Global Information Assurance Certification Paper

Build a Secure Web Server Using Red Hat Linux Version 6.2

Step By Step

By Michael Gray

Introduction

This document will serve as a "cookbook" to building a secure web server, using the RedHat Linux 6.2 distribution, intended for use in a corporate intranet behind a firewall. The following design goals are the basis for the decisions being made during the installation and "hardening" of the system:

- Minimize installed software and running services:
  - Limit the number of running services to the absolute minimum to deliver the secure web service and its required components.
- Keep system software current:
  - Minimize the number of installed software packages to decrease the workload and prevent a "forgotten install" from creating a security hole.
  - Update to the latest patched versions of the installed software packages on a routine or as-required basis.
- Prevent inadvertent release of sensitive information:
  - Ensure administrative access to the server is heavily protected, in that all connections are encrypted end-to-end, by using scp2 and ssh2 for system administration and web file creation/administration.
  - Ensure all web authentication username/password exchanges are conducted in a SSL-encrypted session.
  - For web server log statistics, minimize access to "cooked" logs to authorized personnel only.
- Monitor system:
  - Synchronize the system clock to a known internal time source.
  - Send system logs to a central log server. The central log server will monitor the system logs for unusual activity.
  - Run tripwire periodically against the as-built baseline to detect unauthorized system software and configuration changes. Send results via email as soon as they are completed.
- Minimize vulnerability to external attack across the network:
  - Use tcp_wrappers for all possible network services.
  - Run only essential network services.
- Minimize vulnerability to internal attacks from already-logged-on users with normal privileges:
  - Apply more restrictive permissions to files and remove SUID bits on unnecessary programs.
  - Prevent SUID programs and device files from being created or used in user-writeable file systems.

Assumptions

The following assumptions about the physical and network environment have been made:

- The SecureCorp network is an internal network, using private IP addresses, with no connection to the Internet.
- Any custom programs will be compiled on another system and ported, therefore no compilers will be installed on this system.
- The system will be "locked down" upon placement into production, and system maintenance will be conducted only at scheduled intervals. Tripwire will be used to detect unauthorized changes to the system.
- The server will be placed into a corporate server room where clean power, a controlled environment and physical access is strictly controlled. Therefore the traditional steps to protect the BIOS of the system are not required.
- All of the files we will require for installation are available from CD.

Resources

CD's

Two CD's are required to follow these instructions:

- Linux RedHat 6.2 Installation Disk 1 (bootable CD)
- SecureCorp Custom Install CD, containing:
  - /6.2 - Original RedHat 6.2 RPMs from CD (some are removed to save space)
  - /updates/6.2 - updated RPM's from RedHat
  - /updates/autorpm - up-to-date autorpm files
  - /updates/secureweb - up-to-date secure web server files
  - /security - security-related files, including ssh and TrinityOS
  - /custom - custom setup files for this installation

System and Network Information

The following information is required for this installation:

`Hostname:_____ IP address:_____ Subnet Mask: _____`

`Gateway:_____ DNS Servers: _____ _____ _____`

Step 1 – Install Operating System

1.1 - Record hardware configuration: Manufacturer & Model: _____

CPU: _____ RAM: _____

Disk Controller: _____ Disks: _____ _____

_____ _____

NIC: _____ MAC Address: _____

1.2 - Ensure system is disconnected from network until the operating system is installed and hardened.

1.3 - Install RedHat 6.2 from bootable CD. Select default install mode at boot: prompt.

-Select Language: `English`

-Select Keyboard – `Generic 101 Key`

-Select Mouse – `Generic 3 button mouse (PS/2), Emulate 3 Buttons`

-Select Installation type: `Custom`

- disk druid setup:

Add 6 partitions - *The specific partitions described here are based on a 6 GB disk. For larger disks, increase /home and /var accordingly. / and /tmp should be sufficient at their current size.*:

Mt Pnt	Size	Use
/	1000M	= root partition
/home	3000M	= location of web server home directory
/var	1000M	= location of "growing" files, including logs and mail
/tmp	500M	= location of transient files
<Swap>	256M	recommended approx 2 x RAM size *-Assumes 128MB RAM in server*

- Select to format all partitions, and check for bad blocks

- LILO configuration - Linux boot loader

- Create boot disk
- install LILO on /dev/hda
- Partition: /dev/hda1

- Network Configuration
  deselect "Configure using DHCP"
  Enter network parameters.
- Time settings:
  select Time Zone settings: `America/Halifax`
  select "`System clock uses UTC`"
  == system policy to ensure all hardware clocks are set to the same time.
- Set root password
- Create new user accounts
  none at this time
- Authentication Configuration:
  Enable MD5 passwords
  Enable shadow passwords
  Do not enable NIS
- Select package groups to install:
  Only select: ***Networked Workstation, Web Server,*** and ***Utilities***
  Also select: ***Select individual packages***
- Select Individual Packages:
  - **`Applications`** –
    - **`Archiving`** – add ***unzip, zip***
    - **`Editors`** – add ***vim-enhanced, jed, jed-common***
    - **`File`** – remove ***git***
    - **`Internet`** – remove ***fwhois, finger, rsh, talk***
    - **`Publishing`** – remove ***ghostscript, ghostscript-fonts, groff-perl, pnm2ppq, rhs-print-filters, rdate, screen***
    - **`System`** – add ***dialog***
  - **`Documentation`** – remove ***indexhtml***
  - **`System Environment`**
    - **`Base`** – remove ***chkfontpath, yp-tools***
    - **`Daemons`** – remove ***Xfree86-xfs, finger-server, lpr, nfs-utils, pidentd, portmap, rsh-server, rusers, rusers-server, rwall-server, rwho, telnet-server, talk-server, tftp-server, ypbind, ypserv***
      - add ***xntp3***
    - **`Libraries`** – remove ***Xfree86-libs***
  - **`User Interface`**
    - **`X`** – remove ***urw-fonts***
- Create bootdisk
- Complete installation and reboot

Step 2 – Securing Network Configuration

2.1 - Setup TCP Wrappers

Services running from **inetd**, and the **sshd2** services, use the tcp wrapper daemon to control access to the services.

2.1.1 - Set the default access to deny all

Edit **/etc/hosts.deny** so that the only uncommented lines read:

```
ALL: ALL
```

2.1.2 - Allow access to specific services from specific hosts

Edit **/etc/hosts.allow** and add any lines needed to secure other services.

2.1.3 - Check access list syntax

Run `tcpdchk -v` to make sure no errors exist. Ignore the error about the **sshd** process name not existing in **inetd.conf**.

2.2 - Disable Daemon Services

If inetd.conf does not exist, **inetd** should be installed, as many installations of network software will require it to exist:

```
# rpm -Uvh /mnt/cdrom/updates/6.2/inetd-0.*.rpm
```

Edit **/etc/inetd.conf** and comment out ALL services.

2.3 - Disable RunTime Services

Using `chkconfig --list` examine the list of services scheduled to run at levels 2, 3, 4 and 5. The default runlevel is 3, but any service set at runlevel 2 will run first. Disable at boot and turn off:

```
# chkconfig --level 2345 apmd off
# /etc/rc.d/init.d/apmd stop
# chkconfig --level 2345 gpm off
# /etc/rc.d/init.d/gpm stop
# chkconfig --level 2345 netfs off
# /etc/rc.d/init.d/netfs stop
# chkconfig --level 2345 pcmcia off
# chkconfig --level 2345 kudzu off
```

2.4 - Minimize Information Display

2.4.1 - Network Service Messages

Edit **/etc/issue** and **/etc/issue.net** and replace with the following statement:

```
My Secure Corporation
Unauthorized Access Prohibited
```

Edit **/etc/rc.d/rc.local** and remove the lines that create the **/etc/issue.xxx** files on system startup.

2.4.2 - FTP Message

Edit **/etc/ftpaccess** and change the `greeting` statement to read:

```
greeting brief
```

2.5 - Sendmail

On systems that do not act as mail servers, sendmail should not run in daemon mode (with an open network port), but instead should run in "QUEUE" mode, where it wakes up periodically to send any email that may be queued for delivery. To set queue mode, edit **/etc/sysconfig/sendmail** to read:

```
DAEMON=NO
QUEUE=15m
```

Restart the sendmail daemon

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

2.6 - Install SSH

SSH provides strong encryption using RSA public key algorithms and automatic and transparent encryption of network communications for terminal and file copying operations. We will install the SSH version 2 package from SSH.

Install from custom CD and start it.

```
# rpm -i /mnt/cdrom/security/ssh-2.4.0-1.i386.rpm
# /etc/rc.d/init.d/sshd2 start
```

Set up **/etc/hosts.allow** for ssh2 access for the localhost and any system within the company.

```
sshd2: LOCAL .securecorp.com
```

Step 3 – Configure System Utilities

3.1 - Configure Logging

Set up logging of standard facilities, and set up remote logging to the syslog server at `syslog.securecorp.com`. Edit **/etc/syslog.conf**, and **NOTE that the separators between the facility directives in the first column and the syslog files in the second column MUST be <TAB> characters**.

```
*.info;mail.none;authpriv.none;kern.none      /var/log/messages
kern.*                                        /var/log/kernel
authpriv.*                                    /var/log/secure
mail.*                                        /var/log/maillog
authpriv.*;*.warn;mail.none;kern.none         /dev/tty7
kern.*;mail.*                                 /dev/tty8
local7.*                                      /var/log/boot.log
*.warn;*.err                                  @syslog.securecorp.com
authpriv.*;auth.*                             @syslog.securecorp.com
# cd /var/log
# touch kernel
# chmod 700 messages secure kernel *log
# /etc/rc.d/init.d/syslog restart
```

3.2 - SysLog Rotation

RedHat 6.2 uses the **logrotate** tool to maintain log files. We will set up **logrotate** to keep three month's worth of logs, and rotate them every week. Edit **/etc/logrotate.conf** for the following changes:

```
weekly
rotate 13
compress
```

Add the **kernel** syslog file to the logrotate configuration. Add following lines to the end of **/etc/logrotate.d/syslog** :

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

Also, in the same file, comment out the section that rotates the file **/var/log/wtmp**. This file does not grow large, and if it is rotated, you cannot view the last login times of interactive users further back than a week ago.

3.3 - NTP – Network Time Protocol

Setup the server to connect to our three time servers and synchronize the time across the network. Edit **/etc/ntp.conf** to include the following settings:

```
server ntp1.securecorp.com
server ntp2.securecorp.com
server ntp3.securecorp.com
server 127.127.1.0     # local clock
```

```
fudge 127.127.1.0 stratum 10
```

When the xntp daemon starts, it first attempts to manually force the system clock to the current time, using a list of ntp servers stored in **/etc/ntp/step-tickers**. Create or edit **/etc/ntp/step-tickers** with the following lines:

```
ntp1.securecorp.com
ntp2.securecorp.com
ntp3.securecorp.com
```

Start the **xntpd** daemon and check the clock to confirm its settings:

```
# chkconfig xntpd on
# /etc/rc.d/init.d/xntpd start
# ntpq -n -c peers    # may take over 10 minutes to sync
# date
```

3.4 - Tripwire

3.4.1 - Installation

To install **tripwire**, first update the **rpm** package and install the local tripwire policy:

```
# rpm -Uvh /mnt/cdrom/updates/6.2/rpm-3.0.5-9.6x.i386.rpm
# rpm -i /mnt/cdrom/updates/6.2/tripwire-2*.i386.rpm
# cp /mnt/cdrom/custom/twpol.txt /etc/tripwire/twpol.txt
```

3.4.2 - Initial Setup

Now we run the setup scripts and create the database. Ensure you record the site and local pass phrases and save them in a secure place - if you lose them, the tripwire databases become useless and must be regenerated.

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

site keyfile passphrase: _____

local keyfile passphrase: _____

```
# tripwire --init
# tripwire --check
```

3.4.3 - Policy Modification

You may ignore any **file not found** errors during this process, as the tripwire policy is configured for a fully-loaded installation. Some of the files included in the tripwire policy are created later or are only used for other applications. To remove the **file not found** errors, edit **/etc/tripwire/twpol.txt** to comment out or remove the lines containing the reference to the file not found, and save the new **twpol.txt** file. To install the new policy file, rerun the steps in 3.4.2 above.

When the installation is complete, you will want to accept the changes that have been made since this database was created above. See step **8.1** for details.

Step 4 – Improve File System Security

4.1 - Modify Default File Permissions

Many of the default file permissions used when Linux is installed are too permissive. A script based on a portion of the TrinityOS configuration script will be used to correct the permissions.

```
# /mnt/cdrom/custom/apply-permissions.sh
```

A copy of the script appears as Appendix 1.

4.2 - Do Not Allow SUID and Devices in User File Systems

Attackers that gain access to the system as an unprivileged user will attempt to exploit known vulnerabilities to elevate their privileges to **root**. Many exploits require the creation of SUID programs or of device files in file areas that are user-writeable. To prevent these exploits, modify **/etc/fstab** and change the lines that mount **/tmp** and **/home** to:

```
/dev/hdaX          /home        ext2      nodev,nosuid      1 2
/dev/hdaX          /tmp         ext2      nodev,nosuid      1 2
```

4.3 - Remove SUID where necessary

The following commands will remove the SUID flag on the files which do not require it in our configuration. These commands are set SUID to allow unprivileged users to execute privileged operations. Since on our server most system configuration operations will be conducted as root, the following are unnecessary:

```
# chmod a-s /sbin/netreport  /bin/mount /bin/umount
# chmod a-s /usr/bin/chage /usr/bin/wall /usr/bin/at
# chmod a-s /usr/bin/man /usr/bin/lockfile
# chmod a-s /usr/bin/procmail /usr/bin/chfn
# chmod a-s /usr/bin/chsh /usr/bin/newgrp
# chmod a-s /usr/bin/write /usr/sbin/usernetctl
```

Step 5 - Install Additional and Updated Packages

5.1 - Install and Configure AutoRPM

AutoRPM is a utility that can keep the RPM's of a system consistent with an FTP site or local directory, or maintain a mirror of an RPM site. We will first use AutoRPM to automatically compare and update the RPM's on our server against our internal FTP site which contains a mirror of the contents of **updates.redhat.com**. Then we will change the configuration so that once a day AutoRPM will check for updated RPM's and report via email that an update is available that must be installed interactively.

5.1.1 - Install AutoRPM

Install the packages:

```
# rpm -Uvh /mnt/cdrom/updates/autorpm/perl-libnet-1.0605-2.noarch.rpm
# rpm -Uvh /mnt/cdrom/updates/autorpm/autorpm-1.9.8.4-2.rpm
```

Explicitly set the RedHat version number in **/etc/autorpm.d/autorpm.conf** using statement

`Set_Var("RHVersion","6.2");` (Note that the sample file incorrectly uses Eval_Var)

Replace the contents of **/etc/autorpm.d/pools/redhat-updates** with the address of our internal server which maintains an anonymous ftp site containing the latest RedHat updated RPM's:

`ftp://updates.securecorp.com/pub/updates/${RHVersion}`

Replace the contents of **/etc/autorpm.d/pools/autorpm-updates** with the following address:

`ftp://updates.securecorp.com/pub/updates/autorpm`

Edit the file **/etc/autorpm.d/redhat-update.conf** and change the `Install (Interactive)` statement to `Install (No)` in the `action (new)` section. Set the PGP options.

```
action (updated) {
   Install (Interactive);
   PGP_Require (Yes);
    ...some lines...
}
action (new) {
   # some comments
   Install (No);
}
```

5.1.2 - Install GNU PGP and Keys

Install and set up GNU PGP with the Red Hat public key by copying the public key ring from our custom installation CD:

```
# rpm -Uvh /mnt/cdrom/updates/6.2/gnupg-1*.i386.rpm
# gpg      # to setup GNU pgp files - press CTRL-D to exit
```

```
# cp /mnt/cdrom/custom/pubring.gpg /root/.gnupg/pubring.gpg
```

Test it by confirming the signature of the rpm files

```
# rpm --checksig /mnt/cdrom/updates/6.2/*.rpm
```

5.2 - Use AutoRPM to Update RedHat RPM's

Mount the updates CD and use AutoRPM to compare the installed RPM's with the ones on the CD.

```
# autorpm --dir /mnt/cdrom/updates/6.2
```

After checking the CD, you will immediately be placed in the interactive install mode. Scroll down the list and for every item that lists `(Do Not Upgrade)`, highlight it with the arrow keys, press <RETURN> and select the option to `Toggle-Install`. To complete the installation of the selected items, move to the bottom of the list and select `Apply`, then Exit. To clear the list of new RPM's that we are not interested in,

```
# rm /var/spool/autorpm/interactive.queue
```

Note that by default, when AutoRPM runs the results are mailed to **root**. To change the email destination, edit **/etc/autorpm.d/autorun.conf** and change the underlined portion of the following sample line to the email address of the administrators:

```
Set_Var ("ReportDest", "AdminGrp@securecorp.com");
```

The AutoRPM will check for updates daily, driven by the **/etc/cron.daily/autorpm** script.

5.3 - Use RPM to Update RPM's

Another way upgrade individual RPM's is with the *Freshen* option of the `rpm` command. With all the latest updates in `/mnt/cdrom/updates/6.2`, you can "freshen" only the already installed packages with the command:

```
# rpm -F /mnt/cdrom/updates/6.2/*.rpm
```

5.4 - Upgrade the Kernel

The kernel in the default RedHat 6.2 installation has several known bugs and security problems. It is important that the non-kernel RPM's have been updated using the previous steps before upgrading the kernel. Follow the procedures outlined below to upgrade to the latest version on the updates CD. Note that depending on the CPU type of your computer, you will want to use the i386 or i586 (or smb if multiprocessor) kernels.

First make a rescue floppy. Place an empty floppy in the drive, and

```
# mkbootdisk --device /dev/fd0 2.2.14-5.0
```

Now install the kernel images.

```
# cd /mnt/cdrom/updates
# rpm -ivh kernel-2*.i686.rpm kernel-ibcs-2*.rpm
# rpm -ivh --force kernel-pcmcia-cs-2*.rpm
```

If you boot from SCSI or RAID disks, you need to rebuild the init boot image.

```
# mkinitrd /boot/initrd-2.2.17-14.img 2.2.17-14
```

Edit /etc/lilo.conf and add this section

```
image=/boot/vmlinuz-2.2.17-14
     label=linux
     read-only
     root=/dev/hda1
```

Also in **/etc/lilo/conf**, change the label of the existing section to `label=linux.old`. Now install the new **lilo**.

```
# lilo -v
```

Reboot the system and press `<TAB>` at the `LILO`: prompt. You should be presented with

```
linux      linux.old
```

Press RETURN to continue booting with the new kernel. Remember to build a new rescue floppy now with

```
# mkbootdisk --device /dev/fd0 2.2.17-14
```

Step 6 – Install and Secure the Applications

6.1 - Secure Web Server

6.1.1 - Installation

Install latest secureweb server from CD:

```
rpm -Uvh /mnt/cdrom/updates/secureweb-3.2.2-4.i386.rpm
```

For more security, the web server runs as an unprivileged user. Create user **www** and group **www** :

```
groupadd -g 80 www
useradd -u 80 -g 80 -d /home/httpd -c "httpd user" -s /bin/false www
```

Modify the settings in **/etc/httpd/conf/httpsd.conf** :

```
...
User www
Group www
...
ServerAdmin admingrp@securecorp.com
...
```

6.1.2 - SSL Key Creation and Setup

To create the keys, execute the following:

```
# cd /etc/httpd/conf
# make genkey
```

record PEM pass phrase:_____

```
# make certreq
  Country Name:    CA
  State or Province:  Nova Scotia
  Locality...:  Halifax
  Company ...:  SecureCorp
  Department...:  SecureDiv
  Server Host Name:  hostname.securecorp.com
  Admin EMail:  admingrp@securecorp.com
  challenge password: _____
  optional company name:  leave blank
```

Copy the file **/etc/httpd/conf/ssl.csr/server.csr** to a floppy and send it to the admin staff for signing. They will return it with a new file called **server.crt**, which you must place in **/etc/httpd/conf/ssl.crt/server.crt**

6.1.3 - Start the Server

The server will start only if both key files are installed and correct. Note that you will be prompted for your PEM pass phrase on every restart or reboot of the system, and that the server will not complete the boot process until the passphrase is entered correctly.

```
/etc/rc.d/init.d/httpsd start
```

6.1.4 - Removing PEM Passphrase

To remove the requirement to enter the PEM passphrase on every reboot of the system or every time **httpsd** is restarted, follow these steps. NOTE that this is insecure, and if anyone gets this key they can impersonate the secure web server on the net. However, in our environment it is more important that the server reboot quickly to restore services in case of a failure.

```
# rpm -ivh /mnt/cdrom/updates/6.2/openssl-0*i386.rpm  # install openssl
# cd /etc/httpd/conf/ssl.key
# mv server.key server.key-secure
```

```
# openssl rsa -in server.key-secure -out server.key
# chmod go-rwx * .      # remove all file permissions for 'group' and 'other'
```

6.1.5 - Protect Transmission of Username and Password

By default, the web server will allow any host to retrieve files. To restrict access in a specific directory, create a **.htaccess** file similar to the one for **webalizer** in section 6.2.2. To force the use of SSL to protect the transmission of the username and password, ensure the following line is included in the file:

```
SSLRequireSSL
```

6.2 - Web Statistics Reporting

6.2.1 - Install Webalizer

Install the web statistics package **webalyzer**. The default installation creates a config file at **/etc/webalizer.conf**, and writes a daily cron entry at **/etc/cron.daily/webalizer.cron**. Output from the **webalizer** script is created in the **/home/httpd/html/usage** directory.

```
# rpm -Uvh /mnt/cdrom/6.2/webalizer*.rpm
```

6.2.2 - Protect Access to Webalizer Reports

Limit access to the web statistics to authorized users, by adding **/home/httpd/html/usage/.htaccess** containing:

```
AuthName "Secure Area"
AuthType Basic
AuthUserFile /etc/httpd/conf/htpasswd
require valid-user
SSLRequireSSL
```

Create users for the site:

```
# htpasswd -c /etc/httpd/conf/htpasswd user1
# htpasswd /etc/httpd/conf/htpasswd user2
...
```

Remove world access to the config files for the web server and provide access to **htpasswd** for the **www** user:

```
# chmod -R o-rwx /etc/httpd/conf/* /etc/httpd/conf
# chgrp www /etc/httpd/conf/htpasswd
```

6.3 - Anonymous FTP

Anonymous ftp will be set up for read-only, accessible to the local network only. No ability to write any files will be allowed. The **anonftp** package will set up a "chroot"-ed, safe environment for anonymous ftp in **/home/ftp**:

```
rpm -Uvh anonftp-*.rpm
```

Edit **/etc/hosts.allow** to include the **in.ftpd** service:

```
in.ftpd: .securecorp.com 192.168.
```

Step 7 – Testing

7.1 - Running Processes

Check list of running processes:

```
# ps axf
```

There should be only the following processes running (excluding your own login and other login shells, and the getty processes):

```
  PID TTY      STAT   TIME COMMAND
    1 ?        S      0:14 init [3]
    2 ?        SW     0:10 [kflushd]
    3 ?        SW     0:03 [kupdate]
    4 ?        SW     0:00 [kpiod]
```

```
    5 ?        SW     0:02 [kswapd]
 1239 ?        S      0:02 syslogd -m 0
 1248 ?        S      0:00 klogd
 1262 ?        S      0:00 crond
 1276 ?        S      0:00 inetd
 1307 ?        SL     0:38 xntpd -A
 1465 ?        S      0:00 [sendmail]
 1622 tty1     S      0:00 login -- root
 1666 tty1     S      0:00  \_ -bash
 1623 tty2     S      0:00 /sbin/mingetty tty2
 1624 tty3     S      0:00 /sbin/mingetty tty3
 1625 tty4     S      0:00 /sbin/mingetty tty4
 1626 tty5     S      0:00 /sbin/mingetty tty5
 1665 tty6     S      0:00 /sbin/mingetty tty6
```

7.2 - Open Network Connections

Install the **lsof** (list open files) command:

```
# rpm -ivh /mnt/cdrom/6.2/lsof-*.rpm
```

Check the open ports:

```
# lsof -ni
```

The list should include only the minimal daemons configured earlier:

```
COMMAND   PID USER   FD   TYPE DEVICE SIZE NODE NAME
xntpd    1307 root    4u  IPv4   1490       UDP *:ntp
xntpd    1307 root    5u  IPv4   1491       UDP 127.0.0.1:ntp
xntpd    1307 root    9u  IPv4   1493       UDP 192.168.1.5:ntp
sshd2    1517 root    3u  IPv4   1697       TCP *:ssh (LISTEN)
```

Step 8 – Archival Backup

8.1 - Run Tripwire on Final Configuration

To build a database that reflects the final configuration, you will need to update the existing tripwire database.

```
# tripwire --check          #  builds a report file
# ls -la /var/lib/tripwire/reports
 ...  find the filename of the latest report file
     format is hostname-date-timestamp.twr
# tripwire --update -r /var/lib/tripwire/reports/latest-filename
```

You will be placed in a **vi** edit session with the report file to confirm the authorized changes. All changed files will have a checkbox beside it with its default to accept the change. If you wish to accept all changes, simply exit the editor with the **ZZ** command. You will need to enter the appropriate pass phrases when prompted.

8.2 - Run Archival Backup

Backup using generic Linux utilities:

Uses tape drive. Test by erasing a tape, then running a full backup. Use the Compare feature to test the backup. Do a trial restore upon completion of the backup.

```
# mt -f /dev/st0 erase
# mt -f /dev/st0 rewind
# dump -0au -f /dev/nst0 /          # Note use of non-rewinding tape device
# dump -0au -f /dev/nst0 /var
# dump -0au -f /dev/nst0 /home
# dump -0au -f /dev/nst0 /tmp
# mt -f /dev/st0 rewind             # rewind tape to begin verify pass
# restore -Cv -f /dev/nst0          # checks each partition in order
```

```
# restore -Cv -f /dev/nst0
# restore -Cv -f /dev/nst0
# restore -Cv -f /dev/nst0
# mt -f /dev/st0 rewind
# mv /tmp/install.log /tmp/install.log.bak
# restore -xv -f /dev/nst0 /tmp/install.log
 ... continue running above command until correct dump set is found
 ... enter "1" for volume number when prompted
# mt -f /dev/st0 rewoffl            # eject tape
```

Now confirm that **/tmp/install.log** has been correctly restored.

Do Trial Disaster Recovery from Tape

Use spare disk space to create a new partition, or install a new drive. Then format the partition and restore the root filesystem from tape. At this point, the root filesystem will only require less than 500MB.

```
# mke2fs /dev/hdb1            # assume a new disk with a 1GB partition
# mkdir /mnt/test
# mount /dev/hdb1 /mnt/test
# cd /mnt/test
# mt -f /dev/st0 rewind
# restore -rv -f /dev/st0
# mt -f /dev/st0 rewoffl
```

Check the contents of **/mnt/test**. It should contain an exact copy of the / partition.

8.3 - Disk Image

Install a second hard drive of equal or larger size as your boot disk. Assuming your boot disk is **/dev/hda** and your blank disk is **/dev/hdb**, an identical copy can be made using the following commands:

```
# init 1   # go to single user mode
# dd if=dev/hda of=/dev/hdb bs=1k
# init 3   # return to multiuser mode
# fdisk -l /dev/hdb
# fsck /dev/hdb1
... then run fsck for all other partitions as shown above...
```

Step 9 – Final Signoff

9.1 - Complete Documentation

Document the partition scheme for every disk:

```
# fdisk -l /dev/hda >> /etc/fdisk-l.log
```

or /dev/sda for SCSI

or /dev/ida/c0d0 for hardware RAID

Document the RPM's installed and list the files changed after the initial installation:

```
# rpm -q -a > /etc/rpm/rpms-installed.log
# rpm -Va > /etc/rpm/rpmdiff.log
# chmod 600 /etc/rpm/*
# chmod 700 /etc/rpm
```

9.2 - Store Initial Backup and Drive Image

Save a copy of the partition scheme along with the initial backup tape and any other documentation, including this document, in a safe place off-site.

Appendix 1 - apply_permissions.sh

This script is extracted from the TrinityOS-security script. It is designed to change the default file permissions on a default RedHat installation to a more secure state.

```
echo -e "\n\nSection 8.11"
echo -e "\nAutoFix: Change system permissions for only root and admin groups"
echo "         If you see errors, that probably means that the package "
echo "         wasn't installed."
echo -e "\n** All current daemon settings will be saved in $BACKUP **"
echo      "-----------------------------------------------------------------"
perl wait-for-key
clear
ls -laR /bin > $BACKUP/bin/bin.lst-`date +'%m%d%y'`-`date +%S`
echo -e "\nFiles in /bin "
echo -e   "-------------"
echo "Bru is a commercial backup program but some Linux distros come with it"
chmod 750 /bin/bru
chmod 750 /bin/linuxconf
chmod 750 /bin/mount
chmod 750 /bin/mt
chmod 750 /bin/rpm
chmod 750 /bin/setserial
chmod 750 /bin/umount
echo -e "\n/sbin"
echo -e "-----"
ls -laR /sbin > $BACKUP/sbin/sbin.lst-`date +'%m%d%y'`-`date +%S`
chmod 750 /sbin/accton
chmod 750 /sbin/badblocks
chmod 750 /sbin/ctrlaltdel
chmod 750 /sbin/chkconfig
chmod 750 /sbin/chkraid
chmod 750 /sbin/debugfs
chmod 750 /sbin/depmod
chmod 750 /sbin/dhcpcd
chmod 750 /sbin/dump*
chmod 750 /sbin/fdisk
chmod 750 /sbin/fsck*
chmod 750 /sbin/ftl*
chmod 750 /sbin/getty
chmod 750 /sbin/halt
chmod 750 /sbin/hdparm
chmod 750 /sbin/hwclock
chmod 750 /sbin/ide_info
chmod 750 /sbin/if*
chmod 750 /sbin/init
chmod 750 /sbin/insmod
echo "IPFWADM is only installed for v2.0 kernels"
chmod 750 /sbin/ipfwadm
chmod 750 /sbin/ipx*
chmod 750 /sbin/isapnp
#For 2.0.x kernels
chmod 750 /sbin/kerneld
chmod 750 /sbin/killall*
echo -e "This is the new location for klogd."
echo -e "Please disregard any errors if this doesn't work.\n"
chmod 750 /sbin/klogd
chmod 750 /sbin/lilo
chmod 750 /sbin/mgetty
chmod 750 /sbin/mingetty
chmod 750 /sbin/mk*
chmod 750 /sbin/mod*
chmod 750 /sbin/netreport
chmod 750 /sbin/pam*
chmod 750 /sbin/pcinitrd
chmod 750 /sbin/pnpdump
chmod 750 /sbin/portmap
```

```
chmod 750 /sbin/quotaon
chmod 750 /sbin/raidadd
chmod 750 /sbin/restore
chmod 750 /sbin/runlevel
chmod 750 /sbin/stinit
echo -e "This is the NEW location for sys/klogd."
echo -e "Please disregard any errors if this doesn't work.\n"
chmod 750 /sbin/syslogd
chmod 750 /sbin/swapon
chmod 750 /sbin/tune2fs
chmod 750 /sbin/uugetty
chmod 750 /sbin/vgetty
echo -e "\n/usr/bin"
echo -e   "--------"
ls -laR /usr/bin > $BACKUP/usr/bin/usr-bin.lst-`date +'%m%d%y'`-`date +%S`
chmod 750 /usr/bin/control-panel
chmod 750 /usr/bin/comanche
chmod 750 /usr/bin/eject
chmod 750 /usr/bin/glint
chmod 750 /usr/bin/gnome*
chmod 750 /usr/bin/gpasswd
chmod 750 /usr/bin/ipx*
chmod 750 /usr/bin/kernelcfg
chmod 755 /usr/bin/lp*
chmod 4755 /usr/bin/lpr
chmod 750 /usr/bin/mformat
chmod 750 /usr/bin/minicom
chmod 750 /usr/bin/mtools
chmod 750 /usr/bin/netcfg
chmod 750 /usr/bin/rusers
chmod 750 /usr/bin/rwall
chmod 750 /usr/bin/uucp
echo -e "\n/usr/sbin"
echo -e   "---------"
ls -laR /usr/sbin > $BACKUP/usr/sbin/usr-sbin.lst-`date +'%m%d%y'`-`date +%S`
chmod 750 /usr/sbin/am*
chmod 750 /usr/sbin/at*
chmod 750 /usr/sbin/automount
chmod 750 /usr/sbin/bootp*
chmod 750 /usr/sbin/crond
chmod 750 /usr/sbin/dhc*
chmod 750 /usr/sbin/dip
chmod 750 /usr/sbin/dump*
chmod 750 /usr/sbin/edquota
chmod 750 /usr/sbin/exportfs
chmod 750 /usr/sbin/fixmount
chmod 750 /usr/sbin/ftpshut
chmod 750 /usr/sbin/gated
chmod 750 /usr/sbin/group*
chmod 750 /usr/sbin/grp*
chmod 750 /usr/sbin/imapd
chmod 750 /usr/sbin/in.*
chmod 750 /usr/sbin/inetd
chmod 750 /usr/sbin/ipop*
echo "This is the old location for klogd."
echo -e "Please disregard any errors if this doesn't work.\n"
chmod 750 /usr/sbin/klogd
chmod 750 /usr/sbin/logrotate
chmod 750 /usr/sbin/lp*
chmod 755 /usr/sbin/lsof
chmod 750 /usr/sbin/makemap
chmod 750 /usr/sbin/mk-amd-map
chmod 750 /usr/sbin/mouseconfig
chmod 750 /usr/sbin/named*
chmod 750 /usr/sbin/nmbd
chmod 750 /usr/sbin/newusers
chmod 750 /usr/sbin/ntp*
```

```
chmod 750 /usr/sbin/ntsysv
chmod 750 /usr/sbin/pppd
chmod 750 /usr/sbin/pnpprobe
chmod 750 /usr/sbin/pw*
chmod 750 /usr/sbin/quota*
chmod 750 /usr/sbin/rdev
chmod 750 /usr/sbin/rdist
chmod 750 /usr/sbin/repquota
chmod 750 /usr/sbin/rhbackup
chmod 750 /usr/sbin/rotatelogs
chmod 750 /usr/sbin/rpc*
chmod 750 /usr/sbin/rwhod
chmod 750 /usr/sbin/samba
chmod 750 /usr/sbin/setup
chmod 750 /usr/sbin/showmount
chmod 750 /usr/sbin/smb*
chmod 750 /usr/sbin/sndconfig
chmod 750 /usr/sbin/snmp*
chmod 750 /usr/sbin/squid
echo "This is the OLD location for sysklogd."
echo -e "Please disregard any errors if this doesn't work.\n"
chmod 750 /usr/sbin/syslogd
chmod 750 /usr/sbin/taper
chmod 750 /usr/sbin/tcpd*
chmod 750 /usr/sbin/time*
chmod 750 /usr/sbin/tmpwatch
chmod 750 /usr/sbin/tunelp
chmod 750 /usr/sbin/user*
chmod 750 /usr/sbin/uu*
chmod 750 /usr/sbin/vi*
chmod 750 /usr/sbin/wire-test
chmod 750 /usr/sbin/xntp*
perl wait-for-key
clear
```

Appendix 2 - References

[1] Ranch, David A. "TrinityOS: A Guide to Configuring Your Linux Server for Performance, Security, and Managability", 10 February 2001, http://www.ecst.csuchico.edu/~dranch/LINUX/index-linux.html (10 February 2001)

[2] Campione, Jeff. "Solaris 8 Installation Checklist", http://www.sans.org/y2k/practical/Jeff_Campione_GCUX.htm (10 February 2001)

[3] Wang, Rui. "Secure Linux RedHat 7 Anonymous Upload Server Step by Step", http://www.sans.org/y2k/practical/Rui_Wang_GCUX.html (10 February 2001)

[4] Steves, Kevin. "Building a Bastion Host Using HP-UX 11", 21 July 2000, http://people.hp.se/stevesk/bastion11.html (10 February 2001)

[5] "Bastille-Linux Scripts to Secure Linux", version 1.1.1, SANS Institute, http://www.sans.org/newlook/projects/bastille_linux.htm (10 February 2001)

[6] Brotzman, Lee E. and Ranch, David A.. Securing Linux Step-By-Step, v.1.0, SANS Institute, 2000.

[7] Anonymous, Maximum Linux Security, Indianapolis: SAMS Publishing, 2000.