Global Information Assurance Certification Paper

Practical Assignment

GIAC Certified Unix Security Administrator
Securing Unix Step-by-Step

Hardening a Red Hat Linux Apache Web Server with Snort Installed

Written by Jacqui Chau
19 December 2003
Version 1

1. Abstract

The purpose of this paper is to outline the procedure of hardening an Apache Linux Web server with Snort installed for intrusion detection.

The paper first looks into the recommended hardware, software and architecture for a home-based / small-office web server and then the step-by-step guide will detail how to protect the web server from outside intruders. Finally, the recommended ongoing maintenance required to ensure this web server is continuously secure is outlined.

Contents

2. System Description

2.1 Description of System

This guide is designed for a home-based / small-business web site which requires a very high level of protection against intruders and high-jacking. For example, a small business that relies on accurate timetables, prices and course information to be displayed at all times. This type of business could not afford for someone 'taking over' the system and reproducing pages that displays inaccurate and possibly harmful information.

Another example is a web site that contains personal information/photos/details that need to remain secure. In this type of web site the owner would not want unauthorised persons deleting/modifying this data. Therefore, a high level of security as to who can perform particular tasks is vital.

This guide is designed to hopefully assist these small businesses to create a low-cost, secure web server that will allow them to detect whether any intruders attempt to modify or destroy any data on the server.

The operating system that I have used in this guide, is the open source Red Hat Linux. This operating system is easy to download, easy to install and easy to use.

Once installed, the operating system will be hardened to protect against major vulnerabilities such as:

- Denial Of Service Attacks
- Modification of system critical files (eg. Passwords, network configuration)
- Unauthorised access

OpenSSH is required to securely transfer files between the web server and a 2nd laptop will be used for personal uses, such as checking email.

Tripwire will also be installed, to ensure that none of the web-page files and vital system files are modified without being logged and a user notified.

Snort IDS will be installed to detect whether any intruders are trying to penetrate the system.

2.2 Hardware

The hardware used is a laptop, which allows the web-site to be portable.

I have used the following hardware:

- **Laptop Model**: Dell Lattitude CPx
- **CPU**: Pentium III 500 Mhz. X86
- **RAM**: 128 SDRAM
- **Hard Disk Space**: 6.4 Gig intergated IDE
- **CDROM**: EIDE 32X MAX
- **Network**: PCMCIA

2.3 Software

It is important that every software package you download, has its signature checked to ensure the integrity of the file. You can never rely on web sites and ftp servers to maintain non-corrupt or virus-free install files.

In each section of the guide, it is outlined how/where to download the software, and how to check that the software package signatures are valid.

- **Operating System**: Red Hat Linux 9
- **Web Server**: Apache 2.0 +
- **IDS**: Snort Sensor 2.0 +
- **Integrity Assurance**: Tripwire 2.3 +
- **SSH**: OpenSSH 3.7 +

Testing security Software:

- **Hardening script**: Bastille
- **Security Scanner:** Nessus
- **Port Scanner**: Nmap

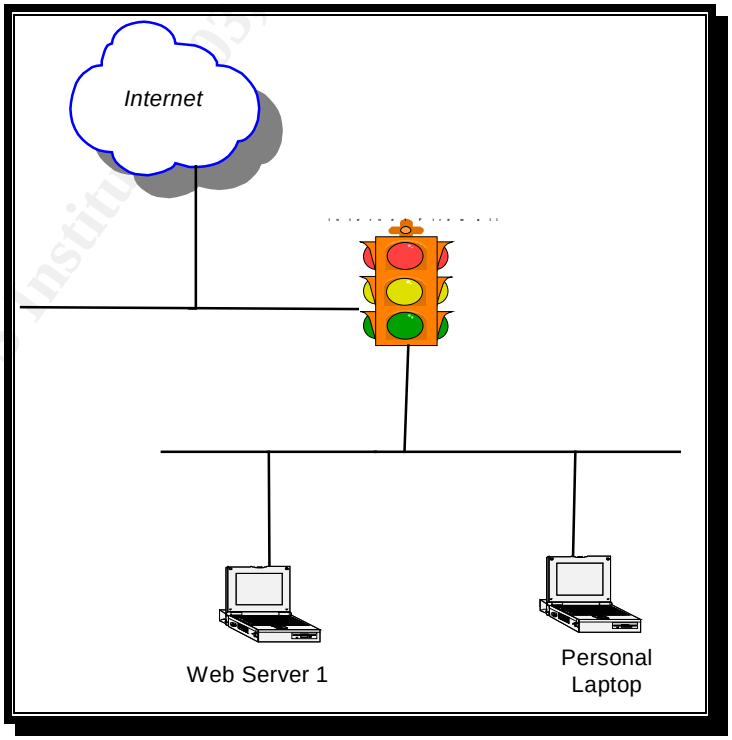
2.4 Topology

The network diagram below shows a typical architecture for web servers.
The web servers are protected behind an Internet firewall that will filter most of the traffic from the outside networks.

Alternatively, the web server may have a personal firewall package installed to reduce the number of devices required in the system. However, it is recommended that if an organisation is concerned about security, then a separate device should used and configured for the firewall.

The second personal laptop will be used to collect email and logs from the Web server. It is good security practice to send logs to an alternative server, for forensics in the event of a successful hack. It is also ideal to keep the web server dedicated to web-server tasks, and perform personal tasks, such as email and Internet surfing to another laptop. The web server will be security hardened and unnecessary services/processes stopped, making it impossible for non web-server tasks to be run.

It is always advised that critical components are load/balanced and made highly available to reduce downtime in the event of failure. However, this is optional and depends on how vital the uptime of the site is.



source: Jacqui Chau

2.5 Analysis of System

The most typical attacks to Web servers that are security key concerns are:

- Denial of Service
- Interception and manipulation of messages
- Forged client requests
- Forged server responses
- Attempts to read the server file system/database
- Attempts to write to the server file system/database

If the web server has a significant amount of coding which calls programs, then there is a risk that this functionality may be offered to the wrong people or that it opens a security hole in the system.

If an intruder was to attempt a denial of service attack, then they could achieve this by flooding the server with email, or pushing the server to work harder by executing resource-heavy programs. A system therefore needs to be configured to not allow processes/programs to be executed that are not part of its everyday operation. For a web server, you only require particular program and services running. Therefore only those programs and services should be installed.

If an unauthorised person was able to successfully get onto the system, you want to make it as difficult as possible for them to do anything, or at least to slow them down. Restricting access to particular files, logging all actions, not allowing root login so they need to know at least 2 passwords, removing permissions to insecure applications such as the 3 ‘r’s will all assist in making the hackers life more difficult.

Someone may want to intercept and manipulate information that is communicated between your server and another remote host. Therefore it is important to use encrypted channel software such as SSH to send data to make it more difficult to sniff. Also, restricting the hosts that are permitted to connect to the server on certain ports is essential to combating this type of attack.

In the event of someone manipulating files, it is important for a file integrity program such as tripwire to be installed to notify you of what has been changed. This will make the restoration of your web server much easier and assist in prosecuting the offender if necessary.

2.6 Secure the Environment

2.6.1 Physical Security

The laptops should be stored in a secure environment to prevent unauthorised people from logging on and compromising the system.
The firewall, IDS and tripwire detection are almost useless if someone has physical access to the system.
You need all levels of security to protect these web servers, but physical security is the one of most critical.

Therefore it is imperative that the following is installed as a basic:

- Laptops locked with a security padlock to a secure object
- Laptops are locked in a secure room with key in which only limited people have access
- Laptops are password protected; so unauthorised people cannot simply start hacking the system from the console.

2.6.2 Firewall Security

A firewall should be installed and configured to block any unnecessary ports and services from being used.

Leaving particular ports and services open, can open security holes and hence making your system incredibly vulnerable.

A separate device should be used to filter this traffic from the Internet into the internal web server farm.

In this guide, I have outlined how to configure Red Hat Linux's firewall function, IPTABLES.

2.6.3 Set up NTP

Setting up the server to obtain the time from a Network Time Protocol (NTP) server is highly recommended and is good security practice. It ensures that logs have the correct timestamp, and makes it more difficult for intruders attempting to cover their tracks.

3. Step-by-Step Guide: Installing Red Hat Linux 9

3.1 Preparation

Download:	Site
Installation CD's (x3)	ftp://ftp.redhat.com/pub/redhat/linux/9/en/iso/i386/shrike-i386-disc1.iso ftp://ftp.redhat.com/pub/redhat/linux/9/en/iso/i386/shrike-i386-disc2.iso ftp://ftp.redhat.com/pub/redhat/linux/9/en/iso/i386/shrike-i386-disc3.iso
MD5Sums	http://ftp.redhat.com/pub/redhat/linux/9/en/iso/i386/MD5SUM
security@redhat.com fingerprint	www.redhat.com/solutions/security/news/publickey.html CA 20 86 86 2B D6 9D FC 65 F6 EC C4 21 91 80 CD DB 42 A6 0E
MD5 Hash	http://www.redhat.com/security/db42a60e.txt
Public Key	http://pgp.mit.edu:11371/pks/lookup?search=0xdb42a60e&op=index

3.1.1 Integrity Test

It is important to ensure that the software you are downloaded comes from a trusted source. Otherwise you may be compromising your system by installing an already insecure operating system.

Verify that the software was uploaded by Red Hat
Open the MD5Sum file and check the hashes for the 3 files.

```
400c7fb292c73b793fb722532abd09ad          shrike-i386-disc1.iso
6b8ba42f56b397d536826c78c9679c0a          shrike-i386-disc2.iso
af38ac4316ba20df2dec5f990913396d          shrike-i386-disc3.iso
```

1. Verify the md5 hash for each file at the command prompt by comparing the results with the MD5Sum file

```
# md5sum shrike-i386-disk1.iso
# md5sum shrike-i386-disk2.iso
# md5sum shrike-i386-disk3.iso
```

2. The Red Hat pgp public key has signed the MD5SUM file. Verify the contents of the file, by going to the directory in which you have saved the MD5 hash and executing the following command:

```
# wget http://www.redhat.com/security/db42a60e.txt
```

```
--12:50:14—http://www.redhat.com/security/db42a60e.txt
> 'db42a60e.txt'
Resolving www.redhat.com…done
Connecting to www.redhat.com[66.187.232.50]:80…connected.
HTTP request sent, awaiting response... 200 ok
Length: 1,838 [text/plain]
100%[==============================>]1,838       .75M/s     ETA 00:00

12:50:17 (1.75 MB/s) – 'db42a60e.txt' saved [1838/1838]
```

3. Import the key into the gpg database

```
# gpg –import db42a60e.txt
```

```
gpg: key DB42A60E: public key "Red Hat, Inc <security@redhat.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1
```

3. Verify the file

```
# gpg –verify MD5Sum
```

```
gpg: Signature made Thu 11 Sep 2003 using DSA key ID DB42A60E
gpg: Good signature from "Red Hat, Inc <security@redhat.com>"
gpg: checking the trustdb
```

3.1.2 Software preparation

Once the files have been verified, use a cd-burner to copy each of the files onto separate CDs.

3.2 Install

The installation will give you several items that you will need to configure. The table on the next page outlines the answers you should enter, and a short explanation as to why you should choose that option.

Reducing the number of vulnerabilities exposed to this web server upon first installation was considered while choosing these options

The first step, is to insert the Red hat installation CD 1, and restart the system.

The system should boot to a welcome screen.

Answer the prompts as outlined below

Question / Option	Explanation	Action
Option to install or upgrade red hat linux in graphical or text mode	Choose how you would like to view the installation screens. Graphical or text based. Graphical is much easier and intuitive to use.	Press the <enter> key
Welcome to Red Hat Linux. To begin testing the cd media before installation press ok	If the cd has been newly created, then it is recommended to perform a media test.	Select <skip>
RedHat9 GUI	Explanation and welcome	Next
Language Selection	Select your language	English
Keyboard	Select your keyboard type	United Kingdom
Mouse Configuration	Select type of mouse you will be using with the system	2 button Mouse (PS/2)
Installation Type	**Server** (allows file sharing, print sharing and web services) **Custom** (more configurable)	Select <Custom>
Disk Partitioning Setup	**Automatic Partitioning:** Selects defaults of: /boot / swap **Manual Partitioning (**Disk Druid**):** Allows you to configure how many partitions you require and the size of each partition	**Manual Partitioning** For details, see **Appendix A: Disk Partitioning**
Boot Loader Configuration	Grand Unified Boot Loader (GNU GRUB) is the default boot loader. It can load multiple operating systems. The boot loader is required in order to boot a system without a boot diskette. It is the first software program that runs when a computer starts and is responsible for loading and transferring control to the operating system kernel software. [1] A boot loader password prevents users from changing options passed to the kernel. For greater system security, it is	Leave default selection of GRUB boot loader /dev/hda Select 'Use a boot loader password'

[1] http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/install-guide/s1-x86-bootloader.html

	recommended that you set a password	
Network Configuration	The system will obtain an ip address from whatever network it is connected to (if you select dhcp)	Enter hostname Ip address (if known) DNS (if known)
Firewall Configuration	**High security:** Should use this option if you are connecting your system to the Internet, but do not plan to run a server. Add trusted devices or to allow additional incoming interfaces www: This protocol is used by Apache to serve web pages	Select a security level for the system: < **High** > *Select* < *Customize* > Select Trusted devices: < eth0 > Allow incoming: WWW (http) DHCP
Additional Language Support	Select your language	English (Great Britain)
Time Zone Selection	Select your timezone	Location: Europe/London
Set Root Password	Ensure that you select a complex password. Use at least 8 characters with mixed uppercase/lowercase, symbols and numbers.	Enter the root (administrator) password for the system
Authentication Configuration		Leave default
Package Group Selection	Select the minimum packages possible Then select '**Select individual packages'** at the bottom of the screen.	Misc: Minimal
Individual Package Selection	It is best to select each individual package, as it ensures you know exactly what is installed onto your system.	Select the following packages: [2] • APACHE • APACHE_DEVEL • APACHE_MANUAL • APMD • CRONTABS • DEVLABEL

[2] http://www.giac.org/practical/GCUX/Rick_Larabee_GCUX.pdf
http://www.redhat.com/docs/manuals/linux/RHL-7.1-Manual/ref-guide/s1-installation-optionalpackages.html

		• DHCLIENT • DIFFUTILS • GNUPG • HESIOD • IPTABLES • LIBCAP • LIBPCAP • LOGROTATE • LOGWATCH • LSOF • M4 • MAILCAP • MAILX • MOD_SSL • NTP • OPENSSH • OPENSSH-CLIENTS • OPENSSH-SERVER • OPENSSH-ASKPASS • OPENSSL • PERL • PERL-FILTER • PERL-TIME-HiRes • PROCMAIL • QUOTA • SENDMAIL • SENDMAIL-CF • SLOCATE • TCPWRAPPERS • TMPWATCH • TRIPWIRE • UNZIP • UTEMPTER • VIXIE-CRON • ZIP
Insert disks 2 and 3 when prompted	Cd's 2 and 3 are required to complete installation	Insert disks 2 and 3 when prompted
Boot Diskette Creation	It is recommended that you create a boot disk. You will require this boot disk in the event of your operating system having	Yes, I do want to create a boot diskette

	problems booting.	
Monitor Configuration	Select your monitor type	Unprobed Monitor Customize Graphical configuration Colour Depth: True Colour (24 bit) Screen Resolution: 800x600
INSTALLATION COMPLETE	INSTALLATION COMPLETE	CD should eject, and system should reboot

3.3 Configuration

3.3.1 Create Groups and users

The root user should not be the only person to have access to the system.

All users should login with their administrator account, and then su to root if necessary.
This is particularly important for auditing purposes. Knowing who performed what and at what time is essential for forensics.

```
# useradd admin
# useradd apache
# useradd snort

# groupadd sysadmin
# groupadd apache
# groupadd snort
```

3.3.2 Configuring Sendmail

Sendmail is required in order to notify the system administrator when something in the system has changed through Tripwire or that someone is trying to access the web server via an illegal port through snort.

Attackers who use sendmail to flood a system with mail can achieve a denial of service. Therefore it is important that these types of attacks are reduced by setting limits in the /etc/mail/sendmail.mc file.

Sendmail should have been installed during the initial setup, and to prevent any illegal users from accessing it, it should be bound to a loopback address.

The main configuration file is the /etc/sendmail.cf. Editing this file directly is not recommended. Instead, the /etc/mail/sendmail.mc file should be edited and the m4 macro processor used, to create the new /etc/sendmail.cf
The steps are outlined below:

1. Backup /etc/mail.sendmail.mc
2. Edit sendmail.mc
3. Run the m4 command to save configuration

```
# /etc/mail/sendmail.mc > /etc/sendmail.cf
```

4. Restart sendmail

```
# /sbin/service/sendmail restart
```

3.3.3 Setup disclaimer banner

It is a good idea to display a 'Authorized use only' message at log-in time, to assists with prosecuting system crackers and warning them about potential prosecution if they login and are unauthorised to do so.
Steps to setting up the desclaimer banner:

1. Clarify the message wording with management before installing.
2. Create a file in /etc called 'issue'
3. Save file.

The banner should appear during bootup of system, just before the user logs in.

4. Step-by-Step Guide: Hardening Red Hat Linux

4.1 Preparation

Download:	Site
Security Patches and Bug Fixes	http://www.redhat.com/apps/support/errata/rh9-errata.html ftp://updates.redhat.com/9/en/os/i386/
Public Keys	www.redhat/com/solutions/security/news/publickey/#key http://pgp.mit.edu:11371/pks/lookup?search=0xdb42a60e&op=index

4.2 Setup Red Hat Update Agent

The Red Hat Update Agent, assists in updating the system with the latest software available from Red Hat.
You need to ensure that a Domain Name Server (DNS) is accessible.

4.3 Manual installation of patches

At the time of writing this paper, these were the patches that were downloaded and installed:

Name of patch	Description
Coreutils-4.5.3.19.0.2.i386.rpm	Close a potential denial of service vulnerability
Openssh-3.5p1-11.i386.rpm Openssh-askpass-3.5p1-11.i386.rpm Openssh-askpass-gnome-3.5p1-11.i386.rpm Openssh-clients-3.5p1-11.i386.rpm Openssh-server-3.5p1-11.i386.rpm	Updated OpenSSH Packages, to fix potential vulnerabilities
Httpd-2.0.40.21.5.i386.rpm Httpd-devel-2.0.40-21.5.i386.rpm	Updated httpd packages fix Apache security vulnerabilities

Httpd-manual-2.0.40-21.5.i386.rpm Mod_ssl-2.0.40-21.5.i386 Rhpl-0.93.4-1.i386.rpm	Updated redhat-config-network package available

4.4 Install Security Patches

1. Download patches
2. Packages are GPG signed by Red Hat for security. Verify Signature for each rpm before installing

```
# rpm –checksig –v filename
```

(eg `# rpm –checksig –v coreutils-4.5.3.19.0.2.i386.rpm`)

```
Coreutils-4.5.3.19.0.2.i386.rpm:
    Header V3 DSA signature: OK, key ID db42a60e
    Header SH1 digest: OK (7181f456513c7112504e67d8ba84960f30d91372)
        MD5 digest: OK (36f1e9cf924c83953e6a9a811f1e2d4a)
        V3 DSA signature: OK, key ID db42a60e
```

3. Verify signature by looking up key at public key server:
http://pgp.mit.edu:11371/pks/lookup?search=0xdb42a60e&op=index

Key id is **db42a60e**

4. Execute the following command for each rpm

```
# rpm –Fvh [filename].rpm
```

4.5 Remove unnecessary scripts and services

The default installation of Linux, installs many useful services and scripts. However, unnecessary scripts and services should be removed, as they could introduce potential security threats.

Exploits to services are discovered daily, so if you don't need the service or script, then remove it.

Network services are potentially insecure, as many network file systems such as NFS and SMB that passes information over the network unencrypted.

Passing clear-text sensitive information over a network that is not encrypted could reveal this information to unwanted persons.

4.5.1 Remove unnecessary services

1. Login as root
2. The following command will give you the number of services that are currently running:

```
# ps aux | wc -l
```

3. Check what services are running

```
# chkconfig --list
```

```
network     0:off   1:off   2:on    3:on    4:on    5:on    6: off
keytable    0:off   1:on    2:on    3:on    4:on    5:on    6:off
pcmcia      0:off   1:off   2:on    3:on    4:on    5:on    6:off
```

4.5.2 Remove unnecessary scripts [3]

Only the following startup scripts are required, the remainder can be removed.

Service	Function
apmd	Advanced Power Management (APM) daemon executes a command when a driver reports certain events. Mainly used for laptops (eg. notify battery low) [4]
Keytable	Is used to load the appropriate keymap in the system according to the /etc/sysconfig/keyboard file and load fonts according to sbin/setsysfont script [5]
Iptables	Stores information used by the kernel to provide packed filtering services

[3] http://www.spitzner.net/linux.html, http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/pdf/rhl-rg-en-73.pdf
[4] http://linux.about.com/library/cmd/blcmdl8_apmd.htm
[5] http://www.comptechdoc.org/os/linux/startupman/linux_sukeytable.html

	Firewall requires this services to be operational
httpd	httpd is the Apache HyperText Transfer Protocol (HTTP) server program
Network	Used to specify information about the desired network configuration
Pcmcia	Required for laptop to function
random	Stores a number of values relating to generating random numbers for the kernel
	Configures raw device bindings. Assigns raw devices to block devices such as hard drive partitions
Sendmail	Mail Transport Agent
sshd	Required for ssh to run
syslog	Required for syslog daemon to run
Xinetd	Starts programs that provide Internet services when a request to the port for that service is received.

Examples of startup scripts that can be removed:

Anacron	Runs cron jobs that were left out due to downtime
Cups	Startup/shutdown for Printing system
Atd	Supports the AT commands
Gmp	Mouse support
Kudzu	Runs hardware probe
Netfs	Mounts and unmounts all NFS, SMB and NCP mount points
Nfsloc	Protocol for file sharing across TCP/IP networks
Portmap	Manages RPC connections, which are used by NFS and NIS protocols
Rhnsd	Connects periodically to Red hat network servers to check for updates, notifications and performs system monitoring tasks

1. Remove each of the above startup scripts by stopping the service

For example:

```
# cd /etc/init.d
# ./netfs stop
```

2. Delete the command from rc.d directories for each script:

```
# chkconfig --del netfs
```

3. Remove script from /etc/init.d directory:

```
# rm netfs
```

4.6 Restrict the amount of information supplied

4.6.1 Restrict the server information displayed at login

Information such as the Linux Distribution name, version, and kernel version is too much for a user to know during login. Therefore they should be removed.

```
# rm –f /etc/issue
# rm –f /etc/issue.net
```

4.7 Modifying user accounts and groups

4.7.1 Delete unnecessary user accounts and groups

During the installation of Red Hat Linux, some default user and group accounts are created. These accounts are often not used, and therefore should be removed to reduce the likelihood of unauthorised access into the system.

1. View the /etc/passwd file see which accounts are unnecessary
2. Edit the /etc/passwd file by executing the following commands to remove these unnecessary user accounts:

```
# userdel adm
# userdel lp
# userdel shutdown
# userdel halt
# userdel news
# userdel uucp
# userdel operator
# userdel games
# userdel gopher
# userdel ftp
# userdel mail
# userdel xfs
# userdel ntp
# userdel mailnull
# userdel webalizer
```

The only users that should be in the /etc/passwd file are:

```
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
gdm:
sync:x:5:0:sync:/sbin:/bin/sync
nobody:x:99:99:Nobody:/:/sbin/nologin
sshd:x74:74:Privilege-separated SSH:/var/empty/sshd:/sbin/nologin
smmsp
apache:x:48:48:Apache:/var/www:/sbin/nologin
```

3. Delete unnecessary groups from /etc/group

```
# groupdel adm
# groupdel lp
# groupdel news
# groupdel uucp
# groupdel games
# groupdel dip
# groupdel users
# groupdel uucp
# groupdel lock
```

The only Groups that should be in the /etc/group file are:

```
root:x:0:root
bin:x:1:root,bin,daemon
daemon:x:2:root,bin,daemon
sys:x:3:root,bin
tty:x:5:
disk:x:6:root
mem:x:8:
kmem:x:9:
wheel:x:10:root
man:x:15:
nobody:x:99:
users:x:100:
slocate:x:21:
sshd:x:74:
sendmail:99
apache:503:503:
```

4.8 Restrict access to files and scripts

4.8.1 Change permissions on /etc/rc.d/init.d

Ensure that only the superuser 'root' is the only user who can read/write and execute startup/stopping scripts. Otherwise a user or an unauthorised user may accidentally/purposely login and modify files.

```
# chmod –R 700 /etc/rc.d/init.d/*
```

4.8.2 Protect sensitive files from being overwritten [6]

To prevent sensitive files from being overwritten or modified by the root user, the chattr command should be used.
It makes denial of service exploits nearly impossible by setting the immutable flag on the file to prevent deletions or modifications.
These files should not require any modification after users and groups have been created.

```
# chattr +i /etc/passwd
# chattr +i /etc/shadow
# chattr +i /etc/group
# chattr+i /etc/services
```

If these files ever require modification, for example when passwords need to be reset, then run the following commands:

```
# chattr –i /etc/passwd
# chattr –i /etc/shadow
# chattr –i /etc/group
# chattr –i /etc/services
```

[6] http://www.openna.com/products/books/sol/solus.php

4.8.3 Changing SUID and GUID programs

Programs with the SGID or SUID bit set run programs with special privileges to the user executing them. This

To find all the programs with the 'S' bit set, the following commands can be executed:

```
# find / -type f \( -perm -04000 -o -perm -02000 \) -exec ls -l {} \;
```

The following files are examples of programs that may require the SGID and SUID to remain set:

```
-rwxr-sr-x 1 root man 35196 Jul 12 03:50 /usr/bin/man
-r-s--x--x 1 root root 13536 Jul 12 07:56 /usr/bin/passwd
-rwxr-sr-x 1 root mail 10932 Jul 12 10:03 /usr/bin/suidperl
-rwsr-sr-x 1 root mail 63772 Jul 12 10:03 /usr/bin/sperl5.6.0
-rwxr-sr-x 1 root slocate 23964 Jul 23 17:48 /usr/bin/slocate
-rwxr-sr-x 1 root utmp 6584 Jul 13 00:46 /usr/sbin/utempter
-rwsr-xr-x 1 root root 14184 Jul 12 20:47 /bin/su
-r-sr-xr-x 1 root root 14732 Jul 26 14:06 /sbin/pwdb_chkpwd
-r-sr-xr-x 1 root root 15340 Jul 26 14:06 /sbin/unix_chkpwd
```

The following files are examples of programs that could have their SUID or SGID bit removed:

```
-rwsr-xr-x 1 root root 34220 Jul 18 14:13 /usr/bin/chage
-rwsr-xr-x 1 root root 36344 Jul 18 14:13 /usr/bin/gpasswd
-r-xr-sr-x 1 root tty 6524 Jul 12 03:19 /usr/bin/wall
-rws--x-x 1 root root 13184 Jul 21 19:15 /usr/bin/chfn
-rws--x-x 1 root root 12640 Jul 21 19:15 /usr/bin/chsh
-rws--x-x 1 root root 5464 Jul 21 19:15 /usr/bin/newgrp
-rwxr-sr-x 1 root tty 8500 Jul 21 19:15 /usr/bin/write
-rwsr-xr-x 1 root root 6288 Jul 26 10:22 /usr/sbin/usernetctl
-rwsr-xr-x 1 root root 20540 Jul 25 07:33 /bin/ping
-rwsr-xr-x 1 root root 55356 Jul 12 05:01 /bin/mount
-rwsr-xr-x 1 root root 25404 Jul 12 05:01 /bin/umount
-rwxr-sr-x 1 root root 4116 Jul 26 10:22 /sbin/netreport
```

To remove the SUID and SGID bit:

```
# chmod a-s /usr/bin/chage
# chmod a-s /usr/bin/gpasswd
# chmod a-s /usr/bin/wall
# chmod a-s /usr/bin/chfn
# chmod a-s /usr/bin/chsh
# chmod a-s /usr/bin/newgrp
# chmod a-s /usr/bin/write
# chmod a-s /usr/sbin/usernetctl
# chmod a-s /bin/ping
# chmod a-s /bin/mount
# chmod a-s /bin/umount
# chmod a-s /sbin/netreport
```

4.9 Restrict Functions and Access

4.9.1 Set login time for root account [7]

Directory: /etc/

File: /profile
Original: Look for HISTSIZE=1000 within the file
Action: Add the following line after the HISTSIZE line:
TMOUT=7200

4.9.2 Disable Ctrl+Alt+Del Keyboard Shutdown [8]

Users should not be allowed to shutdown the webserver at anytime. Only root should have this privilage. Therefore the ctrl+alt+del keyboard shutdown option should be disabled.

Directory:	/etc/
File:	inittab
Original:	ca::ctrlaltdel:/sbin/shutdown -t3 -r now
Action:	Add a '#' to comment out the line: # ca::ctrlaltdel:/sbin/shutdown -t3 -r now

To execute this change:

```
# /sbin/init q
```

4.9.3 Restrict the Virtual Console and TTY devices the root user is allowed to login

It is unnecessary for multiple user accounts to be created for this web server, because once the services have started and the webserver is running, everyday operations should be automated.

However, it is always recommended that any user accounts that are created, should be restricted to login as their own username at the console or tty. If a user wants to perform Super User tasks, then they will need to first login as themselves, then su - This ensures an accurate audit of what actions each user has performed, which could be used in the future for forensics.
It also restricts the devices that this user is allowed to logon to, reducing the number of entry points into the system by anyone who has the root password.

[7] http://www.openna.com/products/books/sol/solus.php
[8] http://www.openna.com/products/books/sol/solus.php

Directory: /etc
File: securetty
Action:
Disable all devices except for vc/1 and tty1, by commenting out the line
Change vc/1 to 1 and tty1 to 1.

Original	New
Vc/1	1
Vc/2	#Vc/2
Vc/3	#Vc/3
Vc/4	#Vc/4
Vc/5	#Vc/5
Vc/6	#Vc/6
Vc/7	#Vc/7
Vc/8	#Vc/8
Vc/9	#Vc/9
Vc/10	#Vc/10
Vc/11	#Vc/11
Tty1	Tty1
Tty2	#Tty2
Tty3	#Tty3
Tty4	#Tty4
Tty5	#Tty5
Tty6	#Tty6
Tty7	#Tty7
Tty9	#Tty9
Tty10	#Tty10
Tty11	#Tty11

4.9.4 Changing password Restrictions [9]

The default minimum password length is 5. However, it should be a minimum 8 mixed alphanumerics. This makes it difficult for anyone doing a password crack to obtain any passwords onto the system.

To change this default, follow the steps below.

1. Remove the following line from **/etc/pam.d/passwd**

Remove

```
password required /lib/security/pam_stack.so service=system-auth
```

2. Remove the following lines from **/etc/pam.d/system-auth/**

Remove

```
password required /lib/security/pam_cracklib.so retry=3
password sufficient /lib/security/pam_unix.so nullok use_authtok md5
shadow
password required /lib/security/pam_deny.so
```

[9] http://www.openna.com/products/books/sol/solus.php

3. To enforce password length, the following lines need to be added to the **/etc/pam.d/passwd** file:

Add

```
password required /lib/security/pam_cracklib.so retry=3 minlen=12
password sufficient /lib/security/pam_unix.so nullok use_authtok md5
shadow
password required /lib/security/pam_deny.so
```

After adding the above lines, the /etc/pam.d/passwd file should look like this:

```
#%PAM-1.0
auth required /lib/security/pam_stack.so service=system-auth
account required /lib/security/pam_stack.so service=system-auth
password required /lib/security/pam_cracklib.so retry=3 minlen=12
password sufficient /lib/security/pam_unix.so nullok use_authtok md5
shadow
password required /lib/security/pam_deny.so
```

After saving the /etc/pam.d/password file, the system-auth file should automatically update:

```
#%PAM-1.0
# This file is auto-generated.
# User changes will be destroyed the next time authconfig is run.
auth required /lib/security/pam_env.so
auth sufficient /lib/security/pam_unix.so likeauth nullok
auth required /lib/security/pam_deny.so
account required /lib/security/pam_unix.so
session required /lib/security/pam_limits.so
session required /lib/security/pam_unix.so
```

4.9.5 Remove 'r' scripts

The **r**sh, **r**login and **r**cp were unix utilities which were created for remote login and copying.
Vulnerabilties in the rlogin program are described in Cert CA-1997-06[10]. This cert advisory, details how numerous rlogin programs contain a programming defect where an internal buffer could be overflowed and code executed as root.

Therefore rsh, rlogin, rcp, telnet and ftp utilities are quickly being replaced by SSH program.
These tools should be disabled or redirected to ssh equivilant programs to prevent dangerous files being uploaded and executed.
Rsh and rlogin is replaced by **ssh**
Rcp is replaced by **scp**
FTP is replaced by **sftp**

[10] http://www.cert.org/advisories/CA-1997-06.html

In this guide, we use OpenSSH.

1. Change to zero permissions
```
# chmod 0 rsh
# chmod 0 rcp
# chmod 0 rlogin
```

All 3 files should now have no permissions

```
---------  1       root    root    11072 Jan 25        2003   rlogin
```

4.9.6 Set umask for users

Setting files to umask of 027 will allow people in your group to have permissions to read your files and not have the ability to write to them. Other users will have no access to the files.

4. Append to end of file the following line to /etc/profile:
   umask 027

5. Append the following line to /root/.bash_profile:
   umask 027

6. Append the following line to /etc/csh.login:
   umask 027

4.9.7 Limit system resource usage

It is a good idea to limit system resources, and preventing core dumps. This will limit the effects of a Denial of Service attack and can be achieved by modify /etc/security/limits.conf

1 Add to the end of the /etc/security/limits.conf file the following:

```
#prevent core dumps
*    hard  core  0
#limit user processes per user to 150
*    soft  nproc 100
*    hard  nproc 100
```

4.10 TCP Wrappers configuration

TCP/IP has always been known to be insecure, and many intrusions have been caused because of it.
TCP_Wrappers is a host-based access control to network services that works by restricting the services that can be used.

When an attempt is made to connect to a tcp-wrapped service, the following

files are used to determine whether or not a host is permitted to connect:
Hosts.allow allows pre-defined ip addresses access to specific daemons, and denies everything else. This file is read before hosts.deny and consists of the daemon name and the client address it is restricted to.
Hosts.deny lists all ip addresses that are denied access to a daemon, and allows everything else.

If a host is not listed in either of these two files, then it is allowed access. Another useful feature of tcp wrappers is that it can execute commands with a client before denying a connection, so you could send an email to the administrator letting them know of a potential intrusion.

4.10.1 Modify /etc/hosts.allow

For this web server to operate, you will require the sshd, sendmail protocols to be enabled. This allows the 2nd laptop access the ssh daemon. As sendmail is restricted to the loopback address, it will need to be added to the /etc/hosts.allow file to allow it to communicate with the sendmail daemon.

```
#
# hosts.allow       This file describes the names of the hosts which are
#                   allowed to use the local INET services, as decided
#                   by the '/usr/sbin/tcpd' server.
#
SSHD:           165.56.10.1
Sendmail:       127.0.0.1
```

4.10.2 Modify /etc/hosts.deny

The /etc/hosts.allow is the first file to be read, and therefore all connections that are allowed and necessary should be already accepted.

Once the /etc/hosts.deny file is read, all other hosts should be denied access. Therefore whenever anything is specified in the /etc/hosts.allow file, the /etc/hosts.deny file should always contain everything else (ALL: ALL)

```
#
# hosts.deny            This file describes the names of the hosts
which are
#                 *not* allowed to use the local INET services, as
decided
#                 by the '/usr/sbin/tcpd' server.
#
ALL:  ALL
```

More information about TCP Wrappers, can be found at the Red Hat Linux website at: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/s1-tcpwrappers-access.html

4.11 IPTABLES configuration

Setting iptables allows you to control what network traffic is allowed in. It inserts and deletes rules from the kernel's packet filtering table. It basically performs like a firewall.

The startup configuration files is located in */etc/init.d/iptables*. This file defines where the rules are located.

The rules are located in */etc/sysconfig/iptables* if you have used the GNOME lokkit during initial installation. Manual customisation of this file is not recommended. The high security option which should be chosen during initial setup, disables almost all network connects except DNS replies and DHCP so that network interfaces can be activated.[11]

The file should contain the following rules:

```
- A    RH-Lokkit-0-50-INPUT  -p    tcp  -m  tcp  --dport    443  --syn   -j   ACCEPT

- A    RH-Lokkit-0-50-INPUT  -p    tcp  -m  tcp  --dport    22  --syn   -j   ACCEPT
```

The following commands are simple examples of how you can to modify the iptables configuration to secure your server.

Command	Function
`iptables –A INPUT –s 134.32.0.0 –j DROP`	Drop all traffic coming from 10.1.x.x network
`iptables –L`	Lists all policies

Example. Output of INPUT rule

```
Chain INPUT (policy ACCEPT)

Target          prot opt  source          destination
DROP            ALL  --   10.1.0.0        anywhere
```

[11] http://www.redhat.com/docs/manuals/linux/RHL-8.0-Manual/custom-guide/s1-basic-firewall-gnomelokkit.html

5. Step-by-Step Guide: Installation of Apache

Apache is a secure open source web server that comes with Red Hat Linux 9.0.

More information about Apache can be found at: http://www.apache.org/

Older versions of Apache (1.2.2 and above, 1.3 through 1.3.24, 2.0 through 2.0.36) have a vulnerability which CERT have been advised. The vulnerability in handling chunk-encoded HTTP requests, may allow remote attackers to execute code. For more information see: http://www.cert.org/advisories/CA-2002-17.html

5.1 Preparation

Download / Obtain	Site / Location
Gcc-3.2.2-5.i386.rpm Kernel-headers-2.2.16-3.i386.rpm	Red Hat Linux Installation CD 2 ftp://ftp.redhat.com/redhat/redhat-6.2/SRPMS/SRPMS/kernel-2.2.14-5.0.src.rpm OR ftp://updates.redhat.com/6.2/en/os/i386/kernel-headers-2.2.16-3.i386.rpm)
binutils-2.13.90.0.18-9.i386.rpm Glibc-devel-2.3.2-27.9.i386.rpm	Red Hat Linux Installation CD 1 ftp://rpmfind.net/9/en/os/i386/glibc-devel-2.3.2-27.9.i386.rpm)
Signatures httpd-2.0.47.tar.gz.asc	http://httpd.apache.org/dev/verification.html ftp://ftp.redhat.com/pub/redhat/linux/9/en/os/i386/RedHat/RPMS/httpd-2.0.47.tar.gz.asc

Pre-requisites:

- Ensure you have network connectivity
- Ensure you have at least 50 MB of space
- Download all files
- C Compiler installed

NOTE: In order for the C Compiler GCC to install, the following dependencies need to be installed:

- Kernel-headers > 2.2.1
- Binutils >= 2.12.90.0.7-1
- Glibc-devel >= 2.2.90-12

5.1.1 Integrity Check

Each file that is downloaded from an external source, needs to be checked to ensure that it comes from a reliable source.

```
# gpg httpd-2.0.44.tar.gz.asc
gpg: Signature made Mon 07 Jul 2003 15:56 BST using DSA key ID DE885DD3
gpg: Can't check signature: public key not found
The public key is not stored locally, therefore needs to be downloaded and verified from a public
keyserver.

# gpg –keyserver pgpkeys.mit.edu –recv-key DE885DD3
gpg: key DE885DD3: public key "Sander Striker <striker@apache.org>" imported
gpg: Total number processed: 1
gpg:               imported: 1
```

In this example, the public key for an entity known as 'Sander Striker <striker@apache.org>' has been received. However, there is no way of verifying that the person known as Sander Striker created this key. Therefore we need to verify the release signature again.

```
  # gpg httpd-2.0.47.tar.gz.asc
gpg: Signature made Mon 07 July 2003 15:56.49 BST using DSA key ID DE885DD3
gpg: Good signature from "Sander Striker <striker@apache.org>"
gpg:                 aka "Sander Striker <striker@striker.nl>"
gpg: checking the trustdb
gpg: no ultimately trusted keys found
gpg: WARNING: This key is not certified with a trusted signature!
gpg:          There is no indication that the signature belongs to the owner.
Fingerprint: 4C1E ADAD B4EF 5007 579C  919C 6635 B6C0 DE88 5DD3
```

We have confirmed that the signature is good and that the file has not been tampered with, now we need to ensure that the real user, Sander Striker, created the key.
The following command, confirms the key fingerprint of the public key

```
  # gpg --fingerprint DE885DD3
pub  1024D/DE885DD3 2002-04-10 Sander Striker <striker@apache.org>
    Key fingerprint = 4C1E ADAD B4EF 5007 579C  919C 6635 B6C0 DE88 5DD3
uid                  Sander Striker <striker@striker.nl>
sub  2048g/532D14CA 2002-04-10
```

5.1.2 Extract Installation files

Once verified, the files need to be extracted.

```
# gunzip httpd-2.0.47.tar.gz
# tar xvf httpd-2.0.47.tar
```

5.2 Configure Apache

The purpose of the web server needs to be careful considered, as this will affect which modules you should or shouldn't install. Modules that will not be used should be disabled to avoid potential break-ins in the event of new security vulnerabilities being developed.
A detailed description of each module can be found at: http://httpd.apache.org/docs/mod/.

To maintain security by only installing the nessessary modules for the web server, the following modules are the only ones required for the web server to run.

Module Name	Description
httpd_core	The core Apache features. Module is required in every Apache installation.
mod_access	Provides access control based on client hostname, IP address, or other characteristics of the client request. This module is needed to use "order", "allow" and "deny" directives, therefore should remain enabled.
mod_auth	Allows the implementation of user authentication using text files (HTTP Basic Authentication), which was specified in functionality assumptions.
mod_dir	Required to search and serve directory index files: "index.html", "default.htm", etc.
mod_log_config	Required to implement logging of requests made to the server.
mod_mime	Required to set the character set, content-encoding, handler, content-language, and MIME types of documents.

Table. Apache Modules. Maj, Artur [12]

[12] http://www.securityfocus.com/infocus/1694

Steps to Configuring Apache

1. Change into httpd directory

```
# cd httpd-2.0.47
```

2. Make a directory for apache

```
# mkdir /usr/local/apache
```

3. Change permissions for apache

```
# chmod 755 /usr/local/apache
```

4. Configure using the modules chosen

```
# ./configure --prefix=/usr/local/apache --disable-
module=all --server-
     uid=apache --server-gid=apache --enable-
module=access --enable-
     module=log_config --enable-module=dir --enable-
module=mime --enable-module=auth
```

5.3 Compile Apache

```
# make
```

5.4 Install Apache

```
# make install
```

5.5 Customise Apache

1. Edit the httpd.conf file

```
# vi usr/local/apache/conf/httpd.conf
```

(See Appendix C)

5.6 Hardening Apache

2. Change permissions to make root the owner of the /usr/local/apache directory and files.

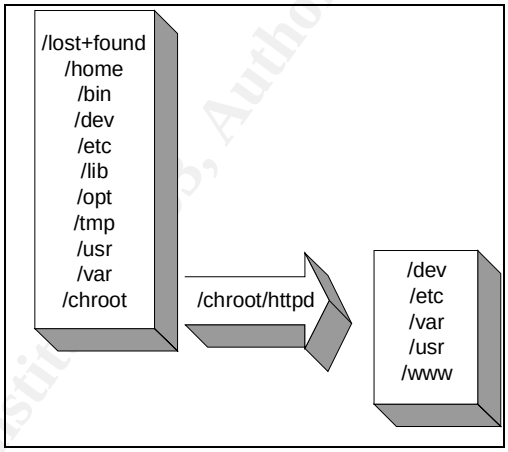
```
# chown –R root:sys /usr/local/apache
```

5.6.1 Chroot File system

If apache is compromised, an attacker could have access to the whole file system. If scripts have been written for the web server that allow security vulnerabilities, then this could compromise the system.

Creating a chroot 'jail'[13] can prevent these two events from occurring by chrooting the httpd daemon, and creating a separate file system for apache to run in.

It basically limits the Apache process's to its own file system, so that the daemons are isolated from the rest of the server.



source: Jacqui Chau

Steps to Creating a chroot system:

1. Ensure you are logged on as the superuser 'root'
3. Create new root directory structure under the /chroot/httpd

```
# mkdir -p /chroot/httpd/dev
# mkdir -p /chroot/httpd/etc
# mkdir -p /chroot/httpd/var/run
# mkdir -p /chroot/httpd/usr/lib
# mkdir -p /chroot/httpd/usr/libexec
# mkdir -p /chroot/httpd/usr/local/apache/bin
# mkdir -p /chroot/httpd/usr/local/apache/logs
```

[13] http://packetstormsecurity.nl/papers/unix/Securing-Optimizing-RH-Linux-1_2.pdf

```
# mkdir -p /chroot/httpd/usr/local/apache/conf
# mkdir -p /chroot/httpd/www
```

4. Ensure all the above directories have access rights 0755
5. Create null device for the Apache user
6. Check the current status of /dev/null

```
# ls -al /dev/null
```

```
crw-rw-rw-   1  root root   1,  3 Jan 30 2003 /dev/null
```

7. Make special device file

```
# mknod /chroot/httpd/dev/null c 2 2
```

8. Check status of special file

```
crw-r—r--   1  root root   2,  2 Nov 12 19:25  chroot/httpd/dev/null
```

9. Change ownership and permissions for the file

```
# chown root:sys /chroot/httpd/dev/null
# chmod 666 /chroot/httpd/dev/null
```

```
crw-rw-rw-   1  root sys   2,  2 Nov 12 19:25  chroot/httpd/dev/null
```

10. Check dependencies by listing the dynamic dependencies of shared libraries or executable files

```
# ldd /usr/local/apache/bin/httpd
```

```
libaprutil-0.so.0 => /usr/local/apache/lib/libaprutil-0.so.0 (0x40017000)
libexpat.so.0 => /usr/local/apache/lib/libexpat.so.0 (0x4002a000)
libapr-0.so.0 => /usr/local/apache/lib/libapr-0.so.0
librt.so.1 => /lib/librt.so.1
libcrypt.so.1 => /lib/libcrypt.so.1
libnsl.so.1 => /lib/libnsl.so.2
libdl.so.2 => /lib/libdl.so.2
libpthread.so.0 => /lib/tls/libpthread.so.0
libc.so.6 => /lib/tls/libc.so.6
/lib/ld-linux.so.2 => /lib/ld-linux.so.2
```

11. Make directories /chroot/lib and /chroot/lib/tls
12. Copy the files from /usr/local/apache to new /chroot filesystem.

```
# cp /usr/local/apache/bin/httpd
/chroot/httpd/usr/local/apache/bin/
# cp /usr/local/apache/lib/libaprutil-0.so.0 /chroot/httpd
/usr/local/apache/lib/
# cp /usr/local/apache/lib/libexpat.so.0
/chroot/httpd/usr/local/apache/lib/
# cp /usr/local/apache/lib/libapr-0.so.0
/chroot/httpd/usr/local/apache/lib/
# cp /lib/librt.so.1 /chroot/httpd/lib/librt.so.1
# cp /lib/libcrypt.so.1 /chroot/httpd/lib/libcrypt.so.1
```

```
# cp /lib/libnsl.so.2 /chroot/httpd/lib/libnsl.so.2
# cp /lib/libdl.so.2 /chroot/httpd/lib/libdl.so.2
# cp /lib/tls/libpthread.so.0
/chroot/httpd/lib/tls/libpthread.so.0
# cp /lib/tls/libc.so.6 /chroot/httpd/lib/tls/libc.so.6
# cp /lib/ld-linux.so.2 /chroot/httpd/lib/ld-linux.so.2
# cp /usr/sbin/httpd /chroot/httpd/usr/sbin/
# cp -r /etc/httpd /chroot/httpd/etc/

# cp /libssl.so.4 /chroot/httpd/lib
# cp /lib/libcrypto.so.4 /chroot/httpd/lib
# cp /lib/libresolv.so.2 /chroot/httpd/lib
```

13. Copy the following configuration files:

```
# cp /etc/hosts /chroot/httpd/etc/
# cp /etc/host.conf /chroot/httpd/etc/
# cp /etc/resolv.conf /chroot/httpd/etc/
# cp /etc/group /chroot/httpd/etc/
# cp /etc/passwd /chroot/httpd/etc/passwd
# cp /usr/local/apache/conf/mime.types
                    /chroot/httpd/usr/local/apache/conf/
```

14. Remove all users from the /chroot/httpd/etc/passwd file except for 'apache' and 'nobody' users.

```
apache:x:503:503::/home/apache:/bin/bash
```

15. Remove all groups from /chroot/httpd/etc/group file except for 'apache' and 'nobody groups.

```
apache:x:503:
```

16. Secure the chroot directories for better security. The 'chattr' command prevents anyone from modifying these system critical files:

```
# chattr +I /chroot/httpd/etc/passwd
# chattr +I /chroot/httpd/etc/group
# chattr +I
/chroot/httpd/etc/httpd/conf/httpd.conf
# chattr +I /chroot/httpd/etc/resolv.conf
# chattr +I /chroot/httpd/etc/hosts
```

17. Ensure localtimezone file is copied over, to ensure an accurate log time.

```
# cp /etc/localtime /chroot/httpd/etc/
```

18. Configure syslogd to log all information to new chroot system

```
rm –f /var/lock/subsys/httpd /var/run/httpd.pid
```

19. The default httpd script file for Apache starts the daemon "httpd" within the default system.

We do not want to run the httpd from the default location, therefore we must modify the httpd script file (/etc/rc.d/init.d/httpd) this so that httpd is started from the chroot system.

Within the /etc/rc.d/httpd file:

Replace:

```
daemon httpd
```

To read:

```
/usr/sbin/chroot /chroot/httpd/ /usr/sbin/httpd
-DSSL
```

Replace:

```
rm -f /var/run/httpd.pid
```

To read:

```
rm -f /chroot/httpd/var/run/httpd.pid
```

20. Restart syslogd daemon with the following command:

```
#  /etc/rc.d/init.d/syslog restart
```

```
Shutting down kernel logger: [ OK ]
Shutting down system logger: [ OK ]
Starting system logger: [ OK ]
Starting kernel logger: [ OK ]
```

5.7 Further configuration changes to Apache

Replace */chroot/httpd/usr/local/apache/conf/httpd.conf* with the file in Appendix B

5.8 Test Apache

1. Copy Apache configuration files into chroot file system.

```
# cp /usr/local/apache/conf/httpd.conf
                          /chroot/httpd/usr/local/apache/conf/
# cp /usr/local/apache/htdocs/index.html.en
                         /chroot/httpd/www/index.html
```

2. Edit the / /chroot/httpd/usr/local/apache/conf/httpd.conf to the following:

```
DocumentRoot "/www"
```

3. Run the server:

```
# chroot /chroot/httpd /usr/local/apache/bin/httpd
```

If there are any errors, then check to ensure you have copied all files, and that they are set to the correct permissions.

Once you have tested that apache is starting, you need to remove the unnecessary apache files from the system. [14]

```
# rm -rf /var/log/httpd/
# rm -rf /etc/httpd/
# rm -rf /home/httpd/
# rm -f /usr/sbin/httpd
```

4. Ensure httpd is running

```
# ps -ef |grep httpd
```

[14] http://packetstormsecurity.nl/papers/unix/Securing-Optimizing-RH-Linux-1_2.pdf

6. Step-by-Step Guide: Installation of OpenSSH

OpenSSH is required for the server to transfer files between the web server and the 2nd laptop.

As telnet, rlogin, rsh, ftp are becoming increasingly less secure, the need for ssh has become more imperative.
OpenSSH is a free version of the SSH protocol suite of network connectivity tools which an increasing number of people on the Internet are starting to utilize. Telnet, rlogin, ftp, and other similar program's transmit passwords across the Internet unencrypted. OpenSSH encrypts all traffic to effectively stop eavesdropping, connection hijacking, and other network attacks.

SSH is the standard for secure communication, however it should be noted that it is not 100% secure. Buffer overflows is a security vulnerability which has been identified by CERT: http://www.cert.org/advisories/CA-2002-36.html
Connecting only to trusted hosts, using firewalls and tcpwrappers can reduce this risk considerably.

The OpenSSH package (openssh-clients-3.5p1-6) should be installed during the initial Red Hat Linux installation. (Should be one of the packages you have selected)
It is recommended to run a version above 3.7, which can be downloaded from: ftp://ftp.openbsd.org/pub/OpenBSD/OpenSSH/ (At time of writing) [15]

6.1 Openssh Configuration

The main configuration file for Openssh is the /etc/ssh/sshd_config.

As we will only be using protocol 2, all protocol 1 settings do not need to be set.

Change the **`/etc/ssh/sshd_config`** to the configuration below:

```
Port 22
Protocol 2
#ListenAddress 0.0.0.0
#ListenAddress::

#HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
#HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_key
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
```

[15] http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2003-0693

```
#Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 768

#Logging
#obsoletes QuietMode and FascistLogging
#SysogFacility AUTH
SyslogFacility AUTHPRIV
LogLevel INFO

#Authentication:
LoginGraceTime 120
PermitRootLogin no
StrictModes yes


RSAAuthentication no
PubKeyAuthentication yes
AuthorizedKeysFile .ssh/authorized_keys

#rhosts authentication should not be used
RhostsAuthentication no
IgnoreRhosts yes
# For this to work you will also need host keys in
/etc/ssh/ssh_known_hosts
RhostRSAAuthentication no
# Similar for protocol version 2
HostBasedAuthentication no
#Change to yes if you don't trust ~/.ssh/known_hosts for
# RhostsRSAAuthentication and HostbasedAuthentication
IgnoreUserKnownHosts no

#To disable tunneled clear text passwords, change to no here!
PasswordAuthentication no
PermitEmptyPasswords no

#Change to no to disable s/key passwords
ChallengeResponseAuthentication no

#Kerberos options
KerberosAuthentication no
KerberosOrLocalPasswd no
KerberosTicketCleanup no

#AFSTokenPassing no

#Kerberos TGT Passing only works with the AFS kaserver
#KerberosTgtPassing no

#Set this to 'yes' to enable PAM keyboard-interactive authentication
#Warning: enabling this may bypass the setting of
'PasswordAuthentication'
PAMAuthenticatioViaKbdInt no

X11Forwarding no

PrintMotd yes
```

```
PrintLastLog yes
KeepAlive yes
UseLogin no
UsePrivilageSeparation yes
PermitUserEnvironment no
Compression yes

MaxStartups 10
#no default banner path
Banner /etc/sshbanner
VerifyReverseMapping no

#Override default of no subsystems
Subsystem   sftp  /usr/libexec/openssh/sftp-server
```

6.2 Starting SSHD

1. Start the sshd service by executing the following command:

```
# /sbin/service sshd start
```

The following banner should appear:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle
attack)!
It is also possible that the RSA host key has just been changed.
```

Host Keys should be created in etc/ssh/ssh_host*key*

After making any changes to the /etc/ssh/sshd_config file after the initial startup, you will need to restart the sshd to push out the changes:

```
# /etc/init/d/ssh restart
```

```
Stopping sshd:                [OK]
Starting sshd:                [OK]
```

Check that that the sshd is running:

```
# ps –ef |grep sshd
```

6.3 **Using OpenSSH**

When you first connect to a host, a message will appear, advising that the connection cannot be established because it does not appear in the list of 'known hosts'. You will need to check the DSA fingerprint and only accept the DSA fingerprint if it is valid. This is an important security feature, as you do not want to allow any other hosts except the 2nd laptop to access this server.

If other hosts were permitted to access this server, then they could potential compromise the web server.

```
The authenticity of host 'personal.net' can't be
established.
DSA key fingerprint is
94:68:3a:51:df:g6:7a:9b:01:5e:b3:07:66:a2:22:0d.
Are you sure you want to continue connecting (yes/no)?
```

You should have an account on the 2nd laptop, so you can enter the username and password when prompted.

Once you have established a trusted communication between the two server and laptops, you will be able to securely transfer files (scp/sftp) and execute commands (ssh) through this encrypted channel.

6.4 Securing OpenSSH

6.4.1 Generating authorisation keys

To allow particular users login via key pair authentication rather than via clear text passwords, authorisation keys need to be generated. Keys need to be exchanged in order to create a trusted channel.

The default directory where user keys are stored are in *~/.ssh/id_rsa*

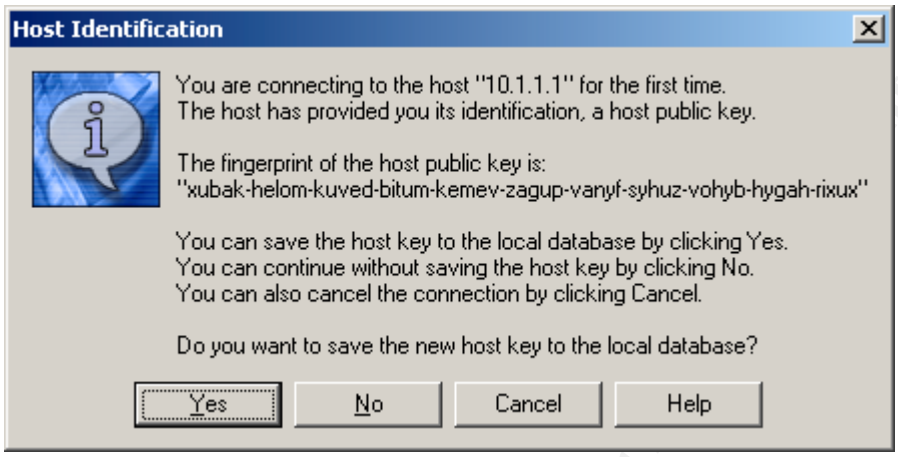
The public key is written to: *~/.ssh/id_rsa.pub*

The private key is written to *~/.ssh/id_rsa*

Copy the contents of *~/.ssh/id_rsa.pub* to *~/.ssh/authorized_keys* on the machine to which you want to connect.

If the file *~/.ssh/authorized_keys* does not exist, you can copy the file ~/.ssh/id_rsa.pub to the file *~/.ssh/authorized_keys* on the other machine.

For example. If using the Windows version of SSH Secure Shell. When you first connect, the following message will appear:



You will need to save the Web server's host public key to the remote computers local database.

6.4.2 Troubleshooting OpenSSH

If you have problems connecting to the 2nd laptop, then check the iptables and tcpwrappers configuration. If you have an additional firewall, then that should also be reviewed.

```
PasswordAuthentication no
```

Password authentication will send the password in clear text. Therefore it is recommended that you keep the default to no, however for troubleshooting.

7. Step-by-Step Guide: Installation of Snort

Snort is an open source network intrusion detection system, capable of performing real-time traffic analysis and packet logging.
It can be used to detect various attacks and probes. For example buffer overflows, stealth port scans, CGI attacks.
Snort is particularly good at watching web servers for web-based attacks, and can be run on the web server itself. With the right rule-sets configured, there are not many attacks that can pass through undetected.
See www.snort for more information.

For a small business / home based web site, it is not necessary to use all of snorts capabilities. The web server should be set to listen to all traffic coming to the specific ip address of the web server and nothing else.

Ensure that a large amount of space is assigned for the logs, as this could quickly fill up the web server and cause a denial-of-service. Fine-tuning the rulesets to report particular alerts can also help reduce the amount of space taken up with logs.

Implementation of Snort allows you to detect, log and minimise most web server attacks before they become too serious.

7.1 Preparation

Download:	Site
snort-2.0.4.tar.gz	http://www.snort.org/dl/
MD5Sums for snort-2.0.4.tar.gz	http://www.snort.org/dl/
PGP Public Key	http://www.snort.org/public-key.html
ftp download site	ftp://rpmfind.net/linux/redhat/9/en/os/i386/RedHat/RPMS/

7.2 Pre-requisites

Libpcap needs to be installed on the sensor before snort configuration can begin.

7.3 Integrity Check

It is vital to check the integrity of any libpcap files downloaded. CERT published an advisory on a libpcap distribution that contained a Trojan horse in November 2002. [16]

Libpcap should have been installed in the initial installation of Red Hat.

The snort PGP should be downloaded from the snort.org site.

1. Open the PGP file and check the signature.

```
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.0.7 (GNU/Linux)

iD8DBQA/vSRUlEkV6hlG5KERAutMAJ49WjuiEa+2ucE93UF4o01SUvqSlACcDPGJ
xeigcCGjggpz4wGAMei1IqQ=
=KR0I
-----END PGP SIGNATURE-----
```

1. Import the keys into the gpg database:

```
# gpg –import snort.pgp
# gpg –verify snort-2.0.5.tar.gz.asc snort-2.0.5.tar.gz
```

2.. Verify the file

```
# gpg –verify MD5Sum
```

7.4 Extraction of Files

1. Unzip file

```
# gunzip snort-2.0.4.tar.gz
```

2. Extract file

```
# tar xvf snort-2.0.4.tar.gz
```

3. Configure file.

Once the./configure command is executed, the script checks that current configuration status. If modules have not been created, then the script will create them.

[16] http://www.cert.org/advisories/CA-2002-30.html

```
# ./configure
```

4. Configure and install Snort

```
# make
```

5. Change permissions on snort directory

```
# chmod –R 750 /var/snort
```

5. Record the packets to a log file

```
#/root/snort2.0.4/src
# ./snort –dev –l /var/snort/log
```

7.5 Snort tests:

Packet mode:

```
# ./snort -dev -l ./log
```

Network Intrusion Detection:

```
# root/snort-2.0.4/src/snort -dev -l var/log/snort
`                    -h 10.1.1.1/24 -c snort.conf
```

7.6 Modify Rulesets

A startup file for snort is not required, as it only needs to be run on a need-to-need basis, but still needs to be used regularly. This paper does not cover how to build the snort rules, but more information can be found at: http://www.snort.org/docs/writing_rules

Always keep in mind that the order in which the options appear in the ruleset, will affect the order in which they are scanned. The rules should be visited on a regular bases to keep up to date with the latest security vulnerabilities.

8. Step-by-Step Guide: Installation of Tripwire

Tripwire is a very powerful open source file integrity program created to monitor changes in a select number of files identified by you (or you can use the defaults).
It provides reports on any changes to any of those files so you can determine whether the changes occurred due to normal operations or whether they where caused by an intruder trying to manipulate or hack into your system. Tripwire has the ability to update system baselines if they report on files that are constantly being modified by yourself. In the event of a suspected 'break-in', you can shut down the system, disconnect the server from the network, begin to repair the system and start forensic activities.

8.1 Preparation

Download:	Site
Tripwire-2.3.1-17.src.rpm	ftp://rpmfind.net/linux/redhat/9/os/i386/SRPMS/

8.2 Installation

1. Install Tripwire

   Insert installation CD 3

   ```
   # rpm –uvh tripwire-2.3.1-17.rpm
   ```

2. Confirm that tripwire is installed

   ```
   # rpm -q tripwire
   ```

Once you have confirmed that tripwire has been installed, check that the /etc/tripwire directory has been created.

Three files should have been generated during this install:

- **twcfg.txt:** High level tripwire configuration file *(See appendix D)*
- **twinstall.sh**: Tripwire configuration script
- **twpol.txt**: rules for various files and directories

3. Edit the twcfg and twpol.txt files

Only in special circumstances, the twcfg files needs to be modified.

The twpol.txt files is where you can edit your rules for files and directories.

It may contain rules for files that currently do not exist on the system. Do not remove these, as hackers could use these directories and files as a method of cracking into a system.

4. Run the configuration script

```
# /etc/tripwire/twinstall.sh
```

It will ask you Creating key files. You will be asked to enter a site keyfile passphrase and a local keyfile passphrase. Choose passwords that are at least 8 characters, with mixed alphnumerics.

The configuration file and policy file will then be signed.

5. Initialise the Tripwire database

```
# /usr/sbin/tripwire –init
```

You will get messages stating '*no such file or directory*'. This is fine, as long as at the end you get the message "*Wrote database file:/var/lib/tripwire/Webserver.twd. The database was successfully generated"*

8.3 Integrity Check

The "integrity check", checks to see whether any files have been changed or modified since you performed your first report.

Make a secure directory /chroot/securedirectory that is not within the /root directory. Putting the secure directory in the /root directory will cause Tripwire to report it as an add/change.

/usr/sbin/tripwire –check | tee > /chroot/securedirectory/tripwirelog.log

This will write the results to a file called *tripwirelog.log*

Print the results of the database/report

Each time you perform a –check, the file is timestamped. Use this file to

compare the changes that have been made to you system since your previous check.

```
# /usr/sbin/twprint -m r --twrfile

                    /var/lib/tripwire/report/timestamp.twr | less
```

The report will display the following:

- Report Summary
- Rule Summary
- Object Detail
- Error Report

All these pieces of information are important, as they indicate whether someone has tampered with your system critical files. If you have files that have changed in which you did not modify yourself, then disconnect the system from the network and investigate.

The script `tripwire --check` should be located in your */etc/cron.daily* directory, and should execute daily. It should be setup so you receive email notification for any usual changes to your system.

9. Ongoing Maintenance

9.1 Patches

Hacker's are getting smarter, and this therefore makes the importance of keeping your system up-to-date and secure vital. Software vendors are constantly updating security patches and bug fixes to keep ahead of potential exploits and vulnerabilities.

It is therefore imperative that these websites are constantly visited and patches installed.

Software	Site	Frequency
Red Hat Linux	ftp://rpmfind.net/linux/redhat/updates/9/en/os/i386/	1 month
Apache	http://httpd.apache.org/dev/verification.html http://httpd.apache.org/security_report.html	1 month
Snort	http://www.snort.org/ http://www.snort.org/dl/	3 months
Tripwire	ftp://rpmfind.net/linux/redhat/updates/9/en/os/i386/	6 months
OpenSSH	ftp://rpmfind.net/linux/redhat/updates/9/en/os/i386/ http://www.openssh.com/ http://www.openbsd.org/errata.html	3 months

For all RPM's, the 'Freshen' command should be used:

```
# rpm -Fvh <filename>.rpm
```

Alternatively the automated Red Hat Linux Agent can be setup, if you feel comfortable with having a connection open continuously.

```
Red Hat Setup Agent
Enter User Account (personal username and password for admin)
Date and Time:
```

```
Sound Card:
Red Hat Network screen: (registers system with a complementary Demo
account from Red Hat Network, so now you can receive the latest
software packages.
Answer: Yes.


GPG keyring does not contain the Red Hat Inc public key. Without it,
you will be unable to verify that packages Update Agent downloads are
securely signed by Red hat.
Your Update Agent Options specify that you want to use GPG. Install
Key? Yes
```

9.1.1 Snort Rule Maintenance

The latest rule set for snort can be downloaded from:
http://www.snort.org/dl/rules/
You will not have to use all these rules, so modifications of the ruleset will need to be customised again.

9.2 Security Bulletin Boards and Mailing lists

Keeping aware of the latest vulnerabilities of each software package is recommended. Many software vendors provide notification services to their customers through mailing lists, however it is difficult for 'free' or open source software vendors to guarantee this type of service.

Therefore you should browse the sites below as frequently as possible to keep up-to-date on the latest security vulnerabilities.

Software	List/Site	Description
Red Hat Linux	http://lwn.net/Alerts/Red_Hat/	Lists recent RH security alerts
Apache	bugs-subscribe@httpd.apache.org cvs-subscribe@httpd.apache.org	Bugs Source change
Snort	N/A	N/A
Tripwire	http://www.tripwire.com	Sign up for the tripwire newsletter
OpenSSH	openssh-unix-announce@mindrot.org.	Updated versions and software
CERT	cert-advisory@cert.org	Cert Advisory mailing list

9.3 General Web Server security

The following links should be visited as frequently as possible, as it gives general information for the web server and it's security:

Linux General Information	http://www.linuxforum.com/ http://www.linuxsecurity.com/ http://lwn.net/security	The following forums frequently have the latest vulnerabilities posted
Apache	http://www.cert.org/advisories	CERT advisory frequently gives information on the latest vulnerabilities discovered Gives information on solutions and latest patches/releases

9.4 Log Review

The snort, tripwire and system logs should be monitored frequently.

It is recommended that you check the logs daily for any unusual activity, even though it is a small system. This is because the web site cannot afford any modifications to any of it's information, as people rely on this accurate information.

This is extremely important for forensics that you check these files regularly, because if you suspect an intrusion, then you will need to tread very carefully to not destroy the evidence.

For example, knowing who rebooted a system, when, and why, are events worth investigating, as it is unusual for a Linux system to reboot itself.

9.5 Backups

You should backup the snort and tripwire logs regularly using a tape backup. It is recommended that you backup daily, however, as this is a small system

weekly should be sufficient.

9.6 Vulnerability Assessment

Regular vulnerability assessments should be scheduled once the system is 'live' to check for any compromises or intrusions. Using Nessus every couple of months and is highly recommended to ensure that there are no security holes in your system.

10. Check Configuration

The system should be checked after installation, configuration and hardening to verify that the system is secure. Performing penetration tests and carrying out a thorough vulnerability assessment can achieve this.

Below are just 5 examples of tests that could be performed to verify that the system is secured against the following vulnerabilities

- Modification of system critical files (eg. Passwords, network configuration)
- Unauthorised access
- Interception and manipulation of messages
- Forged client requests
- Attempts to read the server file system/database
- Attempts to write to the server file system/database

10.1 Run Nessus to check for any security vulnerabilties

Nessus is a scanner that can check for any vulnerabilities that remain on the web server after being hardened.

You should install Nessus on a separate laptop to perform these steps.

Nessus requires both a client and a server.

Setup the following on the laptop that you will be using as the scanner.

1. Download and install *nessusd* and *nessus*

   You can download the latest version of Nessus at: ftp://ftp.gimp.org/pub/gtk/v1.2.

2. Create a **nessusd** account

   The nessusd server has its own users database, each user has their own restrictions, placing extra security precautions as to who has permissions to scan particular devices.

   The utility *nessus-adduser* takes care of the creation of a new account :

```
# nessus-adduser
```

```
Addition of a new nessusd user
------------------------------

Login : webserver
Authentication (pass/cert) [pass] : pass
Password : password

User rules
----------
nessusd has a rules system which allows you to
restrict the hosts
that renaud2 has the right to test. For instance,
you may want
him to be able to scan his own host only.

Please see the nessus-adduser(8) man page for the
rules syntax

Enter the rules for this user, and hit ctrl-D once
you are done :
(the user can have an empty rules set)


Is that ok (y/n) ? [y] y

user added.
```

2. Start nessus daemon:

```
# nessusd –D &
```

3. Start nessues GUI:

```
# nessus &
```

4. Login to the nessusd host
   Nessusd Host: localhost
   Port: 123
   Login: webserver
   Password: ****

5. Target Selection
   Target: 10.1.1.1

6. Select: Start Scan

The scan produced the following security warnings, and I have counteracted these.

- X11 (6000/tcp): The X server doesn't allow any clients to connect to it anyway.
- General/udp: Reviewed firewall rules.
- General/tcp: Reviewed firewall rules.
- General/icmp: Reviewed firewall rules.

10.2 Port Scan

1. Download nmap 3.4.8 from:
http://www.insecure.org/nmap/nmap_download.html
2. Install nmap

```
# rpm -vhU nmap-3.48-1.i386.rpm
```

3. Run nmap. Set it to scan ports 1-1000

The option **–sT** is a TCP connect for unprivileged user

```
# /usr/bin/nmap –sT –P0 –p1-1000 10.1.1.1
Starting nmap v.3.00 (www.insecure.org/nmap/)
Interesting ports on 10.1.1.1)
(The 999 ports scanned but not shown below are in state: closed)
Port          state         service
22/tcp        open          ssh
403/tcp       open          ssl
```

The **–sU** option, is a TCP connect for privileged user (root), and should produce a different list.

```
Starting nmap v.3.00 (www.insecure.org/nmap/)
Interesting ports on 10.1.1.1)
(The 999 ports scanned but not shown below are in state: closed)
Port          state         service
660/udp       open          mac-srvr-admin
```

10.3 Bastille

To ensure the web server was secure, the Bastille harding script was applied. Please see Bastille-Linux web site: http://www.bastille-linux.org/

This script runs through a set of questions, and explains why particular functions should be hardened.

Appendix B for more details and the results.

10.4 Test login

- **Ensure that you cannot login as root at the console**

  A message box should appear, indicating that this account doesn't exist. Once you enter a username other than root, it will allow you into the system. Then you can –su for system admin access.

- **Kernel reboot password**

Ensure you are provided with a prompt for a password if you are attempting to login via a bootup CD or diskette

- **Check for banner**

  A warning banner should appear just before the login screen indicating that this is for authorised use only.

10.5 Verify that only the necessary processes, services and dameons are running

- **Perform a netstat**

```
# netstat –ap
```

This should indicate that only the necessary services are running.

Active Internet connections (servers and established)

Proto	Recv-Q	Send-Q	Local Address	Foreign Address	State
Tcp	0	0	0.0.0.0:443	0.0.0.0:*	Listen
Tcp	0	0	0.0.0.0:22	0.0.0.0:*	Listen
Tcp	0	0	0.0.0.0:25	0.0.0.0:*	Listen

If anything is in a netstat –ap other than port 443 (SSL), 22 (SSH) or 25 (Mail) should be viewed as suspicious, and should be investigated.

Verify that there is a rule to allow port 22 through to local server

```
# iptables -L
```

10.6 Modification of system critical files (eg. Passwords, network configuration)

- **Reset a password**

You should not be able to reset a user password, even as the root user. This is because of the Chattr settings.

```
# passwd apache
```

- **Add an unauthorised user**

You should not be able to modify the /etc/passwd file, because of the chattr setting.

```
# vi /etc/passwd
```

- **Attempt to edit the /etc/hosts**

You should not be able to edit this file, because o the chattr settings

```
# vi /etc/hosts
```

10.7 Unauthorised Access

- **From a unauthorised laptop**

Try to ftp from an unknown host to the web server. You should receive message “Connection refused from remote host” This should be stopped through iptables and tcpwrappers

```
# ftp 10.1.1.1
```

- **From an authorised laptop on an unknown port**

Try to telnet from an known host to the web server on an blocked port such as 21. You should receive message “Connecting To 10.1.1.1...Could not open a connection to host on port 21…connect failed” This should be stopped through iptables and tcpwrappers

```
# telnet 10.1.1. 21
```

- **Use Web server to attempt to rsh or rlogin**

You should receive a message ‘*permission denied’* as all permissions have been set to zero

```
# rsh 10.1.1.2
```

- **Attempt to login into ssh without using a private key**

```
# ssh admin@10.1.1.1
```

You should receive an unauthorised messsage

10.8 Attempt a Forged client requests

Try to execute a command as another user. This should not be able to run because of the changes to SUID and GUID

```
# /usr/sbin/usernetctl
```

Appendix A - Disk Partitioning

Item	Type
Mount point	/boot
File System Type	Ext3
Allowable Drives	Only one drive available, therefore it is automatically selected
Size	75 MB
Fixed Size?	YES
Fill maximum size of (MB)	NO
Fill all available space:	NO
Force to be a primary partitioni	NO
Check for bad blocks	YES

Item	Type
Mount point	/
File System Type	Ext3
Allowable Drives	Only one drive available, therefore it is automatically selected
Size	1000 MB
Fixed Size?	YES
Fill maximum size of (MB)	NO
Fill all available space:	NO
Force to be a primary partitioni	NO
Check for bad blocks	YES

Item	Type
Mount point	/tmp
File System Type	Ext3
Allowable Drives	Only one drive available, therefore it is automatically selected
Size	500 MB
Fixed Size?	YES
Fill maximum size of (MB)	NO
Fill all available space:	NO
Force to be a primary partitioni	NO
Check for bad blocks	YES

Item	Type
Mount point	
File System Type	SWAP
Allowable Drives	Only one drive available, therefore it is automatically selected
Size	1024 MB
Fixed Size?	YES
Fill maximum size of (MB)	NO
Fill all available space:	NO
Force to be a primary partitioni	NO
Check for bad blocks	YES

Item	Type
Mount point	/var
File System Type	Ext3
Allowable Drives	Only one drive available, therefore it is automatically selected
Size	100 MB
Fixed Size?	NO
Fill maximum size of (MB)	YES
Fill all available space:	NO
Force to be a primary partitioni	NO
Check for bad blocks	YES

Appendix B - Hardening Script – Bastille

Bastille Hardening System attempts to "harden" or "tighten" Unix operating systems. This script was run after the manual hardening was performed.

Pre-requisites:

1. **Bastille rpm**

File: Bastille2.1.1-1.0.i386.rpm
Site: http://www.bastille-linux.org
Run: `rpm -ivh Bastille2.1.1-1.0.i386.rpm`

2. **atrpms**

If you are using Red Hat Linux versions 7 or 8
File: atrpms-kickstart-14-1.rh9.at.i386
Site: http://atrpms.physik.fu-berlin.de/install.html
Run: `rpm -ivh atrpms-kickstart-14-1.rh9.at.i386`

3. **perl**

File: Perl-Tk-800.024-.rh9.at.i386.rpm
Site: www.bastille-linux.org/perl-rpm-charl.html
Run: `rpm -ivh Perl-Tk-800.024-.rh9.at.i386.rpm`

1. Copy file **bastille-2.1.1-1.0.i386.rpm** into a temporary directory
2. Run: `rpm -ivh bastille-2.1.1-1.0.i386.rpm`

```
Preparing... ########################################### [100%]
1: Bastille  ########################################### [100%]
```

3. Copy file **perl-Tk-800.022-11.i386**
4. Run: `rpm -ivh perl-Tk-800.024-6.rh9.at.i386.rpm`

Running Bastille 2.1.0

1. > `bastille`
2. type: 'accept' after reading the disclaimer
3. GUI screen should appear
4. Answer all questions about the system

Bastille Questions:

```
1.Would you like to set more restrictive permissions on the administration
utilities
        a. Remove non-root user access to some administrator functions
   5. Would you like to disable SUID status for mount/umount?
        1. Will prevent anyone from mounting drives besides root users
        2. /bin/mount from 4755 to 755
   6. Would you like to disable SUID status for Ping
```

1. Will prevent anyone besides root users to ping and test network connectivity
2. /bin/ping from 4755 to 755
3. /usr/sbin/ping6 755

7. Would you like to disable SUID status for at?
    1. Remove scheduling of an individual task. All tasks can be performed using cron
    2. /usr/bin/at from 4755 to 755
8. Would you like to disable the r-tools
    1. Allow remote connections to other machines
    2. Disables the 'client' side of these tools, so peole cannot use them to connect to other machines
    3. /usr/bin/rcp from 4755 to 0
    4. /usr/bin/rcp to 0
    5. /usr/bin/rlogin to 0
    6. /usr/bin/rsh to 0
    7.
9. Would you like to disable SUID status for usernetctl?
    1. Disable ordinary users from controlling the network interfaces
    2. /usr/sbin/usernetctl from 4750 to 750
10. Would you like to disable SUID status for traceroute
11. /usr/sbin/traceroute from 4755 to 755
12. /usr/sbin/tracerout6 to 755
13. Would you like to disable SUID status for Xfree86?
    1. Disable if this workstation will not be used as a graphical workstation
14. Should Bastille disable clear-text r-protocols that use IP-based authentication?
15. Would you like to enforce password aging?
    1. Force the user to change their password after 180 days (/etc/login.defs)
    2. Add:
       PASS_MAX_DAYS 180
16. Would you like to restrict the use of cron to administrative accounts?
    1. Creates a /etc/cron.allow file of users who may use cron
       root
17. Do you want to set the default umask?
    1. Sets the default permissions for files that you create
18. What umask would you like to set for users on the system?
    1. 027: Only people in your group can read your files, no one can write to them.
    2. NOTE: if your system is converted to trusted mode, this parameter will be overridden by the trusted system default umask, which is 077 (No one on the system can read or write your files)
    3. Append to end of file the following line to /etc/profile:
       umask 027
    4. Appending the following line to /root/.bash_profile:
       umask 027
    5. Appending the following line to /etc/csh.login:
       umask 027
19. Should we disallow root login on tty's 1-6?
    1. Admin must login with an ordinary user account and then user su to become root
    2. /etc/securetty. Replaced line tty1 with: 1
    3. same applied to tty10, tty11, tty2, tty3, tty4, tty5, tty6, tty7, tty8, tty9
    4. eg vc/1
20. Would you like to password-protect the GRUB prompt?
    1. If an attacker has physical access to machine, they could get super-user access through the GRUB command line
    2. /etc/grub.conf. Change permissions from 0600 to 600
21. Enter GRUB Password, please:
22. Would you like to disable CTRL-ALT-DELETE rebooting?

1. Prevent an attacker with access to the machine's keyboard from being able to reboot the machine.

NOT SURE ABOUT THIS ONE: answered no

23. Would you like to password protect single-user mode?
    1. Anyone who can physically interect with system can tell bootloader to bring workstation up in 'single user mode' where they are given root privileges and everyone else is locked out of the system.
    2. Add the following line into /etc/inittab within the section *id:5:initdefault:*

       ~~:S:wait:/sbin/sulogin
24. Would you like to set a default-deny on TCP Wrappers and xinetd?
    1. NOT recommended for most users. Not sure about this one. Answered NO
25. Should Bastille ensure the telnet service does not run on this system
26. Should Bastille ensure inetd's FTP service does not run on this system?
    1. Replace with scp and sftp programs
    2. Removes anonymous ftp users
27. Would you like to display 'Authorized use only' messages at log-in time?
    1. Assists with prosecuting system crackers
    2. Created in the /etc/issue
28. Who is responsible for granting authorization to use this machine? (type in name)
29. Would you like to put limits on system resource usage?
    1. Modify /etc/security/limits.conf to limit the effects of a Denial of Service attack.
    2. Number of allowed core files will be set to zero
    3. Individual users are limited to 150 processes each
    4. /etc/security/limits/conf:

       #prevent core dumps

       ** hard core 0*

       #limit user processes per user to 150

       ** soft nproc 100*

       ** hard nproc 100*
30. Should we restrict console access to a small group of user accounts?
31. Which account should be able to login at console? (root jacqui admin)
    1. Append following line to the end of file /etc/security/access.conf:

       -:ALL EXCEPT root jacqui admin:LOCAL

    2. Append to /etc/pam.d/login:

       account required /lib/security/pam_access.so
    3. Append to /etc/pam.d/xdm

       account required /lib/security/pam_access.so
    4. Append to /etc/pam.d/gdm:

       /account required /ib/security/pam_access.so
32. Would you like to add additional logging?
    1. Will add two additional logging files to the default setup and will also log some status messages to the 7th and 8th virtual terminals
    2. Additional logging files: /var/log/kernel and /var/log/syslog. 'warning and 'error' Select Alt+F7 and ALT+F8
    3. Append to /etc/syslog.conf:

10.8.1.2 Log warning and errors to the new file /var/log/syslog

.warn;.err /var/log/syslog

#Log all kernel messages to the new file /var/log/kernel

kern. var/log/kernel*

#Log all logins to /var/log/loginlog

```
auth.*;user.*;daemon.none  /var/log/loginlog

#Log additional data to the Alt+f7 and alt-f8 screen (Pseudo
TTY 7 and 8)
*info;mail.none;authpriv.none    /dev/tty7
authpriv.*   /dev/tty7
.warn;*.err  /dev/tty7
kern.* /dev/tty7y
mail.* /dev/tty8

*.*    /dev/tty12

Create file /var/syslog
Create file /var/log/kernel
Create file /var/log/loginlog
```

4. Append to /etc/logrotate.d/syslog

```
/var/log/kernel {
      postrotate
            /usr/bin/killall -HUP syslogd
      endscript
}

/var/log/syslog {
      postrotate
            /usr/bin/killall -HUP syslogd
      endscript
}

/var/log/loginlog {
      postrotate
            /usr/bin/killall -HUP syslogd
      endscript
}
```

33. Do you have a remote logging host? NO
34. Would you like to disable apmd?
    1. Used to monitor battery power and is used almost exclusively by notebook/laptop computers: NO
35. Would you like to disable GPM?
    1. Used in console (text) mode to add mouse support to text mode.
    2. Removed link /etc/rc.d/rc2.d/S85gpm
    3. Removed link /etc/rc.d/rc3.d/S85gpm
    4. Removed link /etc/rc.d/rc4.d/S85gpm
    5. Removed link /etc/rc.d/rc5.d/S85gpm
36. Do you want to stop sendmail from running in daemon mode? YES
    1. Removed link to /etc/rc.d/rc2.d/S80sendmail
    2. Removed link to /etc/rc.d/rc3.d/S80sendmail
    3. Removed link to /etc/rc.d/rc4.d/S80sendmail
    4. Removed link to /etc/rc.d/rc5.d/S80sendmail
    5.
37. Would you like to run sendmail via cron to process the queue? NO
38. Would you like to bind the web server to listen only to the localhost?
    1. If you bind the apache web server to the local interface so that it isn't accessible to other machines, it can still server up pages to browsers/web clients on this machine.
    2. Not sure about this one. Answered NO
39. Would you like to bind the web server to a particular interface?
    1. Bind to a specific ip address. YES
    2. File modification to /etc/httpd/conf/httpd.conf
    3. Replace: *Listen *:80* with

```
10.8.1.2.1 Listen 163.187.225.254:80
```

40. Would you like to deactivate the following of symbolic links
    1. Apache runs a user 'nobody' and so it can potentially change/read any world writeable/readable file on the system. If we don't activate this option, a user could potentially allow a web site visitor to view files not in the web page dir.
41. Would you like to deactivate server-side includes?
    1. Way for a web server to execute code to modify web pages
    2. INVESTIGATE FURTHER, have selectede YES
42. Would you like to disable CGI scripts, at least for now?
    1. Disallow users to execute CGI programs
43. Would you like to disable indexes?
    1. Web site visitors can't read the data file even when guess it's name if permissions are changed to non world-readable
44. Would you like to install TMPDIR/TMP scripts
    1. Install scripts that are run when users log in, which safely create suitable temp directories and set the TMPDIR and TMP environment variables. /etc/profile.d script
45. Would you like to run the packet filtering script?
    1. Block certain types of connections to or from your machine
46. Do you need the advanced networking options? NO
47. DNS Servers:
48. Public Interfaces (eth+ ppp+ slip+
49. TCP services to audit: telnet, ftp, imap, pop3, finger, sunrpc, exec, login, linuxconf.ssh
50. UDP services to audit: 31337
51. ICMP services to audit
52. TCP service names or port numbers to allow on public interfaces
53. UDP service names or port numbers to allow on public interfaces
54. Forced passive mode?
55. TCP services to block
56. UDP services to block
57. ICMP allowed types
58. Enable source address verification?
59. Are you finished making changes to your Bastille configuration?

Appendix C - httpd.conf

```
# ==================================================
# Basic settings
# ==================================================
ServerRoot "/usr/local/apache"
PidFile /usr/local/apache/logs/httpd.pid
ScoreBoardFile /usr/local/apache/logs/httpd.scoreboard

# ==================================================
# Performance settings
# ==================================================
Timeout 300
KeepAlive On
MaxKeepAliveRequests 100
KeepAliveTimeout 15
MinSpareServers 5
MaxSpareServers 10
StartServers 5
MaxClients 150
MaxRequestsPerChild 0

# ==================================================
# Apache's modules
# ==================================================
ClearModuleList
AddModule mod_log_config.c
AddModule mod_mime.c
AddModule mod_dir.c
AddModule mod_access.c
AddModule mod_auth.c

# ==================================================
# General settings
# ==================================================
Port 80
User apache
Group apache
ServerAdmin Webmaster@www.ebank.lab
ServerName www.test.site
UseCanonicalName Off
ServerSignature Off
HostnameLookups Off
ServerTokens Prod
<IfModule mod_dir.c>
    DirectoryIndex index.html
</IfModule>
DocumentRoot "/www/ "

# ==================================================
# Access control
# ==================================================
<Directory />
    Options None
    AllowOverride None
    Order deny,allow
    Deny from all
</Directory>
<Directory "/www/test.site">
    Order allow,deny
    Allow from all
```

```
</Directory>

# ==================================================
# MIME encoding
# ==================================================
<IfModule mod_mime.c>
    TypesConfig /usr/local/apache/conf/mime.types
</IfModule>
DefaultType text/plain
<IfModule mod_mime.c>
    AddEncoding x-compress Z
    AddEncoding x-gzip gz tgz
    AddType application/x-tar .tgz
</IfModule>

# ==================================================
# Logs
# ==================================================
LogLevel warn
LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\""
combined
LogFormat "%h %l %u %t \"%r\" %>s %b" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent
ErrorLog /usr/local/apache/logs/error_log

        CustomLog /usr/local/apache/logs/access_log combined
```

Appendix D - Tripwire policy

/etc/tripwire/twcfg.txt

```
ROOT                =/usr/sbin
POLFILE              =/etc/tripwire/tw.pol
DBFILE              =/var/lib/tripwire/$(HOSTNAME).twd
REPORTFILE            =/var/lib/tripwire/report/$(HOSTNAME)-$(DATE).twr
SITEKEYFILE           =/etc/tripwire/site.key
LOCALKEYFILE          =/etc/tripwire/$(HOSTNAME)-local.key
EDITOR              =/bin/vi
LATEPROMPTING         =false
LOOSEDIRECTORYCHECKING =false
MAILNOVIOLATIONS      =true
EMAILREPORTLEVEL      =3
REPORTLEVEL           =3
MAILMETHOD           =SENDMAIL
SYSLOGREPORTING       =false
MAILPROGRAM           =/usr/sbin/sendmail -oi –t
```

References

FAQ.ORG: **Linux Network Administrators Guide: Ch 18. Sendmail.**
URL: http://www.faqs.org/docs/linux_network/x14661.html

Mai, Artur. "**Securing Apache: Step-by-Step"**. *14th* May 2003.

URL: http://www.securityfocus.com/infocus/1694

Roesch, Martin. Green, Chris. "**Snort Users Manual. Snort Release 2.0.0**."

URL: http://www.snort.org/docs/writing_rules/

Chuvakin, Anton. "**Intrusion Detection Response**" 22nd April 2002.

URL: http://www.linuxsecurity.com/feature_stories/ids-active-response.html

Red Hat: Red Hat Linux Reference Guide. "**Chapter 16. "iptables**." 2003

URL: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/ch-iptables.html

Mourani, Gerhard. : "**Securing and Optimizing Red Hat Linux**" 25th March, 2000.
URL:http://packetstormsecurity.nl/papers/unix/Securing-Optimizing-RH-Linux-1_2.pdf

Red Hat: Red Hat Linux Reference Guide. "**An overview of Security-Related Packages**" 2003

URL: http://www.redhat.com/docs/manuals/linux/RHL-7.1-Manual/ref-guide/s1-installation-optionalpackages.html

Harper, Patrick. "**Snort Install Manual. Snort, Apache, PHP, MySQL, and**

Acid Snort: "**Install on RH9.0**" 10th June 2003.

URL: http://www.snort.org/docs/snort_acid_rh9.pdf

Larabee, Rick. "**Developing a Secure and Portable Snort Sensor based on Red Hat 9."** 20th June 2003.

URL: http://www.giac.org/practical/GCUX/Rick_Larabee_GCUX.pdf

Crooke, Adam. "**Securing a Solaris Web Server**." 30th June 2003.

URL: http://www.giac.org/practical/GCUX/Adam_Crooke_GCUX.pdf

Hean, Sean. "**Securing Apache 2.0.44 running under Red Hat 8.0 for the Home Network**. 12th February 2003.

URL: http://www.giac.org/practical/GCUX/Sean_Heare_GCUX.pdf

Grim, Larry. "**Step-by-Step Guide to Securing Red Hat 7.1 Linux**." http://www.giac.org/practical/GCUX/Larry_Grim.GCUX.pdf

Litt, Steve. "**Tripwire Installation and Initial Configuration**". April 2003.

URL:http://www.troubleshooters.com/lpm/200304/200304.htm#_Tripwire_Installation_and_Initial_Configuration

Apache Software Foundation: "**Apache httpd server version 2.0 documentation."** URL: http://httpd.apache.org/docs-2.0/

Red Hat Linux 7.3: The Official Red Hat Linux Reference Guide. **"CH 16: Email"** URL: http://www.redhat.com/docs/manuals/linux/RHL-7.3-Manual/ref-guide/s1-email-sendmail.html

Red Hat Linux 9: Red Hat Linux Security Guide. **"Chapter 5. Server**

Security” URL: http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/security-guide/s1-server-mail.html

CERT/CC, “**Advisory CA-2002-36 Multiple Vulnerabilities in SSH Implementations**”, May 5, 2003, URL: http://www.cert.org/advisories/CA-2002-36.html

Nessus: “**First step : Install Nessus**”. 5th November 2003. URL: http://www.nessus.org/demo/first.html