



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Securing Linux/Unix (Security 506)"
at <http://www.giac.org/registration/gcux>

GIAC Certified UNIX Security Administrator (GCUX)

Practical Assignment Version 1.9
Option 1 – Securing Unix Step by Step

Using and Securing a Red Hat Linux 7.3 server as a DNS, Mail, and Web server.

© SANS Institute 2000 - 2002, Author retains full rights.

Mark Oram
May 20, 2002

TABLE OF CONTENTS

1	Introduction	4
1.1	Project.....	4
1.2	Audience.....	4
1.3	Using this Document	4
2	Description	5
2.1	Goal	5
2.2	Requirements	5
2.2.1	High Level / Executive	5
2.2.2	Technical.....	5
2.3	Hardware	6
2.4	Software.....	7
2.4.1	Base Red Hat Linux 7.3 Operating System.....	7
2.4.2	Updates to the Red Hat Linux 7.3 Operating System	7
2.4.3	Third party Software	7
3	Risk Analysis.....	9
3.1	High Level.....	9
3.2	Exposed Applications	9
3.2.1	Apache.....	9
3.2.2	Bind (DNS).....	9
3.2.3	OpenSSH.....	10
3.2.4	Sendmail.....	10
3.2.5	POP.....	10
3.3	Other Concerns	10
3.3.1	Binding of Ports.....	10
3.3.2	Modification of files.....	10
3.3.3	Detection of Attacks.....	10
3.3.4	Users.....	10
4	Installation.....	12
4.1	Base Red Hat Linux 7.3 Installation.....	12
4.1.1	Manual Method.....	12
4.1.2	Automatic Method.....	18
4.2	Applying updates to Red Hat Linux 7.3.....	19
4.3	Configuring The System	20
4.3.1	OpenSSH.....	20
4.3.2	DNS / Bind.....	21
4.3.3	SMTP / Sendmail	24
4.3.4	SMTPTS (More Sendmail)	30
4.3.5	POP and POPS.....	32
4.3.6	HTTP and HTTPS / Apache	35
4.3.7	Syslog.....	39

4.3.8	Snort.....	39
4.3.9	Kernel level IP filtering (Netfilter / IPTables).....	41
4.3.10	TCP Wrappers	43
4.3.11	Shutting off unneeded services.....	43
4.3.12	Removing SUID/SGID from binaries.....	47
4.3.13	LIDS	50
4.3.14	Tripwire.....	61
4.3.15	BIOS.....	63
5	Maintenance.....	65
5.1	Upgrades.....	65
5.1.1	Disabling LIDS on a tty	65
5.1.2	Red Hat Packages	65
5.1.3	Snort.....	66
5.1.4	LIDS	66
5.1.5	Tripwire.....	67
5.2	Keeping Informed.....	67
5.2.1	Reading the root email	67
5.2.2	Sysstat.....	67
5.2.3	External sources.....	67
6	Verification.....	68
6.1	Verification of Applications.....	68
6.1.1	SSH.....	68
6.1.2	Bind / DNS.....	68
6.1.3	Email	69
6.1.4	Apache.....	71
6.1.5	Syslog.....	72
6.1.6	Snort.....	72
6.1.7	Netfilter/IPTables	72
6.1.8	TCP Wrappers	72
6.1.9	Unneeded Services	73
6.1.10	Set User ID and Set Group ID binaries.....	73
6.1.11	LIDS	73
6.2	Verification of Technical Requirements	74
6.3	Verification of High Level Requirements.....	74
6.4	Other Concerns	74
6.4.1	Backups.....	74
6.4.2	Redundancy	75
7	References	76
8	Files.....	78
8.1	Kickstart	78
8.2	makechrootsendmail.sh	80
8.3	makechrootpop.sh.....	83
8.4	/etc/sysconf/iptables.....	84

8.5	/root/LIDS/createlids.sh	85
8.6	checkupdates.pl.....	92

© SANS Institute 2000 - 2002, Author retains full rights.

1 Introduction

1.1 Project

This paper is submitted as the practical for the GIAC Certified UNIX Security Administrator (GCUX) Practical Assignment Version 1.9. The author of this paper is Mark Oram, and it was submitted on May 20th, 2002.

1.2 Audience

This document has been written for someone who has an understanding of the basic fundamentals of both the Unix/Linux operating system as well as the Internet and the higher level protocols which make up the Internet today.

Some areas the reader is expected to be familiar with:

- Basic Unix
- Unix Commands
- Unix System Administration
- Internet Services on Unix

1.3 Using this Document

The purpose of this document is to be both informative and instructional. The author has included dialog from performing the instructions whenever possible.

Any dialog or file, is in a box with a gray background like:

```
Dialog Box
In a dialog box:
Italics are used for comments.
Plain text is used for text from the system.
Bold is used for user input.
```

Throughout this paper there are several times that passwords are required. In most cases the password has been omitted and something like “<enter password here>” has been inserted in place of the password. It is strongly urged that someone following these directions use a strong password and remember it.

2 Description

2.1 Goal

The goal of this system is to provide a company with the Internet based services they require. Some background on the company is that it is a small company (5 or less employees), who work from various locations (remote from the server). They are looking to host their web site and email on this server. As they are a small company they would like to minimize the amount of system administration that the system requires.

2.2 Requirements

Security is and always will be a trade off. For instance one would not store a useless artifact in Fort Knox, also one would not store the gold in Fort Knox in the same place that one would store a useless artifact. Do to this reason it makes sense to start with a high level goal. Something easy to understand and non-technical which balances all of the requirements of the system. It is then easier to move from that to technical requirements which are easy to implement.

2.2.1 High Level / Executive

- Cost
 - Hardware – Should run on a standard inexpensive PC.
 - Software – Should use Open Source / Freeware when available.
 - Man Power – Cost of installing and maintaining should be kept to a minimum.
- Functionality
 - Services – should provide common services and be a common enough architecture that future expansion should be easily accommodated.
- Security
 - The system should be as secure as is reasonably practical.
 - No user authentication data should be sent in the plain. (non-encrypted)
 - The system should not allow for unauthorized users to utilize any resources that are not deemed public.

2.2.2 Technical

The above high level requirements were then adapted into the following more technical and specific requirements:

- Red Hat Linux 7.3 will be the base Operating System. This fulfills the requirement of using Open Source (minimizing the cost of the software) as well as being able to use standard PC hardware (minimizing the cost of the hardware).
- Whenever possible already existing packaged software will be used instead of manually compiling and maintaining source trees. This will fulfill the

requirement of man power, both during the installation process as well as supporting the machine into the future.

- The server will run the following services externally available:
 - DNS (bind) – For providing name service for the company's domain(s).
 - HTTP (Apache) – For running the public web server.
 - HTTPS (Apache+SSL) – For providing a 'secure' (encrypted) web server.
 - SSH (OpenSSH) – For remote access to the system, this is how remote administration will be performed.
 - POP+SSL – This is how remote email will be retrieved by the users of the company.
 - SMTP (Sendmail) – This is how mail will get delivered to the system from the external world. It will be configured to only allow email to be delivered locally.
 - SMTP+SSL (Sendmail + SSL + Auth) – This is how the users of the system will send email to the outside world. It will allow for the users (who will have to authenticate) to send email externally.
- Each service that can easily support being chrooted will be run in an isolated chrooted environment so that the compromise of one service will not necessarily mean the compromise of the entire system.
- The Server will implement some sort of MAC (Mandatory Access Control).
- The capabilities of the system (the privileges normally granted to root) will be restricted and explicitly granted to those processes that require them.

2.3 Hardware

The hardware used for this system is:

- Dell Dimension V400
- Pentium 2 CPU running at 400 megahertz
- 256 megabytes of RAM
- One 13.677 gigabyte IBM Hard Drive
- ATI Mach64 Video Controller
- 3Com 3c590 10/100 Ethernet Controller
- CRD-8400 CD/DVD-ROM drive
- Standard 1.44mb Floppy Disk Drive
- 1 Parallel Port
- 1 Serial Port
- 2 USB Ports
- Dell ps/2 Keyboard
- Dell M780 Monitor

2.4 Software

2.4.1 Base Red Hat Linux 7.3 Operating System

The base Operating System will be Red Hat 7.3 (<http://www.redhat.com/>) it can be purchased online directly from Red Hat at <http://www.redhat.com/apps/commerce/> or from a local store. If one is familiar with downloading CD images and burning them, the software is freely available, and the disc images can be located at <ftp://pub.redhat/linux/7.3/en/iso/i386/> and a list of mirror sites can be found at <http://www.redhat.com/download/mirror.html> however the scope of doing this is outside of this document.

Some of the notable components of Red Hat Linux 7.3 which will be utilized are:

Software	Version	Use
Apache	1.3.23	Provides web (HTTP) and secure web (HTTPS) services.
BIND	9.2.0	Provides name resolution service (DNS).
OpenSSH	3.1p1	Provides an encrypted method of accessing the server.
POP	v2001.78rh	Post Office Protocol – For users to receive email.
Sendmail	8.11.6	For receiving email via SMTP to local users and delivering from local users to remote users.
Stunnel	3.22	A program to encrypt plaintext communications via Secure Socket Layer (SSL).
Tripwire	2.3.1	A Host based Intrusion Detection System (IDS).
Xinetd	2.3.4	Extended Inetd – used to start programs as Internet services.

2.4.2 Updates to the Red Hat Linux 7.3 Operating System

As with any maintained operating system there are patches and updates available for Red Hat Linux. The updates for 7.3 are available from Red Hat at <http://updates.redhat.com/7.3/en/os/>.

2.4.3 Third party Software

2.4.3.1 Linux Intrusion Detection (LIDS)

LIDS is a security enhancement to the Linux kernel. It provides support for Mandatory Access Controls (MAC) and limiting the capabilities that are granted to a process.

In the typical UNIX environment, the root user has full access to everything, so any process running with root's privileges also has full access to anything. In reality there are specific capabilities which root has by default. These capabilities include being able to change their user id, chown files, override file permissions, bind to privileged ports, etc. (For a full list of the capabilities and what they mean refer to the file /etc/lids/lids.cap

after installing LIDS.) LIDS also allows for supporting MACs which can be used to further limit the accessing of specific files and/or directories to specific programs. The power of this comes from the ability to limit the potential damage that can be done through a compromised service running as root.

The website for LIDS is located at www.lids.org.

2.4.3.2 Snort

Snort is an open sourced network intrusion detection system. It 'sniffs' the network interface and monitors it (via a list of rules) for activity which is suspected of being the signature of an attack.

Having a network based IDS can be helpful constantly running, used to notify the administrator about attacks or as a diagnostic tool when tracking down network activity.

The website for Snort is located at www.snort.org

© SANS Institute 2000 - 2002, Author retains full rights.

3 Risk Analysis

3.1 High Level

This server will be directly connected to the Internet and will need to be able to communicate with any other host on the Internet. These facts place this server in a high risk category. Reviewing the traditional CIA triad of Information Security:

- Confidentiality – All services should be configured so as to best deny unauthorized access to any data or part of the system.
- Integrity – There should be a method of detecting if the integrity of any files have been compromised.
- Availability – This is the least focused on in this document. At the end of this document there is a short mention of some additional precautions that should be taken (outside of the scope of this machine and document) to provide fall back services for DNS and SMTP.)

This machine will be a target of convenience for ‘script kiddies’ scanning for common services (since it is running many of the common services). It will be a target for anyone focusing on the domain in particular that this server is hosting (since the obvious records will resolve to it (www, mail, etc.), as well as any malicious Internet user focusing specifically on this company. (Be it a disgruntled employee, previous employee, or simply someone with a grudge.)

3.2 Exposed Applications

3.2.1 Apache

Apache will need to accept connections on port 80 (http) and port 443 (https), both of these are ‘privileged’ ports. It can change it’s user ID to be run as a non-root user (user apache) after binding to the ports, so that in the event of a vulnerability within the software, the exploitation will not have root privileges. Recent vulnerabilities associated with Apache are actually external and not part of Apache itself. (Modules, and CGI-bins, for example the recent PHP bug <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0081>) Knowing this, it seems logical to disable any unneeded functionality until such time as that functionality is specifically needed.

3.2.2 Bind (DNS)

Bind will need to accept traffic on port 53 which is a ‘privileged’ port. Bind has had numerous security vulnerabilities. There have been three main versions 4.x, 8.x and 9.x. The 9.x branch was a complete rewrite, partially to address the numerous security problems which had been discovered in the previous versions. Bind 9.x internally supports chrooting (restricting itself to a subset of the filesystem) and setting its user ID to a non-root user after binding to port 53.

3.2.3 OpenSSH

OpenSSH will need to accept traffic on port 22. OpenSSH has also had several vulnerabilities recently. Since it is going to be used to remotely administer the machine, there is not a whole lot that can be done to further limit what the program has access to.

3.2.4 Sendmail

Sendmail is the program used to deliver and receive email. It has a history of security problems as well. To minimize the impact from any future vulnerabilities, sendmail can be run in a chrooted environment. Since breaking out of a chrooted environment is not difficult if root permissions can be obtained, the 'root capabilities' will be removed from all programs within the chrooted environment. To minimize having anyone use this service to relay spam, any mail delivered to a remote system through this, should require that the user authenticates before the mail can be delivered. To address confidentiality of the authentication information, sendmail will be available through a SSL tunnel.

3.2.5 POP

POP or Post Office Protocol is the method that will be used for the users of the machine to retrieve their email from the server. To address the concern of the confidentiality of the authentication information, POPS will be available which is POP with SSL encryption.

3.3 Other Concerns

3.3.1 Binding of Ports

It is common for either a vulnerability exploit or an attacker to try to bind a program to a port on the system for future communication. Netfilter/Iptables will be used to explicitly deny all network access other than that explicitly allowed. With this alone, an attacker could still kill off a service which is explicitly allowed and run their software binding to that port. To stop this, using LIDS, the kernel will further restrict which program can bind to what port.

3.3.2 Modification of files

Tripwire will be installed to monitor the integrity of key files installed on the system.

3.3.3 Detection of Attacks

Snort will be installed as a network based intrusion detection system. This can be either run continuously providing alerts, or simply installed ready to use when trying to track down what is happening on the network.

3.3.4 Users

This system has a set set of purposes. (Providing DNS for the domain, receiving and delivering email, and hosting the website.) Due to this, and the desire to keep the

machine as secure as reasonably possible, the machine should not be used for any other purposes, and therefore should have no other users, other than the administrator.

© SANS Institute 2000 - 2002, Author retains full rights.

4 Installation

4.1 Base Red Hat Linux 7.3 Installation

Only follow one of the two methods for installing the base OS (either Manual or Automatic.) The manual method has step by step directions for installing the base operating system as one would normally install it based off of the Installation Manual. The automatic method employs a method called 'kickstart' which is used for an automated install without prompting the user for any information. (Similar to Sun's Jumpstart for their Solaris operating system.)

4.1.1 Manual Method

For this method we are going to do an interactive install from the CDs.

4.1.1.1 Booting

Insert the CD labeled "Installation CD 1 of 3" into the CD-ROM drive, and power up the machine.

4.1.1.2 Welcome to Red Hat Linux 7.3!

The machine should boot to a text based screen with the title "Welcome to Red Hat Linux 7.3!" At the prompt, simply press <Enter>, after this, the kernel will load and you will see the:

4.1.1.3 Graphical Welcome screen.

There is nothing to be done at this screen, simply click on "Next" to proceed to the next step.

4.1.1.4 "Language Selection" screen.

The language selection screen defaults to "English", click on the "Next" button to proceed.

4.1.1.5 "Keyboard Configuration" screen.

The defaults for this screen are:

- Model: Generic 105-key (Intl) PC
- Layout: U.S. English
- Dead keys: Enable dead keys

These are the correct values, so click the "Next" button to continue.

4.1.1.6 "Mouse Configuration" screen.

The default value for this is "3 button PS/2" this should be changed to "2 button PS/2", after making this change, click on the "Next" button.

4.1.1.7 Installation Type

At this point the user is prompted if they are installing or upgrading an existing system. Also, the user can select a specific type of install (Server, Workstation, Custom, etc.) A Custom Install should be selected, and then click on "Next."

4.1.1.8 Disk Partitioning Setup

The user is presented with three choices, to auto partition the disk(s), to use Disk Druid, or to use fdisk. Select fdisk and click on "Next."

4.1.1.9 Fdisk

Click on "hda" at which point the fdisk program will load in that window. Seven usable partitions are going to be made on this drive according to the following table:

Device	Mount Point	Size (MB)	Type
/dev/hda1	/boot	50MB	Ext3
/dev/hda5	/	2048MB	Ext3
/dev/hda6	/usr	3072MB	Ext3
/dev/hda7	/var	4096MB	Ext3
/dev/hda8	/tmp	512MB	Ext3
/dev/hda9	*swap*	512MB	Swap (82)
/dev/hda10	/home	Remaining space	Ext3

Here is a log of the fdisk session:

```
The number of cylinders for this disk is set to 1662.
There is nothing wrong with that, but this is larger then 1024,
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions of LILO)
2) booting and partitioning software from other OSs
   (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p

Disk /tmp/hda: 255 heads, 63 sectors, 1662 cylinders
Units = cylinders of 16065 * 512 bytes

   Device Boot   Start     End   Blocks   ID  System
Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
p
Partition number (1-4): 1
First cylinder (1-1662, default 1): 1
Last cylinder or +size or +sizeM or +sizeK (1-1662, default 1662): +50M

Command (m for help): n
Command action
  e   extended
  p   primary partition (1-4)
e
Partition number (1-4): 2
First cylinder (8-1662, default 8): <enter>
Using default value 8
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): <enter>
```

```

Using default value 1662

Command (m for help): n
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (8-1662, default 8): <enter>
Using default value 8
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): +1024M

Command (m for help): n
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (8-1662, default 139): <enter>
Using default value 139
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): +3072M

Command (m for help): n
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (8-1662, default 531): <enter>
Using default value 531
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): +4096M

Command (m for help): n
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (8-1662, default 1054): <enter>
Using default value 1054
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): +512M

Command (m for help): n
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (8-1662, default 1120): <enter>
Using default value 1120
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): +512M

Command (m for help): t
Partition number (1-9): 9
Hex code (type L to list codes): 82
Changed system type of partition 9 to 82 (Linux swap)

Command (m for help): n
  l   logical (5 or over)
  p   primary partition (1-4)
l
First cylinder (8-1662, default 1186): <enter>
Using default value 1186
Last cylinder or +size or +sizeM or +sizeK (8-1662, default 1662): <enter>
Using default value 1662

Command (m for help): p

Disk /tmp/hda: 255 heads, 63 sectors, 1662 cylinders
Units = cylinders of 16065 * 512 bytes


```

Device	Boot	Start	End	Blocks	ID	System
/tmp/hda1		1	7	56196	83	Linux
/tmp/hda2		8	1662	13293787+	5	Extended
/tmp/hda5		8	138	1052226	83	Linux
/tmp/hda6		139	530	3148708+	83	Linux
/tmp/hda7		531	1053	4200966	83	Linux
/tmp/hda8		1054	1119	530113+	83	Linux
/tmp/hda9		1120	1185	530113+	82	Linux swap
/tmp/hda10		1186	1662	3831471	83	Linux


```
Command (m for help): w
The partition table has been altered!
```

Control is returned back to the installation GUI. Click “Next”, an alert window will pop up saying:

/dev/hda9 has a partition type of 0x82 (Linux swap) but does not appear to be formatted as a Linux swap partition.

Would you like to format this partition as a swap partition?

Click on “Yes” and /dev/hda9 will become formatted as swap and used as swap space for the rest of the installation process.

4.1.1.10 Disk Setup

The partitions created in step 4.1.1.9 now need to be configured. Referring to the table in 4.1.1.9 which specifies the mount point for each partition, go through each partition other than the swap partition and:

- Select “Format As”
- Set the “Format As” type to “ext3”
- Check the “Check for bad blocks?” check box.

After completing this, the screen should look like:

Device	Start	End	Size (MB)	Type	Mount Point	Format
/dev/hda						
-/dev/hda1	1	7	55	ext3	/boot	Yes
-/dev/hda2	8	1662	12982	Extended		
-/dev/hda5	8	138	1028	ext3	/	Yes
-/dev/hda6	139	530	3075	ext3	/usr	Yes
-/dev/hda7	531	1053	4103	ext3	/var	Yes
-/dev/hda8	1054	1119	518	ext3	/tmp	Yes
-/dev/hda9	1120	1185	518	swap	swap	Yes
-/dev/hda10	1186	1662	3742	ext3	/home	Yes

Click on “Next” to continue. An alert box will pop-up saying:

The following, pre-existing partitions have been Selected to be formatted, destroying all data

/dev/hda1	ext3	/boot
/dev/hda10	ext3	/home
/dev/hda5	ext3	/
/dev/hda6	ext3	/usr
/dev/hda7	ext3	/var
/dev/hda8	ext3	/tmp
/dev/hda9	swap	

Select 'Yes' to continue and format these partitions, or 'No' to go back and change these settings.

Click "Yes."

4.1.1.11 Boot Loader Configuration

Use the defaults for this screen. These defaults should be:

- "Use GRUB as the boot loader" should be selected
- Install Boot Loader record on: **/dev/hda Master Boot Record (MBR)**
- Partition: **/dev/hda5**
- "Default Boot Image" should be checked.
- Boot Label: **Red Hat Linux**

Click "Next."

4.1.1.12 Boot Loader Password Configuration

Since this machine will be located at an ISP, it is prudent to ensure that someone can not come in and override or change the behavior of the GRUB boot loader, so we will assign a password. Check the box labeled "Use a GRUB password?", and enter the password, and confirm it. If the passwords match, it should say "Password Accepted" at which point you should click on "Next."

4.1.1.13 Network Configuration

First, uncheck the "Configure using DHCP" option, at which point the specific settings (IP, Netmask, etc.) will become 'un-grayed' and able to accept values. The following are the values used, obviously one should use the values which fit their own network best:

IP Address	192.168.17.220
Netmask	255.255.255.0
Network	192.168.17.0

Broadcast	192.168.17.255
Hostname	newserver.markoram.com
Gateway	192.168.17.1
Primary DNS	127.0.0.1
Secondary DNS	192.168.17.100
Ternary DNS	

Note that primary DNS is specified as 127.0.0.1, this is this machine, contacting it via the loopback network device. This is entered because this machine will be configured as a DNS server after all of the steps are completed.

Click on “Next.”

4.1.1.14 Firewall Configuration

Select “No Firewall.” This will be configured by hand separately after the base operating system has been installed.

4.1.1.15 Additional Language Support

Click “Next.” (After adding any additional language support desired.)

4.1.1.16 Time Zone Selection

Set the time zone to the appropriate one for where your server will reside, and click “Next.”

4.1.1.17 Account Configuration

Enter and confirm the root password, since this machine will not have any shell users other than root to do the occasional maintenance, no other accounts should be created. Click “Next” to proceed.

4.1.1.18 Authentication Configuration

The defaults should be that “enable MD5 passwords” and “enable shadow passwords” are both selected. After making sure that both are selected, click “Next.”

4.1.1.19 Package Group Selection

Set it so that only the following set of groups are selected:

- Network support
- Anonymous FTP server
- SQL Database server
- Web Server

- Router / firewall
- DNS name server
- Utilities
- Software Development
- Kernel Development
- Network Managed Workstation

Then, click “Next.”

4.1.1.20 About To Install

There is nothing to be done at the screen, simply click “Next” to proceed with the install.

4.1.1.21 Installing Packages

First the partitions will be formatted, then the packages will be installed. During the package installation portion of this there is a status bar which tries to estimate how much time is remaining. Part way through the process, the user will be prompted to removed disc1 and insert disc2, when prompted do so, and click on “OK.”

4.1.1.22 Boot Disk Creation

Insert a blank floppy diskette and click on “Next.”

4.1.1.23 “Congratulations” screen

Remove the newly created boot disk, and the CD, and click on “Exit.” The machine will then reboot. This concludes step 4.1.

4.1.2 Automatic Method

Red Hat Linux supports a method of automatically installing. This method is known as ‘kickstart.’ By using this method the installer can not only make it so that they do not need to sit through the install session (and answer individual questions) but can also guarantee that if they are installing multiple machines that they are installing them similarly or can ensure that re-installing a machine can go quickly.

For detailed information about the kickstart installation process please refer to the kickstart chapter of the Red Hat customization manual. This can be found online at <http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/custom-guide/ch-kickstart2.html>.

The kickstart file to install this machine as the manual install did is located in File chapter. Be sure to change any passwords within the kickstart file before using it. To install the system this way, copy the kickstart file to a floppy disk and call the file ks.cfg,

boot the system from the first CD, and at the text “Welcome to Red Hat Linux 7.3!” screen, enter “linux ks=floppy” The system will then go through the process of installing, using the parameters specified in the kickstart file.

For someone maintaining multiple systems, one might want to consider maintaining a network available install area, and doing the install via either an NFS mount, HTTP, or FTP instead of CD based. One great advantage of this is that the admin can install the patches to the install area, so that the updated versions of software are automatically installed with the base operating system. While this is outside of the scope of installing this one system, some pointers to get one started are the Red Hat Customization Guide, and the anaconda package.

4.2 Applying updates to Red Hat Linux 7.3

At the time of this installation, Red Hat 7.3 had just gotten released, and there were no updates available. Any updates available should be installed as prescribed in the Maintenance chapter.

© SANS Institute 2000 - 2002, Author retains full rights.

4.3 Configuring The System

4.3.1 OpenSSH

4.3.1.1 Background

OpenSSH is run to provide an encrypted method of remotely accessing the server and administering it. For detailed information about OpenSSH their website is: <http://www.openssh.org/>. To access the server remotely one will need an SSH client installed on their desktop. If the desktop is a Linux machine, the OpenSSH client program (/usr/bin/ssh) can be used, if a Windows SSH client is needed one can be purchased from Data Fellows (<http://www.datafellows.com/>), information about it can be found at <http://www.datafellows.com/products/ssh/> and it can be purchased or a trial version obtained at <http://www.datafellows.com/download-purchase/>.

4.3.1.2 Configuring the daemon (server)

There are several options we want to disable. All of these options are controlled by the file “/etc/ssh/sshd_config”

- Protocol – By default SSH (the program) allows both versions 1 and 2 of the SSH protocol. Uncomment the line “#Protocol 2,1” by removing the leading ‘#’ and change it to say “Protocol 2” this disables Protocol version 1 of SSH, so that the server will only communicate using the more secure version 2.
- PasswordAuthentication – By default OpenSSH allows for two methods of authentication, one is a key based method where the user stores their public key on the server and logs in authenticating themselves using their private key, the other is ‘password authentication’ where the user simply provides their username and password. While the password authentication can be more convenient (it is more portable since a user does not have to have their private key with them) it is also less secure since it only requires knowledge of the username and password where as the key method requires possessing the private key and knowing the passphrase which protects the private key. To disable password authentication, locate the line “#PasswordAuthentication yes”, uncomment it by removing the leading ‘#’ and change the ‘yes’ to a ‘no’ so that it reads “PasswordAuthentication no”
- X11Forwarding – OpenSSH comes with a method of tunneling Xwindows windows through the SSH protocol. This should be disabled, find the line “X11Forwarding yes” and delete that line.

After making these changes we should restart the SSH daemon and verify that it starts correctly:

```

#Start the SSH daemon
[root@newserver ~]# /etc/rc.d/init.d/sshd restart
Stopping sshd: [ OK ]
Starting sshd: [ OK ]
#Verify that SSHd will start at the appropriate run-levels
[root@newserver ~]# /sbin/chkconfig --list sshd
sshd    0:off  1:off  2:on   3:on   4:on   5:on   6:off

```

If not, use the command “chkconfig --level 0123456 sshd off” to turn it off for all run levels and then the command “chkconfig --level 2345 sshd on” to enable the service for runlevels 2, 3, 4, and 5.

4.3.1.3 Installing the keys

When installing the client software, the user will be walked through the process of generating a public/private key. The public key should be placed in the file `authorized_keys` within the `.ssh` directory. For example, to allow one to log in as root, their public key should be placed in the file `/root/.ssh/authorized_keys`.

4.3.2 DNS / Bind

4.3.2.1 Background

Domain Name Service (DNS) is the standard protocol used for resolving hostnames to IP addresses and vice versa on the Internet. One of the most popular DNS servers is known as Bind and is maintained by the Internet Software Consortium (<http://www.isc.org/>) The website for Bind is located at <http://www.isc.org/products/BIND/>.

The Bind software has had several vulnerabilities there are 3 major versions, the 4.x, 8.x, and 9.x versions. A list of some of the vulnerabilities in the previous Bind software can be located at <http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=bind>. Since the DNS server needs to be and will be publicly available as it will be listed as the primary name server for any domains the company hosts, there are several simple precautions that should be taken. These precautions include running it in a chrooted environment and running it as a non-root user, both of which are easily accomplished.

DNS utilizes port 53, and can use both UDP and TCP. UDP is the standard protocol for lookups, however DNS over TCP is used for some of the more modern/advanced functions. On a UNIX system all ports under 1024 are considered ‘privileged’ and require root permissions to be bound to.

4.3.2.2 Chrooting

The idea of chrooting the environment is that the process will run in is to limit the damage that can be caused if a user can exploit the process and execute instructions through it. The bind/named software already supports chrooting, so this is straight forward to accomplish.

4.3.2.2.1 *Creating the top level directory*

We differentiate this from the root of the chrooted directory so that we can be strict with the directory permission (keeping out other users) but still have 'nice' permissions on the root of the chrooted environment. This is done by issuing the command:

```
#Create the directory which will hold the chrooted root directory
[root@newsrver /]# mkdir /home/named/

#Set strict permissions on the directory above the 'chrooted root'
[root@newsrver /]# chmod 700 /home/named
```

4.3.2.2.2 *Creating the root of the chrooted environment*

This is the root directory of the chrooted environment -- what the process running will see as '/'. We make this directory:

```
#Make the root directory of the chrooted environment
[root@newsrver /]# mkdir /home/named/root/

#and set its permissions similar to the real '/' directory:
[root@newsrver /]# chmod 755 /home/named/root
```

4.3.2.2.3 *Creating the sub-directories*

The bind program (named) requires a minimal amount of libraries and files to exist in it's chrooted environment. The following steps create the needed directories and set the permissions:

```
#Make the top level directories.
[root@newsrver /]# mkdir /home/named/root/{dev,etc,lib,var}
[root@newsrver /]# mkdir -p /home/named/root/var/{named,run/named}

#The named process will need to write out it's pid file so it needs ownership of the run
#directory
[root@newsrver /]# chown named.named /home/named/root/var/run/named

#Copy the default configuration file over to the chrooted environment.
[root@newsrver /]# cp -p /etc/named.conf /home/named/root/etc/

#Copy the default zone files (cache and local) to the chrooted environment.
[root@newsrver /]# cp -p /var/named/* /home/named/root/var/named/
#Copy the configuration file for the BIND 9 name server control utility.
[root@newsrver /]# cp -p /etc/rndc.* /home/named/root/etc/

#Create the /dev/random entry
[root@newsrver /]# mknod /home/named/root/dev/random c 1 8
#Make the dev null device
[root@newsrver /]# mknod /home/named/root/dev/null c 1 3

#and to copy the needed files into these directories
[root@newsrver /]# cp /etc/named.conf /home/named/root/etc
```


4.3.2.3 Configuring

4.3.2.3.1 /etc/sysconfig/named

Most services in Red Hat Linux use a configuration file located in /etc/sysconfig. The configuration file for the named program is called /etc/sysconfig/named. This file should be edited and the following line should be inserted:

```
ROOTDIR="/home/named/root"
```

4.3.2.3.2 Creating and configuring a zone

For the purpose of this document we are going to configure this server to resolve and handle markoram.com.

4.3.2.3.2.1 named.conf

In the file /home/named/root/etc/named.conf add the following block of text:

```
zone "markoram.com." IN {
    type master;
    file "db.markoram.com";
    allow-update { none; };
};
```

4.3.2.3.2.2 Creating the zone file (db.markoram.com)

In the directory /home/named/root/var/named create a file named db.markoram.com with the following contents:

```
$TTL      86400
@         IN      SOA     markoram.com. root.markoram.com. (
                                1997022700    ; Serial
                                28800         ; Refresh
                                14400         ; Retry
                                3600000      ; Expire
                                86400      )
                                IN      NS      ns.markoram.com.

                                86400      IN      A      192.168.17.220
NS        86400      IN      A      192.168.17.220
www       86400      IN      A      192.168.17.220
mail     86400      IN      A      192.168.17.220
newsrver  86400      IN      A      192.168.220
```

/home/named/root/var/named/db.markoram.com

4.3.2.3.3 Starting and testing named

We then start the service by issuing the command:

```
[root@newsrver /]# /etc/rc.d/init.d/named start
Starting named: [ OK ]
```

Verify that we get valid results by specifying a lookup over the loopback device:

```
[root@newsrerver /]# nslookup -sil www.markoram.com 127.0.0.1
Server:                127.0.0.1
Address:               127.0.0.1#53

Non-authoritative answer:
Name:   www.markoram.com
Address: 209.123.128.120

#It returned valid results, so we now check that it is set to start automatically
[root@newsrerver /]# /sbin/chkconfig --list named
named          0:off  1:off  2:off  3:off  4:off  5:off  6:off
#We see that it is not set to run at all, so we set it to run at runlevels 2,3,4, and 5
[root@newsrerver /]# /sbin/chkconfig --level 2,3,4,5 named on
```

4.3.3 SMTP / Sendmail

4.3.3.1 Background

Simple Mail Transport Protocol (SMTP) is the protocol generally used to deliver email on the Internet. One of the most common applications that implements SMTP is known as sendmail. Sendmail is packaged with the Red Hat Linux operating system, and the main website for the application is located at <http://www.sendmail.org/>.

Sendmail has had a long history of bugs and vulnerabilities, both local and remote in nature. To see some of these exploits a simple search at cve.mitre.org (<http://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=sendmail>) will return some information about them.

4.3.3.2 Chrooting and Installing

Since SMTP binds to port 25, and needs to be able to write the received email out to any users mail box, the process needs to be running with root's permissions. To help minimize the risk of running an application like this as root, we will run it in a chrooted environment.

Since sendmail does not internally support the concept of chrooting (like named does for instance) all of the necessary files must be kept into the chrooted environment. To keep in the spirit of minimizing future administrative time and sticking to prepackaged software whenever possible we will use the "--root" option to the *rpm* program to re-install the needed packages into the chrooted environment. This will allow for easily upgrading the packages in the future when updates to software components are made. It also will leave the environment better situated for 3rd party support if the need should ever arise.

The following set of steps can be followed, or for convenience, a script `makechrootsendmail.sh` can be found in the chapter titled “Files.”

4.3.3.2.1 Creating the top level directory

We differentiate this from the root of the chrooted directory so that we can be ‘strict’ with the directory permissions (keeping out other users) but still have ‘nice’ permissions on the root of the chrooted environment. This is done by issuing the command:

```
#First, we make the directory
[root@newsrver /]# mkdir /home/EMAIL
#Then we shutoff permissions to all other users/groups to it
[root@newsrver /]# chmod 700 /home/EMAIL
```

4.3.3.2.2 Creating the root of the chrooted environment

This is the root of the chrooted environment – what the processing running will see as ‘/’.

```
#Create the directory
[root@newsrver /]# mkdir /home/EMAIL/root
#Set the permissions similar to how the true '/' directory is.
[root@newsrver /]# chmod 755 /home/EMAIL/root
```

4.3.3.2.3 Installing the packages

```
[root@newsrver RPMS]# mkdir -p /home/EMAIL/root/var/lib/rpm
[root@newsrver RPMS]# mkdir /home/EMAIL/root/dev/
[root@newsrver RPMS]# mknod /home/EMAIL/root/dev/null c 1 3
[root@newsrver RPMS]# mount /dev/cdrom /mnt/cdrom
mount: block device /dev/cdrom is write-protected, mounting read-only
[root@newsrver RPMS]# cd /mnt/cdrom/RPMS
[root@newsrver RPMS]# rpm -ivh setup-2.5.12-1.noarch.rpm --root /home/EMAIL/root
Preparing...
setup
[root@newsrver RPMS]# rpm -ivh filesystem-2.1.6-2.noarch.rpm --root /home/EMAIL/root
Preparing...
filesystem
[root@newsrver RPMS]# rpm -ivh basesystem-7.0-2.noarch.rpm --root /home/EMAIL/root
Preparing...
[root@newsrver RPMS]# rpm -ivh glibc-common-2.2.5-34.i386.rpm glibc-2.2.5-34.i686.rpm --
root /home/EMAIL/root
Preparing...
glibc-common
glibc
[root@newsrver RPMS]# rpm -ivh db3-3.3.11-6.i386.rpm --root /home/EMAIL/root
Preparing...
db3
[root@newsrver RPMS]# rpm -ivh mktemp-1.5-14.i386.rpm --root /home/EMAIL/root
Preparing...
mktemp
[root@newsrver RPMS]# rpm -ivh termcap-11.0.1-10.noarch.rpm libtermcap-2.0.8-28.i386.rpm
--root /home/EMAIL/root
Preparing...
termcap
libtermcap
```

```

[root@newsrver RPMS]# rpm -ivh bash-2.05a-13.i386.rpm --root /home/EMAIL/root
Preparing... #####
bash
[root@newsrver RPMS]# rpm -ivh words-2-18.noarch.rpm --root /home/EMAIL/root
Preparing... #####
words
[root@newsrver RPMS]# rpm -ivh cracklib-2.7-15.i386.rpm cracklib-dicts-2.7-15.i386.rpm -
-root /home/EMAIL/root
Preparing... #####
cracklib
cracklib-dicts
[root@newsrver RPMS]# rpm -ivh info-4.1-1.i386.rpm --root /home/EMAIL/root --nodeps
Preparing... #####
info
[root@newsrver RPMS]# rpm -ivh pcre-3.9-2.i386.rpm --root /home/EMAIL/root
Preparing... #####
pcre
[root@newsrver RPMS]# rpm -ivh grep-2.5.1-1.i386.rpm --root /home/EMAIL/root
Preparing... #####
grep
[root@newsrver RPMS]# rpm -ivh fileutils-4.1-10.i386.rpm --root /home/EMAIL/root
Preparing... #####
fileutils
[root@newsrver RPMS]# rpm -ivh textutils-2.0.21-1.i386.rpm --root /home/EMAIL/root
Preparing... #####
textutils
[root@newsrver RPMS]# rpm -ivh sed-3.02-11.i386.rpm --root /home/EMAIL/root
Preparing... #####
sed
[root@newsrver RPMS]# rpm -ivh glib-1.2.10-5.i386.rpm --root /home/EMAIL/root
Preparing... #####
glib
[root@newsrver RPMS]# rpm -ivh initscripts-6.67-1.i386.rpm --root /home/EMAIL/root --
nodeps --noscripts
Preparing... #####
warning: group utmp does not exist - using root
warning: group utmp does not exist - using root
initscripts
[root@newsrver RPMS]# rpm -ivh pam-0.75-32.i386.rpm --root /home/EMAIL/root
Preparing... #####
pam
[root@newsrver RPMS]# rpm -ivh gdbm-1.8.0-14.i386.rpm --root /home/EMAIL/root
Preparing... #####
gdbm
[root@newsrver RPMS]# rpm -ivh openssl-0.9.6b-18.i686.rpm --root /home/EMAIL/root
Preparing... #####
openssl
[root@newsrver RPMS]# rpm -ivh cyrus-sasl-1.5.24-25.i386.rpm cyrus-sasl-md5-1.5.24-
25.i386.rpm cyrus-sasl-plain-1.5.24-25.i386.rpm --root /home/EMAIL/root
Preparing... #####
cyrus-sasl
cyrus-sasl-md5
cyrus-sasl-plain
[root@newsrver RPMS]# rpm -ivh procmail-3.22-5.i386.rpm --root /home/EMAIL/root
Preparing... #####
procmail
[root@newsrver RPMS]# rpm -ivh openldap-2.0.23-4.i386.rpm --root /home/EMAIL/root
Preparing... #####
openldap
[root@newsrver RPMS]# rpm -ivh gawk-3.1.0-4.i386.rpm --root /home/EMAIL/root
Preparing... #####
gawk
[root@newsrver RPMS]# rpm -ivh shadow-utils-20000902-7.i386.rpm --root /home/EMAIL/root
Preparing... #####
shadow-utils
[root@newsrver RPMS]# rpm -ivh chkconfig-1.3.5-3.i386.rpm --root /home/EMAIL/root
Preparing... #####
chkconfig
[root@newsrver RPMS]# rpm -ivh sh-utils-2.0.11-14.i386.rpm --root /home/EMAIL/root
Preparing... #####
sh-utils

```

```

[root@newsrserver RPMS]# rpm -ivh sendmail-8.11.6-15.i386.rpm --root /home/EMAIL/root
Preparing...      #####
sendmail          #####
[root@newsrserver RPMS]# cd /
[root@newsrserver RPMS]# umount /mnt/cdrom
[root@newsrserver RPMS]# eject

```

4.3.3.3 Configuring Sendmail

4.3.3.3.1 Modifying the configuration file

Sendmail uses a file called ‘sendmail.cf’ as it’s configuration file. This file resides in the etc directory. The running configuration file is actually assembled from a “sendmail macro config file” located in etc/mail and named sendmail.mc. The preferred method of maintaining the sendmail configuration is to make the changes in the macro file and then rebuild the runtime configuration file from the macro file.

By default, the sendmail which is shipped with Red Hat Linux 7.3 does not run in daemon mode and accept email from over the network. To change this edit /home/EMAIL/root/etc/mail/sendmail.mc and go to line 51, it is the line that reads “DAEMON_OPTIONS(‘Port=smtp,Addr=127.0.0.1, Name=MTA’)” and comment that out by inserting “dnl “ before that.

To stop spammers from relaying email through this sendmail, typically one restricts sendmail from relaying email from a specific list of client IP addresses. Since the users of this system may come from different IP addresses at different times this is not practical. Instead, there is a more advanced option which allows for the user to authenticate with their username and password. With this enabled, anybody can send email to the system to be delivered locally (to a [user@host](#) which it handles) and only authenticated users can use the system to send email to another host on the Internet.

To configure the sendmail mc file for authentication, locate the line “dnl TRUST_AUTH_MECH...” and uncomment that out by removing the leading “dnl “.

After making changes to the sendmail mc file, the runtime configuration file must be rebuilt. This can be accomplished by issuing the command:

```

#Create the run time config file from the macro config file using m4
[root@newsrserver /]# m4 /home/EMAIL/root/etc/mail/sendmail.mc >
/home/EMAIL/root/etc/sendmail.cf

```

4.3.3.3.2 Local-host-names

The file etc/mail/local-host-names controls what names sendmail should accept mail for. Any hostname which resolve to this IP should be placed in this file.

```
#Add all of the hostnames for this host to the local-host-names file.
[root@newsrver /]# cat >>/home/EMAIL/root/etc/email/local-host-names
markoram.com
newsrver.markoram.com
www.markoram.com
mail.markoram.com
^D
```

4.3.3.3.3 Modifying the startup script

Sendmail now needs to be configured so that it starts up in the chrooted environment. There are two steps to this.

4.3.3.3.3.1 /etc/sysconfig/sendmail

Add the following line to the file /etc/sysconfig/sendmail

```
CHROOTDIR="/home/EMAIL/root"
```

This can be done simply by:

```
[root@newsrver /]# cat >>/etc/sysconfig/sendmail
CHROOTDIR="/home/EMAIL/root"
^D
```

4.3.3.3.3.2 /etc/rc.d/init.d/sendmail

The 'startup' portion of the script should be modified to read as follows: (The modifications are in **bold**)

```
start() {
    # Start daemons.

    echo -n "Starting $prog: "
    chroot $CHROOTDIR /usr/bin/newaliases >/dev/null 2>&1
    if text -x $CHROOTDIR/usr/bin/make -a -f $CHROOTDIR/etc/mail/Makefile ; then
        make -C $CHROOTDIR/etc/mail -s
    else
        for I in virtusertable access domaintable mailertable ; do
            if [ -f $CHROOTDIR/etc/mail/$I ] ; then
                makemap hash $CHROOTDIR/etc/mail/$I < $CHROOTDIR/etc/mail/$I
            fi
        done
    fi
    daemon chroot $CHROOTDIR /usr/sbin/sendmail $([ "$DAEMON" = yes ] && echo -bd) \
        $([ -n "$QUEUE" ] && echo -q$QUEUE)

    RETVAL=$?
    echo
    [ $RETVAL -eq 0 ] && touch /var/lock/subsys/sendmail
    return $RETVAL
}
```

4.3.3.4 Adding users

To add users to be able to send and receive email, they need to have the appropriate entries in the passwd and shadow files. The typical method of doing this on a Linux machine is by using *useradd* and *groupadd* respectively. These commands will still work, if we chroot when running them. To add a user 'test' (and lets assign a password of 'J4ng0F3tt'):

```
#First we need to generate the crypted password
[root@newsrver /]# perl -e 'print crypt("J4ng0F3tt","AZ"),"\n";'
AZoAZO68g2Hak
#Add the group for the user
[root@newsrver /]# chroot /home/EMAIL/root /usr/sbin/groupadd testgroup
#Add the user, making his default group the group we just created and use the
#crypted password we just generated.
[root@newsrver /]# chroot /home/EMAIL/root /usr/sbin/useradd test -g testgroup -p
AZoAZO68g2Hak
```

4.3.3.5 Testing

To ensure that mail is being delivered properly, we will connect to the mail server manually and deliver a piece of email to the test user. To do this we will 'speak' the SMTP protocol to the sendmail application. For more detailed information on the SMTP protocol, refer to RFC 821 which can be located at <ftp://ftp.isi.edu/in-notes/rfc821.txt>.

4.3.3.5.1 SMTP

```
[root@newsrver root]# telnet 0 mail
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
220 newsrver.markoram.com ESMTP Sendmail 8.11.6/8.11.6; Thu, 9 May 2002 22:27:46 -0400
ehlo fromtest.com
250-newsrver.markoram.com Hello localhost [127.0.0.1], pleased to meet you
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-SIZE
250-DSN
250-ONEX
250-ETRN
250-XUSR
250 HELP
mail from:someuser@somehost.com
250 2.1.0 someuser@somehost.com... Sender ok
rcpt to:test@markoram.com
250 2.1.5 test@markoram.com... Recipient ok
data
354 Enter mail, end with "." on a line by itself
Testing
.
250 2.0.0 g4A2S1A18972 Message accepted for delivery
quit
221 2.0.0 newsrver.markoram.com closing connection
Connection closed by foreign host.
#Use the standard mail client to read test@markoram.com's mailbox.
[root@newsrver root]# mail -f /home/EMAIL/root/var/spool/mail/test
Mail version 8.1 6/6/93. Type ? for help.
"/home/EMAIL/root/var/spool/mail/test": 1 message 1 new
>N 1 someuser@somehost.com Thu May 9 22:30 12/403
& page 1
```

```
Message 1:
From someuser@somehost.com Thu May 9 22:30:59 2002
Date: Thu, 9 May 2002 22:30:55 -0400
From: someuser@somehost.com

testing2
testing2

& x
```

4.3.4 SMTPS (More Sendmail)

To configure SMTPS, sendmail can be tunneled through a program called *stunnel* which is part of the Red Hat Linux 7.3 installation. Stunnel provides SSL encryption to the remote endpoint (the client) and executes the specified program locally. To facilitate this we will configure this service in xinetd.

Place the following service file in `/etc/xinetd.d/smtps`:

```
# default: off
# description: SMTPS - sendmail tunneled through stunnel to provide SSL
#                encryption
#
service smtps
{
    socket_type    = stream
    wait          = no
    user          = root
    flags         = NAMEINARGS
    server        = /usr/sbin/stunnel
    server_args   = smtps -N smtps -n smtp -L /usr/sbin/chroot -- chroot
                 /home/EMAIL/root /usr/sbin/sendmail -bs -O AuthMechanisms=LOGIN
    log_on_success += USERID
    log_on_failure += USERID
    disable       = no
}
```

Then, restart xinetd:

```
[root@newsrver /]# /etc/rc.d/init.d/xinetd restart
Stopping xinetd: [ OK ]
Starting xinetd: [ OK ]
```

To test SMTPS we will perform the same test as for SMTP, except we will use the program *openssl* to connect to the service instead of telnet.

```
[root@newsrver root]# openssl s_client -connect 0:smtps
CONNECTED(00000003)
depth=0 /C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
verify error:num=18:self signed certificate
verify return:1
depth=0 /C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
verify return:1
---
Certificate chain
 0 s:/C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
  i:/C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
---
Server certificate
-----BEGIN CERTIFICATE-----
```



```

Message 2:
From: someuser@somehost.com Fri May 10 09:25:59 2002
Date: Fri, 10 May 2002 09:25:54 -0400
From: someuser@somehost.com

testing through smtps

& x

```

4.3.5 POP and POPS

4.3.5.1 Background

POP or the Post Office Protocol is a common protocol used for a client to retrieve their email from the server.

The POP program must run within the same environment as the sendmail program. (Since sendmail will receive email delivered from the Internet and record it to a local file and the POP client will connect to the POP server to retrieve this email, it is necessary for them to be accessing those same mail files.)

4.3.5.2 Installing

The POP3 daemon needs to be installed into the chrooted environment as well as in the 'real' system as it was not installed during the base OS install from above. To accomplish this, either follow the steps below, or run the "makechrootpop.sh" script included in the Files section:

```

#Insert the first CD and mount it.
[root@newsrver /]# mount /dev/cdrom /mnt/cdrom
#Install the required kerberos libraries to the chrooted environment
#(They were already installed in the main system.)
[root@newsrver /]# rpm -ivh /mnt/cdrom/RedHat/RPMS/krb5-libs-1.2.4-1.i386.rpm --root
/home/EMAIL/root
Preparing...                               ##### [100%]
1:                                          ##### [100%]
#unmount the CD
[root@newsrver /]# umount /mnt/cdrom
#eject the disc
[root@newsrver /]# eject
#Place disc2 of the Installation CDs into the drive and mount that
[root@newsrver /]# mount /dev/cdrom /mnt/cdrom
#Change our working directory to the RPMS directory on the CD
[root@newsrver /]# cd /mnt/cdrom/RPMS
#Install the package in the 'real' system
[root@newsrver RPMS]# rpm -ivh imap-2001a-10.i386.rpm
Preparing...                               ##### [100%]
1:                                          ##### [100%]
#Install the package to the chrooted environment
[root@newsrver RPMS]# rpm -ivh imap-2001a-10.i386.rpm --root /home/EMAIL/root
Preparing...                               ##### [100%]
1:                                          ##### [100%]
#change directory, unmount the CD and eject it
[root@newsrver RPMS]# cd / ; umount /mnt/cdrom ; eject
#Copy the PEM key file over for ipop3d
[root@newsrver /]# cp -p /usr/share/ssl/certs/ipop3d.pem
/home/EMAIL/root/usr/share/ssl/certs/

```

4.3.5.3 Xinetd Configuration for POP

Xinetd is the replacement for the inetd process which has long been a standard for the various versions of UNIX. What the xinetd and inetd programs do is handle the socket binding for service programs. What this means is that instead of a program binding to the socket itself, the xinetd/inetd process will bind to the port instead and wait for an inbound connection. Upon a connection being initiated, xinetd/inetd will fork (run) a copy of the service. The configuration xinetd configuration file for pop3 is called “/etc/xinetd.d/ipop3”, in this file, we need to change the server, server_args and flags, so that it will chroot to the /home/EMAIL/root environment. Since we do not want users to send their username and password in the clear, we actually want to leave this service disabled, and have everyone use POPS. The final file should look like: (The changed lines are in **bold**.)

```
[root@newsrserver /]# cat /etc/xinetd.d/ipop3
# default: off
# description: The POP3 service allows remote users to access their mail \
#              using an POP3 client such as Netscape Communicator, mutt, \
#              or fetchmail.
service pop3
{
    socket_type           = stream
    wait                 = no
    user                 = root
    flags                = NAMEINARGS
    server              = /usr/sbin/chroot
    server_args        = =I pop3d /home/EMAIL/root /usr/sbin/ipop3d
    log_on_success += HOST DURATION
    log_on_failure      += HOST
    disable              = yes
}
```

4.3.5.4 Xinetd Configuration for POPS

POPS is simply the POP protocol encoded in SSL. The same configuration changes must be made to it's xinetd configuration file which is “/etc/xinetd.d/pops” The contents of that file should be: (The changed lines are in **bold**.)

```
[root@newsrserver /]# cat /etc/xinetd.d/pops
# default: off
# description: The POP3S service allows remote users to access their mail \
#              using an POP3 client with SSL support such as fetchmail.
service pop3s
{
    socket_type           = stream
    wait                 = no
    user                 = root
    flags                = NAMEINARGS
    server              = chroot
    server_args        = ipop3d /home/EMAIL/root /usr/sbin/ipop3d
    log_on_success += HOST DURATION
    log_on_failure      += HOST
    disable              = no
}
```



```

New, TLSv1/SSLv3, Cipher is DES-CBC3-SHA
Server public key is 1024 bit
SSL-Session:
  Protocol   : TLSv1
  Cipher     : DES-CBC3-SHA
  Session-ID: E131194C61A13EE3A3EF8F98C21EECC60F8812880456DEB4C71D5E0E5C53FC5A
  Session-ID-ctx:
  Master-Key:
EBBF7984289F14F1526FFE1ED66ED41DB85C5109226EF8A19F0C2FAFDE727D066498B1ADFC500481B4FF935D0141E38
A
  Key-Arg    : None
  Start Time: 1021905151
  Timeout    : 300 (sec)
  Verify return code: 18 (self signed certificate)
---
+OK POP3 localhost v2001.78rh server ready
USER test
+OK User name accepted, password please
PASS J4ng0F3tt
+OK Mailbox open, 1 messages
LIST
+OK Mailbox scan listing follows
1 372
.
RETR 1
+OK 372 octets
Return-Path: <someuser@somehost.com>
Received: from fromtest.com (localhost [127.0.0.1])
        by newserver.markoram.com (8.11.6/8.11.6) with ESMTTP id g4A2dws19134
        for test@markoram.com; Thu, 9 May 2002 22:40:01 -0400
Date: Thu, 9 May 2002 22:40:01 -0400
From: someuser@somehost.com
Message-Id: <200205100240.g4A2dws19134@newserver.markoram.com>
Status:

Testing
.
QUIT
+OK Sayonara

```

4.3.6 HTTP and HTTPS / Apache

4.3.6.1 Background

Apache is one of the most common web servers in use on the Internet today. Apache is capable of providing both HTTP (Hyper Text Transfer Protocol) (RFC 2616 which can be viewed online at <ftp://ftp.isi.edu/in-notes/rfc2616.txt>) as well as HTTPS which is HTTP using SSL (Secure Socket Layer) or TLS (Transport Layer Security).

The HTTP protocol uses port 80 on the server side, since this is less than 1024, it is considered a privileged port, and therefore needs root permissions when it starts to first bind. However, Apache supports setting its user ID to something different after it has bound to port 80. In the default Red Hat Linux 7.3 install of Apache, the program is configured to run as user apache, and group apache.

Apache, as packaged with Red Hat Linux 7.3, also supports much more than serving up just static web pages. Besides having mod_ssl and OpenSSL support built in to support HTTPS, it also has PHP and mod_perl support built in, both of which are used to serve up dynamic content. It supports DAV (Distributed Authoring and Versioning)

(http://httpd.apache.org/docs-2.0/mod/mod_dav.html) and the execution of CGI (Common Gateway Interface) binaries and scripts. Since we are not going to run Apache in a chrooted environment, it seems only prudent to disable the extended functionality until it is explicitly needed. All of this functionality is controlled in the file `/etc/httpd/conf/httpd.conf`.

4.3.6.2 Configuring Apache

4.3.6.2.1 Disabling PHP

Comment out (by placing a '#' in front of the line) the following lines:

Line	Text
255	<code>LoadModule php_module modules/mod_php.so</code>
258	<code>LoadModule php3_module modules/libphp3.so</code>
261	<code>LoadModule php4_module modules/libphp4.so</code>
334	<code>AddModule mod_php.c</code>
337	<code>AddModule mod_php3.c</code>
340	<code>AddModule mod_php4.c</code>

4.3.6.2.2 Disabling mod_perl

Comment out (by placing a '#' in front of the line) the following lines:

Line	Text
252	<code>LoadModule perl_module modules/libperl.so</code>
331	<code>AddModule mod_perl.c</code>

4.3.6.2.3 Disabling DAV

Comment out (by placing a '#' in front of the line) the following lines:

Line	Text
264	<code>LoadModule dav_module modules/libdav.so</code>
343	<code>AddModule mod_dav.c</code>

4.3.6.2.4 Disabling CGIs

Comment out (by placing a '#' in front of the line) the following lines:

Line	Text
225	<code>LoadModule cgi_module modules/mod_cgi.so</code>
303	<code>AddModule mod_cgi.c</code>
746	<code>ScriptAlias /cgi-bin/ "/var/www/cgi-bin/"</code>
752	<code><Directory "/var/www/cgi-bin"></code>

753	AllowOverride None
754	Options None
755	Order allow,deny
756	Allow from all
757	</Directory>

4.3.6.3 Setting Apache to automatically startup

The httpd service is controlled via chkconfig like many of the other services.

```
[root@newserver root]# chkconfig --list httpd
httpd          0:off 1:off 2:off 3:off 4:off 5:off 6:off
[root@newserver root]# chkconfig --level 2345 httpd on
```

4.3.6.4 Testing

4.3.6.4.1 HTTP

```
#Test that the server replies and gives a valid result
[root@newserver /]# telnet 0 80
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Mon, 20 May 2002 16:56:57 GMT
Server: Apache/1.3.23 (Unix) (Red-Hat/Linux) mod_ssl/2.8.7 OpenSSL/0.9.6b
Last-Modified: Tue, 09 Apr 2002 18:56:58 GMT
ETag: "2acc5-b4a-3cb3397a"
Accept-Ranges: bytes
Content-Length: 2890
Connection: close
Content-Type: text/html
```

4.3.6.4.2 HTTPS

```
#Test that the server gives a valid response. Use openssl.
[root@newserver /]# openssl s_client -connect 0:443
CONNECTED(00000003)
depth=0 /C=--/ST=SomeState/L=SomeCity/O=SomeOrganization/OU=SomeOrganizationalUnit/
CN=localhost.localdomain/Email=root@localhost.localdomain
verify error:num=18:self signed certificate
verify return:1
depth=0 /C=--/ST=SomeState/L=SomeCity/O=SomeOrganization/OU=SomeOrganizationalUnit/
CN=localhost.localdomain/Email=root@localhost.localdomain
verify return:1
---
Certificate chain
 0 s:/C=--/ST=SomeState/L=SomeCity/O=SomeOrganization/OU=SomeOrganizationalUnit/
CN=localhost.localdomain/Email=root@localhost.localdomain
  i:/C=--/ST=SomeState/L=SomeCity/O=SomeOrganization/OU=SomeOrganizationalUnit/
CN=localhost.localdomain/Email=root@localhost.localdomain
---
```

```

Server certificate
-----BEGIN CERTIFICATE-----
MIIEEDCCA3WgAwIBAgIBADANBgkqhkiG9w0BAQQFADCBuzELMAkGA1UEBhMCLS0x
EjAQBGNVBAgTCVNvbWVtdGF0ZTERMA8GA1UEBxMIU29tZUNpdHkxGTAXBGNVBAoT
EFNvbWVpcmdhbml6YXRpb24xHzAdBgNVBAsTF1NvbWVpcmdhbml6YXRpb25hbFVu
aXQxHjAcBgNVBAMTFWxvY2FsaG9zdC5sb2NhbGRvbWVpbjEPMCCGCSqGSIb3DQEJ
ARYacm9vdEBsb2NhbGhvc3QubG9jYWxkb21haW4wHhcNMDIwNTA3MDkxNjU4WhcN
MDMwNTA3MDkxNjU4WjCBuzELMAkGA1UEBhMCLS0xEjAQBGNVBAgTCVNvbWVtdGF0
ZTERMA8GA1UEBxMIU29tZUNpdHkxGTAXBGNVBAoTEFNvbWVpcmdhbml6YXRpb24x
HzAdBgNVBAsTF1NvbWVpcmdhbml6YXRpb25hbFVuXQxHjAcBgNVBAMTFWxvY2Fsa
G9zdC5sb2NhbGRvbWVpbjEPMCCGCSqGSIb3DQEJARYacm9vdEBsb2NhbGhvc3Qu
bG9jYWxkb21haW4wZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJAoGBALLWqGXm8gpj
Bp8ORoXuce85n838xKRmMs2D3SmEcjUSCqk2VpOZH64MCP9MtiO2LPH86iwQE6nf
jn1Vg9S1WEVEN1yka73aujC1PmM1KJSc6TArtN1WYN/tpDQY89MLw9GibqZp9geN
ZRCBrFz4b6ZmVR5hTxrIkqba7JPmCQdnAgMBAAGjggEcMIIBGDAAdBgNVHQ4EFgQU
ejln0CJ8QmT3UnFOcaTHQ9zFzoQwgegGA1UdIwSB4DCB3YAUEjln0CJ8QmT3UnFO
caTHQ9zFzoShgcGkgb4wgbsxCzAJBgNVBAYTAi0tMRiWEAYDVQQIEw1Tb211U3Rh
dGUxETAPBgNVBAcTFCNvbWVudXR5MRkwFwYDVQQKEwBTb211T3JnYW5pemF0aW9u
MR8wHQYDVQQLExZTb211T3JnYW5pemF0aW9uYXVbml0MR4wHAYDVQQDEwVsb2Nhb
Ghvc3QubG9jYWxkb21haW4xKTAnBgkqhkiG9w0BCQEWGnJvb3RAbG9jYWxob3N0
LmxvY2FsaG9zdC5sb2NhbGhvc3QubG9jYWxkb21haW4wZ8wDQYJKoZIhvcNAQEBBQADgYEA
B5fpAwU15KWAR9N9Gj3lC6u3rkRdE6Sag6jbstA+APqHD0dlFL83OMQgsKkIqH3A
yyzBPKHAVLWkyheJVe6+BKYs9elddinSzxTc+CDtKgmWaqR8ugG56SIq50Ed89Pg
6r91HzpLP/Mw6NfcAEY2eu/zEdKpupX/ecjR1MP5HE=
-----END CERTIFICATE-----
subject=/C=--/ST=SomeState/L=SomeCity/O=SomeOrganization/OU=SomeOrganizationalUnit/
CN=localhost.localdomain/Email=root@localhost.localdomain
issuer=/C=--/ST=SomeState/L=SomeCity/O=SomeOrganization/OU=SomeOrganizationalUnit/
CN=localhost.localdomain/Email=root@localhost.localdomain
---
No client certificate CA names sent
---
SSL handshake has read 1596 bytes and written 314 bytes
---
New, TLSv1/SSLv3, Cipher is EDH-RSA-DES-CBC3-SHA
Server public key is 1024 bit
SSL-Session:
  Protocol   : TLSv1
  Cipher     : EDH-RSA-DES-CBC3-SHA
  Session-ID: 78C4F71D872B9B25C8923152C1EE999F754637A87B8406AAE2548C2BD1DDF52B
  Session-ID-ctx:
  Master-Key:
6E3D99503D63B652A0C52D639834C5D3DB67E9565F226D3BCD51A9E9AE01B8D7D61DCB0623BC9C1F25E83953CE4531A
C
  Key-Arg    : None
  Start Time: 1021914166
  Timeout    : 300 (sec)
  Verify return code: 18 (self signed certificate)
---
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Mon, 20 May 2002 17:02:49 GMT
Server: Apache/1.3.23 (Unix) (Red-Hat/Linux) mod_ssl/2.8.7 OpenSSL/0.9.6b
Last-Modified: Tue, 09 Apr 2002 18:56:58 GMT
ETag: "2acc5-b4a-3cb3397a"
Accept-Ranges: bytes
Content-Length: 2890
Connection: close
Content-Type: text/html

closed

```


4.3.7 Syslog

4.3.7.1 Background

Syslog is the standard logging daemon used in Red Hat Linux. The default values are correct for a default install, however since we are running several chrooted environments, those programs running within those chrooted environments can no longer access “/dev/log” (the UNIX socket typically written to in order to communicate with the syslog program.) To remedy this, syslog needs to be configured to monitor additional devices.

4.3.7.2 Configuration

There are two configuration programs for syslog, one “/etc/syslog.conf” tells it what type of message to log where, detailed information about that is available by manning syslogd.conf (“man syslog.conf”) on the system. The other configuration file controls its behavior at startup, this file /etc/sysconfig/syslog is the file we need to modify.

In /etc/sysconfig/syslog there is a line which starts as “SYSLOGD_OPTIONS=” this line should be modified to read:

```
SYSLOGD_OPTIONS="-m 0 -a /home/EMAIL/root/dev/log -a /home/named/root/dev/log"
```

The only change is simply to tell it (via the -a option) to monitor two additional log sockets. Each of those sockets will look like “/dev/log” to any processes running within the chrooted environment.

4.3.8 Snort

4.3.8.1 Background

Snort is a network based intrusion detection system (IDS). Detailed information about snort is available at their website: <http://www.snort.org/>. Having a network based IDS already installed before it is needed can be very helpful in detecting problems and reacting in a proper and timely fashion.

Snort has a dependency which was not installed during the base OS install. This dependency is libpcap and it can be located on the second of the three Red Hat 7.3 Linux CDs, and should be installed prior to installing snort.

4.3.8.2 Installing

4.3.8.2.1 libpcap

To install libpcap, insert disc 2 of Red Hat Linux 7.3 into the CD-ROM drive and:

```
#Mount disc2
[root@newsrver root]# mount /dev/cdrom /mnt/cdrom
#Install the RPM
[root@newsrver root]# rpm -ivh /mnt/cdrom/RedHat/RPMS/libpcap-0.6.2-12.i386.rpm
Preparing.. ##### [100%]
    1:libpcap ##### [100%]
```

```
#Unmount the CD-ROM
[root@newsrver root]# umount /mnt/cdrom
#Eject the CD-ROM
[root@newsrver root]# eject
```

4.3.8.2.2 Snort

The RPM for snort, and its md5 hash can be retrieved from their website, the URLs are <http://www.snort.org/binaries/RPMS/snort-1.8.4-1.i386.rpm> and <http://www.snort.org/binaries/RPMS/snort-1.8.4-1.i386.rpm.md5> respectively.

To install snort, be logged in as root and in a secure directory (e.g. /root)

```
#Download the RPM
[root@newsrver root]# wget http://www.snort.org/dl/binaries/RPMS/snort-1.8.4-1snort.i386.rpm
#Download the MD5
[root@newsrver root]# wget http://www.snort.org/dl/binaries/RPMS/snort-1.8.4-1snort.i386.rpm.md5
#Generate the md5 for the downloaded file
[root@newsrver root]# md5sum snort-1.8.4-1snort.i386.rpm
dd06ccb59231ce340aeaf6c3a9bb4156 snort-1.8.4-1snort.i386.rpm
#And compare to the contents of the .md5 file
#If they match (and they do), install the RPM
[root@newsrver root]# rpm -ivh snort-1.8.4-1snort.i386.rpm
```

4.3.8.3 Chrooting / Configuration

Since snort interprets real world network data there lies the potential for an exploit. To protect our system from this type of potential attack the program can be run in a chrooted environment.

First, edit the start the startup script “/etc/rc.d/init.d/snortd” and locate the line “-i \$INTERFACE -c /etc/snort/snort.conf” and insert “-t /home/snort -u snort” to it that line so that it reads “-i \$INTERFACE -c /etc/snort/snort.conf -t /home/snort -u snort”

Next, we need to make the directories within the chrooted directory:

```
#Make the user, and group 'snort' which the process will run as
[root@newsrver root]# groupadd snort
[root@newsrver root]# useradd -g snort snort
#Make the etc directory where snort will expect to find the configuration file
[root@newsrver root]# mkdir -p /home/snort/etc/snort
#Make the log directory where snort will write it's logs too.
[root@newsrver root]# mkdir -p /home/snort/var/log/snort/
#Change the ownership of the chrooted environment to the snort user and group.
[root@newsrver root]# chown -R snort.snort /home/snort/var
#Copy the snort configuration files to the chrooted directory
[root@newsrver root]# cp -p /etc/snort/* /home/snort/etc/snort/
#Add snortd to the list of services controlled by chkconfig
[root@newsrver root]# chkconfig --add snortd
```

The service can then be started by issuing the commands “/etc/rc.d/init.d/snortd start” and “/etc/rc.d/init.d/snortd stop” respectively. The default settings for starting at runtime can now be controlled by using snortd, and are initialized to start automatically for runlevels 2, 3, 4, and 5.

4.3.9 Kernel level IP filtering (Netfilter / IPTables)

4.3.9.1 Background

The Linux 2.4 kernel has the ability to filter network connections. This is known as ‘netfilter’. This filtering allows for simply ‘hardening’ a server to allowing it to run as a firewall. The filters which are managed by a user-space application called iptables. The website for the netfilter and iptables project can be found at: <http://netfilter.samba.org/>.

This machine is not running as a firewall, however it is desired to make sure that no services are exposed to the world unless they are explicitly allowed. Netfilter and iptables will be used to do this.

4.3.9.2 Configuring

Red Hat Linux 7.3 starts and stops iptables via the rc script /etc/rc.d/init.d/iptables. If given the ‘start’ option it will read the file /etc/sysconfig/iptables and load those rules.

First, lets make a list of what we want the rules to accomplish:

- Default to denying any traffic
- Allow the machine itself to initiate any outbound connections.
- Allow inbound the following:
 - DNS queries udp/53 and tcp/53
 - HTTP tcp/80
 - HTTPS tcp/443
 - POP3S tcp/995
 - SMTP tcp/25
 - SMTPS tcp/465
 - SSH tcp/22

To configure IPTables, either follow the steps outline in the manual steps below, or follow the automatic configuration step.

4.3.9.2.1 Manual Steps

```
#Set defaults to be denied
[root@newsrver root]# /sbin/iptables -P INPUT DROP
[root@newsrver root]# /sbin/iptables -P OUTPUT DROP
```

```

[root@newsrerver root]# /sbin/iptables -P FORWARD DROP
#Allow inbound SSH
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 22 -j ACCEPT
#Allow inbound SMTP
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 25 -j ACCEPT
#Allow inbound DNS queries (standard)
[root@newsrerver root]# /sbin/iptables -I INPUT -p udp --dport 53 -j ACCEPT
#Allow inbound DNS (tcp, zone transfers, etc.)
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 53 -j ACCEPT
#Allow inbound HTTP
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 80 -j ACCEPT
#Allow inbound HTTPS
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 443 -j ACCEPT
#Allow inbound SMTPS
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 465 -j ACCEPT
#Allow inbound POP3S
[root@newsrerver root]# /sbin/iptables -I INPUT -p tcp --dport 995 -j ACCEPT
#Allow this machine to create new connections outbound
[root@newsrerver root]# /sbin/iptables -I OUTPUT -m state --state NEW,RELATED,ESTABLISHED
-j ACCEPT
#Allow already established connections
[root@newsrerver root]# /sbin/iptables -I INPUT -m state --state RELATED,ESTABLISHED -j
ACCEPT
#Now lets view the chains
[root@newsrerver root]# /sbin/iptables -L
Chain INPUT (policy DROP)
target    prot opt source                destination
ACCEPT    all  --  anywhere               anywhere           state RELATED,ESTABLISHED
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:pop3s
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:smtps
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:https
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:http
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:domain
ACCEPT    udp  --  anywhere               anywhere           udp dpt:domain
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:smtp
ACCEPT    tcp  --  anywhere               anywhere           tcp dpt:ssh

Chain FORWARD (policy DROP)
target    prot opt source                destination

Chain OUTPUT (policy DROP)
target    prot opt source                destination
ACCEPT    all  --  anywhere               anywhere           state NEW,RELATED,ESTABLISHED

```

4.3.9.2 Automatic Steps

Simply copy the file labeled “/etc/sysconfig/iptables” from the Files chapter of this document to /etc/sysconfig/iptables on the machine.

4.3.9.3 Testing

To test that the iptables has taken affect, enable a service which is explicitly denied by the iptables, try connecting to it, and then disable the service. If connecting to that service fails, yet connecting to known allowed services works, then the iptables is working correctly.

```

#Check that telnet is disabled
[root@newsrerver root]# chkconfig --list telnet
telnet          off
#Enable telnet
[root@newsrerver root]# chkconfig telnet on
#Check that it is enabled

```

```
[root@newsrserver root]# chkconfig --list telnet
telnet          on
#Test connecting to the new service
[root@newsrserver root]# telnet 0
Trying 0.0.0.0...
telnet: connect to address 0.0.0.0: Connection timed out
#Test connecting to a known allowed service
[root@newsrserver root]# telnet 0 pop3
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
+OK POP3 localhost v2001.78rh server ready
^]
telnet> c
Connection closed.
#Disable telnet again
[root@newsrserver root]# chkconfig telnet off
```

4.3.10 TCP Wrappers

TCP wrappers are another way of restricting network access to services. The xinetd program as well as stunnel will use the TCP wrapper files to lookup if a host is allowed to connect to the service or not. The TCP wrapper files are:

- /etc/hosts.allow – Specifies for each service what hosts are allowed to communicate.
- /etc/hosts.deny – Specifies for each service what hosts are not allowed to communicate.

To start with, make the default to deny everything by adding the entry “ALL: ALL” to /etc/hosts.deny:

```
[root@newsrserver /]# cat >>/etc/hosts.deny
ALL: ALL
^D
```

Then, we want to allow anyone to connect to pop3, pop3s, and smtps. Both pop3 and pop3s will look under the service ‘ipop3d’:

```
[root@newsrserver /]# cat >>/etc/hosts.allow
ipop3d: ALL: ALLOW
smtps: ALL: ALLOW
sshd: ALL: ALLOW
```

4.3.11 Shutting off unneeded services

Using *netstat* and *ps*, we will review what services are running on the machine, and stop any which are not necessary.

```
[root@newsrserver root]# netstat -ap
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
tcp        0      0 *:32768                 :::*                    LISTEN      652/rpc.statd
```

tcp	0	0	*:pop3s	*:*	LISTEN	22397/xinetd	
tcp	0	0	*:pop3	*:*	LISTEN	22397/xinetd	
tcp	0	0	*:sunrpc	*:*	LISTEN	624/portmap	
tcp	0	0	*:http	*:*	LISTEN	21129/httpd	
tcp	0	0	*:smtps	*:*	LISTEN	22397/xinetd	
tcp	0	0	192.168.17.220:domain	*:*	LISTEN	17708/named	
tcp	0	0	newserver.markor:domain	*:*	LISTEN	17708/named	
tcp	0	0	*:ssh	*:*	LISTEN	11148/sshd	
tcp	0	0	*:telnet	*:*	LISTEN	22397/xinetd	
tcp	0	0	*:smtp	*:*	LISTEN		
22462/sendmail: acc							
tcp	0	0	newserver.markoram:rndc	*:*	LISTEN	17708/named	
tcp	0	0	*:https	*:*	LISTEN	21129/httpd	
udp	0	0	*:32768	*:*			
652/rpc.statd							
udp	0	0	*:32776	*:*		17708/named	
udp	0	0	192.168.17.220:domain	*:*		17708/named	
udp	0	0	newserver.markor:domain	*:*		17708/named	
udp	0	0	*:sunrpc	*:*		624/portmap	
Active UNIX domain sockets (servers and established)							
Proto	RefCnt	Flags	Type	State	I-Node	PID/Program name	Path
unix	4	[]	DGRAM		143517	19081/syslogd	/dev/log
unix	3	[]	DGRAM		143519	19081/syslogd	
/home/EMAIL/root/dev/log							
unix	2	[]	DGRAM		143521	19081/syslogd	
/home/named/root/dev/log							
unix	2	[ACC]	STREAM	LISTENING	1563	911/gpm	/dev/gpmctl
unix	2	[]	DGRAM		165954	22462/sendmail: acc	
unix	2	[]	DGRAM		165874	22397/xinetd	
unix	2	[]	DGRAM		143530	19086/klogd	
unix	2	[]	DGRAM		116077	17708/named	
unix	2	[]	DGRAM		1586	929/crond	
unix	2	[]	DGRAM		1203	765/apmd	
unix	2	[]	DGRAM		1057	652/rpc.statd	

From the above output of *netstat* the only two services which we do not want running are *rpc.statd* and *portmap*. We will shut them off using their *rc script* and configure them not to start up automatically at boot time by using *chkconfig*.

```
#Turn off portmap
[root@newserver root]# /etc/rc.d/init.d/portmap stop
Stopping portmapper:
[ OK ]

#Check what runlevels it runs at
[root@newserver root]# chkconfig --list portmap
portmap          0:off 1:off 2:off 3:on  4:on  5:on  6:off
#Set it to never run automatically
[root@newserver root]# chkconfig --level 0123456 portmap off
#Stop runlock
[root@newserver root]# /etc/rc.d/init.d/nfslock stop
Stopping NFS statd:
[ OK ]

[root@newserver root]# chkconfig --list portmap
nfslock          0:off 1:off 2:off 3:on  4:on  5:on  6:off
#Set it to never run automatically.
[root@newserver root]# chkconfig --level 0123456 nfslock off
```

Next, we will review the output of *ps*:

```
[root@newserver root]# ps auxww
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.1  1368  476 ?        S    May07   0:04 init
root         2  0.0  0.0     0     0 ?        SW   May07   0:01 [keventd]
root         3  0.0  0.0     0     0 ?        SW   May07   0:00 [kapmd]
```

root	4	0.0	0.0	0	0	?	SWN	May07	0:00	[ksoftirqd_CPU0]
root	5	0.0	0.0	0	0	?	SW	May07	0:02	[kswapd]
root	6	0.0	0.0	0	0	?	SW	May07	0:00	[bdflush]
root	7	0.0	0.0	0	0	?	SW	May07	0:00	[kupdated]
root	8	0.0	0.0	0	0	?	SW	May07	0:00	[mdrecoveryd]
root	12	0.0	0.0	0	0	?	SW	May07	0:00	[kjournald]
root	91	0.0	0.0	0	0	?	SW	May07	0:00	[khubd]
root	186	0.0	0.0	0	0	?	SW	May07	0:00	[kjournald]
root	187	0.0	0.0	0	0	?	SW	May07	0:07	[kjournald]
root	188	0.0	0.0	0	0	?	SW	May07	0:00	[kjournald]
root	189	0.0	0.0	0	0	?	SW	May07	0:00	[kjournald]
root	190	0.0	0.0	0	0	?	SW	May07	0:01	[kjournald]
root	765	0.0	0.1	1360	480	?	S	May07	0:00	/usr/sbin/apmd -p 10 -w 5
-W -P /etc/sysconfig/apm-scripts/apmscript										
root	911	0.0	0.1	1400	452	?	S	May07	0:00	gpm -t ps/2 -m /dev/mouse
root	929	0.0	0.2	1544	620	?	S	May07	0:00	crond
daemon	965	0.0	0.2	1404	524	?	S	May07	0:00	/usr/sbin/atd
root	974	0.0	0.4	2272	1032	?	S	May07	0:00	login -- root
root	976	0.0	0.4	2272	1032	?	S	May07	0:00	login -- root
root	978	0.0	0.4	2272	1032	?	S	May07	0:00	login -- root
root	11148	0.0	0.4	2620	1188	?	S	May07	0:00	/usr/sbin/sshd
root	12192	0.0	0.5	2500	1348	tty1	S	May07	0:00	-bash
root	15641	0.0	0.5	2468	1332	tty3	S	May09	0:00	-bash
named	17708	0.0	1.0	10520	2652	?	S	May09	0:00	named -u named -t
/home/named/root										
named	17710	0.0	1.0	10520	2652	?	S	May09	0:00	named -u named -t
/home/named/root										
named	17711	0.0	1.0	10520	2652	?	S	May09	0:00	named -u named -t
/home/named/root										
named	17712	0.0	1.0	10520	2652	?	S	May09	0:00	named -u named -t
/home/named/root										
named	17713	0.0	1.0	10520	2652	?	S	May09	0:00	named -u named -t
/home/named/root										
root	19081	0.0	0.2	1428	560	?	S	May09	0:00	syslogd -m 0 -a
/home/EMAIL/root/dev/log -a /home/named/root/dev/log										
root	19086	0.0	0.1	1364	444	?	S	May09	0:00	klogd -x
root	21892	0.0	0.5	2472	1320	tty5	S	13:27	0:00	-bash
root	22162	0.0	0.3	1964	836	tty5	S	13:49	0:00	less /tmp/xinetd.strace
root	22397	0.0	0.3	2208	928	?	S	14:03	0:00	xinetd -stayalive -reuse -
pidfile /var/run/xinetd.pid										
root	22462	0.0	0.7	4660	1892	?	S	14:10	0:00	sendmail: accepting
connections										
root	22575	0.0	0.1	1344	400	tty6	S	14:16	0:00	/sbin/mingetty tty6
root	22577	0.1	0.1	1344	400	tty4	S	14:16	0:00	/sbin/mingetty tty4
root	22580	0.0	0.1	1344	400	tty2	S	14:16	0:00	/sbin/mingetty tty2
root	22581	0.0	0.2	2616	704	pts/1	R	14:16	0:00	ps auxww

*Note, several pages of the *htp* process were deleted.

The only process which is not needed is *gpm*. Since it will not be used and it could take some resources potentially affecting the availability of another service it should be shut off.

```
#Stop GPM - we won't be using it.
[root@newsrver root]# /etc/rc.d/init.d/gpm stop
Shutting down console mouse services: [ OK ]

#Check what runelevels it is set to run at.
[root@newsrver root]# chkconifig --list gpm
gpm          0:off 1:off 2:on  3:on  4:on  5:on  6:off
#Set it to never run automatically
[root@newsrver root]# chkconfig --level 0123456 gpm off
```

To see other services that are at least attempted to be started at runtime, check the rc files for runlevel 3.

```

#See what files are in the directory to be started for runlevel 3.
[root@newserver rc3.d]# ls -l /etc/rc.d/rc3.d/S*
/etc/rc.d/rc3.d/S05kudzu
/etc/rc.d/rc3.d/S08ip6tables
/etc/rc.d/rc3.d/S08ipchains
/etc/rc.d/rc3.d/S08iptables
/etc/rc.d/rc3.d/S09isdn
/etc/rc.d/rc3.d/S10network
/etc/rc.d/rc3.d/S12syslog
/etc/rc.d/rc3.d/S17keytable
/etc/rc.d/rc3.d/S20random
/etc/rc.d/rc3.d/S26apmd
/etc/rc.d/rc3.d/S28autofs
/etc/rc.d/rc3.d/S40snortd
/etc/rc.d/rc3.d/S55named
/etc/rc.d/rc3.d/S55sshd
/etc/rc.d/rc3.d/S56rawdevices
/etc/rc.d/rc3.d/S56xinetd
/etc/rc.d/rc3.d/S80sendmail
/etc/rc.d/rc3.d/S85httpd
/etc/rc.d/rc3.d/S90crond
/etc/rc.d/rc3.d/S95anacron
/etc/rc.d/rc3.d/S95atd
/etc/rc.d/rc3.d/S97rhnisd
/etc/rc.d/rc3.d/S99local

```

Of these services, kudzu, ipchains, isdn, apmd, autofs, and rhnsd can all be disabled.

```

#Disable kudzu - We will not be adding new hardware to this machine regularly
[root@newserver rc3.d]# chkconfig --level 0123456 kudzu off
#Disable ipchains - We are running iptables on this machine
[root@newserver rc3.d]# chkconfig --level 0123456 ipchains off
#Disable apmd - We do not need power management on this machine
[root@newserver rc3.d]# chkconfig --level 0123456 apmd off
#Disable autofs - There are no remote file systems to be mounted
[root@newserver rc3.d]# chkconfig --level 0123456 autofs off
#Disable isdn - There is no ISDN line to this machine
[root@newserver rc3.d]# chkconfig --level 0123456 isdn off
#Disable rhnsd - We do not need any of the Red Hat network services
[root@newserver rc3.d]# chkconfig --level 0123456 rhnsd off

```

The cron jobs set to run on the machine should also be checked. The default install of Red Hat Linux 7.3 installs both cron and anacron. By default there are no entries in root's normal crontab:

```

#Check normal crontab for root
[root@newserver /]# crontab -l
no crontab for root

```

Next, check the /etc/crontab file:

```

[root@newserver /]# cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly

```



```
0-59/5 * * * * root /usr/bin/mrtg /etc/mrtg/mrtg.cfg
```

Since mrtg is not needed on this system, the last line of /etc/crontab should be commented out by placing a '#' at the beginning of that line.

The directories /etc/cron.hourly, /etc/cron.daily, /etc/cron.weekly, and /etc/cron.monthly should all be reviewed to see what is running:

```
[root@newsrserver /]# ls -la /etc/cron.{hourly,daily,weekly,monthly}
/etc/cron.daily:
total 36
drwxr-xr-x  2 root  root  4096 May 17 14:29 .
drwxr-xr-x 43 root  root  4096 May 19 10:24 ..
lrwxrwxrwx  1 root  root  28 May  7 05:15 00-logwatch ->
./log.d/scripts/logwatch.pl
-rwxr-xr-x  1 root  root  135 Apr 17 08:29 00webalizer
-rwxr-xr-x  1 root  root  276 Jun 24 2001 0anacron
-rwxr-xr-x  1 root  root  51 Apr 15 16:27 logrotate
-rwxr-xr-x  1 root  root  418 Mar 25 09:17 makewhatis.cron
-rwxr-xr-x  1 root  root  104 Apr 18 17:35 rpm
-rwxr-xr-x  1 root  root  132 Jun 25 2001 slocate.cron
-rwxr-xr-x  1 root  root  103 Sep  7 2001 tetex.cron
-rwxr-xr-x  1 root  root  193 Aug 29 2001 tmpwatch
-rwxr-xr-x  1 root  root  315 Feb 26 18:11 tripwire-check

/etc/cron.hourly:
total 8
drwxr-xr-x  2 root  root  4096 Jul 19 2001 .
drwxr-xr-x 43 root  root  4096 May 19 10:24 ..

/etc/cron.monthly:
total 12
drwxr-xr-x  2 root  root  4096 May  7 05:14 .
drwxr-xr-x 43 root  root  4096 May 19 10:24 ..
-rwxr-xr-x  1 root  root  278 Jun 24 2001 0anacron

/etc/cron.weekly:
total 16
drwxr-xr-x  2 root  root  4096 May  7 05:14 .
drwxr-xr-x 43 root  root  4096 May 19 10:24 ..
-rwxr-xr-x  1 root  root  277 Jun 24 2001 0anacron
-rwxr-xr-x  1 root  root  414 Mar 25 09:17 makewhatis.cron
```

Of these, /etc/cron.daily/tetex.cron, and /etc/cron.daily/tmpwatch can be deleted as this machine will not have any normal users on it.

```
#Remove tetext.cron
[root@newsrserver /]# rm /etc/cron.daily/tetext.cron
#Delete tmpwatch
[root@newsrserver /]# rm /etc/cron.daily/tmpwatch
```

4.3.12 Removing SUID/SGID from binaries

4.3.12.1 What are SUID/SGID programs?

Certain programs in the UNIX environment need to be able to run with another user's or group's privileges. This is known as "set user ID" and "set group ID" respectively. For instance, when a user needs to change their password, the file that actually gets changed

is /etc/shadow which is owned by root and not even readable by the standard user. For this very reason you will notice that /usr/bin/passwd is owned by root, and has the suid bit flipped. If the set UID bit is true, then the program will run with the privileges of the owner, if the SGID bit is true, then the program will run with the group's privileges. Follow is an example 'ls -l' of a suid and a sgid binary:

```
[root@newsrver /]# ls -l /usr/bin/passwd
-r-s--x--x 1 root root 15104 Mar 13 20:44 /usr/bin/passwd
```

*Notice the 's' in the 'owner' list of permissions on the left? That indicates that the program has the set user ID bit on.

```
[root@newsrver /]# ls -l /usr/bin/write
-rwxr-sr-x 1 root tty 8584 Apr 1 18:26 /usr/bin/write
```

*Notice the 's' in the 'group' list of permissions on the left? This indicates that the program has the set group ID bit on.

4.3.12.2 Finding the SUID/SGID programs

If not already logged in, log in as root then issue the following command:

```
root@newsrver root]# find / -type f \( -perm -4000 -o -perm -2000 \) -ls
176677 20 -rwxr-sr-x 1 root mail 17811 Mar 25 13:03
/home/EMAIL/root/usr/bin/lockfile
176687 36 -rwsr-xr-x 1 root root 34296 Mar 27 20:40
/home/EMAIL/root/usr/bin/chage
176689 36 -rwsr-xr-x 1 root root 36100 Mar 27 20:40
/home/EMAIL/root/usr/bin/gpasswd
448108 20 -rwsr-xr-x 1 root root 17461 Apr 19 12:35
/home/EMAIL/root/usr/sbin/usernetctl
448224 448 -r-sr-xr-x 1 root root 451280 Apr 8 06:55
/home/EMAIL/root/usr/sbin/sendmail.sendmail
176702 20 -rwsr-xr-x 1 root root 19116 Apr 8 12:02
/home/EMAIL/root/bin/su
176656 16 -rwxr-sr-x 1 root root 14657 Apr 19 12:35
/home/EMAIL/root/sbin/netreport
176666 124 -r-sr-xr-x 1 root root 120264 Apr 9 23:24
/home/EMAIL/root/sbin/pwdb_chkpwd
176667 20 -r-sr-xr-x 1 root root 16992 Apr 9 23:24
/home/EMAIL/root/sbin/unix_chkpwd
find: /proc/22698/fd/4: No such file or directory
31535 36 -rwsr-xr-x 1 root root 34296 Mar 27 20:40 /usr/bin/chage
31537 36 -rwsr-xr-x 1 root root 36100 Mar 27 20:40 /usr/bin/gpasswd
31611 40 -rwsr-xr-x 1 root root 37528 Jan 17 12:34 /usr/bin/at
31640 20 -rwxr-sr-x 1 root mail 17811 Mar 25 13:03 /usr/bin/lockfile
31689 28 -rwxr-sr-x 1 root slocate 25020 Jun 25 2001 /usr/bin/slocate
31725 16 -r-s--x--x 1 root root 15104 Mar 13 20:44 /usr/bin/passwd
31762 8 -r-xr-sr-x 1 root tty 6920 Mar 14 15:24 /usr/bin/wall
31774 12 -rws--x--x 1 root root 12072 Apr 1 18:26 /usr/bin/chfn
31775 12 -rws--x--x 1 root root 11496 Apr 1 18:26 /usr/bin/chsh
31793 8 -rws--x--x 1 root root 4764 Apr 1 18:26 /usr/bin/newgrp
31804 12 -rwxr-sr-x 1 root tty 8584 Apr 1 18:26 /usr/bin/write
31809 24 -rwsr-xr-x 1 root root 21080 Apr 15 00:49 /usr/bin/crontab
31846 220 -rwsr-xr-x 1 root root 219932 Apr 4 22:27 /usr/bin/ssh
31900 16 -rwsr-xr-x 1 root root 14588 Jul 24 2001 /usr/bin/rcp
31902 12 -rwsr-xr-x 1 root root 10940 Jul 24 2001 /usr/bin/rlogin
31903 8 -rwsr-xr-x 1 root root 7932 Jul 24 2001 /usr/bin/rsh
32613 88 ---s--x--x 1 root root 84680 Apr 18 12:35 /usr/bin/sudo
```

126044	32	-rwsr-xr-x	1	root	root	32673	Apr	18	17:40	/usr/sbin/ping6
126048	16	-rwsr-xr-x	1	root	root	13994	Apr	18	17:40	/usr/sbin/traceroute6
126165	8	-rwxr-sr-x	1	root	utmp	6604	Jun	24	2001	/usr/sbin/utempter
126191	448	-r-sr-xr-x	1	root	root	451280	Apr	8	06:55	/usr/sbin/sendmail.sendmail
126194	24	-rws--x--x	1	root	root	22388	Apr	15	18:15	/usr/sbin/userhelper
126206	20	-rwsr-xr-x	1	root	root	17461	Apr	19	12:35	/usr/sbin/usernetctl
126270	20	-rwsr-xr-x	1	root	root	20140	Mar	14	19:22	/usr/sbin/traceroute
126297	16	-rwxr-sr-x	1	root	lock	13573	Feb	25	15:42	/usr/sbin/lockdev
126356	24	-r-s--x---	1	root	apache	22826	Apr	9	14:56	/usr/sbin/suexec
29258	36	-rwsr-xr-x	1	root	root	35192	Apr	18	17:40	/bin/ping
29442	64	-rwsr-xr-x	1	root	root	60104	Apr	1	18:26	/bin/mount
29443	32	-rwsr-xr-x	1	root	root	30664	Apr	1	18:26	/bin/umount
29464	20	-rwsr-xr-x	1	root	root	19116	Apr	8	12:02	/bin/su
75748	124	-r-sr-xr-x	1	root	root	120264	Apr	9	23:24	/sbin/pwdb_chkpwd
75749	20	-r-sr-xr-x	1	root	root	16992	Apr	9	23:24	/sbin/unix_chkpwd
75832	16	-rwxr-sr-x	1	root	root	14657	Apr	19	12:35	/sbin/netreport

While thinking about these programs on this specific machine, there are two things to keep in mind. First, there are no 'real' users other than the root user, so most applications that are not services/daemons should not need the set user/group ID. Second, the binaries do not need the read permissions turned on. If the binary is readable, it might help a malicious person understand the program better and thus be able to exploit a vulnerability.

Remove all permissions from the programs which will not be used ever:

```
#First remove s{U/G}id permissions from the binaries
#By making is so no one can run the following programs
[root@newsrver root]# chmod 000 /home/EMAIL/root/usr/bin/chage
[root@newsrver root]# chmod 000 /home/EMAIL/root/usr/bin/gpasswd
[root@newsrver root]# chmod 000 /home/EMAIL/root/usr/sbin/usernetctl
[root@newsrver root]# chmod 000 /home/EMAIL/root/bin/su
[root@newsrver root]# chmod 000 /home/EMAIL/root/sbin/netreport
[root@newsrver root]# chmod 000 /usr/bin/chage
[root@newsrver root]# chmod 000 /usr/bin/gpasswd
[root@newsrver root]# chmod 000 /usr/bin/newgrp
[root@newsrver root]# chmod 000 /usr/bin/rcp
[root@newsrver root]# chmod 000 /usr/bin/rlogin
[root@newsrver root]# chmod 000 /usr/bin/rsh
[root@newsrver root]# chmod 000 /usr/bin/sudo
[root@newsrver root]# chmod 000 /usr/sbin/ping6
[root@newsrver root]# chmod 000 /usr/sbin/traceroute6
[root@newsrver root]# chmod 000 /usr/sbin/userhelper
[root@newsrver root]# chmod 000 /usr/sbin/sernetctl
[root@newsrver root]# chmod 000 /bin/su
```

Leave the ones which might be run by root run-able by root only:

```
#Make is so that only the owner (root) can run these programs
[root@newsrver root]# chmod 100 /usr/bin/at
[root@newsrver root]# chmod 100 /usr/bin/passwd
[root@newsrver root]# chmod 100 /usr/bin/chfn
[root@newsrver root]# chmod 100 /usr/bin/chsh
[root@newsrver root]# chmod 100 /usr/bin/crontab
[root@newsrver root]# chmod 100 /usr/bin/ssh
[root@newsrver root]# chmod 100 /usr/sbin/traceroute
[root@newsrver root]# chmod 100 /bin/ping
[root@newsrver root]# chmod 100 /bin/mount
[root@newsrver root]# chmod 100 /bin/umount
[root@newsrver root]# chmod 100 /sbin/netreport
```

Remove read permissions from the rest:

```
#NOW REMOVE READ FROM THE OTHERS
[root@newsrver root]# chmod -r /home/EMAIL/root/usr/bin/lockfile
[root@newsrver root]# chmod -r /home/EMAIL/root/sbin/pwdb_chkpwd
[root@newsrver root]# chmod -r /home/EMAIL/root/sbin/unix_chkpwd
[root@newsrver root]# chmod -r /usr/bin/lockfile
[root@newsrver root]# chmod -r /usr/bin/slocate
[root@newsrver root]# chmod -r /usr/bin/wall
[root@newsrver root]# chmod -r /usr/bin/write
[root@newsrver root]# chmod -r /usr/sbin/utempter
[root@newsrver root]# chmod -r /usr/sbin/sendmail.sendmail
[root@newsrver root]# chmod -r /usr/sbin/lockdev
[root@newsrver root]# chmod -r /usr/sbin/suexec
[root@newsrver root]# chmod -r /sbin/pwdb_chkpwd
[root@newsrver root]# chmod -r /sbin/unix_chkpwd
```

4.3.13 LIDS

4.3.13.1 Background

LIDS (Linux Intrusion Detection) provides MAC (Mandatory Access Control) support and allows for granularly limiting the root capabilities. There are two parts to LIDS, the kernel patch which enforce the rules and the user space utilities.

4.3.13.2 Retrieving LIDS

The retrieval of the LIDS software requires that the machine be connected to the Internet or that this step be performed on another machine and copied over.

```
[root@newsrver /]# cd /usr/src/
#Get the software
[root@newsrver src]# wget http://www.lids.org/downloads/lids-1.1.1r2-
2.4.18.tar.gz
...output of wget omitted...
#Untar the source
[root@newsrver src]# tar -zvxf lids-1.1.1r2-2.4.18.tar.gz
```

4.3.13.3 The Linux Kernel

Red Hat Linux 7.3 ships with the 2.4.18 kernel. However, while packaging Red Hat adds several patches to it. (If you want detailed information about what patches, you can look at the SPEC file contained in the SRPM.)

LIDS only supports the standard kernel source tree for 2.4.18. Due to this reason there are two methods of patching and compiling the kernel, “the easy way” and “the hard way.” It is preferred to keep the patches that Red Hat applied to the kernel since they applied them for a reason and did testing before releasing the package, however patching that kernel tree with LIDS might cause some other problems. For this reason, the author is going to go through patching both the standard kernel tree, and the Red Hat patched one. If the reader does not feel comfortable with the steps in “the hard way” then they should just follow the steps outline in “the easy way.” If both methods are taken, the Red

Hat kernel patched with LIDS should be tested and monitored closely for a while. If it exhibits any problems, then the 'standard' kernel with LIDS should be used.

First, we install the source RPM (SRPM) for the kernel.

```
#Insert the SRPMS disc2
[root@newsrver root]# mount /dev/cdrom /mnt/cdrom
[root@newsrver root]# rpm -ivh /mnt/cdrom/SRPMS/kernel-source-2.4.18-3.i386.rpm
Preparing... ##### [100%]
   1:kernel-source ##### [100%]
#Unmount the CD-ROM
[root@newsrver root]# umount /mnt/cdrom
#Eject the CD-ROM
[root@newsrver root]# eject
```

4.3.13.3.1 The Easy Way (The Standard Kernel)

Since the source tree installed by the installation process has already been patched, first we must retrieve the standard Linux kernel 2.4.18 source tree.

```
[root@newsrver root]# cd /usr/src
#Retrieve the kernel source
[root@newsrver src]# wget --passive-ftp
ftp://ftp.kernel.org/pub/linux/kernel/v2.4/linux-2.4.18.tar.gz
--16:55:57--  ftp://ftp.kernel.org/pub/linux/kernel/v2.4/linux-2.4.18.tar.gz
=> `linux-2.4.18.tar.gz'
Resolving ftp.kernel.org... done.
Connecting to ftp.kernel.org[204.152.189.113]:21... connected.
Logging in as anonymous ... Logged in!
==> SYST ... done.      ==> PWD ... done.
==> TYPE I ... done.   ==> CWD /pub/linux/kernel/v2.4 ... done.
==> PASV ... done.    ==> RETR linux-2.4.18.tar.gz ... done.
Length: 30,108,170 (unauthoritative)
(Status line)
16:57:38 (294.35 KB/s) - `linux-2.4.18.tar.gz' saved [30108170]

[root@newsrver src]# tar -zxf linux-2.4.18.tar.gz
#Rename the directory so we can keep track of it
[root@newsrver src]# mv linux linux-2.4.18-lids
[root@newsrver src]# cd linux-2.4.18
#Apply the patch
[root@newsrver linux-2.4.18]# patch -p1 <../lids-1.1.1r2-2.4.18/lids-1.1.1r2-
2.4.18.patch
```

Then the kernel needs to be configured. Instead of going through all of the questions, we can reuse the configuration file which was used by Red Hat to build the kernel which shipped with this and only configure the LIDS portion.

```
#This is the config file Red Hat uses in their SRPM for the i686 arch
[root@ linux-2.4.18-lids]# cp /usr/src/redhat/SOURCES/kernel-2.4.18-i686.config .config
#Rename this kernel so that it will be known as 2.4.18-lids
#Backup the original
[root@newsrver linux-2.4.18-lids]# mv Makefile Makefile.orig
#Use sed to add in -lids to the EXTRAVERSION variable
[root@newsrver linux-2.4.18-lids]# sed -e s'/^EXTRAVERSION =/EXTRAVERSION=-lids/'
Makefile.orig >Makefile
#Configure the LIDS options in the kernel
[root@newsrver linux-2.4.18-lids]# cat >>.config
CONFIG_LIDS=y
CONFIG_LIDS_MAX_INODE=1024
CONFIG_LIDS_MAX_SACL=1024
CONFIG_LIDS_MAX_OACL=1024
```

```

# CONFIG_LIDS_HANGUP is not set
CONFIG_LIDS_SA_EXEC_UP=y
# CONFIG_LIDS_NO_EXEC_UP is not set
CONFIG_LIDS_NO_FLOOD_LOG=y
CONFIG_LIDS_TIMEOUT_AFTER_FLOOD=60
CONFIG_LIDS_ALLOW_SWITCH=y
# CONFIG_LIDS_RESTRICT_MODE_SWITCH is not set
CONFIG_LIDS_MAX_TRY=3
CONFIG_LIDS_TTW_FAIL=3
# CONFIG_LIDS_ALLOW_ANY_PROG_SWITCH is not set
CONFIG_LIDS_RELOAD_CONF=y
# CONFIG_LIDS_PORT_SCAN_DETECTOR is not set
# CONFIG_LIDS_SA_THROUGH_NET is not set
# CONFIG_LIDS_DEBUG is not set
^D #Control-D

#Make dep
[root@news server linux-2.4.18-lids]# make dep
...large output omitted...
[root@news server linux-2.4.18-lids]# make
...large output omitted...
[root@news server linux-2.4.18-lids]# make bzImage
...large output omitted...
[root@news server linux-2.4.18-lids]# make modules
...large output omitted...
#Copy the modules into place(/lid/modules/2.4.18-lids)
[root@news server linux-2.4.18-lids]# make modules_install
...large output omitted...
#Copy the kernel to the /boot partition
[root@news server linux-2.4.18-lids]# cp arch/i386/boot/bzImage /boot/vmlinuz-2.4.18-lids
#Make the initrd file
[root@news server linux-2.4.18-lids]# mkinitrd /boot/initrd-2.4.18-lids.img 2.4.18-lids
#Add the new kernel entry to the grub file
[root@news server linux-2.4.18-lids]# cat >>/boot/grub/grub.conf
title Linux + LIDS (2.4.18-lids)
    root (hd0,0)
    kernel /vmlinuz-2.4.18-lids ro root=/dev/hda5
    initrd /initrd-2.4.18-lids.img
^D #Control-D

```

4.3.13.3.2 The Hard Way (Patching the Red Hat Linux kernel)

This time, the Red Hat patched source tree is used as a starting point. Copy the existing tree to a new directory, 2.4.18-3lids so that we can maintain a different tree and a different kernel and libraries. Apply the patch, and be prepared for some errors.

```

[root@news server root]# cd /usr/src
[root@news server src]# mkdir linux-2.4.18-3lids
[root@news server src]# cd linux-2.4.18-3
[root@news server linux-2.4.18-3]# tar -pcf - . | (cd ../linux-2.4.18-3lids/ ; tar -pxf - )
[root@news server linux-2.4.18-3]# cd ../linux-2.4.18-3lids/
[root@news server linux-2.4.18-3lids]# patch -p1 <../lids-1.1.1r2-2.4.18/lids-1.1.1r2-2.4.18.patch
#You will get asked about several patches which fail, simply press enter and
#answer 'Y' to "Skip this patch. A sample of what to expect:
patching file Documentation/Configure.help
Hunk #1 succeeded at 20411 (offset 464 lines).
can't find file to patch at input line 286
Perhaps you used the wrong -p or --strip option?
The text leading up to this was:
-----
|diff -Nru linux-2.4.18-ori/arch/alpha/config.in linux-2.4.18-lids-
|1.1.1r2/arch/alpha/config.in
|--- linux-2.4.18-ori/arch/alpha/config.in      Wed Nov 21 00:49:31 2001

```

```
|+++ linux-2.4.18-lids-1.1.1r2/arch/alpha/config.in Thu Apr 11 18:02:44 2002
-----
File to patch:
Skip this patch? [y]
Skipping patch.
...output omitted..
```

The next step of this process is to analyze the errors and manually fix them. There were several parts of the patch which failed. First, there were a bunch (the ones which prompt the user to be skipped) which are not for the architecture of the machine. Those can safely be skipped. The rest which are of a concern each have a “reject file,” showing what part of the patch failed. To find them:

```
#Use the find command to locate all patch reject files.
[root@newserver linux-2.4.18-3lids]# find . -type f -name *.rej -print
./arch/i386/config.in.rej
./fs/namei.c.rej
./fs/super.c.rej
./init/main.c.rej
./kernel/Makefile.rej
./kernel/fork.c.rej
./kernel/ksyms.c.rej
./net/socket.c.rej
```

4.3.13.3.2.1 arch/i386/config.in

```
[root@newserver i386]# cat config.in.rej
*****
*** 425,427 ****
    fi

    endmenu
--- 425,431 ----
    fi

    endmenu
+
+ # LIDS main menu read
+ source kernel/Config.in
+
```

Looking at the reject code and the file config.in, it looks as if the patch failed simply because the line numbers were off. To add this, simply concatenate into the file:

```
[root@newserver i386]# cat >>config.in
# LIDS main menu read
source kernel/Config.in
^D #Control-D
```

4.3.13.3.2.2 kernel/Makefile

Edit the kernel/Makefile and add in at line 53 the following bold lines so that it looks like:

```
obj-$(CONFIG_UID16) += uid16.o
obj-$(CONFIG_MODULES) += ksyms.o
```

```
obj-$(CONFIG_PM) += pm.o
obj-$(CONFIG_LIDS) += lids.o lids_logs.o rmd160.o
obj-$(CONFIG_LIDS_SA_THROUGH_NET) += klids.o lids_net.o
obj-$(CONFIG_IKCONFIG) += configs.o
obj-$(CONFIG_KALLSYMS) += kallsyms.o
```

4.3.13.3.2.3 fs/namei.c

This patch failed due to other changes to this file. Load the file in an editor and go to line 460 and add the following bold lines so that it looks like the following:

```
int link_path_walk(const char * name, struct nameidata *nd)
{
    struct dentry *dentry;
    struct inode *inode;
    int err, atomic;
    unsigned int lookup_flags = nd->flags;
#ifdef CONFIG_LIDS
    char *lids_file_name;

    lids_file_name = (char *)name;
#endif

    atomic = 0;
    if (lookup_flags & LOOKUP_ATOMIC)
        atomic = 1;
```

4.3.13.3.2.4 fs/super.c

This patch failed because this block of code no longer exists in this file. Since it appears that this patch only logged a failure condition, and does not actually change any functionality, a long time was not invested to locate if this functionality still exists (and needs enhanced logging.)

4.3.13.3.2.5 init/main.c

A function failed to get added to main.c due to a previous patch. To add this function to the bottom of the main.c file:

```
[root@newsrserver init]# cat >>main.c
#ifdef CONFIG_LIDS

/*
 * lids_setup , read lids info from the kernel.
 */
static int __init lids_setup(char *str)
{
    if(strncmp(str,"0",1) == 0)
        lids_load = 0;
    return 0;
}

__setup("lids=",lids_setup);
#endif
^D #Control-D
```


In addition to the reject file being applied, there was a portion of the patch which was applied with a 'fuzz' factor that was installed incorrectly. On line 629 of main.c there is a block that looks like:

```
#ifndef CONFIG_LIDS
/* init the ids file system
  lids_load=_lids_load;
  lids_local_on=1;
  lids_flags=0;
  lids_flag_raise(lids_flags,LIDS_FLAGS_LIDS_LOCAL_ON);
  if (lids_load)
    lids_flag_raise(lids_flags,LIDS_FLAGS_LIDS_ON);

    printk("Linux Intrusion Detection System %s %s
\n",LIDS_VERSION,lids_load==1?"started":"not started");
    if(lids_load) {
#ifdef CONFIG_LIDS_SA_THROUGH_NET
      lids_klids_init();
#endif
#ifdef CONFIG_LIDS_PORT_SCAN_DETECTOR
      lids_port_scanner_detector_init();
#endif
      lids_init();
    }
#endif
```

This should be removed from here. Apparently one of the patches applied by Red Hat in the packaging of the kernel in their distribution moved some of the code from main.c to a new file called do_mounts.c. Edit do_mounts.c and add the bold text at line 21 so that the file looks like:

```
#include <linux/nfs_mount.h>
#include <linux/minix_fs.h>
#include <linux/ext2_fs.h>
#include <linux/romfs_fs.h>
#include <linux/init.h>

#define BUILD_CRAMDISK

#ifndef CONFIG_LIDS
#include <linux/lids.h>
extern int _lids_load;
#endif

extern int get_filesystem_list(char * buf);
```

And, also add at line 857 the block which was deleted from main.c so that it reads:

```
sys_umount("/dev", 0);
sys_mount(".", "/", NULL, MS_MOVE, NULL);
sys_chroot(".");
mount_devfs_fs ();

#ifndef CONFIG_LIDS
/* init the ids file system */
lids_load=_lids_load;
lids_local_on=1;
lids_flags=0;
lids_flag_raise(lids_flags,LIDS_FLAGS_LIDS_LOCAL_ON);
if (lids_load)
    lids_flag_raise(lids_flags,LIDS_FLAGS_LIDS_ON);

printk("Linux Intrusion Detection System %s %s
```

```

\n",LIDS_VERSION,lids_load==1?"started":"not started");
    if(lids_load) {
#ifdef CONFIG_LIDS_SA_THROUGH_NET
        lids_klids_init();
#endif
#ifdef CONFIG_LIDS_PORT_SCAN_DETECTOR
        lids_port_scanner_detector_init();
#endif
        lids_init();
    }
#endif
}

```

4.3.13.3.2.6 kernel/fork.c

Another chunk of the patch which failed due to previous patches. Open fork.c in a text editor and go to line 721, and modify it by adding the bold lines so that it looks like:

```

    if (copy_sighand(clone_flags, p))
        goto bad_fork_cleanup_fs;
    if (copy_mm(clone_flags, p))
        goto bad_fork_cleanup_sighand;
#ifdef CONFIG_LIDS
    if (copy_lids_sys_acl(p))
        goto bad_fork_cleanup_lids;
LIDS_DBG("#### pid %i fork to pid %i\n",current->pid, p->pid);
#endif
    if (copy_namespace(clone_flags, p))
        goto bad_fork_cleanup_mm;
    retval = copy_thread(0, clone_flags, stack_start, stack_size, p, regs);

```

4.3.13.3.2.7 kernel/ksyms.c

To manually patch this file, append the following to the bottom:

```

[root@newsrver kernel]# cat >>ksyms.c

/* LIDS */
#ifdef CONFIG_LIDS
#ifdef CONFIG_MODULES
/* for modules which use capable() */
EXPORT_SYMBOL(lids_load);
EXPORT_SYMBOL(lids_cap_log);
EXPORT_SYMBOL(lids_log);
EXPORT_SYMBOL(lids_bind_checker);
EXPORT_SYMBOL(lids_cap_time_checker);
/* FIXME: add the above line to solve the insmod problem */
#ifdef CONFIG_LIDS_ALLOW_SWITCH
EXPORT_SYMBOL(lids_local_on);
EXPORT_SYMBOL(lids_local_pid);
EXPORT_SYMBOL(lids_local_off);
EXPORT_SYMBOL(tty_name);
#endif
#endif
#ifdef CONFIG_LIDS_HANGUP
EXPORT_SYMBOL(lids_first_time);
EXPORT_SYMBOL(tty_vhangup);
EXPORT_SYMBOL(lids_hangup_console);
#endif
#endif
#endif
^D #Control-D

```

4.3.13.3.2.8 net/socket.c

Reviewing this failed chunk, it was making a static function (sock_map_fd) not static if CONFIG_LIDS. A previous patch already made this always not static, so we can skip this chunk.

4.3.13.3.2.9 kernel/lids.c

Although there is no reject file for lids.c, there are two things which need to be corrected. First, on line 1069, change the variable type for port from int to time_t, so that it looks like:

```
static int lids_init_add_file ( char *buffer, _lids_data_t *data)
{
    char *p,*q;
    int error = -1;
    int is_default_rule = 0;
    int type,inherit;
    unsigned long int s_ino,o_ino;
    kdev_t s_dev,o_dev;
    time_t time[LIDS_TIME_ITEM][2];
    time_t port[LIDS_PORT_ITEM][2];

    p = memscan (buffer, ':', strlen(buffer));
    if (((unsigned long) (p-buffer)) == strlen(buffer)) goto exit;
```

Then, insert before line 41 the line “#include <linux/personality.h>” so that the file looks like:

```
#include <asm/page.h>
#include <asm/pgtable.h>

#include <linux/personality.h>
#include <asm/namei.h>

#include <linux/utime.h>
#include <linux/file.h>
#include <linux/fs.h>
```

One warning which should be noted is that it is possible that the patches introduced during the packaging of the kernel by Red Hat could of introduced some new code which should be protected by LIDS but is not. (Since we applied the patch and located code which moved, but it is possible that new code was introduced which should also be protected.)

Then the kernel needs to be configured. Instead of going through all of the questions, we can reuse the configuration file which was used by Red Hat to build the kernel which shipped with this and only configure the LIDS portion.

```
#This is the config file Red Hat uses in their SRPM for the i686 arch
[root@ linux-2.4.18-3lids]# cp /usr/src/redhat/SOURCES/kernel-2.4.18-i686.config .config
#Rename this kernel so that it will be known as 2.4.18-lids
#Backup the original
[root@newserver linux-2.4.18-3lids]# mv Makefile Makefile.orig
#Use sed to add in -lids to the EXTRAVERSION variable
```

```

[root@newsrserver linux-2.4.18-3lids]# sed -e 's/^EXTRAVERSION = -3custom/EXTRAVERSION=-3lib/' Makefile.orig >Makefile
#Configure the LIDS options in the kernel
[root@newsrserver linux-2.4.18-3lids]# cat >>.config
CONFIG_LIDS=y
CONFIG_LIDS_MAX_INODE=1024
CONFIG_LIDS_MAX_SACL=1024
CONFIG_LIDS_MAX_OACL=1024
# CONFIG_LIDS_HANGUP is not set
CONFIG_LIDS_SA_EXEC_UP=y
# CONFIG_LIDS_NO_EXEC_UP is not set
CONFIG_LIDS_NO_FLOOD_LOG=y
CONFIG_LIDS_TIMEOUT_AFTER_FLOOD=60
CONFIG_LIDS_ALLOW_SWITCH=y
# CONFIG_LIDS_RESTRICT_MODE_SWITCH is not set
CONFIG_LIDS_MAX_TRY=3
CONFIG_LIDS_TTW_FAIL=3
# CONFIG_LIDS_ALLOW_ANY_PROG_SWITCH is not set
CONFIG_LIDS_RELOAD_CONF=y
# CONFIG_LIDS_PORT_SCAN_DETECTOR is not set
# CONFIG_LIDS_SA_THROUGH_NET is not set
# CONFIG_LIDS_DEBUG is not set
^D #Control-D
#Make dep
[root@newsrserver linux-2.4.18-3lids]# make dep
...large output omitted...
[root@newsrserver linux-2.4.18-3lids]# make
...large output omitted...
[root@newsrserver linux-2.4.18-3lids]# make bzImage
...large output omitted...
[root@newsrserver linux-2.4.18-3lids]# make modules
...large output omitted...
#Copy the modules into place(/lid/modules/2.4.18-3lids)
[root@newsrserver linux-2.4.18-3lids]# make modules_install
...large output omitted...
#Copy the kernel to the /boot partition
[root@newsrserver linux-2.4.18-3lids]# cp arch/i386/boot/bzImage /boot/vmlinuz-2.4.18-3lids
#Make the initrd file
[root@newsrserver linux-2.4.18-3lids]# mkinitrd /boot/initrd-2.4.18-lids.img 2.4.18-3lids
#Add the new kernel entry to the grub file
[root@newsrserver linux-2.4.18-3lids]# cat >>/boot/grub/grub.conf
title Red Hat Linux + LIDS (2.4.18-3lids)
    root (hd0,0)
    kernel /vmlinuz-2.4.18-3lids ro root=/dev/hda5
    initrd /initrd-2.4.18-3lids.img
^D #Control-D

```

4.3.13.4 Changing the default kernel

The configuration file for grub (the bootloader) must be configured to use the new kernel. In the file /boot/grub/grub.conf there is a line which reads “default=<number>”. The <number> should be changed to indicate which kernel (in order in grub.conf, starting with the number 0) should be used. For instance, if you installed both “the easy way” and “the hard way” you now have three kernels in there. To boot from vmlinuz-2.4.18-3lids (the last one (the 3rd one) in the grub.conf file) you would set “default=2.”

4.3.13.5 Building and Installing the LIDS Tools

Installing the lids tools is a straight forward process. The only change which must be made is to disable version checking since the lids programs expect to be run in 2.4.18, and when the configure program checks the Red Hat modified kernel source, it gets back a version of 2.4.18-3.

```

[root@newsrerver root]# cd /usr/src/lids-1.1.1r2-2.4.18
#Run the configure command, specific the kernel source directory and disable the version
#checking since the redhat tree modifies the version.h file (appends "-3")
[root@newsrerver lids-1.1.1r2-2.4.18]# ./configure KERNEL_DIR=/usr/src/linux-2.4.18-3lids
--disable-versions-checks
...output omitted...
#Run make to build the tools
[root@newsrerver lids-1.1.1r2-2.4.18]# make
...output omitted...
#Run "make install" to have the binaries installed
#it will prompt for the creation of the LIDS password at the end.
[root@newsrerver lids-1.1.1r2-2.4.18]# make install
...some output omitted..
MAKE PASSWD
enter new password: <enter-password>
reenter new password: <re-enter-password>
wrote password to /etc/lids/lids.pw
make[3]: Leaving directory `/usr/src/lids-1.1.1r2-2.4.18'
make[2]: Leaving directory `/usr/src/lids-1.1.1r2-2.4.18'
make[1]: Leaving directory `/usr/src/lids-1.1.1r2-2.4.18'

```

4.3.13.6 Configuring LIDS

Now, each program that requires privileges to a protected file or any of the privileged ‘capabilities’ will need to have the privileges explicitly granted to them. (For information about the capabilities in Linux, refer to the file /etc/lids/lids.cap for details.) To facilitate configuring LIDS, make a directory under /root where a script can be stored which will reconfigure LIDS.

```

#Make the directory where we will store the script to create the LIDS configuration
[root@newsrerver /root ]# mkdir /root/LIDS

```

In the chapter “Files” there is a file called ‘createlids.sh’ which should be placed into this directory and then run. This will create the configuration file for lids (/etc/lids/lids.conf) and reload that file.

With LIDS, one can protect a file or directory, or give a particular program permissions Set the location of the binary, and clear the existing rules. There are three main methods to using lidsconf to create a rule:

- 1) Protecting an object.
- 2) Granting a subject permission to an object
- 3) Granting a subject a capability.

```

#Example of protecting an object (Deny all access to /etc/shadow)
[root@newsrerver /]# /sbin/lidsconf -A -o /etc/shadow -j DENY
#Example of granting a subject permission to an object (Allow login to read /etc/shadow)
[root@newsrerver /]# /sbin/lidsconf -A -s /bin/login -o /etc/shadow -j READONLY
#Example of granting a subject a capability (Allow sendmail to bind to a port)
[root@newsrerver /]# /sbin/lidsconf -A -s /usr/sbin/sendmail -o CAP_NET_BIND_SERVICE -j
GRANT

```

Another feature, which is important is the inheritance level. This allows you to grant a privilege to a process and allow for the children of that process to inherit the privilege as

well. This can be particularly important when allowing specific scripts (such as cron scripts) to perform certain activities.

To configure LIDS, the file createlids.sh, does the following:

- 1) Protects the LIDS files and directories themselves.
- 2) Defaults most partitions to read-only.
- 3) Allow lock files and log files to be written to.
- 4) Allow processes responsible for shutting down the machine to have the needed permissions.
- 5) Allow apache to change user/group ID and bind to ports 80 and 443.
- 6) Allow SSH to access it's files, and chown files, change user/group id, bind to port 22 and override DAC access.
- 7) Allow bind (named) to chroot, bind to port 53, set user/group id, override DAC access.
- 8) Allow sendmail to write to the needed directories, set user/group id, access the protected files it needs to.
- 9) Allow xinetd to start sendmail (for smtps) and POP3 (and pop3s) this requires chrooting, binding to the necessary ports.
- 10) Allow ipop to be able to access the needed directories.
- 11) Allow snort to chroot, and change user/group id, access the network, as well as write to the necessary directories.
- 12) Allow login to function properly by granting it the proper capabilities and access the required files.
- 13) Allow syslog to write to the necessary dev directories (for the chrooted environments).
- 14) Allow the startup scripts to do what they need.
- 15) Allow the cron scripts to function properly.

Since /etc/mtab has a different inode each time and it is not advisable to give write permissions to /etc, remove /etc/mtab, and create a symbolic link from /proc/mount.

```
#Create a symlink from /proc/mount to /etc/mtab
[root@newserver /]# ln -s /proc/mount /etc/mtab
```

After installing and configuring LIDS, the system should be rebooted.

```
#Reboot the system
[root@newserver /]# shutdown -r now
```

To verify that LIDS is working properly, after the machine has rebooted, log in and try to read a protected file (e.g. /etc/shadow), then disable LIDS on this terminal, and make sure that the file can be read.

```
[root@newserver root]# cat /etc/shadow
cat: /etc/shadow: No such file or directory
#Then disable lids and cat that same file
```

```
[root@news server root]# /sbin/lidsadm -S -- -LIDS
SWITCH
enter password: <enter password>
No global capabilities have changed.
[root@news server root]# cat /etc/shadow
...output omitted...
#Re-enable LIDS on this terminal
[root@news server root]# /sbin/lidsadm -S -- -LIDS
SWITCH
enter password: <enter password>
No global capabilities have changed.
```

4.3.14 Tripwire

4.3.14.1 Background

Tripwire is a host based intrusion detection system. Tripwire maintains a database of information about files which are important, and re-checks these files (through cron) and notifies the administrator if the integrity of any of the files has been compromised.

Tripwire is installed in the default install of Red Hat Linux 7.3, however it needs to be configured, and initialized.

4.3.14.2 Installation and Configuration

If LIDS has already been installed and the system rebooted, LIDS will need to be de-activated on the terminal which the work is done on.

The program *twinstall.sh* is run to create the site key file, the local key file, and the default policy file.

```
#Initial configuration
root@news server root]# /etc/tripwire/twinstall.sh

-----
The Tripwire site and local passphrases are used to
sign a variety of files, such as the configuration,
policy, and database files.

Passphrases should be at least 8 characters in length
and contain both letters and numbers.

See the Tripwire manual for more information.

-----
Creating key files...

(When selecting a passphrase, keep in mind that good passphrases typically
have upper and lower case letters, digits and punctuation marks, and are
at least 8 characters in length.)

Enter the site keyfile passphrase: <enter site password>
Verify the site keyfile passphrase: <re-enter site password>
Generating key (this may take several minutes)...Key generation complete.

(When selecting a passphrase, keep in mind that good passphrases typically
have upper and lower case letters, digits and punctuation marks, and are
at least 8 characters in length.)

Enter the local keyfile passphrase: <enter local password>
```

```

Verify the local keyfile passphrase: <re-enter local password>
Generating key (this may take several minutes)...Key generation complete.

-----

Signing configuration file...
Please enter your site passphrase:
Wrote configuration file: /etc/tripwire/tw.cfg

A clear-text version of the Tripwire configuration file
/etc/tripwire/twcfg.txt
has been preserved for your inspection. It is recommended
that you delete this file manually after you have examined it.

-----

Signing policy file...
Please enter your site passphrase: <enter site password>
Wrote policy file: /etc/tripwire/tw.pol

A clear-text version of the Tripwire policy file
/etc/tripwire/twpol.txt
has been preserved for your inspection. This implements
a minimal policy, intended only to test essential
Tripwire functionality. You should edit the policy file
to describe your system, and then use twadmin to generate
a new signed copy of the Tripwire policy.

```

Next, the policy file needs to be modified. There are numerous files in the default policy file which do not exist on the machine which was just installed. Load up /etc/tw/twcfg.txt and comment out (by placing a '#' before the entry) the following files:

```

/sbin/cardmgr           /sbin/mount.ncpfs     /var/lock/subsys/postgresql
/proc/scsi             /sbin/mount.smb       /var/lock/subsys/pxe
/usr/sbin/fixrmtab     /sbin/mount.smbfs     /var/lock/subsys/radvd
/sbin/accton           /sbin/netconf          /var/lock/subsys/rarpd
/sbin/busybox          /sbin/userconf        /var/lock/subsys/reconfig
/sbin/busybox.anaconda /sbin/uucpconf         /var/lock/subsys/rhnsd
/sbin/ftl_check        /sbin/vregistry       /var/lock/subsys/ripd
/sbin/ftl_format      /var/lock/subsys/ipchains /var/lock/subsys/ripngd
/sbin/mkpv             /var/lock/subsys/ipvsadm /var/lock/subsys/routed
/sbin/pcinitrd         /var/lock/subsys/portmap /var/lock/subsys/rstatd
/sbin/scsiinfo         /var/lock/subsys/ypbind /var/lock/subsys/rusersd
/sbin/adjtimex         /var/lock/subsys/linuxconf /var/lock/subsys/rwalld
/sbin/nuactlun        /var/lock/subsys/amd   /var/lock/subsys/rwhod
/sbin/nuscscitcpd     /var/lock/subsys/apmd  /var/lock/subsys/smb
/sbin/sndconfig        /var/lock/subsys/arpwatch /var/lock/subsys/snmpd
/sbin/ifport          /var/lock/subsys/autofs /var/lock/subsys/squid
/sbin/ifuser           /var/lock/subsys/bcm5820 /var/lock/subsys/tux
/sbin/ipvsadm          /var/lock/subsys/bgpd  /var/lock/subsys/tWnn
/sbin/ipvsadm-restore /var/lock/subsys/bootparamd /var/lock/subsys/ups
/sbin/ipvsadm-save    /var/lock/subsys/canna /var/lock/subsys/vncserver
/sbin/ipx_configure   /var/lock/subsys/Cwnn  /var/lock/subsys/wine
/sbin/ipx_interface   /var/lock/subsys/dhcpd /var/lock/subsys/xfs
/sbin/iptx_internal_net /var/lock/subsys/firewall /var/lock/subsys/ypasswdd
/sbin/iwconfig        /var/lock/subsys/freeWnn /var/lock/subsys/ypserv
/sbin/iwgetid         /var/lock/subsys/gated /var/lock/subsys/ypxfrd
/sbin/iwlist          /var/lock/subsys/gpm   /var/lock/subsys/zebra
/sbin/iwpriv          /var/lock/subsys/identd /etc/conf.d/linuxconf
/sbin/iwspy           /var/lock/subsys/innd  /etc/samba/smb.conf
/sbin/mgetty          /var/lock/subsys/irda  /sbin/sfxload
/sbin/vgetty          /var/lock/subsys/iscsi /bin/aumix-minimal
/sbin/linuxconf       /var/lock/subsys/isdn  /bin/zsh
/sbin/linuxconf-auth  /var/lock/subsys/junkbuster /bin/zsh-4.0.2
/sbin/remadmin        /var/lock/subsys/kadmin /bin/ksh
/sbin/cardctl         /var/lock/subsys/kprop /root/mail

```


/sbin/dump_cis	/var/lock/subsys/krb524	/root/Mail
/sbin/ide_info	/var/lock/subsys/krb5kdc	/root/.amandahosts
/sbin/isapnp	/var/lock/subsys/kudzu	/root/.addressbook.lu
/sbin/lspnp	/var/lock/subsys/kWnn	/root/.addressbook
/sbin/pack_cis	/var/lock/subsys/ldap	/root/.elm
/sbin/prpdump	/var/lock/subsys/lpd	/root/.esd_auth
/sbin/probe	/var/lock/subsys/mars_nwe	/root/.gnome_private
/sbin/pump	/var/lock/subsys/mcscserv	/root/.gnome-desktop
/sbin/genksyms.old	/var/lock/subsys/mysqld	/root/.gnome
/sbin/sash	/var/lock/subsys/nfs	/root/.ICEauthority
/sbin/askrunlevel	/var/lock/subsys/nfslock	/root/.mc
/sbin/fixperm	/var/lock/subsys/nscd	/root/.pinerc
/sbin/fsconf	/var/lock/subsys/ntpd	/root/.sawfish
/sbin/mailconf	/var/lock/subsys/ospf6d	/root/.Xauthority
/sbin/modemconf	/var/lock/subsys/ospfd	/root/.xauth
/sbin/mount.ncp	/var/lock/subsys/pcmcia	/root/.xsession-errors

Then, add in the section labeled "Critical system bootfiles" (omit any kernels that have not been installed):

```

/boot/vmlinuz-2.4.18-3           -> $(SEC_CRIT) ;
/boot/vmlinuz-2.4.18-3lids      -> $(SEC_CRIT) ;
/boot/vmlinuz-2.4.18-1lids      -> $(SEC_CRIT) ;
/boot/initrd-2.4.18-3.img       -> $(SEC_CRIT) ;
/boot/initrd-2.4.18-3lids.img   -> $(SEC_CRIT) ;
/boot/initrd-2.4.18-1lids.img   -> $(SEC_CRIT) ;

```

Change the hostname at the top of the file to the correct hostname. (Example HOSTNAME=newserver.markoram.com)

Create the database:

```

[root@newserver root]# tripwire --init
Please enter your local passphrase: <enter local password>
Parsing policy file: /etc/tripwire/tw.pol
Generating the database...
*** Processing Unix File System ***
Wrote database file: /var/lib/tripwire/newserver.markoram.com.twd
The database was successfully generated.

```

4.3.15 BIOS

Since the server will be located offsite and collocated at an ISPs facilities, physical security is under the control of the ISP. As such, any reasonable measures which can be taken to prevent the unauthorized manipulation of the server should be taken. The issues which need to be addressed are:

- Automatic booting – The machine should automatically return to a power on state if it is unplugged.
- Boot Order / Media – The machine should always boot from the hard drive, not a floppy or the CD-ROM drive.
- The BIOS information should not be able to be changed by an unauthorized person.

To enforce the above, the steps are as follows:

- 1) Reboot the server.
- 2) While at the POST screen press the 'del' key to enter the BIOS configuration
- 3) Select the "Security" tab.
 - a. Set "Setup Password" to a non-guessable yet easy to remember password.
 - b. Set "User Setup Access" to "None."
- 4) Select the "Boot" tab.
 - a. Set the "First Boot Device" to "Hard Drive."
 - b. Set "Restore on AC/Power Loss" to "Power On."
- 5) Select the "Exit" tab and select "Exit Saving Changes."

While the above would not stop a maliciously minded person from opening the machine and disabling the BIOS password via a jumper on the motherboard it is the best that can be done under the circumstances. It should also be verified with the ISP's operations that they will either provide an enclosed cage for the server or control physical access to it so that only authorized personnel can physically access it.

© SANS Institute 2000 - 2002, Author retains full rights.

5 Maintenance

5.1 Upgrades

5.1.1 Disabling LIDS on a tty

In general any modifications to the system will require that LIDS be disabled on the terminal through which the administrator is connected. To disable lids:

```
#Disable LIDS on this tty
[root@newsrver /]# /sbin/lidsadmin -S -- -LIDS
SWITCH
enter password: <enter LIDS password here>
```

After completing any maintenance LIDS is re-enabled by:

```
#Enable LIDS
[root@newsrver /]# /sbin/lidsadm -S -- -LIDS
SWITCH
enter password: <enter LIDS password here>
```

5.1.2 Red Hat Packages

Red Hat makes updates available for the software which makes up their Operating System. These updates are available through FTP from their public FTP site. For the English version of Red Hat Linux 7.3, the URL is: <ftp://updates.redhat.com/7.3/en/> and there are separate directories for each of the architectures underneath that. For this system, we care about the i386, i686, and the noarch directories.

To facilitate keeping track of which patches need to be downloaded and installed, the author has written a small script *checkupdates.pl* which is available in the chapter 'Files.' This program will connect the Red Hat FTP site (<ftp://updates.redhat.com/>) read in what packages have updates, compare this to what is currently installed on the system, and download any which are in need of an update. The program also supports checking multiple install areas, for example in this case both the root system and the /home/EMAIL/root chrooted environment.

Before running *checkupdates.pl* for the first time, the Net::FTP perl library will need to be installed. Net::FTP is part of the perl-libnet package which is located on disc1 of the Installation CDs. To install this:

```
#Insert disc1 and mount it
[root@newsrver /]# mount /dev/cdrom /mnt/cdrom
#Install the RPM
[root@newsrver root]# rpm -ivh /root/perl-libnet-1.0901-17.i386.rpm
Preparing... ##### [100%]
 1:perl-libnet ##### [100%]
```

To run it, make the directory `/root/updates` and then run the program. After the program has completed there will be `/root/updates/root` and `/root/updates/EMAIL` directories which will contain the updates for each respective environment. They can then be installed via `rpm` (`rpm -Uvh <filename>` for the root environment or `rpm -Uvh <filename> --root /home/EMAIL/root` for the chrooted email environment.) After installing the updates, the `/root/updates` directory should be deleted so as not to confuse things when the program is run next time.

To install any updates, LIDS will need to be disabled. Also, the tripwire database will need to be updated.

5.1.3 Snort

As newer versions of snort become available, they will be up for downloading in the same place that it was downloaded from in the Installation section. The instructions for upgrading are to simply download the new RPM, and use the `rpm` command to upgrade to the new version:

```
#Shutdown the current running version
[root@newsrver /]# /etc/rc.d/init.d/snortd stop
Stopping snort: [ OK ]
#Upgrade
[root@newsrver /]# rpm -ivh <path-to-rpm/rpmname>
#Restart the service
[root@newsrver /]# /etc/rc.d/init.d/snortd start
Starting snort: eth0: Setting promiscuous mode.
[ OK ]
```

To update only the signatures, download the tar.gz file from [snort.org](http://www.snort.org) and untar them into the chrooted directory and restart the snort service.

```
#Change directories to the chrooted environment
[root@newsrver /]# cd /home/snort/etc
#Retrieve the updated signatures from http://www.snort.org/dl/signatures/
[root@newsrver etc]# wget http://www.snort.org/dl/signatures/snortrules.tar.gz
#Untar the new files (the files all exist in a directory named rules)
[root@newsrver etc]# tar -zxf snortrules.tar.gz
#Shutdown the current running version
[root@newsrver etc]# /etc/rc.d/init.d/snortd stop
Stopping snort: [ OK ]
#Removing the old rules directory (called snort)
[root@newsrver etc]# rm -fr snort
#Rename the 'rules' directory to 'snort'
[root@newsrver etc]# mv rules snort
#Remove the tar.gz file
[root@newsrver etc]# rm snortrules.tar.gz
```

5.1.4 LIDS

Upgrades to the LIDS system will be able to be found from www.lids.org. The re-installation/upgrading of LIDS will require following the same general steps outlined in the Installation chapter, and following any additional directions provided by the updated version of lids.

5.1.5 Tripwire

Anytime any files or directories being monitored by tripwire are modified, the tripwire database will need to be updated. To perform an update, after tripwire has run its check, you can issue the command:

```
[root@newserver ~]# /usr/sbin/tripwire --update --twrfile  
/var/lib/tripwire/report/<name>.twr
```

Full information and instructions about performing an update after an integrity check can be located in the Red Hat 7.3 Reference manual in the tripwire guide section at:

<http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/ref-guide/s1-tripwire-update-db.html>

5.2 Keeping Informed

5.2.1 Reading the root email

One of the most obvious things and most helpful that can be done is simply monitoring the root email. The output of the cron jobs including tripwire and logwatch are emailed every night.

5.2.2 Sysstat

The sysstat tools can be used to monitor the performance of the system. This can be crucial in knowing the availability of the services. The *sar* command can be used to view the history of the performance of the system over time using the data stored in the */var/log/sa* directory.

5.2.3 External sources

Keep in touch with what is going on in the security and the Red Hat communities.

Red Hat keeps a list of their security errata located at: <http://rhn.redhat.com/errata/rh73-errata-security.html>.

There are several mailing lists which available from www.securityfocus.com:

- Bugtraq
- Focus-Linux
- Vuln-Dev

These can be subscribed too by following the directions posted at <http://online.securityfocus.com/cgi-bin/sfonline/subscribe.pl>.

Several websites which contain informative information:

- www.packetstormsecurity.org
- www.securityfocus.com

6 Verification

6.1 Verification of Applications

In the Installation chapter instructions on how to test were provided for each part as it was configured. After the entire installation has been concluded, it is highly recommended to repeat each of those tests again.

6.1.1 SSH

To test that it is running and answering network connections:

```
#Telnet to port 22 (ssh) to test that it answers
[root@newsrvr root]# telnet 0 22
Trying 0.0.0.0...
Connected to 0.
Escape character is '^'.
SSH-2.0-OpenSSH_3.1p1
```

To fully test this, log into the server from the client where the key was generated from which is allowed in as root.

```
#Example of logging into the server from another unix machine.
[root@newsrvr root]# ssh root@192.168.17.220
The authenticity of host '192.168.17.220 (192.168.17.220)' can't be established.
RSA key fingerprint is 67:b0:c8:c5:dd:d8:c0:3f:10:c8:6c:3c:ec:4c:7a:88.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.17.220' (RSA) to the list of known hosts.
Enter passphrase for key '/root/.ssh/id_dsa': <enter passphrase>
Last login: Mon May 20 10:32:02 2002 from 192.168.17.235
#Log out of this shell
[root@newsrvr root]# exit
```

6.1.2 Bind / DNS

To test that DNS is operating properly, we will perform a few queries.

```
#Test that we can resolve some records which are local in our server
[root@newsrvr root]# nslookup -sil www.markoram.com
Server:          127.0.0.1
Address:         127.0.0.1#53

Name:   www.markoram.com
Address: 192.168.17.220

#Test a lookup which will require contacting other DNS servers.
[root@newsrvr root]# nslookup -sil www.giac.com
Server:          127.0.0.1
Address:         127.0.0.1#53

Non-authoritative answer:
Name:   www.giac.com
Address: 64.29.19.73
Name:   www.giac.com
Address: 66.33.61.127
Name:   www.giac.com
Address: 64.29.19.71
```

6.1.3 Email

For these tests we will use the test account which was created earlier.

6.1.3.1 SMTP

For this test, we are emulating an external person sending email to a local user. (test)
Substitute a valid email address which is under your control for user@somehost.com.

```
[root@news server root]# telnet 0 mail
Trying 0.0.0.0...
Connected to 0.
Escape character is '^'.
220 news server.markoram.com ESMTP Sendmail 8.11.6/8.11.6; Mon, 20 May 2002 10:42:00 -0400
ehlo domaingo.com
250-news server.markoram.com Hello localhost [127.0.0.1], pleased to meet you
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-SIZE
250-DSN
250-ONEX
250-ETRN
250-XUSR
250 HELP
mail from:user@somehost.com
250 2.1.0 user@somehost.com... Sender ok
rcpt to:test@markoram.com
250 2.1.5 test@markoram.com... Recipient ok
data
354 Enter mail, end with "." on a line by itself
Testing testing, testing.
.
250 2.0.0 g4KEgAw00970 Message accepted for delivery
quit
221 2.0.0 news server.markoram.com closing connection
```

Verify that the mail did get delivered, and verify that the file is in the chrooted EMAIL environment.

```
#Use the mail command w/ the -f argument to specify an alternate mailbox
[root@news server root]# mail -f /home/EMAIL/root/var/spool/mail/test
Mail version 8.1 6/6/93. Type ? for help.
"/home/EMAIL/root/var/spool/mail/test": 1 message 1 new [Read only]
>N 1 user@somehost.com Mon May 20 10:42 11/413
& page 1
Message 1:
From user@somehost.com Mon May 20 10:42:30 2002
Date: Mon, 20 May 2002 10:42:17 -0400
From: user@somehost.com

Testing testing, testing.

& x
```

The mail is there, so we know that it did get delivered and that it was delivered within the chrooted environment. Since we exited the *mail* program with the 'x' option, the mailbox is left as it was so that we can test with POPS as well.

It should also be verified that this server will not relay spam. To do this, pick an email address not handled by this server and connect from a **different machine**:

```

#On a different machine we connect and try to relay
[root@othermachine /]# telnet 192.168.17.220 mail
Escape character is '^'.
220 newserver.markoram.com ESMTP Sendmail 8.11.6/8.11.6; Mon, 20 May 2002 10:46:54 -0400
ehlo markoram.com
250-newserver.markoram.com Hello [192.168.17.220], pleased to meet you
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-SIZE
250-DSN
250-ONEX
250-ETRN
250-XUSR
250 HELP
mail from:test@markoram.com
250 2.1.0 test@markoram.com... Sender ok
rcpt to:user@somehost.com
550 5.7.1 user@somehost.com... Relaying denied.
quit
221 2.0.0 newserver.markoram.com closing connection

```

6.1.3.2 SMTPS

To test SMTP we want to deliver email to an outside user after authenticating.

```

#Connect to the SMTPS server and make sure that it supports LOGIN AUTH
[root@newserver root]# openssl s_client -connect 0:465
CONNECTED (00000003)
depth=0 /C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
verify error:num=18:self signed certificate
verify return:1
depth=0 /C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
verify return:1
---
Certificate chain
 0 s:/C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
  i:/C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
---
Server certificate
-----BEGIN CERTIFICATE-----
MIIDDzCCAnigAwIBAgIBADANBgkqhkiG9w0BAQQFADBpMQswCQYDVQQGEwJVUzET
MBEGA1UECBMKTmV3IEplcnNleTESMBAGA1UEBxMJU29tZSBDbXR5MRUwEwYDVQQK
EwxTb211IENvbXBhbnkxGjAYBgNVBAMTEW1haWwubWYya29yYW0uY29tMB4XDTAy
MDUxMDAzMTAyOV0xMDUxMDAzMTAyOV0wTELMAkGA1UEBhMCVVMxEzARBGNV
BAGTCk5ldyBkZXJzZXkxZjAQBGNVBAcTCVNvbWUgQ210eTEVMBMGAlUEChMMU29t
ZSBDb21wYW55MR0wGAYDVQQDExFtYWlsLm1hcmtvcmlmFtLmNvbTCBnzANBgkqhkiG
9w0BAQEFAAOBjQAwYgkCgYEA0Wa9WeJXRgGAlMG6Nbg/0E5gaY9XfC2Wqgis0ESo
KyY9NCPK5x7HEfTINYQqEVF3ZsxlMaZchPXsVxNefcoW+msOpZWKTtXF6iHrSNq
kYsMRcyvX+ZuSxCdWsDM3vdAtulEwWpF4Iry4eN6MD6tiY9kwdgue6itiau03xUL
+fKCAwEAaAObxjCBwzAdBgNVHQ4EFgQUqvNid76xhsstGx27x9H2GtqS1QcwgZMG
AlUdIwSBizCBIUAUqvNid76xhsstGx27x9H2GtqS1QehbaRrMGkxCzAJBgNVBAYT
AlVTMRMwEQYDVQQIEWpOZXcgSmVyc2V5MRIwEAYDVQQHEw1Tb211IENpdHkxFTAT
BgNVBAoTDFNvbWUgQ29tGFueTEaMBGAlUEAxMRbWYyY29tY29tY29tY29tY29tY29t
AQAwDAYDVROTBAAUwAwEB/zANBgkqhkiG9w0BAQQFAAOBgQBN9cBkio2Z/Xu7QQgU
a9zXNJOEzphKKkkc1KXroPi88o6uUTtaAcy0eVuzLxetQP1wdKIqAD/MD1hALqtN
2JgzhSEgGcmt7X+VH834ZTzD0VRuoZRR3L6vPlbagjqhd27h5gnsMQUX3OUVbv
WMUeh4tLeVnga69Npq9C9NNadZQ==
-----END CERTIFICATE-----
subject=/C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
issuer=/C=US/ST=New Jersey/L=Some City/O=Some Company/CN=mail.markoram.com
---
No client certificate CA names sent
---
SSL handshake has read 941 bytes and written 314 bytes
---
New, TLSv1/SSLv3, Cipher is DES-CBC3-SHA

```



```

Server public key is 1024 bit
SSL-Session:
  Protocol   : TLSv1
  Cipher     : DES-CBC3-SHA
  Session-ID: DC2EE5A759892D47558BA0B3B2B154F33A5FC7749AA44B8D42914A1C23D57871
  Session-ID-ctx:
  Master-Key:
D9D7C71C7C1310CBAF23BCC2F19E1FAFC56E429139A42096BF842239CA97373402896124B3AC1B6BA662C8FFA0129D5
4
  Key-Arg    : None
  Start Time: 1021907182
  Timeout    : 300 (sec)
  Verify return code: 18 (self signed certificate)
---
220 newserver.markoram.com ESMTP Sendmail 8.11.6/8.11.6; Mon, 20 May 2002 11:06:22 -0400
ehlo markoram.com
ehlo markoram.com
250-newserver.markoram.com Hello root@localhost, pleased to meet you
250-ENHANCEDSTATUSCODES
250-8BITMIME
250-SIZE
250-DSN
250-ONEX
250-ETRN
250-XUSR
250-AUTH LOGIN   ###←-- it does support LOGIN
250 HELP
quit

```

To fully test this, test sending a piece of email through the system from a desktop. To configure Microsoft Outlook, add an email account like you normally would (Tools->Accounts->Add), using the test account created previously. After completing that, go back into the properties of that email account, go to the “Servers” tab and check “My server requires authentication.” Then go to the “Advanced Tab” and check “This server requires a secure connection (SSL)” for both POP and SMTP. Also, make sure that the ports for SMTP and POP are 465 and 995 respectively.

6.1.3.3 POPs

To retest POPs, repeat the same procedure outline in the Installation chapter, or if an email client has been properly configured from the previous step, use that to send a test email to the test account and retrieve it.

6.1.4 Apache

Verify that the webserver is functioning properly, either by re-performing the tests outline in the Installation chapter, or by loading the site in a web browser from a client machine. Also verify that the secure webserver is loaded by testing https as well.

- <http://192.168.17.220/>
- <https://192.168.17.220/>

6.1.5 Syslog

To ensure that syslog is properly logging events from the chrooted environment, look through /var/log/mail log for events from the chrooted EMAIL. Entries from the previous testing such as the following should be seen.

```
May 20 10:42:29 newserver sendmail[970]: g4KEgAw00970: from=user@somehost.com, size=26, class=0, nrcpts=1, msgid=<200205201442.g4KEgAw00970@newserver.markoram.com>, proto=ESMTP, daemon=MTA, relay=localhost [127.0.0.1]
May 20 10:42:30 newserver sendmail[972]: g4KEgAw00970: to=test@markoram.com, delay=00:00:13, xdelay=00:00:01, mailer=local, pri=30035, dsn=2.0.0, stat=Sent
May 20 10:46:45 newserver sendmail[977]: NOQUEUE: localhost [127.0.0.1] did not issue MAIL/EXPN/VRFY/ETRN during connection to MTA
May 20 10:47:14 newserver sendmail[980]: g4KEl6w00980: from=test@markoram.com, size=0, class=0, nrcpts=1, proto=ESMTP, daemon=MTA, relay=[192.168.17.220]
May 20 10:47:41 newserver sendmail[982]: g4KEl6w00982: ruleset=check_rcpt, arg1=oram@domaingo.com, relay=[192.168.17.100], reject=550 5.7.1 user@somehost.com...
Relaying denied. IP name lookup failed [192.168.17.100]
```

To ensure that events from the chrooted named environment are being logged, look through /var/log/messages for named entries:

```
#Look for named entires in the /var/log/messages log file to ensure that syslog #is logging events from the named chrooted environment.
[root@newserver /]# grep named /var/log/messages
May 20 13:42:10 newserver named[663]: zone 0.0.127.in-addr.arpa/IN: loaded serial 1997022701
May 20 13:42:10 newserver named[663]: zone markoram.com/IN: loaded serial 1997022700
May 20 13:42:10 newserver named[663]: zone localhost/IN: loaded serial 42
May 20 13:42:11 newserver named[663]: running
```

6.1.6 Snort

To verify that snort is working and capturing events, look in /home/snort/var/log/snort. Shortly after the machine has been connected to the Internet, entires will begin showing up in the file ‘alerts.’

6.1.7 Netfilter/IPTables

There are two things to test in regards to the filtering.

- 1) That the allowed traffic works – This is being tested as we test each service.
- 2) That the denied (not explicitly allowed) traffic is truly denied – This was tested after installing the filter rules, by running the telnet daemon and establishing that the filters were working by attempting to telnet to the machine. This test can be repeated to ensure that the filters are indeed denying unauthorized traffic.

6.1.8 TCP Wrappers

To test the TCP wrappers, comment out the entry (by adding a ‘#’ before the line) for ipop3d in /etc/hosts.allow and then try connecting. The attempt will be refused, and the attempt will be logged in /var/log/secure.

```
#Try connecting to the POPS port
```

```
[root@newsrserver snort]# telnet 0 995
Trying 0.0.0.0...
Connected to 0.
Escape character is '^]'.
Connection closed by foreign host.
#The connection was refused.
#Verify that the attempt was logged:
[root@newsrserver snort]# tail -2 /var/log/secure
May 20 11:45:33 newsrserver xinetd[1272]: FAIL: pop3s libwrap from=127.0.0.1
May 20 11:45:33 newsrserver xinetd[718]: EXIT: pop3s pid=1272 duration=0(sec)
```

Do not forget to uncomment the `ipop3d` line in `/etc/hosts.deny` after completing this test.

6.1.9 Unneeded Services

To verify this, repeat the steps from the Install section, including doing a `ps` and `netstat`.

6.1.10 Set User ID and Set Group ID binaries

To verify this, repeat the find commands from the Install section, and verify that the permissions match those to which they were set in the install section.

6.1.11 LIDS

LIDS was installed for two main reasons, to limit access to files unless explicitly granted, and to limit the capabilities unless explicitly granted.

6.1.11.1 Testing File Access

From a shell (which does not have LIDS disabled) try to access a protected file.

```
#Access a protected file
[root@newsrserver /]# cat /etc/shadow
cat: /etc/shadow: No such file or directory
```

6.1.11.2 Testing a capability

From a shell (which does not have LIDS disabled) try to perform a `chroot` which requires the `CAP_SYS_CHROOT` capability.

```
#Find a target program to run in a chrooted environment
[root@newsrserver /]# ls -al /home/EMAIL/root/bin/ls
-rwxr-xr-x 1 root root 46888 Mar 24 20:23 /home/EMAIL/root/bin/ls
#Try to chroot and run it
[root@newsrserver /]# chroot /home/EMAIL/root /bin/ls -l
chroot: cannot change root directory to /home/EMAIL/root: Operation not permitted
```

6.1.11.3 Tripwire

To test tripwire, either run the cronjob by hand (`/etc/cron.daily/tripwire-check`) or simply wait until it runs automatically. (Overnight.) After it has run, there will be an email in root's mailbox with the results of the integrity check.

6.1.11.4 BIOS

To test the boot order and the password protection, place a bootable CD in the CD-ROM drive, and reboot. The machine should ignore the bootable CD and continue to boot from the hard drive.

To test that the BIOS is protected by a password, when rebooting try to enter the BIOS configuration by pressing the delete key.

6.2 Verification of Technical Requirements

The required services (DNS, HTTP, HTTPS, SMTP, POP) have been installed and configured. SMTPS and POPs (SMTP with SSL and POP with SSL) have also been installed and configured to provide a level of encryption to those services which will require the users of the system to send their authentication information across the network. LIDS has been installed to implement mandatory access controls, and every external service except OpenSSH and Apache are running in chrooted environments. The only software which was used for this that is not included in the Red Hat distribution is Snort and LIDS, but even Snort uses the Red Hat package management, keeping the future administration to a minimum.

6.3 Verification of High Level Requirements

The system is a low cost system. It is using common hardware, and open source / freeware. Due to using pre-packaged software in all but one instance (LIDS), and the script written to download updates, the maintenance requirements for performing upgrades has been kept at a minimum.

The necessary functionality is there. The server provides all of the fundamental necessary services for a company to have a presence on the Internet.

While no server can be 100% secure, it seems that all reasonable precautions have been taken to:

- a) limit the exposure to a vulnerability
- b) contain the damage from a vulnerability
- c) provide notification of any system modification
- d) keep private any user communication

6.4 Other Concerns

6.4.1 Backups

Truly a system should be backed up to prevent data loss in the event of a catastrophe. This particular system has two problems in this regard. First, it is remotely hosted at an ISP, and second there is no tape drive. It is recommended to either explore options with the hosting ISP to see if they can accommodate switching tapes and providing the

necessary physical security for those tapes or to either make an occasional visit to do the backup. (Even if only once a month or even less frequent.) Regarding the lack of tape drive, either installing a low cost consumer model, or if another machine is on the same network owned by the same company and it has a tape drive, backing up remotely through a secure protocol to that drive. (For instance, using dump and channeling it through SSH instead of rsh.)

6.4.2 Redundancy

Currently there is no fault tolerance for any of the services offered by this system. It would be recommended to run a DNS server on another host, or find a trusted business partner who is running a DNS server, and have them listed as and be a secondary server for the zones. It is also recommended at that point to create fallback MX records so that any mail attempted to be delivered during the time of a server failure would be queued on a secondary machine until the failure is recovered from.

© SANS Institute 2000 - 2002, Author retains full rights.

7 References

Andreasson, Oskar. "Iptables Tutorial 1.1.9" [Online]

Available at http://www.linuxsecurity.com/resource_files/firewalls/IPTables-Tutorial/iptables-tutorial/iptables-tutorial.html

Apache HTTP Server Documentation Project. "Apache HTTP Server Version 1.3 Documentation" [Online]

Available at <http://httpd.apache.org/docs/>

Bailey, Edward C. "Maximum RPM" [Online]

Available at <http://www.redhat.com/docs/books/max-rpm/max-rpm-html/>

Bremer, Steve. "LIDS FAQ" [Online]

Available at <http://www.lids.org/lids-faq/LIDS-FAQ.html>

Chuvakin, Anton Ph.D. "Using Chroot Securely" [Online]

Available at http://www.linuxsecurity.com/feature_stories/feature_story-99.html

Fielding, R; Gettys, J; Mogul, J; Frystyk, H.; Masinter, L; Leach, P; Berners-Lee T. "Hypertext Transfer Protocol -- HTTP/1.1" [Online]

Available at <ftp://ftp.isi.edu/in-notes/rfc2616.txt>

Huagang, Xie. "Build a Secure System with LIDS" [Online]

Available at http://www.linuxsecurity.com/feature_stories/feature_story-12.html

Meyers, J. & Rose M.; Post Office Protocol - Version 3 (RFC 1939)

<ftp://ftp.isi.edu/in-notes/rfc1939.txt>

Mitre.org. "Common Vulnerabilities and Exposures Data Base" [Online]

Available at <http://cve.mitre.org/>

Postel, Jonathan B.; "SIMPLE MAIL TRANSFER PROTOCOL" (RFC 821) [Online]

Available at <ftp://ftp.isi.edu/in-notes/rfc821.txt>

Ruii, Dragos. "Snort Users FAQ" [Online]

Available at <http://www.snort.org/docs/faq.html>

Red Hat. "Red Hat Linux 7.3 Customization Guide" [Online]

Available at <http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/custom-guide/>

Red Hat. "Red Hat Linux 7.3 Installation Guide" [Online]

Available at <http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/install-guide/>

Red Hat. "Red Hat Linux 7.3 Reference Guide" [Online]

Available at <http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/ref-guide/>

Welte, Harald. "Netfilter / IPTables FAQ" [Online]

Available at <http://netfilter.samba.org/documentation/FAQ/netfilter-faq.html>

© SANS Institute 2000 - 2002, Author retains full rights.

8 Files

8.1 Kickstart

```
#Kickstart file – Configuration file for an automatic install
#Change the rootpw and the bootloader m5pass

# Kickstart file automatically generated by anaconda. #and then modified

install
lang en_US
langsupport --default en_US.iso885915 en_US.iso885915
keyboard us
mouse genericps/2 --device psaux --emulthree
skipx
network --device eth0 --bootproto static --ip 192.168.17.220 --netmask 255.255.2
55.0 --gateway 192.168.17.1 --nameserver 127.0.0.1 --hostname newserver.markoram
.com
rootpw changeme
firewall --disabled
authconfig --enablesshadow --enablemd5
timezone America/New_York
bootloader --password=changeme
clearpart --linux
part /boot --fstype ext3 --onpart hda1
part /var --fstype ext3 --onpart hda7
part /usr --fstype ext3 --onpart hda6
part / --fstype ext3 --onpart hda5
part /tmp --fstype ext3 --onpart hda8
part /home --fstype ext3 --onpart hda10
part swap --onpart hda9

%packages
@ Network Support
@ Anonymous FTP Server
@ SQL Database Server
@ Web Server
@ Router / Firewall
@ DNS Name Server
@ Network Managed Workstation
@ Utilities
@ Software Development
@ Kernel Development
pspell-devel
```



```
unixODBC-devel
rsync
php-devel
bind-devel
php-pgsql
arpwatch
aspell-devel
apache-devel
shapecfg
isd4k-utils-devel

%post
```

© SANS Institute 2000 - 2002, Author retains full rights.

8.2 *makechrootsendmail.sh*

This is the script used to make the chrooted sendmail environment.

```
#!/bin/sh
#makechrootsendmail.sh - This program is used to create the chrooted
#                         sendmail environment
#       It makes the directories, and installs the packages.
#       CD1 of the Red Hat 7.3 Installation is required for this.
# Written by Mark Oram oram@markoram.com on 05/09/2002

if [ -e "/home/EMAIL" ]; then
    echo "Dir already exists!"
    exit 0
fi;

#Make the directory which will hold the chrooted environment
mkdir /home/EMAIL

#Set strict permissions
chmod 700 /home/EMAIL

#Make the top of the chrooted environemnt
mkdir /home/EMAIL/root

#Make the var/lib/rpm directory
mkdir -p /home/EMAIL/root/var/lib/rpm

#Make the dev directory
mkdir /home/EMAIL/root/dev/

#Make the dev/null device
#obtain the type/major/minor from doing a "ls -l /dev/null"
mknod /home/EMAIL/root/dev/null c 1 3

#Mount disc1 of the Installation CDs
mount /dev/cdrom /mnt/cdrom
echo "Please insert disc1 of the Red Hat 7.3 Installation and press enter"
read

#Change directory to the RPM directory of that CD
cd /mnt/cdrom//RPMS
#Start Installing the RPMS
#       First we install setup, basesystem and filesystem, these provide the
```

```

# essential /etc/ files and the directories
rpm -ivh setup-2.5.12-1.noarch.rpm --root /home/EMAIL/root
rpm -ivh filesystem-2.1.6-2.noarch.rpm --root /home/EMAIL/root
rpm -ivh basesystem-7.0-2.noarch.rpm --root /home/EMAIL/root

#Next we start installing the libraries
rpm -ivh glibc-common-2.2.5-34.i386.rpm glibc-2.2.5-34.i686.rpm \
--root /home/EMAIL/root

#Install the db3 libraries
rpm -ivh db3-3.3.11-6.i386.rpm --root /home/EMAIL/root

#Install mktemp (needed by PAM)
rpm -ivh mktemp-1.5-14.i386.rpm --root /home/EMAIL/root

#Install libtermcap (needed by bash)
rpm -ivh termcap-1.1.0.1-10.noarch.rpm libtermcap-2.0.8-28.i386.rpm \
--root /home/EMAIL/root

#Install bash (needed by PAM)
rpm -ivh bash-2.05a-13.i386.rpm --root /home/EMAIL/root

#Install words (needed by cracklib)
rpm -ivh words-2-18.noarch.rpm --root /home/EMAIL/root

#Install cracklib (needed by PAM)
rpm -ivh cracklib* --root /home/EMAIL/root
#Install info (needed by grep)
rpm -ivh info-4.1-1.i386.rpm --root /home/EMAIL/root --nodeps
#Install pcre (needed by grep)
rpm -ivh pcre-3.9-2.i386.rpm --root /home/EMAIL/root
#Install grep (needed by PAM)
rpm -ivh grep-2.5.1-1.i386.rpm --root /home/EMAIL/root
#Install fileutils
rpm -ivh fileutils-4.1-10.i386.rpm --root /home/EMAIL/root
#Install textutils
rpm -ivh textutils-2.0.21-1.i386.rpm --root /home/EMAIL/root
#Install sed
rpm -ivh sed-3.02-11.i386.rpm --root /home/EMAIL/root
#Install glib
rpm -ivh glib-1.2.10-5.i386.rpm --root /home/EMAIL/root
#Install initscripts
rpm -ivh initscripts-6.67-1.i386.rpm --root /home/EMAIL/root \
--nodeps --noscripts

#Install PAM
rpm -ivh pam-0.75-32.i386.rpm --root /home/EMAIL/root

#Install gdbm (needed for cyrus-sasl)

```

```
rpm -ivh gdbm-1.8.0-14.i386.rpm --root /home/EMAIL/root

#Install openssl
rpm -ivh openssl-0.9.6b-18.i686.rpm --root /home/EMAIL/root

#Install cyrus-sasl
rpm -ivh cyrus-sasl-* --root /home/EMAIL/root

#Install procmail
rpm -ivh procmail-3.22-5.i386.rpm --root /home/EMAIL/root

#Install ldap
rpm -ivh openldap-2.0.23-4.i386.rpm --root /home/EMAIL/root

#Install gawk
rpm -ivh gawk-3.1.0-4.i386.rpm --root /home/EMAIL/root

#Install shadowutils (needed for sendmail)
rpm -ivh shadow-utils-20000902-7.i386.rpm --root /home/EMAIL/root

#Install chkconfig (needed for sendmail)
rpm -ivh chkconfig-1.3.5-3.i386.rpm --root /home/EMAIL/root

#Install sh-utils
rpm -ivh sh-utils-2.0.11-14.i386.rpm --root /home/EMAIL/root

#Install sendmail
rpm -ivh sendmail-8.11.6-15.i386.rpm --root /home/EMAIL/root

#Unmount the cd-rom and eject it
cd /
umount /mnt/cdrom
eject
```

© SANS

8.3 *makechrootpop.sh*

This script is used to add the POP3 and POP3S services into the chrooted sendmail environment.

```
#!/bin/sh
# makechrootpop.sh - This is the program which adds the POP and
#                   POPS services to the chrooted sendmail tree
# Written by Mark Oram oram@markoram.com on 05/09/2002

#Mount disc1
echo "Insert Installation disc1 into the CD drive and press enter"
read
mount /dev/cdrom /mnt/cdrom
rpm -ivh /mnt/cdrom/RedHat/RPMS/krb5-libs-1.2.4-1.i386.rpm \
      --root /home/EMAIL/root
umount /mnt/cdrom
eject

echo "Please insert Installation disc2 into the CD drive and press enter"
read
mount /dev/cdrom /mnt/cdrom
rpm -ivh /mnt/cdrom/RedHat/RPMS/krb5-devel-1.2.4-1.i386.rpm \
      --root /home/EMAIL/root
rpm -ivh /mnt/cdrom/RedHat/RPMS/imap-2001a-10.i386.rpm \
      --root /home/EMAIL/root --nodeps
umount /mnt/cdrom
eject

echo "Please remove disc2"
cp /usr/share/ssl/certs/ipop3d.pem /home/EMAIL/root/usr/share/ssl/certs
echo "Finished"
```

© SANS

8.4 /etc/sysconf/iptables

This is the iptables configuration file which gets stored in /etc/sysconf/iptables.

```
# Generated by iptables-save v1.2.5 on Fri May 10 12:43:36 2002
*filter
#Set the default policies to DENY for all
:INPUT DROP [5:708]
:FORWARD DROP [0:0]
:OUTPUT DROP [0:0]
#Allow anything that is part of an already established connection
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
#Allow inbound POP3S
-A INPUT -p tcp -m tcp --dport 995 -j ACCEPT
#Allow inbound SMTPS
-A INPUT -p tcp -m tcp --dport 465 -j ACCEPT
#Allow inbound HTTPS
-A INPUT -p tcp -m tcp --dport 443 -j ACCEPT
#Allow inbound HTTP
-A INPUT -p tcp -m tcp --dport 80 -j ACCEPT
#Allow inbound DNS
-A INPUT -p tcp -m tcp --dport 53 -j ACCEPT
-A INPUT -p udp -m udp --dport 53 -j ACCEPT
#Allow inbound SMTP
-A INPUT -p tcp -m tcp --dport 25 -j ACCEPT
#Allow inbound SSH
-A INPUT -p tcp -m tcp --dport 22 -j ACCEPT
#Allow this machine to initiate any connections
-A OUTPUT -m state --state NEW,RELATED,ESTABLISHED -j ACCEPT
COMMIT
# Completed on Fri May 10 12:43:36 2002
```

8.5 /root/LIDS/createlids.sh

This is the file used to generate the LIDS configuration file (/etc/lids/lids.conf)

```
#!/bin/sh
#Set the location to the binary (lidsconf) which is used to manage the rules
LIDS=/sbin/lidsconf
#Flush the existing rules so there is a fresh start
$LIDS -Z

#Protect the lids binaries and files themselves
$LIDS -A -o /sbin/lidsconf           -j READONLY
$LIDS -A -o /sbin/lidsadm           -j READONLY
$LIDS -A -o /etc/lids               -j DENY

#Default most files systems and directories to read-only
$LIDS -A -o /sbin                   -j READONLY
$LIDS -A -o /bin                    -j READONLY
$LIDS -A -o /etc                    -j READONLY
$LIDS -A -o /boot                   -j READONLY
$LIDS -A -o /usr                    -j READONLY
$LIDS -A -o /var                    -j READONLY
$LIDS -A -o /home                   -j READONLY

#Allow some things to be written to
$LIDS -A -o /var/lock/subsys        -j WRITE
$LIDS -A -o /var/run                -j WRITE
$LIDS -A -o /etc/ioctl.save         -j WRITE
$LIDS -A -o /dev/null               -j WRITE

#Deny sensitive files
#shadow - has the encrypted passwords in it
$LIDS -A -o /etc/shadow              -j DENY
#grub.conf - has the encrypted grub password in it
$LIDS -A -o /boot/grub/grub.conf     -j DENY

#Allow logfiles in var/log
$LIDS -A -o /var/log                 -j APPEND
$LIDS -A -o /bin/login               -j READONLY
$LIDS -A -s /bin/login -o /var/log/wtmp -j WRITE
$LIDS -A -s /bin/login -o /var/log/lastlog -j WRITE
$LIDS -A -o /sbin/init               -j READONLY
$LIDS -A -s /sbin/init -o /var/log/wtmp -j WRITE
$LIDS -A -s /sbin/init -o /var/log/lastlog -j WRITE
$LIDS -A -o /sbin/halt               -j READONLY
```

```

$LIDS -A -s /sbin/halt -o /var/log/wtmp -j WRITE
$LIDS -A -s /sbin/halt -o /var/log/lastlog -j WRITE
$LIDS -A -o /etc/rc.d/rc.sysinit -j READONLY
$LIDS -A -s /etc/rc.d/rc.sysinit -o /var/log/wtmp -i 1 -j WRITE
$LIDS -A -s /etc/rc.d/rc.sysinit -o /var/log/lastlog -i 1 -j WRITE
$LIDS -A -s /etc/rc.d/rc.sysinit -o /var/log/dmesg -i 1 -j WRITE
$LIDS -A -s /etc/rc.d/rc.sysinit -o /var/log -i 1 -j WRITE
$LIDS -A -s /etc/rc.d/rc.sysinit -o CAP_SYS_ADMIN -i 3 -j GRANT

#Shutdown -- it needs to be able to kill off processes
$LIDS -A -s /sbin/init -o CAP_KILL_PROTECTED -j GRANT
$LIDS -A -s /sbin/init -o CAP_KILL -j GRANT
$LIDS -A -o /etc/rc.d/init.d/halt -j READONLY
$LIDS -A -s /etc/rc.d/init.d/halt -o CAP_INIT_KILL -i -1 -j GRANT
$LIDS -A -s /etc/rc.d/init.d/halt -o CAP_KILL -i -1 -j GRANT
$LIDS -A -s /etc/rc.d/init.d/halt -o CAP_NET_ADMIN -i -1 -j GRANT
$LIDS -A -s /etc/rc.d/init.d/halt -o CAP_SYS_ADMIN -i -1 -j GRANT
$LIDS -A -s /etc/rc.d/init.d/halt -o /var/lib/rpm -i 3 -j WRITE

#Update
$LIDS -A -o /sbin/update -j READONLY
$LIDS -A -s /sbin/update -o CAP_SYS_ADMIN -j GRANT

#Apache
$LIDS -A -o /usr/sbin/httpd -j READONLY
#It needs to setuid/setgid after starting
$LIDS -A -s /usr/sbin/httpd -o CAP_SETUID -j GRANT
$LIDS -A -s /usr/sbin/httpd -o CAP_SETGID -j GRANT
$LIDS -A -s /usr/sbin/httpd \
-o CAP_NET_BIND_SERVICE 80,443 -j GRANT

#Config files
$LIDS -A -o /etc/httpd -j DENY
$LIDS -A -s /usr/sbin/httpd -o /etc/httpd -j READONLY
#Log Files
$LIDS -A -s /usr/sbin/httpd -o /var/log/httpd -j WRITE

#OpenSSH
$LIDS -A -o /usr/sbin/sshd -j READONLY
$LIDS -A -s /usr/sbin/sshd -o /etc/shadow -j READONLY
$LIDS -A -o /etc/ssh/sshd_config -j DENY
$LIDS -A -o /etc/ssh/ssh_host_key -j DENY
$LIDS -A -o /etc/ssh/ssh_host_dsa_key -j DENY
$LIDS -A -s /usr/sbin/sshd -o /etc/ssh/sshd_config -j READONLY
$LIDS -A -s /usr/sbin/sshd -o /etc/ssh/ssh_host_dsa_key -j READONLY
$LIDS -A -s /usr/sbin/sshd -o /var/log/wtmp -j WRITE
$LIDS -A -s /usr/sbin/sshd -o /var/log/lastlog -j WRITE

```



```

$LIDS -A -s /usr/sbin/sshd -o CAP_SETUID -j GRANT
$LIDS -A -s /usr/sbin/sshd -o CAP_SETGID -j GRANT
$LIDS -A -s /usr/sbin/sshd -o CAP_FOWNER -j GRANT
$LIDS -A -s /usr/sbin/sshd -o CAP_CHOWN -j GRANT
$LIDS -A -s /usr/sbin/sshd -o CAP_DAC_OVERRIDE -j GRANT
$LIDS -A -s /usr/sbin/sshd -o CAP_NET_BIND_SERVICE 22-22 -j GRANT

#Bind
NAMED=/home/named/root
$LIDS -A -o /usr/sbin/named -j READONLY
$LIDS -A -o /root/named -j READONLY
$LIDS -A -s /usr/sbin/named \
-o CAP_NET_BIND_SERVICE 53-53 -j GRANT
$LIDS -A -s /usr/sbin/named -o CAP_SETPCAP -j GRANT
$LIDS -A -s /usr/sbin/named -o CAP_SYS_CHROOT -j GRANT
$LIDS -A -s /usr/sbin/named -o CAP_SYS_RESOURCE -j GRANT
$LIDS -A -s /usr/sbin/named -o CAP_SETUID -j GRANT
$LIDS -A -s /usr/sbin/named -o CAP_SETGID -j GRANT
$LIDS -A -s /usr/sbin/named -o CAP_DAC_READ_SEARCH -j GRANT
$LIDS -A -s /usr/sbin/named -o /var/run/named -j WRITE
$LIDS -A -s /usr/sbin/named -o $NAMED/var/run/named -j WRITE
$LIDS -A -s /usr/sbin/named -o /dev/null -j WRITE
$LIDS -A -o $NAMED/dev/null -j WRITE
$LIDS -A -s /usr/sbin/named -o $NAMED/dev/null -j WRITE

#Sendmail - both the chrooted and regular (the regular will be used locally
# by things like cron mail etc.)
#and POP - this will require xinetd (as does smtps) and chroot
EMAIL="/home/EMAIL/root"
$LIDS -A -o $EMAIL/dev/null -j WRITE
$LIDS -A -o $EMAIL/tmp -j WRITE
$LIDS -A -o $EMAIL/etc/shadow -j DENY
$LIDS -A -o /etc/mail -j DENY
$LIDS -A -o $EMAIL/etc/mail -j READONLY
$LIDS -A -o /usr/sbin/sendmail -j READONLY
$LIDS -A -o $EMAIL/usr/sbin/sendmail.sendmail -j READONLY
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
-o $EMAIL/etc/passwd -i -1 -j READONLY
$LIDS -A -s /usr/sbin/sendmail \
-o /etc/shadow -i -1 -j READONLY
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
-o $EMAIL/etc/shadow -i -1 -j READONLY
$LIDS -A -s /usr/sbin/sendmail -o /etc/mail -i -1 -j READONLY
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
-o $EMAIL/etc/mail -i -1 -j READONLY
$LIDS -A -s /usr/sbin/sendmail -o CAP_SETUID -i -1 -j GRANT

```

```

$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o CAP_SETUID -i -1                                -j GRANT
$LIDS -A -s /usr/sbin/sendmail -o CAP_SETGID -i -1        -j GRANT
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o CAP_SETGID -i -1                                -j GRANT
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o CAP_SYS_ADMIN -i -1                             -j GRANT
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o CAP_DAC_OVERRIDE -i -1                         -j GRANT
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o CAP_NET_BIND_SERVICE 25-25 -i -1              -j GRANT
$LIDS -A -s /etc/rc.d/init.d/sendmail -o $EMAIL/etc/mail -i 3 -j WRITE
$LIDS -A -s /etc/rc.d/init.d/sendmail \
        -o $EMAIL/etc/aliases -i 3                        -j WRITE
$LIDS -A -s /etc/rc.d/init.d/sendmail \
        -o $EMAIL/etc/aliases.db -i 3                     -j WRITE
$LIDS -A -s /usr/sbin/sendmail.sendmail \
        -o /var/spool/mail                                 -j WRITE
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o $EMAIL/var/spool/mail                          -j WRITE
$LIDS -A -s /usr/sbin/sendmail.sendmail \
        -o /var/spool/mqueue                              -j WRITE
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o $EMAIL/var/spool/mqueue                       -j WRITE
$LIDS -A -s /usr/sbin/sendmail.sendmail \
        -o /etc/mail/statistics                          -j WRITE
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o $EMAIL/etc/mail/statistics                    -j WRITE
$LIDS -A -o $EMAIL/var/run                                -j WRITE
$LIDS -A -s $EMAIL/usr/sbin/sendmail.sendmail \
        -o $EMAIL/var/run -i 5                            -j WRITE
$LIDS -A -o /usr/bin/procmail                             -j READONLY
$LIDS -A -o $EMAIL/usr/bin/procmail                      -j READONLY
$LIDS -A -s /usr/bin/procmail \
        -o /var/spool/mail                                 -j WRITE
$LIDS -A -s $EMAIL/usr/bin/procmail \
        -o $EMAIL/var/spool/mail                          -j WRITE
$LIDS -A -s /etc/rc.d/init.d/sendmail \
        -o CAP_SYS_CHROOT -i 2                            -j GRANT
$LIDS -A -o $EMAIL/usr/sbin/ipop3d                       -j READONLY
$LIDS -A -s $EMAIL/usr/sbin/ipop3d \
        -o $EMAIL/etc/passwd                             -j READONLY
$LIDS -A -s $EMAIL/usr/sbin/ipop3d \
        -o $EMAIL/etc/shadow                             -j READONLY
$LIDS -A -s $EMAIL/usr/sbin/ipop3d -o CAP_SETUID         -j GRANT
$LIDS -A -s $EMAIL/usr/sbin/ipop3d -o CAP_SETGID        -j GRANT

```

```

$LIDS -A -s $EMAIL/usr/sbin/ipop3d \
        -o $EMAIL/var/spool/mail -j WRITE
$LIDS -A -s /usr/sbin/xinetd \
        -o CAP_NET_BIND_SERVICE 110-110 -j GRANT
$LIDS -A -s /usr/sbin/xinetd \
        -o CAP_NET_BIND_SERVICE 465-465 -j GRANT
$LIDS -A -s /usr/sbin/xinetd \
        -o CAP_NET_BIND_SERVICE 995-995 -j GRANT
$LIDS -A -s /usr/sbin/xinetd \
        -o CAP_SYS_CHROOT -i 1 -j GRANT

#Snort
$LIDS -A -o /usr/sbin/snort -j READONLY
$LIDS -A -s /usr/sbin/snort -o CAP_DAC_OVERRIDE -j GRANT
$LIDS -A -s /usr/sbin/snort -o CAP_NET_RAW -j GRANT
$LIDS -A -s /usr/sbin/snort -o CAP_HIDDEN -j GRANT
$LIDS -A -s /usr/sbin/snort -o CAP_SETUID -j GRANT
$LIDS -A -s /usr/sbin/snort -o CAP_SETGID -j GRANT
$LIDS -A -s /usr/sbin/snort -o CAP_CHOWN -j GRANT
$LIDS -A -s /usr/sbin/snort -o CAP_SYS_CHROOT -j GRANT
$LIDS -A -s /usr/sbin/snort -o /home/snort/var/log/snort -j WRITE

#Login
$LIDS -A -s /bin/login -o /etc/shadow -j READONLY
$LIDS -A -s /bin/login -o CAP_SETUID -j GRANT
$LIDS -A -s /bin/login -o CAP_SETGID -j GRANT
$LIDS -A -s /bin/login -o CAP_CHOWN -j GRANT
$LIDS -A -s /bin/login -o CAP_FSETID -j GRANT

#Misc files run at boot time
$LIDS -A -o /sbin/inssmod -j READONLY

#Hide this dir
$LIDS -A -o /root/LIDS -j DENY

#Syslog
$LIDS -A -o /sbin/syslogd -j READONLY
$LIDS -A -s /sbin/syslogd -o /var/log -j WRITE
$LIDS -A -s /sbin/syslogd -o $EMAIL/dev/ -j WRITE
$LIDS -A -s /sbin/syslogd -o $NAMED/dev/ -j WRITE
$LIDS -A -s /sbin/klogd -o CAP_SYS_ADMIN -j GRANT

$LIDS -A -s /etc/rc.d/rc.sysinit -o /etc -j WRITE

$LIDS -A -o /usr/sbin/logrotate -j READONLY
$LIDS -A -s /usr/sbin/logrotate -o /var/log -j WRITE

```

```

#Allow /etc/rc.d/init.d/random to do whatever it wants to the
#/var/lib/random-seed file
$LIDS -A -o /etc/rc.d/init.d/random -j READONLY
#Depth of two, so the shell and dd can inherit
$LIDS -A -s /etc/rc.d/init.d/random \
-o /var/lib/random-seed -i 2 -j WRITE

#/etc/rc.d/init.d/network for shuttingdown
$LIDS -A -o /etc/rc.d/init.d/network -j READONLY
$LIDS -A -s /etc/rc.d/init.d/network -o CAP_NET_ADMIN -i 3 -j GRANT

#Cron stuff
$LIDS -A -s /usr/lib/sa/sadc -o /var/log/sa -j WRITE
$LIDS -A -s /usr/lib/sa/sa2 -o /var/log/sa -i 3 -j WRITE
#Webalizer
$LIDS -A -s /etc/cron.daily/00webalizer \
-o /var/lib/webalizer -i 2 -j WRITE
#Anacron
$LIDS -A -s /etc/cron.daily/0anacron \
-o /var/spool/anacron/cron.daily -i 1 -j WRITE
$LIDS -A -s /etc/cron.weekly/0anacron \
-o /var/spool/anacron/cron.weekly -i 1 -j WRITE
$LIDS -A -s /etc/cron.monthly/0anacron \
-o /var/spool/anacron/cron.monthly -i 1 -j WRITE
#logrotate
$LIDS -A -s /etc/cron.daily/logrotate \
-o /var/lib/logrotate.status -i 1 -j WRITE
#4?
$LIDS -A -s /etc/cron.daily/logrotate \
-o /var/log -i 4 -j WRITE
#makewhatis
$LIDS -A -s /etc/cron.daily/makewhatis.cron \
-o /var/cache/man/whatis -i 2 -j WRITE
$LIDS -A -s /etc/cron.weekly/makewhatis.cron \
-o /var/cache/man/whatis -i 2 -j WRITE
$LIDS -A -s /etc/cron.daily/makewhatis.cron \
-o /var/lock -i 1 -j WRITE
$LIDS -A -s /etc/cron.weekly/makewhatis.cron \
-o /var/lock -i 1 -j WRITE
#rpm
$LIDS -A -s /etc/cron.daily/rpm \
-o /var/log/rpmpkgs -j WRITE
$LIDS -A -s /etc/cron.daily/rpm \
-o /var/lib/rpm -i 2 -j WRITE
#slocate

```

```

$LIDS -A -s /etc/cron.daily/slocate.cron \
      -o /var/lib/slocate -i 1 -j WRITE

#Tripwire
#protect it from prying eyes
$LIDS -A -o /etc/tripwire -j DENY
$LIDS -A -o /var/lib/tripwire/ -j DENY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /etc/tripwire -i 1 -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /var/lib/tripwire -i 1 -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /var/lib/tripwire/report -i 1 -j WRITE
$LIDS -A -s /etc/cron.daily/tripwire-check -o /etc/lids -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check -o /root/LIDS -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check -o /etc/shadow -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /boot/grub/grub.conf -i 1 -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /etc/httpd/conf -i 1 -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /etc/ssh/sshd_config -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /etc/ssh/ssh_host_key -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o /etc/ssh/ssh_host_dsa_key -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o $EMAIL/etc/shadow -j READONLY
$LIDS -A -s /etc/cron.daily/tripwire-check \
      -o $EMAIL/etc/mail -j READONLY

/sbin/lidsadm -S -- +RELOAD_CONF

```

© SANS

8.6 *checkupdates.pl*

This is a quick perl script written by the author to retrieve updates to the system from Red Hat's FTP site.

```
#!/usr/bin/perl
# $Id: checkupdates.pl,v 1.5 2002/05/19 04:08:12 oram Exp oram $

use strict;
use warnings;
use Net::FTP;
use Getopt::Std;

our %opts = ();
getopts('hda:l:p:r:R:B:', \%opts) || usage();

usage() if ($opts{h});
our $debug = (defined($opts{d})) ? 1 : 0;
our $version = defined($opts{r}) ? $opts{r} : '7.3';
# Arch, in order of preference
our $archs = defined($opts{a}) ? $opts{a} : 'i686,i586,i486,i386';
our $password = defined($opts{p}) ? $opts{p} : 'user@somewhere';
our $language = defined($opts{l}) ? $opts{l} : 'en';
our $roots = defined($opts{R}) ? $opts{R} : '/,/home/EMAIL/root';
our $names = defined($opts{N}) ? $opts{N} : 'root,EMAIL';
our @Rnames = split('\,', $names);
our $basedir = defined($opts{B}) ? $opts{B} : '/root/updates';

our $FTPbasedir = "$version/$language/os";

# Datastructure to hold the updates in, it will be:
# updates->{arch}->{name}->{version-release}->{arch}
# I am assuming that there will not be different versions for different
# architectures... (there should not be!)
our $updates = { };
foreach my $starch (split(',', $archs))
{
    $updates->{$starch} = { };
}

# Establish FTP
my $ftp = Net::FTP->new( "updates.redhat.com", Passive => 1, Debug => $debug ) || die
"asdf\n";
$ftp->login( "anonymous", $password ) || die "FTP: Could not login\n";
```

```

#For each arch
foreach my $thisarch (split(',', $archs))
{
    print "Doing $thisarch\n" if ($debug);
    $ftp->cwd( "$FTPbasedir/$thisarch" ) || die "CWD ($FTPbasedir/$thisarch)
failed\n";
    if (my @files = $ftp->ls())
    {
        foreach my $thisfile ( @files )
        {
            #For each file, we need to parse its version
            #and release and store it into memory
            if ( $thisfile =~ /^
                (.+)          #Name
                \-
                (.+)          #Version
                \-
                (.+)          #Release
                \.
                (i386|i486|i686|athlon|noarch) #Arch
                \.rpm$/ix )
            {
                my $name = $1;
                my $version = $2;
                my $release = $3;
                my $arch = $4;
                my $vr = $version . '-' . $release;
                #print "$name|$version|$release|$arch\n";
                if ( (!defined($updates->{$arch}->{$name})) ||
                    (newer($updates->{$arch}->{$name}, $vr))
                )
                {
                    #print "\tWould update!\n";
                    $updates->{$arch}->{$name} = $vr;
                }
                else
                {
                    $updates->{$arch}->{$name} = $vr;
                }
            }
        }
    }
    else
    {
        warn "Could not parse: $thisfile\n" if ($debug);
    }
}
}

```

```

    #ftp->cdup() || die "CDUP failed\n";
}
ftp->cdup();

#Now we get all of the rpms installed (in each location)
#and for each one, check to see if we have an updated version, if so
#retrieve and keep track of the fact that we retrieved it.
#I know this could be done through perl, but this is easier for now.
foreach my $thisroot (split('\,', $roots))
{
    my $curdir = shift(@Rnames);
    mkdir("$basedir/$curdir") if (! -d "$basedir/$curdir");
    chdir("$basedir/$curdir") || die "Could not chdir ($basedir/$curdir): $!\n";
    open(RPM, "/bin/rpm -qa --root $thisroot |") || die "Could not run RPM: $!\n";
    while(<RPM>)
    {
        chomp(my $line = $_);
        if ($line =~ /^
                (.+)          #Name
                \-
                (.+)          #Version
                \-
                (.+)          #Release
                $/ix)
        {
            my $name = $1;
            my $version = $2;
            my $release = $3;
            my $vr = $version . '-' . $release;
            foreach my $starch (split('\,', $sarchs))
            {
                if ( defined($updates->{$starch}) )
                {
                    if (defined(my $x = $updates->{$starch}-
>{$name}))
                    {
                        #print "$x vs. $vr\n";
                        if (newer($x,$vr))
                        {
                            ftp-
>get("$FTPbasedir/$starch/$name-$x.$starch.rpm");
                            last;
                        }
                    }
                }
            }
        }
    }
}

```



```

    }
    else
    {
        warn "Could not parse: $_";
    }
}
close(RPM);
}

#Given two version.release strings, return true if the first is newer than
#the second (uses xcomp and ycomp)
sub newer
{
    my $vr1 = shift;
    my $vr2 = shift;

    my ( $ver1, $rel1 ) = ( $1, $2 ) if ( $vr1 =~ /^(.+)\-(.+)$/ );
    my ( $ver2, $rel2 ) = ( $1, $2 ) if ( $vr2 =~ /^(.+)\-(.+)$/ );

    if ( $ver1 eq $ver2 )
    {
        #If the versions are equal, then the tie breaker is the
        #releases.
        return( xcomp( $rel1, $rel2 ) );
    }
    else
    {
        #If the versions are not equal, compare them
        return( xcomp( $ver1, $ver2 ) );
    }
}

#Given two strings, compare them and return true if a is greater than b
sub xcomp
{
    my $a = shift;
    my $b = shift;

    my @aparts = split('\.', $a);
    my @bparts = split('\.', $b);

    for (my $n = 0; $n <= $#aparts; $n++)
    {
        return 1 if (!defined($bparts[$n]) ||
                    ycomp($aparts[$n], $bparts[$n]));
    }
}

```

```

    }
    #If we are still here, they are equal OR bparts has more
    return 0;
}

sub ycomp
{
    my $a = shift;
    my $b = shift;

    while( $a =~ /([0-9]+|[A-Z]+)/i )
    {
        my $suba = $1;
        $a = $';

        if ( $b =~ /([0-9]+|[A-Z]+)/i )
        {
            my $subb = $1;
            $b = $';
            if ( $suba =~ /[a-z]/i || $subb =~ /[a-z]/i )
            {
                return(1) if ( ($suba cmp $subb) == 1);
            }
            else
            {
                return(1) if ($suba > $subb);
            }
        }
    }

    return(0);
}

sub usage
{
    my $version = '$Revision: 1.5 $';

    print "\n";
    print "checkupdates.pl will connect to updates.redhat.com via FTP\n";
    print "and compare the updates available to the RPMs installed on\n";
    print "the machine. It will then download any updates to already\n";
    print "installed packages. It supports multiple 'roots' (for instance\n";
    print "if you have a chrooted environment for mail, etc)\n\n";

    print "checkupdates.pl $version\n\n";
}

```

```
print "checkupdates.pl [ options ]\n";
print "\t\t-d\t\t: Debug\n";
print "\t\t-r\t\t: Version (default: 7.2)\n";
print "\t\t-a\t\t: Archs (in order of pref)\n\t\t\t\t\t(Default: i686,i586,i486,i486)\n";
print "\t\t-p\t\t: FTP password (default: someuser\@somehost)\n";
print "\t\t-l\t\t: Language (default: en)\n";
print "\t\t-R\t\t: Roots (default: ./home/EMAIL/root)\n";
print "\t\t-N\t\t: Names (default: root,EMAIL)\n";
print "\t\t-B\t\t: Basedir (default: /root/updates)\n";
print "\n";
exit(0);
}
```

© SANS Institute 2000 - 2002, Author retains full rights.

Upcoming Training

Click Here to
{Get CERTIFIED!}



SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced