```
/sbin/chkconfig --level 345 telnet off
/sbin/chkconfig --level 345 finger off
/sbin/chkconfig --level 345 lpd off
/sbin/chkconfig --level 345 httpd on
```

Run a script named `runme` from an NFS share:

```
mkdir /mnt/temp
mount 10.10.0.2:/usr/new-machines /mnt/temp
open -s -w -- /mnt/temp/runme
umount /mnt/temp
```

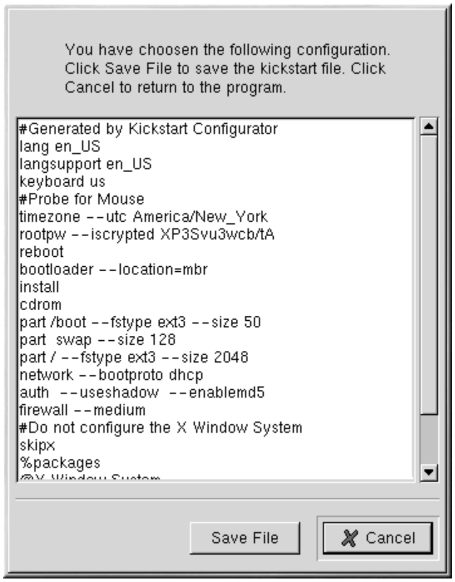
Add a user to the system:

```
/usr/sbin/useradd bob
/usr/bin/chfn -f "Bob Smith" bob
/usr/sbin/usermod -p 'kjdf$04930FTH/ ' bob
```

2.12 Saving the File

After you have finished choosing your kickstart options, click the **Save File** button. A dialog box similar to Figure 2–15, *Confirm Options* will appear to allow you to review your choices before saving the file.

Figure 2–15 Confirm Options



You have choosen the following configuration.
Click Save File to save the kickstart file. Click
Cancel to return to the program.

```
#Generated by Kickstart Configurator
lang en_US
langsupport en_US
keyboard us
#Probe for Mouse
timezone --utc America/New_York
rootpw --iscrypted XP3Svu3wcb/tA
reboot
bootloader --location=mbr
install
cdrom
part /boot --fstype ext3 --size 50
part  swap --size 128
part / --fstype ext3 --size 2048
network --bootproto dhcp
auth  --useshadow  --enablemd5
firewall --medium
#Do not configure the X Window System
skipx
%packages
@X Window System
```

If you are happy with your choices, click the **Save File** button within the dialog box. A save file dialog box will appear and allow you to choose where to save the file. The default file name to save it as is `ks.cfg`.

After saving the file, refer to Section 1.3, *Starting a Kickstart Installation* for information on how to start the kickstart installation.

3 Rescue Mode

When things go wrong, there are ways to fix problems. However, these methods require that you understand the system well. This chapter will describe the ways that you can boot into rescue mode and single user mode, where you can use your own knowledge to repair the system.

3.1 What is Rescue Mode?

Rescue mode provides the ability to boot a small Linux environment entirely from a diskette, CD-ROM, or using some other method.

As the name implies, rescue mode is provided to rescue you from something. During normal operation, your Red Hat Linux system uses files located on your system's hard drive to do everything — run programs, store your files, and more.

However, there may be times when you are unable to get Linux running completely enough to access its files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Linux from that hard drive.

Normally, you will need to get into rescue mode for one of two reasons:

- You are unable to boot Linux.
- You are having hardware or software problems, and you want to get a few important files off your system's hard drive.

Next, we will take a closer look at each of these scenarios.

3.1.1 Unable to Boot Linux

This problem is often caused by the installation of another operating system after you have installed Red Hat Linux. Some other operating systems assume that you have no other operating systems on your computer, and they overwrite the Master Boot Record (MBR) that originally contained the GRUB or LILO boot loader. If the boot loader is overwritten in this manner, you will not be able to boot Red Hat Linux unless you can get into rescue mode.

Another common problem is if you use a partitioning tool to resize a partition or create a new partition from free space after installation and it changes the order of your partitions. If the partition number of your `/` partition changes, the boot loader will not be able to find it to mount the partition. To fix this problem, boot in rescue mode and modify `/boot/grub/grub.conf` if you are using GRUB or `/etc/lilo.conf` if you are using LILO.

3.1.2 Hardware/Software Problems

This category includes a wide variety of different situations. Two examples include failing hard drives and forgetting to run LILO after building a new kernel (if you are using LILO as your boot loader). In both of these situations, you may be unable to boot Red Hat Linux. If you can get into rescue mode, you might be able to resolve the problem or at least get copies of your most important files.

3.1.3 Booting Rescue Mode

To boot your system in rescue mode, boot off of a Red Hat Linux boot disk or Red Hat Linux CD-ROM #1, and enter the following command at the installation boot prompt:

```
boot: linux rescue
```

You can get to the installation boot prompt in one of these ways:

- By booting your system from an installation boot diskette made from the `boot.img` image. This method requires that the Red Hat Linux CD-ROM #1 be inserted as the rescue image or that the rescue image be on the hard drive as an ISO image. [1]
- By booting your system from the Red Hat Linux CD-ROM #1.
- By booting from a network disk made from the `bootnet.img` or PCMCIA boot disk made from `pcmcia.img`. You can only do this if your network connection is working. You will need to identify the network host and transfer type. For an explanation of how to specify this information, see *Installing over the Network* in the *Official Red Hat Linux Installation Guide*.

After booting off a boot disk or Red Hat Linux CD-ROM #1 and providing a valid rescue image, you will see the following message:

```
The rescue environment will now attempt to find your Red Hat
Linux installation and mount it under the directory
/mnt/sysimage.  You can then make any changes required to your
system.  If you want to proceed with this step choose
’Continue’.
If for some reason this process fails you can choose ’Skip’
and this step will be skipped and you will go directly to a
command shell.
```

If you select **Continue**, it will attempt to mount your filesystem under the directory `/mnt/sysimage`. If it fails to mount a partition, it will notify you. If you select **Skip**, your filesystem will not be mounted. Choose **Skip** if you think your filesystem is corrupted.

[1] To create an installation boot diskette, insert a blank floppy disk and use the `images/boot.img` file on the Red Hat Linux CD-ROM #1 with the command `dd if=boot.img of=/dev/fd0`.

Once you have your system in rescue mode, a prompt appears on VC (virtual console) 1 and VC 2 (use the [Ctrl]-[Alt]-[F1] key combination to access VC 1 and [Ctrl]-[Alt]-[F2] key combination to access VC 2):

```
bash#
```

If you selected **Continue** to mount your partitions automatically and they were mounted successfully, you are in single-user mode.

To mount a Linux partition manually inside rescue mode, create a directory such as */foo*, and type the following command:

```
mount -t ext3 /dev/hda5 /foo
```

In the above command, */foo* is a directory that you have created and */dev/hda5* is the partition you want to mount. If the partition is of type `ext2`, replace `ext3` with `ext2`.

If you do not know the names of your partitions, use the following command to list them:

```
fdisk -l
```

If your filesystem is mounted and you want to make your system the root partition, use the command `chroot /mnt/sysimage`. This is useful if you need to run commands such as `rpm` that require your root partition to be mounted as `/`. To exit the chroot environment, type `exit`, and you will return to the `bash#` prompt.

From the `bash#` prompt, you can run many useful commands including:

```
anaconda      gzip        mkfs.ext2   ps
badblocks     head        mknod       python
bash          hwclock     mkraid      python1.5
cat           ifconfig    mkswap      raidstart
chattr        init        mlabel      raidstop
chmod         insmod      mmd         rcp
chroot        less        mmount      rlogin
clock         ln          mmove       rm
collage       loader      modprobe    rmmod
cp            ls          mount       route
cpio          lsattr      mpartition  rpm
dd            lsmod       mrd         rsh
ddcprobe      mattrib     mread       sed
depmode       mbadblocks  mren        sh
df            mcd         mshowfat    sync
e2fsck        mcopy       mt          tac
fdisk         mdel        mtools      tail
fsck          mdeltree    mtype       tar
fsck.ext2     mdir        mv          touch
fsck.ext3     mdu         mzip        traceroute
```

```
ftp                 mformat     open        umount
gnome-pty-helper    minfo       pico        uncpio
grep                mkdir       ping        uniq
gunzip              mke2fs      probe       zcat
```

3.1.4 Booting Single-User Mode Directly

You may be able to boot single-user mode directly. If your system boots, but does not allow you to log in when it has completed booting, try single-user mode.

If you are using GRUB, use the following steps to boot into single-user mode:

1. If you have a GRUB password configured, type `p` and enter the password.
2. Select **Red Hat Linux** with the version of the kernel that you wish to boot and type `e` for edit. You will be presented with a list of items in the configuration file for the title you just selected.
3. Select the line that starts with `kernel` and type `e` to edit the line.
4. Go to the end of the line and type **`single`** as a separate word (press the [Spacebar] and then type **`single`**). Press [Enter] to exit edit mode.
5. Back at the GRUB screen, type `b` to boot into single user mode.

If you are using LILO, specify one of these options at the LILO boot prompt (if you are using the graphical LILO, you must press [Ctrl]-[x] to exit the graphical screen and go to the `boot:` prompt):

```
boot: linux single
boot: linux emergency
```

In single-user mode, you computer boots to runlevel 1. Your local filesystems will be mounted, but your network will not be activated. You will have a usable system maintenance shell.

In emergency mode, you are booted into the most minimal environment possible. The root filesystem will be mounted read-only and almost nothing will be set up. The main advantage of emergency mode over `linux single` is that your init files are not loaded. If init is corrupted or not working, you can still mount filesystems to recover data that could be lost during a re-installation.

Have you ever rebuilt a kernel and, eager to try out your new handiwork, rebooted before running `/sbin/lilo`? If you did not have an entry for an older kernel in `lilo.conf`, you had a problem. If you would like to know a solution to this problem, read this section.

In many cases, you can boot your Red Hat Linux system from the Red Hat Linux boot disk [1] with your root filesystem mounted and ready to go. Here is how to do it:

Enter the following command at the boot disk's `boot:` prompt:

```
linux single root=/dev/hdXX initrd=
```

Replace the `XX` in `/dev/hdXX` with the appropriate letter and number for your root partition.

What does this command do? First, it starts the boot process in single-user mode, with the root partition set to your root partition. The empty `initrd` specification bypasses the installation-related image on the boot disk, which will cause you to enter single-user mode immediately.

Is there a negative side to using this technique? Unfortunately, yes. Because the kernel on the Red Hat Linux boot disk only has support for IDE built-in, if your system is SCSI-based, you will not be able to do this. In that case, you will have to access rescue mode using the **`linux rescue`** command mentioned above.

4 Redundant Array of Independent Disks (RAID)

4.1 What is RAID?

The basic idea behind RAID is to combine multiple small, inexpensive disk drives into an array to accomplish performance or redundancy goals not attainable with one large and expensive drive. This array of drives will appear to the computer as a single logical storage unit or drive.

RAID is a method in which information is spread across several disks, using techniques such as **disk striping** (RAID Level 0), **disk mirroring** (RAID level 1), and **disk striping with parity** (RAID Level 5) to achieve redundancy, lower latency and/or increase bandwidth for reading or writing to disks, and maximize the ability to recover from hard disk crashes.

The underlying concept of RAID is that data may be distributed across each drive in the array in a consistent manner. To do this, the data must first be broken into consistently-sized "chunks" (often 32K or 64K in size, although different sizes can be used). Each chunk is then written to a hard drive in RAID according to the RAID level used. When the data is to be read, the process is reversed, giving the illusion that multiple drives are actually one large drive.

4.2 Who Should Use RAID?

Anyone who needs to keep large quantities of data on hand (such as an average system administrator) would benefit by using RAID technology. Primary reasons to use RAID include:

- Enhanced speed
- Increased storage capacity using a single virtual disk
- Lessening the impact of a disk failure

4.3 Hardware RAID versus Software RAID

There are two possible RAID approaches: Hardware RAID and Software RAID.

4.3.1 Hardware RAID

The hardware-based system manages the RAID subsystem independently from the host and presents to the host only a single disk per RAID array.

An example of a Hardware RAID device would be one that connects to a SCSI controller and presents the RAID arrays as a single SCSI drive. An external RAID system moves all RAID handling "intelligence" into a controller located in the external disk subsystem. The whole subsystem is connected to the host via a normal SCSI controller and appears to the host as a single disk.

RAID controllers also come in the form of cards that *act* like a SCSI controller to the operating system but handle all of the actual drive communications themselves. In these cases, you plug the drives into the RAID controller just like you would a SCSI controller, but then you add them to the RAID controller's configuration, and the operating system never knows the difference.

4.3.2 Software RAID

Software RAID implements the various RAID levels in the kernel disk (block device) code. It offers the cheapest possible solution, as expensive disk controller cards or hot-swap chassis [1]are not required. Software RAID also works with cheaper IDE disks as well as SCSI disks. With today's fast CPUs, Software RAID performance can excel against Hardware RAID.

The MD driver in the Linux kernel is an example of a RAID solution that is completely hardware independent. The performance of a software-based array is dependent on the server CPU performance and load.

For information on configuring Software RAID in the Red Hat Linux installation program, refer to the Chapter 5, *Software RAID Configuration*.

For those interested in learning more about what Software RAID has to offer, here is a brief list of the most important features:

- Threaded rebuild process
- Fully kernel-based configuration
- Portability of arrays between Linux machines without reconstruction
- Backgrounded array reconstruction using idle system resources
- Hot-swappable drive support
- Automatic CPU detection to take advantage of certain CPU optimizations

4.4 RAID Levels and Linear Support

RAID supports various configurations, including levels 0, 1, 4, 5, and linear. These RAID types are defined as follows:

[1] A hot-swap chassis allows you to remove a hard drive without having to power-down your system.

- *Level 0* — RAID level 0, often called "striping," is a performance-oriented striped data mapping technique. This means the data being written to the array is broken down into strips and written across the member disks of the array, allowing high I/O performance at low inherent cost but provides no redundancy. The storage capacity of a level 0 array is equal to the total capacity of the member disks in a Hardware RAID or the total capacity of member partitions in a Software RAID.

- *Level 1* — RAID level 1, or "mirroring," has been used longer than any other form of RAID. Level 1 provides redundancy by writing identical data to each member disk of the array, leaving a "mirrored" copy on each disk. Mirroring remains popular due to its simplicity and high level of data availability. Level 1 operates with two or more disks that may use parallel access for high data-transfer rates when reading but more commonly operate independently to provide high I/O transaction rates. Level 1 provides very good data reliability and improves performance for read-intensive applications but at a relatively high cost [2]The storage capacity of the level 1 array is equal to the capacity of one of the mirrored hard disks in a Hardware RAID or one of the mirrored partitions in a Software RAID.

- *Level 4* — Level 4 uses parity [3]concentrated on a single disk drive to protect data. It's better suited to transaction I/O rather than large file transfers. Because the dedicated parity disk represents an inherent bottleneck, level 4 is seldom used without accompanying technologies such as write-back caching. Although RAID level 4 is an option in some RAID partitioning schemes, it is not an option allowed in Red Hat Linux RAID installations [4]The storage capacity of Hardware RAID level 4 is equal to the capacity of member disks, minus the capacity of one member disk. The storage capacity of Software RAID level 4 is equal to the capacity of the member partitions, minus the size of one of the partitions if they are of equal size.

- *Level 5* — This is the most common type of RAID. By distributing parity across some or all of an array's member disk drives, RAID level 5 eliminates the write bottleneck inherent in level 4. The only performance bottleneck is the parity calculation process. With modern CPUs and Software RAID, that usually isn't a very big problem. As with level 4, the result is asymmetrical performance, with reads substantially outperforming writes. Level 5 is often used with write-back caching to reduce the asymmetry. The storage capacity of Hardware RAID level 5 is equal to

[2] RAID level 1 comes at a high cost because you write the same information to all of the disks in the array, which wastes drive space. For example, if you have RAID level 1 set up so that your root (`/`) partition exists on two 40G drives, you have 80G total but are only able to access 40G of that 80G. The other 40G acts like a mirror of the first 40G.

[3] Parity information is calculated based on the contents of the rest of the member disks in the array. This information can then be used to reconstruct data when one disk in the array fails. The reconstructed data can then be used to satisfy I/O requests to the failed disk before it is replaced and to repopulate the failed disk after it has been replaced.

[4] RAID level 4 takes up the same amount of space as RAID level 5, but level 5 has more advantages than level 4. For this reason, level 4 is not supported.

the capacity of member disks, minus the capacity of one member disk. The storage capacity of Software RAID level 5 is equal to the capacity of the member partitions, minus the size of one of the partitions if they are of equal size.

- *Linear RAID* — Linear RAID is a simple grouping of drives to create a larger virtual drive. In linear RAID, the chunks are allocated sequentially from one member drive, going to the next drive only when the first is completely filled. This grouping provides no performance benefit, as it is unlikely that any I/O operations will be split between member drives. Linear RAID also offers no redundancy and, in fact, decreases reliability —— if any one member drive fails, the entire array cannot be used. The capacity is the total of all member disks.

5 Software RAID Configuration

Read Chapter 4, *Redundant Array of Independent Disks (RAID)* first to learn about RAID and the differences between Hardware and Software RAID and the differences between RAID 0, 1, and 5.

Software RAID can be configured during the graphical installation of Red Hat Linux or during a kickstart installation. You can use fdisk or Disk Druid to create your RAID configuration, but these instructions will focus mainly on using Disk Druid to complete this task.

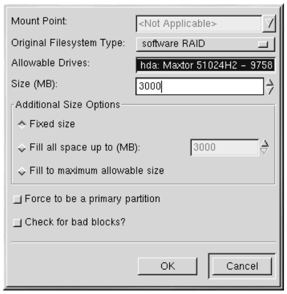
Before you can create a RAID device, you must first create RAID partitions, using the following step-by-step instructions.

Tip: If You Use fdisk

If you are using fdisk to create a RAID partition, remember that instead of creating a partition as type `83`, which is Linux native, you must create the partition as type `fd` (Linux RAID). Also, for best performance, partitions within a given RAID array should span identical cylinders on drives.

- Create a partition. In Disk Druid, choose **New** to create a new partition (see Figure 5–1, *Creating a New RAID Partition*).

Figure 5–1 Creating a New RAID Partition



Mount Point: <Not Applicable>
Original Filesystem Type: software RAID
Allowable Drives: hda: Maxtor 51024H2 – 9758
Size (MB): 3000

Additional Size Options
● Fixed size
○ Fill all space up to (MB): 3000
○ Fill to maximum allowable size

□ Force to be a primary partition
□ Check for bad blocks?

- Choose **software RAID** from the **Filesystem Type** pull-down menu.
- You will not be able to enter a mount point (you will be able to do that once you have created your RAID device).

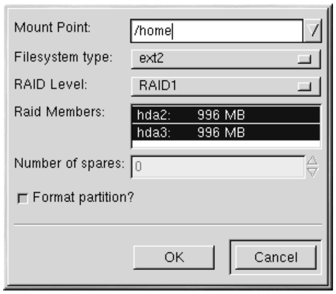
- for **Allowable Drives**, select the drive on which RAID will be created. If you have multiple drives, all drives will be selected here and you must deselect those drives which will *not* have the RAID array on them.
- Enter the size that you want the partition to be.
- Select **Fill to maximum allowable size** if you want the partition to grow to fill all available space on the hard disk. If you make more than one partition growable, the partitions will share the available free space on the disk.
- Select **Force to be a primary partition** if you want the partition to be a primary partition.
- Select **Check for bad blocks?** if you want the installation program to check for bad blocks on the hard drive before formatting it.

Continue these steps to create as many partitions as needed for your RAID setup. Notice that all the partitions do not have to be RAID partitions. For example, you can configure only the `/home` partition as a software RAID device.

Once you have all of your partitions created as **software RAID** partitions, select the **Make RAID** button on the Disk Druid main partitioning screen (see Figure 5–3, *Creating a RAID Array*).

Next, Figure 5–2, *Making a RAID Device* will appear, where you can make a RAID device.

Figure 5–2 Making a RAID Device



Mount Point: /home

Filesystem type: ext2

RAID Level: RAID1

Raid Members:
hda2: 996 MB
hda3: 996 MB

Number of spares: 0

□ Format partition?

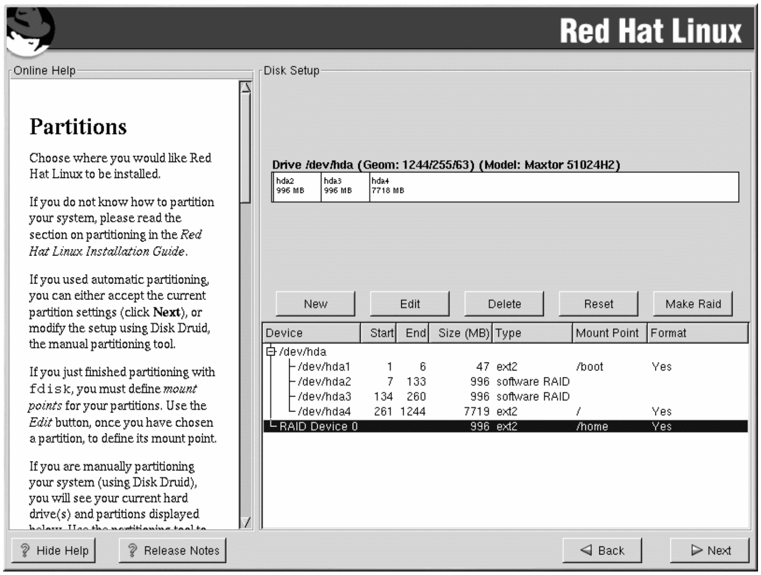
- First, enter a mount point.
- Next, choose the partition type for the partition.
- Choose your RAID type. You can choose from **RAID 0**, **RAID 1**, and **RAID 5**.

Please Note

If you are making a RAID partition of `/boot`, you must choose RAID level 1 and it must use one of the first two drives (IDE first, SCSI second). If you are not creating a RAID partition of `/boot`, and you are making a RAID partition of `/`, it must be RAID level 1 and it must use one of the first two drives (IDE first, SCSI second).

- Select which partitions will go into this RAID array and then click **Ok**.
- A spare partition can be specified for RAID 1 and RAID 5. If a software RAID partition fails, the spare will automatically be used as a replacement. For each spare you want to specify, you must create an additional software RAID partition (in addition to the partitions for the RAID device). In the previous step, select the partitions for the RAID device and the partition(s) for the spare(s). Select the number of spares.
- Select whether you want the partition formatted.
- The RAID device will appear in the **Drive Summary** list as shown in Figure 5–3, *Creating a RAID Array*. At this point, you can continue with your installation process. Refer to the *Official Red Hat Linux Installation Guide* for further instructions.

Figure 5–3 Creating a RAID Array



Partitions

Choose where you would like Red Hat Linux to be installed.

If you do not know how to partition your system, please read the section on partitioning in the *Red Hat Linux Installation Guide*.

If you used automatic partitioning, you can either accept the current partition settings (click **Next**), or modify the setup using Disk Druid, the manual partitioning tool.

If you just finished partitioning with `fdisk`, you must define *mount points* for your partitions. Use the *Edit* button, once you have chosen a partition, to define its mount point.

If you are manually partitioning your system (using Disk Druid), you will see your current hard drive(s) and partitions displayed below. Use the partitioning tool to

Drive /dev/hda (Geom: 1244/255/63) (Model: Maxtor 51024H2)

hda2 996 MB	hda3 996 MB	hda4 7718 MB

Device	Start	End	Size (MB)	Type	Mount Point	Format
/dev/hda						
/dev/hda1	1	6	47	ext2	/boot	Yes
/dev/hda2	7	133	996	software RAID		
/dev/hda3	134	260	996	software RAID		
/dev/hda4	261	1244	7719	ext2	/	Yes
RAID Device 0			996	ext2	/home	Yes

Part II Network-Related References

6 Network Configuration

Red Hat Linux no longer includes the application netcfg to configure your network devices. The Red Hat Network Administration Tool has replaced netcfg and can be used to configure the different types of network devices: Ethernet, Modem, ISDN, xDSL, CIPE, and Wireless.

You can also configure a modem, ISDN, or an xDSL connection with internet-druid. Refer to the *Official Red Hat Linux Getting Started Guide* for more details on internet-druid.

To use the Red Hat Network Administration Tool, you must be running the X Window System and have root privileges. To start the application, use one of the following methods:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Network Configuration**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Network Configuration**.
- Type the command `neat` at a shell prompt (for example, in an XTerm or a GNOME terminal).

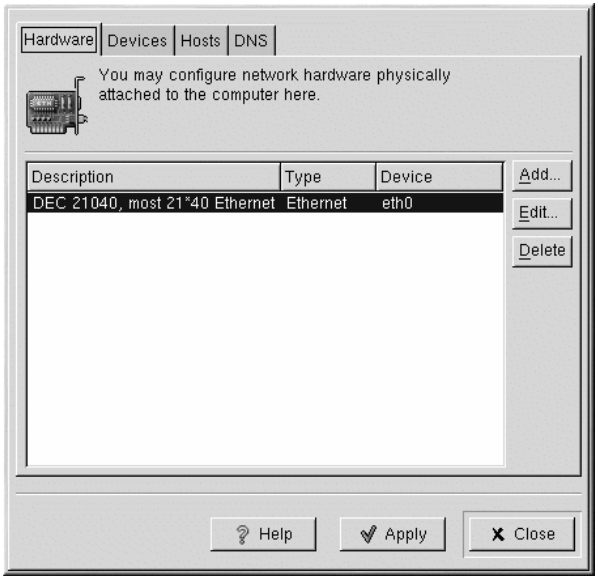
If you make any changes to your network configuration using this tool, you must click the **Apply** button to have the changes take effect.

If you prefer modifying the configuration files, refer to the *Official Red Hat Linux Reference Guide* for information on their location and contents.

6.1 Adding Network Hardware

From the main Red Hat Network Administration Tool window, use the **Hardware** tab to add, edit, or delete Ethernet, modem, ISDN, and token ring hardware configurations.

Figure 6–1 Network Hardware Configuration



You may configure network hardware physically attached to the computer here.

Description	Type	Device
DEC 21040, most 21*40 Ethernet	Ethernet	eth0

6.1.1 Ethernet

You can configure the type of adapter (manufacturer and model) and kernel device name for an Ethernet device. The type of adapter you select determines which kernel module (driver) is loaded for the network interface card. After selecting the adapter, select the kernel device name for the network interface card (`/dev/eth0`, `/dev/eth1`, and so on). You can also configure the device's system resource settings such as IRQ. After configuring the hardware settings for the Ethernet device, go to the **Device** tab to configure its network settings such as using DHCP to obtain an IP address.

6.1.2 Modem

For a modem, you can configure the kernel device name, baud rate, flow control, modem volume, and whether to use touch tone dialing. If you want to configure a modem Internet connection, go to the **Device** tab and select **Modem** as the **Device Type**.

6.1.3 ISDN

For an ISDN device, you can configure the adapter (manufacturer and model), system resources (such as IRQ), and D Channel Protocol. If you want to configure an ISDN Internet connection, go to the **Device** tab and select **ISDN** as the **Device Type**.

6.1.4 Token Ring

For a token ring device, you can select the type of adapter according to the manufacturer and model of the device. The type of adapter determines which kernel modules (driver) is loaded for the device. You can also configure the kernel device name (`/dev/tr0`, `/dev/tr1`, and so on) and the device's system resources such as IRQ. After configuring the hardware settings for the token ring device, go to the **Device** tab to configure its network settings such as using DHCP to obtain an IP address.

6.2 Adding a Device

To add a network device, start Red Hat Network Administration Tool and click **Add** in the **Devices** tab. From the **Device Type** menu, you have the following options:

- Ethernet — Select this option to configure a network interface card (NIC).
- Modem — Select this option to configure a modem for a dial-up connection.
- ISDN — Select this option if you subscribe to an ISDN Internet service.
- xDSL — Select this option if you subscribe to a type of xDSL service such as ADSL.
- CIPE — Select this option to configure a virtual CIPE device.
- Wireless — Select this option to configure a wireless network device.
- Token Ring — Select this option to configure a token ring device.

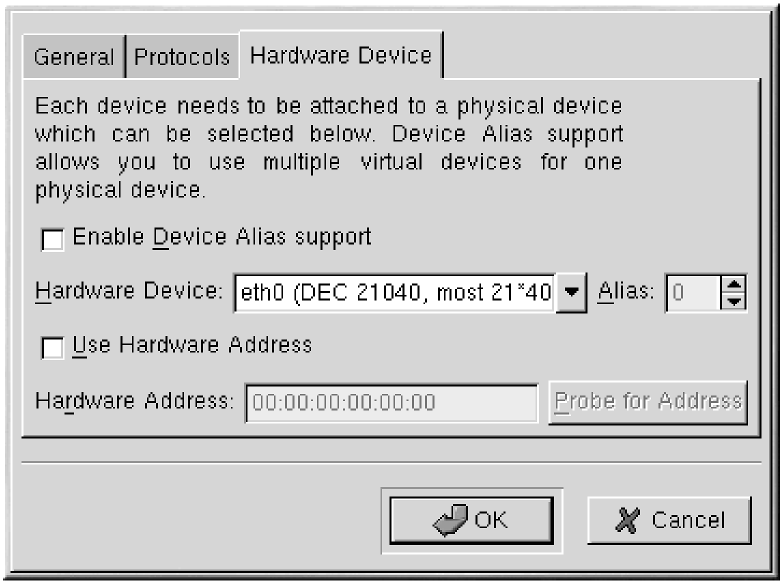
After selecting a device type, you will see a window with tabbed panes. The tabs vary depending on which device type you selected. All device types will have the following tabs:

- **General** — Give the device a nickname, choose to activate the device when the computer boots, and choose to allow users to enable and disable the device.
- **Protocols** — Edit the TCP/IP settings such as an IP address (including DHCP), hostname, and static network routes.

6.2.1 Ethernet

If you selected **Ethernet** as the device type, you will also see a **Hardware Device** tab. Use this tab to configure a device alias. A device alias allows you to setup multiple virtual devices for one physical device.

Figure 6–2 Adding an Ethernet Device



Each device needs to be attached to a physical device which can be selected below. Device Alias support allows you to use multiple virtual devices for one physical device.

□ Enable Device Alias support

Hardware Device: eth0 (DEC 21040, most 21*40 Alias: 0

□ Use Hardware Address

Hardware Address: 00:00:00:00:00:00

6.2.2 Modem

Click the **Provider** tab to enter the phone number, login, and password for your dial-up account. Use the **Compression** tab to enable different forms of compression. The **Options** tab allows you to configure PPP options, and the **Advanced** tab provides pulldown menus to customize the hangup timeout value, the dial mode, and the modem port. You can also configure the device to restart if the connection dies using the **Advanced** tab.

6.2.3 ISDN

The tabs for ISDN configuration are similar to the tabs for **Modem** configuration, except there is an additional tab that allows you to use callback and configure the callback settings.

6.2.4 xDSL

xDSL provides an Internet connection through an Ethernet card. To configure xDSL, you must configure an Ethernet device first. Most xDSL services require you to configure the Ethernet device to obtain an IP address via DHCP. Consult your Internet provider for details. After configuring the Ethernet device, add an xDSL device. From the **Provider** tab, select the appropriate Ethernet device to use to establish your connection.

6.2.5 CIPE

CIPE stands for Crypto IP Encapsulation. It is used to configure an IP tunneling device. For example, CIPE can be used to grant access from the outside world into a Virtual Private Network (VPN). If you need to setup a CIPE device, contact your system administrator for the correct values.

6.2.6 Wireless

The tabs for Wireless configuration are similar to the tabs for an Ethernet device, except there is an extra tab called **Wireless Settings**. This tab allows you to configure the network ID, mode, frequency, channel, transmit rate, and key for the wireless device.

6.2.7 Token Ring

The **Token Ring** device configuration is similar to the **Ethernet** device configuration. There is an additional **Hardware Device** tab.

6.3 Managing DNS Settings

The **DNS** tab allows you to configure the system's hostname, domain, name servers, and search domain. Name servers are used to look up other hosts on the network. Note, the name servers section does not configure the system to be a name server.

The **Hosts** tab allows you to add, edit, or remove hosts from the `/etc/hosts` file. This file contains IP addresses and the hostnames to which the IP addresses should be resolved.

When your system tries to resolve a hostname to an IP address or determine the hostname for an IP address, it refers to the `/etc/hosts` file before using the name servers (if you are using the default Red Hat Linux configuration). If the IP address is listed in the `/etc/hosts` file, the name servers are not used.

To add an entry to the `/etc/hosts` file, click **Add** in the **Hosts** tab, provide the requested information, and click **OK**. Click **Apply** to write the entry to the file.

Tip

To change lookup order, edit the `/etc/host.conf` file. The line `order hosts, bind` specifies that the `/etc/hosts` takes precedence over the name servers. Changing the line to `order bind, hosts` configures your system to resolve hostnames and IP addresses using the name servers first. If the IP address can not be resolved through the name servers, your system looks for the IP address in the `/etc/hosts` file.

7 Basic Firewall Configuration

During the Red Hat Linux installation, you are given the option to choose high, medium or no security level as well as allow specific devices, incoming services, and ports. These levels are based on the GNOME Lokkit firewall configuration application.

After installation, you can change the security level of your system by using GNOME Lokkit.

GNOME Lokkit allows you to configure firewall settings for an average user by constructing basic `ipchains` networking rules. Instead of having to write the rules, this program asks you a series of questions about how you use your system and then write it for you in the file `/etc/sysconfig/ipchains`.

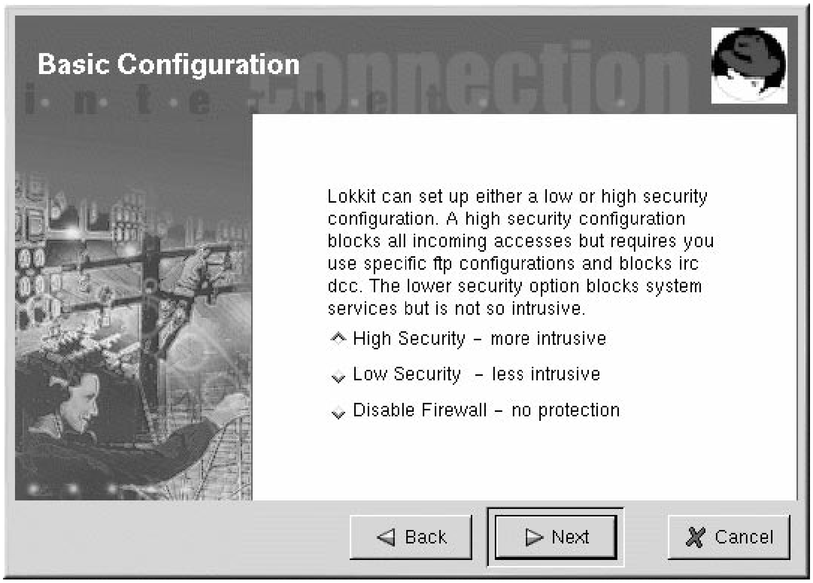
You should not try to use GNOME Lokkit to generate complex firewall rules. It is intended for average users who want to protect themselves while using a modem, cable, or DSL Internet connection. To configure specific firewall rules, refer to the *Firewalling with `iptables`* chapter in the *Official Red Hat Linux Reference Guide*.

To disable specific services and deny specific hosts and users, refer to Chapter 8, *Controlling Access to Services*.

To start GNOME Lokkit, type the command `gnome-lokkit` at a shell prompt as root.

7.1 Basic

Figure 7–1 Basic



Basic Configuration

Lokkit can set up either a low or high security configuration. A high security configuration blocks all incoming accesses but requires you use specific ftp configurations and blocks irc dcc. The lower security option blocks system services but is not so intrusive.

● High Security – more intrusive

○ Low Security – less intrusive

○ Disable Firewall – no protection

After starting the program, choose the appropriate security level for your system:

- **High Security** — This option disables almost all network connects except DNS replies and DHCP so that network interfaces can be activated. IRC, ICQ, and other instant messaging services as well as RealAudio™ will not work without a proxy.

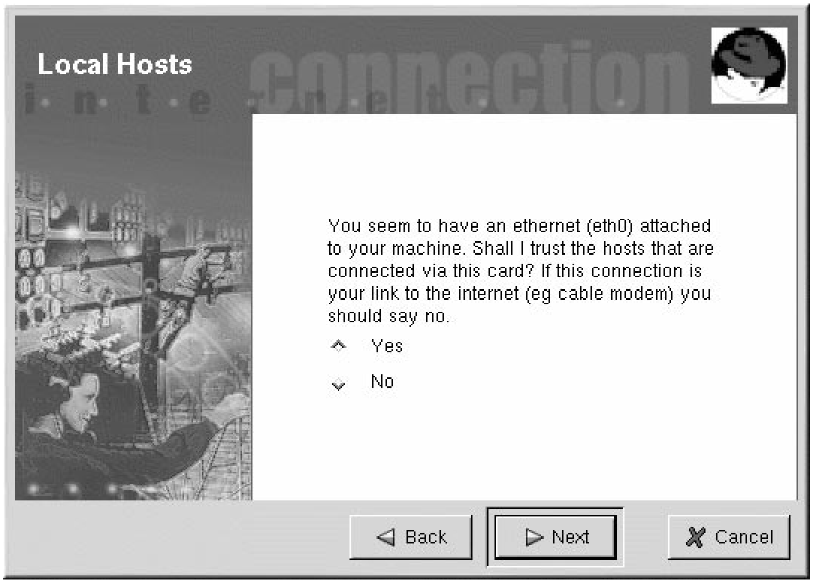
- **Low Security** — This option will not allow remote connections to the system, including NFS connections and remote X Window System sessions. Services that run below port 1023 will not accept connections, including FTP, SSH, Telnet, and HTTP.

- **Disable Firewall** — This option does not create any security rules. It is recommended that this option only be chosen if the system is on a trusted network (not on the Internet), if the system is behind a larger firewall, or if you write your own custom firewall rules. If you choose this option and click **Next**, proceed to Section 7.5, *Activating the Firewall*. The security of your system will not be changed.

7.2 Local Hosts

If there are Ethernet devices on the system, the **Local Hosts** page allows you to configure whether the firewall rules apply to connection requests sent to each device. If the device connects the system to a local area network behind a firewall and does not connect directly to the Internet, select **Yes**. If the Ethernet card connects the system to a cable or DSL modem, it is recommended that you select **No**.

Figure 7–2 Local Hosts



You seem to have an ethernet (eth0) attached to your machine. Shall I trust the hosts that are connected via this card? If this connection is your link to the internet (eg cable modem) you should say no.

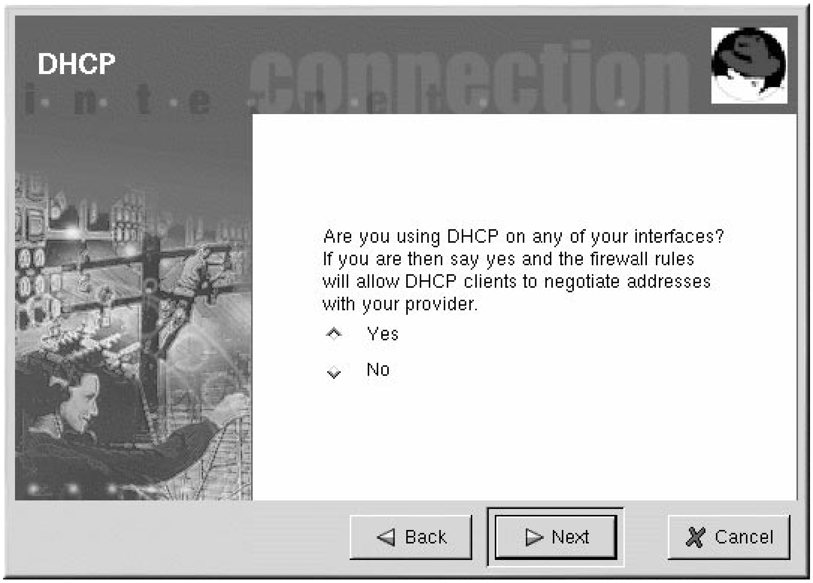
○ Yes

○ No

7.3 DHCP

If you are using DHCP to activate any Ethernet interfaces on the system, you must say **Yes** to the DHCP question. If you say no, you will not be able to establish a connect using the Ethernet interface. Many cable and DSL Internet providers require you to use DHCP to establish an Internet connection.

Figure 7–3 DHCP



Are you using DHCP on any of your interfaces?
If you are then say yes and the firewall rules
will allow DHCP clients to negotiate addresses
with your provider.

◇ Yes

◇ No

7.4 Configuring Services

GNOME Lokkit also allows you to turn common services on and off. If you answer **Yes** to configuring services, you are prompted about the following services:

- **Web Server** — Choose this option if you want people to connect to a Web server such as Apache running on your system. You do not need to choose this option if you want to view pages on your own system or on other servers on the network.
- **Incoming Mail** — Choose this option if your system needs to accept incoming mail. You do not need this option if you retrieve email using IMAP, POP3, or fetchmail.
- **Secure Shell** — Secure Shell, or SSH, is a suite of tools for logging into and executing commands on a remote machine over an encrypted connection. If you need to access your machine remotely through ssh, select this option.
- **Telnet** — Telnet allows you to log into your machine remotely; however, it is not secure. It sends plain text (including passwords) over the network. It is recommended that you use SSH to log into your machine remotely. If you are required to have telnet access to your system, select this option.

Tip

To disable other services that you do not need, you can use Serviceconf. See Section 8.3, *Serviceconf*.

7.5 Activating the Firewall

Clicking **Finish** on the **Activate the Firewall** page will write the firewall rules to `/etc/sysconfig/ipchains` and start the firewall by starting the `ipchains` service.

It is highly recommended that you run GNOME Lokkit from the machine, not from a remote X session. If you disable remote access to your system, you will no longer be able to access it or disable the firewall rules.

Click **Cancel** if you do not want to write the firewall rules.

7.5.1 Mail Relay

A mail relay is a system that allows other systems to send email through it. If your system is a mail relay, someone can possibly use it to spam others from your machine.

If you chose to enable mail services, after you click **Finish** on the **Activate the Firewall** page, you will be prompted to check for mail relay. If you choose **Yes** to check for mail relay, GNOME Lokkit will attempt to connect to the *Mail Abuse Prevention System* website at http://www.mail-abuse.org/ and run a mail relay test program. The results of the test will be displayed when it is finished. If your system is open to mail relay, it is highly recommended that you configure Sendmail to prevent it.

7.5.2 Activating the `ipchains` Service

The firewall rules will only be active if the `ipchains` service is running. To manual start the service, use the command:

```
/sbin/service ipchains restart
```

To ensure that it is started when the system is booted, issue the command:

```
/sbin/chkconfig --level 345 ipchains on
```

Tip

You can also use Serviceconf to activate `ipchains`. See Section 8.3, *Serviceconf*.

8 Controlling Access to Services

Maintaining security on your Red Hat Linux system is extremely important. One way to manage security on your system is to carefully manage access to system services. Your system may need to provide open access to particular services (for example, `httpd` if you are running a Web server). However, if you do not need to provide a service, you should turn it off — this will minimize your exposure to possible bug exploits.

There are several different methods for managing access to system services. You must decide which method you would like to use based on the service, your system's configuration, and your level of Linux expertise.

The easiest way to deny access to a service is to simply turn it off. Both the services managed by `xinetd` (which we will talk about more later in this section) and the services in the `/etc/rc.d` hierarchy can be configured to start or stop using three different applications:

- serviceconf — a graphical application that displays a description of each service, displays whether each service is started at boot time (for runlevels 3, 4, and 5), and allows you to start, stop, and restart each service.
- ntsysv — a text-based application that allows you to configure which services are started at boot time for each runlevel. Changes do not take effect immediately. Services can not be started, stopped, or restarted using this program.
- `chkconfig` — a command-line utility that allows you to turn services on and off for the different runlevels. Changes do not take effect immediately. Services can not be started, stopped, or restarted using this utility.

You may find that these tools are easier to use than the alternatives — editing the numerous symbolic links located in the directories below `/etc/rc.d` by hand or editing the `xinetd` configuration files in `/etc/xinetd.d`.

Another way to manage access to system services is by using `iptables` to configure an IP firewall. If you are a new Linux user, please realize that `iptables` may not be the best solution for you. Setting up `iptables` can be complicated and is best tackled by experienced UNIX/Linux system administrators.

On the other hand, the benefit of using `iptables` is flexibility. For example, if you need a customized solution which provides certain hosts access to certain services, `ipchains` can provide it for you. See the *Official Red Hat Linux Reference Guide* for more information about `iptables`.

Alternatively, if you are looking for a utility which will set general access rules for your home machine, and/or if you are new to Linux, you should try the GNOME Lokkit utility. GNOME Lokkit is a GUI utility which will ask you questions about how you want to use your machine. Based on your answers, it will then configure a simple firewall for you. Refer to Chapter 7, *Basic Firewall Configuration* for more information.

8.1 Runlevels

Before you can configure access to services, you must understand Linux runlevels. A runlevel is a state, or **mode**, that is defined by the services listed in the directory `/etc/rc.d/rc<x>.d`, where *<x>* is the number of the runlevel.

Red Hat Linux uses the following runlevels:

- 0 — Halt
- 1 — Single-user mode
- 2 — Not used (user-definable)
- 3 — Full multi-user mode
- 4 — Not used (user-definable)
- 5 — Full multi-user mode (with an X-based login screen)
- 6 — Reboot

If you configured the X Window System during the Red Hat Linux installation program, you had the option of choosing a graphical or text login screen. If you chose a text login screen, you are operating in runlevel 3. If you chose a graphical login screen, you are operating in runlevel 5.

The default runlevel can be changed by modifying the `/etc/inittab` file, which contains a line near the top of the file similar to the following:

```
id:3:initdefault:
```

Change the number in this line to the desired runlevel. The change will not take effect until you reboot the system.

To change the runlevel immediately, use the command `telinit` followed by the runlevel number. You must be root to use this command.

8.2 TCP Wrappers

Many UNIX system administrators are accustomed to using TCP wrappers to manage access to certain network services. Any network services managed by `xinetd` (as well as any program with built-in support for libwrap) can use TCP wrappers to manage access. `xinetd` can use the `/etc/hosts.allow` and `/etc/hosts.deny` files to configure access to system services. As the names imply, `hosts.allow` contains a list of rules clients allowed to access the network services controlled by `xinetd`, and `hosts.deny` contains rules to deny access. The `hosts.allow` file takes precedence over the hosts.deny file. Permissions to grant or deny access can be based on individual IP address (or hostnames) or on a pattern of clients. See the *Official Red Hat Linux Reference Guide* and the hosts_access man page for details.

8.2.1 xinetd

To control access to Internet services, use xinetd, which is a secure replacement for inetd. The xinetd daemon conserves system resources, provides access control and logging, and can be used to start special-purpose servers. xinetd can be used to provide access only to particular hosts, to deny access to particular hosts, to provide access to a service at certain times, to limit the rate of incoming connections and/or the load created by connections, etc.

xinetd runs constantly and listens on all of the ports for the services it manages. When a connection request arrives for one of its managed services, xinetd starts up the appropriate server for that service.

The configuration file for xinetd is /etc/xinetd.conf, but you will notice upon inspection of the file that it only contains a few defaults and an instruction to include the /etc/xinetd.d directory. To enable or disable a xinetd service, edit its configuration file in the /etc/xinetd.d directory. If the disable attribute is set to **yes**, the service is disabled. If the disable attribute is set to **no**, the service is enabled. If you edit any of the xinetd configuration files or change its enabled status using Serviceconf, ntsysv, or chkconfig, you must restart xinetd with the command service xinetd restart before the changes will take effect. For a list of network services controlled by xinetd list of the contents of the /etc/xinetd.d directory with the command ls /etc/xinetd.d.

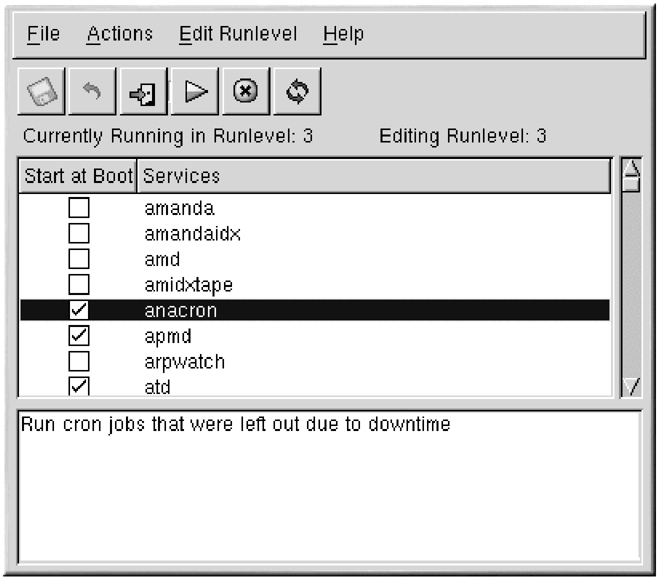
8.3 Serviceconf

Serviceconf is a graphical application developed by Red Hat to configure which SysV services in /etc/rc.d/init.d are started at boot time (for runlevels 3, 4, and 5) and which xinetd services are enabled. It also allows you to start, stop, and restart SysV services as well as restart xinetd.

To start Serviceconf, use one of the following commands:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Serviceconf**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Serviceconf**.
- Type the command serviceconf at a shell prompt (for example, in an XTerm or a GNOME terminal).

Figure 8–1 Serviceconf



Currently Running in Runlevel: 3 Editing Runlevel: 3

Start at Boot	Services
□	amanda
□	amandaidx
□	amd
□	amidxtape
☑	anacron
☑	apmd
□	arpwatch
☑	atd

Run cron jobs that were left out due to downtime

Serviceconf displays the current runlevel as well as which runlevel you are currently editing. To edit a different runlevel, select **Edit Runlevel** from the pulldown menu and select runlevel 3, 4, or 5. Refer to Section 8.1, *Runlevels* for a description of runlevels.

Serviceconf lists the services from `/etc/rc.d/init.d` as well as the services controlled by `xinetd`. Click on a service to display a brief description of that service at the bottom of the window.

To start, stop, or restart a service immediately, select the service and choose the action from the **Actions** pulldown menu. You can also select the service and click the start, stop, or restart button on the toolbar.

If you select an `xinetd` service such as telnet, the **Start**, **Stop**, and **Restart** buttons will not be active. If you change the **Start at Boot** value of an `xinetd` service, you must click the **Save Changes** button to restart `xinetd` and disable/enable the `xinetd` services that you changed.

To enable a service at boot time for the currently selected runlevel, check the checkbox beside the name of the service under the **Start at Boot** column. After configuring the runlevel, you must apply the changes. Select **File** => **Save Changes** from the pulldown menu or click the **Save Changes** button.

WARNING

When you save changes to xinetd services, xinetd is restarted. When you save changes to other services, the runlevel is reconfigured, but the changes do not take effect immediately.

If you check or uncheck the **Start at Boot** value for a service in /etc/rc.d/init.d, the **Save Changes** button will become active. Click it to reconfigure the currently selected runlevel. The changes do not affect the system immediately. For example, assume you are configuring runlevel 3. If you change the **Start at Boot** value for the anacron service from checked to unchecked and then click the **Save Changes** button, the runlevel 3 configuration changes so that anacron is not started at boot time. However, runlevel 3 is not reinitialized, so anacron is still running. Select one of following options at this point:

1. Stop the anacron service — Stop the service by selecting it from the list and clicking the **Stop the selected service** button. A message will be displayed stating that the service was stopped successfully.
2. Re-initialize the runlevel — Reinitialize the runlevel by going to a shell prompt (such as an XTerm or GNOME terminal) and typing the command `telinit 3` (where 3 is the runlevel number). This option is recommended if you change the **Start at Boot** value of more than one service and want to activate the changes immediately.
3. Do nothing else — You do not have to stop the anacron service. You can wait until the system is rebooted for the service to stop. The next time the system is booted, the runlevel will be initialized without the anacron service running.

8.4 ntsysv

The ntsysv utility provides a simple interface for activating or deactivating services. You can use ntsysv to turn an xinetd-managed service on or off. You can also use ntsysv to start or stop a service in the /etc/rc.d hierarchy; in that case, the `ntsysv` command (without options) is used to configure current runlevel. If you want to configure a different runlevel, use something like `ntsysv --levels 016`. (In this example, you would be setting the services for runlevels 0, 1 and 6.)

The ntsysv interface works like the textmode installation program. Use the up and down arrows to navigate up and down the list. The space bar selects/unselects services and is also used to "press" the **Ok** and **Cancel** buttons. To move between the list of services and the **Ok** and **Cancel** buttons, use the [Tab] key. An ***** signifies that a service is set to on. The [F1] key will pop up a short description of each service.

WARNING

Changes do not take effect immediately after using `ntsysv`. You must stop or start the individual service with the command `service` *`daemon`* `stop`. In the previous example, replace *daemon* with the name of the service you want to stop; for example, `httpd`. Replace `stop` with `start` or `restart` to start or restart the service. If you want to start or stop a service which is managed by `xinetd`, use the command `service xinetd restart`.

8.5 chkconfig

The `chkconfig` command can also be used to activate and deactivate services. If you use the `chkconfig --list` command, you will see a list of system services and whether they are started (`on`) or stopped (`off`) in runlevels 0-6 (at the end of the list, you will see a section for the services managed by `xinetd`, which we'll discuss later in this section).

If you use `chkconfig --list` to query a service managed by `xinetd`, you will see whether the `xinetd` service is enabled (on) or disabled (off). For example, the following command shows that `finger` is enabled as an xinetd service:

```
$ chkconfig --list finger
finger          on
```

As shown above, if `xinetd` is running, `finger` is enabled.

If you use `chkconfig --list` to query a service in `/etc/rc.d`, you will see the service's settings for each runlevel, like the following:

```
$ chkconfig --list anacron
anacron         0:off   1:off   2:on    3:on
4:on    5:on    6:off
```

More importantly, `chkconfig` can be used to set a service to be started (or not) in a specific runlevel. For example, if we wanted to turn `nscd` off in runlevels 3, 4, and 5, we'd use a command like this:

```
chkconfig --level 345 nscd off
```

See the `chkconfig` man page for more information on how to use it.

WARNING

Changes do not take effect immediately after using `chkconfig`. You must stop or start the individual service with the command `service` *`daemon`* `stop`. In the previous example, replace *daemon* with the name of the service you want to stop; for example, `httpd`. Replace `stop` with `start` or `restart` to start or restart the service. If you want to start or stop a service which is managed by `xinetd`, use the command `service xinetd restart`.

8.6 Additional Resources

For more information on `xinetd`, refer to the following resources.

8.6.1 Installed Documentation

- `man ntsysv` — The ntsysv manual page.
- `man chkconfig` — The `chkconfig` manual page.
- `man xinetd` — The `xinetd` manual page.
- `man xinetd.conf` — The manual page for the `xinetd.conf` configuration file.
- `man 5 hosts_access` — The manual page for the format of host access control files (in section 5 of the man pages).

8.6.2 Useful Websites

- http://www.xinetd.org — The `xinetd` webpage. It contains the a more detailed list of features and sample configuration files.

9 OpenSSH

OpenSSH is a free, open source implementation of the SSH (Secure SHell) protocols. It replaces `telnet`, `ftp`, `rlogin`, `rsh`, and `rcp` with secure, encrypted network connectivity tools. OpenSSH supports versions 1.3, 1.5, and 2 of the SSH protocol. Since OpenSSH version 2.9, the default protocol in Red Hat Linux 7.2 is version 2, which uses RSA keys as the default.

9.1 Why Use OpenSSH?

If you use OpenSSH tools, you are enhancing the security of your machine. All communications using OpenSSH tools, including passwords, are encrypted. `Telnet` and `ftp` use plaintext passwords and send all information unencrypted. The information can be intercepted, the passwords can be retrieved, and then your system can be compromised by an unauthorized person logging in to your system using one of the intercepted passwords. The OpenSSH set of utilities should be used whenever possible to avoid these security problems.

Another reason to use OpenSSH is that it automatically forwards the `DISPLAY` variable to the client machine. In other words, if you are running the X Window System on your local machine, and you log in to a remote machine using the `ssh` command, when you execute a program on the remote machine that requires X, it will be displayed on your local machine. This is convenient if you prefer graphical system administration tools but do not always have physical access to your server.

9.2 Configuring an OpenSSH Server

To run an OpenSSH server, you must first make sure that you have the proper RPM packages installed. The `openssh-server` package is required and depends on the `openssh` package. Both of these packages are included in Red Hat Linux 7.2.

The OpenSSH daemon uses the configuration file `/etc/ssh/sshd_config`. The default configuration file installed with Red Hat Linux 7.2 should be sufficient for most purposes. If you want to configure the daemon in ways not provided by the default `sshd_config`, read the `sshd` manual page for a list of the keywords that can be defined in the configuration file.

To start the OpenSSH service, use the command `/sbin/service sshd start`. To stop the OpenSSH server, use the command `/sbin/service sshd stop`. If you want the daemon to start automatically at boot time, see Chapter 8, *Controlling Access to Services* for information on how to manage services.

9.3 Configuring an OpenSSH Client

To connect to an OpenSSH server from a client machine, you must have the `openssh-clients` and `openssh` packages installed on the client machine.

9.3.1 Using the ssh Command

The `ssh` command is a secure replacement for the `rlogin`, `rsh`, and `telnet` commands. It allows you to log in to and execute commands on a remote machine.

Logging in to a remote machine with `ssh` is similar to using `telnet`. To log in to a remote machine named penguin.example.net, type the following command at a shell prompt:

```
ssh penguin.example.net
```

The first time you `ssh` to a remote machine, you will see a message similar to the following:

```
The authenticity of host ’penguin.example.net’ can’t be established.
DSA key fingerprint is 94:68:3a:3a:bc:f3:9a:9b:01:5d:b3:07:38:e2:11:0c.
Are you sure you want to continue connecting (yes/no)?
```

Type **`yes`** to continue. This will add the server to your list of known hosts as seen in the following message:

> Warning: Permanently added ’penguin.example.net’ (DSA) to the list of known hosts.

Next, you’ll see a prompt asking for your password for the remote machine. After entering your password, you will be at a shell prompt for the remote machine. If you use `ssh` without any command line options, the username that you are logged in as on the local client machine is passed to the remote machine. If you want to specify a different username, use the following command:

```
ssh -l username penguin.example.net
```

You can also use the syntax `ssh` *`username`*`@penguin.example.net`.

The `ssh` command can be used to execute a command on the remote machine without logging in to a shell prompt. The syntax is `ssh` *`hostname command`*. For example, if you want to execute the command `ls /usr/share/doc` on the remote machine penguin.example.net, type the following command at a shell prompt:

```
ssh penguin.example.net ls /usr/share/doc
```

After you enter the correct password, the contents of `/usr/share/doc` will be displayed, and you will return to your shell prompt.

9.3.2 Using the `scp` Command

The `scp` command can be used to transfer files between machines over a secure, encrypted connection. It is similar to `rcp`.

The general syntax to transfer a local file to a remote system is `scp` *`localfile username@tohostname:/newfilename`*. The *localfile* specifies the source, and the group of *username@tohostname:/newfilename* specifies the destination.

To transfer the local file `shadowman` to your account on penguin.example.net, type the following at a shell prompt (replace *username* with your username):

```
scp shadowman username@penguin.example.net:/home/username
```

This will transfer the local file `shadowman` to `/home/`*`username/`*`shadowman` on penguin.example.net.

The general syntax to transfer a remote file to the local system is `scp` *`username@tohostname:/remotefile`* */newlocalfile*. The *remotefile* specifies the source, and *newlocalfile* specifies the destination.

Multiple files can be specified as the source files. For example, to transfer the contents of the directory `/downloads` to an existing directory called `uploads` on the remote machine penguin.example.net, type the following at a shell prompt:

```
scp /downloads/* username@penguin.example.net:/uploads/
```

9.3.3 Using the `sftp` Command

The `sftp` utility can be used to open a secure, interactive FTP session. It is similar to `ftp` except that it uses a secure, encrypted connection. The general syntax is *`sftp username@hostname.com`*. Once authenticated, you can use a set of commands similar to using FTP. Refer to the `sftp` manual page for a list of these commands. To read the manual page, execute the command `man sftp` at a shell prompt. The `sftp` utility is only available in OpenSSH version 2.5.0p1 and higher.

9.3.4 Generating Key Pairs

If you do not want to enter your password every time you `ssh`, `scp`, or `sftp` to a remote machine, you can generate an authorization key pair.

Separate Authorization Key Pairs

You must have separate authorization key pairs for SSH Protocol 1 (RSA) and SSH Protocol 2 (DSA).

WARNING

Keys must be generated for each user. To generate keys for a user, follow the following steps as the user who wants to connect to remote machines. If you complete the following steps as root, only root will be able to use the keys.

Generating a DSA Key Pair

Use the following steps to generate a DSA key pair. DSA is used by SSH Protocol 2.

1. To generate a DSA key pair to work with version 2 of the protocol, type the following command at a shell prompt:

   ```
   ssh-keygen -t dsa
   ```

   Accept the default file location of `~/.ssh/id_dsa`. Enter a passphrase different from your account password and confirm it by entering it again. [1]

   ### What is a Passphrase?

   A passphrase is a string of words and characters used to authenticate a user. Passphrases differ from passwords in that you can use spaces or tabs in the passphrase. Passphrases are generally longer than passwords because they are usually phrases instead of just a word.

2. Change the permissions of your `.ssh` directory using the command `chmod 755 ~/.ssh`.

3. Copy the contents of `~/.ssh/id_dsa.pub` to `~/.ssh/authorized_keys2` on the machine to which you want to connect. If the file `~/.ssh/authorized_keys2` doesn't exist, you can copy the file `~/.ssh/id_dsa.pub` to the file `~/.ssh/authorized_keys2` on the other machine.[1]

4. If you are running GNOME, skip to *Configuring ssh-agent with GNOME* in Section 9.3.4. If you are not running the X Window System, skip to *Configuring ssh-agent* in Section 9.3.4.

Generating an RSA Key Pair for Version 2

Use the following steps to generate a RSA key pair for version 2 of the SSH protocol. This is the default starting with OpenSSH 2.9.

1. To generate a RSA key pair to work with version 2 of the protocol, type the following command at a shell prompt:

   ```
   ssh-keygen -t rsa
   ```

   Accept the default file location of `~/.ssh/id_rsa`. Enter a passphrase different from your account password and confirm it by entering it again. [1]

2. Change the permissions of your `.ssh` directory using the command `chmod 755 ~/.ssh`.

3. Copy the contents of `~/.ssh/id_rsa.pub` to `~/.ssh/authorized_keys2` on the machine to which you want to connect. If the file `~/.ssh/authorized_keys2` doesn't exist, you can copy the file `~/.ssh/id_rsa.pub` to the file `~/.ssh/authorized_keys2` on the other machine.[1]

4. If you are running GNOME, skip to *Configuring ssh-agent with GNOME* in Section 9.3.4. If you are not running the X Window System, skip to *Configuring ssh-agent* in Section 9.3.4.

Generating an RSA Key Pair for Version 1.3 and 1.5

Use the following steps to generate an RSA key pair, which is used by version 1 of the SSH Protocol. If you are only connecting between Red Hat Linux 7.2 systems, you do not need an RSA key pair.

1. To generate an RSA (for version 1.3 and 1.5 protocol) key pair, type the following command at a shell prompt:

   ```
   ssh-keygen
   ```

   Accept the default file location (`~/.ssh/identity`). Enter a passphrase different from your account password. Confirm the passphrase by entering it again.

2. Change the permissions of your `.ssh` directory and your keys with the commands `chmod 755 ~/.ssh` and `chmod 644 ~/.ssh/identity.pub`.

3. Copy the contents of `~/.ssh/identity.pub` to the file `~/.ssh/authorized_keys` on the machine to which you wish to connect. If the file `~/.ssh/authorized_keys` doesn't exist, you can copy the file `~/.ssh/identity.pub` to the file `~/.ssh/authorized_keys` on the remote machine. [1]
4. If you are running GNOME, skip to *Configuring ssh-agent with GNOME* in Section 9.3.4. If you are not running GNOME, skip to *Configuring ssh-agent* in Section 9.3.4.

Configuring ssh-agent with GNOME

The `ssh-agent` utility can be used to save your passphrase so that you do not have to enter it each time you initiate an `ssh` or `scp` connection. If you are using GNOME, the `openssh-askpass-gnome` utility can be used to prompt you for your passphrase when you log in to GNOME and save it until you log out of GNOME. You will not have to enter your password or passphrase for any `ssh` or `scp` connection made during that GNOME session. If you are not using GNOME, refer to *Configuring ssh-agent* in Section 9.3.4.

To save your passphrase during your GNOME session, follow the following steps:

1. You'll need to have the package `openssh-askpass-gnome` installed; you can use the command `rpm -q openssh-askpass-gnome` to determine if it is installed or not. If it is not installed, install it from your Red Hat CD-ROM set, from a Red Hat FTP mirror site, or using Red Hat Network.
2. If you do not have an `~/.Xclients` file, you can run `switchdesk` to create it. In your `~/.Xclients` file, edit the following line:

   ```
   exec $HOME/.Xclients-default
   ```

   Change the line so that it instead reads:

   ```
   exec /usr/bin/ssh-agent $HOME/.Xclients-default
   ```

3. Open the GNOME Control Center (**GNOME Main Menu Button** => **Programs** => **Settings** => **GNOME Control Center**) and go to **Session** => **Startup Programs**. Click **Add** and enter **`/usr/bin/ssh-add`** in the **Startup Command** text area. Set it a priority to a number higher than any existing commands to ensure that it is executed last. A good priority number for `ssh-add` is 70 or higher. The higher the priority number, the lower the priority. If you have other programs listed, this one should have the lowest priority. Click **OK** to save your settings, and exit the GNOME Control Center.

[1] The ~ stands for the home directory of the currently logged in user. See the *Official Red Hat Linux Getting Started Guide* for more details.

4. Log out and then log back into GNOME; in other words, restart X. After GNOME is started, a dialog box will appear prompting you for your passphrase(s). Enter the passphrase requested. If you have both DSA and RSA key pairs configured, you will be prompted for both. From this point on, you should not be prompted for a password by `ssh`, `scp`, or `sftp`.

Configuring ssh-agent

The `ssh-agent` can be used to store your passphrase so that you do not have to enter it each time you make a `ssh` or `scp` connection. If you are not running the X Window System, follow these steps from a shell prompt. If you are running GNOME but you do not want to configure it to prompt you for your passphrase when you log in (see *Configuring ssh-agent with GNOME* in Section 9.3.4), this procedure will work in a terminal window, such as an xterm. If you are running X but not GNOME, this procedure will work in a terminal window, such as an xterm. However, your passphrase will only be remembered for that terminal window; it is not a global setting.

1. At a shell prompt, type the following command:

   ```
   exec /usr/bin/ssh-agent $SHELL
   ```

   Then type the command

   ```
   ssh-add
   ```

   and enter your passphrase(s). If you have both DSA and RSA key pairs configured, you will be prompted for both.

2. When you log out, your passphrase will be forgotten. You must execute these two commands each time you log in to a virtual console or open a terminal window.

9.4 Additional Resources

The OpenSSH and OpenSSL projects are in constant development, so the most up-to-date information for them will be found on their websites. The man pages for OpenSSH and OpenSSL tools are also good sources of detailed information.

9.4.1 Installed Documentation

- The `ssh`, `scp`, `sftp`, `sshd`, and `ssh-keygen` commands — These man pages include information on how to use these commands as well as all the parameters that can be used with them.

9.4.2 Useful Websites

- http://www.openssh.com — The OpenSSH FAQ page, bug reports, mailing lists, project goals, and a more technical explanation of the security features.
- http://www.openssl.org — The OpenSSL FAQ page, mailing lists, and a description of the project goal.
- http://www.freessh.org — SSH client software for other platforms.

10 Network File System (NFS)

Network File System (NFS) is a way to share files between machines on a network as if the files were located on your local hard drive. Red Hat Linux can be both an NFS server and an NFS client, which means that it can export filesystems to other systems, and mount filesystems exported from other machines.

10.1 Why Use NFS?

NFS is useful for sharing directories of files between multiple users on the same network. For example, a group of users working on the same project can have access to the files for that project using a shared portion of the NFS filesystem (commonly known as an NFS share) mounted in the directory `/myproject`. To access the shared files, the user goes into the `/myproject` directory on his machine. There are no passwords to enter or special commands to remember. The user works as if the directory is on his local machine.

10.2 Mounting NFS Filesystems

Use the `mount` command to mount an NFS filesystem from another machine:

```
mount shadowman:/mnt/export /mnt/local
```

Directory Must Exist

The mount point directory on local machine (`/mnt/local` in the above example) must exist.

In this command, shadowman is the hostname of the NFS fileserver, `/mnt/export` is the filesystem that shadowman is exporting, and `/mnt/local` is a directory on the local machine where we want to mount the filesystem. After the `mount` command runs (and if we have the proper permissions from shadowman) we can enter `ls /mnt/local` and get a listing of the files in `/mnt/export` on shadowman.

10.2.1 Mounting NFS Filesystems using `/etc/fstab`

An alternate way to mount an NFS share from another machine is to add a line to your `/etc/fstab` file. The line must state the hostname of the NFS server, the directory on the server being exported, and the directory on the local machine where you want to mount the filesystem. You must be root to modify the `/etc/fstab` file.

The general syntax for the line in `/etc/fstab` is as follows:

```
server:/usr/local/pub    /pub   nfs    rsize=8192,wsize=8192,timeo=14,intr
```

The mount point /pub must exist on your machine. After adding this line to /etc/fstab, you can type the command mount /pub at a shell prompt, and the mount point /pub will be mounted from the server.

10.2.2 Mounting NFS Filesystems using autofs

A third option for mounting an NFS share is the use of autofs. Autofs uses the automount daemon to manage your mount points by only mounting them dynamically when they are accessed.

Autofs consults the master map configuration file /etc/auto.master to determine which mount points are defined. It then starts an automount process with the appropriate parameters for each mount point. Each line in the master map defines a mount point and a separate map file that defines the filesystems to be mounted under this mount point. For example, the /etc/auto.mnt file might define mount points in the /mnt directory; this relationship would be defined in the /etc/auto.master file.

Each entry in auto.master has three fields. The first field is the mount point. The second field is the location of the map file, and the third field is optional. The third field can contain information such as a timeout value.

For example, to mount the directory /project52 on the remote machine penguin.host.net at the mount point /mnt/myproject on your machine, add the following line to auto.master:

```
/mnt   /etc/auto.mnt --timeout 60
```

Add the following line to /etc/auto.mnt:

```
myproject  -rw,soft,intr,rsize=8192,wsize=8192   penguin.host.net:/project52
```

The first field in /etc/auto.mnt is the name of the /mnt subdirectory. This directory is created dynamically by automount. It should not actually exist on the client machine. The second field contains mount options such as rw for read and write access. The third field is the location of the NFS export including the hostname and directory.

> The directory /mnt must exist on the local filesystem. There should be no subdirectories to /mnt on the local filesystem.

Autofs is a service. To start the service, at a shell prompt, type the following commands:

```
service autofs restart
```

To view the active mount points, type the following command at a shell prompt:

```
service autofs status
```

If you modify the `/etc/auto.master` configuration file while autofs is running, you must tell the automount daemon(s) to reload by typing the following command at a shell prompt:

```
service autofs reload
```

To learn how to configure autofs to start at boot time, refer to Chapter 8, *Controlling Access to Services* for information on managing services.

10.3 Exporting NFS Filesystems

The `/etc/exports` file controls what filesystems you wish to export. Its format is as follows:

```
directory        hostname(options)
```

The `(options)` are not required. For example:

```
/mnt/export     speedy.redhat.com
```

would allow users from `speedy.redhat.com` to mount `/mnt/export` with the default read-only permissions, but:

```
/mnt/export     speedy.redhat.com(rw)
```

would allow users from `speedy.redhat.com` to mount `/mnt/export` with read-write priviledges.

CAUTION

Be careful with spaces in the `/etc/exports` file. If there are no spaces between the hostname and the options in parentheses, the options apply only to the hostname. If there is a space between the hostname and the options, the options apply to the rest of the world. For example, examine the following lines:

```
/mnt/export speedy.redhat.com(rw)
/mnt/export speedy.redhat.com (rw)
```

The first line grants users from `speedy.redhat.com` read-write access and denies all other users. The second line grants users from `speedy.redhat.com` read-only access (the default) and allows the rest of the world read-write access.

Refer to the *Official Red Hat Linux Reference Guide* for a list of options that can be specified in the `/etc/exports` file.

Each time you change `/etc/exports`, you must tell the NFS daemons to examine it for new information, or reload the configuration file:

```
/sbin/service nfs reload
```

10.3.1 Starting and Stopping the Server

On the server that is exporting NFS filesystems, the `nfs` service must be running.

View the status of the NFS daemon with the command

```
/sbin/service nfs status
```

Start the NFS daemon with the command

```
/sbin/service nfs start
```

Stop the NFS daemon with the command

```
/sbin/service nfs stop
```

To start the `nfs` service at boot time, use the command:

```
/sbin/chkconfig --level 345 nfs on
```

You can also use ntsysv or serviceconf to configure which services start at boot time. Refer to Chapter 8, *Controlling Access to Services* for details.

10.4 Additional Resources

This chapter discusses the basics of using NFS. For more detailed information, refer to the following resources.

10.4.1 Installed Documentation

- The man pages for `nfsd`, `mountd`, `exports`, `auto.master`, and `autofs` (in manual sections 5 and 8) — These man pages show the correct syntax for the NFS and autofs configuration files.

10.4.2 Related Books

- *Managing NFS and NIS Services* by Hal Stern; O'Reilly & Associates, Inc.

11 Samba

Samba uses the SMB protocol to share files and printers across a network connection. Operating systems that support this protocol include Microsoft Windows (through its Network Neighborhood), OS/2, and Linux.

11.1 Why Use Samba?

Samba is useful if you have a network of both Windows and Linux machines. Samba will allow files and printers to be shared by all the systems in your network. If you want to share files between Red Hat Linux machines only, refer to Chapter 10, *Network File System (NFS)*. If you want to share printers between Red Hat Linux machines only refer to Chapter 21, *Printer Configuration*.

11.2 Configuring Samba

Samba uses `/etc/samba/smb.conf` as its configuration file. If you change this configuration file, the changes will not take effect until you restart the Samba daemon with the command `service smb restart`.

The default configuration file (`smb.conf`) in Red Hat Linux 7.2 allows users to view their Linux home directories as a Samba share on the Windows machine after they log in using the same username and password. It also shares any printers configured for the Red Hat Linux system as Samba shared printers. In other words, you can attach a printer to your Red Hat Linux system and print to it from the Windows machines on your network.

To specify the Windows workgroup and description string, edit the following lines in your `smb.conf` file:

```
workgroup = WORKGROUPNAME
server string = BRIEF COMMENT ABOUT SERVER
```

Replace *WORKGROUPNAME* with the name of the Windows workgroup to which this machine should belong. The *BRIEF COMMENT ABOUT SERVER* is optional and will be the Windows comment about the Samba system.

To create a Samba share directory on your Linux system, add the following section to your `smb.conf` file (after modifying it to reflect your needs and your system):

```
[sharename]
comment = Insert a comment here
path = /home/share/
valid users = tfox carole
public = no
writable = yes
```

```
    printable = no
    create mask = 0765
```

The above example allows the users tfox and carole to read and write to the directory `/home/share`, on the Samba server, from a Samba client.

11.3 Connecting to a Samba Share

To connect to a Linux Samba share from a Microsoft Windows machine, use Network Neighborhood or Windows Explorer.

To connect to a Samba share from a Linux system, from a shell prompt, type the following command:

```
smbclient //hostname/sharename -U username
```

You will need to replace *hostname* with the hostname or IP address of the Samba server you want to connect to, *sharename* with the name of the shared directory you want to browse, and *username* with the Samba username for the system. Enter the correct password or press [Enter] if no password is required for the user.

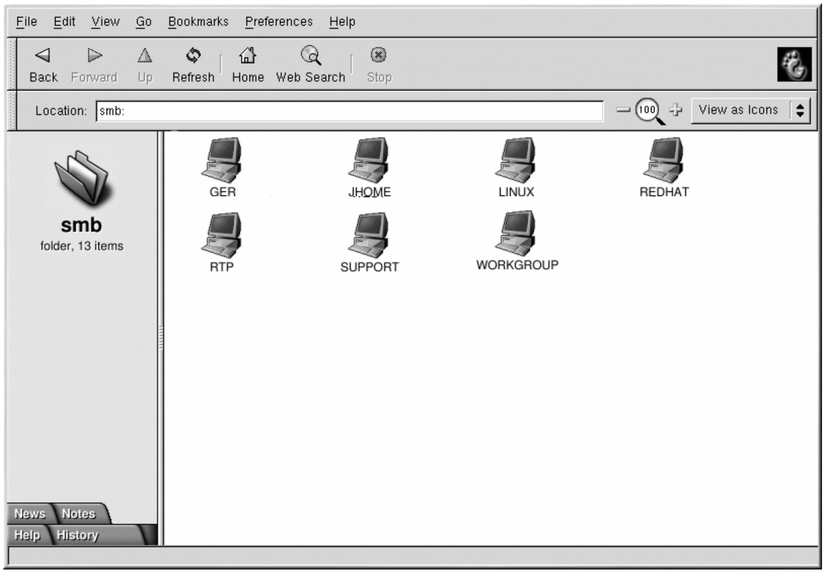
If you see the `smb:\>` prompt, you have successfully logged in. Once you are logged in, type **`help`** for a list of commands. If you wish to browse the contents of your home directory, replace *sharename* with your username. If the `-U` switch is not used, the username of the current user is passed to the Samba server.

To exit `smbclient`, type **`exit`** at the `smb:\>` prompt.

You can also use Nautilus to view available Samba shares on your network. On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **Applications** => **Nautilus** to open a Nautilus window. Type **`smb:`** in the **Location:** bar.

As shown in Figure 11–1, *SMB Browser in Nautilus*, you will see an icon for each available SMB workgroups on your network. To access one, double-click the icon for it.

Figure 11–1 SMB Browser in Nautilus



Location: smb:

View as Icons

If the SMB share you are connecting to requires a user name and password combination, you must specify them in the **Location:** bar using the following syntax (replace user, password, servername, and sharename with the appropriate values:

```
smb://user:password@servername/sharename/
```

11.4 Using Samba with Windows NT 4.0 and Windows 2000

The Microsoft SMB Protocol originally used plaintext passwords. However, Windows 2000 and Windows NT 4.0 with Service Pack 3 or higher require encrypted Samba passwords. To use Samba between a Red Hat Linux system and a system with Windows 2000 or Windows NT 4.0 Service Pack 3 or higher, you can either edit your Windows registry to use plaintext passwords or configure Samba on your Linux system to use encrypted passwords. If you choose to modify your registry, you must do so for all your Windows NT or 2000 machines — this is risky and may cause further conflicts.

To configure Samba on your Red Hat Linux system to use encrypted passwords, follow these steps:

1. Create a separate password file for Samba. To create one based on your existing `/etc/passwd` file, at a shell prompt, type the following command:

```
cat /etc/passwd | mksmbpasswd.sh > /etc/samba/smbpasswd
```

The `mksmbpasswd.sh` script is installed in your `/usr/bin` directory with the `samba` package.

2. Use the command `chmod 600 /etc/samba/smbpasswd` to change permissions on the Samba password file so that only root has read and write permissions.

3. The script does not copy user passwords to the new file. To set each Samba user's password, use the command `smbpasswd` *`username`* (replace *username* with each user's username). A Samba user account will not be active until a Samba password is set for it.

4. The next step is to enable encrypted passwords in the Samba configuration file. In the file `smb.conf`, uncomment the following lines:

```
encrypt password = yes
smb passwd file = /etc/samba/smbpasswd
```

5. To have the changes take effect, restart Samba by typing the command `service smb restart` at a shell prompt.

Additional Information

To read more about *Using Samba with Windows NT 4.0 and Windows 2000*, read `ENCRYPTION.txt`, `Win95.txt`, and `WinNT.txt` in the directory `/usr/share/doc/samba-`*`version-number`*`/docs/textdocs/` (replace *version-number* with the version-number of Samba that you have installed).

11.5 Additional Resources

For configuration options not covered here, please refer to the following resources.

11.5.1 Installed Documentation

- `smb.conf` man page — explains how to configure the Samba configuration file
- `smbd` man page — describes how the Samba daemon works

- `/usr/share/doc/samba-`*`version-number`*`/docs/` — HTML and text help files included with the `samba` package

11.5.2 Useful Websites

- http://www.samba.org — The Samba Web page contains useful documentation, information about mailing lists, and a list of GUI interfaces.

12 Dynamic Host Configuration Protocol (DHCP)

Dynamic Host Configuration Protocol (DHCP) is network protocol for automatically assigning TCP/IP information to client machines. Each DHCP client connect to the centrally-located DHCP server that returns the client's network configuration including IP address, gateway, and DNS servers.

12.1 Why Use DHCP?

DHCP is useful for fast delivery of client network configuration. When configuring the client system, the administrator can choose DHCP and not have to enter an IP address, netmask, gateway, or DNS servers. The client retrieves this information from the DHCP server. DHCP is also useful if an administrator wants to change the IP address of a large number of systems. Instead of reconfiguring all the systems, he can just edit one DHCP configuration file on the server for the new set of IP address. If the DNS servers for an organization changes, the changes are made on the DHCP server, not on all the DHCP clients. Once the network is restarted on the clients (or the clients are rebooted), the changes will take effect.

Furthermore, if a laptop or any type of mobile computer is configured for DHCP, it can be moved from office to office without having to reconfiguring it as long as each office has a DHCP server that allows it to connect to the network.

12.2 Configuring a DHCP Server

You can configure a DHCP server using the configuration file `/etc/dhcpd.conf`.

DHCP also uses the file `/var/lib/dhcp/dhcpd.leases` to store the client lease database. Refer to Section 12.2.2, *Lease Database* for more information.

12.2.1 Configuration File

The first step in configuring a DHCP server is to create the configuration file that stores the network information for the clients. Global options can be declared for all clients, or options can be declared for each client system.

The configuration file can contain any extra tabs or blank lines for easier formatting. The keywords are case-insensitive, and lines beginning with a hash mark (#) are considered comments.

There are two types of statements in the configuration file:

- Parameters — state how to perform a task, whether to perform a task, or what network configuration options to sent to the client.
- Declarations — describe the topology of the network, describe the clients, provide addresses for the clients, or apply a group of parameters to a group of declarations.

Some parameters must start with the `option` keyword. Options configure DHCP options; whereas, parameters configure values that are not optional or control how the DHCP server behaves.

Important

If you change the configuration file, the changes will not take effect until you restart the DHCP daemon with the command `service dhcpd restart`.

Parameters (including options) declared before a section enclosed in curly brackets ({ }) are considered global parameters. Global parameters apply to all the sections below it.

In Figure 12–1, *Example of a subnet declaration*, the `routers`, `subnet-mask`, `domain-name`, `domain-name-servers`, and `time-offset` options are used for both the apex and raleigh `host` statements.

As shown in Figure 12–1, *Example of a subnet declaration*, you can declare a `subnet`. You must include a `subnet` declaration for every subnet in your network. If you do not, the DHCP server will fail to start.

In this example, there are global options for every DHCP client in the subnet and a `range` declared. Clients are assigned an IP address within the `range`.

Figure 12–1 Example of a subnet declaration

```
subnet 192.168.1.0 netmask 255.255.255.0 {
        option routers                  192.168.1.254;
        option subnet-mask              255.255.255.0;

        option domain-name              "example.com";
        option domain-name-servers       192.168.1.1;

        option time-offset              -5;     # Eastern Standard Time

 range 192.168.1.10 192.168.1.100;
}
```

All subnets that share the same physical network should be declared within a `shared-network` declaration as shown in Figure 12–2, *Example of a shared-network declaration*. Parameters within the `shared-network` but outside the enclosed `subnet` declarations are considered global parameters. The name of the `shared-network` should be a descriptive title for the network such as test-lab to describe all the subnets in a test lab environment.

Figure 12–2 Example of a shared-network declaration

```
shared-network name {
    option domain-name              "test.redhat.com";
    option domain-name-servers      ns1.redhat.com, ns2.redhat.com;
    option routers                  192.168.1.254;
    more parameters for EXAMPLE shared-network
    subnet 192.168.1.0 netmask 255.255.255.0 {
        parameters for subnet
        range 192.168.1.1 192.168.1.31;
    }
    subnet 192.168.1.32 netmask 255.255.255.0 {
        parameters for subnet
        range 192.168.1.33 192.168.1.63;
    }
}
```

As demonstrated in Figure 12–3, *Example of a group declaration*, the `group` declaration can be used to apply global parameters to a group of declarations. You can group shared networks, subnets, hosts, or other groups.

Figure 12–3 Example of a group declaration

```
group {
   option routers                  192.168.1.254;
   option subnet-mask              255.255.255.0;

   option domain-name              "example.com";
   option domain-name-servers       192.168.1.1;

   option time-offset              -5;     # Eastern Standard Time

   host apex {
      option host-name "apex.example.com";
      hardware ethernet 00:A0:78:8E:9E:AA;
      fixed-address 192.168.1.4;
   }

   host raleigh {
      option host-name "raleigh.example.com";
      hardware ethernet 00:A1:DD:74:C3:F2;
      fixed-address 192.168.1.6;
   }
}
```

To configure a DHCP server that leases dynamic IP address to system within a subnet, modify Figure 12–4, *Example of the range parameter* with your values. It declares a default lease time, maximum lease time, and network configuration values for the clients. This example assigns IP address in the `range` 192.168.1.10 and 192.168.1.100 to client systems.

Figure 12–4 Example of the range parameter

```
default-lease-time 600;
max-lease-time 7200;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.1.255;
option routers 192.168.1.254;
option domain-name-servers 192.168.1.1, 192.168.1.2;
option domain-name "example.com";

subnet 192.168.1.0 netmask 255.255.255.0 {
   range 192.168.1.10 192.168.1.100;
}
```

To assign an IP address to a client based on the MAC address of the network interface card, use the `hardware ethernet` parameter within a `host` declaration. As demonstrated in Figure 12–5, *Example of a static IP address using DHCP*, the `host apex` declaration specifies that the network interface card with the MAC address 00:A0:78:8E:9E:AA should always been leased the IP address 192.168.1.4.

Notice that you can also use the optional parameter `host-name` to assign a host name to the client.

Figure 12–5 Example of a static IP address using DHCP

```
host apex {
   option host-name "apex.example.com";
   hardware ethernet 00:A0:78:8E:9E:AA;
   fixed-address 192.168.1.4;
}
```

Tip

You can use the sample configuration file in Red Hat Linux 7.2 as a starting point and then add your own custom configuration options to it. Copy it to its proper location with the command

```
cp
/usr/share/doc/dhcp-<version-number>/dhcpd.conf.sample
/etc/dhcpd.conf
```

(where <*version-number*> is the DHCP version you are using).

For a complete list of option statements and what they do, refer to the dhcp-options man page.

12.2.2 Lease Database

On the DHCP server, the file `/var/lib/dhcp/dhcpd.leases` stores the DHCP client lease database. This file should not be modified by hand. DHCP lease information for each recently assigned IP address is automatically stored in the lease database. The information includes the length of the lease, to whom the IP address has been assigned, the start and end dates for the lease, and the MAC address of the network interface card that was used to retreive the lease.

All times in the lease database are in Greenwich Mean Time (GMT), not local time.

The first time you start the DHCP service, it will fail unless the lease database exists. Use the command `touch /var/lib/dhcpd.leases` to create the file before starting the server for the first time. Once the file exists and the server has been started, do not try to create a new lease database file.

The lease database is recreated from time to time so that it is not too large. First, all known leases are saved in a temporary lease database. The `dhcpd.leases` file is renamed `dhcpd.leases~`, and the temporary lease database is written to `dhcpd.leases`.

The DHCP daemon could be killed or the system could crash after the lease database has been renamed to the backup file but before the new file has been written. If this happens, there is no `dhcpd.leases` file that is required to start the service. Do not create a new lease file if this occurs. If you do, all the old leases will be lost and cause many problems. The correct solution is to rename the `dhcpd.leases~` backup file to `dhcpd.leases` and then start the daemon.

12.2.3 Starting and Stopping the Server

Important

Before you start the DHCP server for the first time, it will fail unless there is an existing `dhcpd.leases` file. Use the command `touch /var/lib/dhcp/dhcpd.leases` to create the file before starting the service for the first time (and the first time only). Once the file exists, you do not have to perform this step again.

To start the DHCP service, use the command `/sbin/service dhcpd start`. To stop the DHCP server, use the command `/sbin/service dhcpd stop`. If you want the daemon to start automatically at boot time, see Chapter 8, *Controlling Access to Services* for information on how to manage services.

If you have more than more network interface attached to the system, but you only want the DHCP server to start on one of the interface, you can modify the init script to start only on that device. In the init script located at `/etc/rc.d/init.d/dhcpd`, there is a section that declares what happens when you try to start the service:

```
start() {
        # Start daemons.
        echo -n "Starting dhcpd: "
        daemon /usr/sbin/dhcpd
        RETVAL=$?
        echo
        [ $RETVAL -eq 0 ] && touch /var/lock/subsys/dhcpd
        return $RETVAL
}
```

Modify the line that starts the daemon to include the ethernet device on which you want the DHCP server to run:

```
daemon /usr/sbin/dhcpd eth0
```

This is useful if you have a firewall machine with two network cards. One network card can be configured as a DHCP client to retreive an IP address to the Internet. The other network card can be used as a DHCP server for the internal network behind the firewall.

12.2.4 DHCP Relay Agent

The DHCP Relay Agent (`dhcrelay`) allows you to relay DHCP and BOOTP requests from a subnet with no DHCP server on it to a one or more DHCP servers on other subnets.

When a DHCP client requests information, the DHCP Relay Agent forwards the request to the list of DHCP servers specified when the DHCP Relay Agent is started. When a DHCP server returns a reply, the reply is broadcast or unicast on the network that sent the original request.

To start the DHCP Relay Agent, use the `dhcrelay` command. It can be started with the following options:

Table 12–1 Graphical Update Agent Options

Argument	Description
-i	Names of the network interfaces to configure. If no interface is specified, all network interfaces will be configured, eliminating non-broadcast interfaces if it can.
-p	Port on which `dhcrelay` should listen. The DHCP Relay Agent transmits requests to the servers on this port and transmits responses to the clients on the port one greater than this port.
-d	Force `dhcrelay` to run in the foreground always.
-q	Disable printing the network configuration of `dhcrelay` on startup.

12.3 Configuring a DHCP Client

The first step for configuring a DHCP client is to make sure the kernel recognizes the network interface card. Most cards are recognized during the installation process, and the system is configured to use the correct kernel module for the card. If you install a card after installation, Kudzu should recognize it and prompt you to configure the cooresponding kernel module for it. Be sure to check the Red Hat Linux Hardware Compatibility List available at http://hardware.redhat.com. If the network card is not configured by the installation program or Kudzu and you know which kernel module to load for it, refer to Chapter 24, *Kernel Modules* for details on loading kernel modules.

To configure a DHCP client manually, you need to modify the `/etc/sysconfig/network` file to enable networking and the configuration file for each network device in the `/etc/sysconfig/network-scripts` directory. In this directory, each device should have a configuration file named `ifcfg-eth0` where `eth0` is the network device name.

The `/etc/sysconfig/network` file should contain the following line:

```
NETWORKING=yes
```

You might have more information in this file. You just need to make sure that the `NETWORKING` variable is set to `yes`.

The `/etc/sysconfig/network-scripts/ifcfg-eth0` file should contain the following lines:

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

You need a configuration file for each device that you want to configure to use DHCP.

If you prefer a graphical interface for configuring a DHCP client, refer to Chapter 6, *Network Configuration* for details on using Network Configurator to configure a network interface to use DHCP.

12.4 Additional Resources

For configuration options not covered here, please refer to the following resources.

12.4.1 Installed Documentation

- `dhcpd` man page — describes how the DHCP daemon works
- `dhcpd.conf` man page — explains how to configure the DHCP configuration file; includes some examples
- `dhcpd.leases` man page — explains how to configure the DHCP leases file; includes some examples
- `dhcp-options` man page — explains the syntax for declaring DHCP options in `dhcpd.conf`; includes some examples
- `dhcrelay` man page — explains the DHCP Relay Agent and its configuration options.

12.4.2 Useful Websites

- http://www.linuxdoc.org/HOWTO/mini/DHCP/index.html — *DHCP mini-HOWTO* from the Linux Documentation Project

13 Kerberos

Kerberos is a network authentication protocol created by MIT. It uses key cryptography instead of plain-text passwords. Kerberos offers a layer of system security and makes it harder for an unauthorized user to intercept users' passwords. For more information on how Kerberos works, refer to the *Official Red Hat Linux Reference Guide*.

13.1 Configuring a Kerberos 5 Server

When you're setting up Kerberos, install the server(s) first. If you need to set up slave servers, the details of setting up relationships between master and slave servers are covered in the *Kerberos 5 Installation Guide* (in the `/usr/share/doc/krb5-server-<version-number>` directory).

To install a Kerberos server:

1. Be sure that you have clock synchronization and DNS working on your server before installing Kerberos 5. Pay particular attention to time synchronization between the Kerberos server and its various clients. If the server and client clocks are different by more than five minutes (this default amount is configurable in Kerberos 5), Kerberos clients will not be able to authenticate to the server. This clock synchronization is necessary to prevent an attacker from using an old authenticator to masquerade as a valid user.

   You should set up a Network Time Protocol (NTP) compatible client/server network using Red Hat Linux, even if you aren't using Kerberos. Red Hat Linux 7.2 includes the `ntp` package for easy installation. See http://www.eecis.udel.edu/~ntp for additional information on NTP.

2. Install the `krb5-libs`, `krb5-server`, and `krb5-workstation` packages on the dedicated machine which will run your KDC. This machine needs to be secure — if possible, it should not run any services other than the KDC.

   If you would like to use a Graphical User Interface (GUI) utility to administrate Kerberos, you should also install the `gnome-kerberos` package. It contains `krb5`, a GUI tool for managing tickets, and `gkadmin`, a GUI tool for managing Kerberos realms.

3. Edit the `/etc/krb5.conf` and `/var/kerberos/krb5kdc/kdc.conf` configuration files to reflect your realm name and domain-to-realm mappings. A simple realm can be constructed by replacing instances of *EXAMPLE.COM* and *example.com* with your domain name (be sure to keep uppercase and lowercase names in the correct format) and by changing the KDC from *kerberos.example.com* to the name of your Kerberos server. By convention, all realm names are uppercase and all DNS hostnames and domain names are lowercase. For full details on the formats of these files, see their respective man pages.

4. Create the database using the `kdb5_util` utility from a shell prompt:

```
/usr/kerberos/sbin/kdb5_util create -s
```

The `create` command creates the database that will be used to store keys for your Kerberos realm. The `-s` switch forces creation of a **stash** file in which the master server key is stored. If no stash file is present from which to read the key, the Kerberos server (`krb5kdc`) will prompt the user for the master server password (which can be used to regenerate the key) every time it is started.

5. Edit the `/var/kerberos/krb5kdc/kadm5.acl` file. This file is used by `kadmind` to determine which principals have access to the Kerberos database and their level of access. Most organizations will be able to get by with a single line:

```
*/admin@EXAMPLE.COM  *
```

Most users will be represented in the database by a single principal (with a *NULL*, or empty, instance, such as *joe@EXAMPLE.COM*). With this configuration, users with a second principal with an instance of *admin* (for example, *joe/admin@EXAMPLE.COM*) will be able to wield full power over the realm's Kerberos database.

Once `kadmind` is started on the server, any user will be able to access its services by running `kadmin` or `gkadmin` on any of the clients or servers in the realm. However, only users listed in the `kadm5.acl` file will be able to modify the database in any way, except for changing their own passwords.

Note

The `kadmin` and `gkadmin` utilities communicate with the `kadmind` server over the network, and they use Kerberos to handle authentication. Of course, you need to create the first principal before you can connect to the server over the network to administer it. Create the first principal with the `kadmin.local` command, which is specifically designed to be used on the same host as the KDC and doesn't use Kerberos for authentication.

Type the following `kadmin.local` command at the KDC terminal to create the first principal:

```
/usr/kerberos/sbin/kadmin.local -q "addprinc username/admin"
```

6. Start Kerberos using the following commands:

```
/sbin/service krb5kdc start
/sbin/service kadmin start
/sbin/service krb524 start
```

7. Add principals for your users using the `addprinc` command with `kadmin` or using the **Principal** => **Add** menu option in `gkadmin`. `kadmin` (and `kadmin.local` on the master KDC) is a command line interface to the Kerberos administration system. As such, many commands are available after launching the `kadmin` program. Please see the `kadmin` man page for more information.

8. Verify that your server will issue tickets. First, run `kinit` to obtain a ticket and store it in a credential cache file. Then use `klist` to view the list of credentials in your cache and use `kdestroy` to destroy the cache and the credentials it contains.

Note

By default, `kinit` attempts to authenticate you using the login username of the account you used when you first logged into your system (not the Kerberos server). If that system username does not correspond to a principal in your Kerberos database, you will get an error message. If that happens, just give `kinit` the name of your principal as an argument on the command line (`kinit` *`principal`*).

Once you have completed the steps listed above, your Kerberos server should be up and running. Next, you will need to set up your Kerberos clients.

13.2 Configuring a Kerberos 5 Client

Setting up a Kerberos 5 client is less involved than setting up a server. At minimum, you should install the client packages and provide your clients with a valid `krb5.conf` configuration file. Kerberized versions of `rsh` and `rlogin` will also require some configuration changes.

1. Be sure that you have time synchronization in place between the Kerberos client and KDC. See Section 13.1, *Configuring a Kerberos 5 Server* for more information. In addition, DNS should be working properly on the Kerberos client before installing the Kerberos client programs.

2. Install the `krb5-libs` and `krb5-workstation` packages on all of the clients in your realm. You must supply your own version of `/etc/krb5.conf` for your client workstations; usually this can be the same `krb5.conf` used by the KDC.

3. Before a particular workstation in your realm can allow users to connect using kerberized `rsh` and `rlogin`, that workstation will need to have the `xinetd` package installed and have its own host principal in the Kerberos database. The `kshd` and `klogind` server programs will also need access to the keys for their service's principal.

   Using `kadmin`, add a host principal for the workstation. The instance in this case will be the hostname of the workstation. Because you'll never need to type the password for this principal again, and you probably don't want to bother with coming up with a good password, you can use the `-randkey` option to kadmin's `addprinc` command to create the principal and assign it a random key:

```
addprinc -randkey host/blah.example.com
```

Now that you have created the principal, you can extract the keys for the workstation by running `kadmin` *on the workstation itself*, and using the `ktadd` command within `kadmin`:

```
ktadd -k /etc/krb5.keytab host/blah.example.com
```

In order to use the kerberized versions of `rsh` and `rlogin`, you must enable `klogin`, `eklogin`, and `kshell`. [1]

4. Other kerberized network services will need to be started. To use kerberized `telnet`, you must enable `krb5-telnet`. [1]

   To provide FTP access, create and extract a key for a principal with a root of ftp, with the instance set to the hostname of the FTP server. Then enable `gssftp`. [1].

   The IMAP server included in the `imap` package will use GSS-API authentication using Kerberos 5 if it finds the proper key in `/etc/krb5.keytab`. The root for the principal should be `imap`. The CVS gserver uses a principal with a root of `cvs` and is otherwise identical to a `pserver`.

That should be all you need to do to set up a simple Kerberos realm.

13.3 Additional Resources

For details on how a Kerberos transaction is authenticated, refer to the *Official Red Hat Linux Reference Guide*.

13.3.1 Installed Documentation

- `/usr/share/doc/krb5-server-<version-number>` — The *Kerberos V5 Installation Guide* and the *Kerberos V5 System Administrator's Guide* in PostScript and HTML formats. You must have the `krb5-server` RPM package installed.
- `/usr/share/doc/krb5-workstation-<version-number>` — The *Kerberos V5 UNIX User's Guide* in PostScript and HTML formats. You must have the `krb5-workstation` RPM package installed.

[1] Refer to Chapter 8, *Controlling Access to Services* for details on enabling services.

13.3.2 Useful Websites

- http://web.mit.edu/kerberos/www — *Kerberos: The Network Authentication Protocol* webpage from MIT.
- http://www.nrl.navy.mil/CCS/people/kenh/kerberos-faq.html — The Kerberos Frequently Asked Questions (FAQ).
- ftp://athena-dist.mit.edu/pub/kerberos/doc/usenix.PS — The PostScript version of *Kerberos: An Authentication Service for Open Network Systems* by Jennifer G. Steiner, Clifford Neuman, and Jeffrey I. Schiller. This document is the original paper describing Kerberos.
- http://web.mit.edu/kerberos/www/dialogue.html — *Designing an Authentication System: a Dialogue in Four Scenes* originally by Bill Bryant in 1988, modified by Theodore Ts'o in 1997. This document is a conversation between two developers who are thinking through the creation of a Kerberos-style authentication system. The conversational style of the discussion make this a good starting place for people who are completely unfamiliar with Kerberos.
- http://www.ornl.gov/~jar/HowToKerb.html — *How to Kerberize your Site*.

14 Apache Configuration

Apache Configuration Tool requires the X Window System and root access. To start Apache Configuration Tool, use one of the following methods:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Apache Configuration**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Apache Configuration**.
- Type the command `apacheconf` at a shell prompt (for example, in an XTerm or GNOME-terminal).

Do Not Edit `httpd.conf`

Do not edit the `/etc/httpd/conf/httpd.conf` Apache configuration file if you wish to use this tool. Apache Configuration Tool generates this file after you save your changes and exit the program. If you want to add additional modules or configuration options that are not available in Apache Configuration Tool, you cannot use this tool.

Apache Configuration Tool allows you to configure the `/etc/httpd/conf/httpd.conf` configuration file for your Apache Web server. It does not use the old `srm.conf` or `access.conf` configuration files; leave them empty. Through the graphical interface, you can configure Apache directives such as virtual hosts, logging attributes, and maximum number of connections.

Only modules that are shipped with Red Hat Linux can be configured with Apache Configuration Tool. If additional modules are installed, they can not be configured using this tool.

The general steps for configuring the Apache Web Server using the Apache Configuration Tool are as following:

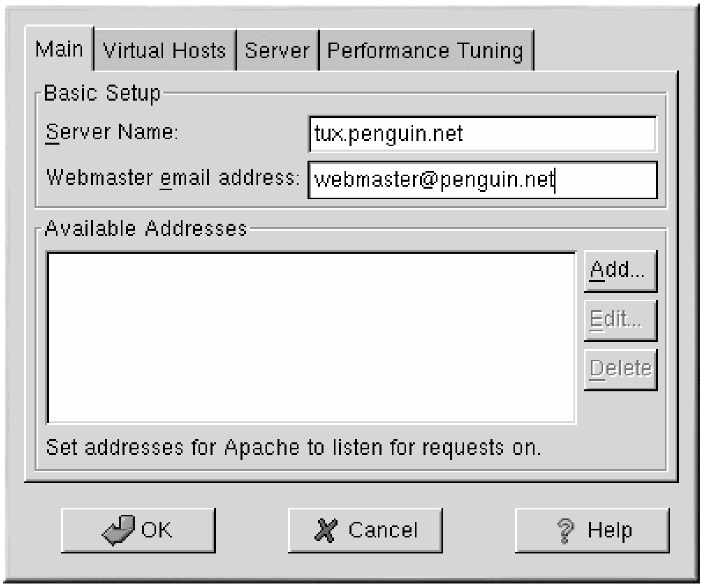
1. Configure the basic settings under the **Main** tab.
2. Click on the **Virtual Hosts** tab and configure the default settings.
3. Under the **Virtual Hosts** tab, configure the Default Virtual Host.
4. If you want to serve more than one URL or virtual host, add the additional virtual hosts.
5. Configure the server settings under the **Server** tab.
6. Configure the connections settings under the **Performance Tuning** tab.

7. Copy all necessary files to the DocumentRoot and cgi-bin directories, and save your settings in the Apache Configuration Tool.

14.1 Basic Settings

Use the **Main** tab to configure the basic server settings.

Figure 14–1 Basic Settings



Server Name: tux.penguin.net

Webmaster email address: webmaster@penguin.net

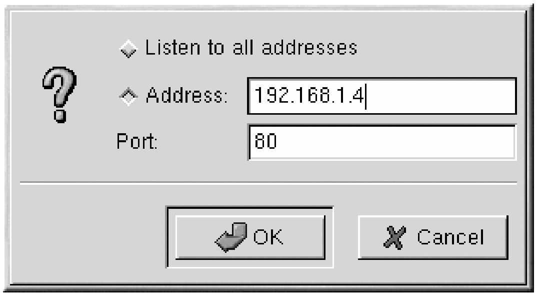
Set addresses for Apache to listen for requests on.

Enter a fully qualified domain name that you have the right to use in the **Server Name** text area. This option corresponds to the ServerName directive in `httpd.conf`. The ServerName directive sets the hostname of the Web server. It is used when creating redirection URLs. If you do not define a Server Name, Apache attempts to resolve it from the IP address of the system. The Server Name does not have to be the domain name resolved from the IP address of the server. For example, you might want to set the Server Name to www.your_domain.com when your server's real DNS name is actually foo.your_domain.com.

Enter the email address of the person who maintains the Web server in the **Webmaster email address** text area. This option corresponds to the ServerAdmin directive in `httpd.conf`. If you configure the server's error pages to contain an email address, this email address will be used so that users can report a problem by sending email to the server's administrator. The default value is root@localhost.

Use the **Available Addresses** area to define the ports on which Apache will accept incoming requests. This option corresponds to the Listen directive in `httpd.conf`. By default, Red Hat configures Apache to listen to ports 80 and 8080 for non-secure Web communications. Click the **Add** button to define additional ports on which to accept requests. A window as shown in Figure 14–2, *Available Addresses* will appear. Either choose the **Listen to all addresses** option to listen to all IP addresses on the defined port or specify a particular IP address over which the server will accept connections in the **Address** field. Only specify one IP address per port number. If you want to specify more than one IP address with the same port number, create an entry for each IP address. If at all possible, use an IP address instead of a domain name to prevent a DNS lookup failure. Refer to http://httpd.apache.org/docs/dns-caveats.html for more information about *Issues Regarding DNS and Apache*. Entering an asterisk (*) in the **Address** field is the same as choosing **Listen to all addresses**. Clicking the **Edit** button shows the same window as the **Add** button except with the fields populated for the selected entry. To delete an entry, select it and click the **Delete** button.

Figure 14–2 Available Addresses



○ Listen to all addresses

● Address: 192.168.1.4

Port: 80

Tip

If you set Apache to listen to a port under 1024, you must be root to start it. For port 1024 and above, `httpd` can be started as a regular user.

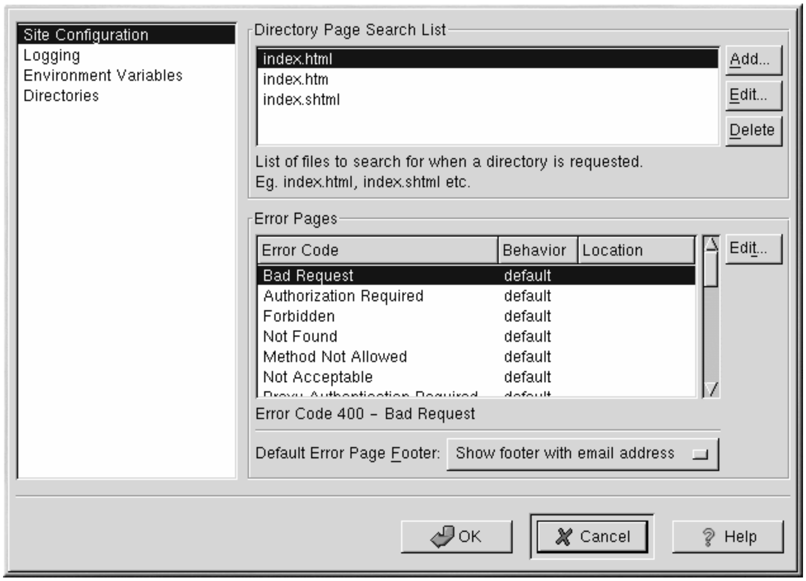
14.2 Default Settings

After defining the Server Name, Webmaster email address, and Available Addresses, click the **Virtual Hosts** tab and click the **Edit Default Settings** button. The window shown in Figure 14–3, *Site Configuration* will appear. Configure the default settings for your Web server in this window. If you add a virtual host, the settings you configure for the virtual host take precedence for that virtual host. For a directive not defined within the virtual host settings, the default value is used.

14.2.1 Site Configuration

The default values for the **Directory Page Search List** and **Error Pages** will work for most servers. If you are unsure of these settings, do not modify them.

Figure 14–3 Site Configuration



Directory Page Search List

- index.html
- index.htm
- index.shtml

List of files to search for when a directory is requested.
Eg. index.html, index.shtml etc.

Error Pages

Error Code	Behavior	Location
Bad Request	default	
Authorization Required	default	
Forbidden	default	
Not Found	default	
Method Not Allowed	default	
Not Acceptable	default	
Proxy Authentication Required	default	

Error Code 400 – Bad Request

Default Error Page Footer: Show footer with email address

The entries listed in the **Directory Page Search List** define the DirectoryIndex directive. The `DirectoryIndex` is the default page served by the server when a user requests an index of a directory by specifying a forward slash (/) at the end of the directory name.

For example, when a user requests the page http://*your_domain*/*this_directory*/, they are going to get either the `DirectoryIndex` page if it exists, or a server-generated directory list. The server will try to find one of the files listed in the DirectoryIndex directive and will return the first one it finds. If it doesn't find any of these files and if `Options Indexes` is set for that directory, the server will generate and return a list, in HTML format, of the subdirectories and files in the directory.

Use the **Error Code** section to configure Apache to redirect the client to a local or external URL if the event of a problem or error. This option corresponds to the ErrorDocument directive. If a problem or error occurs when a client tries to connect to the Apache Web server, the default action is to display the short error message shown in the **Error Code** column. To override this default configuration, select the error code and click the **Edit** button. Choose **Default** to display the default short error message. Choose **URL** to redirect the client to an external URL and enter a complete URL including the http://

in the **Location** field. Choose **File** to redirect the client to an internal URL and enter a file under the Document Root for the Web server. The location must begin the a slash (/) and be relative to the Document Root.

For example, to redirect a 404 Not Found error code to a Web page that you created in a file called `404.html`, copy `404.html` to *`DocumentRoot`*`/errors/404.html`. In this case, *DocumentRoot* is the Document Root directory that you have defined (the default is `/var/www/html`). Then, choose **File** as the Behavior for **404 - Not Found** error code and enter `/errors/404.html` as the **Location**.

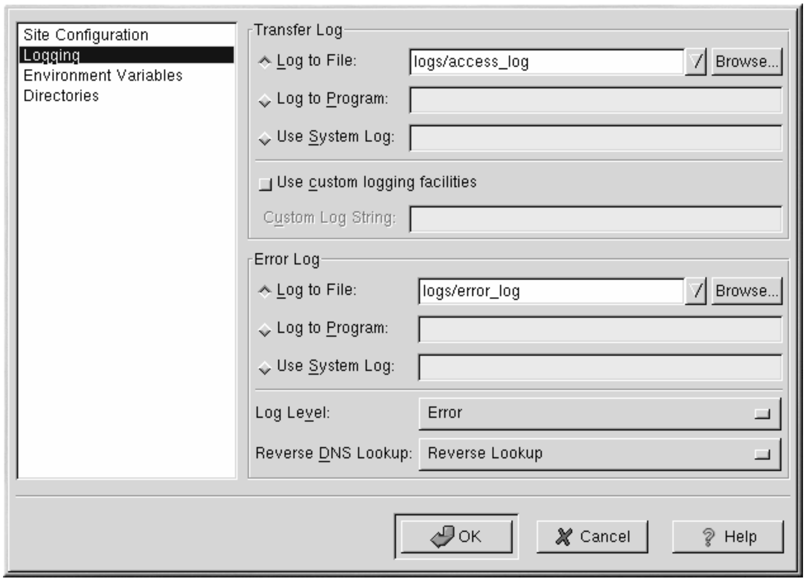
From the **Default Error Page Footer** menu, you can choose one of the following options:

- **Show footer with email address** — Display the default Apache footer at the bottom of all error pages along with the email address of the website maintainer specified by the ServerAdmin directive. Refer to *General Options* in Section 14.3.1 for information about configuring the ServerAdmin directive.
- **Show footer** — Display just the default Apache footer at the bottom of error pages.
- **No footer** — Do not display a footer at the bottom of error pages.

14.2.2 Logging

By default, Apache writes the transfer log to the file `/var/log/httpd/access_log` and the error log to the file `/var/log/httpd/error_log`.

Figure 14–4 Logging



Transfer Log

Log to File: logs/access_log

Log to Program:

Use System Log:

□ Use custom logging facilities

Custom Log String:

Error Log

Log to File: logs/error_log

Log to Program:

Use System Log:

Log Level: Error

Reverse DNS Lookup: Reverse Lookup

The transfer log contains a list of all attempts to access the Web server. It records the IP address of the client that is attempting to connect, the date and time of the attempt, and the file on the Web server that it is trying to retrieve. Enter the name of the path and file in which to store this information. If the path and filename does not start with a slash (/), the path is relative to the server root directory as configured. This option corresponds to the TransferLog directive.

You can configure a custom log format by checking **Use custom logging facilities** and entering a custom log string in the **Custom Log String** field. This configures the LogFormat directive. Refer to http://httpd.apache.org/docs/mod/mod_log_config.html#formats for details on the format of this directive.

The error log contains a list of any server errors that occur. Enter the name of the path and file in which to store this information. If the path and filename does not start with a slash (/), the path is relative to the server root directory as configured. This option corresponds to the ErrorLog directive.

Use the **Log Level** menu to set how verbose the error messages in the error logs will be. It can be set (from least verbose to most verbose) to emerg, alert, crit, error, warn, notice, info or debug. This option corresponds to the LogLevel directive.

The value chosen with the **Reverse DNS Lookup** menu defines the HostnameLookups directive. Choosing **No Reverse Lookup** sets the value to off. Choosing **Reverse Lookup** sets the value to on. Choosing **Double Reverse Lookup** sets the value to double.

If you choose **Reverse Lookup**, your server will automatically resolve the IP address for each connection which requests a document from your Web server. Resolving the IP address means that your server will make one or more connections to the DNS in order to find out the hostname that corresponds to a particular IP address.

If you choose **Double Reverse Lookup**, your server will perform a double-reverse DNS. In other words, after a reverse lookup is performed, a forward lookup is performed on the result. At least one of the IP addresses in the forward lookup must match the address from the first reverse lookup.

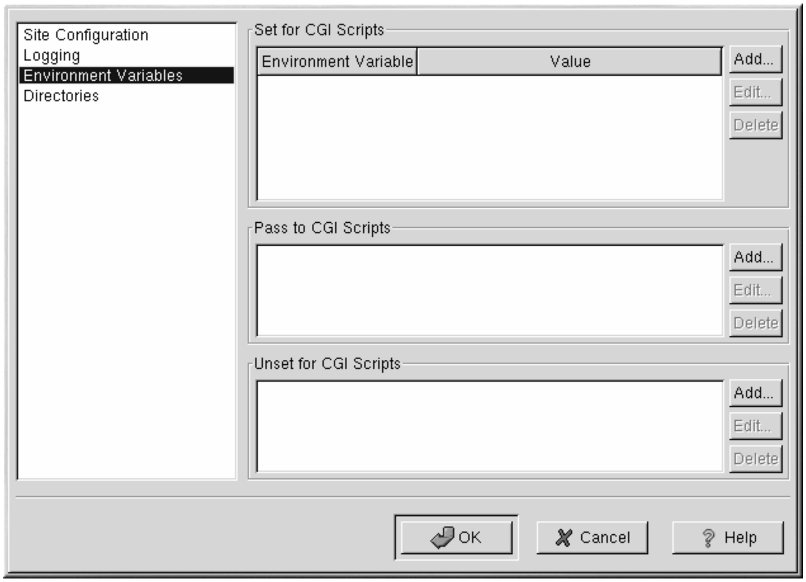
Generally, you should leave this option set to **No Reverse Lookup**, because the DNS requests add a load to your server and may slow it down. If your server is busy, the effects of trying to perform these reverse lookups or double reverse lookups may be quite noticeable.

Reverse lookups and double reverse lookups are also an issue for the Internet as a whole. All of the individual connections made to look up each hostname add up. Therefore, for your own Web server's benefit, as well as for the Internet's benefit, you should leave this option set to **No Reverse Lookup**.

14.2.3 Environment Variables

Apache can use the `mod_env` module to configure the environment variables which are passed to CGI scripts and SSI pages. Use the **Environment Variables** page to configure the directives for this Apache module.

Figure 14–5 Environment Variables



Use the **Set for CGI Scripts** section to set an environment variable that is passed to CGI scripts and SSI pages. For example, to set the environment variable `MAXNUM` to `50`, click the **Add** button inside the **Set for CGI Script** section as shown in Section 14.2.3, *Environment Variables* and type **`MAXNUM`** in the **Environment Variable** text field and **`50`** in the **Value to set** text field. Click **OK**. The **Set for CGI Scripts** section configures the SetEnv directive.

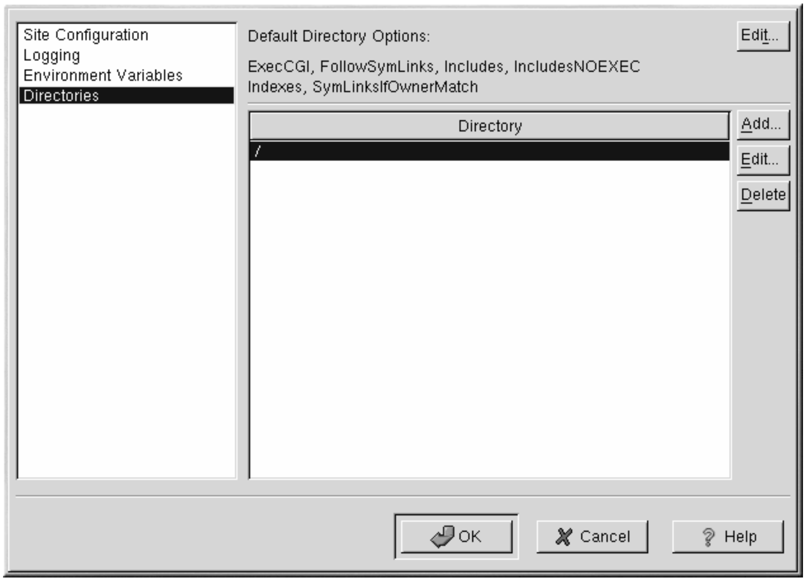
Use the **Pass to CGI Scripts** section to pass the value of an environment variable when Apache was first started to CGI scripts. To see this environment variable, type the command `env` at a shell prompt. Click the **Add** button inside the **Pass to CGI Scripts** section and enter the name of the environment variable in the resulting dialog box. Click **OK**. The **Pass to CGI Scripts** section configures the PassEnv directive.

If you want to remove an environment variable so that the value is not passed to CGI scripts and SSI pages, use the **Unset for CGI Scripts** section. Click **Add** in the **Unset for CGI Scripts** section, and enter the name of the environment variable to unset. This corresponds to the UnsetEnv directive.

14.2.4 Directories

Use the **Directories** page to configure options for specific directories. This corresponds to the <Directory> directive.

Figure 14–6 Directories



Default Directory Options:

ExecCGI, FollowSymLinks, Includes, IncludesNOEXEC
Indexes, SymLinksIfOwnerMatch

Directory
/

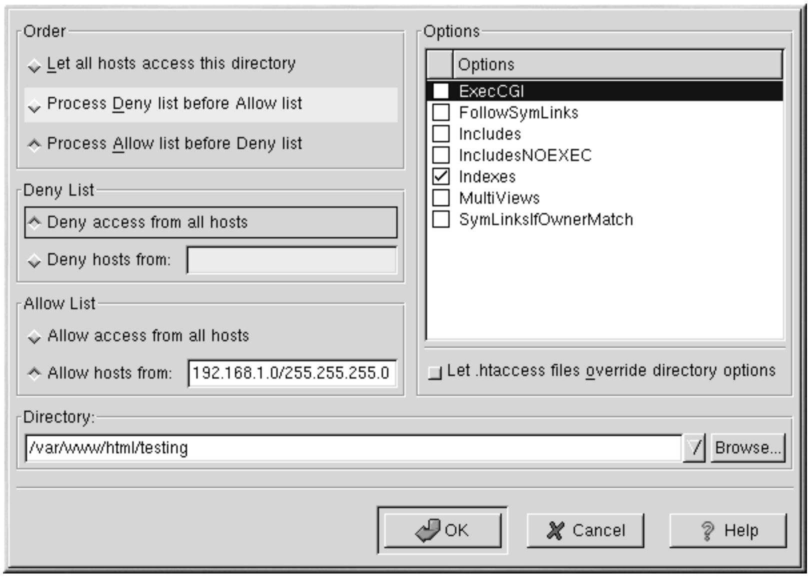
Click the **Edit** button in the top right-hand corner to configure the **Default Directory Options** for all directories that are not specified in the **Directory** list below it. The options that you choose are listed as the Options directive within the <Directory> directive. You can configure the following options:

- **ExecCGI** — Allow execution of CGI scripts. CGI scripts are not executed if this option is not chosen.
- **FollowSymLinks** — Allow symbolic links to be followed.
- **Includes** — Allow server-side includes.
- **IncludesNOEXEC** — Allow server-side includes, but disable the `#exec` and `#include` commands in CGI scripts.
- **Indexes** — Display a formatted list of the directory's contents, if no DirectoryIndex (such as `index.html`) exists in the requested directory.
- **Multiview** — Support content-negotiated multiviews; this option is disabled by default.
- **SymLinksIfOwnerMatch** — Only follow symbolic links if the target file or directory has the same owner as the link.

To specify options for specific directories, click the **Add** button beside the **Directory** list box. The window shown in Figure 14–7, *Directory Settings* appears. Enter the directory to configure in the **Directory** text field at the bottom of the window. Select the options in the right-hand list, and configure the Order directive with the left-hand side options. The Order directive controls the order in which allow and deny directives are evaluated. In the **Allow hosts from** and **Deny hosts from** text field, you can specify one of the following:

- Allow all hosts — Type **`all`** to allow access to all hosts.
- Partial domain name — Allow all hosts whose names match or end with the specified string.
- Full IP address — Allow access to a specific IP address.
- A subnet — Such as **`192.168.1.0/255.255.255.0`**
- A network CIDR specification — such as **`10.3.0.0/16`**

Figure 14–7 Directory Settings



Order
○ Let all hosts access this directory
○ Process Deny list before Allow list
● Process Allow list before Deny list

Deny List
● Deny access from all hosts
○ Deny hosts from:

Allow List
○ Allow access from all hosts
● Allow hosts from: 192.168.1.0/255.255.255.0

Options
☐ ExecCGI
☐ FollowSymLinks
☐ Includes
☐ IncludesNOEXEC
☑ Indexes
☐ MultiViews
☐ SymLinksIfOwnerMatch

☐ Let .htaccess files override directory options

Directory: /var/www/html/testing

If you check the **Let .htaccess files override directory options**, the configuration directives in the `.htaccess` file take precedence.

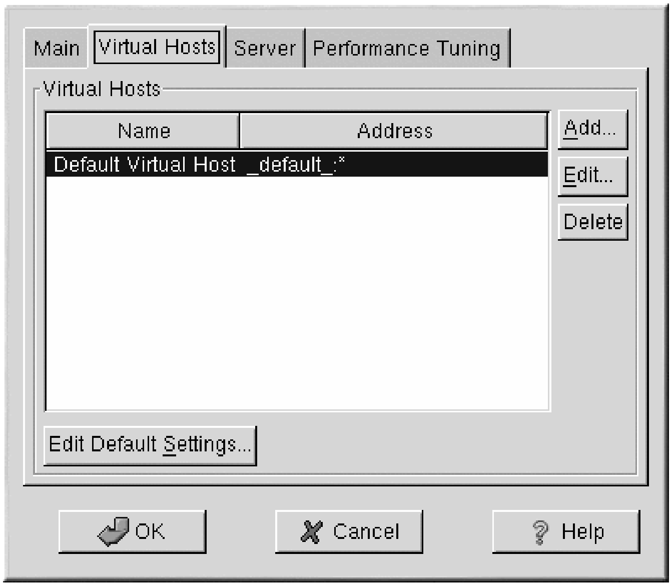
14.3 Virtual Hosts Settings

You can use Apache Configuration Tool to configure virtual hosts. Virtual hosts allow you to run different servers for different IP addresses, different host names, or different ports on the same machine. For example, you can run the website for http://www.your_domain.com and http://www.your_second_domain.com on the same Apache server using virtual hosts. This option corresponds to the <VirtualHost> directive for the default virtual host and IP based virtual hosts. It corresponds to the <NameVirtualHost> directive for a name based virtual host.

The Apache directives set for a virtual host only apply to that particular virtual host. If a directive is set server-wide using the **Edit Default Settings** button and not defined within the virtual host settings, the default setting is used. For example, you can define a **Webmaster email address** in the **Main** tab and not define individual email addresses for each virtual host.

Apache Configuration Tool includes a default virtual host as shown in Figure 14–8, *Virtual Hosts*. Refer to **Default Virtual Host** in Section 14.3.1 for details about the default virtual host.

Figure 14–8 Virtual Hosts



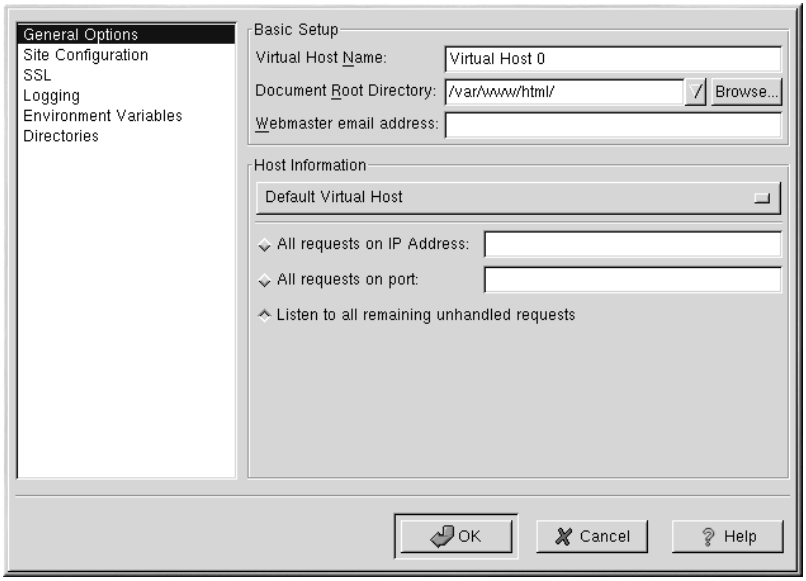
Name	Address
Default Virtual Host	_default_:*

The Apache documentation on your machine or on the Web at http://www.apache.org/docs/vhosts/ provides more information about virtual hosts.

14.3.1 Adding and Editing a Virtual Host

To add a virtual host, click the **Virtual Hosts** tab and then click the **Add** button. The window as shown in Figure 14–9, *Virtual Hosts Configuration* appears. You can also edit a virtual host by selecting it in the list and clicking the **Edit** button.

Figure 14–9 Virtual Hosts Configuration



Basic Setup

Virtual Host Name: Virtual Host 0

Document Root Directory: /var/www/html/

Webmaster email address:

Host Information

Default Virtual Host

◇ All requests on IP Address:

◇ All requests on port:

◆ Listen to all remaining unhandled requests

General Options

The **General Options** settings only apply to the virtual host that you are configuring. Set the name of the Virtual Host in the **Virtual Host Name** text area. This name is used by Apache Configuration Tool to distinguish between virtual hosts.

Set the **Document Root Directory** value to the directory that contains the root document (such as index.html) for the virtual host. This option corresponds to the DocumentRoot directive within the VirtualHost directive. Before Red Hat Linux 7.0, Apache provided with Red Hat Linux used `/home/httpd/html` as the `DocumentRoot`. In Red Hat Linux 7.2, however, the default `DocumentRoot` is `/var/www/html`.

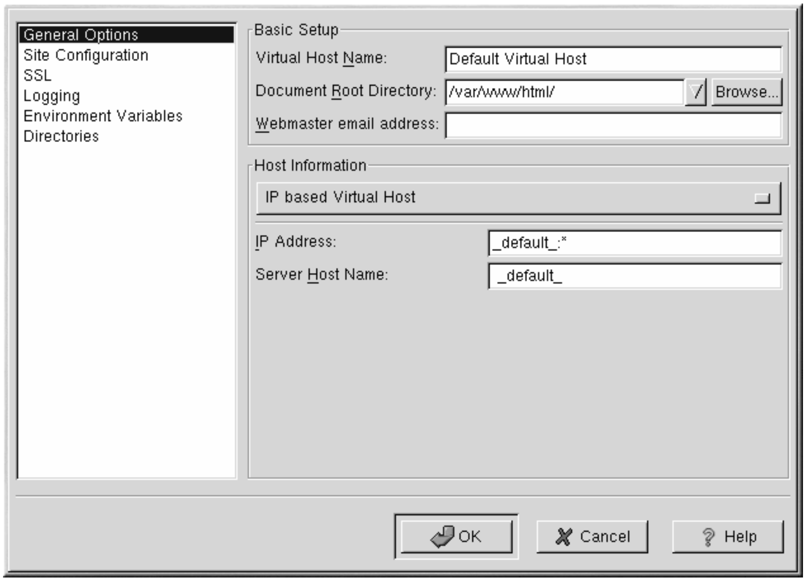
The **Webmaster email address** corresponds to the ServerAdmin directive within the VirtualHost directive. This email address is used in the footer of error pages if you choose to show a footer with an email address on the error pages.

In the **Host Information** section, choose **Default Virtual Host**, **IP based Virtual Host**, or **Name based Virtual Host**.

Default Virtual Host

If you choose **Default Virtual Host**, Figure 14–10, *Default Virtual Hosts* appears. You should only configure one default virtual host. The default virtual host settings are used when the requested IP address is not explicitly listed in another virtual host. If there is no default virtual host defined, the main server settings are used.

Figure 14–10 Default Virtual Hosts



Basic Setup

Virtual Host Name: Default Virtual Host

Document Root Directory: /var/www/html/

Webmaster email address:

Host Information

IP based Virtual Host

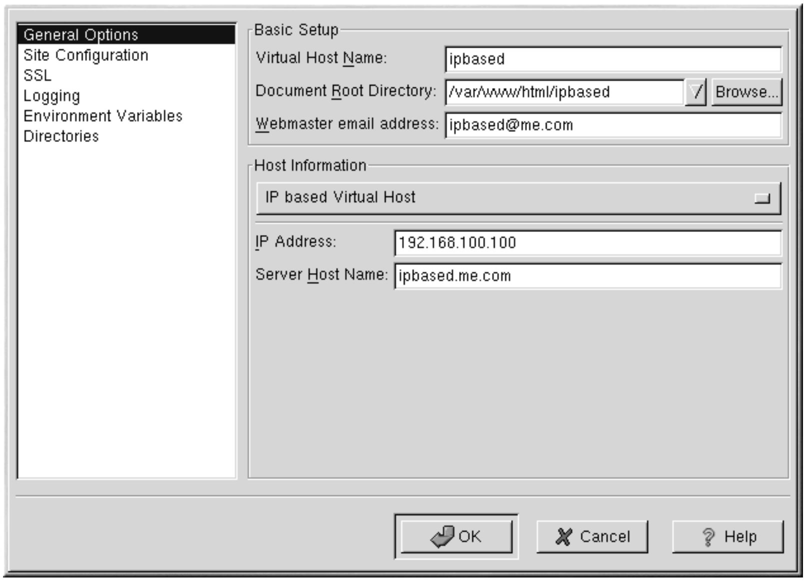
IP Address: _default_:*

Server Host Name: _default_

IP based Virtual Host

If you choose **IP based Virtual Host**, Figure 14–11, *IP Based Virtual Hosts* appears to configure the <VirtualHost> directive based on the IP address of the server. Specify this IP address in the **IP address** field. To specify more than one IP address, separate each IP address with spaces. To specify a port, use the syntax *IP Address:Port*. Use :* to configure all ports for the IP address. Specify the host name for the virtual host in the **Server Host Name** field.

Figure 14–11 IP Based Virtual Hosts



Basic Setup

Virtual Host Name: ipbased

Document Root Directory: /var/www/html/ipbased

Webmaster email address: ipbased@me.com

Host Information

IP based Virtual Host

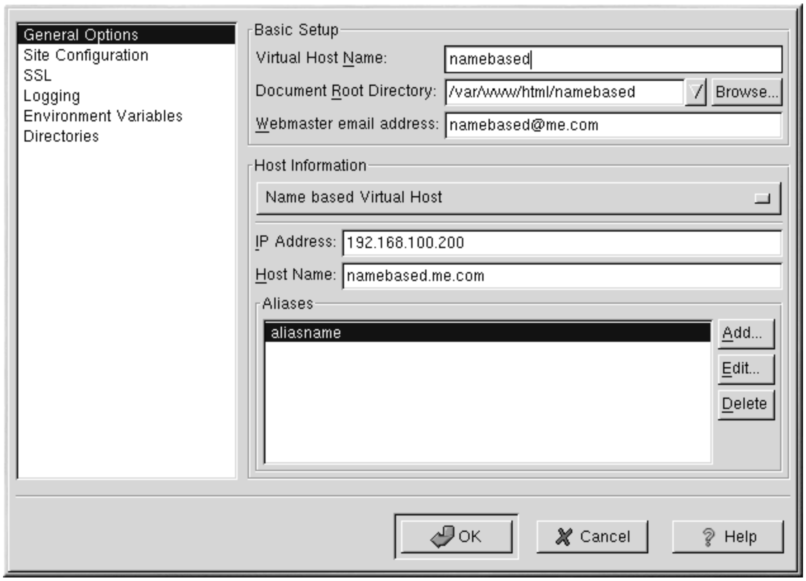
IP Address: 192.168.100.100

Server Host Name: ipbased.me.com

Name based Virtual Host

If you choose **Name based Virtual Host**, Figure 14–12, *Name Based Virtual Hosts* appears to configure the NameVirtualHost Directive based on the host name of the server. Specify the IP address in the **IP address** field. To specify more than one IP address, separate each IP address with spaces. To specify a port, use the syntax *IP Address:Port*. Use :* to configure all ports for the IP address. Specify the host name for the virtual host in the **Server Host Name** field. In the **Aliases** section, click **Add** to add a host name alias. Adding an alias here adds a ServerAlias directive within the NameVirtualHost Directive.

Figure 14–12 Name Based Virtual Hosts



Basic Setup

Virtual Host Name: namebased

Document Root Directory: /var/www/html/namebased

Webmaster email address: namebased@me.com

Host Information

Name based Virtual Host

IP Address: 192.168.100.200

Host Name: namebased.me.com

Aliases

aliasname

SSL

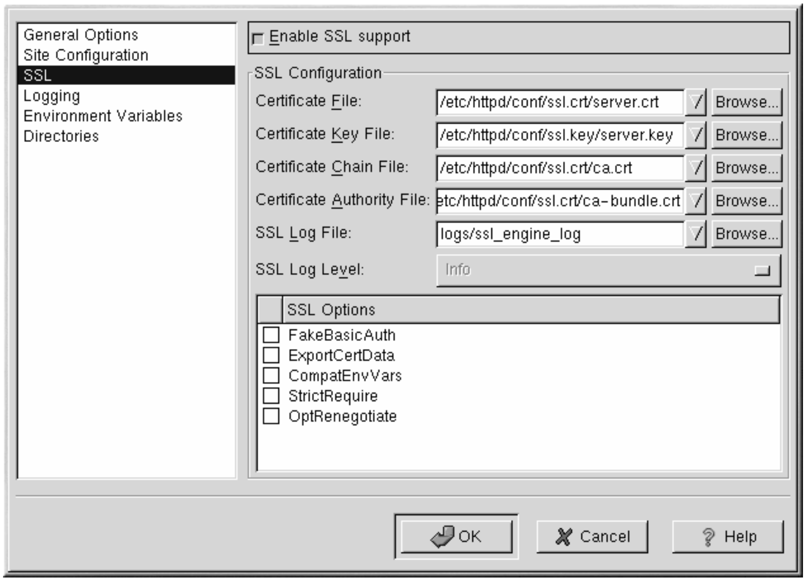
Note

You can not use name based virtual hosts with SSL, because the SSL handshake (when the browser accepts the secure Web server's certificate) occurs before the HTTP request which identifies the appropriate name based virtual host. If you want to use name-based virtual hosts, they will only work with your non-secure Web server.

If an Apache server is not configured with SSL support, communications between an Apache server and its clients are not encrypted. This is appropriate for websites without personal or confidential information. For example, an open source website that distributes open source software and documentation has no need for secure communications. However, an ecommerce website that requires credit card information should use the Apache SSL support to encrypt its communications. Enabling Apache SSL support enables the use of the `mod_ssl` security module. To enable it through Apache Configuration Tool you must allow access through port 443 under the **Main** tab => Available Addresses. Refer to Section 14.1, *Basic Settings* for details. Then, select the virtual host name in the

Virtual Hosts tab, click the **Edit** button, choose **SSL** from the left-hand menu, and check the **Enable SSL Support** option as shown in Figure 14–13, *SSL Support*. The **SSL Configuration** section is pre-configured with the dummy digital certificate. The digital certificate provides authentication for your secure Web server and identifies the secure server to client Web browsers. You must purchase your own digital certificate. Do not use the dummy one provided in Red Hat Linux for your website. For details on purchasing a CA-approved digital certificate, refer to the Chapter 15, *Apache Secure Server Configuration*.

Figure 14–13 SSL Support



□ Enable SSL support

SSL Configuration

Certificate File: /etc/httpd/conf/ssl.crt/server.crt

Certificate Key File: /etc/httpd/conf/ssl.key/server.key

Certificate Chain File: /etc/httpd/conf/ssl.crt/ca.crt

Certificate Authority File: etc/httpd/conf/ssl.crt/ca-bundle.crt

SSL Log File: logs/ssl_engine_log

SSL Log Level: Info

SSL Options

□ FakeBasicAuth
□ ExportCertData
□ CompatEnvVars
□ StrictRequire
□ OptRenegotiate

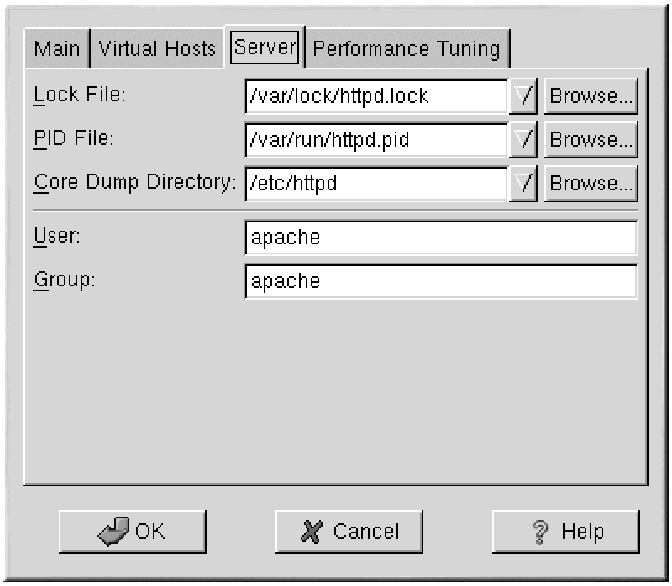
Additional Virtual Host Options

The **Site Configuration**, **Environment Variables**, and **Directories** options for the virtual hosts are the same directives that you set when you clicked the **Edit Default Settings** button, except the options set here are for the individual virtual hosts that you are configuring. Refer to Section 14.2, *Default Settings* for details on these options.

14.4 Server Settings

The **Server** tab allows you to configure basic server settings. The default settings for these options are appropriate for most situations.

Figure 14–14 Server Configuration



Lock File: /var/lock/httpd.lock

PID File: /var/run/httpd.pid

Core Dump Directory: /etc/httpd

User: apache

Group: apache

The **Lock File** value corresponds to the LockFile directive. This directive sets the path to the lockfile used when Apache is compiled with either USE_FCNTL_SERIALIZED_ACCEPT or USE_FLOCK_SERIALIZED_ACCEPT. It must be stored on the local disk. IT should be left to the default value unless the `logs` directory is located on an NFS share. If this is the case, the default value should be changed to a location on the local disk and to a directory that is readable only by root.

The **PID File** value corresponds to the PidFile directive. This directive sets the file in which the server records its process ID (pid). This file should only be readable by root. In most cases, it should be left to the default value.

The **Core Dump Directory** value corresponds to the CoreDumpDirectory directive. Apache tries to switch to this directory before dumping core. The default value is the ServerRoot. However, if the user that the server runs as can not write to this directory, the core dump can not be written. Change this value to a directory writable by the user the server runs as, if you want to write the core dumps to disk for debugging purposes.

The **User** value corresponds to the User directive. It sets the userid used by the server to answer requests. This user's settings determine the server's access. Any files inaccessible to this user will also be inaccessible to your website's visitors. The default for User is apache.

The User should only have privileges so that it can access files which are supposed to be visible to the outside world. The User is also the owner of any CGI processes spawned by the server. The User should not be allowed to execute any code which is not intended to be in response to HTTP requests.

WARNING

Unless you know exactly what you are doing, do not set the User to root. Using root as the User will create large security holes for your Web server.

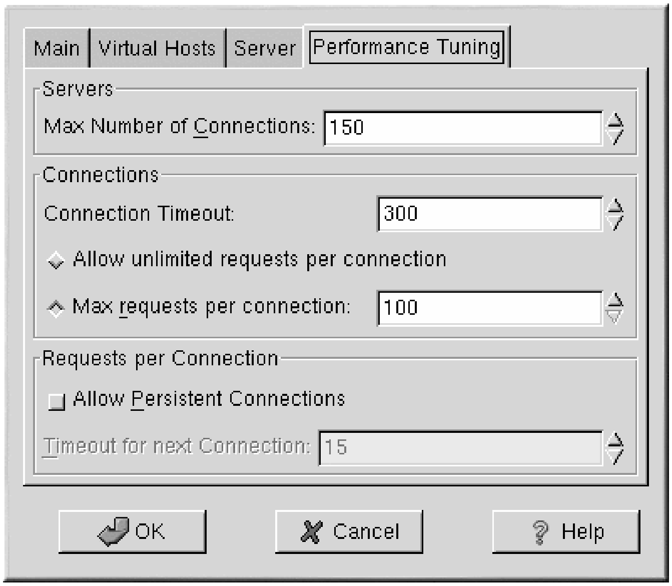
The parent `httpd` process first runs as root during normal operations, but is then immediately handed off to the apache user. The server must start as root because it needs to bind to a port below 1024. Ports below 1024 are reserved for system use, so they can not be used by anyone but root. Once the server has attached itself to its port, however, it hands the process off to the apache user before it accepts any connection requests.

The **Group** value corresponds to the Group directive. The Group directive is similar to the User. The Group sets the group under which the server will answer requests. The default Group is also apache.

14.5 Performance Tuning

Click on the **Performance Tuning** tab to configure the maximum number of child server processes you want and to configure the Apache options for client connections. The default settings for these options are appropriate for most situations. Altering these settings may affect the overall performance of your Web server.

Figure 14–15 Performance Tuning



Servers

Max Number of Connections: 150

Connections

Connection Timeout: 300

○ Allow unlimited requests per connection

● Max requests per connection: 100

Requests per Connection

□ Allow Persistent Connections

Timeout for next Connection: 15

Set **Max Number of Connections** to the maximum number of simultaneous client requests that the server will handle. For each connection, a child `httpd` process is created. After this maximum number of process is reached, no one else will be able to connect to the Web server until a child server process is freed. You can not set this value to higher than 256 without recompiling Apache. This option corresponds to the MaxClients directive.

Connection Timeout defines, in seconds, the amount of time that your server will wait for receipts and transmissions during communications. Specifically, Connection Timeout defines how long your server will wait to receive a GET request, how long it will wait to receive TCP packets on a POST or PUT request and how long it will wait between ACKs responding to TCP packets. By default, Connection Timeout is set to 300 seconds, which is appropriate for most situations. This option corresponds to the TimeOut directive.

Set the **Max requests per connection** to the maximum number of requests allowed per persistent connection. The default value is 100, which should be appropriate for most situations. This option corresponds to the MaxRequestsPerChild directive.

If you check the **Allow unlimited requests per connection** option, the MaxKeepAliveRequests directive to 0, and unlimited requests are allowed.

If you uncheck the **Allow Persistent Connections** option, the KeepAlive directive is set to false. If you check it, the KeepAlive directive is set to true, and the KeepAliveTimeout directive is set to the number that is selected as the **Timeout for next Connection** value. This directive sets the number of seconds your server will wait for a subsequent request, after a request has been served, before it closes the connection. Once a request has been received, the **Connection Timeout** value applies instead.

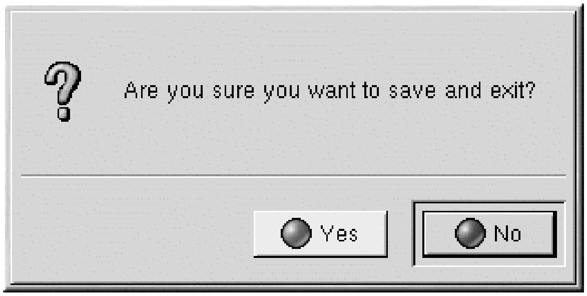
Setting the **Persistent Connections** to a high value may cause a server to slow down, depending on how many users are trying to connect to it. The higher the number, the more server processes waiting for another connection from the last client that connected to it.

14.6 Saving Your Settings

If you do not want to save your Apache configuration settings, click the **Cancel** button in the bottom right corner of the Apache Configuration Tool window. You will be prompted to confirm this decision. If you click **Yes** to confirm this choice, your settings will not be saved.

If you want to save your Apache configuration settings, click the **OK** button in the bottom right corner of the Apache Configuration Tool window. The dialog window shown in Figure 14–16, *Save and Exit* will appear. If you answer **Yes**, your settings will be saved in `/etc/httpd/conf/httpd.conf`. Remember that your original configuration file will be overwritten.

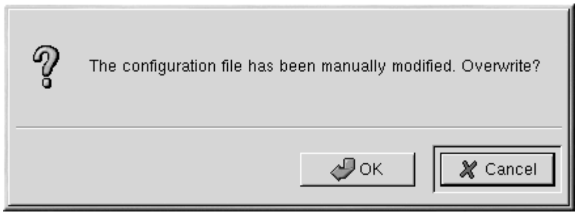
Figure 14–16 Save and Exit



Are you sure you want to save and exit?

If this is the first time that you have used Apache Configuration Tool, you will see the dialog window shown in Figure 14–17, *Configuration File Manually Modified*, warning you that the configuration file has been manually modified. If Apache Configuration Tool detects that the `httpd.conf` configuration file has been manually modified, it will save the manually modified file as `/etc/httpd/conf/httpd.conf.bak`.

Figure 14–17 Configuration File Manually Modified



The configuration file has been manually modified. Overwrite?

Restart Daemon

After saving your settings, you must restart the Apache daemon with the command `service httpd restart`. You must be logged in as root to execute this command.

14.7 Additional Resources

To learn more about Apache, refer to the following resources.

14.7.1 Installed Documentation

- Apache documentation — If you have the `apache-manual` package installed and the Apache Web server daemon (`httpd`) running, you can view the Apache documentation. Open a Web browser, and go to the URL http://localhost on the server that is running Apache. Then, click the **Documentation** link.

14.7.2 Useful Websites

- http://www.apache.org — *The Apache Software Foundation*
- http://httpd.apache.org/docs/ — *Apache HTTP Server Version 1.3 User's Guide*
- http://localhost/manual/index.html — After starting the Apache server on your local system, you can view the *Apache HTTP Server Version 1.3 User's Guide* using this URL.
- http://www.redhat.com/support/docs/apache.html — Red Hat Support maintains a list of useful Apace Web Server links.

- http://www.redhat.com/support/docs/faqs/RH-apache-FAQ/book1.html — The Red Hat Linux Apache Centralized Knowledgebase compiled by Red Hat.

14.7.3 Related Books

- *Apache: The Definitive Guide* by Ben Laurie and Peter Laurie; O'Reilly & Associates, Inc.

15 Apache Secure Server Configuration

15.1 Introduction

This chapter provides basic information on an Apache server with the `mod_ssl` security module enabled to use the OpenSSL library and toolkit. The combination of these three components, provided with Red Hat Linux, will be referred to in this chapter as the secure Web server or just as the secure server.

The `mod_ssl` module is a security module for the Apache Web server. The `mod_ssl` module uses the tools provided by the OpenSSL Project to add a very important feature to Apache — the ability to encrypt communications. In contrast, using regular HTTP, communications between a browser and a Web server are sent in plaintext, which could be intercepted and read by someone along the route between the browser and the server.

This chapter is not meant to be complete and exclusive documentation for any of these programs. When possible, this guide will point you to appropriate places where you can find more in-depth documentation on particular subjects.

This chapter will show you how to install these programs. You will also learn the steps necessary to generate a private key and a certificate request, how to generate your own self-signed certificate, and how to install a certificate to use with your secure Web server.

15.2 An Overview of Security-Related Packages

To enable the secure server, you need to have the following three packages installed at a minimum:

`apache`

The `apache` package contains the `httpd` daemon and related utilities, configuration files, icons, Apache modules, man pages and other files used by the Apache Web server.

`mod_ssl`

The `mod_ssl` package includes the mod_ssl module, which provides strong cryptography for the Apache Web server via the Secure Sockets Layer (SSL) and Transport Layer Security (TLS) protocols.

`openssl`

The `openssl` package contains the OpenSSL toolkit. The OpenSSL toolkit implements the SSL and TLS protocols and also includes a general purpose cryptography library.

`mm`

The `mm` package contains the MM library, which allows multiple instances of the `httpd` daemon to share state information.

Additionally, other software packages included with Red Hat Linux can provide certain security functionalities (but are not required by the secure server to function):

`apache-devel`

The `apache-devel` package contains the Apache include files, header files and the APXS utility. You will need all of these if you intend to load any extra modules, other than the modules provided with this product. Please see the *Official Red Hat Linux Reference Guide* for more information on loading modules into your secure Web server using Apache's DSO functionality.

If you do not intend to load other modules into your secure Web server, you do not need to install this package.

`apache-manual`

The `apache-manual` package contains the Apache Project's *Apache 1.3 User's Guide* in HTML format. This manual is also available on the Web at http://httpd.apache.org/docs/.

OpenSSH packages

The OpenSSH packages provide the OpenSSH set of network connectivity tools for logging into and executing commands on a remote machine. OpenSSH tools encrypt all traffic (including passwords), so you can avoid eavesdropping, connection hijacking, and other attacks on the communications between your machine and the remote machine.

The `openssh` package includes core files needed by both the OpenSSH client programs and the OpenSSH server. The `openssh` package also contains `scp`, a secure replacement for `rcp` (for copying files between machines) and `ftp` (for transferring files between machines).

The `openssh-askpass` package supports the display of a dialog window which prompts for a password during use of the OpenSSH agent with RSA authentication.

The `openssh-askpass-gnome` package contains a GNOME GUI desktop environment dialog window which is displayed when OpenSSH programs prompt for a password. If you are running GNOME and using OpenSSH utilities, you should install this package.

The `openssh-server` package contains the `sshd` secure shell daemon and related files. The secure shell daemon is the server side of the OpenSSH suite, and must be installed on your host if you want to allow SSH clients to connect to your host.

The `openssh-clients` package contains the client programs needed to make encrypted connections to SSH servers, including the following: `ssh`, a secure replacement for `rsh`; and `slogin`, a secure replacement for `rlogin` (for remote login) and `telnet` (for communicating with another host via the TELNET protocol).

For more information about OpenSSH, see Chapter 9, *OpenSSH* and the OpenSSH website at http://www.openssh.com.

openssl-devel

The `openssl-devel` package contains the static libraries and the include file needed to compile applications with support for various cryptographic algorithms and protocols. You need to install this package only if you are developing applications which include SSL support — you do not need this package to use SSL.

stunnel

The `stunnel` package provides the Stunnel SSL wrapper. Stunnel supports the SSL encryption of TCP connections, so it can provide encryption for non-SSL aware daemons and protocols (such as POP, IMAP and LDAP) without requiring any changes to the daemon's code.

Table 15–1, *Security Packages* displays the location of the secure server packages and additional security-related packages within the package groups provided by Red Hat Linux. This table also tells you whether each package is optional or not for the installation of a secure Web server.

Table 15–1 Security Packages

Package Name	Located in Group	Optional?
`apache`	System Environment/Daemons	no
`mod_ssl`	System Environment/Daemons	no
`openssl`	System Environment/Libraries	no
`mm`	System Environment/Libraries	no
`apache-devel`	Development/Libraries	yes
`apache-manual`	Documentation	yes
`openssh`	Applications/Internet	yes
`openssh-askpass`	Applications/Internet	yes
`openssh-askpass-gnome`	Applications/Internet	yes
`openssh-clients`	Applications/Internet	yes
`openssh-server`	System Environment/Daemons	yes

Package Name	Located in Group	Optional?
`openssl-devel`	Development/Libraries	yes
`stunnel`	Applications/Internet	yes

15.3 An Overview of Certificates and Security

Your secure Web server provides security using a combination of the Secure Sockets Layer (SSL) protocol and (in most cases) a digital certificate from a Certificate Authority (CA). SSL handles the encrypted communications and the mutual authentication between browsers and your secure Web server. The CA-approved digital certificate provides authentication for your secure Web server (the CA puts its reputation behind its certification of your organization's identity). When your browser is communicating using SSL encryption, you will see the https:// prefix at the beginning of the Uniform Resource Locator (URL) in the navigation bar.

Encryption depends upon the use of keys (think of them as secret encoder/decoder rings in data format). In conventional or symmetric cryptography, both ends of the transaction have the same key, which they use to decode each other's transmissions. In public or asymmetric cryptography, two keys co-exist: a public key and a private key. A person or an organization keeps their private key a secret, and publishes their public key. Data encoded with the public key can only be decoded with the private key; data encoded with the private key can only be decoded with the public key.

To set up your secure server, you will use public cryptography to create a public and private key pair. In most cases, you will send your certificate request (including your public key), proof of your company's identity, and payment to a CA. The CA will verify the certificate request and your identity, and then send back a certificate for your secure Web server.

A secure server uses a certificate to identify itself to Web browsers. You can generate your own certificate (called a "self-signed" certificate) or you can get a certificate from a Certificate Authority or CA. A certificate from a reputable CA guarantees that a website is associated with a particular company or organization.

Alternatively, you can create your own self-signed certificate. Note, however, that self-signed certificates should not be used in most production environments. Self-signed certificates will not be automatically accepted by a user's browser — the user will be asked by the browser if they want to accept the certificate and create the secure connection. See Section 15.5, *Types of Certificates* for more information on the differences between self-signed and CA-signed certificates.

Once you have a self-signed certificate or a signed certificate from the CA of your choice, you will need to install it on your secure Web server.

15.4 Using Pre-Existing Keys and Certificates

If you already have an existing key and certificate (for example, if you are installing the secure Web server to replace another company's secure Web server product), you will probably be able to use your existing key and certificate with the secure Web server. In the following two situations, you will not be able to use your existing key and certificate:

- *If you are changing your IP address or domain name* — You can not use your old key and certificate if you are changing your IP address or domain name. Certificates are issued for a particular IP address and domain name pair. You will need to get a new certificate if you are changing your IP address or domain name.
- *If you have a certificate from VeriSign and you are changing your server software* — VeriSign is a widely used CA. If you already have a VeriSign certificate for another purpose, you may have been considering using your existing VeriSign certificate with your new secure Web server. However, you will not be allowed to, because VeriSign issues certificates for one particular server software and IP address/domain name combination.

  If you change either of those parameters (for example, if you previously used another secure Web server product and now you want to use the secure Web server), the VeriSign certificate you obtained to use with the previous configuration will not work with the new configuration. You will need to obtain a new certificate.

If you have an existing key and certificate that you can use, you will not have to generate a new key and obtain a new certificate. However, you may need to move and rename the files which contain your key and certificate.

Move your existing key file to:

```
/etc/httpd/conf/ssl.key/server.key
```

Move your existing certificate file to:

```
/etc/httpd/conf/ssl.crt/server.crt
```

After you have moved your key and certificate, skip to Section 15.9, *Testing Your Certificate*.

If you are upgrading from the Red Hat Secure Web Server versions 1.0 and 2.0, your old key (`httpsd.key`) and certificate (`httpsd.crt`) will be located in `/etc/httpd/conf/`. You will need to move and rename your key and certificate, so that the secure Web server can use them. Use the following two commands to move and rename your key and certificate files:

```
mv /etc/httpd/conf/httpsd.key /etc/httpd/conf/ssl.key/server.key
mv /etc/httpd/conf/httpsd.crt /etc/httpd/conf/ssl.crt/server.crt
```

Then start your secure Web server with the command:

```
/sbin/service httpd start
```

For a secure server, you will be prompted to enter your password. After you type it in and press [Enter], the server will start.

You should not need to get a new certificate, if you are upgrading from a previous version of the secure Web server.

15.5 Types of Certificates

If you installed your secure Web server using the Red Hat Linux installation program, a random key and a test certificate are generated and put into the appropriate directories. Before you begin using your secure server, however, you will need to generate your own key and obtain a certificate which correctly identifies your server.

You need a key and a certificate to operate your secure Web server — which means that you can either generate a self-signed certificate or purchase a CA-signed certificate from a CA. What are the differences between the two?

A CA-signed certificate provides two important capabilities for your server:

- Browsers will (usually) automatically recognize the certificate and allow a secure connection to be made, without prompting the user.
- When a CA issues a signed certificate, they are guaranteeing the identity of the organization that is providing the Web pages to the browser.

If your secure server is being accessed by the public at large, your secure Web server needs a certificate signed by a CA, so that people who visit your website can rely that the website is owned by the organization who claims to own it. Before signing a certificate, a CA verifies that the organization requesting the certificate was actually who they claimed to be.

Most Web browsers that support SSL have a list of CAs whose certificates they will automatically accept. If a browser encounters a certificate whose authorizing CA is not in the list, the browser will ask the user to choose whether to accept or decline the connection.

You can generate a self-signed certificate for your secure Web server, but be aware that a self-signed certificate will not provide the same functionality as a CA-signed certificate. A self-signed certificate will not be automatically recognized by users' browsers, and a self-signed certificate does not provide any guarantee concerning the identity of the organization that is providing the website. A CA-signed certificate provides both of these important capabilities for a secure server. If your secure server will be used in a production environment, you will probably need a CA-signed certificate.

The process of getting a certificate from a CA is fairly easy. A quick overview is as follows:

1. Create an encryption private and public key pair.

2. Create a certificate request based on the public key. The certificate request contains information about your server and the company hosting it.
3. Send the certificate request, along with documents proving your identity, to a CA. We cannot tell you which certificate authority to choose. Your decision may be based on your past experiences, or on the experiences of your friends or colleagues, or purely on monetary factors.

   To see a list of CAs, click on the **Security** button on your Navigator toolbar or on the padlock icon at the bottom left of the screen, then click on **Signers** to see a list of certificate signers from whom your browser will accept certificates. You can also search the Web for CAs. Once you have decided upon a CA, you will need to follow the instructions they provide on how to obtain a certificate from them.
4. When the CA is satisfied that you are indeed who you claim to be, they will send you a digital certificate.
5. Install this certificate on your Web server, and begin handling secure transactions.

Whether you are getting a certificate from a CA or generating your own self-signed certificate, the first step is to generate a key. See Section 15.6, *Generating a Key* for instructions on how to generate a key.

15.6 Generating a Key

First, `cd` to the `/etc/httpd/conf` directory. Remove the fake key and certificate that were generated during the installation with the following commands:

```
rm ssl.key/server.key
rm ssl.crt/server.crt
```

Next, you need to create your own random key. Type in the following command:

```
make genkey
```

Your system will display a message similar to the following:

```
umask 77 ; \
/usr/bin/openssl genrsa -des3 1024 > /etc/httpd/conf/ssl.key/server.key
Generating RSA private key, 1024 bit long modulus
.......++++++
................................................................++++++
e is 65537 (0x10001)
Enter PEM pass phrase:
```

You now need to type in a password. For best security, your password should contain at least eight characters, include numbers and/or punctuation, and not be a word in a dictionary. Also, remember that your password is case sensitive.

Note

You will need to remember and enter this password every time you start your secure Web server, so do not forget it.

You will be asked to re-type the password, to verify that it is correct. Once you have typed it in correctly, a file called `server.key`, containing your key, will be created.

Note that if you do not want to type in a password every time you start your secure Web server, you will need to use the following two commands instead of `make genkey` to create the key. Both of these commands should be typed in entirely on one line.

Use the following command:

```
/usr/bin/openssl genrsa 1024 > /etc/httpd/conf/ssl.key/server.key
```

to create your key. Then use this command:

```
chmod go-rwx /etc/httpd/conf/ssl.key/server.key
```

to make sure that the permissions are set correctly on your key.

After you use the above commands to create your key, you will not need to use a password to start your secure Web server.

CAUTION

Disabling the password feature for your secure Web server is a security risk. We DO NOT recommend that you disable the password feature for your secure Web server.

The problems associated with not using a password are directly related to the security maintained on the host machine. For example, if an unscrupulous individual compromises the regular UNIX security on the host machine, that person could obtain your private key (the contents of your `server.key` file). The key could be used to serve Web pages that will appear to be from your Web server.

If UNIX security practices are rigorously maintained on the host computer (all operating system patches and updates are installed as soon as they are available, no unnecessary or risky services are operating, and so on), the secure Web server's password may seem unnecessary. However, since your secure Web server should not need to be re-booted very often, the extra security provided by entering a password is a worthwhile effort in most cases.

The `server.key` file should be owned by the root user on your system and should not be accessible to any other user. Make a backup copy of this file and keep the backup copy in a safe, secure place. You need the backup copy because if you ever lose the `server.key` file after using it to create your certificate request, your certificate will no longer work and the CA will not be able to help you. Your only option would be to request (and pay for) a new certificate.

If you are going to purchase a certificate from a CA, continue to Section 15.7, *Generating a Certificate Request to Send to a CA*. If you are generating your own self-signed certificate, continue to Section 15.8, *Creating a Self-Signed Certificate*.

15.7 Generating a Certificate Request to Send to a CA

Once you have created a key, the next step is to generate a certificate request which you will need to send to the CA of your choice. Type in the following command:

```
make certreq
```

Your system will display the following output and will ask you for your password (unless you disabled the password option):

```
umask 77 ; \
/usr/bin/openssl req -new -key /etc/httpd/conf/ssl.key/server.key
-out /etc/httpd/conf/ssl.csr/server.csr
Using configuration from /usr/share/ssl/openssl.cnf
Enter PEM pass phrase:
```

Type in the password that you chose when you were generating your key. Your system will display some instructions and then ask for a series of responses from you. Your inputs will be incorporated into the certificate request. The display, with example responses, will look like this:

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a
DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter ’.’, the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:North Carolina
Locality Name (eg, city) []:Durham
Organization Name (eg, company) [Internet Widgits]:Test Company
Organizational Unit Name (eg, section) []:Testing
```

```
Common Name (your name or server’s hostname) []:test.example.com
Email Address []:admin@example.com
Please enter the following ’extra’ attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

The default answers appear in brackets `[]` immediately after each request for input. For example, the first information required is the name of the country where the certificate will be used, shown like the following:

```
Country Name (2 letter code) [AU]:
```

The default input, in brackets, is **`AU`**. To accept the default, just press [Enter], or fill in your country’s two letter code.

You will have to type in the rest of the inputs (`State or Province Name`, `Locality Name`, `Organization Name`, `Organizational Unit Name`, `Common Name`, and `Email address`). All of these should be self-explanatory, but you need to follow these guidelines:

- Do not abbreviate the locality or state. Write them out (for example, St. Louis should be written out as Saint Louis).
- If you are sending this CSR to a CA, be very careful to provide correct information for all of the fields, but especially for the `Organization Name` and the `Common Name`. CAs check the information provided in the CSR to determine whether your organization is responsible for what you provided as the `Common Name`. CAs will reject CSRs which include information they perceive as invalid.
- For `Common Name`, make sure you type in the *real* name of your secure Web server (a valid DNS name) and not any aliases which the server may have.
- The `Email Address` should be the email address for the webmaster or system administrator.
- Avoid any special characters like @, #, &, !, etc. Some CAs will reject a certificate request which contains a special character. So, if your company name includes an ampersand (&), spell it out as "and" instead of "&."
- Do not use either of the extra attributes (`A challenge password` and `An optional company name`). To continue without entering these fields, just press [Enter] to accept the blank default for both inputs.

When you have finished entering your information, a file named `server.csr` will be created. This file is your certificate request, ready to send to your CA.

After you have decided on a CA, follow the instructions they provide on their website. Their instructions will tell you how to send your certificate request, any other documentation that they require, and your payment to them.

After you have fulfilled the CA's requirements, they will send a certificate to you (usually by email). Save (or cut and paste) the certificate that they send you as `/etc/httpd/conf/ssl.crt/server.crt`.

15.8 Creating a Self-Signed Certificate

You can create your own self-signed certificate. Please note that a self-signed certificate will not provide the security guarantees provided by a CA-signed certificate. See Section 15.5, *Types of Certificates* for more details about certificates.

If you would like to make your own self-signed certificate, you will first need to create a random key using the instructions provided in Section 15.6, *Generating a Key*. Once you have a key, use the following command:

```
make testcert
```

You will see the following output and you will be prompted for your password (unless you generated a key without a password):

```
umask 77 ; \
/usr/bin/openssl req -new -key /etc/httpd/conf/ssl.key/server.key
-x509 -days 365 -out /etc/httpd/conf/ssl.crt/server.crt
Using configuration from /usr/share/ssl/openssl.cnf
Enter PEM pass phrase:
```

After you enter your password (or without a prompt if you created a key without a password), you will be asked for more information. The computer's output and a set of inputs looks like the following (you will need to provide the correct information for your organization and host):

```
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a
DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:North Carolina
Locality Name (eg, city) []:Durham
Organization Name (eg, company) [Internet Widgits]:My Company, Inc.
Organizational Unit Name (eg, section) []:Documentation
```

```
Common Name (your name or server’s hostname) []:myhost.example.com
Email Address []:myemail@example.com
```

After you provide the correct information, a self-signed certificate will be created and placed in `/etc/httpd/conf/ssl.crt/server.crt`. You will need to restart your secure server after generating the certificate with the command

```
/sbin/service httpd restart
```

15.9 Testing Your Certificate

When the secure server is installed by the Red Hat Linux installation program, a random key and a generic certificate are installed, for testing purposes. You can connect to your secure server using this certificate. For any purposes other than testing, however, you need to get a certificate from a CA or generate a self-signed certificate. See Section 15.5, *Types of Certificates* if you need more information on the different types of certificates available.

If you have purchased a certificate from a CA or generated a self-signed certificate, you should have a file named `/etc/httpd/conf/ssl.key/server.key`, containing your key, and a file named `/etc/httpd/conf/ssl.crt/server.crt`, containing your certificate. If your key and certificate are somewhere else, move them to these directories. If you changed any of the default locations or filenames for the secure Web server in your Apache configuration files, you should put these two files in the appropriate directory, based on your modifications.

Now, restart your server with the command:

```
/sbin/service httpd restart
```

If your key file is encrypted, you will be asked for the password. Type in your password and your server should start.

Point your Web browser to your server’s home page. The URL to access your secure Web server will look like this:

```
https://your_domain
```

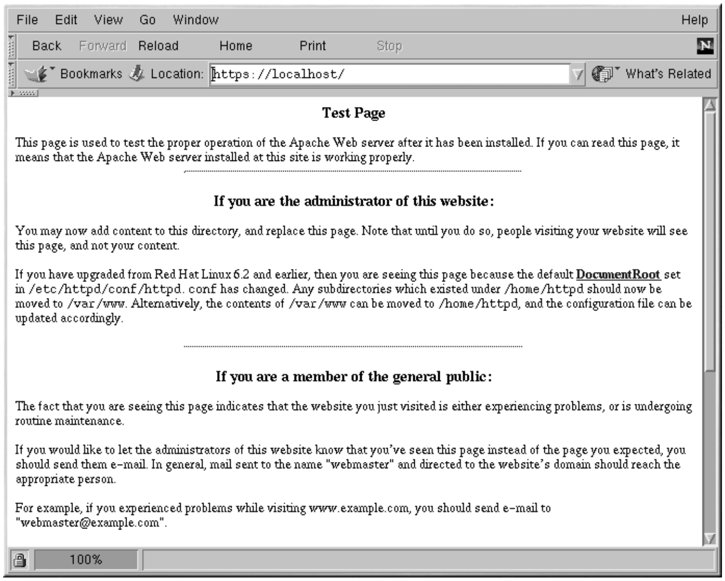
Note

Note the "s" after "http." The https: prefix is used for secure HTTP transactions.

If you are using a CA-signed certificate from a well-known CA, your browser will probably automatically accept the certificate (without prompting you for input) and create the secure connection. Your browser will not automatically recognize a test or a self-signed certificate, because the certificate is not signed by a CA. If you are not using a certificate from a CA, follow the instructions provided by your browser to accept the certificate. You can just accept the defaults by clicking **Next** until the dialogs are finished.

Once your browser accepts the certificate, your secure Web server will show you a default home page as shown in Figure 15–1, *The Default Home Page*.

Figure 15–1 The Default Home Page



Test Page

This page is used to test the proper operation of the Apache Web server after it has been installed. If you can read this page, it means that the Apache Web server installed at this site is working properly.

If you are the administrator of this website:

You may now add content to this directory, and replace this page. Note that until you do so, people visiting your website will see this page, and not your content.

If you have upgraded from Red Hat Linux 6.2 and earlier, then you are seeing this page because the default **DocumentRoot** set in /etc/httpd/conf/httpd. conf has changed. Any subdirectories which existed under /home/httpd should now be moved to /var/www. Alternatively, the contents of /var/www can be moved to /home/httpd, and the configuration file can be updated accordingly.

If you are a member of the general public:

The fact that you are seeing this page indicates that the website you just visited is either experiencing problems, or is undergoing routine maintenance.

If you would like to let the administrators of this website know that you've seen this page instead of the page you expected, you should send them e-mail. In general, mail sent to the name "webmaster" and directed to the website's domain should reach the appropriate person.

For example, if you experienced problems while visiting www.example.com, you should send e-mail to "webmaster@example.com".

15.10 Accessing Your Secure Server

To access your secure server, use a URL like this:

```
https://your_domain
```

Note that URLs which are intended to connect to your secure Web server should begin with the https: protocol designator instead of the more common http: protocol designator.

Your non-secure server can be accessed using an URL like this:

```
http://your_domain
```

The standard port for secure Web communications is port 443. The standard port for non-secure Web communications is port 80. The secure Web server default configuration listens on both of the two standard ports. Therefore, you will not need to specify the port number in a URL (the port number is assumed).

However, if you configure your server to listen on a non-standard port (i.e., anything besides 80 or 443), you will need to specify the port number in every URL which is intended to connect to the server on the non-standard port.

For example, you may have configured your server so that you have a virtual host running non-secured on port 12331. Any URLs intended to connect to that virtual host must specify the port number in the URL. The following URL example will attempt to connect to a non-secure Web server listening on port 12331:

```
http://your_domain:12331
```

Some of the example URLs used in this manual may need to be changed, depending upon whether you are accessing your secure Web server or your non-secure Web server. Please view all URLs in this manual as general examples and not as explicit instructions that will work under all circumstances.

15.11 Additional Resources

If you followed the steps outlined in Chapter 15, *Apache Secure Server Configuration* but you experienced a problem, the first thing you should do is check the Red Hat Errata section of the Red Hat website at http://www.redhat.com/support/errata.

If you purchased an Official Red Hat product which included support, you are entitled to technical support. Be sure to visit the Red Hat Support website at http://www.redhat.com/support to register for support.

You may want to subscribe to the redhat-secure-server mailing list. You can subscribe to this mailing list at http://www.redhat.com/mailing-lists.

You can also subscribe to the redhat-secure-server mailing list by emailing `redhat-secure-server-request@redhat.com` and include the word "subscribe" (without the quotation marks) in the Subject line.

Refer to Section 14.7, *Additional Resources* for additional references about Apache.

15.11.1 Installed Documentation

- `mod_ssl documentation` — Open a Web browser, and go to the URL http://localhost/manual/mod/mod_ssl/ on the server that is running Apache.

15.11.2 Useful Websites

- http://www.modssl.org — The `mod_ssl` website is the definitive source for information about `mod_ssl`. The website includes a wealth of documentation, including a *User Manual* at http://www.modssl.org/docs.

15.11.3 Related Books

Apache: The Definitive Guide, 2nd edition, by Ben Laurie and Peter Laurie, O’Reilly & Associates, Inc.

16 BIND Configuration

This chapter assumes that you have a basic understanding of BIND and DNS; it does not attempt to explain the concepts of BIND and DNS. This chapter does explain how to use BIND Configuration Tool (`bindconf`) to configure basic BIND server zones for BIND version 8. BIND Configuration Tool creates the `/etc/named.conf` configuration file and the zone configuration files in the `/var/named` directory each time you apply your changes.

If you require more functionality than this tool provides, you can create the `/etc/named.conf` configuration file using BIND Configuration Tool and then add your customized settings. However, once you manually modify the configuration file, you cannot use BIND Configuration Tool to edit the custom configuration settings that were added manually.

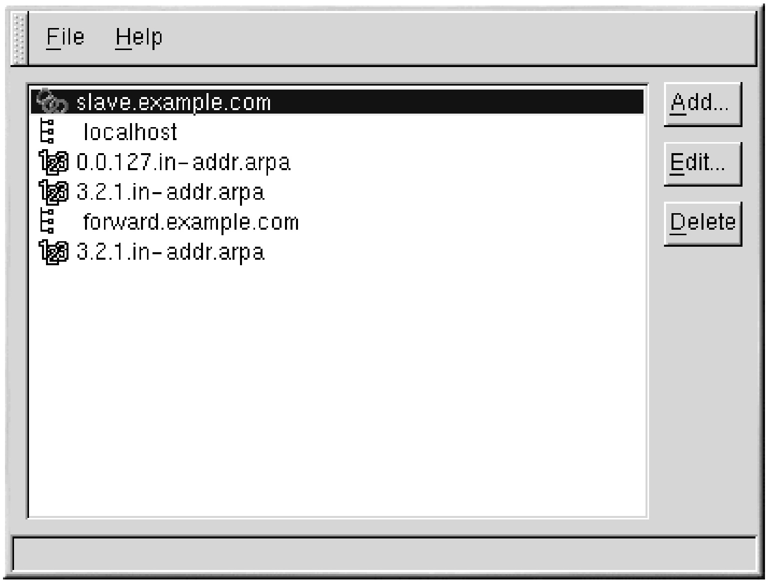
Do Not Edit `/etc/named.conf`

Do not edit the `/etc/named.conf` configuration file. BIND Configuration Tool generates this file after you apply your changes. If you want to configure settings that are not configurable using BIND Configuration Tool, then do not use it.

BIND Configuration Tool requires the X Window System and root access. To start BIND Configuration Tool, use one of the following methods:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Configure DNS**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Configure DNS**.
- Type the command `bindconf` at a shell prompt (for example, in an XTerm or GNOME-terminal).

Figure 16–1 bindconf



BIND Configuration Tool configures the default zone directory to be `/var/named`. All zone files specified are relative to this directory. BIND Configuration Tool also includes basic syntax checking when values are entered. For example, if a valid entry is an IP address, you are only allowed to type numbers and the dot (.) character into the text area.

BIND Configuration Tool allows you to add a forward master zone, a reverse master zone, and a slave zone. After adding the zones, you can edit or delete them from the main window as shown in Figure 16–1, *bindconf*.

After adding, editing, or deleting a zone, you must choose **File** => **Apply** to write the `/etc/named.conf` configuration file and all the individual zone files in the `/var/named` directory. Applying your changes will also have the `named` service reload the configuration files. You can also choose **File** => **Exit** and click **Yes** to **Do you want to apply your changes before exiting?**

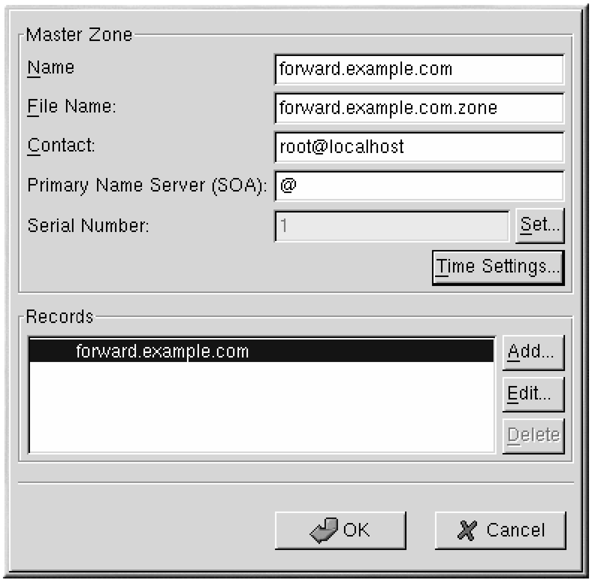
16.1 Adding a Forward Master Zone

To add a forward master zone (also known as a primary master), click the **Add** button, select **Forward Master Zone**, and enter the domain name for the master zone in the **Domain name** text area.

A new window as shown in Figure 16–2, *Adding a Forward Master Zone* will appear with the following options:

- **Name** — Domain name that was just entered in the previous window.
- **File Name** — File name of the DNS database file, relative to `/var/named`.
- **Contact** — Email address of the main contact for the master zone.
- **Primary Name Server (SOA)** — State of authority (SOA) record. This specifies the name server that is the best resource of information for this domain. The default value is @, which means that the SOA is the same as the domain name entered in the **Name** field above.
- **Serial Number** — The serial number of the DNS database file. This number must be incremented each time the file is changed, so that the slave name servers for the zone will retrieve the latest data. BIND Configuration Tool increments this number each time the configuration changes. It can also be incremented manually by clicking the **Set** button next to the **Serial Number** value.
- **Time Settings** — The **Refresh**, **Retry**, **Expire**, and **Minimum** TTL (Time to Live) values that are stored in the DNS database file.
- **Records** — Add, edit, and delete record resources of type **Host**, **Alias**, and **Name server**.

Figure 16–2 Adding a Forward Master Zone



Master Zone

Name forward.example.com

File Name: forward.example.com.zone

Contact: root@localhost

Primary Name Server (SOA): @

Serial Number: 1

Records

forward.example.com

The configuration shown in Figure 16–2, *Adding a Forward Master Zone* creates the following entry in `/etc/named.conf`:

```
zone  "forward.example.com" {
 type master;
 file  "forward.example.com.zone";
};
```

It also creates the file `/var/named/forward.example.com.zone` with the following information:

```
$TTL 86400
@ IN SOA @  root.localhost (
   1 ; serial
   28800 ; refresh
   7200 ; retry
   604800 ; expire
   86400 ; ttl
   )
```

After configuring the Forward Master Zone, click **OK** to return to the main window as shown in Figure 16–1, *bindconf*. From the pulldown menu, choose **File** => **Apply** to write the `/etc/named.conf` configuration file, write all the individual zone files in the `/var/named` directory, and have the daemon reload the configuration files.

16.2 Adding a Reverse Master Zone

To add a reverse master zone, click the **Add** button and select **Reverse Master Zone**. Enter the first three octets of the IP address range that you want to configure. For example, if you are configuring the IP address range 192.168.10.0/255.255.255.0, enter 192.168.10 in the **IP Address (first 3 Octets)** text area.

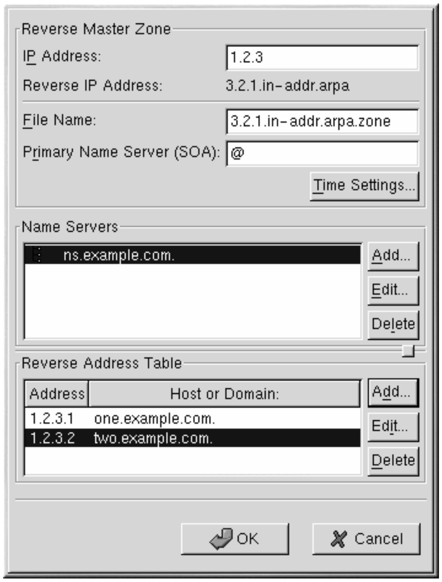
A new window will appear, as shown in Figure 16–3, *Adding a Reverse Master Zone*, with the following options:

1. **IP Address** — The first three octets that you just entered in the previous window.
2. **Reverse IP Address** — Non-editable. Pre-populated based on the IP Address entered.
3. **File Name** — File name of DNS database file in the `/var/named` directory.
4. **Primary Name Server (SOA)** — State of authority (SOA) record. This specifies the name server that is the best resource of information for this domain. The default value is @, which means that the SOA is the same as the domain name entered in the **Name** field above.
5. **Time Settings** — The **Refresh**, **Retry**, **Expire**, and **Minimum** TTL (Time to Live) values that are stored in the DNS database file.

6. **Name Servers** — Add, edit, and delete name servers for for the reverse master zone. At least one name server is required.

7. **Reverse Address Table** — List of IP addresses within the reverse master zone and their hostnames. For example, for the reverse master zone 1.2.3, you can add 1.2.3.100 in the **Reverse Address Table** with the hostname foo.example.com. The hostname must end with a period (.) to specify that it is a full hostname.

Figure 16–3 Adding a Reverse Master Zone



Reverse Master Zone

IP Address: 1.2.3

Reverse IP Address: 3.2.1.in-addr.arpa

File Name: 3.2.1.in-addr.arpa.zone

Primary Name Server (SOA): @

Name Servers

ns.example.com.

Reverse Address Table

Address	Host or Domain:
1.2.3.1	one.example.com.
1.2.3.2	two.example.com.

The configuration shown in Figure 16–3, *Adding a Reverse Master Zone* creates the following entry in `/etc/named.conf`:

```
zone  "3.2.1.in-addr.arpa" {
 type master;
 file  "3.2.1.in-addr.arpa.zone";
};
```

It also creates the file `/var/named/3.2.1.in-addr.arpa.zone` with the following information:

```
$TTL 86400
@ IN SOA @ root.localhost (
   2 ; serial
```

```
        28800 ; refresh
        7200 ; retry
        604800 ; expire
        86400 ; ttk
        )


    @ IN NS ns.example.com.

    1 IN PTR one.example.com.
    2 IN PTR two.example.com.
```

After configuring the Reverse Master Zone, click **OK** to return to the main window, as shown in Figure 16–1, *bindconf*. From the pulldown menu, choose **File** => **Apply** to write the `/etc/named.conf` configuration file, write all the individual zone files in the `/var/named` directory, and have the daemon reload the configuration files.

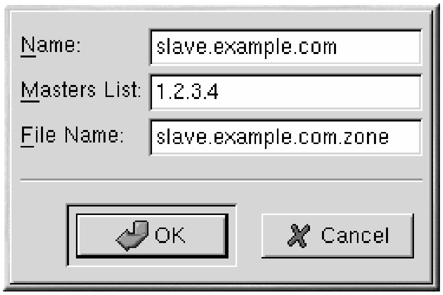
16.3 Adding a Slave Zone

To add a slave zone (also known as a secondary master), click the **Add** button and select **Slave Zone**. Enter the domain name for the slave zone in the **Domain name** text area.

A new window will appear, as shown in Figure 16–4, *Adding a Slave Zone*, with the following options:

- **Name** — The domain name that was entered in the previous window.
- **Masters List** — The name server from which the slave zone retrieves its data. This value must be a valid IP address. You can only enter numbers and dots (.) in the text area.
- **File Name** — File name of the DNS database file in `/var/named`.

Figure 16–4 Adding a Slave Zone



Name: slave.example.com

Masters List: 1.2.3.4

File Name: slave.example.com.zone

The configuration shown in Figure 16–4, *Adding a Slave Zone* creates the following entry in `/etc/named.conf`:

```
zone "slave.example.com" {
 type slave;
 file "slave.example.com.zone";
 masters {
   1.2.3.4;
  };
};
```

The configuration file `/var/named/slave.example.com.zone` is created by the `named` service when it downloads the zone data from the master server(s).

After configuring the slave zone, click **OK** to return to the main window as shown in Figure 16–1, *bindconf*. From the pulldown menu, choose **File** => **Apply** to write the `/etc/named.conf` configuration file and have the daemon reload the configuration files.

Part III System Configuration

17 Console Access

When normal (non-root) users log into a computer locally, they are given two types of special permissions:

1. They can run certain programs that they would not otherwise be able to run
2. They can access certain files (normally special device files used to access diskettes, CD-ROMs, and so on) that they would not otherwise be able to access

Since there are multiple consoles on a single computer and multiple users can be logged into the computer locally at the same time, one of the users has to "win" the race to access the files. The first user to log in at the console owns those files. Once the first user logs out, the next user who logs in will own the files.

In contrast, *every* user who logs in at the console will be allowed to run programs that accomplish tasks normally restricted to the root user. If X is running, these actions can be included as menu items in a graphical user interface. As shipped, the console-accessible programs include `halt`, `poweroff`, and `reboot`.

17.1 Disabling Shutdown Via Ctrl-Alt-Del

By default, `/etc/inittab` specifies that your system is set to shutdown and reboot the system in response to a [Ctrl]-[Alt]-[Del] key combination used at the console. If you wouldd like to completely disable this ability, you will need to comment out the following line in `/etc/inittab` by putting a hash mark (#) in front of it:

```
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

Alternatively, you may just want to allow certain non-root users the right to shutdown the system from the console using [Ctrl]-[Alt]-[Del]. You can restrict this privilege to certain users, by taking the following steps:

1. Add a `-a` option to the `/etc/inittab` line shown above, so that it reads:

   ```
   ca::ctrlaltdel:/sbin/shutdown -a -t3 -r now
   ```

   The `-a` flag tells `shutdown` to look for the `/etc/shutdown.allow` file, which yoy will create in the next step.

2. Create a file named `shutdown.allow` in `/etc`. The `shutdown.allow` file should list the usernames of any users who are allowed to shutdown the system using [Ctrl]-[Alt]-[Del]. The format of the `/etc/shutdown.allow` file is a list of usernames, one per line, like the following:

   ```
   stephen
   ```

```
        jack
        sophie
```

According to this example shutdown.allow file, stephen, jack, and sophie are allowed to shutdown the system from the console using [Ctrl]-[Alt]-[Del]. When that key combination is used, the shutdown -a in /etc/inittab checks to see if any of the users in /etc/shutdown.allow (or root) are logged in on a virtual console. If one of them is, the shutdown of the system will continue; if not, an error message will be written to the system console instead.

For more information on shutdown.allow see the shutdown man page.

17.2 Disabling Console Program Access

In order to disable access by users to console programs, you should run this command as root:

```
rm -f /etc/security/console.apps/*
```

In environments where the console is otherwise secured (BIOS and boot loader passwords are set, [Ctrl]-[Alt]-[Delete] is disabled, the power and reset switches are disabled, and so forth), you may not want to allow any user at the console to run poweroff, halt, and reboot, which are accessible from the console by default.

To remove these abilities, run the following commands as root:

```
rm -f /etc/security/console.apps/poweroff
rm -f /etc/security/console.apps/halt
rm -f /etc/security/console.apps/reboot
```

17.3 Disabling All Console Access

The PAM pam_console.so module manages console file permissions and authentication. (See the *Official Red Hat Linux Reference Guide* for more information on configuring PAM.) If you want to disable all console access, including program and file access, comment out all lines that refer to pam_console.so in the /etc/pam.d directory. As root, the following script will do the trick:

```
cd /etc/pam.d
for i in * ; do
sed '/[^#].*pam_console.so/s/^/#/' < $i > foo && mv foo $i
done
```

17.4 Defining the Console

The pam_console.so module uses the /etc/security/console.perms file to determine the permissions for users at the system console. The syntax of the file is very flexible; you can edit the file so that these instructions no longer apply. However, the default file has a line that looks like this:

```
 <console>=tty[0-9][0-9]* :[0-9]\.[0-9] :[0-9]
```

When users log in, they are attached to some sort of named terminal, either an X server with a name like `:0` or `mymachine.example.com:1.0` or a device like `/dev/ttyS0` or `/dev/pts/2`. The default is to define that local virtual consoles and local X servers are considered local, but if you want to consider the serial terminal next to you on port `/dev/ttyS1` to also be local, you can change that line to read:

```
<console>=tty[0-9][0-9]* :[0-9]\.[0-9] :[0-9] /dev/ttyS1
```

17.5 Making Files Accessible From the Console

In `/etc/security/console.perms`, there is a section with lines like:

```
<floppy>=/dev/fd[0-1]* \
    /dev/floppy/*
<sound>=/dev/dsp* /dev/audio* /dev/midi* \
    /dev/mixer* /dev/sequencer \
    /dev/sound/*
<cdrom>=/dev/cdrom* /dev/cdwriter*
```

You can add your own lines to this section, if necessary. Make sure that any lines you add refer to the appropriate device. For example, you could add the following line:

```
<scanner>=/dev/scanner
```

(Of course, make sure that `/dev/sga` is really your scanner and not, say, your hard drive.)

That's the first step. The second step is to define what is done with those files. Look in the last section of `/etc/security/console.perms` for lines similar to:

```
<console> 0660 <floppy> 0660 root.floppy
<console> 0600 <sound>  0640 root
<console> 0600 <cdrom>  0600 root.disk
```

and add a line like:

```
<console> 0600 <scanner> 0600 root
```

Then, when you log in at the console, you will be given ownership of the `/dev/sga` device and the permissions will be 0600 (readable and writable by you only). When you log out, the device will be owned by root and still have 0600 (now: readable and writable by root only) permissions.

17.6 Enabling Console Access for Other Applications

If you wish to make other applications accessible to console users, you will have to do just a little bit more work.

First of all, console access *only* works for applications which reside in `/sbin` or `/usr/sbin`, so the application that you wish to run must be there. After verifying that, do the following steps:

1. Create a link from the name of your application, such as our sample *`foo`* program, to the `/usr/bin/consolehelper` application:

   ```
   cd /usr/bin
   ln -s consolehelper foo
   ```

2. Create the file `/etc/security/console.apps/`*`foo`*:

   ```
   touch /etc/security/console.apps/foo
   ```

3. Create a PAM configuration file for the *`foo`* service in `/etc/pam.d/`. An easy way to do this is to start with a copy of the halt service's PAM configuration file, and then modify the file if you want to change the behavior:

   ```
   cp /etc/pam.d/halt /etc/pam.d/foo
   ```

Now, when you run `/usr/bin/`*`foo`*, it will call `consolehelper`, which will authenticate the user with the help of `/usr/sbin/userhelper`. To authenticate the user, `consolehelper` will ask for the user's password if `/etc/pam.d/`*`foo`* is a copy of `/etc/pam.d/halt` (otherwise, it will do precisely what is specified in `/etc/pam.d/`*`foo`*) and then run `/usr/sbin/`*`foo`* with root permissions.

17.7 The `floppy` Group

If, for whatever reason, console access is not appropriate for you and you need to give non-root users access to your system's diskette drive, this can be done using the `floppy` group. Simply add the user(s) to the `floppy` group using the tool of your choice. Here's an example showing how `gpasswd` can be used to add user fred to the `floppy` group:

```
[root@bigdog root]# gpasswd -a fred floppy
Adding user fred to group floppy
[root@bigdog root]#
```

Now, user fred will now be able to access the system's diskette drive from the console.

18 Time and Date Configuration

Red Hat Linux no longer includes timetool. The dateconfig utility has replaced timetool. The dateconfig allows the user to change the system date and time, to configure the time zone used by the system, and to setup the Network Time Protocol (NTP) daemon to synchronize the system clock with a time server.

To use dateconfig, you must be running the X Window System and have root privileges. To start dateconfig, use one of the following methods:

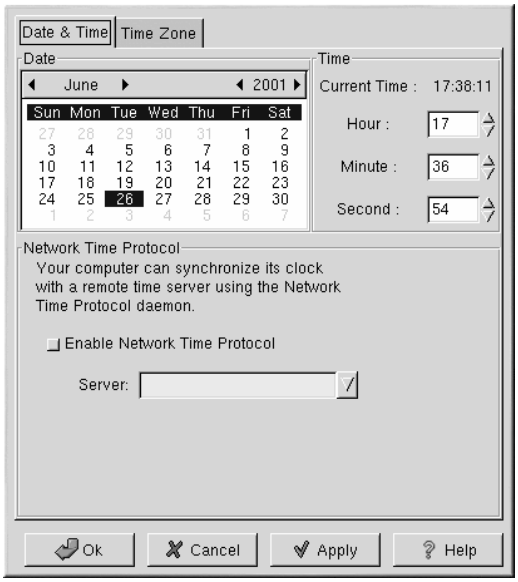
- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Date/Time Properties**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Date/Time Properties**.
- Type the command `dateconfig` at a shell prompt (for example, in an XTerm or a GNOME terminal). [1]

18.1 Time and Date Properties

As shown in Figure 18–1, *Time and Date Properties*, the first tabbed window that appears is for configuring the system date and time and the NTP daemon.

[1] If you type `timetool` at a shell prompt, Dateconfig will start.

Figure 18–1 Time and Date Properties



June 2001

Sun	Mon	Tue	Wed	Thu	Fri	Sat
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Current Time : 17:38:11

Hour : 17

Minute : 36

Second : 54

Your computer can synchronize its clock with a remote time server using the Network Time Protocol daemon.

☐ Enable Network Time Protocol

Server:

To change the date, use the arrows to the left and right of the month to change the month. Use the arrows to the left and right of the year to change the year, and click on the day of the week to change the day of the week. Changes will not take place until you click the **Apply** button.

To change the time, use the up and down arrow buttons beside the **Hour**, **Minute**, and **Second** in the **Time** section. Changes will not take place until you click the **Apply** button.

Note

Changing the date and time will change the system clock as well as the hardware clock. Clicking **Apply** or **Ok** is equivalent to executing the `date` and `hwclock` commands with the selected date and time.

The Network Time Protocol (NTP) daemon synchronizes the system clock with a remote time server or time source (such as a satellite). dateconfig allows you to configure a NTP daemon to synchronize your system clock with a remote server. To enable this feature, click the **Enable Network Time Protocol** button. This will enable the **Server** pulldown menu. You can choose one of the predefined servers or type a server name in the pulldown menu. Your system will not start synchronizing with the NTP server until you click **Apply**. After you click **Apply**, the configuration will be saved and the

NTP daemon (`ntpd`) will be started (or restarted if it is already running). If you want this daemon to start automatically at boot time, you need to execute the command `/sbin/chkconfig --level 345 ntpd on` to enable `ntpd` for runlevels 3, 4, and 5.

For more information on NTP, read the NTP documentation available in the `/usr/share/doc/ntp-`*`version-number`* directory.

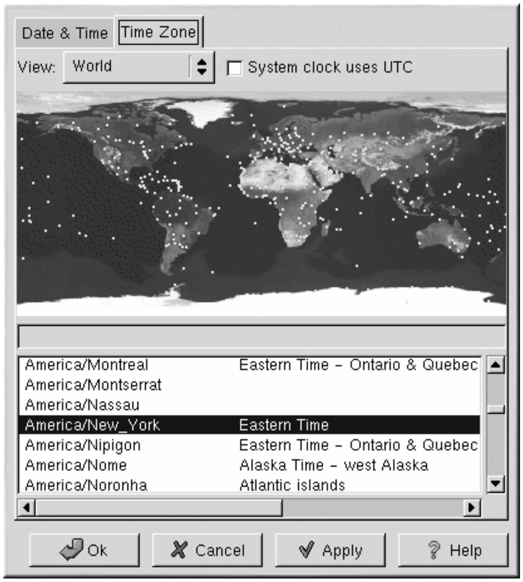
Clicking the **Apply** button will apply any changes that you have made to the date and time, the NTP daemon settings, and the time zone settings. Clicking the **Ok** button will apply the changes and then exit the program.

18.2 Time Zone Configuration

To configure the system time zone, click the **Time Zone** tab as shown in Figure 18–2, *Time and Date Properties*. The time zone can be changed by either using the interactive map or by choosing the desired time zone from the list below the map. To use the map, click on the city that represents the desired time zone. A red **X** will appear and the time zone selection will change in the list below the map. Click **Apply** to save the changes. Click **Ok** to apply the changes and exit the program.

If your system clock is set to use UTC, select the **System clock uses UTC** option. UTC stands for the universal time zone, also known as Greenwich mean time (GMT). Other time zones are determined by adding or subtracting from the UTC time.

Figure 18–2 Time and Date Properties



View: World

□ System clock uses UTC

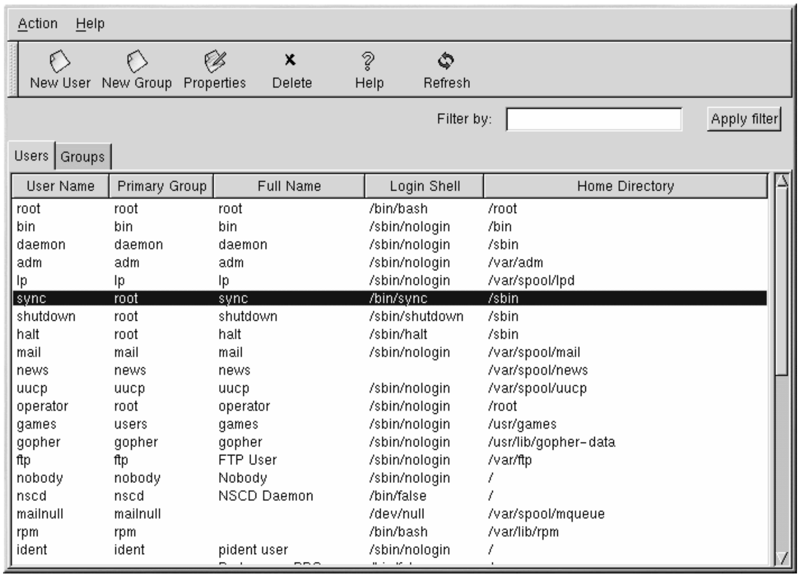
America/Montreal	Eastern Time – Ontario & Quebec
America/Montserrat	
America/Nassau	
America/New_York	Eastern Time
America/Nipigon	Eastern Time – Ontario & Quebec
America/Nome	Alaska Time – west Alaska
America/Noronha	Atlantic islands

19 User and Group Configuration

To use User Manager, you must be running the X Window System and have root privileges. To start User Manager, use one of the following methods:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **User Manager**.
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **User Manager**.
- Type the command `redhat-config-users` at a shell prompt (for example, in an XTerm or a GNOME terminal).

Figure 19–1 User Manager



User Name	Primary Group	Full Name	Login Shell	Home Directory
root	root	root	/bin/bash	/root
bin	bin	bin	/sbin/nologin	/bin
daemon	daemon	daemon	/sbin/nologin	/sbin
adm	adm	adm	/sbin/nologin	/var/adm
lp	lp	lp	/sbin/nologin	/var/spool/lpd
sync	root	sync	/bin/sync	/sbin
shutdown	root	shutdown	/sbin/shutdown	/sbin
halt	root	halt	/sbin/halt	/sbin
mail	mail	mail	/sbin/nologin	/var/spool/mail
news	news	news		/var/spool/news
uucp	uucp	uucp	/sbin/nologin	/var/spool/uucp
operator	root	operator	/sbin/nologin	/root
games	users	games	/sbin/nologin	/usr/games
gopher	gopher	gopher	/sbin/nologin	/usr/lib/gopher-data
ftp	ftp	FTP User	/sbin/nologin	/var/ftp
nobody	nobody	Nobody	/sbin/nologin	/
nscd	nscd	NSCD Daemon	/bin/false	/
mailnull	mailnull		/dev/null	/var/spool/mqueue
rpm	rpm		/bin/bash	/var/lib/rpm
ident	ident	pident user	/sbin/nologin	/

User Manager allows you to view, modify, add, and delete local users and groups. To view a list of all local users on the system, click the **Users** tab. To view a list of all local groups on the system, click the **Groups** tab.

If you need to find a specific user or group, type the first few letters of the name in the **Filter by** field. Press [Enter] or click the **Apply filter** button. The filtered list will be displayed.

For more information on users and groups, refer to the *Official Red Hat Linux Reference Guide*.

19.1 Adding a New User

To add a new user, click the **New User** button. A window as shown in Figure 19–2, *New User* will appear. Type the username and full name for the new user in the appropriate fields. Type the user's password in the **Password** and **Confirm Password** fields. The password must be at least six characters.

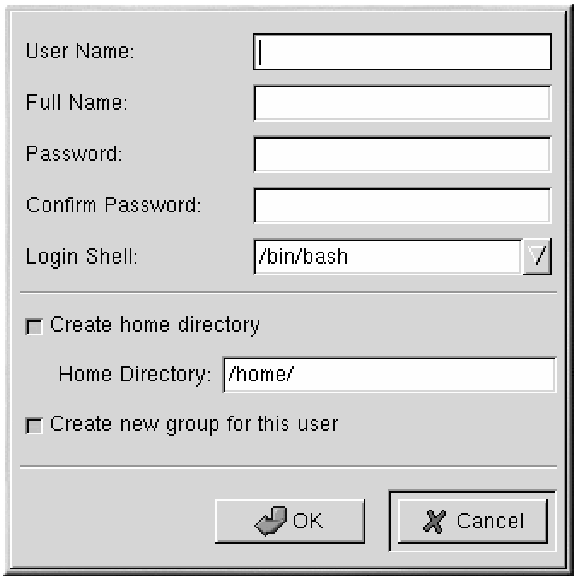
Tip

The longer the user's password, the more difficult it is for someone else to guess it and log in to the user's account without permission. It is also recommended that the password not be based on a word and that the password be a combination of letters, numbers, and special characters.

Select a login shell. If you are not sure which shell to select, accept the default value of `/bin/bash`. The default home directory is `/home/`*`username`*. You can change the home directory that is created for the user, or you can choose not to create the home directory by unselecting **Create home directory**.

Red Hat uses a **user private group** (UPG) scheme. The UPG scheme does not add or change anything in the standard UNIX way of handling groups; it simply offers a new convention. Whenever you create a new user, by default, a unique group with the same name as the user is created. If you do not want to create this group, unselect **Create new group for this user**. Click **OK** to create the user.

Figure 19–2 New User



User Name:

Full Name:

Password:

Confirm Password:

Login Shell: /bin/bash

☐ Create home directory

Home Directory: /home/

☐ Create new group for this user

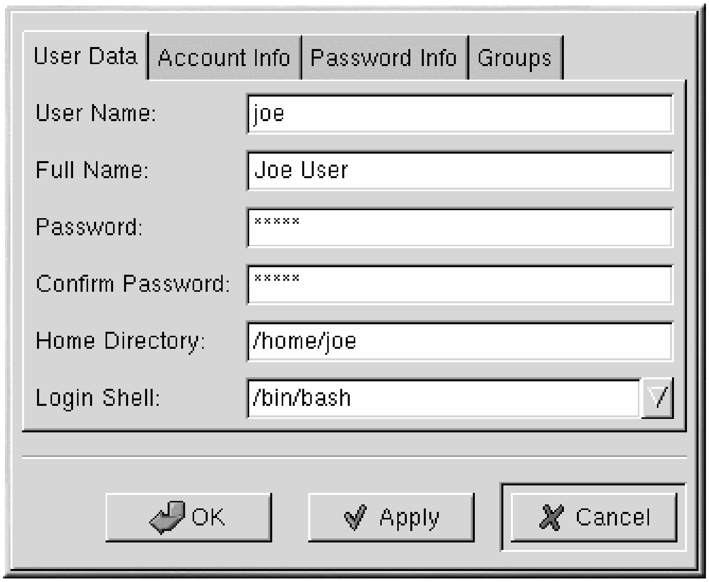
To configure more advanced user properties such as password expiration, modify the user’s properties after adding the user. Refer to Section 19.2, *Modifying User Properties* for more information.

To add the user to more user groups, click on the **User** tab, select the user, and click **Properties**. In the **User Properties** window, select the **Groups** tab. Select the groups that you want the user to be a member of.

19.2 Modifying User Properties

To view the properties of an existing user, click on the **Users** tab, select the user from the user list, and click **Properties** from the button menu (or choose **Action** => **Properties** from the pull-down menu). A window similar to Figure 19–3, *User Properties* will appear.

Figure 19–3 User Properties



User Name:	joe
Full Name:	Joe User
Password:	*****
Confirm Password:	*****
Home Directory:	/home/joe
Login Shell:	/bin/bash

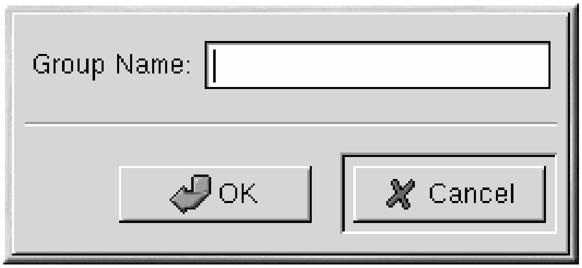
The **User Properties** window is divided into tabbed pages:

- **User Data** — Basic user information configured when you added the user. Use this tab to change the user’s full name, password, home directory, or login shell.
- **Account Info** — Select **Enable account expiration** if you want the account to expire on a certain date. Enter the date in the provided fields. Select **User account is locked** to lock the user account so that the user can not log in to the system.
- **Password Info** — This tab shows the date that the user lasted changed his password. To force the user to change his password after a certain number of days, select **Enable password expiration**. You can also set the number of days before the user is allowed to change his password, the number of days before the user is warned to change his password, and days before the account become inactive.
- **Groups** — Select the groups that you want the user to be a member of.

19.3 Adding a New Group

To add a new user group, click the **New Group** button. You will be prompted enter a group name. Type the name of the new group and click **OK**.

Figure 19–4 New Group

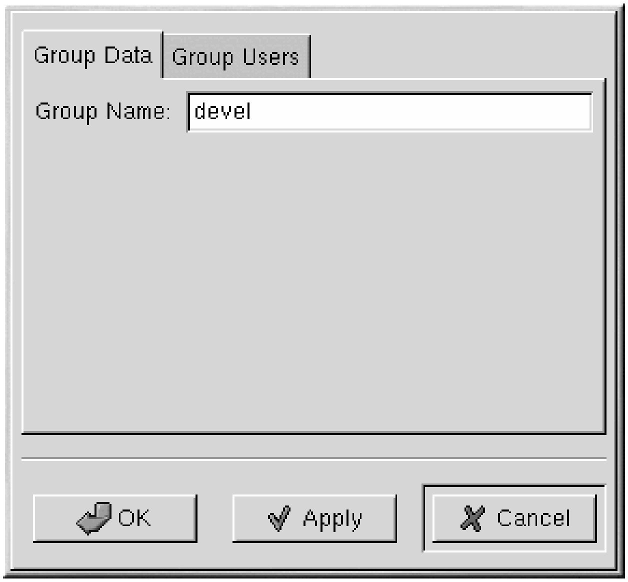


To add users to the group, refer to Section 19.4, *Modifying Group Properties*.

19.4 Modifying Group Properties

To view the properties of an existing group, select the group from the group list and click **Properties** from the button menu (or choose **Action** => **Properties** from the pull-down menu). A window similar to Figure 19–3, *User Properties* will appear.

Figure 19–5 Group Properties



Group Name: devel

The **Group Users** tab displays which users are members of the group. Select additional users to add them to the group, and unselect users to remove from the group. Click **OK** or **Apply** to modify the users in the group.

20 Gathering System Information

Before you learn how to configure your system, you should learn how to gather essential system information. For example, you should know how to find the amount of free memory, how your hard drive is partitioned, and what processes are running. This chapter discusses how to retrieve this type of information from your Red Hat Linux system using simple commands and a few simple programs.

20.1 System Processes

The `ps ax` command displays a list of current system processes, including processes owned by other users. To display the owner of the processes along with the processes use the command `ps aux`. This list is a static list; in other words, it is a snapshot of what is running when you invoked the command. If you want a constantly updated list of running processes, use `top` as described below.

You can use the `ps` command in combination with the `grep` command to see if a process is running. For example, to determine if Gnome-RPM is still running, use the command `ps ax | grep gnorpm`.

The `top` command displays currently running processes and important information about them including their memory and CPU usage. The list is both real-time and interactive. An example of `top`'s output is provided as follows:

```
  6:14pm  up 2 days, 19:29,  5 users,  load average: 0.10, 0.06, 0.07
71 processes: 68 sleeping, 2 running, 1 zombie, 0 stopped
CPU states:  2.7% user,  0.5% system,  0.0% nice, 96.6% idle
Mem:   256812K av,  252016K used,    4796K free,   97228K shrd,   43300K buff
Swap:  265032K av,    1328K used,  263704K free                   86180K cached

  PID USER     PRI  NI  SIZE  RSS SHARE STAT %CPU %MEM   TIME COMMAND
15775 joe        5   0 11028  10M  3192 S     1.5  4.2   0:46 emacs
14429 root      15   0 63620  62M  3284 R     0.5 24.7  63:33 X
17372 joe       11   0  1056 1056   840 R     0.5  0.4   0:00 top
17356 joe        2   0  4104 4104  3244 S     0.3  1.5   0:00 gnome-terminal
14461 joe        1   0  3584 3584  2104 S     0.1  1.3   0:17 sawfish
    1 root       0   0   544  544   476 S     0.0  0.2   0:06 init
    2 root       0   0     0    0     0 SW    0.0  0.0   0:00 kflushd
    3 root       1   0     0    0     0 SW    0.0  0.0   0:24 kupdate
    4 root       0   0     0    0     0 SW    0.0  0.0   0:00 kpiod
    5 root       0   0     0    0     0 SW    0.0  0.0   0:29 kswapd
  347 root       0   0   556  556   460 S     0.0  0.2   0:00 syslogd
  357 root       0   0   712  712   360 S     0.0  0.2   0:00 klogd
  372 bin        0   0   692  692   584 S     0.0  0.2   0:00 portmap
  388 root       0   0     0    0     0 SW    0.0  0.0   0:00 lockd
  389 root       0   0     0    0     0 SW     0.0  0.0   0:00 rpciod
  414 root       0   0   436  432   372 S      0.0  0.1   0:00 apmd
  476 root       0   0   592  592   496 S      0.0  0.2   0:00 automount
```

To exit `top`, press the [q] key.

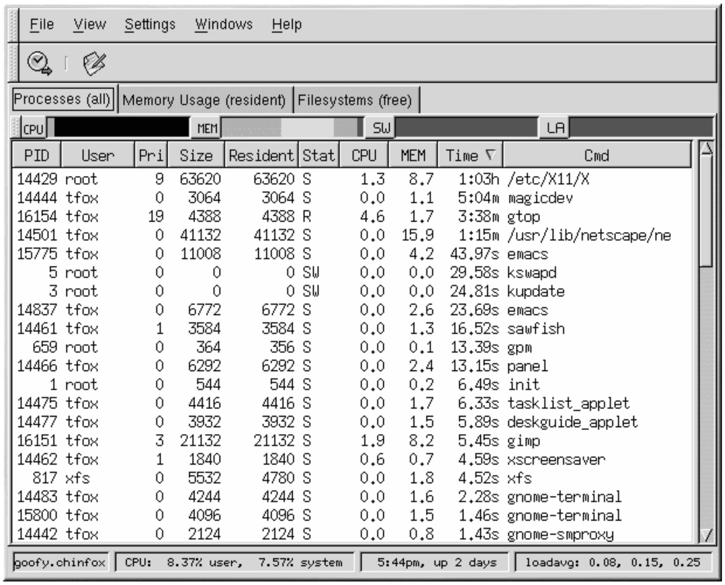
Useful interactive commands that you can use with `top` include the following:

Table 20–1 Interactive `top` commands

Command	Description
[Space]	Immediately refresh the display
[h]	Display a help screen
[k]	Kill a process. You will be prompted for the process ID and the signal to send to it.
[n]	Change the number of processes displayed. You will be prompted to enter the number.
[u]	Sort by user.
[M]	Sort by memory usage.
[P]	Sort by CPU usage.

If you would like to use a graphical interface for `top`, you can use GNOME System Monitor. To start it, go to the **GNOME Main Menu** Button => **Programs** => **System** => **System Monitor** or type `gtop` at a shell prompt.

Figure 20–1 GNOME System Monitor



PID	User	Pri	Size	Resident	Stat	CPU	MEM	Time ▽	Cmd
14429	root	9	63620	63620	S	1.3	8.7	1:03h	/etc/X11/X
14444	tfox	0	3064	3064	S	0.0	1.1	5:04m	magicdev
16154	tfox	19	4388	4388	R	4.6	1.7	3:38m	gtop
14501	tfox	0	41132	41132	S	0.0	15.9	1:15m	/usr/lib/netscape/ne
15775	tfox	0	11008	11008	S	0.0	4.2	43.97s	emacs
5	root	0	0	0	SW	0.0	0.0	29.58s	kswapd
3	root	0	0	0	SW	0.0	0.0	24.81s	kupdate
14837	tfox	0	6772	6772	S	0.0	2.6	23.69s	emacs
14461	tfox	1	3584	3584	S	0.0	1.3	16.52s	sawfish
659	root	0	364	356	S	0.0	0.1	13.39s	gpm
14466	tfox	0	6292	6292	S	0.0	2.4	13.15s	panel
1	root	0	544	544	S	0.0	0.2	6.49s	init
14475	tfox	0	4416	4416	S	0.0	1.7	6.33s	tasklist_applet
14477	tfox	0	3932	3932	S	0.0	1.5	5.89s	deskguide_applet
16151	tfox	3	21132	21132	S	1.9	8.2	5.45s	gimp
14462	tfox	1	1840	1840	S	0.6	0.7	4.59s	xscreensaver
817	xfs	0	5532	4780	S	0.0	1.8	4.52s	xfs
14483	tfox	0	4244	4244	S	0.0	1.6	2.28s	gnome-terminal
15800	tfox	0	4096	4096	S	0.0	1.5	1.46s	gnome-terminal
14442	tfox	0	2124	2124	S	0.0	0.8	1.43s	gnome-smproxy

goofy.chinfox | CPU: 8.37% user, 7.57% system | 5:44pm, up 2 days | loadavg: 0.08, 0.15, 0.25

20.2 Memory Usage

The `free` command displays the total amount of physical memory and swap space for the system as well as the amount of memory that are used, free, shared, in kernel buffers, and cached.

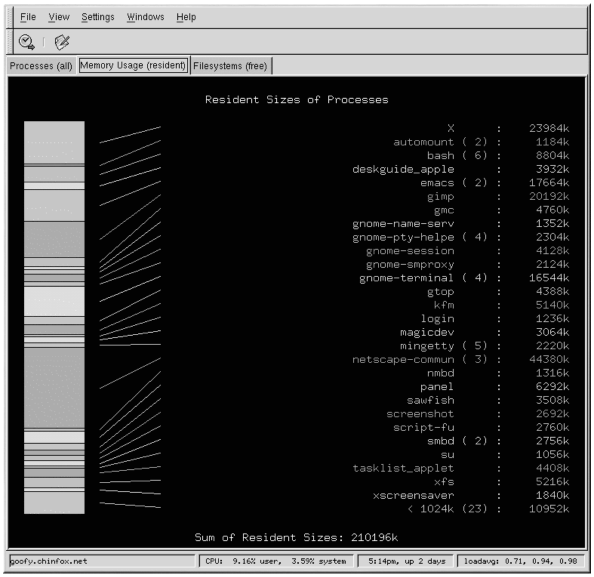
```
              total       used       free     shared    buffers     cached
Mem:         256812     240668      16144     105176      50520      81848
-/+ buffers/cache:      108300     148512
Swap:        265032        780     264252
```

The command `free -m` shows the same information in megabytes, which are easier to read.

```
              total       used       free     shared    buffers     cached
Mem:            250        235         15        102         49         79
-/+ buffers/cache:         105        145
Swap:           258          0        258
```

If you would like to use a graphical interface with `free`, you can use GNOME System Monitor. To start it, go to the **GNOME Main Menu** Button => **Programs** => **System** => **System Monitor** or type `gtop` at a shell prompt. Then choose the **Memory Usage** tab.

Figure 20–2 GNOME System Monitor



```
                     Resident Sizes of Processes

                                       X         :    23984k
                               automount ( 2)    :     1184k
                                    bash ( 6)    :     8804k
                         deskguide_apple         :     3932k
                                   emacs ( 2)    :    17664k
                                    gimp         :    20192k
                                     gmc         :     4760k
                         gnome-name-serv         :     1352k
                         gnome-pty-helpe ( 4)    :     2304k
                           gnome-session         :     4128k
                           gnome-smproxy         :     2124k
                          gnome-terminal ( 4)    :    16544k
                                    gtop         :     4388k
                                     kfm         :     5140k
                                   login         :     1236k
                                magicdev         :     3064k
                                mingetty ( 5)    :     2220k
                         netscape-commun ( 3)    :    44380k
                                    nmbd         :     1316k
                                   panel         :     6292k
                                 sawfish         :     3508k
                              screenshot         :     2692k
                               script-fu         :     2760k
                                    smbd ( 2)    :     2756k
                                      su         :     1056k
                         tasklist_applet         :     4408k
                                     xfs         :     5216k
                             xscreensaver        :     1840k
                                 < 1024k (23)    :    10952k

                   Sum of Resident Sizes: 210196k
```

```
goofy.chinfox.net | CPU:  9.16% user,  3.59% system | 5:14pm, up 2 days | loadavg: 0.71, 0.94, 0.98
```

20.3 Filesystems

The df command reports the system's disk space usage. If you type the command df at a shell prompt, the output looks similar to the following:

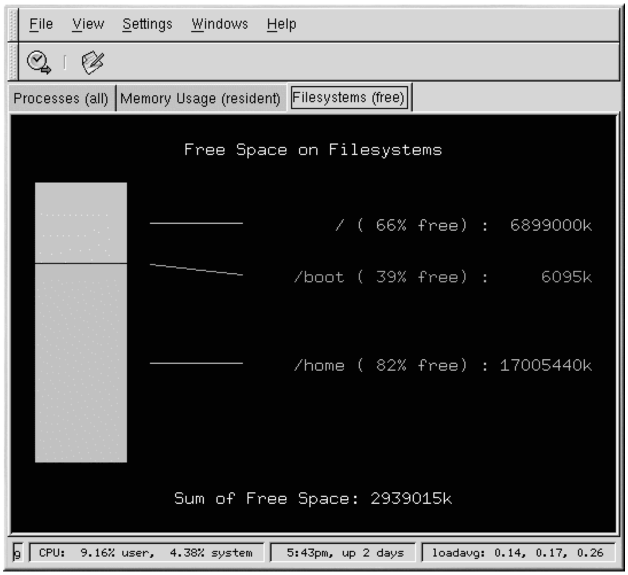
```
Filesystem           1k-blocks      Used Available Use% Mounted on
/dev/hda2             10325716   2902060   6899140  30% /
/dev/hda1                15554      8656      6095  59% /boot
/dev/hda3             20722644   2664256  17005732  14% /home
```

By default, this utility shows the partition size in 1 kilobyte blocks and the amount of used and available disk space in kilobytes. To view the information in megabytes and gigabytes, use the command df -h. The -h argument stands for human-readable format. The output looks similar to the following:

```
Filesystem            Size  Used Avail Use% Mounted on
/dev/hda2             9.8G  2.8G  6.5G  30% /
/dev/hda1              15M  8.5M  5.9M  59% /boot
/dev/hda3              20G  2.6G   16G  14% /home
```

To view the system's disk space usage in a graphical format, use the **Filesystems** tab in the GNOME System Monitor. To start it, go to the **GNOME Main Menu** Button => **Programs** => **System** => **System Monitor** or type `gtop` at a shell prompt. Then choose the **Filesystems** tab.

Figure 20–3 GNOME System Monitor



```
Free Space on Filesystems

/ ( 66% free) :  6899000k
/boot ( 39% free) :     6095k
/home ( 82% free) : 17005440k

Sum of Free Space: 2939015k
```

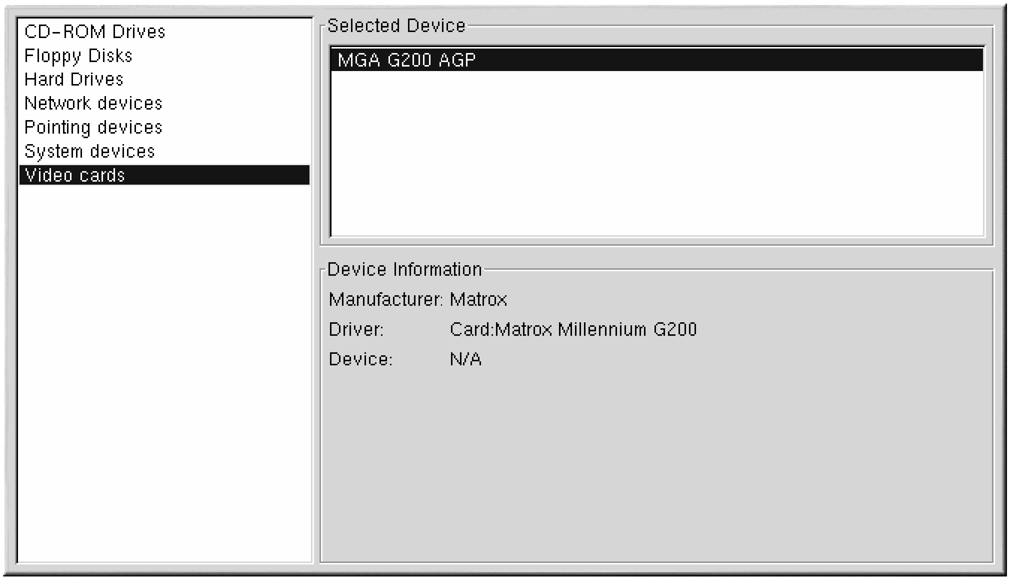
```
CPU:  9.16% user,  4.38% system | 5:43pm, up 2 days | loadavg: 0.14, 0.17, 0.26
```

The `du` command displays the estimated amount of space being used by files in a directory. If you type `du` at a shell prompt, the disk usage for each of the subdirectories will be displayed in a list. The grand total for the current directory and subdirectories will also be shown, as the last line in the list. If you do not want to see all the subdirectories, use the command `du -hs` to see only the grand total for the directory in human-readable format. Use the `du --help` command to see more options.

20.4 Hardware

If you are having trouble configuring your hardware or just want to know what hardware is in your system, you can use the Hardware Browser application to display the hardware that can be probed. To start the program, type `hwbrowser` at a shell prompt. As shown in Figure 20–4, *Hardware Browser*, it displays your CD-ROM devices, floppy disks, hard drives and their partitions, network devices, pointing devices, system devices, and video cards. Click on the category name in the left menu, and the information will be displayed.

Figure 20–4 Hardware Browser



Manufacturer: Matrox
Driver: Card:Matrox Millennium G200
Device: N/A

You can also use the `lspci` command to list all PCI devices. Use the command `lspci -v` for more verbose information or `lspci --v` for very verbose output.

20.5 Sysreport

Sysreport is a system utility created to collect important system data, in order to assist the Red Hat Technical Support and Development Teams in solving customer problems. Sysreport gathers as much system information as is possible, while avoiding certain actions: the creation of a very large file; the invasion of the user's privacy; and the collection of information that could be detrimental to the integrity of the system.

To start Sysreport, you must be logged in as root. As root, at a shell prompt type the command `sysreport`.

You will then see the following message:

```
This utility will go through and collect some detailed information
about the hardware and setup of your Red Hat Linux system.
This information will be used to diagnose problems with your system,
and will be considered confidential information.  Red Hat will use
this information for diagnostic purposes ONLY.
```

```
Please wait while we collect information about your system.

This process may take awhile to complete....
No changes will be made to your system during this process.

NOTE: You can safely ignore a failed message.This only means a file
we were checking for did not exist.

Press ENTER to continue, or CTRL-C to quit.
```

As the message says, ignore any failed messages. Sysreport checks for all possible Red Hat Linux system packages. If you do not have every Red Hat Linux package installed, you will see failed messages.

After pressing [Enter], Sysreport will gather information about your system's configuration. When it is finished, you will see the following message:

```
Enter your first initial and last name with no spaces (example: jsmith):
```

Enter the information requested, and press [Enter]. Sysreport will place a compressed TAR file in the `/tmp` directory beginning with the initial and last name you just entered. You will see a message telling you to email this file to the Red Hat support team. However, even if you do not need support, you can use this information to back up most of your system's configuration.

Use the command `tar ztvf` *`filename`* with the name of the compressed TAR file that you created to display a list of its contents.

20.6 Additional Resources

To learn more about gathering system information, refer to the following resources.

20.6.1 Installed Documentation

- `ps --help` — The `ps --help` displays a list of options that can be used with `ps`.
- `top` manual page — Type `man top` to learn more about `top` and its many options.
- `free` manual page — type `man free` to learn more about `free` and its many options.
- `df` manual page — Type `man df` to learn more about the `df` command and its many options.
- `du` manual page — Type `man du` to learn more about the `du` command and its many options.
- `/proc` — The contents of the `/proc` directory can also be used to gather more detailed system information. Refer to the *Official Red Hat Linux Reference Guide* for additional information about the `/proc` directory.

20.6.2 Useful Websites

- http://www.ibiblio.org/shadow/sysreport/ — The Sysreport Web page provides the latest version and instructions.

21 Printer Configuration

Red Hat Linux no longer includes printtool. The printconf utility has replaced printtool. The printconf utility maintains the `/etc/printcap` configuration file, print spool directories, and print filters.

To use printconf, you must be running the X Window System and have root privileges. To start printconf, use one of the following methods:

- On the GNOME desktop, go to the **Main Menu Button** (on the Panel) => **Programs** => **System** => **Printer Configuration**
- On the KDE desktop, go to the **Main Menu Button** (on the Panel) => **Red Hat** => **System** => **Printer Configuration**.
- Type the command `printconf-gui` at a shell prompt (for example, in an XTerm or a GNOME terminal). [1]

You can also run printconf as a text-based application if you do not have the X Window System installed, or you just prefer the text-based interface. To run it, log in as root (or use the command `su` to temporarily change to the root user), and type the command `/usr/sbin/printconf-tui` from a shell prompt.

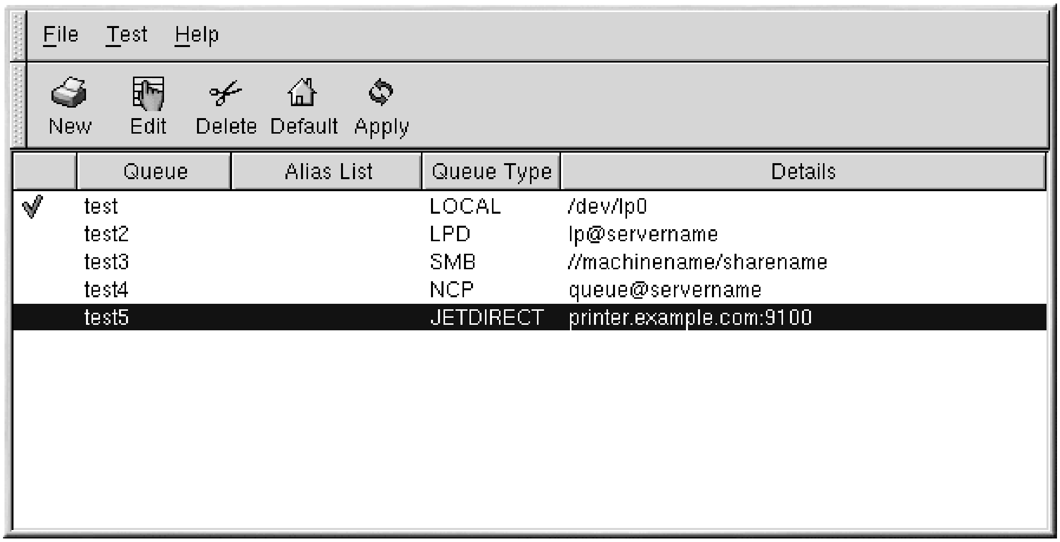
Do Not Edit `/etc/printcap`

Do not edit the `/etc/printcap` file. Each time the printer daemon (`lpd`) is started or restarted, a new `/etc/printcap` file is dynamically created.

If you want to add a printer without using printconf, edit the `/etc/printcap.local` file. The entries in `/etc/printcap.local` are not displayed in printconf but are read by the printer daemon. If you upgrade your system from a previous version of Red Hat Linux, your existing configuration file is converted to the new format used by printconf. Each time a new configuration file is generated by printconf, the old file is saved as `/etc/printcap.old`.

[1] If you type `printtool` at a shell prompt, printconf will start.

Figure 21–1 printconf



	Queue	Alias List	Queue Type	Details
✓	test		LOCAL	/dev/lp0
	test2		LPD	lp@servername
	test3		SMB	//machinename/sharename
	test4		NCP	queue@servername
	test5		JETDIRECT	printer.example.com:9100

Five types of print queues can be configured with printconf:

- **Local Printer** — a printer attached directly to your computer through a parallel or USB port. In the main printer list as shown in Figure 21–1, *printconf*, the **Queue Type** for a local printer is set to **LOCAL**.
- **Unix Printer (lpd Spool)** — a printer attached to a different UNIX system that can be accessed over a TCP/IP network (or example, a printer attached to another Red Hat Linux system on your network). In the main printer list as shown in Figure 21–1, *printconf*, the **Queue Type** for a remote UNIX printer is set to **LPD**.
- **Windows Printer (SMB)** — a printer attached to a different system which is sharing a printer over a SMB network (for example, a printer attached to a Microsoft Windows machine). In the main printer list as shown in Figure 21–1, *printconf*, the **Queue Type** for a remote Windows printer is set to **SMB**.
- **Novell Printer (NCP Queue)** — a printer attached to a different system which uses Novell's NetWare network technology. In the main printer list as shown in Figure 21–1, *printconf*, the **Queue Type** for a remote Novell printer is set to **NCP**.
- **JetDirect Printer** — a printer connected directly to the network instead of to a computer. In the main printer list as shown in Figure 21–1, *printconf*, the **Queue Type** for a JetDirect printer is set to **JETDIRECT**.

Important

If you add a new print queue or modify an existing one, you need to restart the printer daemon (`lpd`) for the changes to take effect.

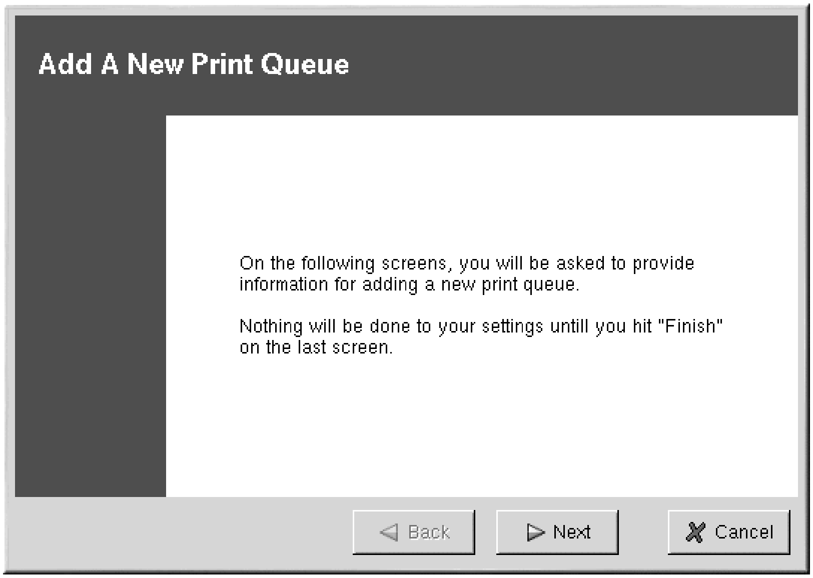
Clicking the **Apply** button saves any changes that you have made and restarts the printer daemon. The changes are not written to the `/etc/printcap` configuration file until the printer daemon (`lpd`) is restarted. Alternatively, you can choose **File** => **Save Changes** and then choose **File** => **Restart lpd** to save your changes and then restart the printer daemon.

If a printer appears in the main printer list with the **Queue Type** set to **INVALID**, the printer configuration is missing options that are required for the printer to function properly. To remove this printer from the list, select it from the list and click the **Delete** button.

21.1 Adding a Local Printer

To add a local printer such as one attached to the parallel port or USB port of your computer, click the **New** button in the main printconf window. The window shown in Figure 21–2, *Adding a Printer* will appear. Click **Next** to proceed.

Figure 21–2 Adding a Printer



Add A New Print Queue

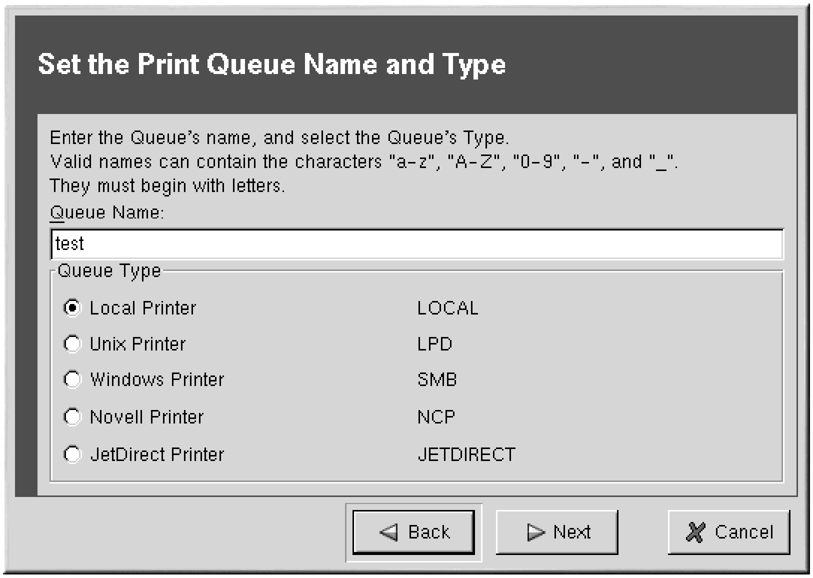
On the following screens, you will be asked to provide information for adding a new print queue.

Nothing will be done to your settings untill you hit "Finish" on the last screen.

You will then see the screen shown in Figure 21–3, *Adding a Local Printer*. Enter a unique name for the printer in the **Queue Name** text field. This can be any descriptive name for your printer. The printer name cannot contain spaces and must begin with a letter a through z or A through Z. The valid characters are a through z, A through Z, 0 through 9, -, and _.

Select **Local Printer** from the **Queue Type** menu, and click **Next**.

Figure 21–3 Adding a Local Printer



Set the Print Queue Name and Type

Enter the Queue's name, and select the Queue's Type.
Valid names can contain the characters "a-z", "A-Z", "0-9", "-", and "_".
They must begin with letters.

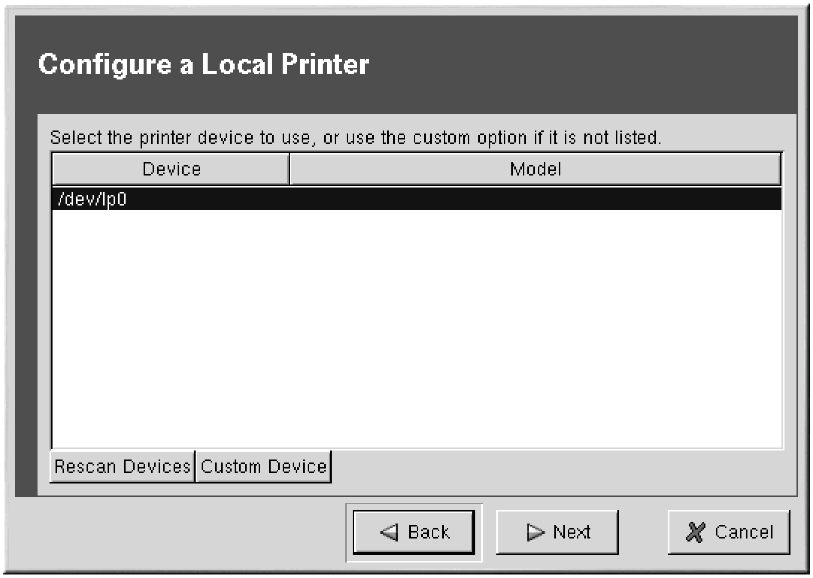
Queue Name: test

Queue Type

● Local Printer	LOCAL
○ Unix Printer	LPD
○ Windows Printer	SMB
○ Novell Printer	NCP
○ JetDirect Printer	JETDIRECT

printconf attempts to detect your printer device and display it as shown in Figure 21–4, *Choosing a Printer Device*. If your printer device is not shown, click **Custom Device**. Type the name of your printer device and click **OK** to add it to the printer device list. A printer device attached to the parallel port is usually referred to as `/dev/lp0`. A printer device attached to the USB port is usually referred to as `/dev/usblp0`. After selecting your printer device, click **Next**.

Figure 21–4 Choosing a Printer Device



Configure a Local Printer

Select the printer device to use, or use the custom option if it is not listed.

Device	Model
/dev/lp0	

Next, printconf will try to detect which printer is attached to the printer device. Skip to Section 21.6, *Selecting the Print Driver and Finishing* to continue.

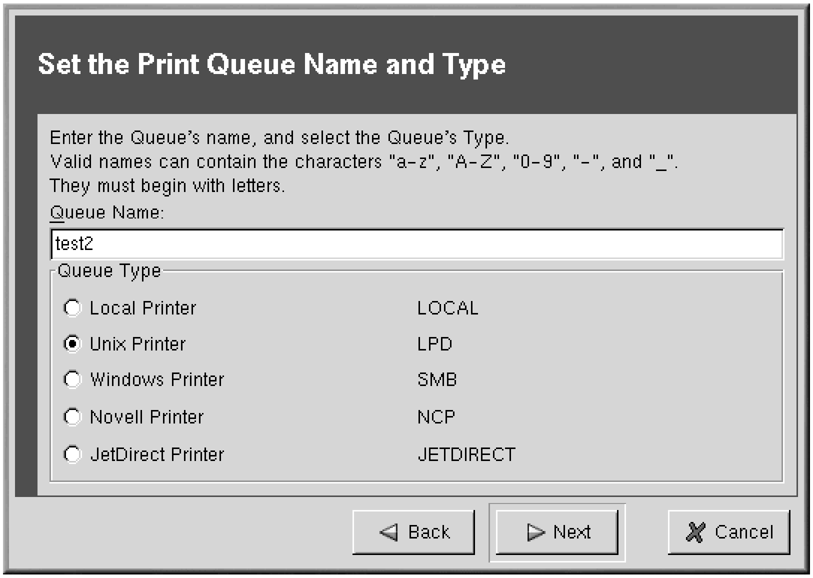
21.2 Adding a Remote UNIX Printer

To add a remote UNIX printer, such as one attached to a different Linux system on the same network, click the **New** button in the main printconf window. The window shown in Figure 21–2, *Adding a Printer* will appear. Click **Next** to proceed.

You will then see the screen shown in Figure 21–5, *Adding a Remote Printer*. Enter a unique name for the printer in the **Queue Name** text field. The printer name can not contain spaces and must begin with a letter a through z or A through Z. The valid characters are a through z, A through Z, 0 through 9, -, and _.

Select **Unix Printer** from the **Queue Type** menu, and click **Next**.

Figure 21–5 Adding a Remote Printer



Set the Print Queue Name and Type

Enter the Queue's name, and select the Queue's Type.
Valid names can contain the characters "a-z", "A-Z", "0-9", "-", and "_".
They must begin with letters.

Queue Name:

test2

Queue Type

○ Local Printer	LOCAL
● Unix Printer	LPD
○ Windows Printer	SMB
○ Novell Printer	NCP
○ JetDirect Printer	JETDIRECT

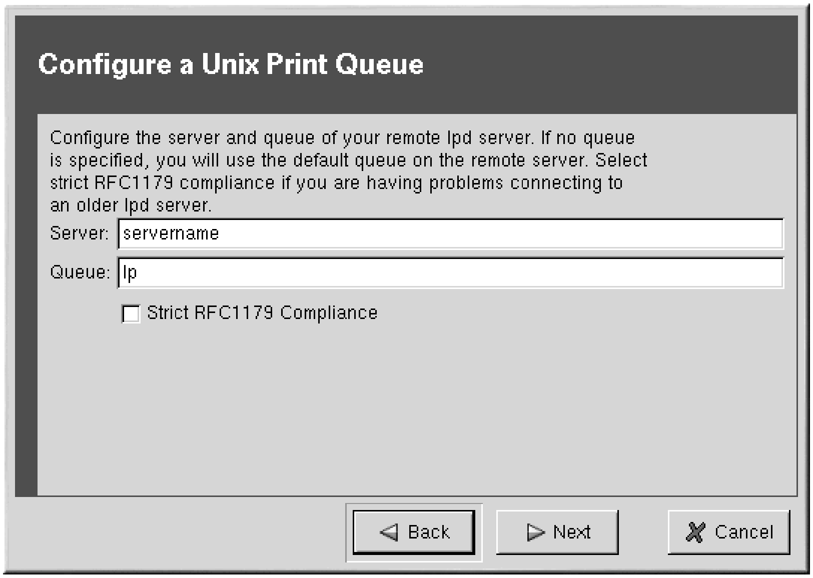
Text fields for the following options appears as shown in Figure 21–6, *Choosing the Printer Server*:

- **Server** — The hostname or IP address of the remote machine to which the printer is attached.
- **Queue** — The remote printer queue. The default printer queue is usually `lp`.

By default, the **Strict RFC1179 Compliance** option is not chosen. If you are having problems printing to a non-Linux `lpd` queue, choose this option to disable enhanced LPRng printing features.

Click **Next** to continue.

Figure 21–6 Choosing the Printer Server



Configure a Unix Print Queue

Configure the server and queue of your remote lpd server. If no queue is specified, you will use the default queue on the remote server. Select strict RFC1179 compliance if you are having problems connecting to an older lpd server.

Server: servername

Queue: lp

□ Strict RFC1179 Compliance

The next step is to select the type of printer that is connected to the remote system. Skip to Section 21.6, *Selecting the Print Driver and Finishing* to continue.

Important

The remote machine must be configured to allow the local machine to print on the desired queue. As root, create the file `/etc/hosts.lpd` on the remote machine to which the printer is attached. On separate lines in the file, add the IP address or hostname of each machine which should have printing privileges.

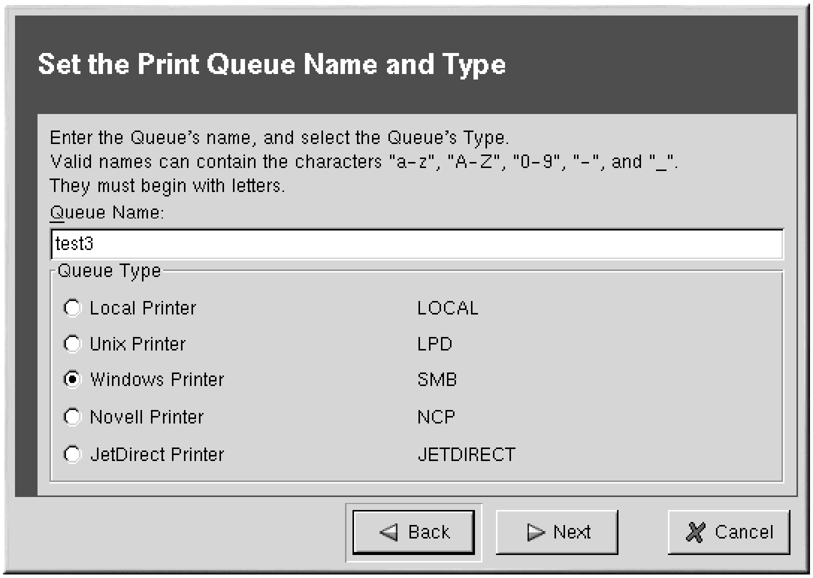
21.3 Adding a Samba (SMB) Printer

To add printer which is accessed using the SMB protocol, click the **New** button in the main printconf window. The window shown in Figure 21–2, *Adding a Printer* will appear. Click **Next** to proceed.

You will see the screen shown in Figure 21–7, *Adding a SMB Printer*. Enter a unique name for the printer in the **Queue Name** text field. The printer name cannot contain spaces and must begin with a letter a through z or A through Z. The valid characters are a through z, A through Z, 0 through 9, -, and _.

Select **Windows Printer** from the **Queue Type** menu, and click **Next**. If the printer is attached to a Microsoft Windows system, choose this queue type.

Figure 21–7 Adding a SMB Printer



Set the Print Queue Name and Type

Enter the Queue's name, and select the Queue's Type.
Valid names can contain the characters "a-z", "A-Z", "0-9", "-", and "_".
They must begin with letters.

Queue Name: test3

Queue Type

○ Local Printer LOCAL
○ Unix Printer LPD
● Windows Printer SMB
○ Novell Printer NCP
○ JetDirect Printer JETDIRECT

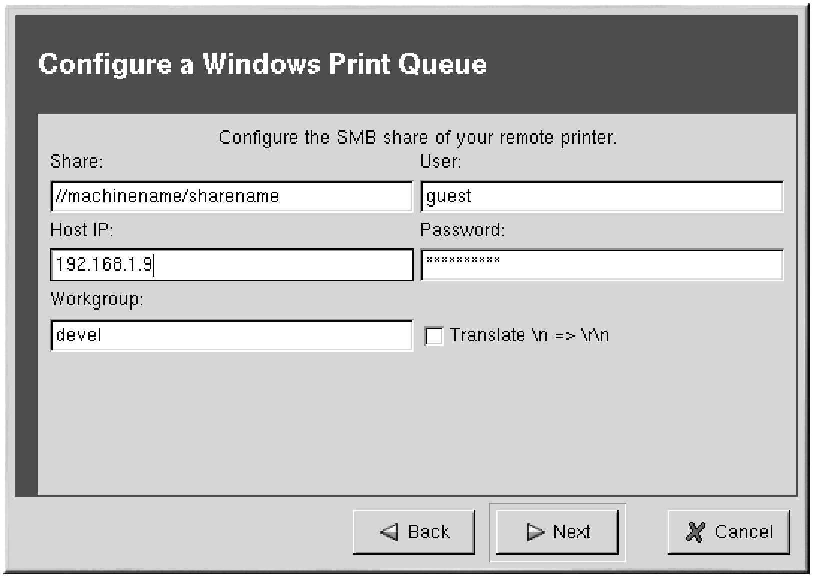
Text fields for the following options appear as shown in Figure 21–8, *Choosing the Print Server*:

- **Share** — The name of the shared printer on which you want to print. This name must be the same name defined as the Samba printer on the remote Windows machine. Notice the syntax of *//machinename/sharename*.
- **User** — The name of the user you must log in as to access the printer. This user must exist on the Windows system, and the user must have permission to access the printer. The user name is typically **`guest`** for Windows servers, or **nobody** for Samba servers.
- **Host IP** — The hostname or IP address of the remote system that is sharing the SMB printer.
- **Password** — The password (if required) for the user specified in the **User** field.
- **Workgroup** — The name of the workgroup on the machine running Samba.

Click the **Translate \n => \r\n** button to translate the end of line characters to a form that is readable by a Microsoft Windows system.

Click **Next** to continue.

Figure 21–8 Choosing the Print Server



Configure a Windows Print Queue

Configure the SMB share of your remote printer.

Share: //machinename/sharename

User: guest

Host IP: 192.168.1.9

Password: **********

Workgroup: devel

□ Translate \n => \r\n

The next step is to select the type of printer that is connected to the remote SMB system. Skip to Section 21.6, *Selecting the Print Driver and Finishing* to continue.

Note

If you require a username and password for an SMB (LAN Manager) or NCP (NetWare) print queue, they are stored unencrypted in a local script, Thus, it is possible for another person to learn the username and password. To avoid this, the username and password to use the printer should be different from the username and password used for the user's account on the local Red Hat Linux system. If they are different, then the only possible security compromise would be unauthorized use of the printer. If there are file shares from the SMB server, it is recommended that they also use a different password than the one for the print queue.

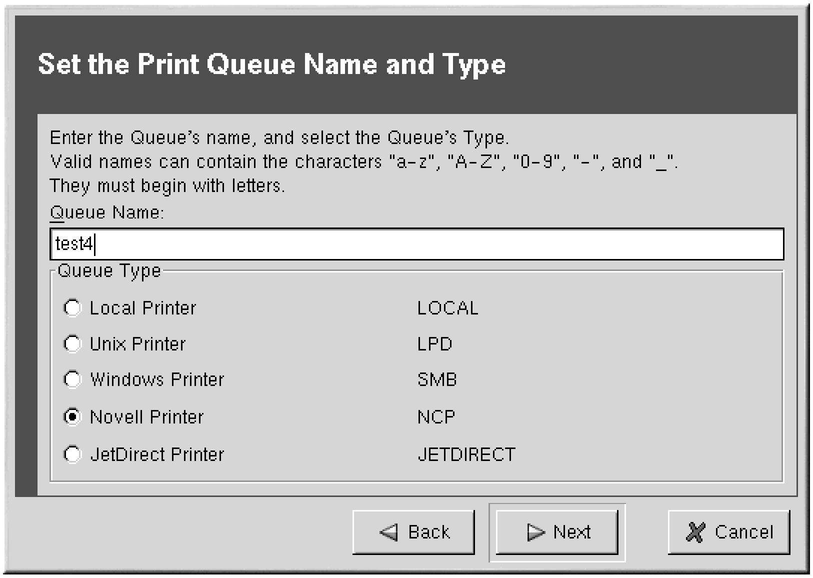
21.4 Adding a Novell NetWare (NCP) Printer

To add a Novell NetWare (NCP) printer, click the **New** button in the main printconf window. The window shown in Figure 21–1, *printconf* will appear. Click **Next** to proceed.

You will see the screen shown in Figure 21–9, *Adding an NCP Printer*. Enter a unique name for the printer in the **Queue Name** text field. The printer name cannot contain spaces and must begin with a letter a through z or A through Z. The valid characters are a through z, A through Z, 0 through 9, -, and _.

Select **Novell Printer** from the **Queue Type** menu, and click **Next**.

Figure 21–9 Adding an NCP Printer



Set the Print Queue Name and Type

Enter the Queue's name, and select the Queue's Type.
Valid names can contain the characters "a-z", "A-Z", "0-9", "-", and "_".
They must begin with letters.

Queue Name: test4

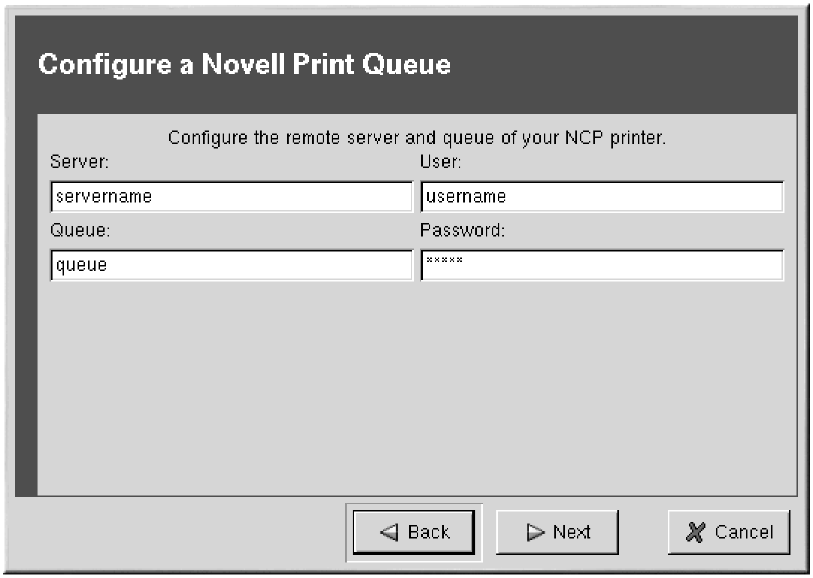
Queue Type

○ Local Printer LOCAL
○ Unix Printer LPD
○ Windows Printer SMB
● Novell Printer NCP
○ JetDirect Printer JETDIRECT

Text fields for the following options appear below the **Queue Type** menu as shown in Figure 21–10, *Choosing the Print Server*:

- **Server** — The hostname or IP address of the NCP system to which the printer is attached.
- **Queue** — The remote queue for the printer on the NCP system.
- **User** — The name of the user you must log in as to access the printer.
- **Password** — The password for the user specified in the **User** field above.

Figure 21–10 Choosing the Print Server



Configure a Novell Print Queue

Configure the remote server and queue of your NCP printer.

Server: servername

User: username

Queue: queue

Password: *****

The next step is to select the type of printer that is connected to the remote NCP system. Skip to Section 21.6, *Selecting the Print Driver and Finishing* to continue.

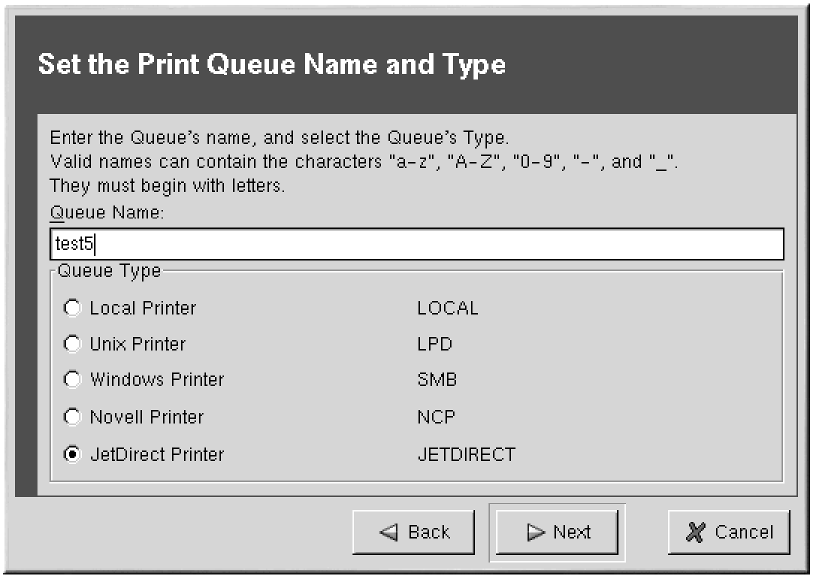
21.5 Adding a JetDirect Printer

To add a JetDirect printer, click the **New** button in the main printconf window. The window shown in Figure 21–1, *printconf* will appear. Click **Next** to proceed.

You will see the screen shown in Figure 21–11, *Adding a JetDirect Printer*. Enter a unique name for the printer in the **Queue Name** text field. The printer name cannot contain spaces and must begin with a letter a through z or A through Z. The valid characters are a through z, A through Z, 0 through 9, -, and _.

Select **JetDirect Printer** from the **Queue Type** menu, and click **Next**.

Figure 21–11 Adding a JetDirect Printer



Set the Print Queue Name and Type

Enter the Queue's name, and select the Queue's Type.
Valid names can contain the characters "a-z", "A-Z", "0-9", "-", and "_".
They must begin with letters.

Queue Name: test5

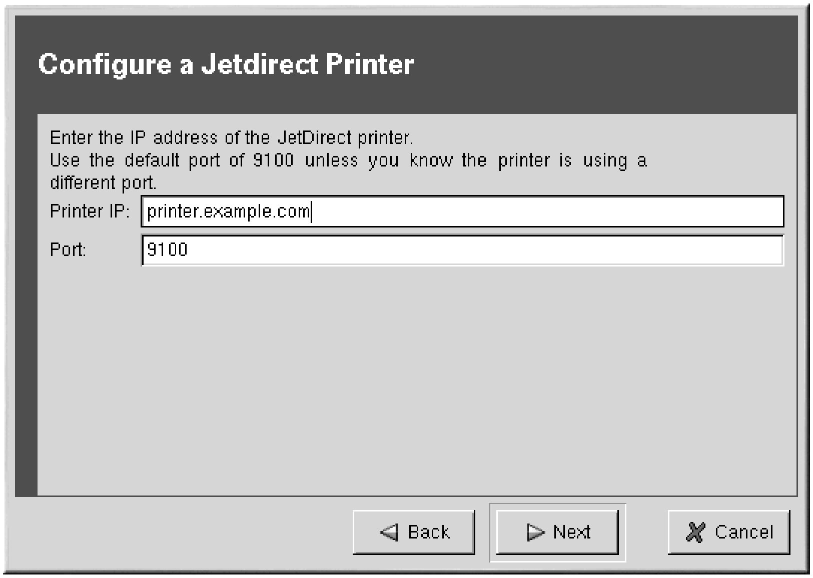
Queue Type

○ Local Printer	LOCAL
○ Unix Printer	LPD
○ Windows Printer	SMB
○ Novell Printer	NCP
● JetDirect Printer	JETDIRECT

Text fields for the following options appear below the **Queue Type** menu as shown in Figure 21–12, *Choosing a Print Server*:

- **Printer IP** — The hostname or IP address of the JetDirect printer.
- **Port** — The port on the JetDirect printer that is listening for print jobs.

Figure 21–12 Choosing a Print Server



Configure a Jetdirect Printer

Enter the IP address of the JetDirect printer.
Use the default port of 9100 unless you know the printer is using a different port.

Printer IP: printer.example.com

Port: 9100

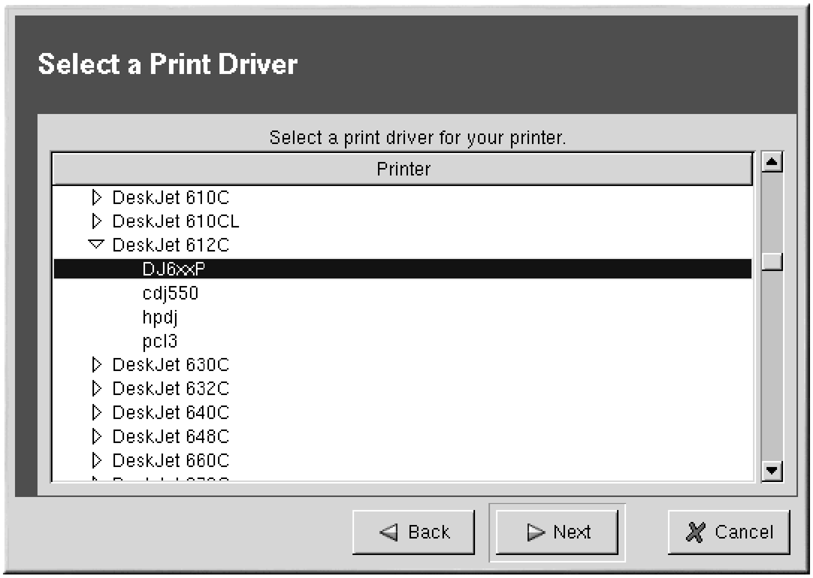
The next step is to select the type of printer that is connected to the JetDirect system. Skip to Section 21.6, *Selecting the Print Driver and Finishing* to continue.

21.6 Selecting the Print Driver and Finishing

After selecting the queue type of the printer, the next step in adding a printer is to select the print driver.

You will see a window similar to Figure 21–13, *Selecting a Print Driver*. If you are configuring a local printer, select the print driver from the list. The printers are divided by manufacturers. Click the arrow beside the manufacturer for your printer. Find your printer from the expanded list, and click the arrow beside the printer name. A list of drivers for your printer will appear. Select one. If you do not know which one to use, select the first one in the list. If you are having problems using that driver, edit the print queue in printconf and select a different driver.

Figure 21–13 Selecting a Print Driver



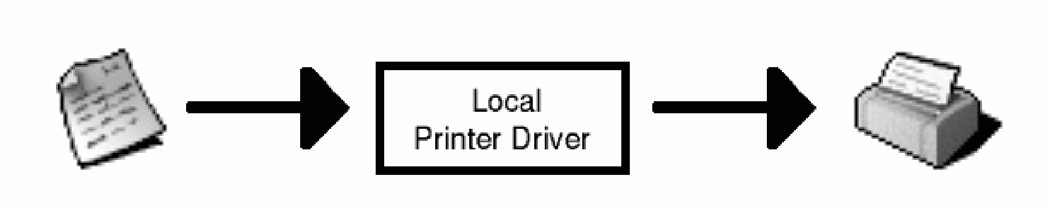
Select a Print Driver

Select a print driver for your printer.

Printer
- ▷ DeskJet 610C
- ▷ DeskJet 610CL
- ▽ DeskJet 612C
  - DJ6xxP
  - cdj550
  - hpdj
  - pcl3
- ▷ DeskJet 630C
- ▷ DeskJet 632C
- ▷ DeskJet 640C
- ▷ DeskJet 648C
- ▷ DeskJet 660C

As shown in Figure 21–14, *Correct Print Driver Configuration*, the print driver processes the data that you want to print into a format the printer can understand. Since a local printer is attached directly to your computer, you need to select a print driver to process the data that is sent to the printer.

Figure 21–14 Correct Print Driver Configuration

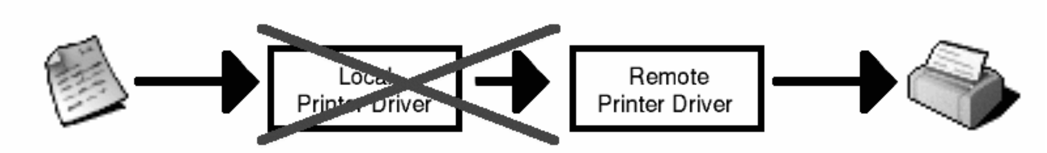


If you are configuring a remote printer (LPD, SMB, or NCP), the remote print server usually has its own print driver. If you select an additional print driver on your local computer, the data will be filtered more than once, and the data will be converted to a format that the printer can not understand.

To make sure the data is not filtered more than once, first try selecting **Raw Print Queue** or **Postscript Printer** if you are configuring a remote printer. After applying the changes, print a test page to test this configuration. If the test fails, the remote print server might not have a print driver configured.

Try selecting a print driver according to the manufacturer and model of the remote printer, applying the changes, and printing a test page.

Figure 21–15 Incorrect Print Driver Configuration



21.6.1 Confirming Printer Configuration

The last step is to confirm your printer configuration. Click **Finish** if this is the printer that you want to add. Click **Back** to modify the printer configuration.

Click the **Apply** button in the main window to save your changes to the `/etc/printcap` configuration file and restart the printer daemon (`lpd`). After applying the changes, print a test page to ensure the configuration is correct. Refer to Section 21.7, *Printing a Test Page* for details.

If you need to print characters beyond the basic ASCII set (including those used for languages such as Japanese), you need to go to your driver options and select **Rerender Postscript**. Refer to Section 21.8, *Modifying Existing Printers* for details. You can also configure options such as paper size if you edit the print queue after adding it.

21.7 Printing a Test Page

After you have configured your printer, you should print a test page to make sure the printer is functioning properly. To print a test page, select the printer that you want to test from the printer list, and select the appropriate test page from the **Test** pulldown menu.

If you change the print driver or modify the driver options, you should print a test page to test the different configuration.

21.8 Modifying Existing Printers

To delete an existing printer, select the printer and click the **Delete** button on the toolbar. The printer will be removed from the printer list. Click **Apply** to save the changes and restart the printer daemon.

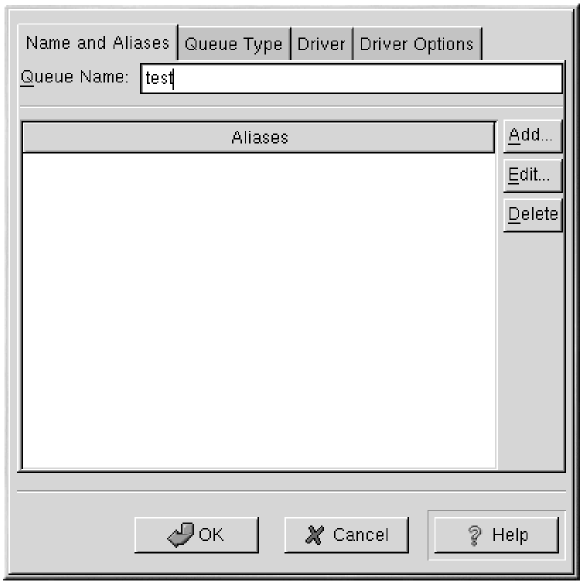
To set the default printer, select the printer from the printer list and click the **Default** button on the toolbar. The default printer icon ✓ appears in the first column of the printer list beside the default printer.

If you want to modify an imported printer's settings, you cannot modify its settings directly. You must override the printer. You can only override an imported printer that has been imported using the alchemist libraries. Imported printers have the ⇋ symbol beside them in the first column of the printer list.

To override the printer, select the printer, and choose **File** => **Override Queue** from the pulldown menu. After overriding a printer, the original imported printer will have the ✱ symbol beside it in the first column of the printer list.

After adding your printer(s), you can edit settings by selecting the printer from the printer list and clicking the **Edit** button. The tabbed window shown in Figure 21–16, *Editing a Printer* will appear. The window contains the current values for the printer that you selected to edit. Make any changes, and click **OK**. Click **Apply** in the main printconf window to save the changes and restart the printer daemon.

Figure 21–16 Editing a Printer



Queue Name: test

21.8.1 Names and Aliases

If you want to rename a printer, change the value of **Queue Name** in the **Names and Aliases** tab. Click **OK** to return to the main window. The name of the printer should change in the printer list. Click **Apply** to save the change and restart the printer daemon.

A printer alias is an alternate name for a printer. To add an alias for an existing printer, click the **Add** button in the **Name and Aliases** tab, enter the name of the alias, and click **OK**. Click **OK** again to return to the main window. Click **Apply** to save the aliases and restart the printer daemon. A printer can have more than one alias.

21.8.2 Queue Type

The **Queue Type** tab shows queue type that you selected when adding the printer and its settings. You can change the queue type of the printer or just change the settings. After making modifications, click **OK** to return to the main window. Click **Apply** to save the change and restart the printer daemon.

Depending on which queue type you choose, you will see different different options. Refer to the appropriate section on adding a printer for a description of the options.

21.8.3 Driver

The **Driver** tab shows which print driver is currently being used. This is the same list that you used when adding the printer. If you change the print driver, click **OK** to return to the main window. Click **Apply** to save the change and restart the printer daemon.

21.8.4 Driver Options

The **Driver Options** tab displays advanced printer options. Options vary for each print driver. Common options include:

- **Send FF** stands for send **form feed**. If the last page of your print job is not ejected from the printer (for example, the form feed light flashes), try selecting this option. If this does not work, try selecting **Send EOT** instead. Some printers require both **Send FF** and **Send EOT** to eject the last page.
- **Send EOT** stands for send **end of transmission**. Refer to **Send FF** above.
- **Assume Unknown Data is Text** should be selected if your print driver does not recognize some of the data sent to it. Only select it if you are having problems printing. If this option is selected, the print driver will assume that any data that it can not recognize is text and try to print it as text. If you select this option and **Convert Text to Postscript**, the print driver will assume the unknown data is text and then convert it to PostScript.
- **Rerender Postscript** should be selected if you are printing characters beyond the basic ASCII set but they are not printing correctly (such as Japanese characters). This option will rerender non-standard PostScript fonts so that they are printed correctly.

  If your printer does not support the fonts you are trying to print, try selecting this option. For example, you should select this option if you are printing Japanese fonts to a non-Japanese printer.

Extra time is required to perform this action. Do not choose it unless you are having problems printing the correct fonts.

- **Convert Text to Postscript** is selected by default. If your printer can print plain text, try unselecting this when printing plain text documents to decrease the time it takes to print.
- **Page Size** allows you to select the paper size for your printer such as US Letter, US Legal, A3, and A4.
- **Effective Filter Locale** defaults to **C**. If you are printing Japanese characters, select **ja_JP**. Otherwise, accept the default of **C**.

If you modify the driver options, click **OK** to return to the main window. Click **Apply** to save the change and restart the printer daemon.

21.9 Saving the Configuration File

When you save your printer configuration using printconf, it creates its own configuration file that is used to create the /etc/printcap file that the printer daemon (lpd) reads. You can use the command line options to save or restore this file. If you save your /etc/printcap file and overwrite your existing /etc/printcap file with the saved file, your printer configuration will not be restored. Each time the printer daemon is restarted, it creates a new /etc/printcap file from the special printconf configuration file. If you have configured a backup system for your configuration files, you should use the following method to save your printer configuration. If you added any custom settings in the /etc/printcap.local file, you should save it as part of your backup system also.

To save your printer configuration, type this command as root:

```
/usr/sbin/printconf-tui --Xexport > settings.xml
```

Your configuration is saved to the file settings.xml.

If you save this file, you can restore your printer settings. This is useful if your printer configuration is deleted, you reinstall Red Hat Linux and do not have your printer configuration file anymore, or you want to use the same printer configuration on multiple systems. To restore the configuration, type this command as root:

```
/usr/sbin/printconf-tui --Ximport < settings.xml
```

If you already have a configuration file (you have configured one or more printers on the system already) and you try to import another configuration file, the existing configuration file will be overwritten. If you want to keep your existing configuration and add the configuration in the saved file, you can merge the files with the following command (as root):

```
/usr/sbin/printconf-tui --Ximport --merge < settings.xml
```

Your printer list will then consist of the printers you configured on the system as well as the printers you imported from the saved configuration file. If the imported configuration file has a print queue with the same name as an existing print queue on the system, the print queue from the imported file will override the existing printer.

After importing the configuration file (with or without the `merge` command), you must restart the printer daemon with the command `/sbin/service lpd restart` or by starting printconf and clicking **Apply**.

21.10 Managing Your Print Jobs

When you send a print job to the printer daemon such as printing text file from Emacs or printing an image from The GIMP, the print job is added to the print spool queue. The print spool queue is a list of print jobs that have been sent to the printer and information about each print request such as the status of the request, the username of the person who sent the request, the hostname of the system that sent the request, the job number, and more. To view the list of print jobs in the print spool, open a shell prompt and type the command `lpq`. The last few lines will look similar to the following:

Example 21–1 Example of `lpq` output

```
Rank   Owner/ID             Class  Job Files        Size Time
active user@localhost+902     A    902 sample.txt   2050 01:20:46
```

If you want to cancel a print job, find the job number of the request with the command `lpq` and then use the command `lprm` *`job number`*. For example, `lprm 902` would cancel the print job in Example 21–1, *Example of `lpq` output*. You must have proper permissions to cancel a print job. You can not cancel print jobs that were started by other users unless you are logged in as root on machine to which the printer is attached.

You can also print a file directly from a shell prompt. For example, the command `lpr sample.txt` will print the text file `sample.txt`. The print filter determines what type of file it is and converts it a format the printer can understand.

21.11 Additional Resources

To learn more about printing on Red Hat Linux, refer to the following resources.

21.11.1 Installed Documentation

- `man printcap` — The manual page for the `/etc/printcap` printer configuration file.
- `map lpr` — The manual page for the `lpr` command that allows you to print files from the command line.
- `man lpd` — The manual page for the printer daemon.

- `man lprm` — The manual page on the command line utility to remove print jobs from the printer spool queue.

21.11.2 Useful Websites

- http://www.linuxprinting.org — *GNU/Linux Printing* contains a large amount information about printing in Linux.

22 Automated Tasks

In Linux, tasks can be configured to run automatically within a given period of time and on given dates. Red Hat Linux comes preconfigured to run certain system tasks to keep your system updated. For example, the slocate database is updated daily. A system administrator can use automated tasks to perform periodic backups, monitor the system, run custom scripts, and more.

22.1 Cron

Cron is a daemon that can be used to execute scheduled tasks according to a combination of the time, day of the month, month, day of the week, and week.

Cron assumes that the system is on continuously. If the system is not on when a task is scheduled it is not executed. To configure tasks based on periods instead of exact times, see Section 22.3, *Anacron*.

To use the cron service, you must have the `vixie-cron` RPM package installed. To determine if the package is installed, use the command `rpm -q vixie-cron`.

22.2 Configuring a Cron Task

The main configuration file for cron, `/etc/crontab`, contains the following lines:

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
```

The first four lines are variables used to configure the environment in which the cron tasks are run. The value of the `SHELL` variable tells the system which shell environment to use (in this example the bash shell), and the `PATH` variable defines the path used to execute commands. The output of the cron tasks are emailed to the username defined with the `MAILTO` variable. If the `MAILTO` variable is defined as an empty string (`MAILTO=""`), email will not be sent. The `HOME` variable can be used to set the home directory to use when executing commands or scripts.

Each line in the `/etc/crontab` file has the format:

```
minute   hour   day   month   dayofweek   command
```

- `minute` — any integer from 0 to 59
- `hour` — any integer from 0 to 23
- `day` — any integer from 1 to 31 (must be a valid day if a month is specified)
- `month` — any integer from 1 to 12 (or the short name of the month such as jan, feb, and so on)
- `dayofweek` — any integer from 0 to 7 where 0 or 7 represents Sunday (or the short name of the week such as sun, mon, and so on)
- `command` — the command to execute. The command can either be a command such as `ls /proc >> /tmp/proc` or the command to execute a custom script that you wrote.

For any of the above values, an asterisk (*) can be used to specify all valid values. For example, an asterisk for the month value means execute the command every month within the constraints of the other values.

A hyphen (-) between integers specifies a range of integers. For example, **`1-4`** means the integers 1, 2, 3, and 4.

A list of values separated by commas (,) specifies a list. For example, **`3, 4, 6, 8`** indicates those four specific integers.

The forward slash (/) can be used to specify step values. The value of an integer can be skipped within a range by following the range with ***`/<integer>`***. For example, **`0-59/2`** can be used to define every other minute in the minute field. Step values can also be used with an asterisk. For instance, the value **`*/3`** can be used in the month field to skip every third month.

Any lines that begin with a hash mark (#) are comments and are not processed.

Example 22–1 Examples of crontabs

```
# record the memory usage of the system every monday
# at 3:30AM in the file /tmp/meminfo
30 3 * * mon cat /proc/meminfo >> /tmp/meminfo
# run custom script the first day of every month at 4:10AM
10 4 1 * * /root/scripts/backup.sh
```

As you can see from the `/etc/crontab` file, it uses the `run-parts` script to execute the scripts in the `/etc/cron.hourly`, `/etc/cron.daily`, `/etc/cron.weekly`, and `/etc/cron.monthly` files on an hourly, daily, weekly, or monthly basis respectively. The files in these directory should be shell scripts.

If a cron tasks needs to be executed on a schedule other than hourly, daily, weekly, or monthly, it can be added to the `/etc/cron.d` directory. All files in this directory use the same syntax as `/etc/crontab`.

The cron daemon checks the `etc/crontab` file, the `etc/cron.d/` directory, and the `/var/spool/cron` directory every minute for any changes. If any changes are found, they are loaded into memory. Thus, the daemon does not need to be restarted if a crontab file is changed.

Users other than root can configure cron tasks by using the `crontab` utility. All user-defined crontabs are stored in the `/var/spool/cron` directory and are executed using the usernames of the users that created them. To create a crontab as a user, login as that user and type the command `crontab -e` to edit the user's crontab using the editor specified by the `VISUAL` or `EDITOR` environment variable. The file uses the same format as `/etc/crontab`. When the changes to the crontab are saved, the crontab is stored according to username and written to the file `/var/spool/cron/`*`username`*.

22.2.1 Starting and Stopping the Service

To start the cron service, use the command `/sbin/service crond start`. To stop the service, use the command `/sbin/service crond stop`. It is recommended that you start the service at boot time. Refer to Chapter 8, *Controlling Access to Services* for details on starting the cron service automatically at boot time.

22.3 Anacron

Anacron is a task scheduler similar to cron except that it does not require the system to run continuously. It can be used to run the daily, weekly, and monthly jobs usually run by cron.

To use the Anacron service, you must have the `anacron` RPM package installed. To determine if the package is installed, use the command `rpm -q anacron`.

22.3.1 Configuration File

Anacron tasks are listed in the configuration file `/etc/anacron`. Each line in the configuration file corresponds to a task and has the format:

```
period   delay   job-identifier  command
```

- `period` — frequency (in days) to execute the command
- `delay` — delay time in minutes
- `job-identifier` — description of the task, used in Anacron messages and as the name of the job's timestamp file, can contain any non-blank characters (except slashes).
- `command` — command to execute

For each tasks, Anacron determines if the task has been executed within the period specified in the `period` field of the configuration file. If it has not been executed within the given period, Anacron executes the command specified in the `command` field after waiting the number of minutes specified in the `delay` field.

After the task is completed, Anacron records the date in a timestamp file in the `/var/spool/anacron` directory. Only the date is used (not the time), and the value of the `job-identifier` is used as the filename for the timestamp file.

Environment variables such as `SHELL` and `PATH` can be defined at the top of `/etc/anacron` as with the cron configuration file.

The default configuration file looks similar to the following:

Figure 22–1 Default anacrontab

```
# /etc/anacrontab: configuration file for anacron

# See anacron(8) and anacrontab(5) for details.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# These entries are useful for a Red Hat Linux system.
1       5       cron.daily              run-parts /etc/cron.daily
7       10      cron.weekly             run-parts /etc/cron.weekly
30      15      cron.monthly    run-parts /etc/cron.monthly
```

As you can see in Figure 22–1, *Default anacrontab*, the anacrontab for Red Hat Linux is configured to make sure the daily, weekly, and monthly cron tasks are run.

22.3.2 Starting and Stopping the Service

To start the anacron service, use the command `/sbin/service anacron start`. To stop the service, use the command `/sbin/service anacron stop`. It is recommended that you start the service at boot time. Refer to Chapter 8, *Controlling Access to Services* for details on starting the anacron service automatically at boot time.

22.4 Additional Resources

To learn more about configuring automated tasks, refer to the following resources.

22.4.1 Installed Documentation

- `cron` man page — overview of cron
- `crontab` man pages in sections 1 and 5 — The man page in section 1 contains an overview of the `crontab` file. The man page in section 5 contains the format for the file and some example entries.

- `anacron` man page — description of anacron and its command line options.
- `anacrontab` man page — brief overview of the anacron configuration file.
- Anacron README file — the Anacron README file located at `/usr/share/doc/anacron-<version>/README` describes Anacron.

23 Ugrading the Kernel

The kernel that comes with Red Hat Linux is custom built by the Red Hat kernel team to ensure its integrity and compatibility with supported hardware. Before Red Hat releases a kernel, it must pass a rigorous set of quality assurance tests. The kernel RPM package now creates the `initrd` image if needed. It is no longer necessary to use the `mkinitrd` command after installing a different kernel if you install the kernel from the Red Hat RPM package.

Official Red Hat Linux kernels are packaged in RPM format so that they are easy to upgrade and verify. To upgrade to a newer version of a Red Hat Linux kernel, you need to obtain the latest Red Hat Linux kernel in RPM format, install the new kernel from the RPM packages, and configure the boot loader to boot the new kernel.

This chapter discusses the steps necessary to upgrade the kernel on an x86 system only.

WARNING

Building your own custom kernel is not supported by the Red Hat Linux Installation Support Team. For more information on building a custom kernel from the source code, refer to Appendix A, *Building a Custom Kernel*.

23.1 The 2.4 Kernel

Red Hat Linux now ships with the 2.4 kernel. Here are the highlights of the 2.4 kernel as shipped with Red Hat Linux:

- The directory for the kernel source is now `/usr/src/linux-2.4` instead of `/usr/src/linux`.
- Better SMP support.
- Support for up to 64 gigabytes of physical RAM — the enterprise kernel installed with Red Hat Linux 7.2 is compiled to support 64 gigabytes of physical memory.
- Better multimedia support including the maestro3 module for the ESS Allegro sound card.
- Better USB support.
- Preliminary support for IEEE 1394, also referred to as FireWire™, devices.

23.2 Preparing to Upgrade

Before you upgrade your kernel, you need to take a few precautionary steps. The first step is to make sure you have a working boot diskette for your system in case a problem occurs. If the boot loader is not configured properly to boot the new kernel, you will not be able to boot your system unless you have a boot diskette.

To create a boot diskette for your system, you need to determine which version of the kernel you are currently running. Execute the following command:

```
uname -r
```

You must be root to create a boot diskette for your system. Login as root at a shell prompt, and type the following command (where *kernelversion* is the output of the `uname -r` command):

```
/sbin/mkbootdisk kernelversion
```

Tip

Refer to the man page for `mkbootdisk` for more options.

Reboot your machine with the boot disk and verify that it works before continuing.

Hopefully, you will not have to use the diskette, but you should store it in a safe place just in case.

You should also determine which kernel packages you have installed. Some are optional and not required to use the kernel. For example, the `kernel-doc` package contains all the documentation for the kernel and is not required to boot the kernel. The `kernel-source` package is also not required because you do not have to compile the kernel from source.

To determine which kernel packages you have installed, execute the following command at a shell prompt:

```
rpm -qa | grep kernel
```

The output will contain some or all of the following packages, depending on what type of installation you performed (your version numbers may differ):

```
kernel-headers-2.4.7-3
kernel-2.4.7-3
kernel-source-2.4.7-3
kernel-doc-2.4.7-3
kernel-pcmcia-cs-3.1.24-2
```

From the output, you can determine which packages you need to download for the kernel upgrade. The only required package is the `kernel-`*`version-number`* package. If you are uprgrading the kernel on a laptop or are using PCMCIA, the `kernel-pcmcia-cs` package is also required.

You do not need the `kernel-headers` and `kernel-source` packages unless you plan to recompile the kernel yourself or plan to perform kernel development. The `kernel-doc` package contains kernel development documentation and is not required.

23.3 Downloading the Upgraded Kernel

There are several ways to determine if there is an updated kernel available for your system.

- Go to http://www.redhat.com/support/errata/, choose the version of Red Hat Linux you are using, and view the errata for it. Kernel errata are usually under the **Security Advisories** section. From the list of errata, click the kernel errata to view the detailed errata report for it. In the errata report, there is a list of required RPM packages and a link to download them from the Red Hat FTP site. You can also download them from a Red Hat FTP mirror site. A list of mirror sites is available at http://www.redhat.com/download/mirror.html.
- Use Red Hat Network. You can use Red Hat Network to download the kernel RPM packages and then manually upgrade to the latest kernel. Or, if you have elected to let the Red Hat Update Agent upgrade packages for you, Red Hat Network can download the lastest kernel, upgrade the kernel on your system, create an initial RAM disk if needed, and configure the boot loader to boot the new kernel. All you have to do is reboot into the new kernel. For more information, refer to the *Red Hat Network User Reference Guide* available at http://www.redhat.com/support/manuals/RHNetwork/ref-guide/.

If there is an updated kernel for the version of Red Hat Linux you are running, download it using one of these methods. If you used Red Hat Network to upgrade your kernel automatically, you are finished — just reboot your system to use the new kernel. If you just downloaded the RPM packages from the Red Hat Linux errata page or from Red Hat Network, proceed to Section 23.4, *Performing the Upgrade*.

23.4 Performing the Upgrade

Now that you have the necessary kernel RPM packages, you can upgrade your existing kernel. At a shell prompt as root, change to the directory that contains the kernel RPM packages and follow these steps.

You probably want to keep the older kernel in case you have problems with the new kernel. Use the `-i` argument instead of `-U` to install the `kernel` package (the version might vary):

```
rpm -ivh kernel-2.4.7-3.i386.rpm
```

If you plan to upgrade the kernel-headers, kernel-source, and kernel-docs packages, you probably do not need to keep the older versions. Use the following commands to upgrade these packages (the versions might vary):

```
rpm -Uvh kernel-header-2.4.7-3.i386.rpm
rpm -Uvh kernel-source-2.4.7-3.i386.rpm
rpm -Uvh kernel-docs-2.4.7-3.i386.rpm
```

If you are using PCMCIA (for example, a laptop), you also need to install the kernel-pcmcia-cs and keep the old version. If you use the -i switch, it will probably return a conflict because the older kernel needs this package to boot with PCMCIA support. To work around this, use the --force switch as follows (the version might vary):

```
rpm -ivh --force kernel-pcmcia-cs-3.1.24-2.i386.rpm
```

If you are using a SCSI controller, you need an initial RAM disk. The purpose of the initial RAM disk is to allow a modular kernel to have access to modules that it might need to boot from before the kernel has access to the device where the modules normally reside.

The initial RAM disk is created by using the mkinitrd command. However, the Red Hat RPM package performs this step for you. To verify that it was created, use the command ls -l /boot. You should see the file initrd-2.4.7-3.img (the version should match the version of the kernel you just installed).

Now that you have installed the new kernel, you need to configure the boot loader to boot the new kernel. Refer to Section 23.5, *Configuring the Boot Loader* for details.

23.5 Configuring the Boot Loader

Now that you have the new kernel installed, you must configure the boot loader to boot the new kernel. This is a crucial step. If you do not perform this step or if you perform it incorrectly, you will not be able to boot your system. If this happens, boot your system with the boot diskette you created earlier and try configuring the boot loader again.

In order to provide a redundant boot source to protect from a possible error in a new kernel, you should keep the original kernel available. During the installation of Red Hat Linux 7.2, you had the option to choose either LILO or GRUB as your boot loader. Refer to the appropriate section that follows.

23.5.1 GRUB

If you selected GRUB as your boot loader, you need to modify the file /boot/grub/grub.conf. The default GRUB configuration file looks similar to the following:

```
# NOTICE:  You have a /boot partition.  This means that
#          all kernel paths are relative to /boot/
default=0
```

```
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Linux (2.4.7-3)
        root (hd0,0)
        kernel /vmlinuz-2.4.7-3 ro root=/dev/hda3
        initrd /initrd-2.4.7-3.img
```

If you created a separate `/boot` partition, the paths to the kernel and initrd image are relative to the `/boot` partition.

To add your new kernel to GRUB, copy the existing section to a new one and modify it to boot your new kernel image (and initrd image if you have any SCSI devices and created an initrd image). Be sure the title of the new section is different from the title of the section to boot the old kernel. By default, Red Hat Linux uses linux and the kernel version in parentheses to differentiate between different kernels for GRUB to boot. In our example, the new `/boot/grub/grub.conf` file would look like the following:

```
# NOTICE:  You have a /boot partition.  This means that
#          all kernel paths are relative to /boot/
default=0
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz
title Red Hat Linux (2.4.7-3-jul2001)
        root (hd0,0)
        kernel /vmlinuz-2.4.7-3-jul2001 ro root=/dev/hda3
        initrd /initrd-2.4.7-3-jul2001.img

title Red Hat Linux (2.4.7-3)
        root (hd0,0)
        kernel /vmlinuz-2.4.7-3 ro root=/dev/hda3
        initrd /initrd-2.4.7-3.img
```

The default boot entry is set to number 0. To make your new kernel the default, either place its section first or change the default entry number to the appropriate number (remember that it starts counting with 0).

From now on, when the system boots you will see `Red Hat Linux (2.4.7-3-jul2001)` and `Red Hat Linux (2.4.7-3)` as GRUB boot options. To boot the default kernel, press [Enter] or wait for it to time out. If you want to boot the old kernel, select it and press [Enter].

You can begin testing your new kernel by rebooting your computer and watching the messages to ensure your hardware is detected properly.

23.5.2 LILO

To configure LILO to boot the new kernel, you need to update the /etc/lilo.conf file and run the command /sbin/lilo.

The default /etc/lilo.conf file looks similar to the following:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
linear
default=linux

image=/boot/vmlinuz-2.4.7-3
 label=linux
        initrd=initrd-2.4.7-3.img
 read-only
 root=/dev/hda5
```

To add your new kernel to LILO, copy the existing section to a new one and modify it to boot your new kernel image (and initrd image if you have any SCSI devices and created an initrd image). Also, rename the label of the old kernel to something such as **linux-old**. Your /etc/lilo.conf should look similar to the following:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
linear
default=linux

image=/boot/vmlinuz-2.4.7-3-jul2001
label=linux
initrd=initrd-2.4.7-3-jul2001.img
read-only
root=/dev/hda5

image=/boot/vmlinuz-2.4.7-3
label=linux-old
initrd=initrd-2.4.7-3.img
read-only
```

```
    root=/dev/hda5
```

To activate your changes, run the command `/sbin/lilo`. If all goes well, you will see output similar to the following:

```
    Added linux *
    Added linux-old
```

The * after linux means that the section labeled `linux` is the default kernel that LILO will boot.

From now on, when the system boots you will see `linux` and `linux-old` as LILO boot options.

To boot the new kernel (`linux`) simply press [Enter], or wait for LILO to time out. If you want to boot the old kernel (`linux-old`), select `linux-old` and press [Enter].

You can begin testing your new kernel by rebooting your computer and watching the messages to ensure your hardware is detected properly.

23.6 Additional Resources

For more information on upgrading the Linux kernel on Red Hat Linux and recompiling the kernel, refer to these resources.

23.6.1 Installed Documentation

- `/usr/src/linux-2.4/Documentation` — Advanced documentation on the Linux kernel and its modules. These documents are mostly meant for people interested in contributing to the kernel source code and understanding how the kernel works.

23.6.2 Useful Websites

- http://www.redhat.com/support/docs/howto/kernel-upgrade/kernel-upgrade.html — *Upgrading the Linux Kernel on Red Hat Linux Systems* by the Red Hat Support Team
- http://www.redhat.com/mirrors/LDP/HOWTO/Kernel-HOWTO.html — *The Linux Kernel HOWTO* from the Linux Documentation Project
- http://www.gnu.org/software/grub/grub.html — *GNU GRUB* webpage

24 Kernel Modules

The Linux kernel has a modular design. At boot time, only a minimal resident kernel is loaded into memory. Thereafter, whenever a user requests a feature that is not present in the resident kernel, a kernel module is dynamically loaded into memory. After a specified period of inactivity, the module may be removed from memory.

The mechanism that supports dynamic loading of modules is a kernel thread called `kmod`. Modules are not loaded unless they are needed. When the kernel requests a module, the module is loaded along with all its module dependencies.

Red Hat Linux also includes a cron task that removes all unused modules every ten minutes. The cron task is located in the file `/etc/cron.d/kmod`. Refer to Section 22.1, *Cron* for more information on cron tasks.

When you install Red Hat Linux, the hardware on your system is probed and you provide information about how the system will be typically used and which programs should be loaded. Based on this probing and the information you provide, the installation program decides which modules need to be loaded at boot time. The installation program sets up the dynamic loading mechanism to work transparently. If you build your own custom kernel, you can make all of these decisions for yourself.

If you add new hardware after installation and the hardware requires a kernel module, you need to set up the dynamic loading mechanism. `Kudzu` usually detects new hardware. You can also add the new driver by editing the module configuration file, `/etc/modules.conf`.

For example, if your system included a model SMC EtherPower 10 PCI network adapter at the time of installation, the module configuration file will contain the following line:

```
alias eth0 tulip
```

After installation, if you install a second identical network adapter to your system, add the following line to `/etc/modules.conf`:

```
alias eth1 tulip
```

See the *Official Red Hat Linux Reference Guide* for an alphabetical list of kernel modules and the hardware supported by the modules.

24.1 Kernel Module Utilities

You can also use a group of commands to list, load, or unload kernel modules. These commands are useful if you want to try different modules or see if a module has been loaded successfully.

The command `/sbin/lsmod` displays a list of currently loaded modules.

Example 24–1 Example `lsmod` output

```
Module                  Size  Used by
sr_mod                 15264   0 (autoclean)
mga                    95984   1
agpgart                23392   3
nfs                    79008   1 (autoclean)
lockd                  52464   1 (autoclean) [nfs]
sunrpc                 61328   1 (autoclean) [nfs lockd]
autofs                 11264   4 (autoclean)
3c59x                  25344   1 (autoclean)
ipchains               38976   0 (unused)
ide-scsi                8352   0
scsi_mod               95104   2 [sr_mod ide-scsi]
ide-cd                 26848   0
cdrom                  27232   0 [sr_mod ide-cd]
usb-uhci               20720   0 (unused)
usbcore                49664   1 [usb-uhci]
```

As you can see in Example 24–1, *Example `lsmod` output*, `lsmod` displays the size, use count, and referring modules for each module currently loaded.

To load a kernel module, you can use the command `/sbin/insmod` followed by the kernel module name. By default, `insmod` tries to load the module from the `/lib/modules/<kernel-version>/kernel/drivers` subdirectories. There is a subdirectory for each type of module, such as the `net` subdirectory for network interface drivers. Some kernel modules have module dependencies — other modules must be loaded first for it to load. To resolve these dependencies, you can either load the module dependencies and then load the module you want, or you can use the command `/sbin/modprobe` followed by the module name to load the module along with its dependencies.

For example, the command

```
/sbin/modprobe tulip
```

loads the `tulip` network interface module.

To unload kernel modules, use the command `/sbin/rmmod` followed by the module name. The `rmmod` utility will only unload modules that are not in use and that are not a dependency of other modules in use.

For example, the command

```
/sbin/rmmod tulip
```

unloads the `tulip` network interface module.

Another useful kernel module utility is `modinfo`. You can use the command `/sbin/modinfo` to display information about a kernel module. The general syntax is:

```
    /sbin/modinfo [options] <module>
```

Options include `-d` that displays a brief description of the module and `-p` that lists the parameters the module supports. For a complete list of options, refer to the `modinfo` man page (`man modinfo`).

24.2 Additional Resources

For more information on kernel modules and their utilities, refer to the following resources.

24.2.1 Installed Documentation

- lsmod man page — description and explanation of its output.
- insmod man page — description and list of command line options.
- modprobe man page — description and list of command line options.
- rmmod man page — description and list of command line options.
- modinfo man page — description and list of command line options.
- `/usr/src/linux-2.4/Documentation/kmod.txt` — description of kmod and why it replaced kerneld.
- `/usr/src/linux-2.4/Documentation/modules.txt` — how to compile and use kernel modules.

Part IV Package Management

25 Package Management with RPM

The Red Hat Package Manager (RPM) is an open packaging system, available for anyone to use, which runs on Red Hat Linux as well as other Linux and UNIX systems. Red Hat, Inc. encourages other vendors to use RPM for their own products. RPM is distributable under the terms of the GPL.

For the end user, RPM makes system updates easy. Installing, uninstalling, and upgrading RPM packages can be accomplished with short commands. RPM maintains a database of installed packages and their files, so you can invoke powerful queries and verifications on your system. If you prefer a graphical interface, you can use Gnome-RPM to perform many RPM commands.

During upgrades, RPM handles configuration files carefully, so that you never lose your customizations — something that you will not accomplish with regular `.tar.gz` files.

For the developer, RPM allows you to take software source code and package it into source and binary packages for end users. This process is quite simple and is driven from a single file and optional patches that you create. This clear delineation of "pristine" sources and your patches and build instructions eases the maintenance of the package as new versions of the software are released.

Run RPM Commands as Root

Because RPM makes changes to your system, you must be root in order to install, remove, or upgrade an RPM package.

25.1 RPM Design Goals

In order to understand how to use RPM, it can be helpful to understand RPM's design goals:

Upgradability

Using RPM, you can upgrade individual components of your system without completely reinstalling. When you get a new release of an operating system based on RPM (such as Red Hat Linux), you don't need to reinstall on your machine (as you do with operating systems based on other packaging systems). RPM allows intelligent, fully-automated, in-place upgrades of your system. Configuration files in packages are preserved across upgrades, so you won't lose your customizations. There are no special upgrade files need to upgrade a package because the same RPM file is used to install and upgrade the package on your system.

Powerful Querying

RPM is designed to provide powerful querying options. You can do searches through your entire database for packages or just for certain files. You can also easily find out what package a file belongs to and from where the package came. The files an RPM package contains are in a compressed archive, with a custom binary header containing useful information about the package and its contents, allowing you to query individual packages quickly and easily.

System Verification

Another powerful feature is the ability to verify packages. If you are worried that you deleted an important file for some package, simply verify the package. You will be notified of any anomalies. At that point, you can reinstall the package if necessary. Any configuration files that you modified are preserved during reinstallation.

Pristine Sources

A crucial design goal was to allow the use of "pristine" software sources, as distributed by the original authors of the software. With RPM, you have the pristine sources along with any patches that were used, plus complete build instructions. This is an important advantage for several reasons. For instance, if a new version of a program comes out, you do not necessarily have to start from scratch to get it to compile. You can look at the patch to see what you *might* need to do. All the compiled-in defaults, and all of the changes that were made to get the software to build properly are easily visible using this technique.

The goal of keeping sources pristine may only seem important for developers, but it results in higher quality software for end users, too. We would like to thank the folks from the BOGUS distribution for originating the pristine source concept.

25.2 Using RPM

RPM has five basic modes of operation (not counting package building): installing, uninstalling, upgrading, querying, and verifying. This section contains an overview of each mode. For complete details and options try `rpm --help`, or turn to Section 25.5, *Additional Resources* for more information on RPM.

25.2.1 Finding RPMs

Before using an RPM, you must know where to find them. An Internet search will return many RPM repositories, but if you are looking for RPM packages built by Red Hat, they can be found at the following locations:

- The official Red Hat Linux CD-ROMs
- The Red Hat Errata Page available at http://www.redhat.com/support/errata
- A Red Hat FTP Mirror Site available at http://www.redhat.com/mirrors.html
- Red Hat Network — See Chapter 27, *Red Hat Network* for more details on Red Hat Network

25.2.2 Installing

RPM packages typically have file names like `foo-1.0-1.i386.rpm`. The file name includes the package name (`foo`), version (`1.0`), release (`1`), and architecture (`i386`). Installing a package is as simple as typing the following command at a shell prompt:

```
# rpm -ivh foo-1.0-1.i386.rpm
foo                        ####################################
#
```

As you can see, RPM prints out the name of the package and then prints a succession of hash marks as the package is installed as a progress meter.

Note

Although a command like `rpm -ivh foo-1.0-1.i386.rpm` is commonly used to install an RPM package, you may want to consider using `rpm -Uvh foo-1.0-1.i386.rpm` instead. `-U` is commonly used for upgrading a package, but it will also install new packages. See Section 25.2.4, *Upgrading* for more information about using the `-U` RPM option.

Installing packages is designed to be simple, but you may sometimes see errors:

Package Already Installed

If the package of the same version is already installed, you will see:

```
# rpm -ivh foo-1.0-1.i386.rpm
foo                      package foo-1.0-1 is already installed
#
```

If you want to install the package anyway and the same version you are trying to install is already installed, you can use the `--replacepkgs` option, which tells RPM to ignore the error:

```
# rpm -ivh --replacepkgs foo-1.0-1.i386.rpm
foo                        ####################################
#
```

This option is helpful if files installed from the RPM were deleted or if you want the original configuration files from the RPM to be installed.

Conflicting Files

If you attempt to install a package that contains a file which has already been installed by another package or an earlier version of the same package, you'll see:

```
# rpm -ivh foo-1.0-1.i386.rpm
foo           /usr/bin/foo conflicts with file from bar-1.0-1
#
```

To make RPM ignore this error, use the `--replacefiles` option:

```
# rpm -ivh --replacefiles foo-1.0-1.i386.rpm
foo                       ####################################
#
```

Unresolved Dependency

RPM packages can "depend" on other packages, which means that they require other packages to be installed in order to run properly. If you try to install a package which has an unresolved dependency, you'll see:

```
# rpm -ivh foo-1.0-1.i386.rpm
failed dependencies:
        bar is needed by foo-1.0-1
#
```

To handle this error you should install the requested packages. If you want to force the installation anyway (a bad idea since the package probably will not run correctly), use the `--nodeps` option.

25.2.3 Uninstalling

Uninstalling a package is just as simple as installing one. Type the following command at a shell prompt:

```
# rpm -e foo
#
```

Note

Notice that we used the package *name* `foo`, not the name of the original package *file* `foo-1.0-1.i386.rpm`. To uninstall a package, you will need to replace `foo` with the actual package name of the original package.

You can encounter a dependency error when uninstalling a package if another installed package depends on the one you are trying to remove. For example:

```
# rpm -e foo
removing these packages would break dependencies:
        foo is needed by bar-1.0-1
#
```

To cause RPM to ignore this error and uninstall the package anyway (which is also a bad idea since the package that depends on it will probably fail to work properly), use the `--nodeps` option.

25.2.4 Upgrading

Upgrading a package is similar to installing one. Type the following command at a shell prompt:

```
# rpm -Uvh foo-2.0-1.i386.rpm
foo                       ####################################
#
```

What you do not see above is that RPM automatically uninstalled any old versions of the `foo` package. In fact, you may want to always use `-U` to install packages, since it will work even when there are no previous versions of the package installed.

Since RPM performs intelligent upgrading of packages with configuration files, you may see a message like the following:

```
saving /etc/foo.conf as /etc/foo.conf.rpmsave
```

This message means that your changes to the configuration file may not be "forward compatible" with the new configuration file in the package, so RPM saved your original file, and installed a new one. You should investigate the differences between the two configuration files and resolve them as soon as possible, to ensure that your system continues to function properly.

Upgrading is really a combination of uninstalling and installing, so during an RPM upgrade you can encounter uninstalling and installing errors, plus one more. If RPM thinks you are trying to upgrade to a package with an *older* version number, you will see:

```
# rpm -Uvh foo-1.0-1.i386.rpm
foo    package foo-2.0-1 (which is newer) is already installed
#
```

To cause RPM to "upgrade" anyway, use the `--oldpackage` option:

```
# rpm -Uvh --oldpackage foo-1.0-1.i386.rpm
foo                       ####################################
#
```

25.2.5 Freshening

Freshening a package is similar to upgrading one. Type the following command at a shell prompt:

```
# rpm -Fvh foo-1.2-1.i386.rpm
foo                       ####################################
#
```

RPM's freshen option checks the versions of the packages specified on the command line against the versions of packages that have already been installed on your system. When a newer version of an already-installed package is processed by RPM's freshen option, it will be upgraded to the newer version. However, RPM's freshen option will not install a package if no previously-installed package of the same name exists. This differs from RPM's upgrade option, as an upgrade *will* install packages, whether or not an older version of the package was already installed.

RPM's freshen option works for single packages or a group of packages. If you have just downloaded a large number of different packages, and you only want to upgrade those packages that are already installed on your system, freshening will do the job. If you use freshening, you will not have to deleting any unwanted packages from the group that you downloaded before using RPM.

In this case, you can simply issue the following command:

```
# rpm -Fvh *.rpm
```

RPM will automatically upgrade only those packages that are already installed.

25.2.6 Querying

Use the `rpm -q` command to query the database of installed packages. The `rpm -q foo` command will print the package name, version, and release number of the installed package foo:

```
# rpm -q foo
foo-2.0-1
#
```

Note

Notice that we used the package *name* `foo`. To query a package, you will need to replace `foo` with the actual package name.

Instead of specifying the package name, you can use the following options with `-q` to specify the package(s) you want to query. These are called *Package Specification Options*.

- `-a` queries all currently installed packages.
- `-f` *`<file>`* will query the package which owns *`<file>`*. When specifying a file, you must specify the full path of the file (for example, `/usr/bin/ls`).
- `-p` *`<packagefile>`* queries the package *`<packagefile>`*.

There are a number of ways to specify what information to display about queried packages. The following options are used to select the type of information for which you are searching. These are called *Information Selection Options*.

- `-i` displays package information including name, description, release, size, build date, install date, vendor, and other miscellaneous information.
- `-l` displays the list of files that the package contains.
- `-s` displays the state of all the files in the package.
- `-d` displays a list of files marked as documentation (man pages, info pages, READMEs, etc.).
- `-c` displays a list of files marked as configuration files. These are the files you change after installation to adapt the package to your system (for example, `sendmail.cf`, `passwd`, `inittab`, etc.).

For the options that display lists of files, you can add `-v` to the command to display the lists in a familiar `ls -l` format.

25.2.7 Verifying

Verifying a package compares information about files installed from a package with the same information from the original package. Among other things, verifying compares the size, MD5 sum, permissions, type, owner, and group of each file.

The command `rpm -V` verifies a package. You can use any of the *Package Selection Options* listed for querying to specify the packages you wish to verify. A simple use of verifying is `rpm -V foo`, which verifies that all the files in the foo package are as they were when they were originally installed. For example:

- To verify a package containing a particular file:

  ```
  rpm -Vf /bin/vi
  ```

- To verify ALL installed packages:

  ```
  rpm -Va
  ```

- To verify an installed package against an RPM package file:

  ```
  rpm -Vp foo-1.0-1.i386.rpm
  ```

  This command can be useful if you suspect that your RPM databases are corrupt.

If everything verified properly, there will be no output. If there are any discrepancies they will be displayed. The format of the output is a string of eight characters (a `c` denotes a configuration file) and then the file name. Each of the eight characters denotes the result of a comparison of one attribute of the file to the value of that attribute recorded in the RPM database. A single . (a period) means the test passed. The following characters denote failure of certain tests:

- 5 — MD5 checksum
- S — file size
- L — symbolic link
- T — file modification time
- D — device
- U — user
- G — group
- M — mode (includes permissions and file type)
- ? — unreadable file

If you see any output, use your best judgment to determine if you should remove or reinstall the package, or fix the problem in another way.

25.3 Checking a Package's Signature

If you wish to verify that a package has not been corrupted or tampered with, examine only the md5sum by typing the following command at a shell prompt (replace coolapp with the filename of your RPM package):

```
rpm --checksig --nogpg coolapp-1.1-1.rpm
```

You'll see the message `coolapp-1.1-1.rpm:  md5 OK`. This brief message means that the file was not corrupted by the download.

On the other hand, how trustworthy is the developer who created the package? If the package is **signed** with the developer's GnuPG **key**, you'll know that the developer really is who they say they are.

An RPM package can be signed using Gnu Privacy Guard (or GnuPG), to help you make certain your downloaded package is trustworthy.

GnuPG is a tool for secure communication; it is a complete and free replacement for the encryption technology of PGP, an electronic privacy program. With GnuPG, you can authenticate the validity of documents, and encrypt/decrypt data to and from other recipients. GnuPG is capable of decrypting and verifying PGP 5.*x* files, as well.

During the installation of Red Hat Linux, GnuPG is installed by default. That way you can immediately start using GnuPG to verify any packages that you receive from Red Hat. First, you will need to import Red Hat's public key.

25.3.1 Importing Keys

When you import a public key, you add that key to your **keyring** (a file in which public and secret keys are kept). Then, when you download a document or file from that entity, you can check the validity of that document against the key you added to your keyring.

To import a key, use the `--import` option. To demonstrate, download and import Red Hat's public key. That way, any time you want to validate a package from Red Hat, you will be able to check it against the key you retrieved.

You can find Red Hat's key at http://www.redhat.com/about/contact.html. Using your browser, download the key by pressing the [Shift] key while you click on the download link, then click the **OK** button to save the file (for example `redhat2.asc`). Then, at the shell prompt, import the key with the following command:

```
gpg --import redhat2.asc
```

The resulting message tells you that the key was processed. To check that the key was added, type `gpg --list-keys`. You'll see the key you just downloaded from Red Hat, as well as your own keys.

```
[newuser@localhost newuser]$ gpg --list-keys
/home/newuser/.gnupg/pubring.gpg
----------------------------------------
pub  1024D/DB42A60E 1999-09-23 Red Hat, Inc <security@redhat.com>
sub  2048g/961630A2 1999-09-23
```

Keys Do Not Have to be Links

Sometimes, you will not be able to download a key from a link. Keys are text files, so they can be moved to your machine in any way a regular text file can be saved. As long as you know the name and location of the file you saved, you can import it to your keyring.

25.3.2 Verifying Packages

To check the GnuPG signature of an RPM file after importing the builder's GnuPG key, use the following command (replace coolapp with the filename of your RPM package):

```
rpm --checksig coolapp-1.1-1.rpm
```

If all goes well, you will see the message: `md5 gpg OK`. That means that the package is not corrupt.

25.3.3 More about GnuPG

For more information about GnuPG, see Appendix B, *Getting Started with Gnu Privacy Guard*.

25.4 Impressing Your Friends with RPM

RPM is a useful tool for both managing your system and diagnosing and fixing problems. The best way to make sense of all of its options is to look at some examples.

- Perhaps you have deleted some files by accident, but you are not sure what you deleted. If you want to verify your entire system and see what might be missing, you could try the following command:

  ```
  rpm -Va
  ```

  If some files are missing or appear to have been corrupted, you should probably either re-install the package or uninstall, then re-install the package.

- At some point, you might see a file that you do not recognize. To find out which package owns it, you would enter:

  ```
  rpm -qf /usr/X11R6/bin/ghostview
  ```

  The output would look like the following:

  ```
  gv-3.5.8-10
  ```

- We can combine the above two examples in the following scenario. Say you are having problems with `/usr/bin/paste`. You would like to verify the package that owns that program, but you do not know which package owns `paste`. Simply enter the following command:

  ```
  rpm -Vf /usr/bin/paste
  ```

  and the appropriate package will be verified.

- Do you want to find out more information about a particular program? You can try the following command to locate the documentation which came with the package that owns that program:

  ```
  rpm -qdf /usr/bin/md5sum
  ```

  The output would be like the following:

  ```
  /usr/share/doc/textutils-2.0a/NEWS
  /usr/share/doc/textutils-2.0a/README
  /usr/info/textutils.info.gz
  /usr/man/man1/cat.1.gz
  /usr/man/man1/cksum.1.gz
  /usr/man/man1/comm.1.gz
  /usr/man/man1/csplit.1.gz
  ```

```
/usr/man/man1/cut.1.gz
/usr/man/man1/expand.1.gz
/usr/man/man1/fmt.1.gz
/usr/man/man1/fold.1.gz
/usr/man/man1/head.1.gz
/usr/man/man1/join.1.gz
/usr/man/man1/md5sum.1.gz
/usr/man/man1/nl.1.gz
/usr/man/man1/od.1.gz
/usr/man/man1/paste.1.gz
/usr/man/man1/pr.1.gz
/usr/man/man1/ptx.1.gz
/usr/man/man1/sort.1.gz
/usr/man/man1/split.1.gz
/usr/man/man1/sum.1.gz
/usr/man/man1/tac.1.gz
/usr/man/man1/tail.1.gz
/usr/man/man1/tr.1.gz
/usr/man/man1/tsort.1.gz
/usr/man/man1/unexpand.1.gz
/usr/man/man1/uniq.1.gz
/usr/man/man1/wc.1.gz
```

- You may find a new RPM, but you don’t know what it does. To find information about it, use the following command:

  ```
  rpm -qip sndconfig-0.48-1.i386.rpm
  ```

  The output would look like the following:

  ```
  Name        : sndconfig                Relocations: (not relocateable)
  Version     : 0.48                          Vendor: Red Hat
  Release     : 1                         Build Date: Mon 10 Jul 2000 02:25:40
  Install date: (none)                    Build Host: porky.devel.redhat.com
  Group       : Applications/Multimedia   Source RPM: sndconfig-0.48-1.src.rpm
  Size        : 461734                       License: GPL
  Packager    : Red Hat <http://bugzilla.redhat.com/bugzilla>
  Summary     : The Red Hat Linux sound configuration tool.
  Description :
  Sndconfig is a text based tool which sets up the configuration files
  you’ll need to use a sound card with a Red Hat Linux system.
  Sndconfig can be used to set the proper sound type for programs which
  use the /dev/dsp, /dev/audio and /dev/mixer devices.  The sound
  settings are saved by the aumix and sysV runlevel scripts.
  ```

- Perhaps you now want to see what files the `sndconfig` RPM installs. You would enter the following:

  ```
  rpm -qlp sndconfig-0.48-1.i386.rpm
  ```

  The output will look like the following:

  ```
  /usr/sbin/pnpprobe
  /usr/sbin/sndconfig
  /usr/share/locale/cs/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/da/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/de/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/es/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/fr/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/hu/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/id/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/is/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/it/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/ko/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/no/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/pt/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/pt_BR/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/ro/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/ru/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/sk/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/sl/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/sr/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/sv/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/tr/LC_MESSAGES/sndconfig.mo
  /usr/share/locale/uk/LC_MESSAGES/sndconfig.mo
  /usr/share/man/man8/pnpprobe.8.gz
  /usr/share/man/man8/sndconfig.8.gz
  /usr/share/sndconfig/sample.au
  /usr/share/sndconfig/sample.midi
  ```

These are just a few examples. As you use it, you will find many more uses for RPM.

25.5 Additional Resources

RPM is an extremely complex utility with many options and methods for querying, installing, upgrading, and removing packages. Refer to the following resources to learn more about RPM.

25.5.1 Installed Documentation

- `rpm --help` — This command displays a quick reference of RPM parameters.
- `man rpm` — The RPM man page will give you more detail about RPM parameters than the `rpm --help` command.

25.5.2 Useful Websites

- http://www.rpm.org/
- http://www.redhat.com/support/mailing-lists/ — The RPM mailing list is archived here. To subscribe, send mail to `rpm-list-request@redhat.com` with the word `subscribe` in the subject line.

25.5.3 Related Books

- *Maximum RPM* by Ed Bailey; Red Hat Press — An online version of the book is available at http://www.rpm.org/ and http://www.redhat.com/support/books/.

26 Gnome-RPM

If you do not want to use the command-line version of RPM, you can use Gnome-RPM, a graphical interface for Red Hat Package Manager (RPM). To learn more about RPM technology, turn to Chapter 25, *Package Management with RPM*.

Gnome-RPM (which is also referred to as gnorpm) allows users to easily work with RPM technology and features a friendly interface. It is "GNOME-compliant," meaning that it seamlessly integrates into GNOME, a graphical X Window System desktop environment provided with Red Hat Linux.

Using Gnome-RPM, you can easily accomplish the following tasks:

- install RPM packages
- uninstall RPM packages
- upgrade RPM packages
- find new RPM packages
- query RPM packages
- verify RPM packages

The Gnome-RPM interface provides a menu, a toolbar, a tree, and a window which displays currently installed packages.

To perform a Gnome-RPM task, you usually find and select packages, then choose the type of operation to perform using either a button on the toolbar, from the menu or by right-clicking with the mouse.

- Installing a package places all of the components of that package on your system in the correct locations.
- Uninstalling a package removes all components of the package except for configuration files you have modified.
- Upgrading a package installs the new version and uninstalls all other versions that were previously installed.

You can also use the **Web find** option to search the Internet for newly released packages. You can direct Gnome-RPM to search for particular distributions when you want to look for new packages. (If you have a slow connection, this option can take some time to fully execute.) See Section 26.4, *Configuration* for more information about this feature.

WARNING

Be careful when using Web find, since there is no way to verify the integrity of the many packages which are available at numerous repositories. Before installing packages, you should perform a query on that package to help you determine whether it can be trusted. Packages not produced by Red Hat are not supported in any way by Red Hat. Refer to Section 26.5.2, *Verifying Packages* to learn more about verifying packages.

Using Gnome-RPM to perform all of these and many other operations is the same as using RPM commands from the shell prompt. However, the graphical nature of Gnome-RPM may make these operations easier to perform. Gnome-RPM can display packages in a variety of different ways. Refer to Section 26.3, *Installing New Packages* for more information on using filters to identify packages.

You can install, upgrade, or uninstall several packages with a few button clicks. Similarly, you can query and verify more than one package at a time. Since Gnome-RPM is integrated with GNOME, you can also perform installation, query and verification on packages from within the GNOME File Manager.

If you want to maintain official Red Hat Linux packages, it is recommended that you use Red Hat Network or the Red Hat Linux errata page available at http://www.redhat.com/support/errata/. Packages from Red Hat have been verified for integrity and are GPG signed by Red Hat so that you can make sure they are the official packages.

26.1 Starting Gnome-RPM

To start Gnome-RPM, use one of the following methods:

- On the GNOME desktop, go to **Main Menu Button** (on the panel) => **Programs** => **System** => **GnoRPM**
- On the KDE desktop, go to **Main Menu Button** (on the panel) => **Programs** => **System** => **GnoRPM**
- At a shell prompt, type `gnorpm &`

You will see the main Gnome-RPM window (as shown in Figure 26–1, *Main Gnome-RPM Window*).

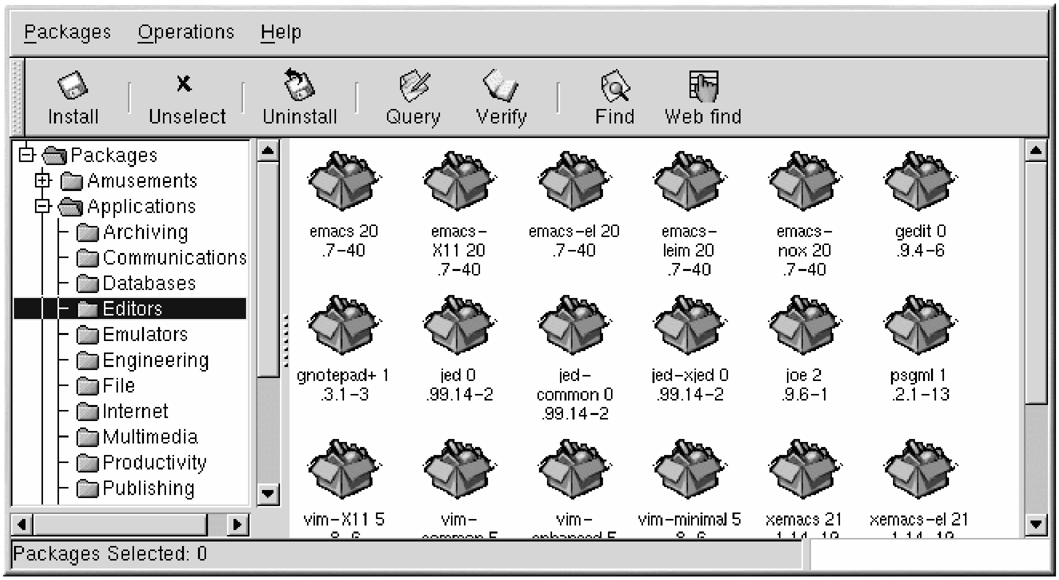
Note

If you would like to install, upgrade or uninstall packages, you must be root. The easiest way to become root is to type the `su` command and [Enter] at a shell prompt. Then type the root password. However, you do not have to be root to query and verify packages.

The Gnome-RPM interface consists of the following:

- Package Display — on the left; allows you to browse and select packages on your system
- Display window — to the right of the package panel; shows you contents from folders in the panel
- Toolbar — above the display and panel; a graphical display of package tools
- Menu — above the toolbar; contains text-based commands, as well as help info, preferences and other settings
- Status bar — beneath the panel and display windows; shows the total number of selected packages

Figure 26–1 Main Gnome-RPM Window



Packages Selected: 0

26.2 The Package Display

Each folder icon in the tree view at left represents a group of packages. Each group can contain subgroups. For example, the folder **Applications** contains the folder **Editors** that contains text editors such as Emacs, ed, vim, and GXedit.

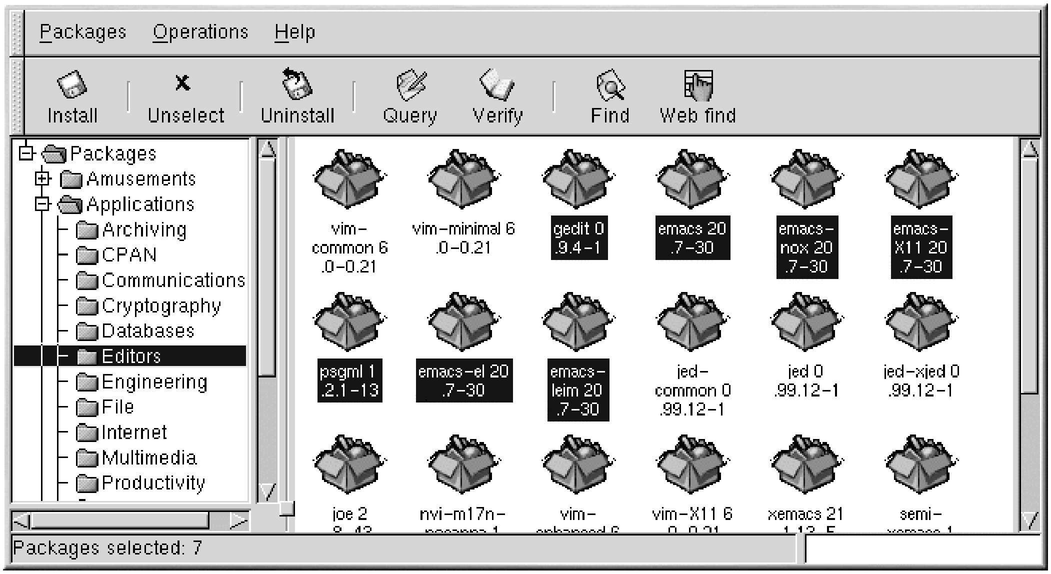
The tree view can be expanded and collapsed, so you can easily navigate through the packages. A folder which appears with a **+** next to it indicates that there are subfolders within that category.

To expand a group into subgroups, click once on the **+** with your left mouse button. To view the packages within the subgroup, left-click once on a folder name. The display window will then show you the contents of that folder. By default, you will be presented with icons that represent the packages. You can change that view to a list view by selecting **Operations** => **Preferences** from the pull-down menu, clicking on the **Package Listing** tab, and selecting **View as list**. Refer to Section 26.4, *Configuration* for more information about customizing Gnome-RPM settings.

26.2.1 Selecting Packages

To select a single package, click on it with the left mouse button. When a package is selected, its title will be highlighted as shown in Figure 26–2, *Selecting Packages in Gnome-RPM*). To unselect a package, either click on an empty space in the display panel with the left mouse button, or click on the **Unselect** button on the toolbar. When you unselect a package, the highlighting will disappear.

Figure 26–2 Selecting Packages in Gnome-RPM



Packages selected: 7

You can select and unselect multiple packages, in more than one folder in the tree panel. To select more than one package, hold down the [Ctrl] key and left-click on packages; each selected package will be highlighted.

To select a group of packages within a folder, left-click on one package. Hold down the [Shift] key and left-click on the final package you wish to select. You will see that all of the packages between your starting and ending selections will be highlighted for selection.

The status bar at the bottom of Gnome-RPM will display the total number of packages you have selected.

26.3 Installing New Packages

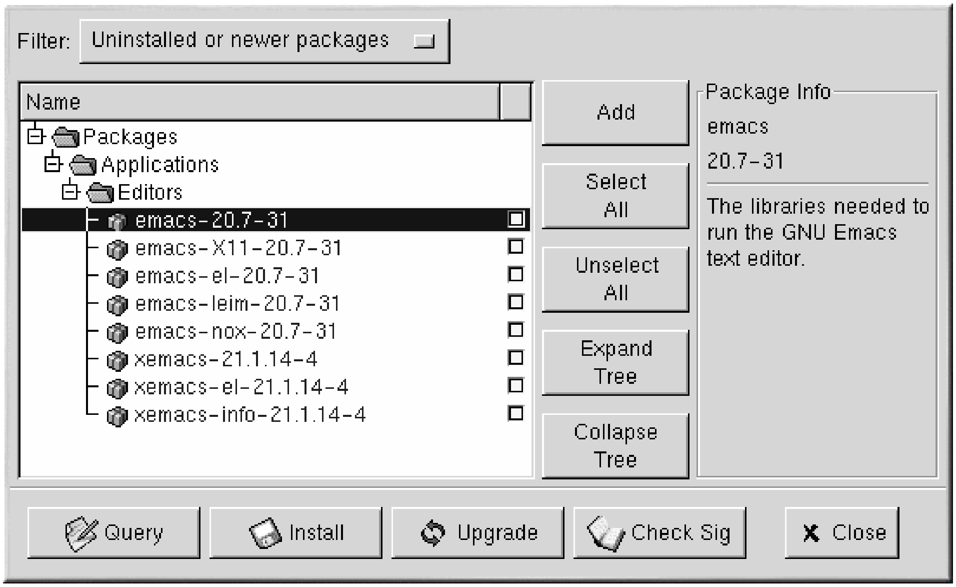
To install new packages, choose **Install** from the toolbar. In the **Install** window, your view will depend upon what you have selected under **Filter**.

Filter can be used to narrow your choices for viewing packages. Available filters include the following:

- All packages
- All packages except for those that are already installed

- Only uninstalled packages
- Only newer packages
- Uninstalled or newer packages

Figure 26–3 The Install Window



Filter: Uninstalled or newer packages

Name

- Packages
  - Applications
    - Editors
      - emacs-20.7-31 □
      - emacs-X11-20.7-31 □
      - emacs-el-20.7-31 □
      - emacs-leim-20.7-31 □
      - emacs-nox-20.7-31 □
      - xemacs-21.1.14-4 □
      - xemacs-el-21.1.14-4 □
      - xemacs-info-21.1.14-4 □

Package Info

emacs

20.7-31

The libraries needed to run the GNU Emacs text editor.

Click on the **Add** button. By default, if your CD-ROM is mounted with a Red Hat Linux CD-ROM, Gnome-RPM will search in `/mnt/cdrom/RedHat/RPMS` for new packages. (You can change the default path in the **Install Window** tab of the **Operations** => **Preferences** dialog. See Section 26.4, *Configuration* for more information.)

If no packages are available in the default path, you will see an **Add Packages** window. You can select the location of your new package using the standard file dialog window.

If you click on a package, you'll see a brief description of the package in the **Package Info** panel of the **Install** window. To perform an installation or a query on the package, select the check box next to the package, then click on the **Install** button. You can also query a selected package. On the **Package Info** window, you can also perform the installation

To choose an item, double-click on it with your left mouse button, or click on the **Add** button. The selected package(s) will be added to the **Install** window.

In addition to installing the packages from within the **Install** window, you can install a package after performing a query on the selected package. Click on **Query**, which will open the **Package Info** window. Here, you can find a variety of details about the package you've selected to install, including the origination of the package, the date it was built, its size and more.

If the package already exists on your system and you're querying a newer version, the **Package Info** window provides an **Upgrade** button, which will upgrade the package to the newer version.

You can also drag and drop packages from GNOME File Manager. Within the File Manager, left-click on the selected package. While still holding down the mouse button, drag the file to the **Install** window and place it within the **Name** panel.

When dragging files to the **Install** window from the File Manager, you'll notice that the file is displayed as an icon while it's being dragged toward Gnome-RPM. Once inside the **Name** panel, you'll see that the package is checked for installation by default, and its information appears in the **Package Info** panel to the right.

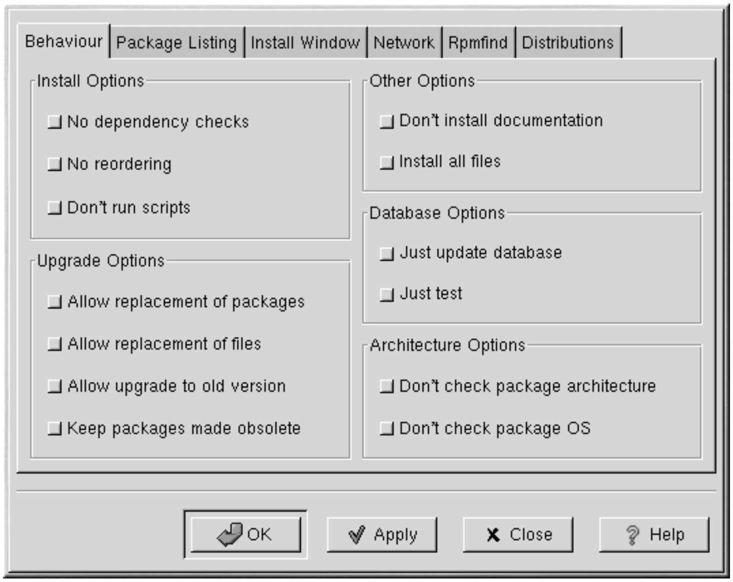
To install the package, select the **Install** button. You'll see a progress indicator as your package is being installed.

26.4 Configuration

Gnome-RPM offers a wide selection of choices for installing and uninstalling packages, documentation and other features. You can customize Gnome-RPM using the **Preferences** dialog, which you can access from **Operations** => **Preferences** on the menu. To make selections in the **Preferences** dialog, select the boxes next to the options.

Under the **Behaviour** tab, you'll find a number of options for configuring the way Gnome-RPM installs, uninstalls and upgrades packages. The **Behaviour** tab is divided into five sections: **Install Options**, **Upgrade Options**, **Other Options**, **Database Options** and **Architecture Options**. Note that by default these boxes are not selected (see Figure 26–4, *Behaviour Tab in Preferences*).

Figure 26–4 Behaviour Tab in Preferences



Install Options

☐ No dependency checks

☐ No reordering

☐ Don't run scripts

Upgrade Options

☐ Allow replacement of packages

☐ Allow replacement of files

☐ Allow upgrade to old version

☐ Keep packages made obsolete

Other Options

☐ Don't install documentation

☐ Install all files

Database Options

☐ Just update database

☐ Just test

Architecture Options

☐ Don't check package architecture

☐ Don't check package OS

Under **Install Options**, you have the following choices:

- **No dependency checks** — When selected, this will install or upgrade a package without checking for other files that the program may depend on in order to work. Unless you know what you're doing, we strongly suggest that you not use this option as some packages may depend on other packages in order to function correctly.
- **No reordering** — This option is useful if RPM is unable to change the installation order of some packages to satisfy dependencies.
- **Don't run scripts** — Pre- and post-install scripts are sequences of commands that are sometimes included in packages to assist with installation. Selecting this option is similar to the `--noscripts` option when installing packages from the shell prompt.

Under **Upgrade Options**, you can select the following:

- **Allow replacement of packages** — Replaces a package with a new copy of the same package. Similar to the `--replacepkgs` option from the shell prompt. This option can be useful if an installed package has become damaged or requires repair to function correctly.

- **Allow replacement of files** — Allows the replacement of files which are owned by another package. The shell prompt equivalent for this RPM option is `--replacefiles`. This option can be useful when two packages include files that are named the same but contain different contents.
- **Allow upgrade to old version** — Like the shell prompt RPM command equivalent `--oldpackage`, this option allows you to "upgrade" to an earlier package. It can be useful if the latest version of a package doesn't function correctly on your system.
- **Keep packages made obsolete** — Prevents packages listed in an Obsoletes header from being removed.

In **Other Options**, you can select:

- **Don't install documentation** — Like `--excludedocs`, this option can save on disk space by excluding documentation such as man pages or other information related to the package.
- **Install all files** — Installs all files in the package.

The choices available in **Database Options** and **Architecture Options** allow you to decide, among other things, whether you want to perform a "test" installation (which will check for file conflicts without actually performing an install), or whether you want to exclude packages for other operating systems or system architectures.

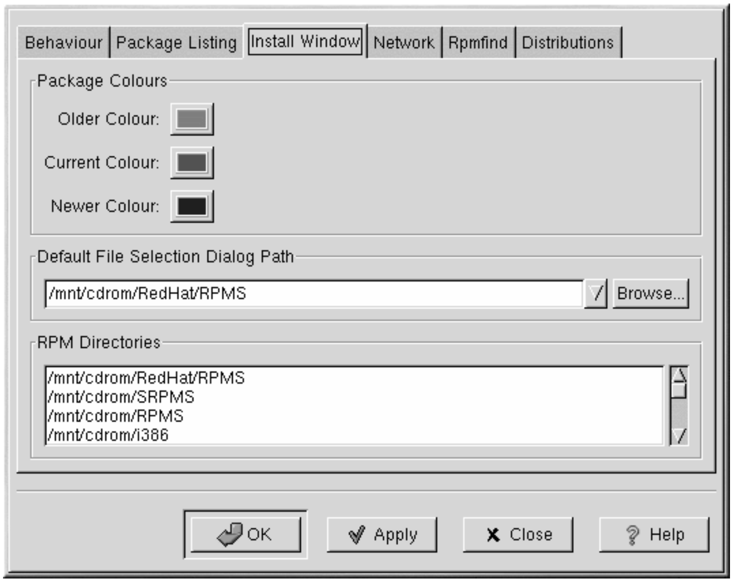
In the **Package Listing** tab, you'll find a choice of displays for your packages: either **View as icons**, which will be graphically-based, or **View as list**, which is not graphical but can provide more information about the packages.

In **Install Window,**, you can specify the path where Gnome-RPM can find new RPMs on your system. Refer to Figure 26–5, *Install Window* for an example of this dialog. If you're using your Red Hat Linux CD-ROM, this path will probably be

```
/mnt/cdrom/RedHat/RPMS
```

If you download new RPMs from the Internet or want to install RPMs via a NFS-mounted CD-ROM this path will be different for you.

Figure 26–5 Install Window



Package Colours

Older Colour: ▨

Current Colour: ▨

Newer Colour: ■

Default File Selection Dialog Path

/mnt/cdrom/RedHat/RPMS

RPM Directories

/mnt/cdrom/RedHat/RPMS
/mnt/cdrom/SRPMS
/mnt/cdrom/RPMS
/mnt/cdrom/i386

To change this path, type the full path to the RPMs you'd like to work with. Choosing the **Apply** or **OK** buttons will save this path, making it the default path for future sessions. You can also determine the default path by selecting the **Browse...** button, and visually navigating through the **RPMPath** window.

After changing the install path and closing the dialog box, you can use the **Install** button to view the packages available in the new location.

(If the path for your RPMs doesn't match the default path in your preferences, you'll be presented with a window for browsing through your filesystem, which will allow you to select the correct path for your new RPMs.)

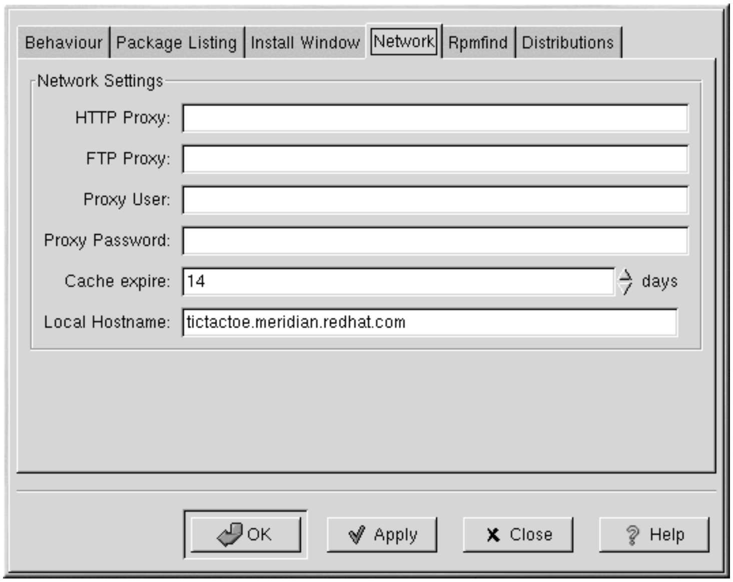
Under **Package Colours,** you'll find color coding for packages. The default setting for older packages is gray; for current packages, the color is green; for newer packages than those installed, the color is blue. These color values can be customized to suit your needs.

The **RPM Directories** field contains a list of default locations where Gnome-RPM will search for packages when the **Install** window is first opened. For example, `/mnt/cdrom/RedHat/RPMS` is listed by default. If you have the Red Hat Linux CD mounted in this location, Gnome-RPM will search it for RPM packages when you open the **Install** window.

In the **Network** tab, you have the ability to specify proxies for use with HTTP and FTP transfers, as well as user and password names (see Figure 26–6, *Network Settings*). Note, however, that the password will not be stored securely.

In the **Cache expire** field, you can set the length of time before data from the rpmfind database is considered to be out of date.

Figure 26–6 Network Settings



Network Settings

HTTP Proxy:

FTP Proxy:

Proxy User:

Proxy Password:

Cache expire: 14 days

Local Hostname: tictactoe.meridian.redhat.com

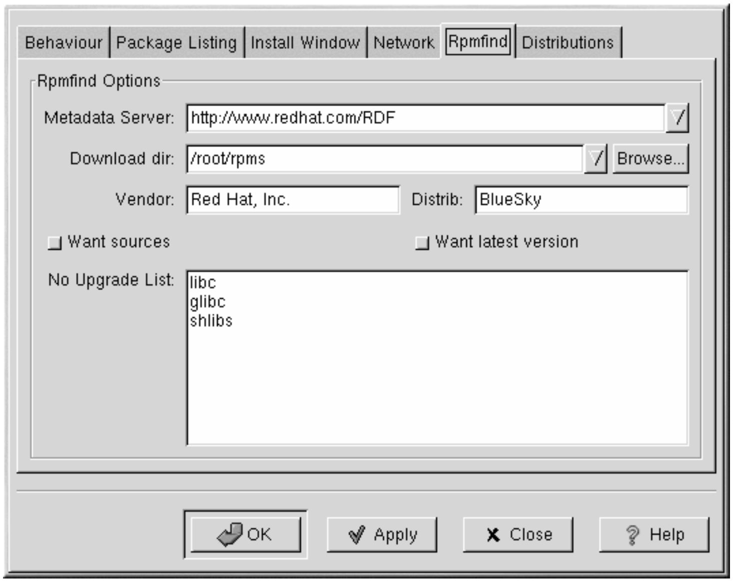
In **Rpmfind** and **Distributions**, you'll find settings and options which correspond to the **Web find** feature.

The Rpmfind system allows the user to search the Internet for packages by name, summary, architecture and more (see Figure 26–7, *The Rpmfind Window*). The user is then given the option of downloading and installing the most appropriate packages for their system. To learn more about Rpmfind, go to http://rpmfind.net/.

CAUTION

Packages not produced by Red Hat are not supported by Red Hat because Red Hat can not verify the integrity of these packages and how they interact with official Red Hat packages. Use caution when installing packages downloaded using **Rpmfind**.

Figure 26–7 The Rpmfind Window



Rpmfind Options

Metadata Server: http://www.redhat.com/RDF

Download dir: /root/rpms

Vendor: Red Hat, Inc.

Distrib: BlueSky

□ Want sources

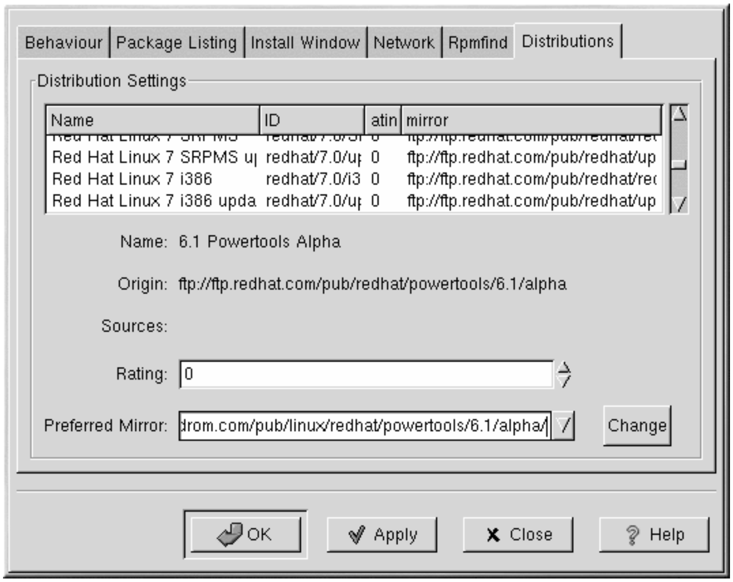
□ Want latest version

No Upgrade List:
libc
glibc
shlibs

The **Metadata server** sets the server to be used for searches. The **Download dir:** entry allows you to specify where you want the files to be placed.

You can also specify the vendor, distribution name and whether to find sources and/or the latest files.

Figure 26–8 Distribution Settings in Preferences



Name	ID	atin	mirror
Red Hat Linux 7 SRPMS	redhat/7.0/SI	0	ftp://ftp.redhat.com/pub/redhat/red
Red Hat Linux 7 SRPMS u	redhat/7.0/u	0	ftp://ftp.redhat.com/pub/redhat/up
Red Hat Linux 7 i386	redhat/7.0/i3	0	ftp://ftp.redhat.com/pub/redhat/re
Red Hat Linux 7 i386 upda	redhat/7.0/u	0	ftp://ftp.redhat.com/pub/redhat/up

Name: 6.1 Powertools Alpha

Origin: ftp://ftp.redhat.com/pub/redhat/powertools/6.1/alpha

Sources:

Rating: 0

Preferred Mirror: drom.com/pub/linux/redhat/powertools/6.1/alpha/

In **Distribution Settings**, you can set the options for choosing the most appropriate package out of the selections Rpmfind returns, as well as which mirror you would like to use. The higher the rating you indicate for your selection (as shown in Figure 26–8, *Distribution Settings in Preferences*), the higher the priority it will receive; a lower rating (such as "-1") will specify that packages not be recommended.

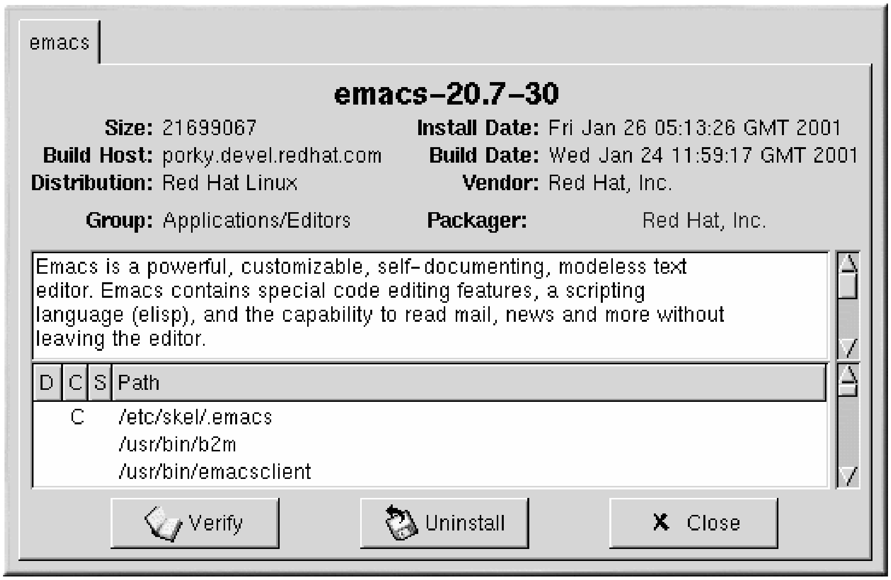
26.5 Package Manipulation

26.5.1 Querying Packages

The easiest way to query packages is to use the **Query** option from the menu at the top. If you want to query more than one package, make all your selections and then press the **Query** button on the menu.

You'll see a window like the one shown in Figure 26–9, *Query Window*. The more packages you've queried, the more tabs you'll find within the **Query** box, each tab representing a **Query** window for a package.

Figure 26–9 Query Window



emacs

emacs-20.7-30

Size: 21699067
Build Host: porky.devel.redhat.com
Distribution: Red Hat Linux
Group: Applications/Editors

Install Date: Fri Jan 26 05:13:26 GMT 2001
Build Date: Wed Jan 24 11:59:17 GMT 2001
Vendor: Red Hat, Inc.
Packager: Red Hat, Inc.

Emacs is a powerful, customizable, self-documenting, modeless text editor. Emacs contains special code editing features, a scripting language (elisp), and the capability to read mail, news and more without leaving the editor.

D	C	S	Path
	C		/etc/skel/.emacs
			/usr/bin/b2m
			/usr/bin/emacsclient

The name of the package is centered at the top of the box. Below, the box is divided into two columns of listed information; below this information, you’ll see a display area showing package files.

In the left column in the information list, you’ll find the size of the file, the machine on which the file is found, the name of the package distribution and its group.

In the right column, you’ll find the date that the package was installed on your machine, the date the package was built, the name of the vendor and the name of the group who packaged the software. If the package has not been installed on your machine, that space will simply read, "not installed."

Below the description is a list of the files contained in the package. If a D appears in its related column to the left of the path, that file is a documentation file and would be a good thing to read for help on using the application. If a C appears in its respective column, the file is a configuration file. Under the S column, you can view the state of the package; here, you’ll see if any files are missing from the package (this probably means that there is a problem with the package).

If you’re querying a package that’s already installed, you’ll also find two additional buttons at the bottom of this window: **Verify** and **Uninstall**. If you’re performing a query on a package that hasn’t been installed yet, the buttons on the bottom will be labeled **Install**, **Upgrade** and **Check Sig**.

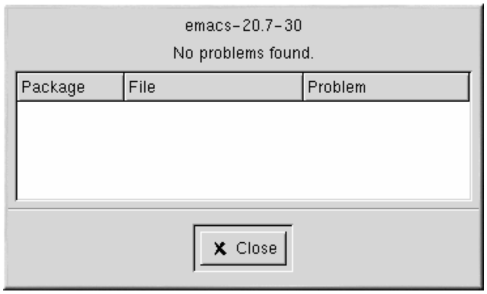
To close the query window without performing any action, left-click on the **X** at the top right of the window bar.

26.5.2 Verifying Packages

Verifying a package checks all of the files in the package to ensure they match the ones present on your system. The checksum, file size, permissions, and owner attributes are all checked against the database. This check can be used when you suspect that one of the program's files has become corrupted for some reason.

Choosing the packages to verify is like choosing the packages to query. Select the packages in the display window and use the **Verify** button on the toolbar or from **Packages** => **Verify** on the menu. A window opens like the one in Figure 26–10, *Verify Window*.

Figure 26–10 Verify Window



emacs-20.7-30

No problems found.

Package	File	Problem

As the package is being checked, you'll see the progress in the **Verify** window. If there are any problems discovered during the verify process, they'll be described in the main display area.

26.5.3 Uninstalling Packages

Uninstalling a package removes the application and associated files from your machine. When a package is uninstalled, any files it uses that are not needed by other packages on your system are also removed. Configuration files that have been modified are copied to `<filename>.rpmsave` so you can reuse them later.

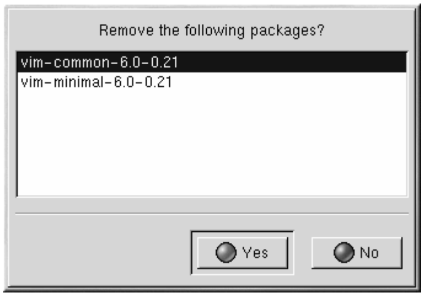
Note

You must be root to uninstall packages.

If uninstalling a package would break "dependencies" (which could interfere with the operation of applications that require one or more of the removed files in the package), a dialog will pop up, asking you to confirm the deletion.

You can uninstall a selected package in a variety of ways: from the menu, under **Packages**; from the toolbar and from the **Query** function. If you decide to remove more than one package at a time, you can choose more than one package in the same way as you would when installing, querying or verifying. The total number of selections will be displayed in the status bar on the bottom of the main window.

Figure 26–11 Uninstall Window



Remove the following packages?

vim-common-6.0-0.21
vim-minimal-6.0-0.21

Once you've begun to uninstall packages, Gnome-RPM asks for confirmation, showing a window like the one in Figure 26–11, *Uninstall Window*. All of the packages that are about to be uninstalled are listed. You should carefully check the list to make sure that you're not about to remove something you want to keep. Clicking the **Yes** button will start the uninstallation process. After it is completed, the packages and groups that have been removed will disappear from any open windows.

Upgrading Packages

When a new version of a package is released, it is easy to install it on your system. Select the package from the window of available packages in the same way you select packages for installation. You can begin the upgrade process in two ways: either the **Upgrade** button on the toolbar or using **Operations** => **Upgrade** on the menu. You simply **Add** packages in the same manner as you would during a new package installation.

During the upgrade, you'll see a progress indicator like the one for shown when you are installing packages. When it's finished, any old versions of the packages will be removed, unless you specify otherwise (refer to Section 26.4, *Configuration* for more information).

In most cases, you should upgrade packages rather than uninstall the old versions of a package and then install the new ones. If you use upgrade, any changes you made to package configuration files are preserved properly. If you uninstall an old version of a package and then install a new package, your changes could be lost.

If you run out of disk space during an installation, the install will fail. However, the package which was being installed when the error occurred may leave some files around. To clean up after this error, reinstall the package after you’ve made more disk space available.

27 Red Hat Network

Red Hat Network is an Internet solution for managing a Red Hat Linux system or a network of Red Hat Linux systems. All Security Alerts, Bug Fix Alerts, and Enhancement Alerts (collective known as Errata Alerts) can be downloaded directly from Red Hat using the Red Hat Update Agent standalone application or through the Software Manager Web interface available at http://rhn.redhat.com/.

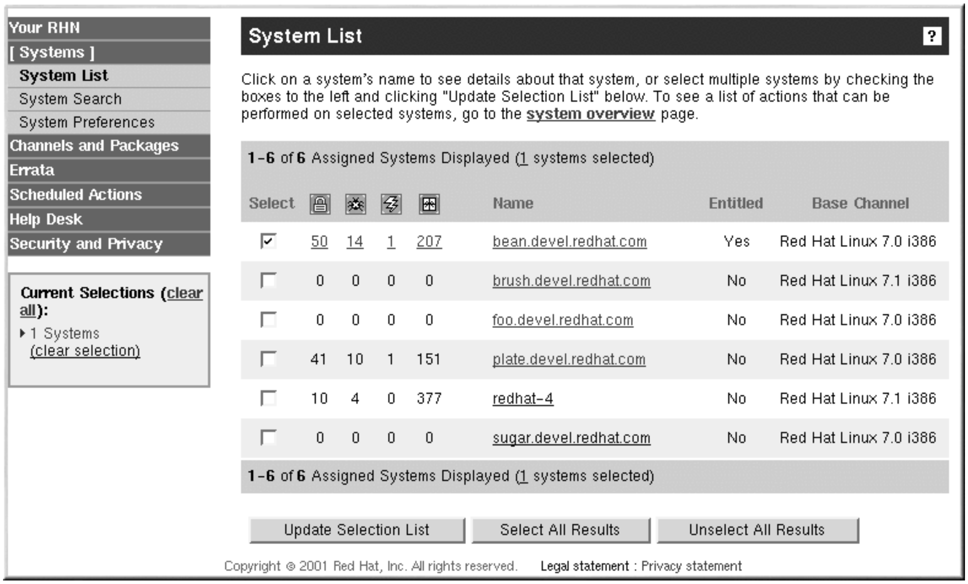
To start using Red Hat Network, follow these three basic steps:

1. Create a System Profile by running the Red Hat Network Registration Client (`rhn_register`) on the system that you want to register.
2. Log in to RHN at http://rhn.redhat.com/ and entitle the system to Software Manager. Everyone receives a free Red Hat Network Software Manager subscription for one system. Additional subscriptions are $19.95/month for each system.
3. Start scheduling updates through the Software Manager Web interface or download Errata Updates through the Red Hat Update Agent.

For more detailed instructions, read the *Red Hat Network User Reference Guide* available at http://www.redhat.com/support/manuals/RHNetwork/ref-guide/.

The Software Manager interface allows you to view a list of all your registered systems as shown in Figure 27–1, *System List*, view details of Errata Alerts, install packages on entitled systems, apply Errata Updates to entitled systems, monitor the status of any pending actions, and much more.

Figure 27–1 System List



Current Selections (clear all):

- 1 Systems (clear selection)

System List

Click on a system's name to see details about that system, or select multiple systems by checking the boxes to the left and clicking "Update Selection List" below. To see a list of actions that can be performed on selected systems, go to the **system overview** page.

1-6 of **6** Assigned Systems Displayed (1 systems selected)

Select					Name	Entitled	Base Channel
☑	50	14	1	207	bean.devel.redhat.com	Yes	Red Hat Linux 7.0 i386
□	0	0	0	0	brush.devel.redhat.com	No	Red Hat Linux 7.1 i386
□	0	0	0	0	foo.devel.redhat.com	No	Red Hat Linux 7.0 i386
□	41	10	1	151	plate.devel.redhat.com	No	Red Hat Linux 7.0 i386
□	10	4	0	377	redhat-4	No	Red Hat Linux 7.1 i386
□	0	0	0	0	sugar.devel.redhat.com	No	Red Hat Linux 7.0 i386

1-6 of **6** Assigned Systems Displayed (1 systems selected)

Copyright © 2001 Red Hat, Inc. All rights reserved. Legal statement : Privacy statement

Part V Appendixes

A Building a Custom Kernel

Many people new to Linux often ask, "Why should I build my own kernel?" Given the advances that have been made in the use of kernel modules, the most accurate response to that question is, "Unless you already know why you need to build your own kernel, you probably do not need to."

In the past, you had to recompile the kernel if you added new hardware on your system. In other words, the kernel was **static**. Improvements in the Linux 2.0.*x* kernels allowed for many hardware drivers to be **modularized** into components that are loaded on demand. However, major problems existed when users had multiple kernels that had been compiled for different configuration options on their system; for example, SMP versus UP kernels. Further Linux 2.4.*x* kernel modularization advancements allow for multiple kernels to co-exist more easily, but they can not share modules.

For information on handling kernel modules see Chapter 24, *Kernel Modules*. Unless you are recompiling a customized kernel for your system, you will not see many changes in how kernel modules are handled.

A.1 Building a Modularized Kernel

The instructions in this section apply to building a modularized kernel. If you are interested in building a monolithic kernel instead, see Section A.4, *Building a Monolithic Kernel* for an explanation of the different aspects of building and installing a monolithic kernel.

The following steps will guide you through building a custom kernel for the x86 architecture:

> **Note**
>
> This example uses 2.4.7-3 as the kernel version. Your kernel version might differ. To determine your kernel version, type the command `uname -r`. Replace 2.4.7-3 with your kernel version.

1. The most important step is to make sure that you have a working emergency boot disk in case you make a mistake. If you didn't make a boot disk during the installation, use the `mkbootdisk` command to make one now. The standard command is similar to `mkbootdisk --device /dev/fd0 2.4.`*`x`*, where 2.4.*x* is the full version of your kernel (such as 2.4.7-3). Once done, test the boot disk to make sure that it will boot the system.

2. You must have both the `kernel-headers` and `kernel-source` packages installed. Issue the commands `rpm -q kernel-headers` and `rpm -q kernel-source` to determine their versions, if they are installed. If they are not installed, install them from the Red Hat Linux CD 1 or the Red Hat FTP site available at ftp://ftp.redhat.com (a list of mirrors is available at

http://www.redhat.com/mirrors.html). Refer to Chapter 25, *Package Management with RPM* for information on installing RPM packages.

3. Open a shell prompt and change to the directory `/usr/src/linux-2.4`. All commands from this point forward must be issued from this directory.

4. It is important that you begin a kernel build with the source tree in a known condition. Therefore, it is recommended that you begin with the command `make mrproper`. This will remove any configuration files along with the remains of any previous builds that may be scattered around the source tree. If you already have a working configuration file (`/usr/src/linux-2.4/.config`) that you want to use, back it up to a different directory before running this command and copy it back after running the command. If you use an existing configuration file, skip the next step.

5. Now you must create a configuration file that will determine which components to include in your new kernel.

   If you are running the X Window System, the recommended method is to use the command `make xconfig`. Components are listed in different levels of menus and are selected using a mouse. You can select **Y** (yes), **N** (no), or **M** (module). After choosing your components, click the **Save and Exit button** to create the configuration file `/usr/src/linux-2.4/.config` and exit the Linux Kernel Configuration program.

   Other available methods for kernel configuration are listed below:

   - `make config` — An interactive text program. Components are presented in a linear format and you answer them one at a time. This method does not require the X Window System and does not allow you to change your answers to previous questions.
   - `make menuconfig` — A text-mode, menu driven program. Components are presented in a menu of categories; you select the desired components in the same manner used in the text-mode Red Hat Linux installation program. Toggle the tag corresponding to the item you want included: **[*]** (built-in), **[ ]** (exclude), **<M>** (module), or **< >** (module capable). This method does not require the X Window System.
   - `make oldconfig` — This is a non-interactive script that will set up your configuration file to contain the default settings. If you're using the default Red Hat kernel, it will create a configuration file for the kernel that shipped with Red Hat Linux for your architecture. This is useful for setting up your kernel to known working defaults and then turning off features that you don't want.

Note

To use `kmod` (see Chapter 24, *Kernel Modules* for details) and kernel modules you must answer **`Yes`** to `kmod support` and `module version (CONFIG_MODVERSIONS) support` during the configuration.

6. After creating a `/usr/src/linux-2.4/.config` file, use the command `make dep` to set up all the dependencies correctly.

7. Use the command `make clean` to prepare the source tree for the build.

8. The next step in making a modularized kernel is to edit `/usr/src/linux-2.4/Makefile` so that you do not overwrite your existing kernel. The method described here is the easiest to recover from in the event of a mishap. If you are interested in other possibilities, details can be found at http://www.redhat.com/mirrors/LDP/HOWTO/Kernel-HOWTO.html or in the `Makefile` in `/usr/src/linux-2.4` on your Linux system.

   Edit `/usr/src/linux-2.4/Makefile` and modify the line beginning with `EXTRAVERSION =` to match a "unique" name by appending the date to the end of the string. For example, if you are compiling kernel version 2.4.7-3 you can append the flag to look similar to `EXTRAVERSION = -0.1.21-jul2001`). This will allow you to have the old working kernel and the new kernel, version 2.4.7-3-jul2001, on your system at the same time.

9. Build the kernel with `make bzImage`.

10. Build any modules you configured with `make modules`.

11. Install the kernel modules (even if you didn't build any) with `make modules_install`. Make sure that you type the underscore (_). This will install the kernel modules into the directory path `/lib/modules/`*`KERNELVERSION`*`/kernel/drivers` , where *KERNELVERSION* is the version specified in the `Makefile`. Our example would be `/lib/modules/2.4.7-3-jul2001/kernel/drivers/`.

12. If you have a SCSI adapter and you made your SCSI driver modular, build a new `initrd` image (see Section A.2, *Making an initrd Image*; note that there are few practical reasons to make the SCSI driver modular in a custom kernel). Unless you have a specific reason to create an `initrd` image, do not create one and do not add it to `lilo.conf`.

13. Use `make install` to copy your new kernel and its associated files to the proper directories.

14. The kernel is built and installed now. The next step is configuring the boot loader to boot the new kernel. Refer to Section A.3, *Configuring the Boot Loader* for more information.

A.2 Making an initrd Image

An initrd image is needed for loading your SCSI module at boot time. If you do not need an initrd image, do not make one and do not edit lilo.conf or grub.conf to include this image.

The /sbin/mkinitrd shell script can build a proper initrd image for your machine if the following conditions are met:

- The loopback block device is available.
- The /etc/modules.conf file has a line for your SCSI adapter; for example:

```
alias scsi_hostadapter BusLogic
```

To build the new initrd image, run /sbin/mkinitrd with parameters such as this:

```
/sbin/mkinitrd /boot/initrd-2.4.7-3-jul2001.img 2.4.7-3-jul2001
```

In the above example, /boot/initrd-2.4.7-3-jul2001.img is the filename of the new initrd image. 2.4.7-3-jul2001 is the kernel whose modules (from /lib/modules) should be used in the initrd image. This is not necessarily the same as the version number of the currently running kernel.

A.3 Configuring the Boot Loader

Now that you have recompiled your kernel, you must configure the boot loader to boot the new kernel. This is a crucial step. If you do not perform this step or if you perform it incorrectly, you will not be able to boot your system. If this happens, boot your system with the boot diskette you created earlier and try configuring the boot loader again. If your boot diskette does not work, refer to Chapter 3, *Rescue Mode* for more information about rescue mode.

In order to provide a redundant boot source to protect from a possible error in a new kernel, you should keep the original kernel available. During the installation of Red Hat Linux 7.2, you had the option to choose either LILO or GRUB as your boot loader. Refer to the appropriate section that follows.

A.3.1 GRUB

If you selected GRUB as your boot loader, you need to modify the file /boot/grub/grub.conf. The default GRUB configuration file looks similar to the following:

```
# NOTICE:  You have a /boot partition.  This means that
#          all kernel paths are relative to /boot/
default=0
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz
```

```
title Red Hat Linux (2.4.7-3)
        root (hd0,0)
        kernel /vmlinuz-2.4.7-3 ro root=/dev/hda3
        initrd /initrd-2.4.7-3.img
```

If you created a separate `/boot` partition, the paths to the kernel and initrd image are relative to the `/boot` partition.

To add your new kernel to GRUB, copy the existing title section to a new one and modify it to boot your new kernel image (and initrd image if you have any SCSI devices and have created an initrd image). Be sure the title of the new section is different from the title of the section to boot the old kernel. By default, Red Hat Linux uses `Red Hat Linux` and the kernel version in parentheses to differentiate between different kernels for GRUB to boot. In our example, the new `/boot/grub/grub.conf` file would look like the following:

```
# NOTICE:  You have a /boot partition.  This means that
#          all kernel paths are relative to /boot/
default=0
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz

title Red Hat Linux (2.4.7-3-jul2001)
        root (hd0,0)
        kernel /vmlinuz-2.4.7-3-jul2001 ro root=/dev/hda3
        initrd /initrd-2.4.7-3-jul2001.img

title Red Hat Linux (2.4.7-3)
        root (hd0,0)
        kernel /vmlinuz-2.4.7-3 ro root=/dev/hda3
        initrd /initrd-2.4.7-3.img
```

The default boot entry is set to number 0. To make your new kernel the default, either place its section first or change the default entry number to the appropriate number (remember that it starts counting with 0). For GRUB, you do not need to run any commands after modifying the configuration file.

From now on, when the system boots you will see `Red Hat Linux (2.4.7-3-jul2001)` and `Red Hat Linux (2.4.7-3)` as GRUB boot options. To boot the default kernel, press [Enter] or wait for it to time out. If you want to boot the old kernel, select it and press [Enter].

You can begin testing your new kernel by rebooting your computer and watching the messages to ensure your hardware is detected properly.

A.3.2 LILO

To configure LILO to boot the new kernel, you need to update the `/etc/lilo.conf` file and run the command `/sbin/lilo -v`.

The default `/etc/lilo.conf` file looks similar to the following:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
linear
default=linux

image=/boot/vmlinuz-2.4.7-3
 label=linux
        initrd=initrd-2.4.7-3.img
 read-only
 root=/dev/hda5
```

To add your new kernel to LILO, copy the existing image section to a new one and modify it to boot your new kernel image (and initrd image if you have any SCSI devices and have created an initrd image). Also, rename the label of the old kernel to something such as **`linux-old`**. Your `/etc/lilo.conf` should look similar to the following:

```
boot=/dev/hda
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
linear
default=linux

image=/boot/vmlinuz-2.4.7-3-jul2001
label=linux
initrd=initrd-2.4.7-3-jul2001.img
read-only
root=/dev/hda5

image=/boot/vmlinuz-2.4.7-3
label=linux-old
initrd=initrd-2.4.7-3.img
read-only
```

```
    root=/dev/hda5
```

To activate your changes, run the command `/sbin/lilo -v`. If all goes well, you will see output similar to the following:

```
LILO version 21.4-4, Copyright (C) 1992-1998 Werner Almesberger
'lba32' extensions Copyright (C) 1999,2000 John Coffman

Reading boot sector from /dev/hda
Merging with /boot/boot.b
Mapping message file /boot/message
Boot image: /boot/vmlinuz-2.4.7-3
Added linux *
Boot image: /boot/vmlinuz-2.4.7-3-jul2001
Added linux-old
Writing boot sector.
```

Be sure the messages contains `Writing boot sector.` The * after linux means that the section labeled `linux` is the default kernel that LILO will boot.

From now on, when the system boots you will see `linux` and `linux-old` as LILO boot options.

To boot the new kernel (`linux`) simply press [Enter], or wait for LILO to time out. If you want to boot the old kernel (`linux-old`), select `linux-old` and press [Enter].

You can begin testing your new kernel by rebooting your computer and watching the messages to ensure your hardware is detected properly.

A.4 Building a Monolithic Kernel

To build a monolithic kernel, follow the same steps as building a modularized kernel, with a few exceptions.

- When configuring the kernel, do not compile anything as a module. In other words, only answer **`Yes`** or **`No`** to the questions. Also, you should answer **`No`** to `kmod support` and `module version (CONFIG_MODVERSIONS) support`.
- Omit the following steps:

  ```
  make modules
  make modules_install
  ```

- Edit `lilo.conf` and add the line `append=nomodules`.

B Getting Started with Gnu Privacy Guard

B.1 An Introduction to GnuPG

Have you ever wondered if your email can be read during its transmission from you to other people, or from other people to you? Unfortunately, complete strangers could conceivably intercept or even tamper with your email.

In traditional (also known as "snail") mail, letters are usually sealed within envelopes, stamped and delivered from post office branch to branch until they reach their destination. But sending mail through the Internet is much less secure; email is usually transmitted as unencrypted text from server to server. No special steps are taken to protect your correspondence from being seen or tampered with by other people.

To help you protect your privacy, Red Hat Linux 7.2 includes GnuPG, the GNU Privacy Guard, which is installed by default during a typical Red Hat Linux installation. It is also referred to as GPG.

GnuPG is a tool for secure communication; it is a complete and free replacement for the encryption technology of PGP (Pretty Good Privacy, a widely popular encryption application). Using GnuPG, you can encrypt your data and correspondence, and authenticate your correspondence by **digitally signing** your work. GnuPG is also capable of decrypting and verifying PGP 5.*x*.

Because GnuPG is compatible with other encryption standards, your secure correspondence will probably be compatible with email applications on other operating systems, such as Windows and Macintosh.

GnuPG uses **public key cryptography** to provide users with a secure exchange of data. In a public key cryptography scheme, you generate two keys: a public key and a private key. You exchange your public key with correspondents or with a keyserver; you should never reveal your private key.

Encryption depends upon the use of keys. In conventional or symmetric cryptography, both ends of the transaction have the same key, which they use to decode each other’s transmissions. In public key cryptography, two keys co-exist: a public key and a private key. A person or an organization keeps their private key a secret, and publishes their public key. Data encoded with the public key can only be decoded with the private key; data encoded with the private key can only be decoded with the public key.

Do Not Reveal Your Private Key

Remember that your public key can be given to anyone with whom you want to communicate securely, but you must never give away your private key.

For the most part, cryptography is beyond the scope of this publication; volumes have been written about the subject. In this chapter, however, we hope you'll gain enough understanding about GnuPG to begin using cryptography in your own correspondence. For more information about GnuPG, including an online users guide, visit http://www.gnupg.org/. If you want to learn more about GnuPG, PGP and encryption technology, see Section B.7, *Additional Resources*.

More Information From the Shell Prompt

Like most system tools for Red Hat Linux, you'll find documentation on GnuPG in the man pages and info pages. At a shell prompt, just type `man gpg` or `info gpg` for a quick reference of GnuPG commands and options.

B.2 Generating a Keypair

To begin using GnuPG, you must first generate a new keypair: a public key and a private key.

To generate a keypair, at a shell prompt, type the following command:

```
gpg --gen-key
```

Since you work with your user account most frequently, you should perform this action while logged in to your user account (and not as root).

You will see an introductory screen, with key options, including one recommended option (the default), similar to the following:

```
gpg (GnuPG) 1.0.1; Copyright (C) 1999 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

Please select what kind of key you want:
(1) DSA and ElGamal (default)
(2) DSA (sign only)
(4) ElGamal (sign and encrypt)
Your selection?
```

In fact, most of the screens which require you to choose an option will list the default option, within parentheses. You can accept the default options simply by pressing [Enter].

In the first screen, you should accept the default option: `(1) DSA and ElGamal`. This option will allow you to create a digital signature and encrypt (and decrypt) with two types of technologies. Type **1** and then press [Enter].

Next, choose the key size, or how long the key should be. Generally, the longer the key, the more resistant against attacks your messages will be. The default size, 1024 bits, should be sufficiently strong for most users, so press [Enter].

The next option asks you to specify how long you want your key to be valid. Usually, the default (`0 = key does not expire`) is fine. If you do choose an expiration date, remember that anyone with whom you exchanged your public key will also have to be informed of its expiration, and supplied with a new public key.

Your next task is to provide a user ID, with your name, your email address, and an optional comment. When you are finished, you'll be presented with a summary of the information you entered.

Once you accept your choices, you'll have to enter a passphrase.

Use a Good Passphrase

Like your account passwords, a good passphrase is essential for optimal security in GnuPG. For example, mix your passphrase with upper- and lowercase letters, use numbers, or punctuation marks.

Once you enter and verify your passphrase, your keys will be generated. You will see a message similar to the following:

```
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
+++++.+++++.++++++++....++++++++++..+++++.+++++.+++++++.+++++++
+++.++++++++++++++++++++++++++++++++++++++..........................++++
```

When the activity on the screen ceases, your new keys will be made and placed in the directory `.gnupg` in your home directory. To list your keys, use the command `gpg --list-keys`; you'll see something similar to the following:

```
[newuser@localhost newuser]$ gpg --list-keys
/home/newuser/.gnupg/pubring.gpg
-----------------------------------------
pub  1024D/B7085C8A 2000-04-18 Your Name <you@yourisp.net>
```

```
sub  1024g/E12AF9C4 2000-04-18
```

B.3 Generating a Revocation Certificate

Once you have created your keypair, you should create a revocation certificate for your public key. If you forget your passphrase, or if it has been compromised, you can publish this certificate to inform users that your public key should no longer be used.

Why Revoke a Key You Just Created?

When you generate a revocation certificate, you are not revoking the key you just created. Instead, you're giving yourself a safe way to revoke your key from public use. Let's say you create a key, then you forget your passphrase, switch ISPs (addresses), or suffer a hard drive crash. The revocation certificate can then be used to disqualify your public key.

Your signature will be valid to others who read your correspondence before your key is revoked, and you will be able to decrypt messages received prior to its revocation. To generate a revocation certificate, use the `--gen-revoke` option.

```
[newuser@localhost newuser]$ gpg --output revoke.asc
--gen-revoke  <you@yourisp.net>
```

Note that if you omit the `--output revoke.asc` option from the above, your revocation certificate will be returned to the standard output, which is your monitor screen. While you can copy and paste the contents of the output into a file of your choice using a text editor, such as Pico, it is probably easier to send the output to a file in your login directory. That way, you can keep the certificate for use later, or move it to a floppy disk and store it someplace safe.

The creation of a revocation certificate will look like the following:

```
[newuser@localhost newuser]$ gpg --output revoke.asc
--gen-revoke  <you@yourisp.net>

 sec  1024D/823D25A9 2000-04-26  Your Name <you@yourisp.net>

 Create a revocation certificate for this key? y

 You need a passphrase to unlock the secret key for
 user: "Your Name <you@yourisp.net>"
 1024-bit DSA key, ID 823D25A9, created 2000-04-26

 ASCII armored output forced.
 Revocation certificate created.
```

Once your revocation certificate has been created (`revoke.asc`), it will be located in your login directory. You should copy the certificate to a floppy diskette and store it in a secure place. (If you don't know how to copy a file to a diskette in Red Hat Linux, see the *Official Red Hat Linux Getting Started Guide*.)

B.4 Exporting your Public Key

Before you can use public key cryptography, other people must have a copy of your public key. To send your key to correspondents or to a keyserver, you must **export** the key.

To export your key, so you can display it on a Web page or paste it in email, type the following:

```
[newuser@localhost newuser]$ gpg --armor --export
<you@yourisp.net> > mykey.asc
```

You will not see any output, because not only did you export your public key, you redirected the output to a file called, for example, `mykey.asc`. (Without the addition of `> mykey.asc`, the key would have been displayed as the standard output on the monitor screen.)

Now, the file `mykey.asc` can be inserted into email or exported to a keyserver. To see the key, type `less mykey.asc` to open the file in a pager (type [q] to quit the pager). It should look like the following:

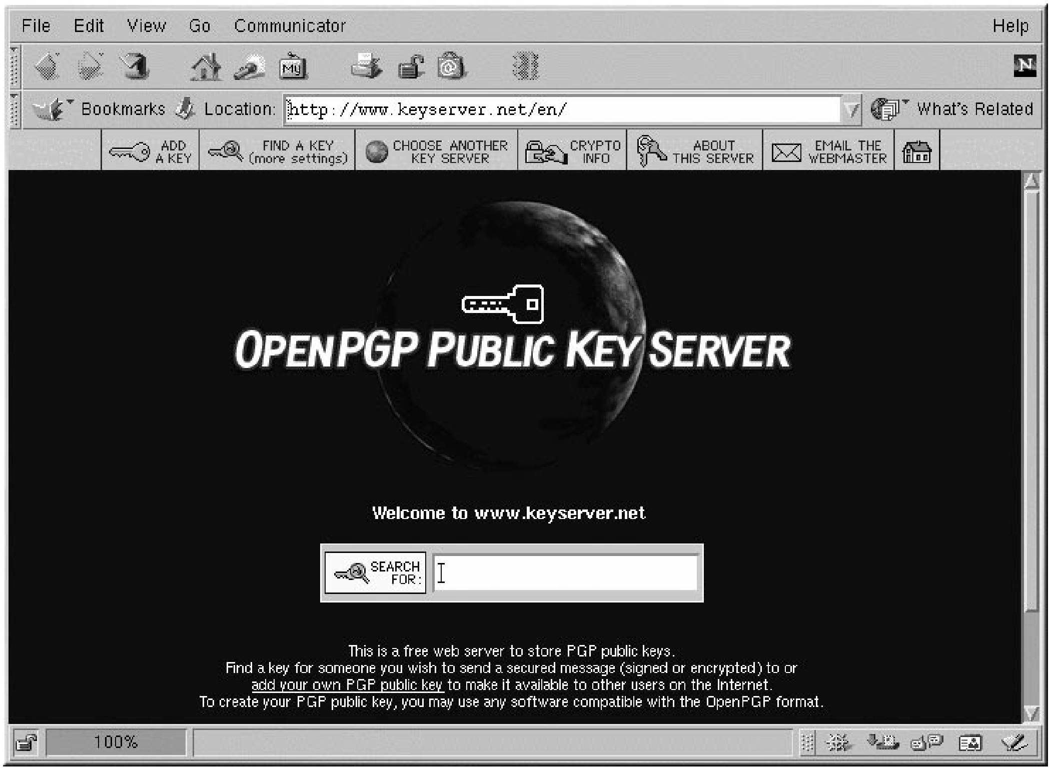
```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.0.1 (GNU/Linux)
Comment: For info see http://www.gnupg.org

mQGiBDkHP3URBACkWGsYh43pkXU9wj/X1G67K8/DSrl85r7dNtHNfLL/ewil10k2
q8saWJn26QZPsDVqdUJMOdHfJ6kQTAt9NzQbgcVrxLYNfgeBsvkHF/POtnYcZRgL
tZ6syBBWs8JB4xt5V09iJSGAMPUQE8Jpdn2aRXPApdoDw179LM8Rq6r+gwCg5ZZa
pGNlkgFu24WM5wC1zg4QTbMD/3MJCSxfL99Ek5HXcB3yhj+o0LmIrGAVBgoWdrRd
BIGjQQFhV1NSwC8YhN/4nGHWpaTxgEtnb4CI1wI/G3DK9olYMyRJinkGJ6XYfP3b
cCQmqATDF5ugIAmdditnw7deXqn/eavaMxRXJM/RQSgJJyVpbAO2OqKe6L6Inb5H
kjcZA/9obTm499dDMRQ/CNR92fA5pr0zriy/ziLUow+cqI59nt+bEb9nY1mfmUN6
SW0jCH+pIQH5lerV+EookyOyq3ocUdjeRYF/d2jl9xmeSyL2H3tDvnuE6vgqFU/N
sdvby4B2Iku7S/h06W6GPQAe+pzdyX9vS+Pnf8osu7W3j60WprQkUGF1bCBHYWxs
YWdoZXIgPHBhdWxnYWxsQHJlZGhhdC5jb20+iFYEExECABYFAjkHP3UECwoEAwMV
AwIDFgIBAheAAAoJEJECmvGCPSWpMjQAoNF2zvRgdR/8or9pBhu95zeSnkb7AKCm
/uXVS0a5KoN7J61/1vEwx11poLkBDQQ5Bz+MEAQA8ztcWRJjW8cHCgLaE402jyqQ
37gDT/n4VS66nU+YItzDFScVmgMuFRzhibLblfO9TpZzxEbSF3T6p9hLLnHCQ1bD
HRsKfh0eJYMMqB3+HyUpNeqCMEEd9AnWD9P4rQtO7Pes38sV0lX0OSvsTyMG9wEB
vSNZk+Rl+phA55r1s8cAAwUEAJjqazvk0bgFrw1OPG9m7fEeDlvPSV6HSA0fvz4w
c7ckfpuxg/URQNf3TJA00Acprk8Gg8J2CtebAyR/sP5IsrK5l1luGdk+l0M85FpT
/cen2OdJtToAF/6fGnIkeCeP1O5aWTbDgdAUHBRykpdWU3GJ7NS6923fVg5khQWg
uwrAiEYEGBECAAYFAjkHP4wACgkQkQKa8YI9JamliwCfXox/HjlorMKnQRJkeBcZ
iLyPH1QAoI33Ft/0HBqLtqdtP4vWYQRbibjW
```

```
=BMEc
-----END PGP PUBLIC KEY BLOCK-----
```

B.4.1 Exporting to a Keyserver

If you are only writing to a few correspondents, you can export your public key and send it to them personally. If you correspond with many people, however, distribution of your key can be time consuming. Instead, you can use a keyserver.

Figure B–1 The Home Page of Keyserver.Net



http://www.keyserver.net/en/

Welcome to www.keyserver.net

This is a free web server to store PGP public keys.
Find a key for someone you wish to send a secured message (signed or encrypted) to or
add your own PGP public key to make it available to other users on the Internet.
To create your PGP public key, you may use any software compatible with the OpenPGP format.

A keyserver is a repository on the Internet which can store and distribute your public key to anyone who requests it. Many keyservers are available, and most try to remain synchronized with each other; sending your key to one keyserver is like distributing it to them all. A correspondent can request your public key from a from a keyserver, import that key to their keyring, and they are ready for secure correspondence with you.

Which Keyserver Should You Use?

Because most keyservers are synchronized, sending your public key to one keyserver is usually as good as sending it to them all. You can, however, locate different keyservers. One place to begin your search for keyservers and more information is *Keyserver.Net*, at http://www.keyserver.net; another location is *Robert's Crypto & PGP Links: Keyservers*, at http://crypto.yashy.com/www/Keyservers/.

You can send your public key from either the shell prompt or from a browser (as in Figure B–1, *The Home Page of Keyserver.Net*); of course, you must be online to send or receive keys from a keyserver.

- From the shell prompt, type the following:

  ```
  gpg --keyserver search.keyserver.net --send-key you@yourisp.net
  ```

- From your browser, go to Keyserver.Net (http://www.keyserver.net) and select the option to add your own PGP public key.

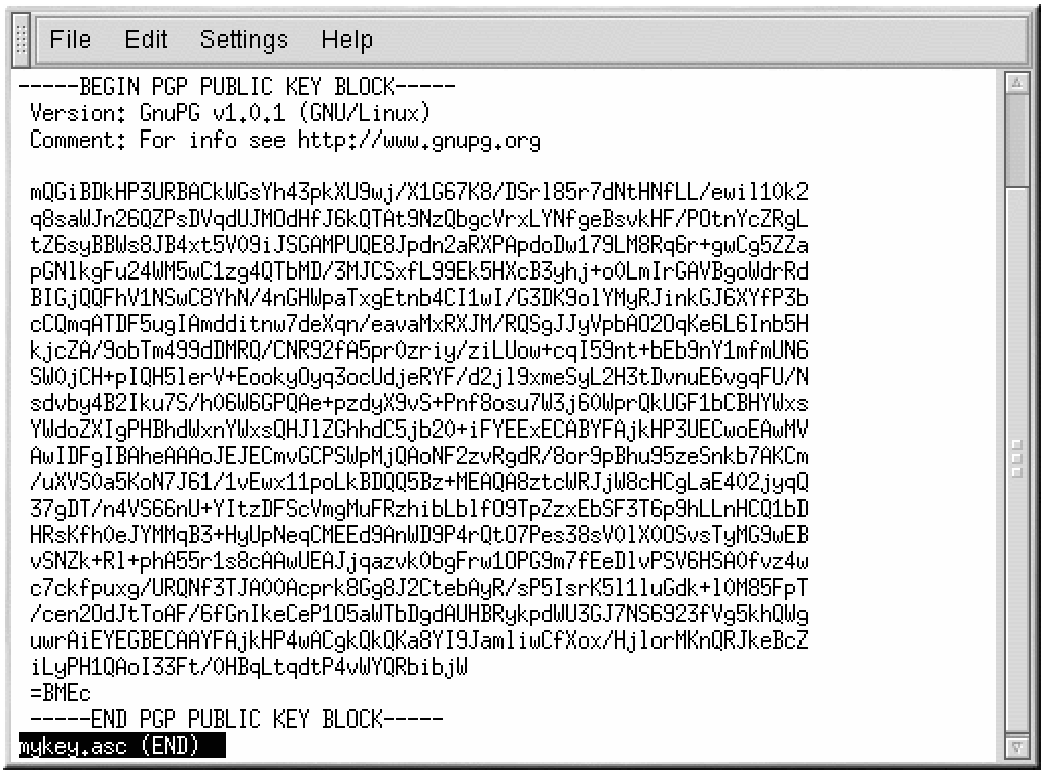
  Your next task is to copy and paste your public key into the appropriate area on the Web page. If you need instructions on how to do that, use the following:

  – Open your exported public key file (such as *mykey.asc*, which was created in Section B.4, *Exporting your Public Key*) with a pager — for example, use the `less mykey.asc` command.

  – Using your mouse, copy the file by highlighting all the lines from the `BEGIN PGP` to `END PGP` notations (see Figure B–2, *Copying Your Public Key*).

  – Paste the contents of the file *mykey.asc* into the appropriate area of the page on Keyserver.Net by middle-clicking with your mouse (or left- and right-clicking if you're using a two-button mouse). Then select the **Submit** button on the keyserver page. (If you make a mistake, press the **Reset** button on the page to clear your pasted key.)

Figure B–2 Copying Your Public Key



```
-----BEGIN PGP PUBLIC KEY BLOCK-----
 Version: GnuPG v1.0.1 (GNU/Linux)
 Comment: For info see http://www.gnupg.org

 mQGiBDkHP3URBACkWGsYh43pkXU9wj/X1G67K8/DSrl85r7dNtHNfLL/ewil1Ok2
 q8saWJn26QZPsDVqdUJMOdHfJ6kQTAt9NzQbgcVrxLYNfgeBsvkHF/POtnYcZRgL
 tZ6syBBWs8JB4xt5VO9iJSGAMPUQE8Jpdn2aRXPApdoDw179LM8Rq6r+gwCg5ZZa
 pGNlkgFu24WM5wC1zg4QTbMD/3MJCSxfL99Ek5HXcB3yhj+oOLmIrGAVBgoWdrRd
 BIGjQQFhV1NSwC8YhN/4nGHWpaTxgEtnb4CI1wI/G3DK9olYMyRJinkGJ6XYfP3b
 cCQmqATDF5ugIAmdditnw7deXqn/eavaMxRXJM/RQSgJJyVpbAO2OqKe6L6Inb5H
 kjcZA/9obTm499dDMRQ/CNR92fA5prOzriy/ziLUow+cqI59nt+bEb9nY1mfmUN6
 SWOjCH+pIQH5lerV+EookyOyq3ocUdjeRYF/d2jl9xmeSyL2H3tDvnuE6vgqFU/N
 sdvby4B2Iku7S/hO6W6GPQAe+pzdyX9vS+Pnf8osu7W3j6OWprQkUGF1bCBHYWxs
 YWdoZXIgPHBhdWxnYWxsQHJlZGhhdC5jb20+iFYEExECABYFAjkHP3UECwoEAwMV
 AwIDFgIBAheAAAoJEJECmvGCPSWpMjQAoNF2zvRgdR/8or9pBhu95zeSnkb7AKCm
 /uXVSOa5KoN7J61/1vEwx11poLkBDQQ5Bz+MEAQA8ztcWRJjW8cHCgLaE402jyqQ
 37gDT/n4VS66nU+YItzDFScVmgMuFRzhibLblfO9TpZzxEbSF3T6p9hLLnHCQ1bD
 HRsKfhOeJYMMqB3+HyUpNeqCMEEd9AnWD9P4rQtO7Pes38sVOlXOOSvsTyMG9wEB
 vSNZk+Rl+phA55r1s8cAAwUEAJjqazvkObgFrw1OPG9m7fEeDlvPSV6HSAOfvz4w
 c7ckfpuxg/URQNf3TJAOOAcprk8Gg8J2CtebAyR/sP5IsrK5l1luGdk+lOM85FpT
 /cen2OdJtToAF/6fGnIkeCeP1O5aWTbDgdAUHBRykpdWU3GJ7NS6923fVg5khQWg
 uwrAiEYEGBECAAYFAjkHP4wACgkQkQKa8YI9JamliwCfXox/HjlorMKnQRJkeBcZ
 iLyPH1QAoI33Ft/OHBqLtqdtP4vWYQRbibjW
 =BMEc
 -----END PGP PUBLIC KEY BLOCK-----
mykey.asc (END)
```

Note that if you are submitting your key to another Web-based keyserver, the above transaction will be essentially the same.

That is all you need to do. Regardless of whether you use the shell prompt or the Web, you will see a message that your key was successfully submitted — either at the shell prompt or at the keyserver's website. From now on, users who want to communicate securely with you can import your public key and add it to their keyring.

B.5 Importing a Public Key

The other end of key exchange — importing other people's public keys to your keyring — is just as simple as exporting keys. When you import someone's public key, you can decrypt their mail and check their digital signature against their public key on your keyring.

One of the easiest ways to import a key is to download the key or save it from a website. To learn how to import Red Hat's key, refer to Section 25.3.1, *Importing Keys*.

After downloading a key, use the command `gpg --import key.asc` to add it to your keyring.

Another way to save a key is to use a browser's **Save As** feature. If you are using a browser such as Navigator, and you locate a key at a keyserver, you can save the page as a text file (go to **File** => **Save As**). In the drop-down box next to **Format for saved document**, choose **Text**. Then, you can import the key — but remember the name of the file you saved. For example, if you saved a key as a text file called *newkey.txt*, to import the file, at a shell prompt, type:

```
[newuser@localhost newuser]$ gpg --import newkey.txt
 gpg: key F78FFE84: public key imported
   gpg: Total number processed: 1
   gpg:               imported: 1
```

To check that the process was successful, use the `gpg --list-keys` command; you should see your newly imported key listed on your keyring.

B.6 What Are Digital Signatures?

Digital signatures can be compared to your written signature. Unlike traditional correspondence, in which it might be possible to tamper with your written signature, digital signatures can not be forged. That is because the signature is created with your unique secret key, and can be verified by your recipient using your public key.

A digital signature timestamps a document; essentially, that means that the time you signed the document is part of that signature. So if anyone tries to modify the document, the verification of the signature will fail. Some email applications, such as Exmh or KDE's KMail, include the ability to sign documents with GnuPG within the application's interface.

Two useful types of digital signatures are **clearsigned** documents and **detached signatures**. Both types of signatures incorporate the same security of authenticity, without requiring your recipient to decrypt your entire message.

In a clearsigned message, your signature appears as a text block within the context of your letter; a detached signature is sent as a separate file with your correspondence.

B.7 Additional Resources

There is more to encryption technology than can be covered in one slim introduction to GnuPG. Here are some resources where you can learn more.

B.7.1 Useful Websites

- http://www.gnupg.org — The GnuPG website with links to the latest GnuPG releases, a comprehensive user’s guide, and other cryptography resources.
- http://hotwired.lycos.com/webmonkey/backend/security/tutorials/tutorial1.html — Visit the *Encryption Tutorial* from Webmonkey to learn more about encryption and how to apply encryption techniques.
- http://www.eff.org/pub/Privacy — The Electronic Frontier Foundation, "Privacy, Security, Crypto, & Surveillance" Archive.

B.7.2 Related Books

- *The Official PGP User’s Guide* by Philip R. Zimmerman; MIT Press
- *PGP: Pretty Good Privacy* by Simson Garfinkel; O’Reilly & Associates, Inc.
- *E-Mail Security: How to Keep Your Electronic Messages Private* by Bruce Schneier; John Wiley & Sons

Index