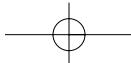


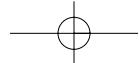
Chapter 2

Hardening the Operating System

Solutions in this chapter:

- Updating the Operating System
- Handling Maintenance Issues
- Manually Disabling Unnecessary Services and Ports
- Locking Down Ports
- Hardening the System with Bastille
- Controlling and Auditing Root Access with Sudo
- Managing Your Log Files
- Using Logging Enhancers
- Security Enhanced Linux
- Securing Novell SUSE Linux
- Novell AppArmor
- Host Intrusion Prevention System
- Linux Benchmark Tools





Introduction

Linux is capable of high-end security; however, the out-of-the-box configurations must be altered to meet the security needs of most businesses with an Internet presence. This chapter shows you the steps for securing a Linux system—called *hardening* the server—using both manual methods and open source security solutions. The hardening process focuses on the operating system, and is important regardless of the services offered by the server. The steps will vary slightly between services, such as e-mail and Hypertext Transfer Protocol (HTTP), but are essential for protecting any server that is connected to a network, especially the Internet. Hardening the operating system allows the server to operate efficiently and securely.

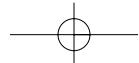
This chapter includes the essential steps an administrator must follow to harden a Unix system; specifically, a Red Hat Linux system. These steps include updating the system, disabling unnecessary services, locking down ports, logging, and maintenance. Later in this chapter you may find some information for Novell SUSE Linux. Open source programs allow administrators to automate these processes using Bastille, sudo, logging enhancers such as SWATCH, and antivirus software. Before you implement these programs, you should first understand how to harden a system manually.

Updating the Operating System

An operating system may contain many security vulnerabilities and software bugs when it is first released. Vendors, such as Red Hat, provide updates to the operating system to fix these vulnerabilities and bugs. In fact, many consulting firms recommend that companies do not purchase and implement new operating systems until the first update is available. In most cases, the first update will fix many of the problems encountered with the first release of the operating system. In this section, you will learn where to find the most current Red Hat Linux errata and updates.

Red Hat Linux Errata and Update Service Packages

The first step in hardening a Linux server is to apply the most current errata and Update Service Package to the operating system. The Update Service Package provides the latest fixes and additions to the operating system. It is a collection of fixes, corrections, and updates to the Red Hat products, such as bug fixes, security advisories, package enhancements, and add-on software. Updates can be downloaded individually as errata, but it is a good idea to start with the latest Update Service Package, and then install errata as necessary. However, you must pay to receive the Update Service Packages, and the errata are free. Many errata and Update Service Packages are not required upgrades. You need to read the documentation to determine if you need to install it.



The Update Service Packages include all of the errata in one package to keep your system up to date. After you pay for the service, you can download them directly from the Red Hat Web site. To find out more about the Update Service Packages, visit the secure site www.redhat.com/apps/support/.

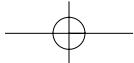
You may also launch the **Software Updater** from Applications | **System Tools** | **Software Updater** from the taskbar (Red Hat Enterprise Linux 5). You have to register yourselves with RHN (Red Hat Network) and send the hardware and software profile for Red Hat to recommend appropriate updates for your system. Figure 2.1 shows the registration process through Software Updater.

Figure 2.1 Software Updater



Handling Maintenance Issues

You should apply the latest service pack and updates before the server goes live, and constantly maintain the server after it is deployed to make sure the most current required patches are installed. The more time an operating system is available to the public, the more time malicious hackers have to exploit discovered vulnerabilities. Vendors offer patches to fix these vulnerabilities as quickly as possible; in some cases, the fixes are available at the vendor's site the same day.



20 Chapter 2 • Hardening the Operating System

Administrators must also regularly test their systems using security analyzer software. Security analyzer software scans systems to uncover security vulnerabilities, and recommends fixes to close the security hole.

This section discusses the maintenance required to ensure that your systems are safe from the daily threats of the Internet.

Red Hat Linux Errata: Fixes and Advisories

Once your Red Hat system is live, you must make sure that the most current required Red Hat errata are installed. These errata include bug fixes, corrections, and updates to Red Hat products. You should always check the Red Hat site at www.redhat.com/apps/support for the latest errata news. The following list defines the different types of errata found at the Red Hat Updates and Errata site.

- **Bug fixes** Address coding errors discovered after the release of the product, and may be critical to program functionality. These Red Hat Package Manager tools (RPMs) can be downloaded for free. Bug fixes provide a fix to specific issues, such as a certain error message that may occur when completing an operating system task. Bug fixes should only be installed if your system experiences a specific problem. Another helpful resource is Bugzilla, the Red Hat bug-tracking system at <https://bugzilla.redhat.com/>. You may report a bug that you have encountered in your system through Bugzilla. Figure 2-2 shows one such notification of a bug by a user.
- **Security advisories** Provide updates that eliminate security vulnerabilities on the system. Red Hat recommends that all administrators download and install the security upgrades to avoid denial-of-service (DoS) and intrusion attacks that can result from these weaknesses. For example, a security update can be downloaded for a vulnerability that caused a memory overflow due to improper input verification in Netscape's Joint Photographic Experts Group (JPEG) code. Security updates are located at <http://www.redhat.com/security/updates/>
- **Package enhancements** Provide updates to the functions and features of the operating system or specific applications. Package enhancements are usually not critical to the system's integrity; they often fix functionality programs, such as an RPM that provides new features.

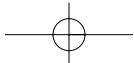


Figure 2.2 Notification of a Bug through Bugzilla

Bugzilla Bug 236416: RHEL 5 fails to get EDID data from monitor and sets low resolution

Last Comment | Clone as Bug | Show Bug Activity | Format for Printing

Alias		Priority	high
Product	Red Hat Enterprise Linux	Severity	high
Version	5	Status	ASSIGNED
Component	xorg-x11-drv-vesa	Resolution	
OS	Linux	Add CC	
Hardware	i386	CC	cra@wpi.edu ddomingo@redhat.com jneely@ncsu.edu riek@redhat.com
Reporter	(wdc@mit.edu)	<input type="checkbox"/> Remove selected CCs	
Assigned To	ajax (ajackson@redhat.com)		

Save Changes

Bug Comments

Opened by (wdc@mit.edu) on 2007-04-13 14:41 EST [reply]

Description of problem:

I just installed RHEL 5 client, and noticed that sometimes the X resolution is properly set, as I specified, to 1200x1024, but often, upon restart of the X server, it dumbs down the resolution to 800x600.

I will attach two Xorg.0.log outputs showing how the VESA BBE DDE read is said to be successful,

You also have an option of sending the bug through the **Bug Reporting Tool**. This pops-up automatically when you encounter an error during your routine work on your system. Figure 2.3 shows the Bug Reporting tool.

If you click on Show details you may find the information shown below (partial output shown here). This information is based on the nature of the bug, software and hardware configuration, and will vary from system to system. Though you may not be able to make out all that is captured by the bug reporting tool, experts in the Red Hat support will be able decode the same and work on the fixes.

22 Chapter 2 • Hardening the Operating System

Figure 2.3 Bug Reporting Tool



```
Distribution: Red Hat Enterprise Linux Server release 5 (Tikanga)
Gnome Release: 2.16.0 2006-09-04 (Red Hat, Inc)
BugBuddy Version: 2.16.0
```

```
Memory status: size: 147779584 vsize: 0 resident: 147779584 share: 0 rss: 68427776
rss_rlim: 0
CPU usage: start_time: 1189756814 rtime: 0 utime: 2224 stime: 0 cutime:2027 cstime:
0 timeout: 197 it_real_value: 0 frequency: 93
```

```
Backtrace was generated from '/usr/bin/yelp'
```

```
(no debugging symbols found)
Using host libthread_db library "/lib/libthread_db.so.1".
(no debugging symbols found)
[Thread debugging using libthread_db enabled]
[New Thread -1208363296 (LWP 3961)]
[New Thread -1255404656 (LWP 4181)]
[New Thread -1243546736 (LWP 3963)]
[New Thread -1210463344 (LWP 3962)]
(no debugging symbols found)
(no debugging symbols found)
```

```

0x002ae402 in __kernel_vsyscall ()
#0 0x002ae402 in __kernel_vsyscall ()
#1 0x0033dc5b in __waitpid_nocancel () from /lib/libpthread.so.0
#2 0x051d1c26 in gnome_gtk_module_info_get () from /usr/lib/libgnomeui-2.so.0
#3 <signal handler called>
. . .
#48 0x08051811 in g_cclosure_marshall_VOID_VOID ()

Thread 4 (Thread -1210463344 (LWP 3962)):
#0 0x002ae402 in __kernel_vsyscall ()
No symbol table info available.
#1 0x0090a5b3 in poll () from /lib/libc.so.6
No symbol table info available.
. . .
#8 0x0091414e in clone () from /lib/libc.so.6
No symbol table info available.

Thread 2 (Thread -1255404656 (LWP 4181)):
#0 0x002ae402 in __kernel_vsyscall ()
No symbol table info available.
#1 0x0033a3cc in pthread_cond_timedwait@@GLIBC_2.3.2 ()
    from /lib/libpthread.so.0
. . .
#48 0x08051811 in g_cclosure_marshall_VOID_VOID ()
No symbol table info available.
#0 0x002ae402 in __kernel_vsyscall ()

```

Bug Fix Case Study

Once you register your system with Red Hat Network, time-to-time you may receive emails with a subject ‘RHN Errata Alert’. These alerts are specific to the system you registered consisting summary of the problem, a detailed description and the actions recommended to resolve the problem.

In this case study the following mail received from Red Hat provides the details of ‘kernel security update’ required by the registered system (partial output shown):

Red Hat Network has determined that the following advisory is applicable to one or more of the systems you have registered:

Complete information about this errata can be found at the following location:
<https://rhn.redhat.com/rhn/errata/details/Details.do?eid=5984>

Security Advisory - RHSA-2007:0705-2

Summary:

Important: kernel security update

Updated kernel packages that fix various security issues in the Red Hat Enterprise Linux 5 kernel are now available.



24 Chapter 2 • Hardening the Operating System

This update has been rated as having important security impact by the Red Hat Security Response Team.

Description:

The Linux kernel handles the basic functions of the operating system.

These new kernel packages contain fixes for the following security issues:

* a flaw in the DRM driver for Intel graphics cards that allowed a local user to access any part of the main memory. To access the DRM functionality a user must have access to the X server which is granted through the graphical login. This also only affected systems with an Intel 965 or later graphic chipset. (CVE-2007-3851, Important)

* a flaw in the VFAT compat ioctl handling on 64-bit systems that allowed a local user to corrupt a kernel_dirent struct and cause a denial of service (system crash). (CVE-2007-2878, Important)

. (output truncated)

Red Hat Enterprise Linux 5 users are advised to upgrade to these packages, which contain backported patches to correct these issues.

References:

<http://www.redhat.com/security/updates/classification/#important>

Taking Action

You may address the issues outlined in this advisory in two ways:

- select your server name by clicking on its name from the list available at the following location, and then schedule an errata update for it:
<https://rhn.redhat.com/rhn/systems/SystemList.do>

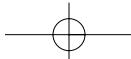
- run the Update Agent on each affected server.

. (output truncated)

Affected Systems List

This Errata Advisory may apply to the systems listed below. If you know that this errata does not apply to a system listed, it might be possible that the package profile for that server is out of date. In that case you should run 'up2date -p' as root on the system in question to refresh your software profile.

There is 1 affected system registered in 'Your RHN' (only systems for which you



have explicitly enabled Errata Alerts are shown).

Release	Arch	Profile Name
5Server	i686	linux11

The Red Hat Network Team

As you may notice from the above mail the registered system requires a kernel security update. Now you need to follow the steps outlined under ‘Taking Action’ section to ensure your system is updated. In this case this advisory recommends you schedule errata update and run the Update Agent on the affected server.

Manually Disabling Unnecessary Services and Ports

As a Linux administrator or a security administrator it is essential for you to define the following:

- Role of the server (web, database, proxy, ftp, dns, dhcp or others)
- Services that are required to perform a specific server role (for example, Apache for web server)
- Ports required to be opened (for example, HTTP, port 80)

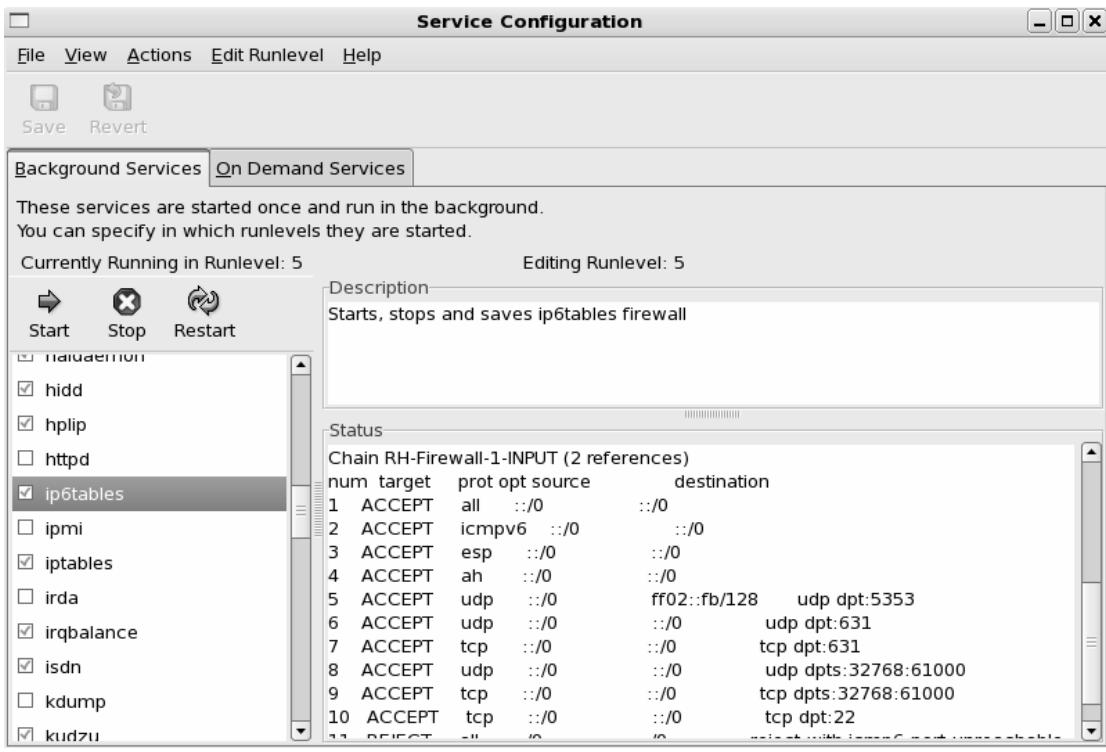
All the other services should be disabled and all other ports to be closed. When the above tasks are performed, the server becomes a specialized server to play only the designated role.

To harden a server, you must first disable any unnecessary services and ports. This process involves removing any unnecessary services, such as the Linux rlogin service, and locking down unnecessary Transmission Control Protocol/User Datagram Protocol (TCP/UDP) ports. Once these services and ports are secure, you must then regularly maintain the system. Figure 2-4 shows Service Configuration in Red Hat Linux.

System | Administration | Services opens the **Service Configuration** utility. You may select or deselect the services, start, stop or restart and edit the run level of individual services. In the Figure 2.4 you may notice the service ‘ip6tables’ is enabled, and the Description of the service and status is displayed.

26 Chapter 2 • Hardening the Operating System

Figure 2.4 Service Configuration



Though modern Linux distributions have enhanced the GUI to cover most of the administrative tasks, it's essential for good administrators to know how to perform the tasks in the absence of a GUI. Let us discuss about how to manually disable several vulnerable services.

Services to Disable

Linux, by nature, is more secure than most operating systems. Regardless, there are still uncertainties to every new Linux kernel that is released, and many security vulnerabilities that have not been discovered. Most Linux services are not vulnerable to these exploits. However, an administrator can reduce the amount of risk by removing unnecessary services. Red Hat Linux includes many services, so it makes sense that administrators customize the system to suit the company needs. Remember, you are removing risk when you remove unnecessary services.

The xinetd.conf File

Though newer and more sophisticated way managing network services are available in modern Linux distributions, /etc/xinetd.conf file still controls many Unix services, including

File Transfer Protocol (FTP) and Telnet. It determines what services are available to the system. The xinetd (like inetd in earlier versions) service is a “super server” listening for incoming network activity for a range of services. It determines the actual nature of the service being requested and launches the appropriate server. The primary reason for the design is to avoid having to start and run a large number of low-volume servers. Additionally, xinetd’s ability to launch services on demand means that only the needed number of servers is run.

The etc/xinetd.conf file directs requests for xinetd services to the /etc/xinetd.d directory. Each xinetd service has a configuration file in the xinetd.d directory. If a service is commented out in its specified configuration file, the service is unavailable. Because xinetd is so powerful, only the root should be able to configure its services.

The /etc/xinetd.d directory makes it simple to disable services that your system is not using. For example, you can disable the FTP and Telnet services by commenting out the FTP and Telnet entries in the respective file and restarting the service. If the service is commented out, it will not restart. The next section demonstrates how to disable the Telnet, FTP, and rlogin services.

Telnet and FTP

Most administrators find it convenient to log in to their Unix machines over a network for administration purposes. This allows the administrator to work remotely while maintaining network services. However, in a high-security environment, only physical access may be permitted for administering a server. In this case, you should disable the Telnet interactive login utility. Once disabled, no one can access the machine via Telnet.

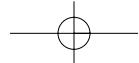
1. To disable Telnet, you must edit the /etc/xinetd.d/telnet file. Open the **Telnet file**, using vi or an editor of your choice.
2. Comment out the **service telnet** line by adding a number sign (#) before **service telnet**:

```
#service telnet
```

3. Write and quit the file.
4. Next, you must restart xinetd by entering:

```
/etc/rc.d/init.d/xinetd restart
Stopping xinetd: [OK]
Starting xinetd: [OK]
```

5. Attempt to log on to the system using Telnet. You should fail.
6. Note that commenting out the service line in the respective xinetd.d directory can disable many services.



28 Chapter 2 • Hardening the Operating System

7. Disable the FTP service using the same method (e.g., edit the /xinetd.d/wu-ftpd file by commenting out the `service ftp` line and restarting xinetd).
8. Attempt to access the system via FTP. You should be unable to log in to the server.

The Rlogin Service

The remote login (rlogin) service is enabled by default in the /etc/xinetd.d/rlogin file. Rlogin has security vulnerabilities because it can bypass the password prompt to access a system remotely. There are two services associated with rlogin: login and RSH (remote shell). To disable these services, open the **/xinetd.d/rlogin file** and comment out the **service login** line. Then, open the **/etc/xinetd.d/rsh file** and comment out the **service shell** line. Restart xinetd to ensure that your system is no longer offering these services.

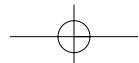
Locking Down Ports

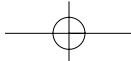
TCP/IP networks assign a port to each service, such as HTTP, Simple Mail Transfer Protocol (SMTP), and Post Office Protocol version 3 (POP3). This port is given a number, called a port number, used to link incoming data to the correct service. For example, if a client browser is requesting to view a server's Web page, the request will be directed to port 80 on the server. The Web service receives the request and sends the Web page to the client. Each service is assigned a port number, and each port number has a TCP and UDP port. For example, port 53 is used for the Domain Name System (DNS) and has a TCP port and a UDP port. TCP port 53 is used for zone transfers between DNS servers; UDP port 53 is used for common DNS queries—resolving domain names to IP addresses.

Well-Known and Registered Ports

There are two ranges of ports used for TCP/IP networks: well-known ports and registered ports. The well-known ports are the network services that have been assigned a specific port number (as defined by /etc/services). For example, SMTP is assigned port 25, and HTTP is assigned port 80. Servers listen on the network for requests at the well-known ports. Registered ports are temporary ports, usually used by clients, and will vary each time a service is used. Registered ports are also called ephemeral ports, because they last for only a brief time. The port is then abandoned and can be used by other services.

The port number ranges are classified, as shown in Table 2.1, according to Request for Comments (RFC) 1700. To access RFC 1700, go to <ftp://ftp.isi.edu/in-notes/rfc1700.txt>. Table 2.2 is a list of well-known TCP/UDP port numbers.



**Table 2.1** Port Number Ranges for Various Types

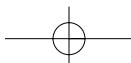
Type	Port Number Range
Well-known	1–1023
Registered	1024–65535

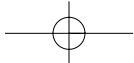
NOTE

Connections to ports number 1023 and below are assumed to run with root-level privileges. This means that untrusted services should never be configured with a port number below 1024.

Table 2.2 Commonly Used Well-Known TCP/UDP Port Numbers

Protocol	Port Number
FTP (Default data)	20
FTP (Connection dialog, control)	21
Telnet	23
SMTP	25
DNS	53
DHCP BOOTP Server	67
DHCP BOOTP Client	68
TFTP	69
Gopher	70
HTTP	80
POP3	110
NNTP	119
NetBIOS Session Service	139
Internet Message Access Protocol (IMAP), version 2	143





Determining Ports to Block

When determining which ports to block on your server, you must first determine which services you require. In most cases, block all ports that are not exclusively required by these services. This is tricky, because you can easily block yourself from services you need, especially services that use ephemeral ports, as explained earlier.

If your server is an exclusive e-mail server running SMTP and IMAP, you can block all TCP ports except ports 25 and 143, respectively. If your server is an exclusive HTTP server, you can block all ports except TCP port 80. In both cases, you can block all UDP ports since SMTP and IMAP all use TCP services exclusively.

However, if you want to use your server as an HTTP client (i.e., for accessing operating system updates) or as an e-mail client to a remote mail server, you will restrict the system by doing this. Clients require registered UDP ports for DNS, as well as registered TCP ports for establishing connections with Web servers.

If you open only the corresponding UDP ports 25, 80, and 143, DNS requests are blocked because DNS queries use UDP port 53, and DNS answers use a UDP registered port (e.g., the response stating that www.syngress.com=155.212.56.73). Even if you open port 53, a different registered port may be assigned each time for the answer. Attempting to allow access to a randomly assigned registered port is almost impossible and a waste of time. The same problem applies with TCP connections that require ephemeral ports.

Therefore, you should either open all TCP/UDP registered ports (so you can use your server as a client), or block them (except for the services you require) and access resources, such as operating system updates, another way. You can download the updates from another computer.

Blocking Ports

To block TCP/UDP services in Linux, you must disable the service that uses the specific port. You may use the GUI interface of firewall services offered by most of the Linux distributions. In Red Hat Enterprise Linux (RHEL) 5, **System | Administration | Security Level and Firewall** opens up the firewall configuration utility. Figure 2.5 shows the firewall is enabled and the selected services are trusted to run.

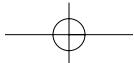
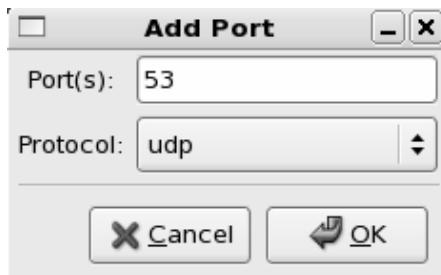


Figure 2.5 Security Level & Firewall Configuration

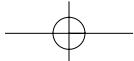
To allow a service to run, just check and enable the service and to block, uncheck the service. If you want to add any non-standard port or a custom port to be allowed by the firewall, then click on **Other ports** and add the protocol type (tcp or udp) and the port number, as shown in Figure 2.6.

Figure 2.6 Adding a Custom Port or Service

The following section discusses disabling ports assigned to stand-alone services.

Stand-Alone Services

To disable ports whose corresponding services are not included in the /etc/xinetd.d directory, you must kill the service's process and make sure that service does not automatically



32 Chapter 2 • Hardening the Operating System

restart upon reboot. These services are called stand-alone services. For example, port 111 is assigned a stand-alone portmapper service not required for most e-mail servers. The portmapper service, which is technically part of the Sun Remote Procedure Call (RPC) service, runs on server machines and assigns port numbers to RPC packets, such as NIS and NFS packets. Because these RPC services are not used by most e-mail services, port 111 is not necessary. To disable port 111, you must disable the portmapper service as follows:

1. To disable the portmapper service, identify the process identifier (PID) for portmap by entering:

```
ps aux | grep portmap
```

2. The second column lists the PID number. The last column lists the process using that PID. To stop the portmapper service, identify the PID number and enter:

```
kill -9 [PID NUMBER]
```

3. To make sure the service does not restart during reboot, enter:

```
Ntsysv (or use system-config-services gui utility from the terminal window)
```

4. Scroll down to the portmap service and uncheck the check box next to the service. Click **OK**. The portmap service will no longer restart at bootup.

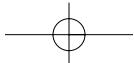
NOTE

Some ports, such as port 80, are not activated unless the service is installed. For example, if you have not installed Apache server, then port 80 is not used. There is no need to block the port because it is already disabled.

Hardening the System with Bastille

Bastille is an open source program that facilitates the hardening of a Linux system. It performs many of the tasks discussed in this chapter such as disabling services and ports that are not required for the system's job functions. The program also offers a wider range of additional services, from installing a firewall (ipchains/iptables) to implementing secure shell (SSH).

Bastille is powerful and can save administrators time from configuring each individual file and program throughout the operating system. Instead, the administrator answers a series of "Yes" and "No" questions through an interactive GUI. The program automatically implements the administrator's preferences based on the answers to the questions.



Bastille is written specifically to Red Hat Linux and Mandrake Linux, but can be easily modified to run on most Unix flavors. The specific Red Hat/Mandrake content has been generalized, and now the hard-code filenames are represented as variables. These variables are set automatically at runtime. Before you install Bastille on your system ensure your Linux version is supported by Bastille.

Bastille Functions

The following list highlights the security features offered by Bastille to secure your system. You will choose which feature you want to implement on your system during the question-and-answer wizard. For example, many servers do not need to provide firewall or Network Address Translation (NAT), so you may not need to configure ipchains/iptables. This is a partial list of features offered by Bastille and may vary as new versions of Bastille are released. More information about each of these features is explained in the program.

- **Apply restrictive permissions on administrator utilities** Allows only the root to read and execute common Administrator utilities such as ifconfig, linux-conf, ping, traceroute, and runlevel). It disables the SUID root status for these programs, so nonroot users cannot use them.
- **Disable r-protocols** The r-protocols allow users to log on to remote systems using IP-based authentication. IP-based authentication permits only specific IP addresses to remotely log on to a system. Because this authentication is based on the IP address, a hacker who has discovered an authorized IP address can create *spoofed* packets that appear to be from the authorized system.
- **Implement password aging** Default Red Hat Linux systems allow passwords to expire after 99,999 days. Because this is too long in a secure environment, Bastille offers to change the password expiration time to 180 days. These configurations are written to the /etc/login.defs file, as shown in Figure 2.7.
- **Disable CTRL-ALT-DELETE rebooting** This disallows rebooting the machine by this method.
- **Optimize TCP Wrappers** This choice modifies the inetc.conf (pre-Red Hat Linux 7 versions only) and /etc/hosts.allow files so that inetc must contact TCP Wrappers whenever it gets a request, instead of automatically running the requested service. TCP Wrappers will determine if the requesting IP address is allowed to run the particular service. If the request is not allowed, the request is denied and the attempt is logged. Although IP-based authentication can be vulnerable, this optimization adds a layer of security to the process. This is not recommended for most scenarios.

34 Chapter 2 • Hardening the Operating System

Figure 2.7 The /etc/login.defs File Configured for 180-Day Password Expiration

```

root@localhost:/etc
File Edit View Terminal Tabs Help
# *REQUIRED*
# Directory where mailboxes reside, _or_ name of file, relative to the
# home directory. If you _do_ define both, MAIL_DIR takes precedence.
# QMAIL_DIR is for Qmail
#
#QMAIL_DIR      Maildir
MAIL_DIR        /var/spool/mail
#MAIL_FILE      .mail

# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be used.
#      PASS_MIN_DAYS   Minimum number of days allowed between password changes.
#      PASS_MIN_LEN    Minimum acceptable password length.
#      PASS_WARN_AGE   Number of days warning given before a password expires.
#
PASS_MAX_DAYS   180
PASS_MIN_DAYS   0
PASS_MIN_LEN    5
PASS_WARN_AGE   7

#
# Min/max values for automatic uid selection in useradd
:[]
```

- **Add Authorized Use banners** These banners automatically appear whenever anyone logs on to the system. Authorized Use banners are helpful in prosecuting malicious hackers, and should be added to every system on your network that allows access to the network. An information bulletin from the U.S. Department of Energy's Computer Incident Advisory Capability can be found at <http://ciac.llnl.gov/ciac/bulletins/j-043.shtml>. The bulletin is titled “Creating Login Banners” and explains what is required within login banners for government computers. It also includes how to create banners and provides the text from the approved banner for Federal Government computer systems.
- **Limit system resource usage** If you limit system resource usage, you can reduce the chances of server failure from a DoS attack. If you choose to limit system resource usage in Bastille, the following changes will occur:
 - Individual file size is limited to 40MB.
 - Each individual user is limited to 150 processes.
 - The allowable core files number is configured to zero. Core files are used for system troubleshooting. They are large and exploitable if a hacker gains control of them: they can grow and consume your file system.

- **Restrict console access** Anyone with access to the console has special rights, such as CD-ROM mounting. Bastille can specify which user accounts are allowed to log on via the console.
- **Additional and remote logging** Two additional logs can be added to /var/log/:
 - **/var/log/kernel** (kernel messages)
 - **/var/log/syslog** (error and warning severity messages)You can also log to a remote logging host if one exists.
- **Process accounting setup** Allows you to log the commands of all users. It also records when the commands were executed. This log file is helpful in retracing a hacker's steps into your system, but the file can become large quickly. If the hacker has root access, the hacker can remove this accounting log.
- **Deactivate NFS and Samba** Allows you to disable NFS and Samba services. Samba provides a share file system. Unless firewall is configured to block the packets or administrator secures these services Bastille recommends to deactivate these services.
- **Harden Apache Web server** httpd should be deactivated if the service is not required. If you decide to use Apache, you can perform the steps shown in the "Hardening the Apache Web Server" sidebar in Bastille to run the service.

Bastille Versions

Bastille's current release 3.0.9 incorporates several important changes that make the program even more powerful and easy to use. The examples in this book use Bastille 3.0.9.

Implementing Bastille

Bastille is available for free download at www.sourceforge.net. The program is offered in tarball and rpm format. It must be installed by a root user in his or her root directory (a tarball is a collection of archived files that have been archived using the Unix tar program and have the .tar extension). Ensure perl/tk library is installed on your system as Bastille is a collection of Perl scripts.

The program automatically implements the administrator's preferences based on the answers to the questions, and saves them in the /root/Bastille/config file, as shown in Figure 2.8.

36 Chapter 2 • Hardening the Operating System

Figure 2.8 Bastille Configuration File

```

root@localhost.localdomain: /root/Bastille/undo/backup/etc
File Sessions Options Help
# Q: Would you like to run the ipchains script? [N]
IPChains_ip_intro="N"
# Q: Would you like to download and install the updated RPMs? [Default: No]
PatchDownload,patchownload="N"
# Q: Would you like to set more restrictive permissions on the administration utilities? [N]
FilePermissions_generalperms="N"
# Q: Would you like to disable SUID status for mount/umount?
FilePermissions_suidmount="N"
# Q: Would you like to disable SUID status for ping? [Y]
FilePermissions_suidping="Y"
# Q: Would you like to disable SUID status for dump and restore? [Y]
FilePermissions_suiddump="Y"
# Q: Would you like to disable SUID status for cardctl? [Y]
FilePermissions_suidcard="Y"
# Q: Would you like to disable SUID status for at? [Y]
FilePermissions_suidat="Y"
# Q: Would you like to disable SUID status for DOSEMU? [Y]
FilePermissions_suiddos="Y"
# Q: Would you like to disable SUID status for news server tools? [Y]
FilePermissions_suidnews="Y"
# Q: Would you like to disable SUID status for printing utilities? [N]
FilePermissions_suidprint="N"
# Q: Would you like to disable SUID status for the r-tools? [Y]
FilePermissions_suidrtool="Y"
# Q: Would you like to disable SUID status for usernetctl? [Y]
FilePermissions_suidusernetctl="Y"
# Q: Would you like to disable SUID status for traceroute? [N]
FilePermissions_suidtrace="N"
# Q: Would you like to set up a second UID 0 account? [N]
AccountSecurity_secondadmin="N"
# Q: May we take strong steps to disallow the dangerous r-protocols? [Y]
AccountSecurity_protectrhost="Y"
# Q: Would you like to enforce password aging? [Y]

```

Bastille allows the same configuration to be implemented on other systems. To do this, administrators need to install Bastille on that machine, copy the config file and the BackEnd file to the new system's `~/Bastille` directory, and then run the command:

```
#BastilleBackend
```

Damage & Defense...

Logging Your Configurations in Bastille

As with many security programs, Bastille is relatively simple to implement, but it's easy to lose track of the changes you implemented. This can be a problem if you are unable to perform a typical operation on the system, or are denied access to a command or service. Many times, it is because you locked down part of the system by mistake, or misjudged the impact of a particular Bastille choice.

It is always a good idea to create a hard-copy log of the options you select in Bastille, or any security configurations you implement on your system. Create a log with answers given to each question during the implementation and keep the hard copies in a safe place.

If your system goes down, you can access the hard copies and recreate your Bastille configurations. Of course, if your system became unusable due to Bastille, it

Continued

will help you determine what went wrong. This is especially helpful if you are unable to access the /root/Bastille/config file, which saves the administrator's preferences based on the answers to the Bastille questions.

Follow these steps to install and configure Bastille:

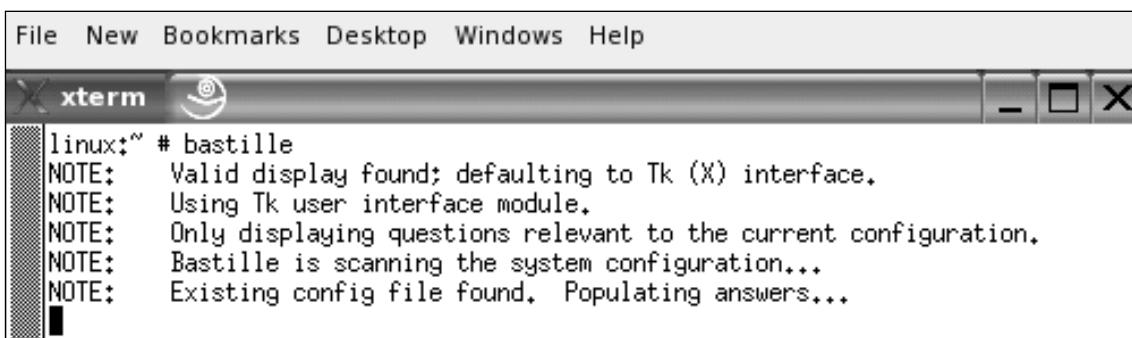
1. Log in as root.
2. Download the rpm file to your root directory. The filename will resemble:

Bastille-3.0.9-1.0.noarch.rpm

3. Double-click on the package icon (through GUI) or use command line:

rpm -i Bastille-3.0.9-1.0.noarch.rpm

Figure 2.9 Starting Bastille

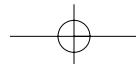


4. To run **Bastille GUI**, enter the following in the Bastille directory (Figure 2.9):

./bastille

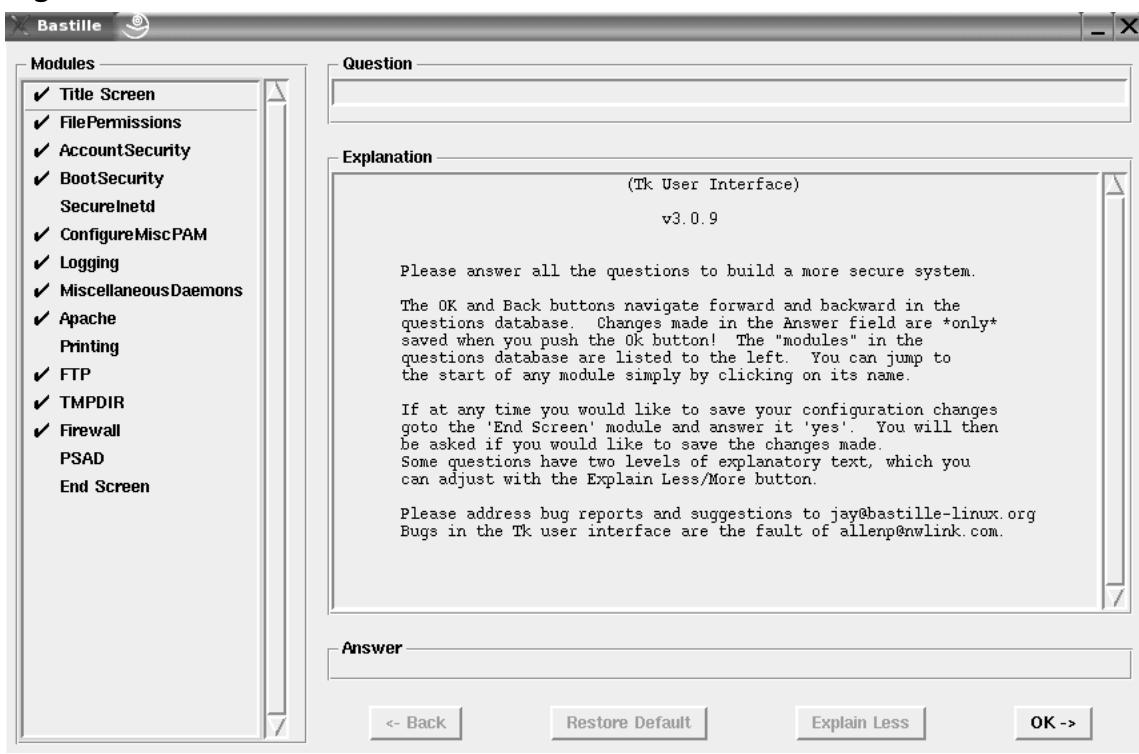
The opening Bastille screen appears, as shown in Figure 2.10.

5. All choices you implement in Bastille are logged to the /root/Bastille/config file. We strongly recommend that you make a backup of the config file before running Bastille and keep a manual log.



38 Chapter 2 • Hardening the Operating System

Figure 2.10 Bastille GUI



6. The opening screen appears, identifying how to navigate through the Bastille configuration process. Select **Next** to access the first configuration screen, as shown in Figure 2.11.
7. Table 2.5 leads you through the configuration process. You can use Bastille to secure a system based on your system's services and needs. Go through the explanation given below every question and understand the changes Bastille will perform based on your choice.

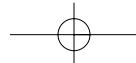
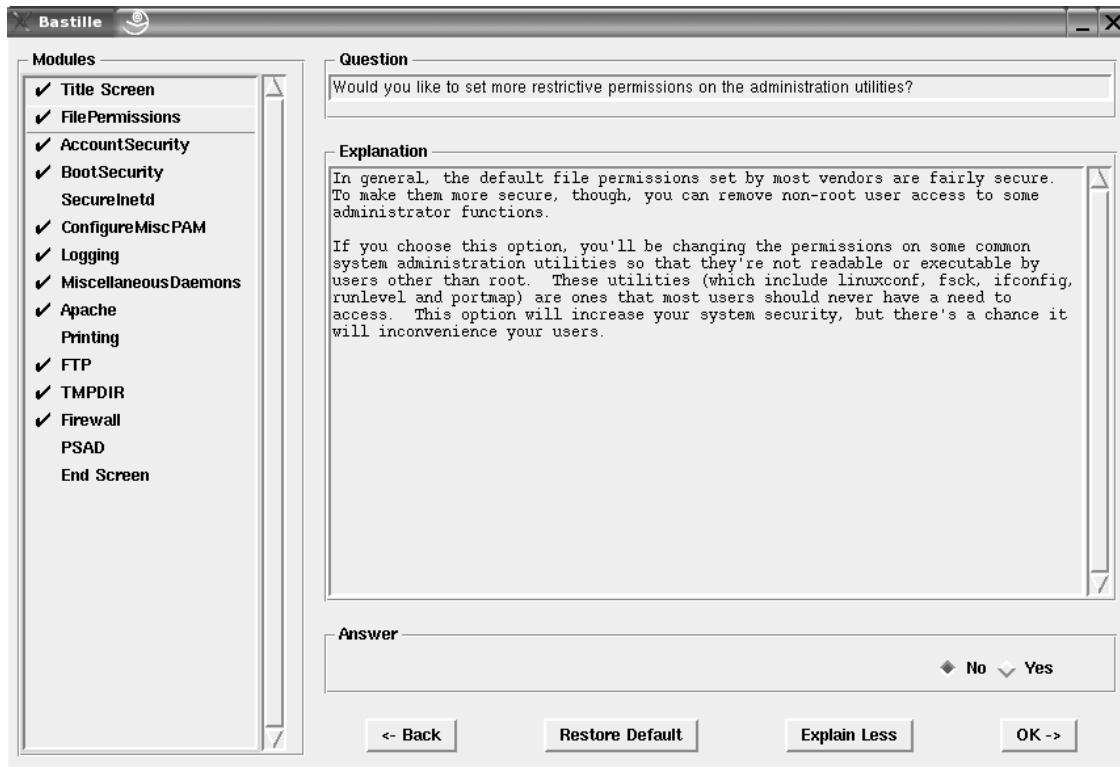
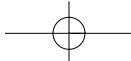


Figure 2.11 Bastille Linux Question-and-Answer Wizard**Table 2.5** Bastille Linux Questions

Questions	
1	Would you like to set more restrictive permissions on the administration utilities?
2	Would you like to disable SUID status for mount/umount?
3	Would you like to disable SUID status for ping?
4	Would you like to disable SUID status for at?
5	Would you like to disable the r-tools?
6	Should Bastille disable clear-text r-protocols that use IP-based authentication?
7	Would you like to enforce password aging?
8	Should we disallow root login on tty's 1-6?
9	Would you like to password-protect the GRUB prompt?
10	Would you like to disable CTRL-ALT-DELETE rebooting?

Continued

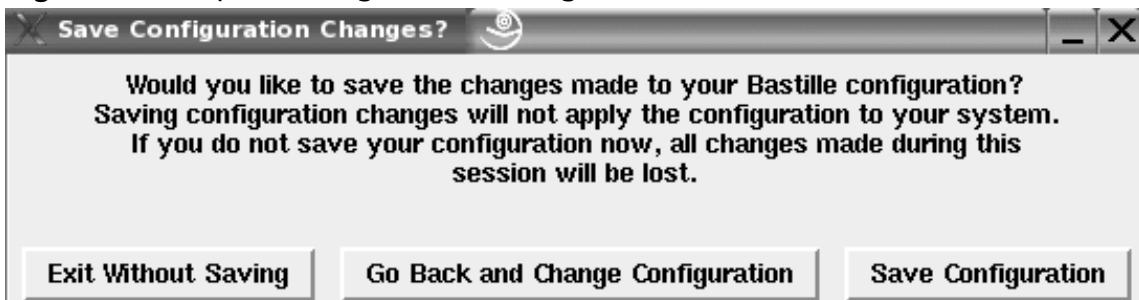


40 Chapter 2 • Hardening the Operating System

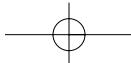
Table 2.5 continued Bastille Linux Questions

Questions	
11	Would you like to set a default-deny on TCP Wrappers and xinetd?
12	Would you like to display "Authorized Use" messages at log-in time?
13	Who is responsible for granting authorization to use this machine?
14	Would you like to put limits on system resource usage?
15	Should we restrict console access to a small group of user accounts?
16	Would you like to add additional logging?
17	Do you have a remote logging host?
18	Would you like to setup process accounting?
19	Would you like to disable acpid and/or apmd?
20	Would you like to deactivate NFS and Samba?
21	Would you like to deactivate the HP OfficeJet (hpoj) script on this machine?
22	Would you like to deactivate the ISDN script on this machine?
23	Would you like to disable printing?
24	Would you like to install TMPDIR/TMP scripts?
25	Would you like to run the packet filtering script?
26	Are you finished making changes to your Bastille configuration?

8. Bastille asks if you wish to implement these changes, as shown in Figure 2.12.

Figure 2.12 Implementing Bastille Changes

9. Select **Save Configuration** if you want to just save the configuration without applying changes. Select **Exit Without Saving** if you want to discard the changes. Select **Go Back and Change Configuration** if you want to apply the changes.



10. If you implemented password aging to 60 days, observe the changes Bastille made to the login.def file by entering:

```
cat /etc/login.defs | less
```

11. Press any key to display the next page. Press **q** to access the prompt.
12. You applied limits to system resources by limiting individual users to 150 processes, and configuring the allowable core files number to zero. Observe the changes Bastille made to the limits.conf file by entering:

```
cat /etc/security/limits.conf | less
```

13. Press any key to display the next page. Press **q** to access the prompt.

Undoing Bastille Changes

At the time of this writing, a reliable automatic undo feature did not exist in Bastille. To undo the changes, you can run through the configuration questions again and select different answers. There are two other options. There is a Perl script named Undo.pl in the Bastille directory that is designed to undo all changes except for RPM installations. There is also a backup directory located at /root/Bastille/undo/backup that contains all the original system files that Bastille modified. The backup directory structure is the same as the system's directory, so you can manually replace the files fairly easily.

You cannot undo your Bastille configurations by simply removing Bastille. If you do this, your changes will still be written to their specific files. If you want to remove the program and your settings, you must undo your changes, and then remove the Bastille directory.

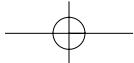
The following steps demonstrate three ways to undo the changes that you implemented in Table 2.5.

1. One method to undo Bastille configurations is to run through the configuration questions again and select different answers.
2. A second method to undo Bastille configurations is to run the automated Perl script that will undo the changes. The script is named Undo.pl, and is designed to undo all changes except for RPM installations. To run the Undo.pl script, access the **Bastille directory** and enter:

```
./Undo
```

3. A third method to undo Bastille configurations is to manually remove the changes. This can be done by replacing each file that was changed with the backup files in the Bastille directory. The backup directory is located at:

```
/root/Bastille/undo/backup
```



42 Chapter 2 • Hardening the Operating System

The backup files contain the original files before they were changed, so the original configurations are intact. Bastille makes a backup file of each file before the file is modified.

4. For example, to change password aging back to its default 99,999 days, replace the login.defs file with the backup file. Enter the following:

```
cd /root/Bastille/undo/backup/etc/login.defs  
cp logindefs /etc/login.defs  
cp: overwrite '/etc/login.defs'? y
```

The backup file replaces the current file, thus returning the password expiration configuration to its default setting.

As you can see, Bastille is a powerful security tool that helps you harden your system. It is relatively simple to use, and can save administrators a great deal of time because it automatically configures the required files for each selection. Administrators do not have to manually write to each file, or disable services individually. Bastille is recommended for any Unix system that offers services, whether it is a LAN or Internet server.

Controlling and Auditing Root Access with Sudo

Superuser Do (sudo) is an open source security tool that allows an administrator to give specific users or groups the ability to run certain commands as root or as another user. Sudo (current release is 1.6.9p5) is available for download from www.gratisoft.us/sudo/download.html. The program can also log commands and arguments entered by specified system users. The developers of sudo state that the basic philosophy of the program is to “give as few privileges as possible but still allow people to get their work done.” Sudo was first released to the public in the summer of 1986. The program is distributed freely under an ISC-style license. The Sudo Main Page is located at <http://www.gratisoft.us/sudo/>, as shown in Figure 2.13.

The program is a command-line tool that operates one command at a time. Table 2.6 lists several important features of sudo.

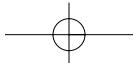
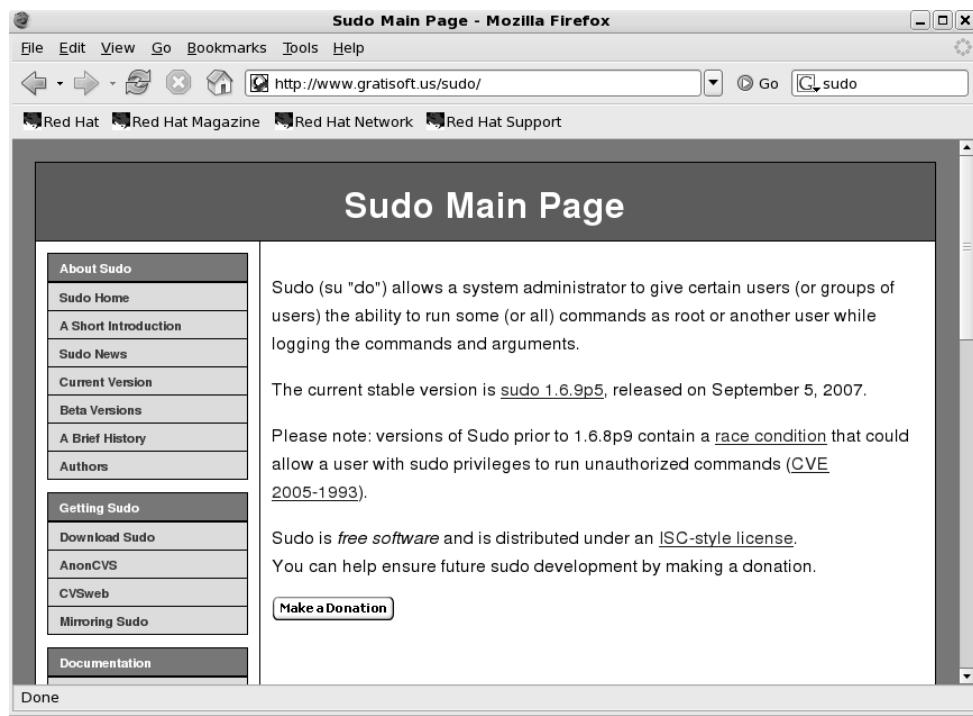
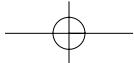


Figure 2.13 Sudo Home Page**Table 2.6** Sudo Features

Feature	Description
Command logging	Commands and arguments can be logged. Commands entered can be traced to the user. Ideal for system auditing.
Centralized logging of multiple systems	Sudo can be used with the system log daemon (syslog) to log all commands to a central host.
Command restrictions	Each user or group of users can be limited to what commands they are allowed to enter on the system.
Ticketing system	The ticketing system sets a time limit by creating a ticket when a user logs on to sudo. The ticket is valid for a configurable amount of time. Each new command refreshes the ticket for the predefined amount of time. The default time is five minutes.
Centralized administration of multiple systems	The sudo configurations are written to the /etc/sudoers file. This file can be used on multiple systems and allows administration from a central host. The file is designed to allow user privileges on a host-by-host basis.



44 Chapter 2 • Hardening the Operating System

Because sudo logs all commands run as root (or specified otherwise), many administrators use it instead of using the root shell. This allows them to log their own commands for troubleshooting and additional security.

The ticketing system is ideal because if the root user walks away from the system while still logged in (a very bad idea), another user cannot access the system simply because he or she has physical access to the keyboard.

After the ticket expires, users must log on to the system again. A shorter time is recommended, such as the default five minutes. The ticketing system also allows users to remove their ticket file.

System Requirements

To install and run sudo from the source distribution, you must have a system running Unix. Almost all versions of Unix support the sudo source distribution, including almost all flavors of POSIX, BSD, and SYSV. You must also install the C compiler and the make utility.

Sudo is known to run on the following Unix flavors: Auspex, SunOS, Solaris, ISC, RISCos, SCO, HP-UX, Ultrix, IRIX, NEXTSTEP, DEC Unix, AIX, ConvexOS, BSD/OS, OpenBSD, Linux, UnixWare, Pyramid, ATT, SINIX, ReliantUNIX, NCR, Unicos, DG/UX, Dynix/ptx, DC-Osx, HI-UX/MPP, SVR4, and NonStop-UX. It also runs on Mac OSX Server.

The Sudo Command

The **sudo** command allows a user to execute a command as a superuser or another user. All configurations for sudo are written to the /etc/sudoers file. The sudoers file specifies whether that command is allowed by that particular user.

In order to use sudo, the user must have already supplied a username and password. If a user attempts to run the command via sudo and that user is not in the sudoers file, an e-mail is automatically sent to the administrator, indicating that an unauthorized user is accessing the system.

Once a user logs in to sudo, a ticket is issued that is valid by default for five minutes. A user can update the ticket by issuing the **-v** flag, which will validate the ticket for another five minutes. The command is entered as follows:

```
sudo -v
```

If an unauthorized user runs the **-v** flag, an e-mail will not be sent to the administrator. The **-v** flag informs the unauthorized user that he or she is not a valid user. If the user enters command via sudo anyway, an e-mail will then be sent to the administrator.

Sudo logs login attempts, successful and unsuccessful, to the *syslog(3)* file by default. However, this can be changed during sudo configuration. Some of the command-line options listed in Table 2.7 are used by sudo.

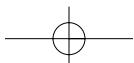


Table 2.7 Selected Sudo Command Options

Option	Option Name	Description
-V	Version	Prints version number and exits.
-l	List	Lists the commands that are allowed and denied by current user.
-h	Help	Prints usage message and exits.
-v	Validate	Updates the user's ticket for a configured amount of time (default is five minutes). If required, the user must re-enter the user password.
-k	Kill	Expires the user's ticket. Completing this option requires the user to re-enter the user password to update the ticket.
-K	Sure kill	Removes the user's ticket entirely. User must log in with username and password after running this option.
-u	User	Runs the specific command as the username specified. The user specified can be any user except root. If you want to enter a uid, enter #uid instead of the username.

Installing Sudo

Download Sudor tarball from www.gratisoft.us/sudo/download.html to any directory you choose. Sudo has been downloaded to the /root directory for this example. This exercise was performed on a Red Hat Enterprise Linux version 5.0.

1. Access the directory where you downloaded sudo, and decompress the tar file (your sudo version number will vary depending on the version of sudo that you downloaded) by entering:

```
tar -zxvf sudo-1.6.3p5.tar.gz
```

2. A directory will be created, such as sudo-1.6.3p5.
3. Access the sudo directory by entering:

```
cd sudo-1.6.3p5
```

4. To create a makefile and config.h file that will allow you to configure sudo, enter:

```
./configure
```



46 Chapter 2 • Hardening the Operating System

5. You can add options to the **./configure** command to customize your sudo installation. Simply append the options to your **./configure** command. The entire list of options is available in the /sudo/INSTALL file.
6. You can also edit makefile to change the default paths for installation, as well as the other configurations listed in /sudo/INSTALL file. If you require this change, open **makefile** in a text editor. For example, enter:

`vi Makefile`

7. Locate the “Where to install things...” section of makefile, as shown in Figure 2.14.

Figure 2.14 Sudo Makefile

```

root@localhost:~/sudo-1.6.9p5
File Edit View Terminal Tabs Help
# C preprocessor flags
CPPFLAGS = -I. -I$(srcdir)

# Usually -O and/or -g
CFLAGS = -O2

# Flags to pass to the link stage
LDFLAGS =
SUDO_LDFLAGS = $(LDFLAGS)

# Where to install things...
prefix = /usr/local
exec_prefix = ${prefix}
bindir = ${exec_prefix}/bin
sbindir = ${exec_prefix}/sbin
sysconfdir = /etc
mandir = ${datarootdir}/man
noexecdir = ${exec_prefix}/libexec
datarootdir = ${prefix}/share

# Directory in which to install sudo.
sudodir = $(bindir)

```

8. Change the default paths if necessary. For this example, we recommend that you use the default paths.
9. Quit the file. If you use the vi text editor, enter:

`:q`

10. (Optional) You can also change the default installation paths when you run the **./configure** command (you ran the **configure** command in a previous step). To do this, enter an option after the command. For example, by default the sudoers file is installed in the /etc directory. You can change this location by entering:

`./configure - -sysconfdir=DIR`

where DIR is the new installation directory.

11. To compile sudo, run the **make** command by entering:

`make`

12. (Optional) You will probably need GNU if you install sudo in a directory other than the source file directory. If you have errors during installation, read the TROUBLESHOOTING and PORTING files.
13. To install sudo, you must be the root user. Run the **make install** command to install the man pages, visudo, and a basic sudoers file by entering:

```
make install
```

NOTE

Any existing sudoers file will not be overwritten.

14. You have installed sudo. The next section explains how to configure it to suit your system's needs.

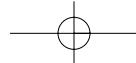
Configuring Sudo

To configure sudo, you must edit the %/sudo-1.6.9p5/sudoers file. The sudoers file defines which users are allowed to execute what commands. Only the root user is allowed to edit the file, and it must be edited with the **visudo** command. A sample.sudoers file is included in the sudo directory, and is shown in Figure 2.15.

Figure 2.15 Sample.Sudoers File

```
root@localhost:~/sudo-1.6.9p5
File Edit View Terminal Tabs Help
# sudoers file.
#
# This file MUST be edited with the 'visudo' command as root.
#
# See the sudoers man page for the details on how to write a sudoers file.
#
# Host alias specification
#
# User alias specification
#
# Cmnd alias specification
#
# Defaults specification
#
# Runas alias specification
#
# User privilege specification
root    ALL=(ALL) SETENV: ALL
#
# Uncomment to allow people in group wheel to run all commands
# and set environment variables.
# %wheel        ALL=(ALL) SETENV: ALL
```

The **visudo** command opens the sudoers file, by default, in the vi text editor. The **vi** commands are used to edit and write the file. You can change the default text editor used by visudo using the compile time option. Visudo uses the EDITOR environment variable. The **visudo** command performs the following tasks when editing the sudoers file:



48 Chapter 2 • Hardening the Operating System

- **Checks for parse errors** Visudo will not save any changes if a syntax error exists. It will state the line number of the error and prompt you for guidance. You will be offered a “What Now?” prompt and three choices: “e” to re-edit the file, “x” to exit without saving, and “Q” to quit and save changes. A syntax error result is shown in Figure 2.16.

Figure 2.16 Visudo Parse Error

>>> sudoers file: syntax error, line 76 <<<' followed by a 'What now?' prompt with a cursor."/>

```
root@localhost:~/sudo-1.6.9p5
File Edit View Terminal Tabs Help
[root@localhost sudo-1.6.9p5]# vi sudoers
[root@localhost sudo-1.6.9p5]# visudo
visudo: /etc/sudoers.tmp unchanged
[root@localhost sudo-1.6.9p5]# visudo
">>>> sudoers file: syntax error, line 76 <<<
What now? 
```

NOTE

If a syntax error exists in the sudoers file and you choose **Q** to quit and save the visudo changes, sudo will not run until the problem is corrected. You must run visudo again, fix the problem, and save the file again. It is recommended that you select **e** to attempt to fix the problem, or **x** to exit without saving (if you are not sure of what went wrong).

- **Prevents multiple edits to the file simultaneously** If you attempt to run visudo while the sudoers file is being edited, you will receive an error message informing you to try again at a later time

The sudoers file consists of two different types of entries, *user specifications* and *aliases*. The following examples show you how to use user specifications, which define which user is allowed to run what commands. Aliases are basically variables.

The sudoers file contains a root entry. The default sudoers file is shown in Figure 2.17. The user privilege specification is listed as

```
root    ALL=(ALL) ALL
```

This configuration allows the root user to issue all commands.

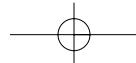
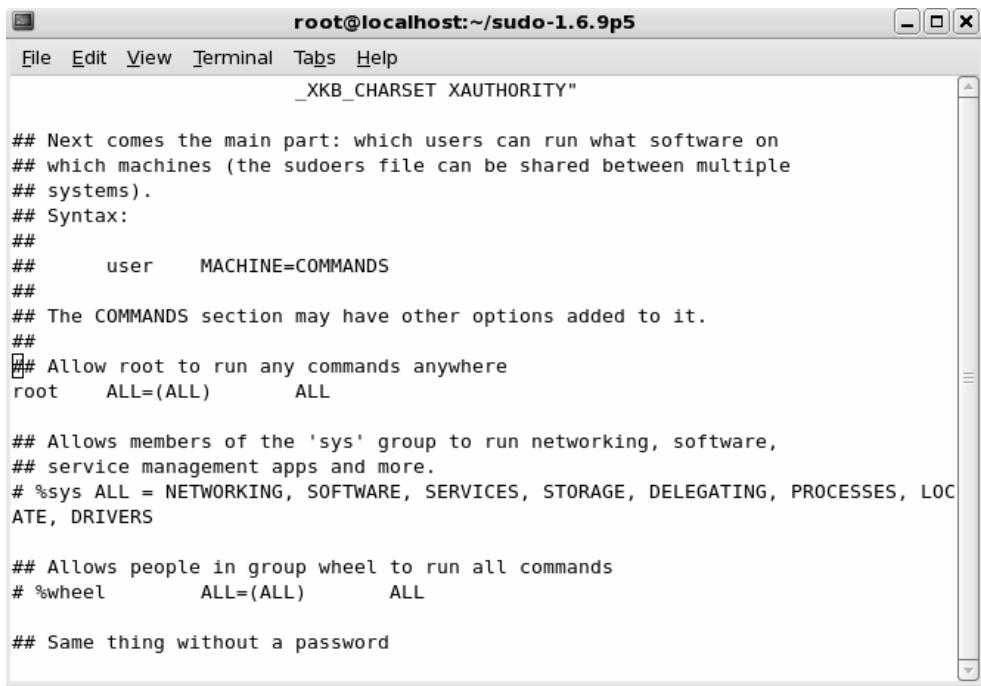


Figure 2.17 Default Sudoers File Allowing the Root User Access to All Commands


```

root@localhost:~/sudo-1.6.9p5
File Edit View Terminal Tabs Help
_XKB_CHARSET XAUTHORITY"

## Next comes the main part: which users can run what software on
## which machines (the sudoers file can be shared between multiple
## systems).
## Syntax:
##
##      user      MACHINE=COMMANDS
##
## The COMMANDS section may have other options added to it.
##
## Allow root to run any commands anywhere
root    ALL=(ALL)      ALL

## Allows members of the 'sys' group to run networking, software,
## service management apps and more.
# %sys  ALL = NETWORKING, SOFTWARE, SERVICES, STORAGE, DELEGATING, PROCESSES, LOC
ATE, DRIVERS

## Allows people in group wheel to run all commands
# %wheel        ALL=(ALL)      ALL

## Same thing without a password

```

To allow other users to run commands as root, you must enter those users in the sudoers file. You must also list the host on which they are allowed to run the commands. Last, you must list the specific commands that those users are allowed to run as root. In the following steps, you will create user *bob* and allow him to run several commands as root using sudo on your system.

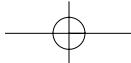
1. Open the sudoers file by entering:

`visudo`

2. The sudoers file opens in vi. Locate the “User privilege specification” section. After the root entry, enter the following (press **i** to insert text):

`bob your-hostname = /sbin/ifconfig, /bin/kill, /bin/ls`

3. This line allows user bob to run the **ifconfig**, **kill** and **ls** commands as root.



NOTE

By default, all commands you list in sudoers will run as root unless you specify otherwise. For example, bob could run commands as user *bugman* if desired. You would enter:

```
bob your-hostname = (bugman) /sbin/ifconfig
```

In this case, the **ifconfig** command will run as user bugman. You can allow bob to enter commands as several different users.

```
bob your-hostname = (bugman) /sbin/ifconfig, (root) /bin/kill, /bin/ls
```

The **kill** and **ls** commands will run as root, while the **ifconfig** command runs as bugman. At the command line, bob will enter:

```
sudo -u bugman /sbin/ifconfig
```

3. Press **ESC** to write and quit the file. Then, enter:

```
:wq
```

This command writes and quits the file using vi.

4. Now you must create user bob. Enter:

```
useradd bob
```

5. Create a password for user bob by entering:

```
passwd bob
Changing password for user bob
New UNIX password:
Retype new UNIX password:
passwd: all authentication tokens updated successfully
```

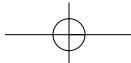
Running Sudo

You have configured sudo to allow user bob root privileges for the **ifconfig**, **kill**, and **ls** commands. When bob wants to run these commands, he must first enter the **sudo** command, and then his password.

1. Log on as user bob.
2. To find out what commands bob has root access to, enter the following:

```
sudo -l
```

3. If this is your first time running sudo as user bob, a warning will display:



We trust you have received the usual lecture from the local System Administrator. It usually boils down to these three things:

- #1) Respect the privacy of others.
- #2) Think before you type
- #3) With great power comes great responsibility

4. A password prompt appears. Do *not* enter the root password. Enter bob's password:

5. The commands that bob is allowed to run on this host are listed, as shown in Figure 2.18.

Figure 2.18 Commands That User Bob Can Run as Root

```
bob@localhost:~$ sudo -l
We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:
#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

Password:
User bob may run the following commands on this host:
(root) /sbin/ifconfig
(root) /bin/kill
(root) /bin/ls
[bob@localhost ~]$
```

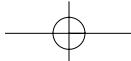
6. To test your sudo configurations, run an ifconfig option that requires root permission without using sudo. Enter:

/sbin/ifconfig eth0 down

Permission is denied because bob is not allowed to deactivate the system's interface.

7. To deactivate the interface, bob must use sudo. Enter:

sudo /sbin/ifconfig eth0 down



52 Chapter 2 • Hardening the Operating System

You will be successful. Please note that sudo will ask for the bob's password if bob's ticket has expired (the default is five minutes). If you run this command within five minutes from the last, you will not be prompted for a password.

8. Reactivate the interface. Enter:

```
sudo /sbin/ifconfig eth0 up
```

9. Next, restart one of the httpd processes using the **kill** command by entering:

```
ps aux | grep httpd
```

10. Choose an Apache PID from the list that appears (If Apache is not installed, select a different service process to restart). Enter:

```
kill -HUP [PID NUMBER]
```

11. You are not allowed to restart the httpd process because you are not root. You will receive the following result:

```
bash: kill: (PID NUMBER) - Not owner
```

12. Instead, use sudo to run the command as root by entering:

```
sudo kill -HUP (PID NUMBER)
```

You should be successful.

13. Next, you will list the root user directory as user bob using the **ls** command. Enter:

```
ls /root
```

Permission is denied because you are not root.

14. Again, use sudo to run the command as root:

```
sudo ls /root
```

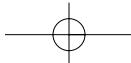
Permission is granted and the root user's directory is displayed.

15. To expire bob's timestamp, enter the command **sudo -k**. Bob will have to enter a password the next time he uses sudo.

No Password

In some situations, entering a password each time sudo is run is redundant because the user has already logged on to the system. Sudo offers a way around this monotonous task by using the NOPASSWD tag in the sudoers file.

1. To remove the password requirement in the sudoers file, log on as *root* and enter:



```
visudo
```

2. The sudoers file opens in vi. Modify bob's user privilege specification to match the following (press **i** to insert text):

```
bob    your-hostname = NOPASSWD: /sbin/ifconfig, /bin/kill, /bin/ls
```

3. Press **ESC**. Enter **:wq** to write and quit the file.
4. Log on as bob. Deactivate the interface using sudo:

```
sudo /sbin/ifconfig eth0 down
```

You will not be prompted for your password and the command will run as root.

5. Reactivate the interface. Enter:

```
sudo /sbin/ifconfig eth0 up
```

Sudo Logging

As mentioned previously, sudo logs which users run what commands. Logging does not occur automatically. You must set up sudo and syslogd to log commands. This involves two steps. First, you must create a sudo logfile in /var/log. Second, you must configure syslog.conf to log sudo commands. The following steps show you how to configure sudo logging.

1. Log on as root. Create a sudo log file in /var/log/. Enter:

```
touch /var/log/sudo
```

2. Next, you must add a line in the syslog.conf file to direct logging to your sudo logging file. Open syslog.conf by entering the following:

```
vi /etc/syslog.conf
```

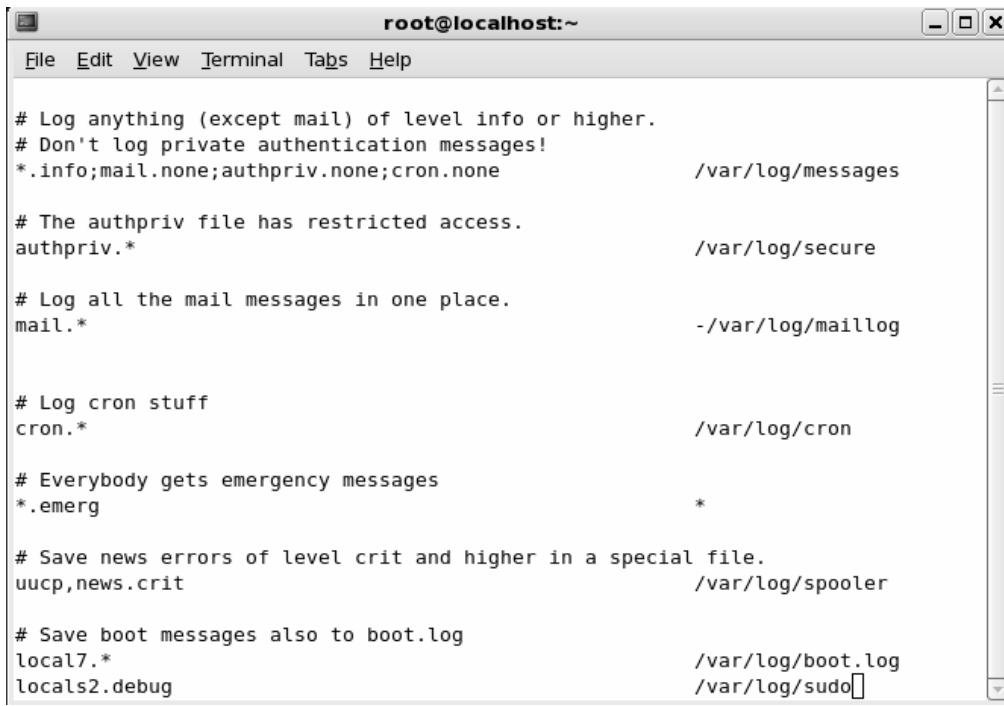
3. Enter the following line at the end of the syslog.conf file (press **i** to insert text). The white space must be created using TAB, not the SPACE BAR.

```
local2.debug          /var/log/sudo
```

4. This syslog.conf entry logs all successful and unsuccessful sudo commands to the /var/log/sudo file. You can also log to a network host by indicating the network host instead of a local directory. The syslog.conf file is shown in Figure 2.19.

54 Chapter 2 • Hardening the Operating System

Figure 2.19 Editing the Syslog.conf file for Sudo Logging



```
# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none          /var/log/messages

# The authpriv file has restricted access.
authpriv.*                                         /var/log/secure

# Log all the mail messages in one place.
mail.*                                            -/var/log/maillog

# Log cron stuff
cron.*                                           /var/log/cron

# Everybody gets emergency messages
*.emerg                                         *

# Save news errors of level crit and higher in a special file.
uucp,news.crit                                    /var/log/spooler

# Save boot messages also to boot.log
local7.*                                         /var/log/boot.log
locals2.debug                                     /var/log/sudo
```

5. Press **ESC** to write and quit the file. Then, enter:

```
:wq
```

6. Since you have modified the syslog.conf file, you need to restart syslogd. To send a HUP signal to syslogd, you must first know the syslogd process identifier (PID). To identify the syslog PID, enter:

```
ps aux | grep syslogd
```

7. The second column lists the PID number. The last column lists the process using that PID. To restart syslogd, identify the PID number and enter:

```
kill -HUP [PID NUMBER]
```

8. First, you will generate log entries for user bob. Log on as user bob.
9. Enter the following **ifconfig** commands while logged on as user bob:

```
sudo -l
sudo /sbin/ifconfig eth0 down
sudo /sbin/ifconfig eth0 up
```

10. Restart one of the httpd processes (or another process) using the **kill** command by entering:

```
ps aux | grep httpd
```

11. Choose an Apache (httpd) PID from the list that appears. Enter:

```
sudo kill -HUP [PID NUMBER]
```

12. Now list the root user directory as user bob. Enter:

```
sudo ls /root
```

13. Log on as root and view the sudo log file. All the sudo commands that bob entered are listed, as shown in Figure 2.20.

Figure 2.20 Sudo Log File Displaying User Bob's Commands

```
root@localhost:/var/log
File Edit View Terminal Tabs Help
Sep 14 14:24:38 localhost sudo:      bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=list
Sep 14 14:24:48 localhost sudo:      bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=/sbin/ifconfig eth0 down
Sep 14 14:24:51 localhost sudo:      bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=/sbin/ifconfig eth0 up
Sep 14 14:25:00 localhost sudo:      bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=/bin/ls /root
~
```

14. You can log any root commands by simply typing **sudo** before each command. For example, make sure that you are logged on as root and enter the following commands (or any commands you choose):

```
sudo useradd susan
sudo passwd susan
sudo vi /hosts
```

15. Access and view the sudo log file by entering:

```
sudo cat /var/log/sudo
```

All root user entries are logged, including the **cat** command you just entered, as shown in Figure 2.21.

56 Chapter 2 • Hardening the Operating System

Figure 2.21 Sudo Log File Displaying Root User Commands

```

root@localhost:~ Sep 14 14:24:38 localhost sudo: bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=list
root@localhost:~ Sep 14 14:24:48 localhost sudo: bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=/sbin/ifconfig eth0 down
root@localhost:~ Sep 14 14:24:51 localhost sudo: bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=/sbin/ifconfig eth0 up
root@localhost:~ Sep 14 14:25:00 localhost sudo: bob : TTY=pts/1 ; PWD=/home/bob ; USER=root ; COMMAND=/bin/ls /root
root@localhost:~ Sep 14 14:42:24 localhost sudo: root : TTY=pts/1 ; PWD=/root ; USER=root ; COMMAND=/usr/sbin/useradd susan
root@localhost:~ Sep 14 14:42:29 localhost sudo: root : TTY=pts/1 ; PWD=/root ; USER=root ; COMMAND=/usr/bin/passwd susan
root@localhost:~ Sep 14 14:42:53 localhost sudo: root : TTY=pts/1 ; PWD=/root ; USER=root ; COMMAND=/bin/vi /hosts
root@localhost:~ Sep 14 14:43:00 localhost sudo: root : TTY=pts/1 ; PWD=/root ; USER=root ; COMMAND=/bin/vi /etc/hosts
root@localhost:~ Sep 14 14:43:15 localhost sudo: root : TTY=pts/1 ; PWD=/root ; USER=root ; COMMAND=/bin/cat /var/log/sudo
~
```

As you can see, sudo is extremely helpful for controlling and auditing root access. It allows a system administrator to distribute root system tasks without distributing the root password. An administrator can control what root access is needed for each user, and can customize system access based on those needs.

Sudo is used almost entirely by system administrators, and is a great way to train new system administrators. New administrators can be given a new account with only selected root privileges. The master administrator can then review the work of the administrator in training.

This section discussed one of the many ways to use sudo; it focused primarily on user specifications in the sudoers file. For more information on extending sudo, such as using aliases, please consult the sudo man file available at www.gratisoft.us/sudo/man/sudo.html.

Managing Your Log Files

Another aspect of system security is managing your log files. By default, Linux offer modest logging so that administrators can see who and what has accessed their system. More logging is available (both more detail and logging on more services), but Linux keeps it brief so that you don't fill your hard disk with log information. This section briefly discusses helpful commands and programs that provide access to system logs.

Linux offers commands that allow administrators to access useful log files. Two commands of interest are **last** and **lastlog**. The message file also offers useful data for determining possible security breaches on your system.

The **last** command displays data such as who is logged on to the system, who recently logged on, and when the system has rebooted. For example, you may receive data such as the following:

```

root    pts/1  :0.0   Tue Sep 18 01:13      still logged in
root    pts/1  :0.0   Tue Sep 18 01:02 - 01:06  (00:03)
root    :0       Tue Sep 18 01:00      still logged in
```

The **lastlog** command displays the users and services that have accounts on your machine. It lists the last time each account logged in to the system, or if the account has ever logged in. Each service in Linux is given an account. This is very helpful because if a service logged in without your knowledge, a hacker may be responsible. This would indicate that the hacker controls your system and is currently exploiting it. It could also mean that another administrator started the service without telling you.

The messages file is a log file that displays a list of recent activity on the system. For example, it lists if a password was changed and who changed it. It identifies when a user session opens and closes. It also lists the time and date each event took place. It can be viewed by entering the following command:

```
tail /var/log/messages
```

If you prefer a GUI to view your log files, a program called *SWATCH* (not installed by default) allows an instant and real-time display for various log files. It can view any log files you specify and is discussed in the next section.

The Linux logs should be checked frequently to determine if any security violations have occurred on your system. Logs do not offer solutions, so you must analyze the data and decide how to counteract the attack.

Using Logging Enhancers

Logging enhancers are tools that simplify logging by allowing logging information to be filtered and often displaying logs in simplified formats. Many open source logging programs exist to make system administration much easier. Viewing text-based files with hundreds or thousands of entries can be burdensome, especially if you are only looking for one specific error entry. Logging enhancers can make logging a much more user-friendly experience, and greatly expand and customize the information you need to log.

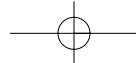
The next sections explain three popular logging services used by administrators: *SWATCH*, *scanlogd*, and the next generation of *syslogd* (*syslogd-*ng**).

SWATCH

Simple WATCHer or Simple WATCHdog (*SWATCH*) is an open source package that allows administrators to efficiently monitor system activity. It can monitor events on a system, or a large number of systems, by monitoring system logs for specified events.

SWATCH'S main function is to monitor messages actively as they are written to a log files through the Unix *syslog* utility. *SWATCH* requires Perl 5 to function.

SWATCH is efficient because it allows administrators to modify the *SWATCH* configuration file (*/etc/swatchrc*) to filter logging entries and respond to certain events. For example, *SWATCH* can monitor the system for bad login attempts, and e-mail the administrator whenever this failed authentication event occurs. It can monitor and alter when



58 Chapter 2 • Hardening the Operating System

system halts and reboots occur, when a user upgrades to root using the **su** command, when the file system is full, and when someone is sniffing the system. It can monitor anything desired from the log files.

To learn about SWATCH and download the program, you need to visit the SWATCH home page at <http://swatch.sourceforge.net>. The SWATCH home page is shown in Figure 2.22.

Figure 2.22 SWATCH Home Page

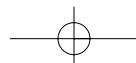


NOTE

At the time of this writing, version 3.2.2 of SWATCH was available for download.

SWATCH uses two required fields: *pattern(s)* and *action(s)*.

- **Patterns** The SWATCH configuration file looks for *patterns* in logging entries. For example, bad login attempts display a “Failed Authentication” error.



- **Actions** Whenever a pattern is discovered, SWATCH seeks an *action*, such as e-mailing the administrator of the failed login attempt.

Two optional fields are used to further customize the configuration file:

- **Throttle** Throttle determines the amount of time that SWATCH will ignore repeated logged entries before listing the entry again. This saves administrators' time from viewing 300 identical "Failed Authentication" errors. However, a secure system should limit the number of login attempts. The throttle entry is defined as HH:MM:SS, where H represents hours, M represents minutes, and S represents seconds.
- **Timestamp** Timestamp defines the length and location of the timestamp. The timestamp entry is defined as start:length.

The following are two examples from the SWATCH configuration file. SWATCH will actively watch for these messages as they are written to their respective log files through the syslog utility. The first example monitors logging for failed login attempts and e-mails root when a failed login attempt occurs.

```
#Failed login attempts
watchfor      /failed/
              echo bold
              mail addresses=root,subject=Failed Authentication
```

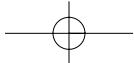
The second example monitors your log files and e-mails root when a user sued to gain root access.

```
#Users sued to gain root access
watchfor      /su:/
              echo bold
              mail addresses=root,subject=User sued to root
```

SWATCH filters logging files so that administrators only receive the information they require. It saves a lot of time and trouble once configured and is recommended for system administrators who are overwhelmed by log files (and perhaps do not use them for that reason). To download an RPM version of SWATCH, visit www.rpmfind.net.

Scanlogd

Scanlogd is an open source program that detects and logs TCP-port scanning on a system. For example, it can detect nmap scans. Nmap is a program used by hackers to create a "map" of your network. It is often the first step a hacker takes once he or she has access to your network to determine which system to hack. Nmap lists the systems and the services



60 Chapter 2 • Hardening the Operating System

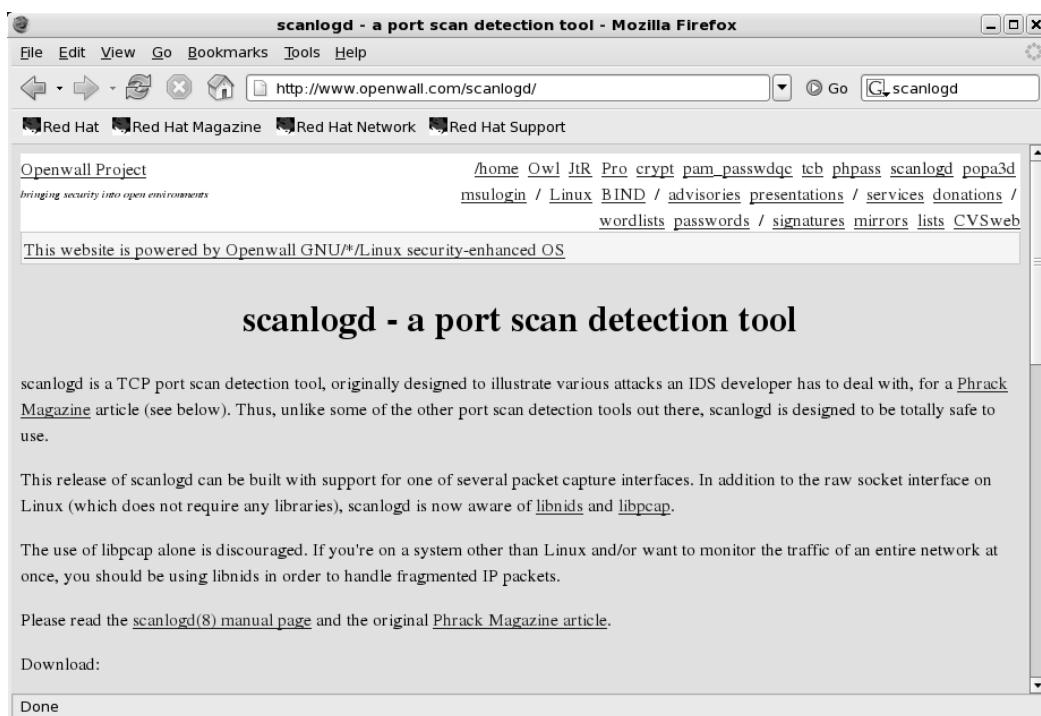
on the network. Scanlogd can alert an administrator when the network is being mapped, but it cannot stop the intrusion.

SECURITY ALERT!

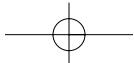
Scanlogd was originally designed to illustrate attacks, not to fix them. Therefore, even though it is safe to run on your system, it does not prevent hacking attacks. You must read the system log to discover what happened to your system, and then determine the appropriate solution.

Scanlogd writes one line per scan using the syslog(3) mechanism. It also logs when a source address sends many packets to several different ports in a short amount of time. You can learn about scanlogd and download the program at www.openwall.com/scanlogd. The scanlogd home page is shown in Figure 2.23.

Figure 2.23 Scanlogd Home Page



Because scanlogd is only meant to detect scans, it is totally safe to run on your system. It must have access to raw IP packets to function, and can capture packets coming in and out



of the system interface, or across the network to which the system is attached. In addition, scanlogd supports the raw socket interface on libnids, libpcap, and Linux.

Syslogd-ng

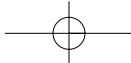
*Syslogd-*ng** is a logging daemon that is the replacement for the traditional syslogd. The “*ng*” is an acronym for “next generation.” The original syslogd was the general Unix logging daemon that handled requests for syslog services, but was difficult to configure. Syslogd-*ng* is easier to configure and offers additional logging features, such as more configurations. For example, syslogd-*ng* allows administrators to filter messages based on priority, as well as the content of the messages. You can also forward logs on TCP, sort logs to different destinations, and create a direct log stream to various hosts. The syslog-*ng* home page is shown in Figure 2.245 and is located at www.balabit.com/network-security/syslog-ng/.

Figure 2.24 Syslog-*ng* Home Page



The basic problem with system logs is that they contain a lot of unimportant information. This information is often called *noise*. Many events are lost because they are buried in the noise. Syslog made it difficult to choose only the important messages.

The reason this occurs is that messages are sent to different destinations depending on the assigned facility/priority pair. These destinations are very broad, and include general



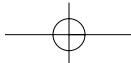
62 Chapter 2 • Hardening the Operating System

facilities such as mail, news, auth, and so forth, and priorities ranging from alert to debug. Many programs use the facilities; so many unneeded messages are written to their logs. In many cases, the message and the facility are not even related. Syslogd-*ng* filters messages based on message content in addition to the facility/priority pair. Using this method, only the messages that are needed are logged.

Figure 2.25 Syslog-*ng* Platforms

Name	Type	Size	Date
Parent directory	Directory		
aix-5.2_syslog-ng-1.6	Directory		03/07/2007
aix-5.2_syslog-ng-2.0	Directory		07/22/2007
aix-5.2_syslog-ng-2.1	Directory		03/07/2007
debian-etch_syslog-ng-2.0	Directory		07/22/2007
debian-sarge_syslog-ng-1.6	Directory		02/02/2007
debian-sarge_syslog-ng-2.0	Directory		07/22/2007
rhel-3_syslog-ng-1.6	Directory		02/02/2007
rhel-3_syslog-ng-2.0	Directory		07/22/2007
rhel-4_syslog-ng-1.6	Directory		11/10/2006
rhel-4_syslog-ng-2.0	Directory		07/22/2007
rhel-5_syslog-ng-2.0	Directory		07/27/2007
solaris-8_syslog-ng-1.6	Directory		11/10/2006
solaris-8_syslog-ng-2.0	Directory		07/23/2007
solaris-9_syslog-ng-1.6	Directory		11/10/2006
solaris-9_syslog-ng-2.0	Directory		07/23/2007
suse-10.0_syslog-ng-1.6	Directory		02/02/2007
suse-10.0_syslog-ng-2.0	Directory		07/22/2007
suse-10.1_syslog-ng-1.6	Directory		02/02/2007
suse-10.1_syslog-ng-2.0	Directory		07/22/2007

Syslogd-*ng* is available on AIX, Debian, Red Hat Enterprise Linux 3, 4, 5, Sun Solaris 8, 9, and Suse Linux 10. At the time of this writing, the latest stable version was 2.0. You can learn more about syslog-*ng* and download manuals from www.balabit.com/support/documentation/. The product is available in Open Source and Premium editions. You may also request for an evaluation before deploying it in your production network. Additional support packages (such as libdi8) and database specific binaries may be required. Figure 2-26 shows supported versions available for download from Balabit.



Security Enhanced Linux

Security Enhanced Linux or SELinux is a research project initiative from NSA (National Security Agency). Recommendations of this project are incorporated in various Linux distributions. The project focuses on utilizing mandatory access control (MAC) architecture incorporated into the kernel subsystems rather than working on the security vulnerabilities on the operating system itself. More information on SELinux, whitepapers, presentation and downloads are available at www.nsa.gov/selinux/index.cfm.

Red Hat Linux has incorporated SELinux in its releases. If you have not enabled SELinux during the installation stage, the same can be enabled from

Let's look at the components of SELinux specific to Red Hat Enterprise Linux (RHEL) in version 5. Before that you need to enable SELinux.

1. To enable SELinux click on **System | Administration | Security Level and Firewall**
2. Click on the second tab **SELinux**. You will find three options Enforcing (policy enforced state), Permissive (warnings only, no policy enforcement) and Disabled (SELinux policy fully disabled state). Once you have enabled SELinux, the system may re-label the file system for SELinux. This requires a restart.
3. To manage SELinux click on **System | Administration | SELinux Management Tool**. This tool has status, boolean, file labeling, user mapping, SELinux user, translation, network port and policy module configuration options. First screen that opens is the status screen. This includes the current status for system default enforcing mode, current enforcing mode, and system default policy type. In this case it's permissive, permissive and targeted as shown in the Figure 2.26. When you enabled SELinux the file system is already relabeled. You do not have to select the check box 'relabel on next reboot' once again.
4. Next is the **Boolean** screen. In this screen, you choose individual services and enable or disable (yes or no) specific policy settings. This is where you configure run-time Booleans. For example, for HTTPD Service you may enable allow HTTPD cgi support, allow HTTPD to read home directories and allow HTTPD to support built-in scripting as shown in the Figure 2.27.

64 Chapter 2 • Hardening the Operating System

Figure 2.26 SELinux Management Tool

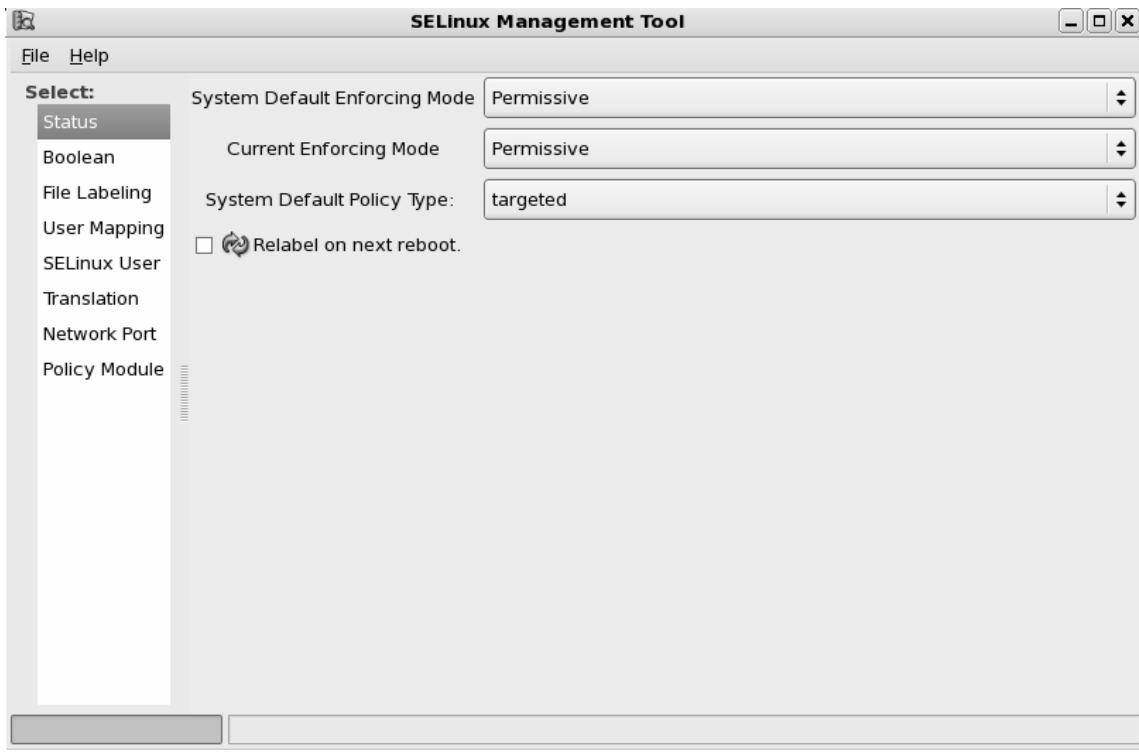
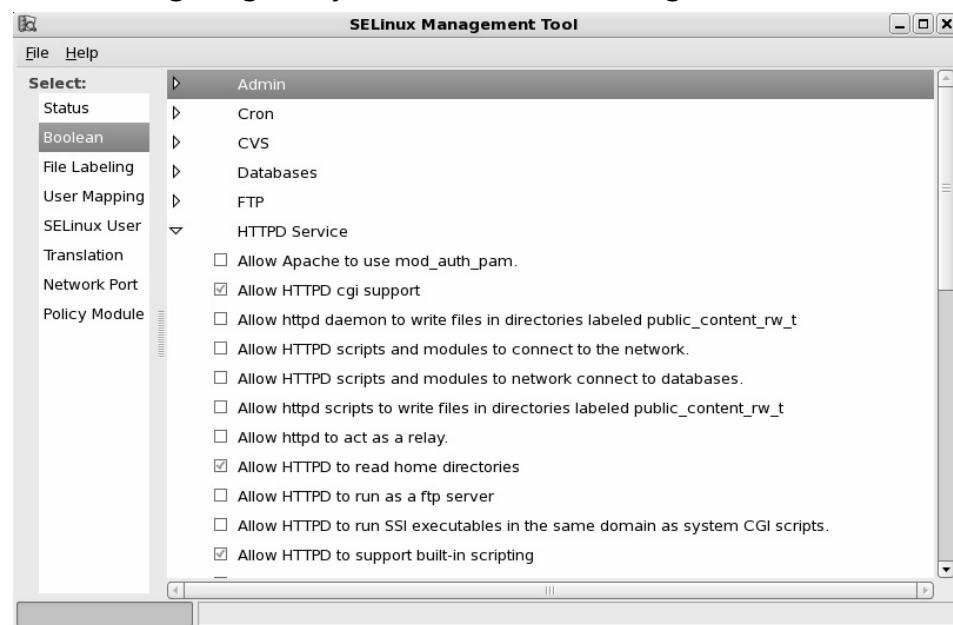
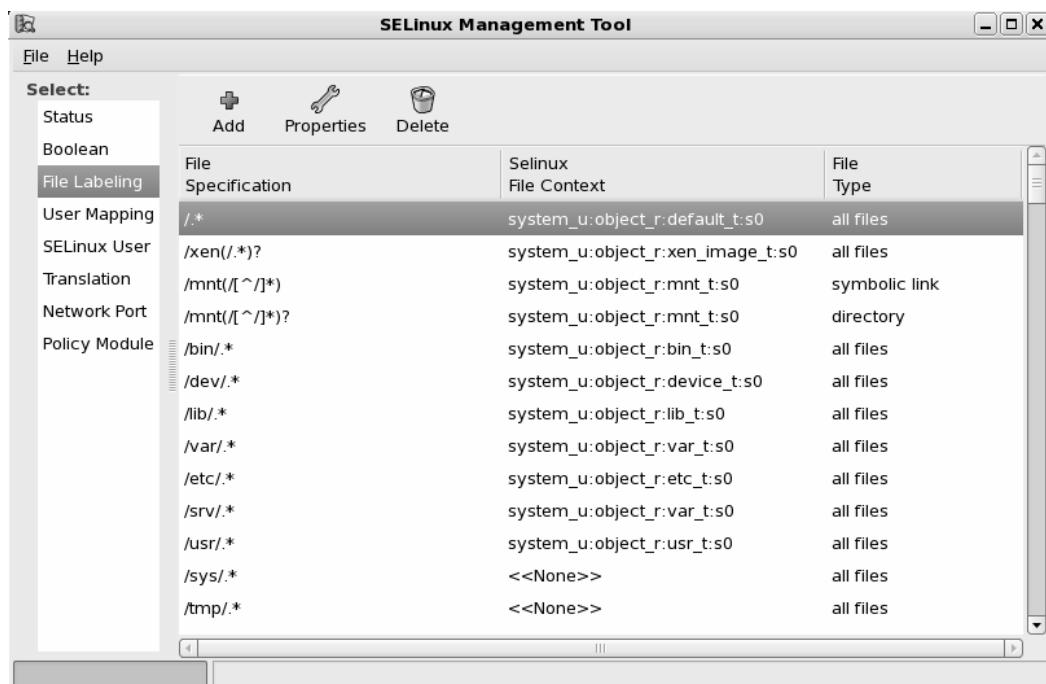


Figure 2.27 Configuring Policy for HTTPD Service using Boolean Tab

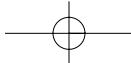


5. You may change the file labeling from the **File Labeling** option. You need provide file specification (normally the root, top level directories and mount points), file type (all files, symbolic link, directory, regular file, character device, etc), SELinux Type and MLS (Multi-Level Security such as top secret, secret, confidential and unclassified) details. Click on any row and click on properties to edit the configuration. Figure 2.28 shows File Labeling through SELinux management tool.

Figure 2.28 File Labeling through SELinux Management tool



6. **User Mapping** allows you to categorize the users. For example, files labeled as ‘confidential’ can only be accessed by the users with a similar categorization (provided the local discretionary access control system also permits the action). This is known as MCS or Multi-Category Security. Figure 2.29 shows user mapping.
7. **SELinux User** option allows you to assign SELinux roles to the SELinux users.
8. **Translation** option allows you to set the sensitivity level translations as shown in Figure 2.30.



66 Chapter 2 • Hardening the Operating System

Figure 2.29 User Mapping

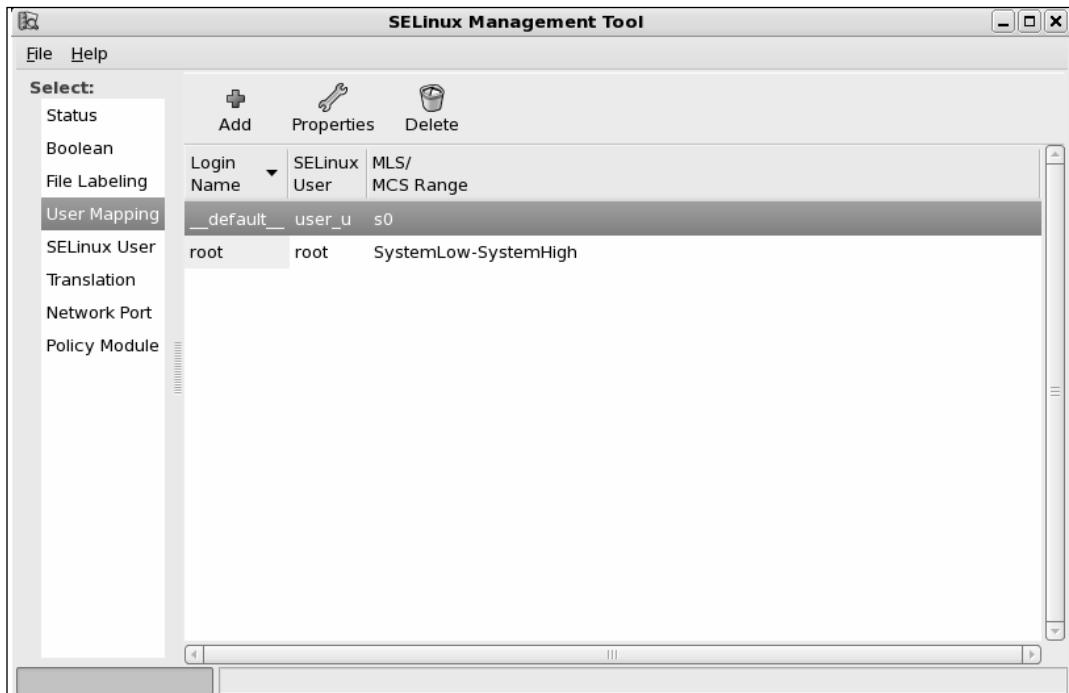
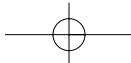
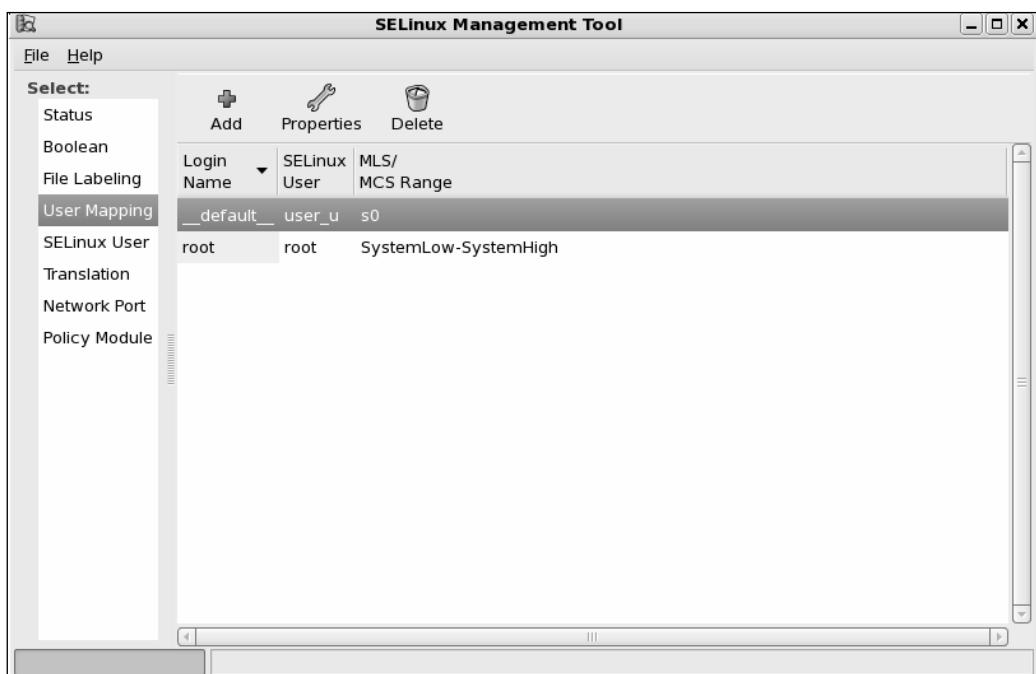
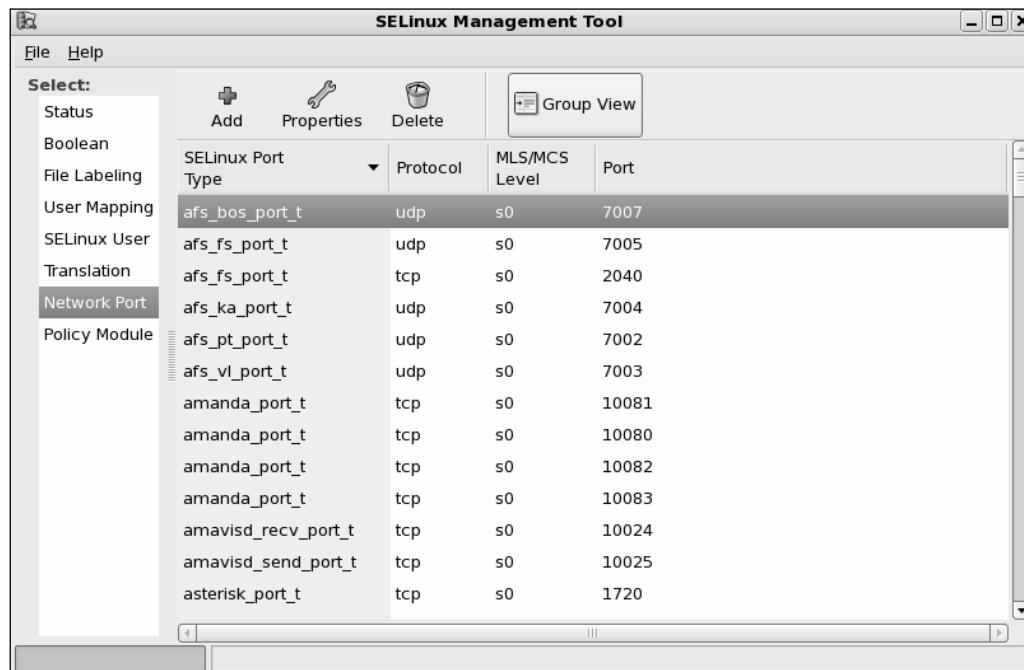


Figure 2.30 Translation



9. **Network Port** option allows you to configure MLS/MCS levels for various network ports. You need to specify the port number, protocol type (tcp or udp), SELinux type and the MLS/MCS Level. You may notice port number 80, protocol tcp, SELinux type http_port_t is configured for MLS/MCS level s0. Figure 2.31 shows Network Port option of SELinux Management tool.

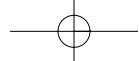
Figure 2.31 Network Port



10. **Policy Module** option allows you to enable or disable audit for specific modules. This will enable additional audit rules that are not reported in the log files.

NOTE

For more information refer to the Red Hat Enterprise Linux 5.0.0 Deployment Guide. To read more about SELinux please visit www.nsa.gov/selinux/.



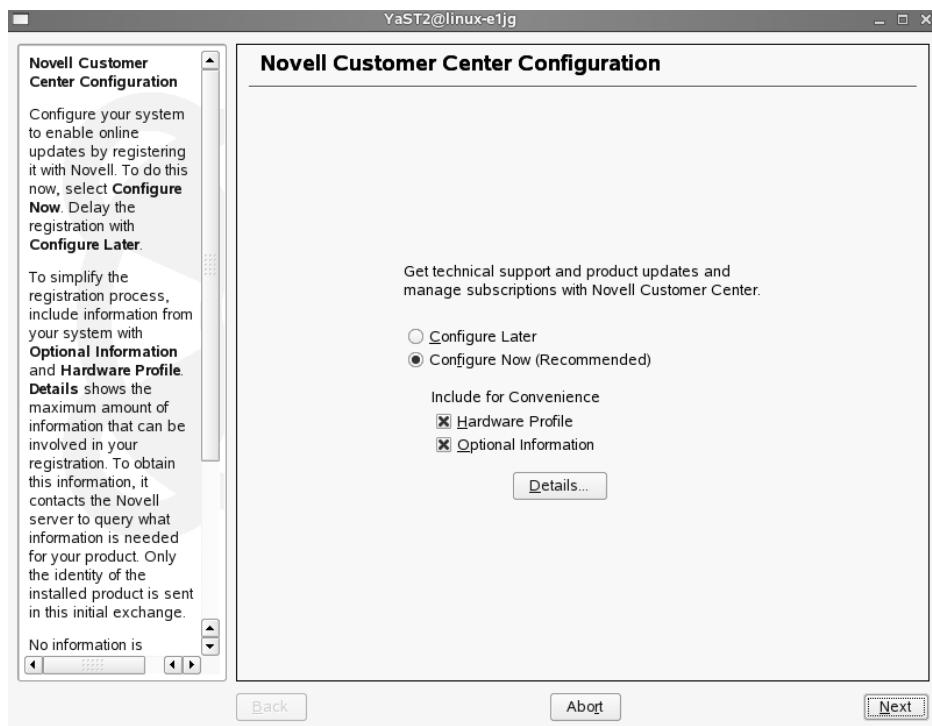
Securing Novell SUSE Linux

SUSE Linux from Novell is another popular vendor supported Linux. SUSE Linux is available for both servers and desktops. Similar to Red Hat Enterprise Linux, SUSE Linux Enterprise Server version 10 provides graphical user interface, easy to install and configure, automatic recognition of variety of hardware, server management and administration tools, automatic updates and technical support.

In this section let us look at the security tasks on Novell SUSE Linux that we have already discussed above on Red Hat Linux.

1. **Novell Customer Center** configuration helps you to create the profile of your SUSE Linux server to get the technical support, product updates and manage your subscriptions (licenses). Figure 2.32 shows the Novell customer center configuration.

Figure 2.32 Novell Customer Center Configuration



2. **System Services (Runlevel) Services** tool of YaST (Yet another Setup Tool) allows you to enable or disable your services. Earlier in this chapter we saw how to manually disable the services. You can also edit the run level of individual services. Figure 2.33 shows the service configuration tool.

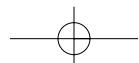
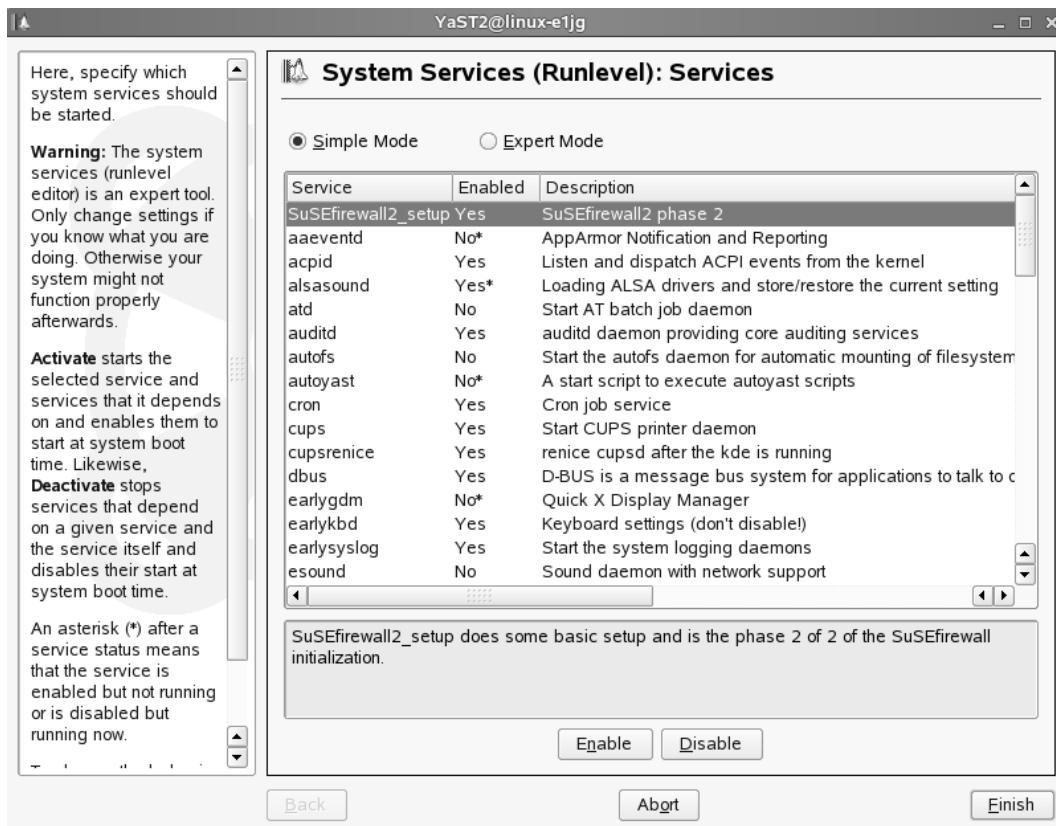


Figure 2.33 System Services of YaST

- Novell SUSE Linux provides you with various graphical user interface tools to configure security settings. To configure local security settings click on **Computer | YaST | Security and Users | Local Security**. Figure 2.34 shows the interface where you can configure password settings. These are some of the settings you performed earlier with Bastille Linux.

Local Security setting allows you to configure various security settings specific to the server:

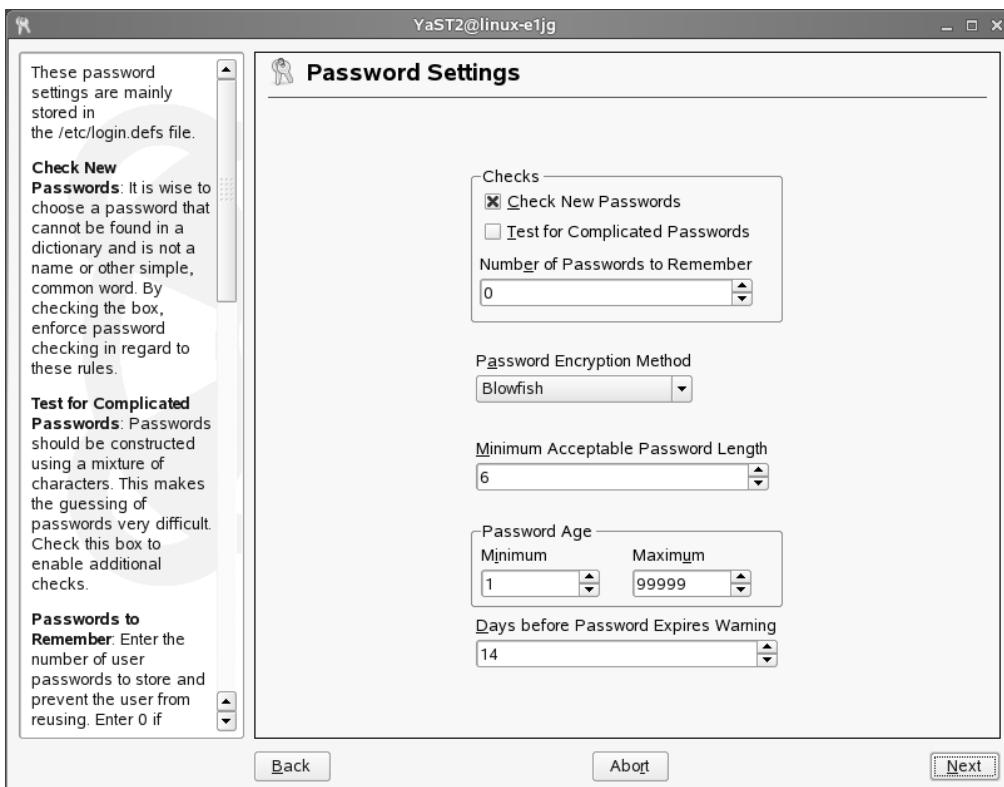
- System type** options such as home workstation, networked workstation, and network server. When you choose Network Server (network server is the computer that provides services) you will find more options to configure security.
- Password settings** such as check for new passwords, test for complicated passwords, number of passwords to remember, password encryption method, and minimum acceptable password length and password age.



70 Chapter 2 • Hardening the Operating System

6. **Boot settings** such as interpretation of Ctrl + Alt + Del (options available are ignore, reboot, halt) and shutdown behavior of KDM (only root, all users, nobody and automatic). KDM is the KDE login manager.
7. **Login Security** settings such as delay after incorrect login attempt, record successful login attempts.
8. **User Security** settings such as user ID limitations and group ID limitations (default is 1000 to 60000).
9. **Other Security Settings** such as file permissions (options available are easy, secure, paranoid which is extremely restrictive) and user launching updatedb (options are nobody and root). Updatedb is a program that runs after every boot to update the database with the location information of the files.

Figure 2.34 Local Security Settings



10. **Computer | YaST | Miscellaneous | View System Log.** This will open /var/log/messages. Look for keywords related to the services that you are troubleshooting. Figure 2.35 shows log messages.

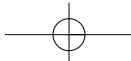
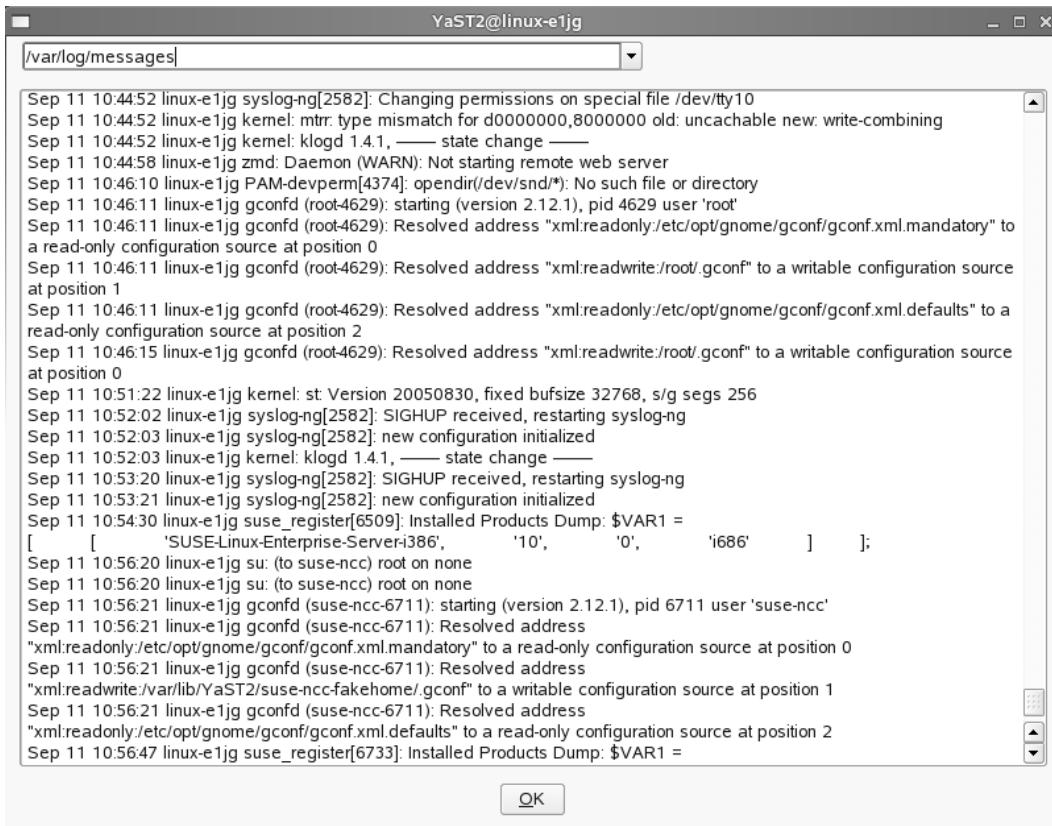


Figure 2.35 Logs


The screenshot shows a terminal window titled "YaST2@linux-e1jg" with the command "/var/log/messages" entered in the title bar. The window displays a scrollable list of log messages from the system boot process. The messages include kernel logs, syslog-NG logs, and configuration logs for services like gconfd and suse_register.

```

Sep 11 10:44:52 linux-e1jg syslog-ng[2582]: Changing permissions on special file /dev/tty10
Sep 11 10:44:52 linux-e1jg kernel: mtrr type mismatch for d0000000,8000000 old: uncachable new: write-combining
Sep 11 10:44:52 linux-e1jg kernel: klogd 1.4.1, — state change —
Sep 11 10:44:58 linux-e1jg zmd: Daemon (WARN): Not starting remote web server
Sep 11 10:46:10 linux-e1jg PAM-devperm[4374]: opendir(/dev/snd/*): No such file or directory
Sep 11 10:46:11 linux-e1jg gconfd (root-4629): starting (version 2.12.1), pid 4629 user 'root'
Sep 11 10:46:11 linux-e1jg gconfd (root-4629): Resolved address "xml:readonly:/etc/opt/gnome/gconf/gconf.xml.mandatory" to a read-only configuration source at position 0
Sep 11 10:46:11 linux-e1jg gconfd (root-4629): Resolved address "xml:readwrite:/root/.gconf" to a writable configuration source at position 1
Sep 11 10:46:11 linux-e1jg gconfd (root-4629): Resolved address "xml:readonly:/etc/opt/gnome/gconf/gconf.xml.defaults" to a read-only configuration source at position 2
Sep 11 10:46:15 linux-e1jg gconfd (root-4629): Resolved address "xml:readwrite:/root/.gconf" to a writable configuration source at position 0
Sep 11 10:51:22 linux-e1jg kernel: st: Version 20050830, fixed bufsize 32768, s/g segs 256
Sep 11 10:52:02 linux-e1jg syslog-NG[2582]: SIGHUP received, restarting syslog-NG
Sep 11 10:52:03 linux-e1jg syslog-NG[2582]: new configuration initialized
Sep 11 10:52:03 linux-e1jg kernel: klogd 1.4.1, — state change —
Sep 11 10:52:20 linux-e1jg syslog-NG[2582]: SIGHUP received, restarting syslog-NG
Sep 11 10:53:21 linux-e1jg syslog-NG[2582]: new configuration initialized
Sep 11 10:54:30 linux-e1jg suse_register[6509]: Installed Products Dump: $VAR1 =
[ 'SUSE-Linux-Enterprise-Server-i386', '10', '0', '1686' ];
Sep 11 10:56:20 linux-e1jg su: (to suse-ncc) root on none
Sep 11 10:56:20 linux-e1jg su: (to suse-ncc) root on none
Sep 11 10:56:21 linux-e1jg gconfd (suse-ncc-6711): starting (version 2.12.1), pid 6711 user 'suse-ncc'
Sep 11 10:56:21 linux-e1jg gconfd (suse-ncc-6711): Resolved address "xml:readonly:/etc/opt/gnome/gconf/gconf.xml.mandatory" to a read-only configuration source at position 0
Sep 11 10:56:21 linux-e1jg gconfd (suse-ncc-6711): Resolved address "xml:readwrite:/var/lib/YaST2/suse-ncc-fakehome/.gconf" to a writable configuration source at position 1
Sep 11 10:56:21 linux-e1jg gconfd (suse-ncc-6711): Resolved address "xml:readonly:/etc/opt/gnome/gconf/gconf.xml.defaults" to a read-only configuration source at position 2
Sep 11 10:56:47 linux-e1jg suse_register[6733]: Installed Products Dump: $VAR1 =

```

11. Computer | YaST | Miscellaneous | View Start-up Log. This will open /var/log/boot.msg. This is a good place to locate the errors and status regarding the services that start or fail to start during the boot time. Figure 2.36 shows boot messages.

72 Chapter 2 • Hardening the Operating System

Figure 2.36 Boot Messages

```

YaST2@linux-e1jg
/var/log/boot.msg

COLUMNS=123 PATH=/bin:/usr/bin:/sbin:/usr/sbin vga=0x317 RUNLEVEL=5 PWD=/
SPLASHCFG=/etc/bootsplash/themes/SuSE-SLES/config/bootsplash-1024x768.cfg PREVLEVEL=N LINES=44 SHLVL=2
HOME=/ splash=silent SPLASH=yes ROOTFS_BLKDEV=/dev/sdb7 _=/usr/bin/ionice DAEMON=/usr/sbin/cupsd ]
Starting cupsddone
<notice>startproc: execve (/opt/gnome/sbin/gdm) [ /opt/gnome/sbin/gdm ], [ CONSOLE=/dev/console
ROOTFS_FSTYPE=reiserfs TERM=linux SHELL=/bin/sh ROOTFS_FSCK=0 INIT_VERSION=sysvinit-2.86
KDEROOTHOME=/root kdm REDIRECT=/dev/pty1 COLUMNS=123 PATH=/bin:/usr/bin:/sbin:/usr/sbin vga=0x317
RUNLEVEL=5 PWD=/ SPLASHCFG=/etc/bootsplash/themes/SuSE-SLES/config/bootsplash-1024x768.cfg LANG=en_US UTF-8
PREVLEVEL=N LINES=44 QT_SYSTEM_DIR=/usr/share/desktop-data SHLVL=2 HOME=/ XCURSOR_THEME=Industrial
WINDOWMANAGER=/usr/X11R6/bin/gnome splash=silent SPLASH=yes ROOTFS_BLKDEV=/dev/sdb7 _=/sbin/startproc
DAEMON=/opt/gnome/sbin/gdm ]
Starting service gdmdone
<notice>killproc: kill(2586,20)
<notice>killproc: kill(2582,1)
<notice>killproc: kill(2586,18)
<notice>killproc: kill(2586,12)
<notice>startproc: execve (/usr/sbin/powersaved) [ /usr/sbin/powersaved -d -f /var/run/acpid.socket -v 3 ], [ HOME=/
PATH=/bin:/usr/bin:/sbin:/usr/sbin SHELL=/bin/sh RUNLEVEL=5 PREVLEVEL=N DAEMON=/usr/sbin/powersaved ]
Starting powersaved: done
Starting mail service (Postfix)done
Starting CRON daemon<notice>startproc: execve (/usr/sbin/cron) [ /usr/sbin/cron ], [ CONSOLE=/dev/console
ROOTFS_FSTYPE=reiserfs TERM=linux SHELL=/bin/sh ROOTFS_FSCK=0 LC_ALL=POSIX INIT_VERSION=sysvinit-2.86
REDIRECT=/dev/pty1 COLUMNS=123 PATH=/bin:/usr/bin:/sbin:/usr/sbin vga=0x317 RUNLEVEL=5 PWD=/
SPLASHCFG=/etc/bootsplash/themes/SuSE-SLES/config/bootsplash-1024x768.cfg PREVLEVEL=N LINES=44 SHLVL=2
HOME=/ splash=silent SPLASH=yes ROOTFS_BLKDEV=/dev/sdb7 _=/sbin/startproc DAEMON=/usr/sbin/cron ]
done
Starting Firewall Initialization (phase 2 of 2) SuSEfirewall2: Warning: ip6tables does not support state matching. Extended IPv6
support disabled.
done
Master Resource Control: runlevel 5 has been reached
Skipped services in runlevel 5: irq_balancer smbfss nfs
<notice>killproc: kill(2425,3)

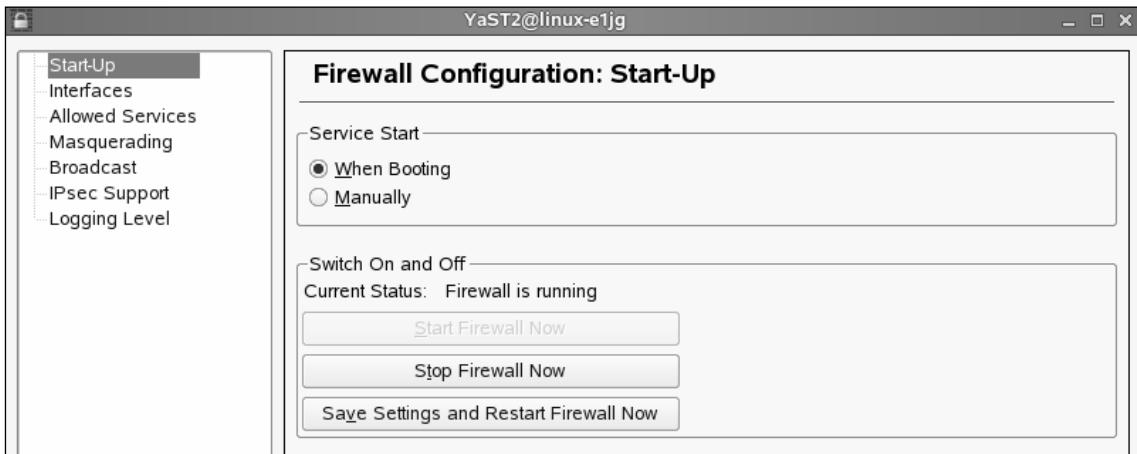
OK

```

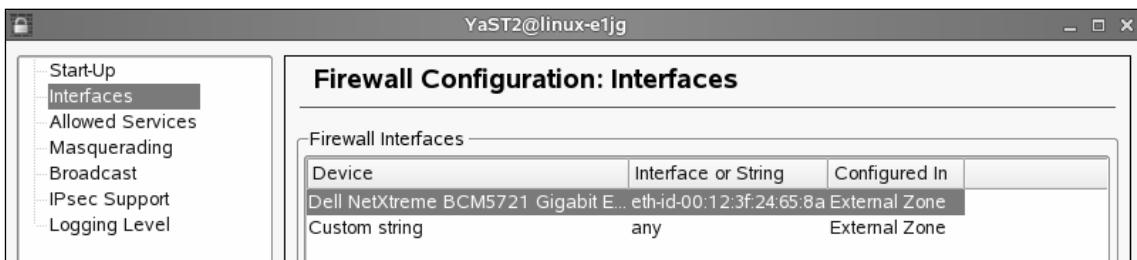
Firewall Configuration

SUSE Linux provides you with a graphical interface to configure firewall on your server. Following section describes the options available to configure firewall. To configure the firewall on your server click on **Computer | YaST | Security and Users | Firewall**.

1. To start the firewall when the server starts select '**When Booting**' radio button in the **Start-Up** page. You have the options to start the firewall manually, start or stop the firewall or save the settings to restart the firewall. Figure 2.37 shows the start-up screen of firewall configuration.

Figure 2.37 SUSE Linux Firewall Configuration

2. **Interfaces** participating in the firewall configuration can be configured from **Interfaces** page. In a typical firewall server you may find more than one interface. Figure 2.38 shows Interfaces option of firewall configuration.

Figure 2.38 Firewall Interfaces

3. **Allowed Services** screen allows you to choose the permitted services for every zone. Figure 2.39 shows allowed services page for external zone. You also have an option (not shown in the picture) to enable (checkbox) protection for internal zone. If you could not find the services from the list you can add custom services (ports) manually by specifying the same. Click **Add** to provide tcp, udp, rpc ports and ip protocol options.

74 Chapter 2 • Hardening the Operating System

Figure 2.39 Allowed Services



4. **Masquerading** option allows you to perform address translation (or address hiding). You need at least one external and one internal interface to configure masquerading.
5. In addition to the above you have options to configure **Broadcast**, **IPsec** support and **Logging** levels.

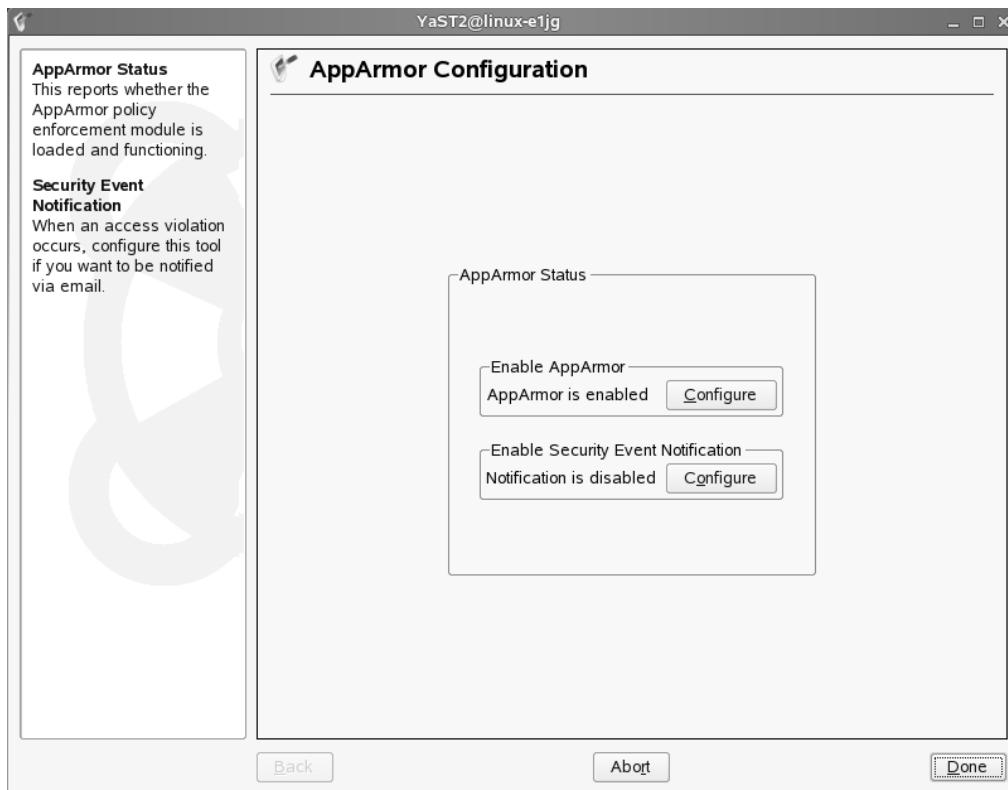
Novell AppArmor

Novell AppArmor provides a mechanism to protect your applications from their vulnerabilities. Novell calls it as '*immunizing*.' To immunize your system you need to install AppArmor (installed by default in Novell SUSE Linux Enterprise Server 10), setup AppArmor profiles and reboot the system. This section briefly describes the process of configuring AppArmor on your SUSE Linux server.

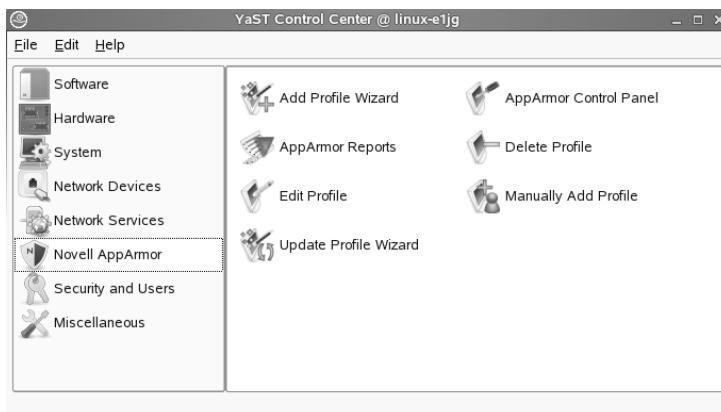
Most of the common application and services have profiles created by default by AppArmor. When you add new applications and services you can create profiles individually for these applications. Make sure you stop the applications before you create profiles. When you create the profile, AppArmor runs *autodep* to analyze and create an initial profile for the application. A manual profiling by you is very much required to avoid restricting the application too granularly that it can not access the directories required to perform its routine functions. AppArmor has a learning mode to monitor various access of the application when you run the same. Information gathered by the learning mode is included in the initial profile created by AppArmor. Creating profiles for critical applications ensures the application run in a specific security environment ensuring no damage to the system due to the vulnerabilities that may get developed time-to-time.

You need to configure AppArmor to immunize programs that may grant privileges, open ports, cron jobs, server daemons and web applications.

1. First you need to enable AppArmor as shown in the Figure 2.40.

Figure 2.40 Enabling AppArmor

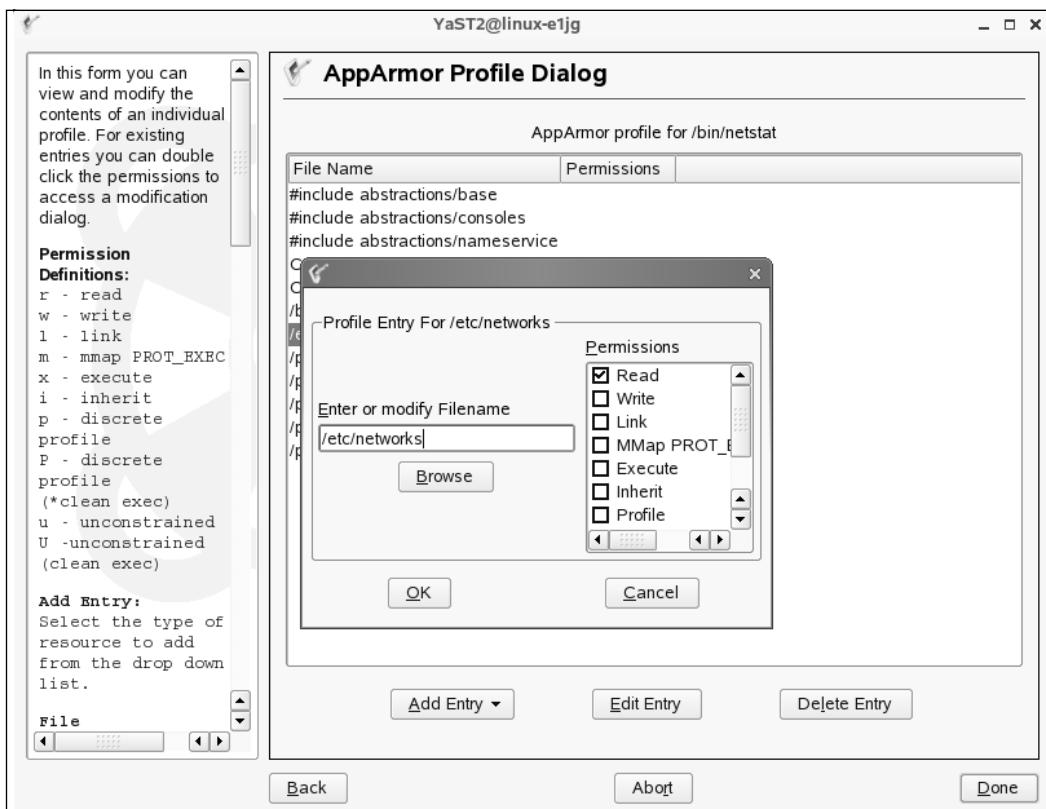
2. Click **Computer | YaST | Novell AppArmor** as shown in Figure 2.41. You have the following options:
3. Add profile wizard, AppArmor control panel, AppArmor reports, delete profile, edit profile, manually add profile and update profile wizard.

Figure 2.41 Novell AppArmor Configuration

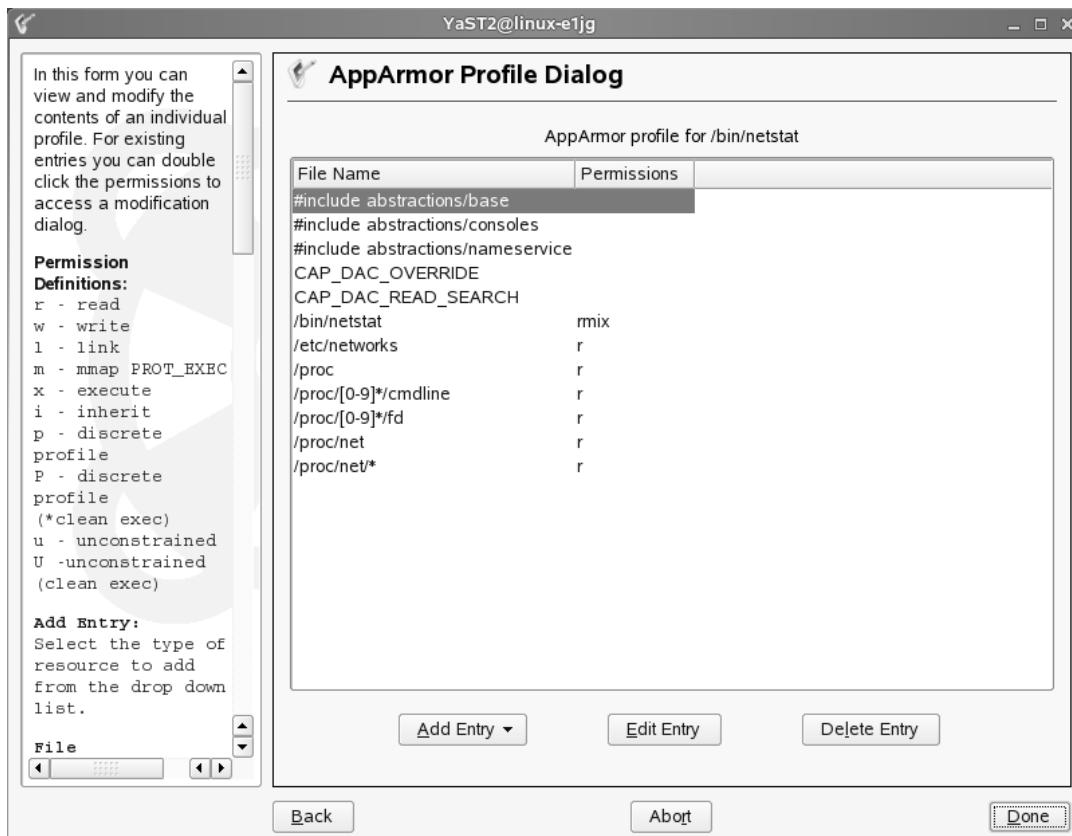
76 Chapter 2 • Hardening the Operating System

4. Then click on **Edit Profile**. Select the entry for ‘netstat’ and click on ‘Edit Entry’. You can see the available permissions for the application netstat on /etc/networks file. The options you have are read, write, link, mmap, execute, inherit, discrete profile, unconstrained, Unstrained (clean exec). Figure 2.42 shows AppArmor profile dialog.

Figure 2.42 AppArmor Profile



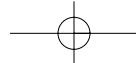
5. Finally, exit the **Edit Entry** screen. You will see the summary of permissions available for the application netstat to run in your server. Figure 2.43 shows the profile for netstat.

Figure 2.43 AppArmor Profile for Netstat**NOTE**

For more information refer to the AppArmor Administration Guide at www.novell.com

Host Intrusion Prevention System

Host Intrusion Prevention System or HIPS as they are popularly known creates a shield around your servers and ensure no malicious attacks or unwarranted changes happen to the system. Apart from providing behavioral rule based protection, signature based analysis and stateful firewall level protection; HIPS also ensure no changes happen to core operating system files from unknown exploits. Commercial grade HIPS provide colorful graphical reporting, advanced alerting system, centralized management and round-the-clock updates



78 Chapter 2 • Hardening the Operating System

and technical support from the vendors. Many security vendors have come-up with HIPS solutions for Unix/Linux variants. Though the discussion of every HIPS solution or their features is beyond the scope of this book, find below a list of HIPS products available Cisco, McAfee, ISS (now IBM ISS) and Enterasys. Web server and database server specific editions are also available that helps you to prevent attacks such as SQL injection and directory traversal. Needless to say, all these vendors have products for Microsoft platforms as well.

Cisco Security Agent (CSA version 5.2) available for following Linux Server editions apart from Microsoft Windows. CSA is also available for Linux desktop editions.

- Solaris 8 SPARC architecture (64-bit kernel)
- Solaris 9 SPARC architecture (64-bit kernel)
- Red Hat Enterprise Linux 3.0 ES and AS
- Red Hat Enterprise Linux 4.0 ES and AS

Enterasys Dragon Host Sensors are available for the following operating systems:

- Linux
- AIX
- Solaris,
- HP-UX

Enterasys Dragon Host Sensors for Web Intrusion Prevention support:

- WebIPS for Apache with Linux
- WebIPS for Solaris

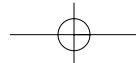
IBM Internet Security Systems is available for following Unix variants apart from Microsoft Windows:

- IBM RealSecure Server Sensor 7.0 for Solaris
- IBM RealSecure Server Sensor 7.0 for HP-UX
- IBM RealSecure Server Sensor 7.0 for AIX

McAfee Host Intrusion Prevention:

Available platforms include:

- Red Hat Enterprise Linux 4.0 (32-bit only, servers only) supported kernels:
 - 2.6.9-11.EL-2.6.9-11.EL-smp
 - 2.6.9-22.EL-2.6.9-22.EL-smp
 - 2.6.9-34.EL-2.6.9-34.EL-smp



- Sun Solaris (servers only)
- SPARC Solaris 8, sun4u (32-bit or 64-bit kernel)
- SPARC Solaris 9, sun4u (32-bit or 64-bit kernel)
- SPARC Solaris 10, sun4u (64-bit kernel only)

McAfee Host Intrusion Prevention is also available on following web server platforms:

- Apache 1.3.6 and higher
- Apache 2.0.42 or higher
- iPlanet 4.0 and 4.1
- Sun Java System (formerly Sun ONE) 6.0 and 6.1

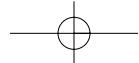
Linux Benchmark Tools

Center for Internet Security (CIS) is a non-profit organization that helps organizations to mitigate the business risks and stoppage of ecommerce services due to lack of security measures. CIS periodically releases benchmark and scoring tools. Benchmark tools are available for operating systems, network devices and applications.

You need to register before you can download the benchmark and scoring tools. CIS categorizes these tools as level 1 and level 2 tools. Level 1 tool is for system administrations with a basic knowledge of security. These tools are less likely to cause any interruption while run and can be monitored by tools provided by CIS.

CIS tool runs a series of tests on the server or the network devices and produces html and xml reports. The reports provide information on the following areas:

- Patches, Packages and Initial Lockdown
- Minimize xinetd network services
- Minimize boot services
- Kernel Tuning/Network Parameter Modifications
- Logging
- File/Directory Permissions/Access
- System Access, Authentication, and Authorization
- User Accounts and Environment
- Warning Banners
- Reboot



80 Chapter 2 • Hardening the Operating System

- Anti-Virus Consideration
- Remove Backup Files

In this exercise let's download Red Hat Linux benchmark tool, run the tests and see the results.

1. Download Red Hat Linux benchmark tool from www.cisecurity.org/bench_linux.html. The filename should resemble:

```
ng_scoring_tool-1.0-linux-nojvm.tar
```

2. Download and install Java runtime from www.java.com. Red Hat is currently supported only through a non-JVM (Java Virtual Machine) package. For SUSE Linux you have a bundled jvm package. The Java file name should resemble

```
jre-1_5_0_12-linux-i586.bin
```

3. Execute the .bin file downloaded above to install Java run time.

```
#./ jre-1_5_0_12-linux-i586.bin
```

4. Set the Java environment variables.

```
#set JAVA_HOME=/jre1-1.5.0.12 (the name of the direct created by the above execution file. /jre1-1.5.012/bin is the location where the executables of Java is stored)
```

```
#export JAVA_HOME
```

5. Install the CIS Red Hat Linux benchmark tool

```
#tar xvf ng_scoring_tool-1.0-linux-nojvm.tar.tar (this will unzip ng_scoring_tool-1.0-linux-nojvm.jar)
```

```
#/jre1-1.5.012/bin/java -jar ng_scoring_tool-1.0-linux-nojvm.jar
```

Follow the simple graphical installation wizard to complete the installation at the default path /opt/CISngtool.

6. To start the tool (as shown in the Figure 2.44)

```
# cd /opt/CISngtool
```

```
# ./ng.sh
```

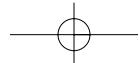
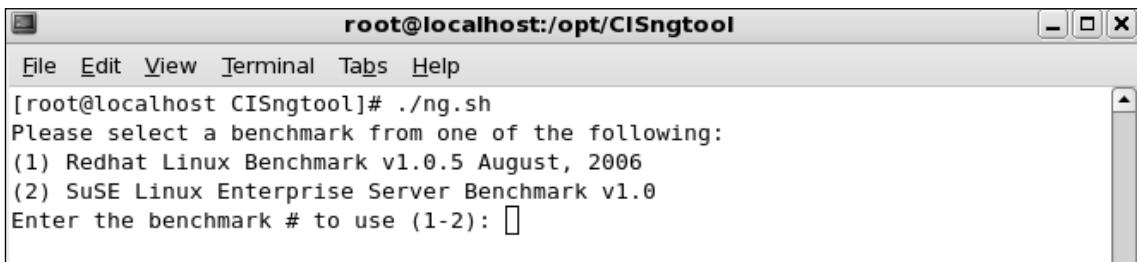
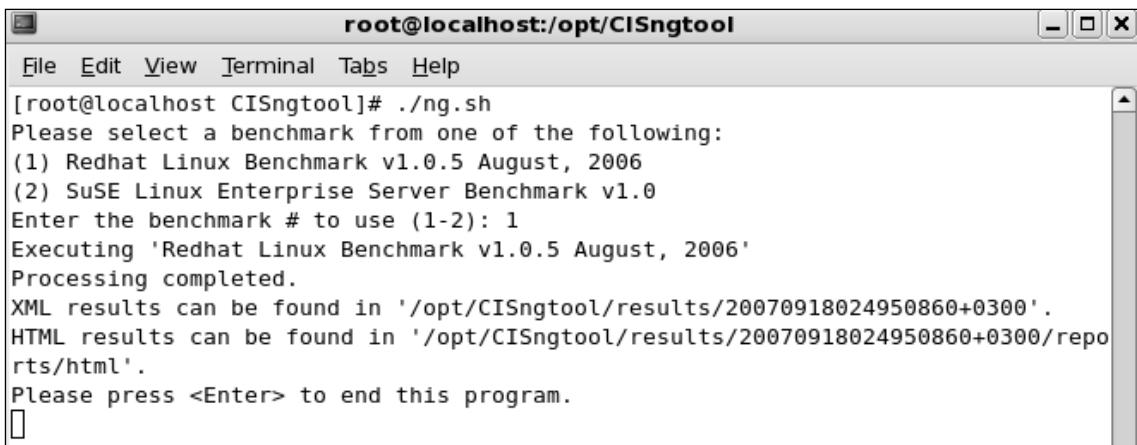


Figure 2.44 Executing CIS Red Hat Linux Benchmark tool


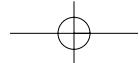
```
root@localhost:~/opt/CISngtool
File Edit View Terminal Tabs Help
[root@localhost CISngtool]# ./ng.sh
Please select a benchmark from one of the following:
(1) Redhat Linux Benchmark v1.0.5 August, 2006
(2) SuSE Linux Enterprise Server Benchmark v1.0
Enter the benchmark # to use (1-2):
```

7. Type **1** and press **Enter** to run the Redhat Linux Benchmark v1.0.5 August, 2006 benchmark tool (as shown in the Figure 2-45). After few minutes of running the tests xml and html results are stored in the /opt/CISngtool/results directory with the date and time stamp as the name of the html file.
8. Press **Enter** to end this program.
9. Open the html file generated by the tool to view the results.

Figure 2.45 Testing Process


```
root@localhost:~/opt/CISngtool
File Edit View Terminal Tabs Help
[root@localhost CISngtool]# ./ng.sh
Please select a benchmark from one of the following:
(1) Redhat Linux Benchmark v1.0.5 August, 2006
(2) SuSE Linux Enterprise Server Benchmark v1.0
Enter the benchmark # to use (1-2): 1
Executing 'Redhat Linux Benchmark v1.0.5 August, 2006'
Processing completed.
XML results can be found in '/opt/CISngtool/results/20070918024950860+0300'.
HTML results can be found in '/opt/CISngtool/results/20070918024950860+0300/reports/html'.
Please press <Enter> to end this program.
```

10. View the results categorized with details such as passed and failed items, actual and maximum scores along with the name of the server, scan date and time. Figure 2.46 shows the test results.



82 Chapter 2 • Hardening the Operating System

Figure 2.46 Test Results

Description	Items		Score	
	Passed	Failed	Actual	Max
1 Patches, Packages and Initial Lockdown	1	2	3.704	11.111
2 Minimize xinetd network services	6	2	8.333	11.111
3 Minimize boot services	11	10	5.820	11.111
4 Kernel Tuning/Network Parameter Modifications	0	2	0.000	11.111
5 Logging	2	2	5.556	11.111
6 File/Directory Permissions/Access	1	8	1.235	11.111
7 System Access, Authentication, and Authorization	2	9	2.020	11.111
8 User Accounts and Environment	6	6	5.556	11.111

11. **Click** and **Expand** the view to see the results from the individual categories. Figure 2.47 shows the categories of the results. You may notice the status of individual test results such as not tested, failed or passed.
12. **Click** and further **Expand** the view to read the description of individual tests, for example, **click** 2.1 Disable Standard Services. Figure 2.48 shows the description of the test.

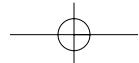


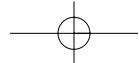
Figure 2.47 Result Categories

Description	Status
1 Patches, Packages and Initial Lockdown	
1.1 Apply Latest OS Patches	Not Tested
1.2 Validate Your System Before Making Changes	Not Tested
1.3 Configure SSH	Failed
1.4 Enable System Accounting	Failed
1.5 Install and Run Bastille	Passed
2 Minimize xinetd network services	
2.1 Disable Standard Services	Failed
2.2 Configure TCP Wrappers and Firewall to Limit Access	Failed
2.3 Only Enable telnet If Absolutely Necessary	Passed
2.4 Only Enable FTP If Absolutely Necessary	Passed
2.5 Only Enable rlogin/rsh/rcp If Absolutely Necessary	Passed

Figure 2.48 Details of Individual Tests

2.1 Disable Standard Services	OVAL5	Failed
Description		
2.2 Configure TCP Wrappers and Firewall to Limit Access	Check Type:	Status:
Description		
TCP Wrappers and Host-Based Firewalls are presented together as they are similar and complementary in functionality.		

CIS also provides downloads for SUSE Linux and Slackware Linux. PDF documents that are downloaded as a part of the benchmark tool archives consist of step-by-step instructions to implement the desired security level on your servers. These recommendations help you to harden your Linux servers. By running the tests periodically you can ensure that your servers are not exposed to serious threats.

**NOTE**

For more information and to test CIS benchmark tools visit
www.cisecurity.org

Summary

This chapter covered the basics of hardening a server to avoid security vulnerabilities using Linux. The main sections covered disabling unnecessary services, locking down ports, Bastille, sudo, and logging enhancers.

It is extremely important to install the latest service pack or updates to the operating system, which fix many security vulnerabilities and bugs before you install any programs. Many services provided with operating systems are not required and can be removed. The key to remember is that the fewer services running, the less potential vulnerability.

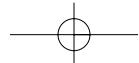
TCP/UDP ports were covered in this chapter, and how each port is used by specific services. If you block ports on your server, you block the services that use those ports. Locking down ports is an excellent way to reduce exploitations of your system.

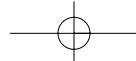
Maintaining your server involves downloading service packs and updates, and requires regularly installing bug fixes, security patches, and software updates. These items are available through the operating system vendors, as well as the specific vendors that created the software that you implement.

Bastille is an open source program that facilitates the hardening of a Linux system. It performs many of the tasks listed previously, disabling services and ports that are not required for the system's job functions. Bastille is powerful and can save administrators time from configuring each individual file and program throughout the operating system. Instead, administrators answer a series of "Yes" and "No" questions through an interactive graphical interface. The program automatically implements the administrators' preferences based on the answers to the questions.

Superuser Do (sudo) is an open source security tool that allows an administrator to give specific users or groups the ability to run certain commands as root or as another user. The program can also log commands and arguments entered by specified system users. The developers of sudo state that the basic philosophy of the program is to "give as few privileges as possible, but still allow people to get their work done."

Logging enhancers are tools that simplify logging by allowing logging information to be filtered and often displaying logs in simplified formats. Many open source logging programs exist to make system administration easier. You were introduced in this chapter to SWATCH, scanlogd, and syslog-ng.





SWATCH is an open source package that allows administrators to efficiently monitor system activity. It can monitor events on a system, or a large number of systems, by monitoring system logs for specified events. SWATCH's main function is to monitor messages actively as they are written to log files through the Unix syslog utility.

Scanlogd is an open source program that detects and logs TCP-port scanning on a system. Scanlogd can alert an administrator when the network is being mapped, but it cannot stop the intrusion.

Syslogd-ng is a logging daemon that is the replacement for the traditional syslogd. The “ng” is an acronym for “next generation.” The original syslogd was the general Unix logging daemon that handled request for syslog services, but was difficult to configure. Syslogd-ng is easier to configure and offers additional logging features, such as more configurations. For example, syslogd-ng allows administrators to filter messages based on priority, as well as the content of the messages.

Security Enhanced Linux (SELinux) is an initiative from NSA along with the computer security research community to offer enhanced security in the operating systems. SELinux enhances security by utilizing mandatory access control features in Linux. Red Hat offers SELinux from Red Hat Enterprise Linux (RHEL) version 4 onwards.

Novell SUSE Linux is another popular Linux distribution that offers graphical user interface right from the installation, configuration and management. SUSE Linux provides graphical firewall configuration tools as well as to configure finer security settings.

Novell AppArmor protects the system from inherent vulnerabilities of the applications. Novell calls it as *immunizing* the system. By creating AppArmor profiles for open ports, cron jobs and web applications security can be enhanced on Linux servers.

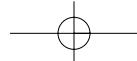
Host Intrusion Prevention System (HIPS) shields the servers from zero-day attacks, providing signature-based, behavior-based protection and ensures core operating system files are not changed by external attacks. Cisco, Enterasys, McAfee, IBM ISS and several other security vendors provide HIPS on major operating system platforms.

Center for Internet Security (CIS) periodically releases benchmarking and scoring tools for operating systems, network devices and applications. You may download these free-of-cost tools and run them on your servers to analyze the security. CIS benchmark tool is available for Red Hat, SUSE and Slackware Linux.

Solutions Fast Track

Updating the Operating Systems

- Operating system releases usually contain software bugs and security vulnerabilities.
- Operating system vendors or organizations offer fixes, corrections, and updates to the system. For example, Red Hat offers this material at its Web site, which includes Update Service Packages and the Red Hat Network.



86 Chapter 2 • Hardening the Operating System

- You should always ensure your system has the latest necessary upgrades. Many errata and Update Service Packages are not required for every system. You should always read the associated documentation to determine if you need to install it.

Handling Maintenance Issues

- After your system goes live, you must always maintain it by making sure the most current patches and errata are installed, which include the fixes, corrections, and updates to the system, as well as the applications running on it.
- You should always check the Red Hat or the appropriate vendor site for the latest errata news and security advisories.
- For example, Red Hat security advisories provide updates that eliminate security vulnerabilities on the system. Red Hat recommends that all administrators download and install the security upgrades to avoid denial-of-service (DoS) and intrusion attacks that can result from these weaknesses.

Manually Disabling Unnecessary Services and Ports

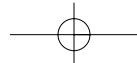
- You should always disable vulnerable services and ports on your system that are not used. You are removing risk when you remove unnecessary services.
- The /etc/xinetd.d directory makes it simple to disable services that your system is not using. For example, you can disable the FTP and Telnet services by commenting out the FTP and Telnet entries in the respective file and restarting the service. If the service is commented out, it will not restart.

Locking Down Ports

- When determining which ports to block on your server, you must first determine which services you require. In most cases, block all ports that are not exclusively required by these services.
- To block TCP/UDP services in Linux, you must disable the service that uses the specific port.

Hardening the System with Bastille

- The Bastille program facilitates the hardening of a Linux system. It saves administrators time from configuring each individual file and program throughout the operating system.



- Administrators answer a series of “Yes” and “No” questions through an interactive graphical interface. The program automatically implements the administrators’ preferences based on the answers to the questions.
- Bastille can apply restrictive permissions on administrator utilities; disable unnecessary services and ports, and much more.

Controlling and Auditing Root Access with Sudo

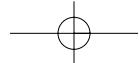
- Sudo (Superuser Do) allows an administrator to give specific users or groups the ability to run certain commands as root or as another user.
- Sudo features command logging, command restrictions, centralized administration of multiple systems, and much more.
- The **sudo** command is used to execute a command as a superuser or another user. In order to use the **sudo** command, the user must supply a username and password. If a user attempts to run the command via sudo and that user is not entered in the sudoers file, an e-mail is automatically sent to the administrator, indicating that an unauthorized user is accessing the system.

Managing Your Log Files

- Logging allows administrators to see who and what has accessed their system. Many helpful Linux log files are located in the /var/log directory.
- Linux offers commands that allow administrators to access useful log files. Two commands of interest are *last* and *lastlog*. The message file also offers useful data for determining possible security breaches on your system.
- The Linux logs should be checked frequently to determine if any security violations have occurred on your system. Logs do not offer solutions, so you must analyze the data and decide how to counteract the attack.

Using Logging Enhancers

- Logging enhancers are tools that simplify logging by allowing logging information to be filtered and often displaying logs in simplified formats.
- Viewing text-based files with hundreds or thousands of entries can be burdensome, especially if you are only looking for one specific error entry.
- Three popular logging services used by administrators are SWATCH, scanlogd, and the next generation of syslogd (syslogd-*ng*).



Security Enhanced Linux

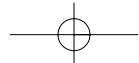
- Security Enhanced Linux (SELinux) is a research project from NSA and the computer security community supported by major Linux vendors.
- SELinux enhances security by working on mandatory access control architecture
- Red Hat Enterprise Linux provides SELinux Management Tool to configure SELinux.

Securing Novell SUSE Linux

- Novell SUSE Linux is another popular Linux distribution. SUSE Linux comes in server and desktop editions.
- Novell provides graphical user interface to install, configure and manage Linux servers.
- Most of the common tasks, configuring OS security settings and firewall can be performed through graphical user interface

Novell AppArmor

- AppArmor is used to create application specific security profiles.
- AppArmor performs static analysis to provide an initial profile that consists of permissions required on file system for the application to run smoothly.
- AppArmor ensures application vulnerabilities do not affect the system.



Frequently Asked Questions

The following Frequently Asked Questions, answered by the authors of this book, are designed to both measure your understanding of the concepts presented in this chapter and to assist you with real-life implementation of these concepts. To have your questions about this chapter answered by the author, browse to www.syngress.com/solutions and click on the "Ask the Author" form.

Q: I have a server that is strictly a mail server and uses SMTP and POP3. However, I want to download security patches from my vendor's Web site directly to the server. Even though I open the TCP/UDP port 80 (HTTP) and port 53 (DNS), I am unable to download the patches on the mail server.

A: You can download your updates from another system on your network and install the updates on your server through a CD.

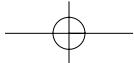
Q: Should I place my e-mail server behind the firewall, or in a service network (that is, a "demilitarized zone")?

A: Standard practice is to place the e-mail server in the DMZ. A DMZ is usually comprised of a screening router that blocks most attacks (denial-of-service, system scanning, attacks against Microsoft NetBIOS ports, etc.), and a firewall device that authoritatively blocks incoming traffic, effectively separating the internal network from the world. The DMZ exists between the screening router and the firewall. However, it is often a best practice to place the e-mail server behind the firewall itself. If you do this, however, you must make sure your firewall is configured correctly. Otherwise, a malicious user can take advantage of a misconfigured firewall and gain access to your internal network.

Q: When I install Bastille and run configure, why does the program report that the C compiler cannot create an executable?

A: This error most likely indicates that your system does not have a functioning compiler. It often occurs because you do not have a license, or part of the compiler suite cannot be located. Access and view the config.log to determine the cause. Many compiler components are found in /usr/css/bin. This path may not be identified in the environment variable PATH.

Q: By default, sudo uses syslog(3) for logging. Since I did not change this default during setup, why am I not generating any logging messages?



90 Chapter 2 • Hardening the Operating System

A: In order to generate sudo log files, you need to create a /var/log/sudo file, and add an entry to the syslog.conf file. Since the default log facility is local2, you must add the following line with TAB keys separating the facility (local2.debug) from the destination (a local logging file).

```
local2.debug          /var/log/sudo
```

You must then restart syslogd to ensure that it re-reads the file.

Q: I am tired of entering my password in sudo each time my ticket expires. How can I avoid this hassle?

A: Use the NOPASSWD tag in sudoers for specific users and commands by inserting the tag before the command list. If you want to disable all sudo passwords, there are two methods. You can run configure with the — **without-passwd** option, or you can add **!authenticate** to the Default line in sudoers. Finally, you can disable passwords to users and hosts in sudoers by adding specific user or host Defaults entries. See the sudo man file for specifics on disabling sudo password prompts.

