THE CENTER FOR INTERNET SECURITY SM

Linux Benchmark v1.1.0

(Red Hat Linux 7.0 and later)

Linux Benchmark v1.1.0

July 29, 2003

Agreed Terms of Use

Background.

CIS provides benchmarks, scoring tools, software, data, information, suggestions, ideas, and other services and materials from the CIS website or elsewhere ("**Products**") as a public service to Internet users worldwide. Recommendations contained in the Products ("**Recommendations**") result from a consensus-building process that involves many security experts and are generally generic in nature. The Recommendations are intended to provide helpful information to organizations attempting to evaluate or improve the security of their networks, systems and devices. Proper use of the Recommendations requires careful analysis and adaptation to specific user requirements. The Recommendations are not in any way intended to be a "quick fix" for anyone's information security needs.

No representations, warranties and covenants.

CIS makes no representations, warranties or covenants whatsoever as to (i) the positive or negative effect of the Products or the Recommendations on the operation or the security of any particular network, computer system, network device, software, hardware, or any component of any of the foregoing or (ii) the accuracy, reliability, timeliness or completeness of any Product or Recommendation. CIS is providing the Products and the Recommendations "as is" and "as available" without representations, warranties or covenants of any kind.

User agreements.

By using the Products and/or the Recommendations, I and/or my organization ("**we**") agree and acknowledge that:

1. No network, system, device, hardware, software or component can be made fully secure;
2. We are using the Products and the Recommendations solely at our own risk;
3. We are not compensating CIS to assume any liabilities associated with our use of the Products or the Recommendations, even risks that result from CIS's negligence or failure to perform;
4. We have the sole responsibility to evaluate the risks and benefits of the Products and Recommendations to us and to adapt the Products and the Recommendations to our particular circumstances and requirements;
5. Neither CIS, nor any CIS Party (defined below) has any responsibility to make any corrections, updates, upgrades or bug fixes or to notify us if it chooses at its sole option to do so; and

6. Neither CIS nor any CIS Party has or will have any liability to us whatsoever (whether based in contract, tort, strict liability or otherwise) for any direct, indirect, incidental, consequential, or special damages (including without limitation, loss of profits, loss of sales, loss of or damage to reputation, loss of customers, loss of software, data, information or emails, loss of privacy, loss of use of any computer or other equipment, business interruption, wasted management or other staff resources or claims of any kind against us from third parties) arising out of or in any way connected with our use of or our inability to use any of the Products or Recommendations (even if CIS has been advised of the possibility of such damages), including without limitation any liability associated with infringement of intellectual property, defects, bugs, errors, omissions, viruses, worms, backdoors, Trojan horses or other harmful items.

Grant of limited rights.

CIS hereby grants each user the following rights, but only so long as the user complies with all of the terms of these Agreed Terms of Use:

1. Except to the extent that we may have received additional authorization pursuant to a written agreement with CIS, each user may download, install and use each of the Products on a single computer;

2. Each user may print one or more copies of any Product or any component of a Product that is in a .txt, .pdf, .doc, .mcw, or .rtf format, provided that all such copies are printed in full and are kept intact, including without limitation the text of this Agreed Terms of Use in its entirety.

Retention of intellectual property rights; limitations on distribution.

The Products are protected by copyright and other intellectual property laws and by international treaties. We acknowledge and agree that we are not acquiring title to any intellectual property rights in the Products and that full title and all ownership rights to the Products will remain the exclusive property of CIS or CIS Parties. CIS reserves all rights not expressly granted to users in the preceding section entitled "Grant of limited rights."

Subject to the paragraph entitled "Special Rules" (which includes a waiver, granted to some classes of CIS Members, of certain limitations in this paragraph), and except as we may have otherwise agreed in a written agreement with CIS, we agree that we will not (i) decompile, disassemble, reverse engineer, or otherwise attempt to derive the source code for any software Product that is not already in the form of source code; (ii) distribute, redistribute, encumber, sell, rent, lease, lend, sublicense, or otherwise transfer or exploit rights to any Product or any component of a Product; (iii) post any Product or any component of a Product on any website, bulletin board, ftp server, newsgroup, or other similar mechanism or device, without regard to whether such mechanism or device is internal or external, (iv) remove or alter trademark, logo, copyright or other proprietary notices, legends, symbols or labels in any Product or any component of a Product; (v) remove these Agreed Terms of Use from, or alter these Agreed Terms of Use as they appear in, any Product or any component of a Product; (vi) use any Product or any component of a Product with any derivative works based directly on a Product or any component of a Product; (vii) use any Product or any component of a Product with other products or applications that are directly and specifically dependent on such Product or any component for any part of their functionality, or (viii) represent or claim a particular level of compliance with a CIS Benchmark, scoring tool or other Product. We will not facilitate or otherwise aid other individuals or entities in any of the activities listed in this paragraph.

We hereby agree to indemnify, defend and hold CIS and all of its officers, directors, members, contributors, employees, authors, developers, agents, affiliates, licensors, information and service providers, software suppliers, hardware suppliers, and all other persons who aided CIS in the creation, development or maintenance of the Products or Recommendations ("**CIS Parties**") harmless from and against any and all liability, losses, costs and expenses (including attorneys' fees and court costs) incurred by CIS or any CIS Party in connection with any claim arising out of any violation by us of the preceding paragraph, including without limitation CIS's right, at our expense, to assume the exclusive defense and control of any matter subject to this indemnification, and in such case, we agree to cooperate with CIS in its defense of such claim.

We further agree that all CIS Parties are third-party beneficiaries of our undertakings in these Agreed Terms of Use.

Special rules.

The distribution of the NSA Security Recommendations is subject to the terms of the NSA Legal Notice and the terms contained in the NSA Security Recommendations themselves (http://nsa2.www.conxion.com/cisco/notice.htm).

CIS has created and will from time to time create special rules for its members and for other persons and organizations with which CIS has a written contractual relationship. Those special rules will override and supersede these Agreed Terms of Use with respect to the users who are covered by the special rules.

CIS hereby grants each CIS Security Consulting or Software Vendor Member and each CIS Organizational User Member, but only so long as such Member remains in good standing with CIS and complies with all of the terms of these Agreed Terms of Use, the right to distribute the Products and Recommendations within such Member's own organization, whether by manual or electronic means. Each such Member acknowledges and agrees that the foregoing grant is subject to the terms of such Member's membership arrangement with CIS and may, therefore, be modified or terminated by CIS at any time.

Choice of law; jurisdiction; venue.

We acknowledge and agree that these Agreed Terms of Use will be governed by and construed in accordance with the laws of the State of Maryland, that any action at law or in equity arising out of or relating to these Agreed Terms of Use shall be filed only in the courts located in the State of Maryland, that we hereby consent and submit to the personal jurisdiction of such courts for the purposes of litigating any such action. If any of these Agreed Terms of Use shall be determined to be unlawful, void, or for any reason unenforceable, then such terms shall be deemed severable and shall not affect the validity and enforceability of any remaining provisions.

We acknowledge and agree that we have read these Agreed Terms of Use in their entirety, understand them and agree to be bound by them in all respects.

Agreed Terms of Use - Version 1.2 (March 20, 2003)

CIS Linux Benchmark

A Word about Shaded Items

Desktop systems typically have different security expectations than server-class systems. In an effort to facilitate use of this benchmark on these different classes of machines, shaded text has been used to indicate questions and/or actions that are typically not applicable in the desktop environment. These shaded items may be skipped on desktop platforms.

Root Shell Environment Assumed

The actions listed in this document are written with the assumption that they will be executed by the `root` user running the `/bin/bash` shell and without `noclobber` set.

Executing Actions

The actions listed in this document are written with the assumption that they will be executed in the order presented here. Some actions may need to be modified if the order is changed. Actions are written so that they may be copied directly from this document into a root shell window with a "cut-and-paste" operation.

You may find that many of the "chkconfig" actions, which activate or deactivate services, produce the message "error reading information on service <service>: No such file or directory." These messages are quite normal and should not cause alarm -- they simply indicate that the program being referenced was not installed on your machine. As Red Hat Linux installs allow a great deal of flexibility in what software you choose to install, these messages are unavoidable.

Reboot Required

Rebooting the system is required after completing all of the actions below in order to complete the re-configuration of the system. In many cases, the changes made in the steps below will not take effect until this reboot is performed.

Backup Key Files

Before performing the steps of this benchmark it is a good idea to make backup copies of critical configuration files that may get modified by various benchmark items:

```
for file in /etc/inetd.conf /etc/hosts.equiv \
/etc/ftpusers /etc/passwd /etc/shadow /etc/hosts.allow \
/etc/hosts.deny /etc/proftpd.conf \
/etc/rc.d/init.d/functions /etc/inittab \
/etc/sysconfig/sendmail /etc/security/limits.conf \
/etc/exports /etc/sysctl.conf /etc/syslog.conf \
/etc/fstab /etc/security.console.perms /root/.rhosts \
/root/.shosts /etc/shosts.equiv /etc/X11/xdm/Xservers \
/etc/X11/xinit/xserverrc /etc/X11/gdm/gdm.conf \
/etc/cron.allow /etc/cron.deny /etc/at.allow \
/etc/at.deny /etc/crontab /etc/motd /etc/issue \
/usr/share/config/kdm/kdmrc /etc/X11/gdm/gdm.conf \
/etc/securetty /etc/security/access.conf /etc/lilo.conf \
/etc/grub.conf /etc/login.defs /etc/group /etc/profile \
/etc/csh.login /etc/csh.cshrc /etc/bashrc \
/etc/ssh/sshd_config /etc/ssh/ssh_config  \
/etc/cups/cupsd.conf /etc/{,vsftpd/}vsftpd.conf \
/etc/logrotate.conf /root/.bashrc /root/.bash_profile \
/root/.cshrc /root/.tcshrc /etc/vsftpd.ftpusers ; do
        [ -f $file ] && /bin/cp $file $file-preCIS
done
for dir in /etc/xinetd.d /etc/rc[0123456].d \
/var/spool/cron /etc/cron.* /etc/logrotate.d /var/log \
/etc/pam.d /etc/skel ; do
        [ -d $dir ] && /bin/cp -r $dir $dir-preCIS
done
```

1 Patches

1.1 Apply latest OS patches

Action:

1. Make a special directory if it doesn't yet exist:

```
mkdir /usr/local/updates
cd /usr/local/updates
```

2. Download all patches: (replace 8.0 with your distribution version -- remember that Red Hat 9 is "9" not "9.0.")

```
wget ftp://updates.redhat.com/8.0/en/os/i386/\*.rpm
wget ftp://updates.redhat.com/8.0/en/os/noarch/\*.rpm
```

(Note: some firewalls block active FTP and thus will require that you use:
`wget --passive-ftp ftp://URL`)

3. Execute the following command:

```
rpm -Fvh *
```

Discussion:

Developing a procedure for keeping up-to-date with vendor patches is critical for the security and reliability of the system. Vendors issue operating system updates when they become aware of security vulnerabilities and other serious functionality issues, but it is up to their customers to actually download and install these patches.

It's also important to observe that your applications work properly after patching. Though problems in patches are quite rare in Red Hat Linux, it is generally recommended that any patch be deployed to a non-production system first for testing.

Note: Downloading all required patches from the vendor often requires substantial time. As a full complement of patches might exceed 500 megabytes, the time required should not be underestimated.

Some RPMs may need to be installed before others. For the most part, the rpm utility that you use in this process will understand and solve dependencies. Red Hat creates separate instructions for special cases, like the replacement of the kernel or the general C library glibc. You may need to examine the list of updates that you have downloaded to check for any of these cases.

Finally, there is some risk to using an non-patched, non-hardened machine to download the patches, as this entails placing a system with security vulnerabilities on the Internet. Please consider these issues carefully.

Red Hat offers at least partially automated patch download and installation, via Red Hat `up2date`. This guide does not recommend either the adoption or rejection of this tool, as no simple consensus exists for use of automation in patching.

1.2 Configure SSH

Action:

```
cd /etc/ssh
awk '($1=="Protocol") { print "Protocol 2"; next };
    { print }' ssh_config >ssh_config.new
/bin/mv ssh_config.new ssh_config
/bin/chown root:root ssh_config
/bin/chmod 644 ssh_config
if [ "`egrep -l ^Protocol ssh_config`" == "" ]; then
     echo 'Protocol 2' >>ssh_config
fi
awk '/^#?Protocol/ { print "Protocol 2"; next };
     /^#?X11Forwarding/ \
         { print "X11Forwarding yes"; next };
     /^#?IgnoreRhosts/  \
         { print "IgnoreRhosts yes"; next };
     /^#?RhostsAuthentication/     \
         { print " RhostsAuthentication no"; next };
     /^#?RhostsRSAAuthentication/  \
         { print "RhostsRSAAuthentication no"; next };
     /^#?HostbasedAuthentication/  \
         { print "HostbasedAuthentication no"; next };
     /^#?PermitRootLogin/          \
         { print "PermitRootLogin no"; next };
     /^#?PermitEmptyPasswords/     \
         { print "PermitEmptyPasswords no"; next };
     {print}' sshd_config >sshd_config.new
/bin/mv sshd_config.new sshd_config
/bin/chown root:root sshd_config
/bin/chmod 600 sshd_config
```

Discussion:

OpenSSH is a popular free distribution of the standards-track SSH protocols which has become the standard implementation on Linux distributions. For more information on OpenSSH, see www.openssh.org.

The settings in this section attempt to ensure safe defaults for both the client and the server. Specifically, both the `ssh` and the `sshd` server are configured to use only SSH protocol 2, as security vulnerabilities have been found in the first SSH protocol. This may cause compatibility issues at sites still using the vulnerable SSH protocol 1 – these sites should endeavor to configure all systems to use only SSH protocol 2.

2 Minimize `xinetd` network services

2.1 Disable standard services

Action:

```
cd /etc/xinetd.d
for file in chargen chargen-udp cups-lpd daytime \
daytime-udp echo echo-udp eklogin finger gssftp imap \
imaps ipop2 ipop3 krb5-telnet klogin kshell ktalk ntalk \
pop3s rexec rlogin rsh rsync servers services sgi_fam \
talk telnet tftp time time-udp vsftpd wu-ftpd ; do
     chkconfig $file off
done
```

Discussion:

On Linux, `xinetd` has outpaced `inetd` as the default network superserver. Most distributions have been using `xinetd` for some time, there are still many servers that do run `inetd`.

The stock `inetd` and `xinetd` configurations have gotten better and better with each major release over the past years. In 1999, at the time of Red Hat 5.2, distributions offered many services which were either rarely-used or for which there were more secure alternatives. After enabling SSH, it is possible to nearly do away with all `xinetd`-based services, since SSH provides both a secure login mechanism and a means of transferring files to and from the system. The actions above will disable all standard services normally enabled in the Red Hat `xinetd` configuration.

The rest of the actions in this section give the administrator the option of re-enabling certain services. Rather than disabling and then re-enabling these services, experienced administrators may wish to simply disable only those services that they know are unnecessary for their systems.

2.2 Enable `telnet` if absolutely necessary

Question:

Is there a mission-critical reason that requires users to access this system via `telnet`, rather than the more secure SSH protocol?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig telnet on
```

Discussion:

`telnet` uses an unencrypted network protocol, which means data from the login session (such as passwords and all other data transmitted during the session) can be stolen by eavesdroppers on the network, and also that the session can be hijacked by outsiders to gain access to the remote system. The freely-available SSH utilities that ship with Red Hat Linux (see http://www.openssh.com/) provide encrypted network logins and should be used instead.

2.3 Enable FTP if absolutely necessary

Question:

Is this machine an (anonymous) FTP server, or is there a mission-critical reason why data must be transferred to and from this system via `ftp`, rather than `sftp` or `scp`?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig wu-ftpd on
chkconfig vsftpd on
```

Discussion:

Like `telnet`, the FTP protocol is unencrypted, which means passwords and other data transmitted during the session can be captured by sniffing the network, and that the FTP session itself can be hijacked by an external attacker. SSH provides two different encrypted file transfer mechanisms-`scp` and `sftp`-and should be used instead. Even if FTP is required because the local system is an anonymous FTP server, consider requiring non-anonymous users on the system to transfer files via SSH-based protocols. For further information on restricting FTP access to the system, see Item 7.7 below.

Red Hat Linux used WU-FTPd by default through version 7.3. As of version 8.0, Red Hat used vsftpd by default, but shipped with WU-FTPd as an option. The shell commands above activate whichever is present.

Note: Any directory writable by an anonymous FTP server should probably have its own partition. This helps prevent a compromised FTP server from filling a hard drive used by other services.

2.4 Enable `rlogin/rsh/rcp` if absolutely necessary

Question:

Is there a mission-critical reason why `rlogin/rsh/rcp` must be used instead of the more secure `ssh/scp`?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig shell on
chkconfig rsh on
chkconfig login on
chkconfig rlogin on
```

Discussion:

SSH was designed to be a drop-in replacement for these protocols. Given the wide availability of free SSH implementations, it seems unlikely that there is ever a case where these tools cannot be replaced with SSH (again, see http://www.openssh.com/).

If these protocols are left enabled, please also see Item 7.7 for additional security-related configuration settings.

2.5 Enable TFTP if absolutely necessary

Question:

Is this system a boot server or is there some other mission-critical reason why data must be transferred to and from this system via TFTP?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig tftp on
if [ ! -d "/tftpboot" ] ; then
  /bin/mkdir -m 0755 /tftpboot && \
  /bin/chown root:root /tftpboot
fi
```

Discussion:

TFTP is typically used for network booting of diskless workstations, X-terminals, and other similar devices. Routers and other network devices may copy configuration data to remote systems via TFTP for backup. However, unless this system is needed in one of these roles, it is best to leave the TFTP service disabled.

2.6 Enable `IMAP` if absolutely necessary

Question:

Is this machine a mail server with a mission-critical reason to use `imap` to serve mail to remote mail clients?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig imaps on
```

Discussion:

Remote mail clients (like Eudora, Netscape Mail and Kmail) may retrieve mail from remote mail servers using IMAP, the Internet Message Access Protocol, or POP, the Post Office Protocol. If this system is a mail server that must offer this protocol, `imaps` may be activated.

`imaps` activates an SSL-encrypted, and thus much safer, version of IMAP. Standard IMAP is not encrypted and thus allows an attacker to eavesdrop on e-mails being transferred or to take over the connection. It may, based on which authentication method is used, allow an attacker to steal user passwords as well. IMAP-SSL suffers none of these problems.

You may wish to generate a new SSL certificate. For more information, consult http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/s1-email-mua.html#S2-EMAIL-SECURITY.

Should you have an absolute need to reactivate the insecure IMAP server without encrypting SSL, you may apply the above action with `imap` in place of `imaps`.

2.7 Enable `POP` if absolutely necessary

Question:

Is this machine a mail server with a mission-critical reason to use `pop` to serve mail to remote mail clients?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig pop3s on
```

Discussion:

Remote mail clients (like Eudora, Netscape Mail and Kmail) may retrieve mail from remote mail servers using IMAP, the Internet Message Access Protocol, or POP, the Post Office Protocol. If this system is a mail server that must offer the POP protocol, `pop3s` may be activated.

`pop3s` activates an SSL-encrypted, and thus much safer, version of POP. Standard POP is not encrypted and thus allows an attacker to eavesdrop on e-mails being transferred or to take over the connection. It may, based on which authentication method is used, allow an attacker to steal user passwords as well. POP-SSL suffers none of these problems.

You may wish to generate a new SSL certificate. For more information, consult http://www.redhat.com/docs/manuals/linux/RHL-9-Manual/ref-guide/s1-email-mua.html#S2-EMAIL-SECURITY.

Should you have an absolute need to reactivate the insecure POP server without encrypting SSL, you may apply the above action with `pop3` in place of `pop3s`.

3 Minimize boot services

3.1 Set daemon umask

Action:

```
cd /etc/rc.d/init.d
if [ "`grep -l umask functions`" == "" ]; then
   echo "umask 022" >> functions
fi
```

Discussion:

The system default `umask` should be set to at least 022 in order to prevent daemon processes (such as the syslog daemon) from creating world-writable files by default. More restrictive `umask` values (such as `077`) can be used but may cause problems for certain applications – consult vendor documentation for further information.

3.2 Disable xinetd, if possible

Action:

```
cd /etc/xinetd.d
if \
[`awk '($1=="disable" &&$3=="no"){print}' * |wc -l` == 0 ]
then
   chkconfig --level 12345 xinetd off
fi
```

Discussion:

If the actions in Section 2 of this benchmark resulted in no services being enabled in `/etc/xinetd.d`, then one may as well disable the `xinetd` service completely on this system.

3.3 Disable email server, if possible

Question:

Is this system a mail server-that is, does this machine receive and process email from other hosts?

If the answer to this question is yes, then ***do not*** perform the action below.

Action:

```
cd /etc/sysconfig
cat <<END_ENTRIES > sendmail
DAEMON=no
QUEUE=1h
END_ENTRIES
/bin/chown root:root sendmail
/bin/chmod 644 sendmail
```

Discussion:

It is possible to run a Unix system with the Sendmail daemon disabled and still allow users on that system to send email out from that machine. Running Sendmail in "*daemon mode*" (with the `-bd` command-line option) is only required on machines that act as *mail servers*, receiving and processing email from other hosts on the network. Note that if the system is an email server, the administrator is encouraged to search the Web for additional documentation on Sendmail security issues. Some information is available at http://www.deer-run.com/~hal/dns-sendmail/DNSandSendmail.pdf and at http://www.sendmail.org.

Though recent versions of Red Hat have set Sendmail to listen only to the *loopback* network interface, this document still deactivates "*daemon mode*." Listening on the *loopback* interface still presents a higher level of exposure to attack than listening on no network interfaces. Experienced administrators will understand that a chroot-jailed user or program can still interact with a Sendmail process listening on the *loopback* interface.

Note that the initial release of Red Hat 8.0 has a bug in its /etc/init.d/sendmail start script. This script ignores the DAEMON variable in Red Hat 8.0, preventing the above configuration change from affecting Sendmail. This is corrected in a Sendmail patch available from Red Hat, which you should definitely install. You will not need to repeat the step above after installing that patch.

3.4 Disable GUI Login if possible

Question:

Is there a mission-critical reason to run a GUI login program on this system?

If the answer to this question is no, proceed with the actions below.

Action:

```
sed 's/id:5:initdefault:/id:3:initdefault:/' \
   < /etc/inittab > /etc/inittab.new
/bin/mv /etc/inittab.new /etc/inittab
/bin/chown root:root /etc/inittab
/bin/chmod 0600 /etc/inittab
```

Discussion:

There's usually no reason to run X Windows on a dedicated server machine, like a dedicated webserver. This action disables graphical login, if present, leaving the user to login via a normal text-based console. If you elect to deactivate the GUI login screen, users can still run X Windows by typing `startx` at the shell prompt.

In Red Hat Linux, there are two main runlevels that the system runs in. If this system boots directly into X Windows, so as to allow graphical login or easy use of specialized X terminals, then runlevel 5 is appropriate. Otherwise, for normal text-based console login, runlevel 3 is desirable. GUI login is activated or deactivated by changing this runlevel in `/etc/inittab`. Again, note that runlevel 3 still allows the user to run X Windows by typing `startx` at the shell prompt.

3.5 Disable X font server if possible

Question:

Is there a mission-critical reason to run X Windows on this system?

If the answer to this question is no, proceed with the actions below.

Action:

```
chkconfig xfs off
```

Discussion:

There's usually no reason to run X Windows on a dedicated server machine, like a dedicated webserver. If you won't be using an X server on this machine, this action will deactivate the font server.

3.6 Disable Standard Boot Services

Action:

```
for file in apmd canna FreeWnn gpm hpoj innd irda isdn \
kdcrotate lvs mars-nwe oki4daemon privoxy rstatd rusersd \
rwalld rwhod spamassassin wine
do
     chkconfig --level 12345 $file off
done
for file in nfs nfslock autofs ypbind ypserv yppasswdd \
            portmap smb netfs lpd apache httpd tux snmpd \
            named postgresql mysqld webmin kudzu squid \
            cups
do
     chkconfig --level 12345 $file off
done
for user in rpc rpcuser lp apache http httpd named dns \
            mysql postgres squid
do
     /usr/sbin/usermod -L $user
done
```

Discussion:

Every system daemon that does not have a clear and necessary purpose on the host should be deactivated. This greatly reduces the chances that the machine will be running a vulnerable daemon when the next vulnerability is discovered in its operating system.

Red Hat Linux uses a facility called `chkconfig` to manage all the SysV rc-scripts. `chkconfig` adds or deletes links in each of the appropriate runlevel directories (`/etc/rc.d/rc*.d`) to activate or deactivate each of the rc-scripts.

This process "chkconfig's" all of the rc-scripts off, so that the local administrator can easily reactivate any of these scripts upon discovery of a mission-critical need for one of these services. One could reactivate the *daemon* script by typing `chkconfig daemon on` in most cases, which activates it in runlevels 2 through 5. If one of these runlevels is undesirable, like runlevel 2 for the NFS script, or the script needs to run in one of the other available runlevels, `chkconfig` takes the argument "`–level <levels>`" where one can explicitly specify runlevels that it should act on.

Note that vendor patches may restore some of the original entries in the `/etc/rc.d/rc*.d` directories – it is always a good idea to check these boot directories and remove any scripts that may have been added by the patch installation process.

The rest of the actions in this section give the administrator the option of re-enabling certain services – in particular, the services that are disabled in the second loop in the "**Action**" section above. Rather than disabling and then re-enabling these servies,

experienced administrators may wish to simply disable only those services that they know are unnecessary for their systems.

The third loop in the "**Action**" section locks daemon-user accounts related to servers that we examine, both by setting a lockout password and by changing the shell to `/dev/null`. This will not prevent a given daemon from running as these users – it simply confirms that these users are not available for human login.

The fourth loop in the "**Action**" section deactivates GUI login. While the process for deactivating GUI login doesn't retain the parallelism of deactivating rc-scripts with `chkconfig`, it's implemented here to maintain parallelism with the Center for Internet Security's other benchmarks.

Note: not all of the scripts listed above will exist on all systems, as this is a superset of the available rc-scripts in the various Red Hat distribution versions. The benchmark's recommended action will register some trivial errors on each distribution version as a result – these are not cause for alarm.

3.7 Only enable SMB (Windows filesharing) processes if absolutely necessary

Question:

Is this machine sharing files via the Windows filesharing protocols?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig smb on
```

Discussion:

Red Hat Linux includes the popular Open Source Samba server for providing file and print services to Windows-based systems. This allows a Unix system to act as a file or print server in on a Windows network, and even act as a Domain Controller (authentication server) to older Windows operating systems. However, if this functionality is not required by the site, the service should be disabled.

3.8 Only enable NFS server processes if absolutely necessary

Question:

Is this machine an NFS file server?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig --level 345 nfs on
```

Discussion:

NFS is frequently exploited to gain unauthorized access to files and systems. Clearly there is no need to run the NFS server-related daemons on hosts that are not NFS servers. If the system is an NFS server, the admin should take reasonable precautions when exporting file systems, including restricting NFS access to a specific range of local IP addresses and exporting file systems "read-only" where appropriate. For more information, consult the `exports` manual page.

3.9 Only enable NFS client processes if absolutely necessary

Question:

Is there a mission-critical reason why this system must access file systems from remote servers via NFS?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig --level 345 nfslock on
chkconfig --level 345 autofs on
```

Discussion:

Again, unless there is a significant need for this system to acquire data via NFS, administrators should disable NFS-related services. Note that other file transfer schemes (such as `rdist` via SSH) can often be preferable to NFS for certain applications.

3.10 Only enable NIS client processes if absolutely necessary

Question:

Is there a mission-critical reason why this machine must be an NIS client?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig ypbind on
```

Discussion:

Unless this site must use NIS, it should really be avoided. While it can be very useful for transparently scaling the number of workstations, it's not well designed for security.

3.11 Only enable NIS server processes if absolutely necessary

Question:

Is there a mission-critical reason why this machine must be an NIS server?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig ypserv on
chkconfig yppasswdd on
```

Discussion:

Unless this site must use NIS, it should be avoided. While it can be very useful for transparently scaling the number of workstations, it's not well designed for security.

3.12 Only enable RPC portmap process if absolutely necessary

Question:

Are any of the following statements true?

- *This machine is an NFS client or server*
- *This machine is an NIS (YP) or NIS+ client or server*
- *The machine runs a third-party software application which is dependent on RPC support*

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig --level 345 portmap on
```

Discussion:

RPC-based services typically use very weak or non-existent authentication and yet may share very sensitive information. Unless one of the services listed above is required on this machine, best to disable RPC-based tools completely. If there is uncertainty in whether or not a particular third-party application requires RPC services, consult with the application vendor.

3.13 Only enable `netfs` script if absolutely necessary

Question:

Is this machine sharing files via the NFS, Novell Netware or Windows filesharing protocols?

If the answer to this question is yes, proceed with the actions below.

Action:

```
chkconfig --level 345 netfs on
```

Discussion:

If there are no network filesharing protocols being used, one can deactivate the *netfs* script. This script mounts network drives on the client. Though this is not a persistent daemon and thus not so dangerous, thinning out the `/etc/rc.d/rcN.d` directories makes the system much easier to audit.

3.14 Only enable printer daemon processes if absolutely necessary

Question:

Is this system a print server, or is there a mission-critical reason why users must submit print jobs from this system?

If the answer to this question is yes, then perform the action below.

Action:

```
if [ -e /etc/init.d/cups ] ; then
   chkconfig cups on
   sed 's/^\#User lp/User lp/' /etc/cups/cupsd.conf \
   >/etc/cups/cupsd.conf.new
   sed 's/^\#Group sys/Group sys/' \
   /etc/cups/cupsd.conf.new >/etc/cups/cupsd.conf
   rm -f /etc/cups/cupsd.conf.new
   /bin/chown lp:sys /etc/cups/cupsd.conf
   /bin/chmod 600 /etc/cups/cupsd.conf
fi
chkconfig hpoj on
chkconfig lpd on
```

Discussion:

If users will never print files from this machine and the system will never be used as a print server by other hosts on the network, then it is safe to disable the print daemon,

`lpd` or `cupsd`. The Unix print servers have generally had a poor security record – be sure to keep up-to-date on vendor patches.

Note that this item also sets `cupsd`, when present, to run as a non-root user and group, namely user lp and group sys.

Finally, note that you need only activate hpoj if you're using a "multi-function" device from Hewlett Packard, like one of the OfficeJet, LaserJet, Printer/Scanner/Copier ("PSC"), and PhotoSmart printer products.

3.15 Only enable Web server processes if absolutely necessary

Question:

Is there a mission-critical reason why this system must run a Web server?

If the answer to this question is yes, then perform the action below.

Action:

```
for file in apache httpd tux ; do
  chkconfig $file on
done
```

Discussion:

Even if this machine is a Web server, the local site may choose not to use the Web server provided with Linux in favor of a locally developed and supported Web environment.

The *TUX* server is a new kernel-based webserver, only available on the most recent Linux distributions, which gains performance while sacrificing the flexibility offerred by a full-featured webserver like Apache. Even if a webserver is kept running, the startup script for either TUX (tux) or apache (apache/httpd) should be deactivated.

3.16 Only enable SNMP processes if absolutely necessary

Question:

Are hosts at this site remotely monitored by a tool (e.g., HP OpenView, MRTG, Cricket) that relies on SNMP?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig snmpd on
```

Discussion:

If SNMP is used to monitor the hosts on this network, experts recommend changing the default community string used to access data via SNMP. On Red Hat 7.0 and later systems, this parameter can be changed by modifying the last word on the following line in `/etc/snmp/snmpd.conf`:

```
com2sec notConfigUser    default   public
```

3.17 Only enable DNS server process if absolutely necessary

Question:

Is this machine a DNS server, or nameserver, for this site?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig named on
```

Discussion:

Most of the machines in the organization do not need a DNS server running on the box. Unless this is one of the organization's name servers, it is safe to shut this down.

If this must be left active, please patch often and consider tightening the configuration. One highly suggested configuration is to bind the DNS server program in a chroot environment. This significantly restricts the resources that the DNS server has access to on the system, reducing this set to the minimum required for the program to function properly. The BIND nameserver that most machines run has had major security problems recently.

3.18 Only enable SQL server processes if absolutely necessary

Question:

Is this machine an SQL (database) server?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig postgresql on
chkconfig mysqld on
```

Discussion:

If this machine does not need to run the mainstream database (SQL) servers Postgres or MySQL, it is safe to deactivate them.

3.19 Only enable Webmin processes if absolutely necessary

Question:

Does the site absolutely need to administer the system through the remote webmin tool?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig webmin on
```

Discussion:

One can remotely administer a system through the relatively safe SSH remote shell system. Webmin, and other tools like it, can be dangerous as they have a history of bad authentication or session management. If this site currently uses webmin or a similar remote administration tool, it should research and consider this decision carefully.

3.20 Only enable Squid cache server if absolutely necessary

Question:

Do you use the squid web cache to speed up web transactions?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig squid on
```

Discussion:

Squid can actually be beneficial to security, as it imposes a proxy between the client and server. On the other hand, if it is not being used, it should be deactivated. This deactivation decreases the risk of system compromise should a security vulnerability later be discovered in Squid. Finally, if your site does used Squid, do configure it carefully. Many Squid caches are badly configured to either allow outsider attackers to probe internal machines through the firewall or to use the cache to hide their true source IP address from their target hosts. Each site should configure Squid to not allow people outside their perimeter to use the cache without authentication of some sort.

3.21 Only enable Kudzu hardware detection if absolutely necessary

Question:

Does the site absolutely need to allow users at the console to add hardware to the system?

If the answer to this question is yes, then perform the action below.

Action:

```
chkconfig --level 345 kudzu on
```

Discussion:

Kudzu is Red Hat's hardware detection program, which is normally set to run during system startup. It detects changes in hardware and, without demanding authentication of any sort, allows the user at the console to configure that hardware. This lack of authentication presents the primary danger – any user sitting at the console during a reboot can configure any new devices added to the system.

This configuration is an unnecessary risk for most sites, with the exception of those that need to allow users to easily make hardware changes without having a root password. Sites in the exception class might need to allow students to connect external hard drives, backup drives or other potentially common external devices.

If you deactivate this rc-script, Kudzu is still accessible. You will simply need to run `/etc/rc.d/init.d/kudzu` at the shell prompt, while logged in as root.

4 Kernel Tuning

4.1 Network Parameter Modifications

Action:

```
cat <<END_SCRIPT >> /etc/sysctl.conf
net.ipv4.tcp_max_syn_backlog = 4096
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.all.accept_redirects = 0
net.ipv4.conf.all.secure_redirects = 0

net.ipv4.conf.default.rp_filter = 1
net.ipv4.conf.default.accept_source_route = 0
net.ipv4.conf.default.accept_redirects = 0
net.ipv4.conf.default.secure_redirects = 0
END_SCRIPT
/bin/chown root:root /etc/sysctl.conf
/bin/chmod 0600 /etc/sysctl.conf
```

Discussion:

For an explanation of some of these parameters, see `/Documentation/networking/ip-sysctl.txt` in your local copy of the kernel source or read the latest from the *Cross-referencing Linux* site:

http://lxr.linux.no/source/Documentation/networking/ip-sysctl.txt

4.2 Additional Network Parameter Modifications

Question:

Is this system going to be used as a firewall or gateway to pass network traffic between different networks?

If the answer to this question is yes, then ***do not*** perform the action below.

Action:

```
cat <<END_SCRIPT >> /etc/sysctl.conf
net.ipv4.ip_forward = 0
net.ipv4.conf.all.send_redirects = 0
net.ipv4.conf.default.send_redirects = 0
END_SCRIPT
/bin/chown root:root /etc/sysctl.conf
/bin/chmod 0600 /etc/sysctl.conf
```

Discussion:

For an explanation of some of these parameters, see `/Documentation/networking/ip-sysctl.txt` in your local copy of the kernel source or read the latest from the *Cross-referencing Linux* site:

http://lxr.linux.no/source/Documentation/networking/ip-sysctl.txt

5 Logging

The items in this section cover enabling various different forms of system logging in order to keep track of activity on the system. Because it is often necessary to correlate log information from many different systems (particularly after a security incident) experts recommend establishing some form of time synchronization among systems and devices connected to the local network. The standard Internet protocol for time synchronization is the Network Time Protocol (NTP), which is supported by most network-ready devices. More information on NTP can be found at `http://www.ntp.org` and `http://www.ibiblio.org/pub/Linux/docs/HOWTO/other-formats/html_single/TimePrecision-HOWTO.html`.

5.1 Capture messages sent to syslog `AUTHPRIV` facility

Action:

```
if [ `grep -c 'authpriv\.' /etc/syslog.conf` -eq 0 ]
then
   echo -e "authpriv.*\t\t\t\t/var/log/secure" \
      >>/etc/syslog.conf
fi
touch /var/log/secure
/bin/chown root:root /var/log/secure
/bin/chmod 600 /var/log/secure
```

Discussion:

Not all Linux distributions, especially the older ones, capture logging information which is sent to the `LOG_AUTHPRIV` facilities. This is unfortunate, since a great deal of important security-related information is sent via these channels (e.g., successful and failed `su` attempts, failed login attempts, root login attempts, etc.). The above action causes this information to be captured in the `/var/log/secure` file (which is only readable by the superuser). This file should be reviewed and archived on a regular basis.

5.2 Capture detailed FTP daemon logs

Action:

```
if [ -f /etc/ftpaccess ] ; then
  cd /etc/xinetd.d
  awk '/server_args/ \
  { print "        server_args = -l -a -d" ; next };
  { print }' wu-ftpd >wu-ftpd.new
  /bin/mv wu-ftpd.new wu-ftpd
  /bin/chown root:root wu-ftpd
  /bin/chmod 644 wu-ftpd
fi
if [ -f /etc/vsftpd.conf ] ; then
  file="/etc/vsftpd.conf"
else
  file="/etc/vsftpd/vsftpd.conf"
fi
if [ -f $file ] ; then
  awk '/^#?xferlog_std_format/ \
          { print "xferlog_std_format=NO"; next };
       /^#?log_ftp_protocol/ \
          { print "log_ftp_protocol=YES"; next };
       { print }' $file >${file}.new
  if [ `egrep -c log_ftp_protocol $file` == 0 ] ; then
     echo "log_ftp_protocol=YES" >>${file}.new
  fi
  /bin/mv ${file}.new $file
  /bin/chmod 0600 $file
  /bin/chown root:root $file
fi
```

Discussion:

Red Hat already logs connections and all files transferred in WU-FTPd and vsftpd. The modifications above ensure that all commands sent to the server are logged. In WU-FTPd, the Action above also requires the server to log all security violations or policy boundary conditions and to ensure that file transfers are logged to syslog, in addition to the default `/var/log/xferlog`.

5.3 Confirm permissions on system log files

Action:

```
cd /var/log
/bin/chmod o-w boot.log* cron* dmesg ksyms* httpd/* \
maillog* messages* news/* pgsql rpmpkgs* samba/* \
scrollkeeper.log secure* spooler* squid/* vbox/* wtmp
/bin/chmod o-rx boot.log* cron* maillog* messages* pgsql \
secure* spooler* squid/*
/bin/chmod g-w boot.log* cron* dmesg httpd/* ksyms* \
maillog* messages* pgsql rpmpkgs* samba/* \
scrollkeeper.log secure* spooler*
/bin/chmod g-rx boot.log* cron* maillog* messages* pgsql \
secure* spooler*
/bin/chmod o-w gdm/ httpd/ news/ samba/ squid/ vbox/
/bin/chmod o-rx httpd/ samba/ squid/
/bin/chmod g-w gdm/ httpd/ news/ samba/ squid/ vbox/
/bin/chmod g-rx httpd/ samba/

/bin/chown -R root:root .
/bin/chgrp utmp wtmp
/bin/chown -R news:news news

/bin/chown postgres:postgres pgsql
/bin/chown -R squid:squid squid
```

Discussion:

It's critical to protect system log files from being modified by unauthorized individuals. Also, certain logs contain sensitive data that should only be available to the system administrator.

6 File/Directory Permissions/Access

6.1 Add 'nodev' option to appropriate partitions in /etc/fstab

Action:

```
awk '($3 ~ /^ext[23]$/ && $2 != "/") \
      { $4 = $4 ",nodev" }; \
      { print }' /etc/fstab >/etc/fstab.new
/bin/mv /etc/fstab.new /etc/fstab
/bin/chown root:root /etc/fstab
/bin/chmod 0644 /etc/fstab
```

Discussion:

Placing "nodev" on these partitions prevents users from mounting unauthorized devices on any partitions that we know should not contain devices. There should be little need to mount devices on any partitions other than /dev.

One notable exception, of course, is the case where system programs are being placed into "chroot prisons" – these often require that several devices be created in the chroot directory. If you are using chroot prisons on your machines, you should be careful with the nodev option.

6.2 Add 'nosuid' and 'nodev' option for removable media in /etc/fstab

Action:

```
awk '($2 ~ /^\/m.*\/(floppy|cdrom)$/) \
     { $4 = $4 ",nosuid,nodev" }; \
     { print }' /etc/fstab >/etc/fstab.new
/bin/mv /etc/fstab.new /etc/fstab
/bin/chown root:root /etc/fstab
/bin/chmod 0644 /etc/fstab
```

Discussion:

Removable media is one vector by which malicious software can be introduced onto the system. By forcing these file systems to be mounted with the nosuid option, the administrator prevents users from bringing set-UID programs onto the system via CD-ROMs and floppy disks. We also force these filesystems to mount with the nodev option, as explained in item 6.1.

If this machine has multiple CD-ROM or floppy drives, additional action must be taken. Simply add nosuid to the fourth field for the /etc/fstab lines that reference those drives.

6.3 Disable user-mounted removable filesystems

Question:

Is there a mission-critical reason to allow unprivileged users to mount CD-ROMs and floppy disk file systems on this system?

If the answer to this question is yes, then ***do not*** perform the action below.

Action:

```
cd /etc/security
awk '($1 == "<console>") && ($3 !~ \
     /sound|fb|kbd|joystick|v4l|mainboard|gpm|scanner/) \
          { $1 = "#<console>" };
     { print }' console.perms >console.perms.new
/bin/mv console.perms.new console.perms
/bin/chown root:root console.perms
/bin/chmod 0600 console.perms
```

Discussion:

In Red Hat Linux, the `pam_console` PAM module gives the user at console (the machine's true physical keyboard) temporarily enhanced privileges. This is configured through the `/etc/security/console.perms` file. Under the Red Hat-shpped settings, the console user is given ownership of the floppy and CD-ROM drive, along with a host of other devices. Many of these devices correspond to removable media and thus represent a security risk. This item disables the enhanced privileges on these devices.

Be aware that allowing users to mount and access data from removable media drives makes it easier for malicious programs and data to be imported onto the network.

6.4 Verify `passwd`, `shadow`, and `group` file permissions

Action:

```
cd /etc
/bin/chown root:root passwd shadow group
/bin/chmod 644 passwd group
/bin/chmod 400 shadow
```

Discussion:

These are the default owners and access permissions for these files.

6.5 World-writable directories should have their sticky bit set

Action:

The automated tool supplied with this benchmark will flag world-writable directories that do not have the sticky bit set.

Administrators who wish to obtain a list of these directories may execute the following commands

```
for part in `awk '($3 == "ext2" || $3 == "ext3") \
                  { print $2 }' /etc/fstab`
do
   find $part -xdev -type d \
        \( -perm -0002 -a ! -perm -1000 \) -print
done
```

Discussion:

When the so-called "sticky bit" is set on a directory, then only the owner of a file may remove that file from the directory (as opposed to the usual behavior where anybody with write access to that directory may remove the file). Setting the sticky bit prevents users from overwriting each other's files, whether accidentally or maliciously, and is generally appropriate for most world-writable directories. However, consult appropriate vendor documentation before blindly applying the sticky bit to any world writable directories found in order to avoid breaking any application dependencies on a given directory.

6.6 Find unauthorized world-writable files

Action:

The automated testing tool supplied with this benchmark will flag unexpected world-writable files on the system.

Administrators who wish to obtain a list of the world-writable files currently installed on the system may run the following commands:

```
for part in \
   `awk '($3 == "ext2" || $3 == "ext3") \
   { print $2 }' /etc/fstab`
do
   find $part -perm -0002 -type f -xdev -print
done
```

Discussion:

Data in world-writables files can be modified and compromised by any user on the system. World-writable files may also indicate an incorrectly written script or program that could potentially be the cause of a larger compromise to the system's integrity.

Generally removing write access for the "other" category (`chmod o-w <filename>`) is advisable, but always consult relevant vendor documentation in order to avoid breaking any application dependencies on a given file.

6.7 Find unauthorized SUID/SGID system executables

Action:

The automated testing tool supplied with this benchmark will flag unexpected set-UID and set-GID applications on the system.

Administrators who wish to obtain a list of the set-UID and set-GID programs currently installed on the system may run the following commands:

```
for part in \
   `awk '($3 == "ext2" || $3 == "ext3") \
   { print $2 }' /etc/fstab`
do
        find $part \( -perm -04000 -o -perm -02000 \) \
             -type f -xdev -print
done
```

Discussion:

The administrator should take care to ensure that no rogue set-UID programs have been introduced into the system. In addition, if possible, the administrator should attempt a Set-UID audit and reduction.

7 System Access, Authentication, and Authorization

7.1 Remove `.rhosts` support in PAM configuration files

Action:

```
for file in  /etc/pam.d/* ; do
   grep -v rhosts_auth $file > ${file}.new
   /bin/mv ${file}.new $file
   /bin/chown root:root $file
   /bin/chmod 644 $file
done
```

Discussion:

Used in conjunction with the BSD-style "r-commands" (`rlogin`, `rsh`, `rcp`), `.rhosts` files implement a weak form of authentication based on the network address or host name of the remote computer (which can be spoofed by a potential attacker to

exploit the local system). Disabling `.rhosts` support helps prevent users from subverting the system's normal access control mechanisms.

If `.rhosts` support is required for some reason, some basic precautions should be taken when creating and managing `.rhosts` files. Never use the "+" wildcard character in `.rhosts` files. In fact, `.rhosts` entries should always specify a specific trusted host name along with the user name of the trusted account on that system (e.g., "`trustedhost alice`" and not just "`trustedhost`"). Avoid establishing trust relationships with systems outside of the organization's security perimeter and/or systems not controlled by the local administrative staff. Firewalls and other network security elements should actually block `rlogin`/`rsh`/`rcp` access from external hosts. Finally, make sure that `.rhosts` files are only readable by the owner of the file (i.e., these files should be mode `600`).

7.2 Create symlinks for dangerous files

Action:

```
for file in /root/.rhosts /root/.shosts /etc/hosts.equiv \
            /etc/shosts.equiv ;
do
    /bin/rm  -f $file
    ln -s /dev/null $file
done
```

Discussion:

The `/root/.rhosts`, `/root/.shosts`, and `/etc/hosts.equiv` files enable a weak form of access control (see the discussion of `.rhosts` files in the item above). Attackers will often target these files as part of their exploit scripts. By linking these files to `/dev/null`, any data that an attacker writes to these files is simply discarded (though an astute attacker can still remove the link prior to writing their malicious data).

7.3 Create `ftpusers` files

Action:

```
for name in `cut -d: -f1 /etc/passwd`
do
   if [ `id -u $name` -lt 500 ]
   then
      echo $name >> /etc/ftpusers
   fi
done
/bin/chown root:root /etc/ftpusers
/bin/chmod 600 /etc/ftpusers
if [ -e /etc/vsftpd.conf ] || \
[ -e /etc/vsftpd/vsftpd.conf ]; then
   /bin/rm -f /etc/vsftpd.ftpusers
   /bin/cp /etc/ftpusers /etc/vsftpd.ftpusers
fi
```

Discussion:

`/etc/ftpusers` and `/etc/vsftp.ftpusers` contain a list of users who *are not* allowed to access the system via WU-FTPd and vsftpd, respectively. Generally, only normal users should ever access the system via FTP-there should be no reason for "system" type accounts to be transferring information via this mechanism. Certainly the `root` account should *never* be allowed to transfer files directly via FTP.

7.4 Prevent X server from listening on port 6000/tcp

Action:

```
if [ -e /etc/X11/xdm/Xservers ] ; then
cd /etc/X11/xdm
awk '($1 !~ /^#/ && $3 == "/usr/X11R6/bin/X") \
         { $3 = $3 " -nolisten tcp" };
     { print }' Xservers > Xservers.new
/bin/mv Xservers.new Xservers
/bin/chown root:root Xservers
/bin/chmod 444 Xservers
fi
if [ -e /etc/X11/gdm/gdm.conf ] ; then
cd /etc/X11/gdm
awk -F= '($2 ~ /\/X$/) \
            { printf("%s -nolisten tcp\n", $0); next };
         { print }' gdm.conf > gdm.conf.new
/bin/mv gdm.conf.new gdm.conf
/bin/chown root:root gdm.conf
/bin/chmod 644 gdm.conf
fi
if [ -d /etc/X11/xinit ] ; then
cd /etc/X11/xinit
if [ -e xserverrc ] ; then
   awk '/X/ && !/^#/ \
   { print $0 " :0 -nolisten tcp \$@"; next }; \
   { print }' xserverrc > xserverrc.new
   /bin/mv xserverrc.new xserverrc
else
   cat <<END >xserverrc
#!/bin/bash
exec X :0 -nolisten tcp \$@
END
fi
/bin/chown root:root xserverrc
/bin/chmod 755 xserverrc
fi
```

Discussion:

X servers listen on port 6000/tcp for messages from remote clients running on other systems. However, X Windows uses a relatively insecure authentication protocol—an attacker who is able to gain unauthorized access to the local X server can easily compromise the system. Invoking the "`-nolisten tcp`" option causes the X server not to listen on port 6000/tcp by default.

This does prevent authorized remote X clients from displaying windows on the local system as well. However, the forwarding of X events via SSH will still happen normally. This is the preferred and more secure method transmitting results from remote X clients in any event.

7.5 Restrict `at`/`cron` to authorized users

Action:

```
cd /etc/
/bin/rm  -f cron.deny at.deny
echo root >cron.allow
echo root >at.allow
/bin/chown root:root cron.allow at.allow
/bin/chmod 400 cron.allow at.allow
```

Discussion:

The `cron.allow` and `at.allow` files are a list of users who are allowed to run the `crontab` and `at` commands to submit jobs to be run at scheduled intervals. On many systems, only the system administrator needs the ability to schedule jobs.

Note that even though a given user is not listed in `cron.allow`, `cron` jobs can still be run as that user. `cron.allow` only controls administrative access to the `crontab` command for scheduling and modifying `cron` jobs.

7.6 Restrict permissions on crontab files

Action:

```
/bin/chown root:root /etc/crontab
/bin/chmod 400 /etc/crontab
/bin/chown -R root:root /var/spool/cron
/bin/chmod -R go-rwx /var/spool/cron
/bin/chown -R root:root /etc/cron.*
/bin/chmod -R go-rwx /etc/cron.*
```

Discussion:

The system crontab files are accessed only by the `cron` daemon (which runs with superuser privileges) and the `crontab` command (which is set-UID to root). Allowing unprivileged users to read or (even worse) modify system crontab files can create the potential for a local user on the system to gain elevated privileges.

7.7 Create appropriate warning banners

Action:

1. Create banners for console and X mode:

```
if [ "`egrep -l Authorized /etc/motd`" == "" ]; then
   echo "Authorized uses only. All activity may be \
monitored and reported." >>/etc/motd
fi
if [ "`egrep -l Authorized /etc/issue`" == "" ]; then
   echo "Authorized uses only. All activity may be \
monitored and reported." >>/etc/issue
fi
if [ "`egrep -l Authorized /etc/issue.net`" == "" ]; then
   echo "Authorized uses only. All activity may be \
monitored and reported." >>/etc/issue.net
fi
/bin/chown root:root /etc/motd /etc/issue /etc/issue.net
/bin/chmod 644 /etc/motd /etc/issue /etc/issue.net

if [ -e /etc/X11/xdm/kdmrc ] ; then
cd /etc/X11/xdm
awk '/GreetString=/ \
     { print "GreetString=Authorized uses only!"; next };
     { print }' kdmrc >kdmrc.new
/bin/mv kdmrc.new kdmrc
/bin/chown root:root kdmrc
/bin/chmod 644 kdmrc
fi
if [ -e /etc/X11/gdm/gdm.conf ] ; then
cd /etc/X11/gdm
awk '/^Greeter=/ && /gdmgreeter/ \
        { printf("#%s\n", $0); next };
     /^#Greeter=/ && /gdmlogin/ \
        { $1 = "Greeter=/usr/bin/gdmlogin" };
     /Welcome=/ \
        { print "Welcome=Authorized uses only!"; next };
     { print }' gdm.conf >gdm.conf.new
/bin/mv gdm.conf.new gdm.conf
/bin/chown root:root gdm.conf
/bin/chmod 644 gdm.conf
fi
```

2. Create "authorized only" banners for network services using TCP Wrappers:

```
mkdir /etc/banners ; cd /etc/banners
if [ -e /usr/doc/tcp_wrappers-7.6/Banners.Makefile ]; then
  file=/usr/doc/tcp_wrappers-7.6/Banners.Makefile
else
  file=/usr/share/doc/tcp_wrappers-7.6/Banners.Makefile
fi
cp $file Makefile
echo "Authorized uses only.  All activity may be \
monitored and reported." > prototype
make
cd /etc/xinetd.d
for file in telnet krb5-telnet ; do
   if [ -f $file ]; then
      awk '( $1 == "}" ) \
           { print "banner = /etc/banners/in.telnetd" };
           { print }' $file >$file.new
      /bin/mv $file.new $file
   fi
done
for file in wu-ftpd gssftp ; do
   if [ -f $file ]; then
      awk '( $1 == "}" ) \
           { print "banner = /etc/banners/in.ftpd" };
           { print }' $file >$file.new
      /bin/mv $file.new $file
   fi
done
for file in rsh kshell ; do
   if [ -f $file ]; then
      awk '( $1 == "}" ) \
           { print "banner = /etc/banners/in.rshd" };
           { print }' $file >$file.new
      /bin/mv $file.new $file
   fi
done
for file in rlogin klogin eklogin ; do
   if [ -f $file ]; then
      awk '( $1 == "}" ) \
           { print "banner = /etc/banners/in.rlogind" };
           { print }' $file >$file.new
      /bin/mv $file.new $file
fi ; done
/bin/chown root:root {krb5-,}telnet gssftp wu-ftpd rsh \
kshell rlogin klogin eklogin
/bin/chmod 644 {krb5-,}telnet gssftp wu-ftpd rsh kshell \
rlogin klogin eklogin
```

3. Create "authorized only" banners vsftpd, if applicable:

```
cd /etc
if [ -d vsftpd ]; then
   cd vsftpd
fi
if [ -e vsftpd.conf ] ; then
   echo "ftpd_banner=Authorized uses only. All activity \
may be monitored and reported." >> vsftpd.conf
fi
```

Discussion:

It is a widely held belief that presenting some sort of statutory warning message at login time will assist the prosecution of trespassers on the computer system. Changing some of these login banners also has the side effect of hiding OS version information and other detailed system information, which an attacker might find useful when targeting their attacks (though there are other mechanisms available for acquiring this information). Clearly, the content of all such statutory warnings should be reviewed by the organization's local legal counsel before the above modifications are made.

This item configures console and X Windows "authorized users only" banner messages.

7.8 Configure `xinetd` access control

Action:

Insert the following line into the "`defaults`" block in `/etc/xinetd.conf`:

`only_from=<net>/<num_bits> <net>/<num_bits>`

where each *`<net>/<num_bits>`* combination represents one network block in use by your organization.

For example: `only_from=192.168.1.0/24` would restrict connections to only the `192.168.1.0/24` network, with netmask `255.255.255.0`.

Discussion:

This item configures `xinetd` to use simple IP-based access control and log connections.

Just as `xinetd`'s access control mechanisms are used to monitor illicit connection attempts, the popular PortSentry tool (http://www.psionic.com/products/portsentry.html) can be used to monitor access attempts on unused ports. Note that running PortSentry may result in the CIS testing

tools reporting "false positives" for "active" ports that are actually being held by the PortSentry daemon. Consider replacing the PortSentry daemon with PSAD, short for Port Scan Attack Detector, available from http://www.cipherdyne.com/psad/. Unlike PortSentry, PSAD doesn't have to hold open ports -- instead, it communicates directly with the kernel.

7.9 Restrict root logins to system console

Action:

```
/bin/cp /dev/null /etc/securetty
for i in 1 2 3 4 5 6; do
  echo tty$i >>/etc/securetty
  echo vc/$i >>/etc/securetty
done
echo console >>/etc/securetty
/bin/chown root:root /etc/securetty
/bin/chmod 400 /etc/securetty
```

Discussion:

Anonymous root logins should never be allowed, except on the system console in emergency situations. At all other times, the administrator should access the system via an unprivileged account and use some authorized mechanism (such as the `su` command, or the freely-available `sudo` package) to gain additional privilege. These mechanisms provide at least some limited audit trail in the event of problems.

7.10 Set LILO/GRUB Password

Action (if you have an `/etc/lilo.conf` file):

1. Add the following lines to the *beginning* of `/etc/lilo.conf`

```
restricted
password=<password>
```

Replace <*password*> with an appropriate password for your organization.

2. Execute the following commands as root

```
/bin/chown root:root /etc/lilo.conf
/bin/chmod 600 /etc/lilo.conf
lilo
```

Action (if you have an `/etc/grub.conf` file):

1. Add this line to `/etc/grub.conf` before the first uncommented line.

```
password <password>
```

Replace <*password*> with an appropriate password for your organization.

2. Execute the following commands as root

```
/bin/chown root:root /etc/grub.conf
/bin/chmod 600 /etc/grub.conf
```

Discussion:

By default on most Linux systems, the bootloader prompt allows an attacker to subvert the normal boot process very easily. The action above will allow the system to boot normally, only requiring a password when the user attempts to modify the boot process by passing commands to LILO or GRUB. Make sure to replace <*password*> in the actions above with a good password.

7.11 Require authentication for single-user-mode

Action:

```
cd /etc
if [ "`grep -l sulogin inittab`" = "" ]; then
   awk '{ print };
        /^id:[0123456sS]:initdefault:/ \
           { print "~~:S:wait:/sbin/sulogin" }' \
      inittab >inittab.new
   /bin/mv inittab.new inittab
   /bin/chown root:root inittab
   /bin/chmod 644 inittab
fi
```

Discussion:

By default on Red Hat Linux, you can enter single user mode simply by typing "linux single" at the LILO prompt or in the GRUB boot-editing menu. Some believe that this is left in to ease support of users with lost root passwords. In any case, it represents a clear security risk – authentication should always be required for root-level access. It should be noted that it is extremely difficult to prevent compromise by any attacker who has knowledge, tools, and full physical access to a system. This kind of measure simply increases the difficulty of compromise by requiring more of each of these factors.

These last two items have attempted to address concerns of physical/boot security. To make these preparations more complete, one should consider setting the BIOS to boot only from the main hard disk and locking this setting with a BIOS password. For more

information on reducing the threat posed by an attacker with physical/boot access, consider the article "Anyone with a Screwdriver Can Break In," available via www.bastille-linux.org/jay/anyone-with-a-screwdriver.html.

7.12 Restrict NFS client requests to privileged ports

Action:

Add the `secure` option to all entries in the `/etc/exports` file. The following Perl code will perform this action automatically.

```
perl -i.orig -pe \
        'next if (/^\s*#/ || /^\s*$/);
        ($res, @hst) = split(" ");
        foreach $ent (@hst) {
            undef(%set);
            ($optlist) = $ent =~ /\((.*?)\)/;
            foreach $opt (split(/,/, $optlist)) {
                $set{$opt} = 1;
            }
            delete($set{"insecure"});
            $set{"secure"} = 1;
            $ent =~ s/\(.*?\)//;
            $ent .= "(" . join(",", keys(%set)) . ")";
        }
        $hst[0] = "(secure)" unless (@hst);
        $_ = "$res\t" . join(" ", @hst) . "\n";' \
/etc/exports
```

Discussion:

Setting the `secure` parameter causes the NFS server process on the local system to ignore NFS client requests that do not originate from the privileged port range (ports less than 1024). This should not hinder normal NFS operations but may block some automated NFS attacks that are run by unprivileged users.

8 User Accounts and Environment

Note that the items in this section are tasks that the local administrator should undertake on a regular, ongoing basis—perhaps in an automated fashion via `cron`. The automated host-based scanning tools provided from the Center for Internet Security can be used for this purpose. These scanning tools are typically provided with this document, but are also available for free download from http://www.CISecurity.org/.

8.1 Block system accounts

Action:

```
for name in `cut -d: -f1 /etc/passwd`; do
  uid=`id -u $name`
  if [ $uid -lt 500 -a $name != 'root' ]; then
      /usr/sbin/usermod -L -s /dev/null $name
  fi
done
```

Discussion:

These accounts are non-human system accounts that should be made less useful to an attacker by locking them and setting the shell to a shell not in /etc/shells. They can even be deleted if the machines does not use the daemon/service that each is responsible for, though it is safest to simply deactivate them as is done here.

To deactivate them, lock the password and set the login shell to an invalid shell. `/dev/null` is a good choice because it is not a valid login shell, and should an attacker attempt to replace it with a copy of a valid shell the system will not operate properly.

8.2 Verify that there are no accounts with empty password fields

Action:

The command

```
awk -F: '($2 == "") { print $1 }' /etc/shadow
```

should return no lines of output.

Discussion:

An account with an empty password field means that anybody may log in as that user without providing a password at all. All accounts should have strong passwords or should be locked by using a password string like "`NP`" or "`*LOCKED*`".

8.3 Set account expiration parameters on active accounts

Action:

```
cd /etc
awk '($1 ~ /^PASS_MAX_DAYS/)  { $2="90" }
     ($1 ~ /^PASS_MIN_DAYS/)  { $2="7" }
     ($1 ~ /^PASS_WARN_AGE/)  { $2="28" }
     ($1 ~ /^PASS_MIN_LEN/)   { $2="6" }
     { print } ' login.defs > login.defs.new
/bin/mv login.defs.new login.defs
/bin/chown root:root login.defs
/bin/chmod 640 login.defs
for name in `cut -d: -f1 /etc/passwd`; do
  uid=`id -u $name`
  if [ $uid -ge 500 -a $uid != 65534 ]; then
     /usr/bin/chage -m 7 -M 90 -W 28 $name
  fi
done
```

Discussion:

It is a good idea to force users to change passwords on a regular basis. The commands above will set all active accounts (except the `root` account) to force password changes every 90 days, and then prevent password changes for seven days thereafter. Users will begin receiving warnings 28 days before their password expires. Sites also have the option of expiring idle accounts after a certain number of days (see the on-line manual page for the `usermod` command, particularly the `-f` option). Finally, the instructions above set a minimum password length of 6 characters.

These are recommended starting values, but sites may choose to make them more restrictive depending on local policies.

8.4 Verify no legacy '+' entries exist in `passwd`, `shadow`, and `group` files

Action:

The command

```
grep ^+: /etc/passwd /etc/shadow /etc/group
```

should return no lines of output.

Discussion:

'+' entries in various files used to be markers for systems to insert data from NIS maps at a certain point in a system configuration file. These entries may provide an avenue

for attackers to gain privileged access on the system, and should be deleted if they exist.

8.5 Verify that no UID 0 accounts exist other than `root`

Action:

The command

```
awk -F: '($3 == 0) { print $1 }' /etc/passwd
```

should return only the word "`root`".

Discussion:

Any account with UID 0 has superuser privileges on the system. The only superuser account on the machine should be the `root` account, and it should be accessed by logging in as an unprivileged user and using the `su` command (or equivalent) to gain additional privilege.

Finer granularity access control for administrative access can be obtained by using the freely-available `sudo` program (http://www.courtesan.com/sudo/).

8.6 No '.' or group/world-writable directory in root's `$PATH`

Action:

The automated testing tool supplied with this benchmark will alert the administrator if action is required.

Discussion:

Including the current working directory ('.') or other writable directory in root's executable path makes it likely that an attacker can gain superuser access by forcing an administrator operating as root to execute a Trojan horse program.

8.7 User home directories should be mode 750 or more restrictive

Action:

```
for dir in \
   `awk -F: '($3 >= 500) { print $6 }' /etc/passwd`
do
   /bin/chmod g-w $dir
   /bin/chmod o-rwx $dir
done
```

Discussion:

Group or world-writable user home directories may enable malicious users to steal or modify other users' data or to gain another user's system privileges. Disabling "read" and "execute" access for users who are not members of the same group (the "other" access category) allows for appropriate use of discretionary access control by each user. While the above modifications are relatively benign, making global modifications to user home directories without alerting the user community can result in unexpected outages and unhappy users.

8.8 No user dot-files should be world writable

Action:

```
for dir in \
   `awk -F: '($3 >= 500) { print $6 }' /etc/passwd`
do
    for file in $dir/.[A-Za-z0-9]*; do
        if [ ! -h "$file" -a -f "$file" ]; then
            /bin/chmod go-w "$file"
        fi
    done
done
```

Discussion:

World-writable user configuration files may enable malicious users to steal or modify other users' data or to gain another user's system privileges. While the above modifications are relatively benign, making global modifications to user home directories without alerting the user community can result in unexpected outages and unhappy users.

8.9 Remove user `.netrc` files

Action:

```
for dir in `cut -f6 -d: /etc/passwd`
do
   /bin/rm  -f $dir/.netrc
done
```

Discussion:

`.netrc` files may contain unencrypted passwords which may be used to attack other systems. While the above modifications are relatively benign, making global modifications to user home directories without alerting the user community can result in unexpected outages and unhappy users.

8.10 Set default umask for users

Action:

```
cd /etc
for file in profile csh.login csh.cshrc bashrc
do
  if [ `egrep -c umask\.\*77 $file` -eq 0 ];
  then
     echo "umask 077" >> $file
  fi
  /bin/chown root:root $file
  /bin/chmod 444 $file
done
cd /root
for file in .bash_profile .bashrc .cshrc .tcshrc
do
  echo "umask 077" >>$file
  /bin/chown root:root $file
done
```

Discussion:

With a default `umask` setting of `077`, files and directories created by users will not be readable by any other user on the system. The user creating the file has the discretion of making their files and directories readable by others via the `chmod` command. Users who wish to allow their files and directories to be readable by others by default may choose a different default `umask` by inserting the `umask` command into the standard shell configuration files (`.profile`, `.cshrc`, etc.) in their home directories. A `umask` of `027` would make files and directories readable by users in the same Unix group, while a `umask` of `022` would make files readable by every user on the system.

We adjust root's `umask` setting separately in this item, as root shells don't necessarily read the system-wide configuration files. For example, root sessions using bash don't appear to get `umask` settings from `/etc/profile`.

8.11 Disable core dumps

Question:

Do you have developers who need to debug crashed programs or send low-level debugging information to software developers/vendors?

If the answer to this question is yes, then **do not** perform the action below.

Action:

```
cat <<END_ENTRIES >>/etc/security/limits.conf
*               soft    core            0
*               hard    core            0
END_ENTRIES
```

Discussion:

Core dumps can consume large amounts of disk space and may contain sensitive data. On the other hand, developers using this system may require core files in order to aid in debugging. The `limits.conf` file can be used to grant core dump ability to individual users or groups of users.