FreeBSD Benchmark v1.0.5

(FreeBSD 4.10 and above)

Copyright 2001-2004, The Center for Internet Security

Edited by Tom Rhodes

Send feedback to:

Freebsd-feedback@lists.cisecurity.org

http://www.CISecurity.org/

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 or any CIS Party has or will have any liability to use 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 right; 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 have 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 of 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 or 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. Will 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 attorney's 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 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/ (http://nsa2.www.conxion.com/cisco/notice.htm)).

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

CIS hereby grants each CIS Security Conuslting 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 the terms of these Agreed Terms of Use, the right to distribute the Products and Recommendations within such Member's own oganization, whether by manual or electronic means. Each such Member acknowldges 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, and 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 purpose 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)

Table of Contents

I. CIS FreeBSD Benchmark

1. 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 to desktop systems in a large enterprise environment. These shaded items may be skipped on these desktop platforms.

2. 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 `/sbin/sh` shell and without `noclobber` set.

3. Executing Actions

The actions listed in this document are written with 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 they may be copied directly from this document into a `root` shell window with a “cut-and-paste” operation.

4. 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.

5. 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:

```
cp -r /etc /etc.old

mv /etc/rc.conf /etc/rc.conf.preCIS

for x in hostname nisdomainname dhclient firewall \
filter pf route gateway atm static ifconfig;
do grep $x /etc/rc.conf.preCIS >> /etc/rc.conf;
done;
```

Chapter 1. Patches and Additional Software Configuration

1.1. Apply the latest OS patches

Action:

Extract the version of your FreeBSD release from the CD by entering the following command:

```
/sbin/mount /cdrom
```

```
sh /cdrom/src/src-install.sh all
```

Install the latest version of `CVSup` from the FreeBSD `FTP` server:

```
pkg_add -r cvsup-without-gui
```

Create the required SUP file for the installed release:

> **Note:** The “XX” in the following examples designates a local `CVSup` host. Please use one of the mirror servers listed at http://www.FreeBSD.org/handbook/cvsup.html#HANDBOOK-MIRRORS-CHAPTER-SGML-MIRRORS-PRIMARY-CVSUP

Action (FreeBSD 4.10):

```
cat <<EOF>> /root/src
         *default host=cvsupXX.FreeBSD.org
         *default prefix=/usr/
         *default base=/usr/local/etc/cvsup
         *default release=cvs
         *default tag=RELENG_4_10
         *default delete use-rel-suffix compress
         src-all
         EOF
```

Action (FreeBSD 4.11):

```
cat <<EOF>> /root/src
         *default host=cvsupXX.FreeBSD.org
         *default prefix=/usr/
         *default base=/usr/local/etc/cvsup
         *default release=cvs
         *default tag=RELENG_4_11
         *default delete use-rel-suffix compress
         src-all
         EOF
```

Action (FreeBSD 5.3):

```
cat <<EOF>> /root/src
```

```
         *default host=cvsupXX.FreeBSD.org
         *default prefix=/usr/
         *default base=/usr/local/etc/cvsup
         *default release=cvs
         *default tag=RELENG_5_3
         *default delete use-rel-suffix compress
         src-all
         EOF
```

Action (FreeBSD 5.4):

```
cat <<EOF>> /root/src
         *default host=cvsupXX.FreeBSD.org
         *default prefix=/usr/
         *default base=/usr/local/etc/cvsup
         *default release=cvs
         *default tag=RELENG_5_4
         *default delete use-rel-suffix compress
         src-all
         EOF
```

Create the required directory and begin grabbing the patch files for FreeBSD:

```
mkdir -p /usr/local/etc/cvsup/sup
```

```
/usr/local/bin/cvsup -g /root/src
```

Finally rebuild and reinstall all the components required for FreeBSD to run with the latest security patches. It usually only requires a simple run of the `make` command as specified in the FreeBSD Security Advisory. If this is the first time a system is patched then all advisories should be checked or a `make buildworld && make installworld` should be issued in the `/usr/src` directory.

Discussion:

The operating system should be promptly patched after a security hole is located. This can be an extremely cumbersome process for FreeBSD but the steps above should ensure that the latest security patches are applied to FreeBSD. The first command is optional, but it will save the administrator a great deal of trouble to issue it; however, it's only required once on each FreeBSD system to extract the base system.

Administrators may wish to subscribe to the FreeBSD administrators security advisories list by visiting: http://lists.freebsd.org/mailman/listinfo/freebsd-security-notifications/

> **Caution**: The recommended `make buildworld buildkernel installkernel installworld` should only be used when a large amount of security patches were applied as it opens the door for *foot-shooting*. Administrators who feel that their system requires this update method should review the instructions in the FreeBSD handbook at: http://www.freebsd.org/doc/handbook/makeworld.html.

1.2. Enable `SSH`

Action:

```
awk '/^#Protocol/     { $2 = "2" };                   \
/^#Protocol/            { $1 = "Protocol" };            \
/^#PermitRootLogin/     { $1 = "PermitRootLogin};       \
/^#Banner/              { $2 = "/etc/motd" };           \
/^#Banner/              { $1 = "Banner" }               \
{ print }' /etc/sshd/sshd_config > /etc/sshd/sshd_config.new

mv /etc/ssh/sshd_config.new /etc/ssh/sshd_config

chmod 600 /etc/ssh/sshd_config
```

Finally, enable `sshd` on system boot by issuing the following command:

```
echo 'sshd_enable="YES"' >> /etc/rc.conf
```

Discussion:

`Openssh` is a freely distributed version of the popular Secure Shell package and provides for secure remote logins and file transfers by using encryption. FreeBSD includes this package, in the default installation but with support for both protocols one (1) and two (2). The previous command sets protocol two (2) as the default which will support the DSA encryption algorithm along with RSA during the public key authentication attempt. Protocol two (2) also supports more encryption mechanisms such as `3DES` and `Blowfish`; see the manual page for more information. The aforementioned command also enables support for the login banner, or the message of the day (`motd`) which is set here to use the `/etc/motd` file. We will provide instructions on banners in Section 7.5.

1.3. Enable `TCP Wrappers` and a host based firewall

Ensure that `hosts.allow` exists in `/etc` by issuing the following command:

```
ls /etc/hosts.allow
```

FreeBSD packages `TCP Wrappers` as contributed software; thus it is installed by default. To enable it with `inetd`, issue the following command:

```
cat <<EOF>> /etc/rc.conf
inetd_enable="YES"
inetd_flags="-Wwl -C60"
EOF
```

The `TCP Wrappers` software provides an administrator with the ability to enhance security in several ways. For instance, `TCP Wrappers` can choose who will or will not have network access to certain network daemons based on an accept/deny policy. The configuration of such policies are set in `/etc/hosts.allow` and the accompanying hosts_access(5) manual page.

> **Note:** Unlike most UNIX™ environments, the `hosts.deny` has been deprecated in FreeBSD for a few years now. Everything is configured in the `hosts.allow` file. See the FreeBSD Handbook for more information.

In almost every case, it would be wise to enable and configure a firewall. With regards to `TCP Wrappers` they may be used together for added security, permitting several different actions to take place depending on connection, reply to services, etc. To enable the firewall during every system initialization, perform the following actions:

```
echo 'ipfw_load="YES"' >> /boot/loader.conf
```

Select a policy:

Loading the firewall module will offer a default to deny policy. All Internet connections will be blocked regardless of where the connection originates. The next item will discuss configuring a default policy.

```
echo 'firewall_enable="YES"' >> /etc/rc.conf

echo 'firewall_type="open"' >> /etc/rc.conf
```

Discussion:

Now the firewall script will be run at system initialization, but an administrator must now choose a firewall type and configure accordingly. FreeBSD includes an `open` for a completely open configuration; a `client` which will attempt to protect this machine; a `simple` which will attempt to protect the network; a `closed` which will close off all connections other than those to the `loopback` interface. Either selection will load the respected policy from the `rc.firewall` file and load it into the kernel module.

There is an opportunity to load a customized firewall script, see the `rc.conf` file for more information on that option and corresponding firewall options.

Chapter 2. Minimize `inetd` Services

2.1. Disable all `inetd` daemons

Action:

```
cp /usr/share/examples/etc/inetd.conf /etc
```

Discussion:

The Internet “super server” listens for connections on various sockets. Once a connection has been established, `inetd` will invoke the appropriate daemon to service the request. The stock `inetd.conf` in FreeBSD has everything disabled by default unless modified by the system administrator or modified during the installation. The command above will reinstall the stock configuration file thus setting all services to disable. If the original configuration ever needs restored, this can be done by copying the `/etc.old/inetd.conf` file back into `/etc`.

2.2. Only enable `telnetd` if absolutely necessary

Question:

Is there a mission-critical reason that requires users to access this system via `telnet` in place of the secure `SSH` protocol?

If the answer to this question is yes, then proceed by issuing the command below:

Action:

```
sed -i .preCIS -e 's/#telnet/telnet/g' /etc/inetd.conf
```

Discussion:

The `telnetd` permits users to log into a remote host and access a shell. The authentication and data transfer method is plain text, a method subject to attack by a malicious user. A hacker could sniff packets in transfer and even hijack the connection. When possible the `SSH` server should be used instead.

2.3. Only enable `ftpd` if absolutely necessary

Question:

Is or will this machine be an anonymous FTP server? Or is there any reason data should be transferred via `ftp` in place of the more secure features provided as part of `OpenSSH` like `scp` and `sftp`?

If the question to the previous question is yes, then proceed by issuing the commands below:

Action:

```
 awk '/^#ftp/ { $1 = "ftp" }; \
/ftp/ { $8 = "-sll" } \
{ print }' /etc/inetd.conf >> /etc/inetd.conf.new

 mv /etc/inetd.conf.new /etc/inetd.conf
```

Discussion:

The ftp server works similar to telnet as all the data is passed as clear text; thus is susceptible to network attacks. The OpenSSH package includes a Secure Copy protocol (scp) and a Secure File Transfer Protocol (sftp) which should be considered minimal for non-anonymous users.

For anonymous ftp access, the system will require an account for the ftp user which should have the shell set to nonexistant. The home directory structure should include a pub, bin, and etc. The etc directory should include a copy of the passwd and group files with a welcome banner named ftpmotd.

2.4. Only enable rlogin/rsh/rcp if absolutely necessary

Question:

Does any mission-critical reason exist to have rlogin, rsh, and rcp in place over the more secure features provided by OpenSSH?

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

Action:

```
 sed -i .preCIS -e 's/#shell/shell/g; s/#login/login/g' \
/etc/inetd.conf
```

Discussion:

FreeBSD provides these protocols as a means of connecting with older machines or machines which cannot make use of the SSH replacements. In this day and age of computing, there should be no real need to enable these in place of OpenSSH.

2.5. Only enable TFTP if absolutely necessary

Question:

Is this system a boot server for the network, or is there some other mission-critical reason why data should be transferred via TFTP?

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

Action:

```
 awk '/^#tftp/ { $1 = "tftp" }; \
/tftp/ { $7 = "-n" } \
{ print }' /etc/inetd.conf >> /etc/inetd.conf.new

sed -i .preCIS -e 's/#tftp/tftp/g' /etc/inetd.conf

mkdir -m 777 /tftpboot
```

Discussion:

The TFTP is typically used for diskless workstations, CISCO™ Routers, and similar devices. This allows these devices, most without hard disk drivers, to connect to remote systems and copy configuration files, or perhaps do back ups; however, unless TFTP has any use on the network, it should be disabled. Adding -n will prevent the logging of nonexistant file name requests.

2.6. Only enable finger if absolutely necessary

Question:

Is there a mission-critical reason this system would ever need to offer access to user specific .plan files either internally or externally?

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

Action:

```
awk '/^#finger/ { $1 = "finger" }; \
/finger/ { $7 = "-l" } \
{ print }' /etc/inetd.conf >> /etc/inetd.conf.new

mv /etc/inetd.conf.new /etc/inetd.conf
```

Discussion:

Some sites utilize the finger daemon to access user specific .plan files. These files are sometimes used to store PGP keys, current project information, etc. When invoked on a user, the finger utility will return that user's information which could include their phone number, name, address, where their email is forwarded to, etc. The returned information varies depending on what that user has entered. When possible, the -l and -s flags should be passed to enable logging and secure mode.

2.7. Only enable Kerberos-related daemons if absolutely necessary

The following only applies to FreeBSD 4.X and 5.0 releases. The 5.2 release removed them.

Question:

Is the Kerberos security system in use at this site?

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

Action (FreeBSD 4.X and 5.0):

```
sed -i .preCIS -e 's/#klogin/klogin/g; \
s/#eklogin/eklogin/g; \
s/#kshell/kshell/g; \
s/#kip/kip/g' /etc/inetd.conf
```

Action (FreeBSD 5.1 and later):

```
cat<<EOF>>/etc/rc.conf
kerberos5_enable="YES"
kadmind5_server_enable="YES"
kpasswdd_server_enable="YES"
EOF
```

Discussion:

`KerberosIV` support in the base system was deprecated in favor of `Kerberos5 (Heimdal)` with the release of FreeBSD 5.1. It is still available as part of the ports collection (`ports/security/krb4`, and added with `pkg_add -r krb4`), but not as part of the installation. See the following link for more information: http://web.mit.edu/Kerberos/www/.

2.8. Minimize the `inetd.conf` file

Action:

```
mv /etc/inetd.conf /etc/inetd.conf.new

grep -v '#' /etc/inetd.conf.new > /etc/inetd.conf

chown root:wheel /etc/inetd.conf && chmod 444 /etc/inetd.conf
```

Discussion:

FreeBSD provides a “stock” `inetd.conf` configuration file with commented out options. Some scripts have been known to automatically enable `inetd` services by removing the comment character from entries that are normally commented out. By purging virtually all of the unused services from the configuration file, it will be easier for an administrator to notice newly added services during auditing periods.

> **Note:** The original `inetd.conf` file still exists as `inetd.conf.new` should any service need re-enabled by the administrator. Just edit the `inetd.conf.new` file and re-run the aforementioned `grep` command.

Chapter 3. Minimize boot services

3.1. Disable `login` prompts on serial ports

The default FreeBSD configuration does not permit `login` on serial ports. Ensure they are disabled by issuing the following command:

Action:

```
grep dialup /etc/ttys | grep on
```

If the word `on` shows up anywhere on the screen, please set it to off with:

```
awk '($4 == "dialup") { $5 = "off" } { print }' /etc/ttys > /etc/ttys.new
```

```
mv /etc/ttys.new /etc/ttys
```

Discussion:

To prevent malicious users from connecting terminals and other devices to the serial ports, the `login` prompt should be set to disabled.

3.2. Set password on single user console

Action:

```
awk '($1 == "console") { $5 = "insecure" } { print }' /etc/ttys > /etc/ttys.new
```

```
mv /etc/ttys.new /etc/ttys
```

Discussion:

When the system is rebooted due to power failure or otherwise, administrators can issue the `-s` flag to cause a single user mode boot. When the system boots into single user mode, they get prompted with an unprotected root shell. So to protect the system from unauthorized access in this manner, the above command sets the console to insecure, ultimately requiring the `root` password to be entered before the system may be accessed.

3.3. Set daemon umask

Action (FreeBSD 4.X):

```
find /etc/ /usr/local/etc/rc.d | xargs grep 'umask'
```

Action (FreeBSD 5.X):

```
find /etc/ /usr/local/etc/rc.d/ | xargs grep 'umask'
```

Discussion:

All daemons should run with an 022 umask setting, this will prevent their processes from creating world-writable files by default. The default setting for FreeBSD is always 022, and in some extremely rare cases (such as scripts in the /etc/periodic directory) the more restrictive 077 umask will be used. The commands above will reveal all current umask settings. To modify any umask setting which differs from the above, issue the following command:

```
sed -i .preCIS -e 's/XXX/022/g' FILE
```

Where XXX is the current umask setting and FILE is the file with the offending umask setting.

3.4. Prevent syslogd from accepting messages from the network

Question:

Is this machine a log server or does it, for any reason, need to receive syslogd messages from other machines over the network?

Action:

```
echo 'syslogd_flags="-s"' >> /etc/rc.conf
```

Discussion:

By default, the system logging daemon known as syslogd will listen for log messages on port 514/udp. This is done without any authentication and thus is susceptible to denial of service attacks. A malicious user may also abuse this ability to fill up log files to such an extent that subsequent attacks may either be unnoticeable or not logged at all. By adding the -s to syslogd's startup options, we hinder the ability for syslogd to interact with the network all together.

> **Note:** It is considered common and good practice to set up one or more machines as a central log server; however, unless this machine is a log server for the network, syslogd should not have the ability to listen for incoming log messages.

3.5. Disable the email server if possible

Question:

Is this system an email server for other hosts on the network or over the Internet?

If the answer to this question is yes, then ***do not*** proceed with the actions below.

Action (FreeBSD 5.X, 4.X):

```
echo '"sendmail_enable="NONE"' >> /etc/rc.conf
```

Action (FreeBSD 5.1 and later):

```
cat<<EOF>> /etc/rc.conf
sendmail_enable="NO"
sendmail_submit_enable="NO"
sendmail_outbound_enable="NO"
sendmail_msp_queue_enable="NO"
EOF
```

Discussion:

FreeBSD offers the ability to disable Sendmail from listening for remote network connections without limiting the use of email services to users. In cases where Sendmail needs to run in other modes, the available options are listed in the rc.sendmail manual page and can be added to the rc.conf file in a method similar to the above.

> **Note:** If the system is an email server, the administrator is encouraged to read over the various amounts of security documentation written about Sendmail. The FreeBSD handbook is a good starting point and is located at: http://www.FreeBSD.org/handbook/. Another good place to visit is the Sendmail homepage: http://www.sendmail.org/.

3.6. Only enable BIND if absolutely necessary

Question:

Does this machine handle DNS requests for the local network nor the Internet?

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

Action:

```
echo 'named_enable="YES"' >> /etc/rc.conf
```

Discussion:

The BIND DNS server maps IP addresses to hostnames across the Internet and supplies these services to other hosts on the local local network. Though it has been widely implemented, BIND has a long history of security flaws. Most administrators implement BIND in a sandbox, better known as a chroot environment, for added security. See the jail and chroot manual pages for more information on implementing this feature. The FreeBSD handbook (http://www.FreeBSD.org/handbook/) has instructions on this design which are specific to FreeBSD.

> **Note:** FreeBSD 5.3 and post 5.3 releases include BIND9 by default. The boot up option has remained the same; however, the server will automatically be placed in a chroot environment. The 4.X series must have the chroot configured manually.

3.7. Only enable other RPC-based services 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 Kerberos security system is in use at this site.*
- *This machine delivers boot information to other machines on the network.*
- *This machine runs third-party software or some other utility which requires RPC support.*

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

Action (FreeBSD 4.X):

```
cat<<EOF>> /etc/rc.conf
rpc_lockd_enable="YES"
rpc_statd_enable="YES"
portmap_enable="YES"
EOF
```

Action (FreeBSD 5.X):

```
cat<<EOF>> /etc/rc.conf
rpc_lockd_enable="YES"
rpc_statd_enable="YES"
rpcbind_enable="YES"
EOF
```

Discussion:

`RPC` based services usually incorporate very weak or sometimes non-existant authentication and may be open for attack. Unless there is an absolute need to run these processes then they should be left disabled. If it is unknown whether or not any third-party software requires `RPC` services the documentation for that software should be reviewed.

3.8. Only enable the `NFS` server if absolutely necessary

Question:

Is this machine an `NFS` file server?

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

Action (FreeBSD 4.X):

```
cat<<EOF>>/etc/rc.conf
nfs_server_enable="YES"
single_mountd_enable="YES"
EOF
```

Action (FreeBSD 5.X):

```
cat<<EOF>>/etc/rc.conf
nfs_server_enable="YES"
mountd_enable="YES"
EOF
```

Discussion:

`NFS` servers are commonly used to share data between machines at rapid rate; however, this service is often exploited to gain unauthorized access to files and external file systems. There is no reason to run these services if they have no use in the network. When `NFS` is enabled, the administrator should take reasonable precautions to ensure that exports are properly secured by limiting access from certain `IP`s, using the “read-only” and “nosuid” options, and make use of other security options available. See the `nfsd` and `rc.conf` manual pages for more information.

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 systems via NFS?

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

Action:

```
echo 'nfs_client_enable="YES"' >> /etc/rc.conf
```

Discussion:

Unless there is a significant need to acquire data on remote partitions via NFS, an administrator should just disable NFS related daemons. This does not alleviate the ability to transfer files between networked machines. This ability can be provided through utilities such as scp which is part of the OpenSSH package.

3.10. Block NFS connections to non-privileged ports

Action:

```
echo 'nfs_reserved_port_only="YES"' >> /etc/rc.conf
```

Discussion:

This option will force the NFS server to ignore requests on ports above the “privileged port range” (i.e. ports fewer than 1024). Setting this option should not affect NFS operations in any way; indeed, it may block some NFS attacks which are run by unprivileged users.

3.11. Block non-privileged mountd requests

Action:

```
echo 'weak_mountd_authentication="NO"' >> /etc/rc.conf
```

Discussion:

The mountd server is accessed by NFS clients to handle remote mount requests. Some services such as PCNFSD will make what are considered non-privileged mount requests. Because these requests are not authenticated in any way they pose as a potential security risk. Setting this option will force mountd to ignore these requests.

3.12. Only enable NIS if absolutely necessary

Question:

Will this machine act as an NIS (YP) server for the network?

If the answer to this question is yes, issue the following command:

Action:

```
cat<<EOF>> /etc/rc.conf
nis_server_enable="YES"
nis_ypxfrd_enable="YES"
nis_yppasswdd_enable="YES"
rpc_ypupdated_enable="YES"
EOF
```

Discussion:

The NIS system is frequently implemented in large networks to cut down administration overhead for multiple machines. Yet like any other network are open to attack from malicious users. Unless the NIS service is in use, disabling it is recommended.

3.13. Only enable NIS client daemons if absolutely necessary

Question:

Will this machine utilize services offered by a machine acting as an NIS server for the network?

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

Action:

```
cat<<EOF>> /etc/rc.conf
nis_client_enable="YES"
nis_ypset_enable="YES"
EOF
```

Discussion:

The daemons permitting NIS logins are of no use on an NIS client, and as such they should be disabled.

3.14. Only enable the printer daemons if absolutely necessary

Question:

Does this system act as 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, proceed with the actions below:

Action:

```
echo 'lpd_enable="YES"' >> /etc/rc.conf
```

Discussion:

If the users of this machine have no need to print files either from this machine either locally or via the network, then it is safe to disable all print services. To ensure proper configuration, administrators should review the printcap(5) manual page. The administrator may wish to check out the LPRng print system (see http://www.lprng.org (http://www.lprng.org/) and the FreeBSD port located in /usr/ports/sysutils/LPRng, which can be installed with `pkg_add -r LPRng`, which was designed with security in mind. Administrators may also wish to check out the `CUPS` printing software in `/usr/ports/print/cups` installed with the `pkg_add -r cups` command.

Chapter 4. Kernel Tuning

4.1. Disable core dumps

Action:

```
echo 'kern.coredump=0' >> /etc/sysctl.conf
```

Discussion:

Core dumps may contain sensitive data while consuming a large amount of disk space; however, if any of the local users are developing software, they may require core files for debugging purposes. For these cases, the `/etc/login.conf` file may be more beneficial. Review the manual page for `login.conf` to get an idea of the available options.

4.2. Set a default secure level

Action:

```
echo 'kern.securelevel=1' >> /etc/sysctl.conf
```

Discussion:

FreeBSD offers a securelevel feature which will set a default system security profile. Setting this to a value of one (`1`) will set the system immutable and system append-only flags on files (see the `chflags` manual page). These flags cannot be turned off once this is set, and certain devices, for instance `/dev/mem`, may not be opened for writing.

> **Caution**: Under the above setting modules may not be loaded or loaded into the running kernel. The securelevel may not be lowered once raised to a higher value without a system reboot. This may make it difficult for an administrator to patch a system because they will need to enter single user mode (`shutdown now`) before replacing any utilities.

4.3. Block users from viewing unowned processes

OS Revisions:

The following action only applies to 5.X systems.

Action:

```
echo 'security.bsd.see_other_uids=0' >> /etc/sysctl.conf
```

Discussion:

While it can be argued that this is a bit extreme, removing the ability for users to gather information about processes they do not own can help prevent exploits for specific daemons.

4.4. Block users from viewing processes in other groups

OS Revisions:

The following action only applies to 5.X systems.

Action:

```
echo 'security.bsd.see_other_gids=0' >> /etc/sysctl.conf
```

Discussion:

Blocking users from obtaining information on process running under different group IDs may prevent exploits for specific utilities and/or daemons.

Chapter 5. Logging

5.1. Capture ftpd and inetd information

OS Revisions:

FreeBSD 5.X has this feature enabled by default. Although the log file is called debug.log.

Action (FreeBSD 4.X):

```
printf "daemon.debug\t\t\t\t/var/log/daemon.log\n" \
>> /etc/syslog.conf

touch /var/log/daemon.log

chown root:wheel /var/log/daemon.log

chmod 600 /var/log/daemon.log
```

Discussion:

If the FTP daemon was enabled as stated in Item 2.3, then all debugging (-d) and connection information (-l) will be logged. Thus the syslog daemon will capture the appropriate information to a file.

Log files for the appropriate FreeBSD release should be reviewed and archived regularly. Both releases support the newsyslog utility which will rotate and manage system log files on a regular basis.

5.2. Enable system accounting

Action:

```
touch /var/account/acct

accton /var/account/acct

echo 'accounting_enable="YES"' >> /etc/rc.conf
```

Discussion:

System accounting will keep track of system data including CPU utilization, disk I/O, what users are doing on the system, etc. The collected statistics can be viewed with the sa command. When this feature is enabled, the administrator is advised to read over the features provided by sa as the gathered information can grow at an extremely rapid rate.

> **Note:** Accounting information is purged on a regular basis by the FreeBSD default cron jobs. Administrators may wish to archive this information in another location for data preservation.

5.3. Enable logging of packets received on closed ports

Action:

```
cat<<EOF>>/etc/sysctl.conf
net.inet.tcp.log_in_vain=1
net.inet.udp.log_in_vain=1
EOF
```

Discussion:

The log in vain option will log requests to closed ports. Failed connection attempts will be logged to the `/var/log/messages` file for administrator review. This will permit an administrator to see connection attempts on closed or restricted ports.

5.4. Set permissions on system log files

Action:

```
chmod g-w,o-r /var/log/* && chmod a+r /var/log/wtmp
```

Discussion:

Users should never have a reason to modify system log files and certain log files may contain sensitive data which should only be viewed by the system administrator.

5.5. Configure `newsyslog` for secure file permissions

Action:

```
sed -i .preCIS -e 's/644/600/g; s/640/600/g' /etc/newsyslog.conf &&\
awk '($1 == "/var/log/wtmp") { $4 = "644" }; \
($1 == "/var/log/lastlog") { $4 = "644" } { print }' \
/etc/newsyslog.conf > /etc/newsyslog.conf.new &&\
mv /etc/newsyslog.conf.new /etc/newsyslog.conf
```

Discussion:

The permissions set in the previous item are futile as the `newsyslog` utility will reset them during the next rotation. By making modifications to the log file permission settings in `/etc/newsyslog.conf`, they will continue to be set correctly.

5.6. Configure `periodic` log files

Action:

```
cat<<EOF>> /etc/periodic.conf
    daily_output=/var/log/daily.log
    EOF

echo "/var/log/daily.log root:wheel      600  7     100  @T23    C"
```

Discussion:

By default, FreeBSD does daily checks which are normally run early in the morning of `localtime`. These checks are also normally sent to the `root` alias via `Sendmail`; however, in cases where sendmail may not be running due to security reasons, these daily security checks should be dropped into a log file. Here, we will not only log to the `daily.log` log file, but rotate it every day at the twenty third (23rd) hour.

Chapter 6. File/Directory Permissions/Access

6.1. Add `nosuid` option to `/etc/fstab`

Action:

```
awk '/cdrom/ { $4 = "nosuid,ro,noauto"} {print}' /etc/fstab >> /etc/fstab.new
```

```
mv /etc/fstab.new /etc/fstab
```

Discussion:

It is possible; however, unlikely that malicious software may be introduced by removable media. By forcing the `nosuid` option on these file systems, set-UID programs will not be introduced from removable media via CD-ROMs.

6.2. Verify passwd, master.passwd, and group file permissions

Action:

```
chown root:wheel /etc/passwd /etc/master.passwd /etc/group /etc/pwd.db /etc/spwd.db
```

```
chmod 644 /etc/passwd /etc/group /etc/pwd.db
```

```
chmod 600 /etc/master.passwd /etc/spwd.db
```

Discussion:

Unlike many UNIX™ variants, FreeBSD has a `/etc/master.passwd` file in place of the usual `/etc/shadow` file. There should be no reason for users to view the `/etc/master.passwd` file; thus permissions are set accordingly. The password database files also have their permissions changed accordingly. More information about the database format can be found in the pwd_mkdb(8) manual page.

6.3. Set `sticky bit` on world writable directories

Action:

```
for FS in `awk '($3 == "ufs")  { print $2 }' /etc/fstab`; do \
     find -x -f $FS \( -type directory -perm -0002 -a ! -perm -1000 \) -print; done;
```

Discussion:

The “sticky” prevents the removal of files by users who do not own them when set on a directory. The previous command will print a list of all directories which do not have this permission bit set, if you are sure all of these directories have mode 777 set, you can add the “sticky” bit on them with:

```
for FS in `awk '($3 == "ufs")  { print $2 }' /etc/fstab`; do \
    find -x -f $FS \( -type directory -perm -0002 -a ! -perm -1000 \) -exec chmod 1777 {} \; ;
done;
```

This command will not modify the current 777 mode, but apply the “sticky” bit or modify the directory to have mode 777. Thus it should be used with caution.

6.4. Find world writable files

Action:

```
for FS in `awk '($3 == "ufs") { print $2 }' /etc/fstab`; do \
find -x -f $FS \( -type file -perm -0002 -a ! -perm -1000 \) -print;
done;
```

Discussion:

World writable files have the potential to leak sensitive information, or have that information manipulated by malicious users. The only files which should appear in this list are those located in the `/tmp` directory.

6.5. Find SUID and SGID files

Action:

```
for FS in `awk '($3 == "ufs") { print $2 }' /etc/fstab`; do \
find -x -f $FS \( -type file -perm -04000 -o -perm -02000 \) -print;
done;
```

Discussion:

Executable files with the SUID or SGID bit set will run with effective UID/GID of the utility owner. The previous command will print a list of the offending files for the administrator. The administrator should then take measures to review the security implications of leaving these utilities SUID/SGID.

6.6. User home directories should be kept private

Action:

```
for x in `awk -F: '($3 >= 1001) && ($3 != 65534) {print $6}' /etc/passwd`;
do chmod -H 0700 $x; \
done;
```

Discussion:

At times it is good practice to keep the home directories private. This will block users from viewing files located in home directories owned by other users. Users will be required to implicitly allow files to be viewed by everyone.

> **Warning**: Making a modification such as this without any notice could make users irate. It is recommended this action be done with caution. A more realistic approach would be to use `chmod` with the `u=rwx,g=rwx,o-r` flags on all home directories. This will permit their public_html directories to be visible on the Internet but block local users from getting a directory listing.

6.7. Find “Unowned” Files and Directories

Action:

```
find / \( -nouser -o -nogroup \) -print
```

Discussion:

At times, an administrator must remove users from the system. On occasion, not all files owned by that user are removed. As such, when a new user inherits a formally used UID/GID they could end up owning a previous users files. This could provide the user more access then originally intended, permit the sharing of assumed confidential data, etc. Stale files such as these should be either removed or have a new owner assigned manually.

Chapter 7. System Access, Authentication, and Authorization

7.1. Remove weak authentication services from PAM

Action (FreeBSD 4.X):

```
printf "rexecd\tauth\trequired\tpam_deny.so\n" >> /etc/pam.conf

printf "rsh\tauth\trequired\tpam_deny.so\n" >> /etc/pam.conf
```

Action (FreeBSD 5.X):

```
sed -i .preCIS -e 's/nologin/deny/g' /etc/pam.d/rsh /etc/pam.d/rexecd
```

Discussion:

It is recommended to remove support for services which utilize weak authentication mechanisms, such as rsh and rexecd.

> **Note:** FreeBSD 5.X has a completely different PAM configuration than 4.X, dropping the pam.conf file and storing all the files in /etc/pam.d.

7.2. All .rhosts files should be readable only by their owner

Question:

Is the rlogin, rsh, or rcp services employed on this network?

If the answer to the previous question is yes, proceed with actions in the following two sections:

Action:

```
find / -type file -name '.rhosts' | xargs chmod 600
```

Discussion:

Setting the mode for all .rhosts files to 600 will ensure that only the owner will have read and write capabilities to them. When using the “r-services”, be sure to take the proper security precautions; such as using “trustedhost hostname” in place of just “trustedhost”.

7.3. Symlink hosts.equiv to /dev/null

Action (FreeBSD 4.X):

```
rm /etc/hosts.equiv && ln -s /dev/null /etc/hosts.equiv
```

Action (FreeBSD 5.X):

```
grep -v 'pam_rhosts' /etc/pam.d/rsh > /etc/pam.d/rsh.new
```

```
mv /etc/pam.d/rsh.new /etc/pam.d/rsh
```

Discussion:

The `/etc/hosts.equiv` file enables a weak form of access control based on hostname or host address which can be spoofed by an attacker. Creating a symlink for this file to `/dev/null` will help prevent attackers from adding data to it. In FreeBSD 5.X, removing the existence of the `pam_rhosts` from `PAM` configuration files will make render `hosts.equiv` useless.

7.4. Restrict `at/cron` to authorized users

Action:

```
echo 'root' > /var/cron/allow
```

```
echo 'root' > /var/at/allow
```

```
chown root:wheel /var/cron/allow /var/at/allow
```

```
chmod 400 /var/cron/allow /var/at/allow
```

```
chmod 0640 /etc/crontab
```

Discussion:

If the `cron/allow` and `at/allow` files exist then only the users listed in those files will be granted access to the `crontab` and `at` utilities. Using this method in place of creating the `deny` file should cut administration overhead down as new users will already be denied access to these utilities.

> **Note:** Most system administrators schedule jobs to run at given intervals. In fact, FreeBSD includes several of them which do periodic dumps of system statistics and then mails them to the root user. The above settings will not hinder this usage, nor will it prevent `cron` from running jobs as different users (e.g. `daemon`).

7.5. Create warning banners for the system

Action:

```
rm /etc/motd
```

```
echo "Authorized users only.  All activity may be \
monitored and reported.  Use of this system implies the \
acceptance of such monitoring." >> /etc/motd
```

```
chmod 644 /etc/motd
```

Discussion:

Having a warning message displayed at login time may assist in the prosecution of trespassers on the computer system. Guidelines set forth by the United States Department of Defense require that warning messages contain at least the name of the organization that owns the system, the fact that the system is subject to monitoring, and that such monitoring is compliant with local statutes, and that use of the system implies consent to such monitoring. The organization's legal council and/or site security administrator should review the content of all messages before the aforementioned modifications are made.

Administrators may also want to read over the DoJ's banner website located at: http://www.usdoj.gov/criminal/cybercrime/s&sappendix2002.htm which contains information on how banners are used, should be worded, and different cases where banners have been brought up in court.

> **Note:** Countries other than the United States may have different legislatures governing such messages and prosecution methods. It is at the sole discretion of the organization/individual to implement this item.

7.6. Remove the `X wrapper` and enable `xdm`

Action:

```
pkg_delete -r wrapper-\*

sed -i .preCIS -e '/xdm -nodaemon/s/off/on/' /etc/ttys
```

Discussion:

The `wrapper` package offers the ability for users to use their own `startx` script or run `X` directly. To keep users from doing this the above actions will disable the use of `wrapper` scripts with `X`. These commands have the side effect of setting `xdm` to enabled, thus forcing an `X` login prompt and adding additional security.

7.7. Prevent `xdm` from listening on port 6000/TCP

Action:

```
sed -e '/^:/s/$/ -nolisten tcp/' </usr/X11R6/lib/X11/xdm/Xservers >/etc/X11/xdm/Xservers
```

Discussion:

By default, FreeBSD sets this as disabled; but it could always be changed. The command above will disable `xdm`'s ability to accept connections on port 6000; forcing administrators to enable the `xdm` utility.

> **Note:** This may not be a requirement as running `X11` on a server class system is extremely unusual and is not installed by default on FreeBSD. Workstation implementations; however, may have `X11` installed to provide a `GUI` which is where this command will be most useful.

Chapter 8. User Accounts and Environment

8.1. Block system accounts

Action:

```
pw moduser uucp -s /sbin/nologin
```

Discussion:

The only system account on FreeBSD without the shell set to `/sbin/nologin` is `uucp`. The above command will correct this.

> **Warning**: The `uucp` account should *not* be modified this way if `uucp` services will be used at this site.

8.2. Verify that accounts either have a password or are disabled

Action:

```
awk -F: '{ print $1 $2 }' /etc/master.passwd
```

Discussion:

The output from the aforementioned command should only produce two fields of output: a username and the password associated with that username. There should be no blank space after any of the usernames in the output. An asterisk character (`*`) designates a disabled account. Accounts without any password should either be disabled or removed using the `pw` utility.

> **Note:** On systems utilizing `NIS` services, the `ypcat` command should be used to print the contents of the master `password` file:
>
> ```
> ypcat passwd | awk -F: '{ print $1 $2 }'
> ```

8.3. Set account expiration parameters on all active user accounts

Action:

```
for x in `awk -F: '($3 >= 1001) && ($3 != 65534) { print $1 }' /etc/passwd`;
do pw usermod $x -e +91d; \
done;
```

Discussion:

It's a good idea to set expiration time on all active accounts in the system. The aforementioned command will create a default expiration time on all users to designate when the password should be changed. This will force a password change every 91 days (3 months).

8.4. Create default adduser.conf file

Action (FreeBSD 5.X):

```
cat<<EOF>> /etc/adduser.conf
# Configuration file for adduser(8).
# NOTE: only *some* variables are saved.
defaultLgroup=
defaultclass=default
defaultgroups=
passwdtype=yes
homeprefix=/home
defaultshell=/bin/csh
udotdir=/usr/share/skel
msgfile=/etc/adduser.msg
disableflag=
upwexpire=91d
uexpire=
EOF
```

Discussion:

Since the previous command will only set the expiration time on active accounts; future accounts added with the `adduser` utility will remain unaffected. Creating an `adduser.conf` file with the expiration time defined will ensure that future account additions with the `adduser` utility will hold true to the 91 day policy.

8.5. Remove the `toor` user.

Action:

```
pw deluser toor
```

Discussion:

It should be noted that FreeBSD adds a `toor` user by default with the UID of zero (0). This account should be removed from the system.

8.6. Verify that `root` is the only user with UID 0

Action:

The command:

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

should return only the word “`root`”

> **Note:** On systems utilizing `NIS` services, the `ypcat` command should be used to print the contents of the master `password` file:
>
> ```
> ypcat passwd | awk -F: '($3 == 0) { print $1 }'
> ```

Discussion:

Any account on the system that has a UID or GID of zero (0) will will have superuser privileges; thus, the only user who should have this UID is `root`. It is good practice to never login as `root` directly, but to login as a user who is in the group of `wheel` and using the `su` command.

Administrators may want better access control than provided by the `su` utility. For those cases, the `sudo` utility can be installed. See the website http://www.courtesan.com/sudo/, or install it using `pkg_add -r sudo`.

8.7. 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 "$file" ]; then\
chmod go-w "$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.

8.8. Set default umask for users

Action:

```
sed -i .preCIS -e 's/umask=022/umask=077/' /etc/login.conf

for x in /etc/profile /etc/csh.login /etc/csh.cshrc;
do echo "umask 077" >> $x; done;

find /usr/share/skel -name 'dot*' |\
xargs sed -i .preCIS -e 's/22/77/g'
```

Discussion:

Using a default `umask` setting of `077` will ensure that files and directories created by users cannot be viewed by their peers. This can be overwritten by users by changing the umask setting in their shell configuration files (`.cshrc`, `.profile`, etc). The final command will alter the default installed shell configuration files so that the addition of users in the future will not see these settings overwritten.

8.9. Set “mesg n” as the default for all users

Action:

```
sed -i .preCIS -e 's/#[[:space:]]mesg[[:space:]]y/mesg n/g' /etc/profile

sed -i .preCIS -e 's/#[[:space:]]mesg[[:space:]]y/mesg n/g' /etc/csh.login
```

Discussion:

Setting this will block attempts for users to use the `write` or `talk` utilities. These utilities permit users to carry on discussions over their terminals. This will also increase the security on tty devices by raising their permissions when users login. Since these utilities see little if any use at most sites, there is no real loss of functionality.

8.10. Use Blowfish encryption for all users by default

Action:

```
sed -i .preCIS -e 's/passwd_format=md5/passwd_format=blf/' /etc/login.conf

cap_mkdb /etc/login.conf

for x in `awk -F: '($3 >= 1001) && ($3 != 65534) || ($3 == 0) { print $1 }' /etc/passwd`;
     do pw usermod $x -L default; \
     done;
```

MD5 encryption hashes are powerful, but in recent years other, more reliable ciphers have been adopted. Blowfish is one of the more powerful algorithms out there and fully supported for the FreeBSD password file database. Users will need to change their passwords for the settings to take effect as well as having the `login.conf` database rebuilt as is done here. There are interoperability issues with `NIS` and `NIS+` configurations. In those cases, other algorithms are supported, including `MD5` which is currently the default, and `des`. Administrators should also familiarize themselves with the **FIPS-180** standard which contains information about US government accepted password hashes. Administrators working for the government may be required to use a different and more accepted algorithm over Blowfish.

II. Appendices

Appendix A. System Update Software

FreeBSD Update

For those who may wish to try a non-supported binary update system on FreeBSD, FreeBSD Update is available. This only requires installing the port with `pkg_add -r freebsd-update` and modifying the sample configuration file `/usr/local/etc/freebsd-update.conf.sample` to fit your situation.

Portupgrade

For add-on software not supported by the FreeBSD project, the `portupgrade` package is available from the ports collection. Simply install `/usr/ports/sysutils/portupgrade` with a `pkg_add -r portupgrade`, run `/usr/local/sbin/pkgdb -F` to build a package database and then periodically run the `/usr/local/sbin/portupgrade PORTNAME` command. See the manual pages for a list of available `portupgrade` options.

Portaudit

The `portaudit` software package will scan the versions of installed third party software from the FreeBSD ports collection for known vulnerabilities. The offending packages will then be printed to `stdout`. When installed, `portaudit -a` will be included as part of the daily security output with known vulnerable ports being added to the email/log file along with an accompanying reference number. Install portaudit from the `/usr/ports/security/portaudit` directory. View the database online by visiting http://vuxml.FreeBSD.org (vuxml.FreeBSD.org/).

Appendix B. References

The Center for Internet Security

Free benchmark documents and security tools for various OS platforms and applications: http://www.cisecurity.org/

Pre-compiled software packages for various OS platforms: ftp://ftp.cisecurity.org/

The FreeBSD Project

Patches and related documentation: http://www.FreeBSD.org/security/

Known port vulnerabilities database: http://vuxml.FreeBSD.org/

The FreeBSD documentation project: http://www.FreeBSD.org/docs.html/

The TrustedBSD Project: http://www.TrustedBSD.org/

The FreeBSD security manual page: http://www.freebsd.org/cgi/man.cgi?query=security&manpath=FreeBSD+5.2-current&format=html (http://www.freebsd.org/cgi/man.cgi?query=security&manpath=FreeBSD+5.2-current&format=html)

Miscellaneous Documentation

Primary source for information on NTP: http://www.ntp.org/

Information on MIT Kerberos: http://web.mit.edu/kerberos/www/

Apache "Security Tips" document: http://httpd.apache.org/docs-2.0/misc/security_tips.html/

Information on Sendmail and DNS: http://www.sendmail.org/

Software

The FreeBSD ports collection: http://www.FreeBSD.org/ports/

OpenSSH (secure encrypted network logins): http://www.openssh.org/

TCP Wrappers source distribution and documentation: ftp://ftp.porcupine.org/

PortSentry (monitors unused network ports for unauthorized access): http://www.psionic.com/products/portsentry.html/

Open Source Sendmail (email server) distributions: ftp://ftp.sendmail.org/

LPRng (Open Source replacement printing system for Unix): http://www.lprng.org/

Tripwire (free and commercial file system integrity checking software): http://www.tripwire.com/products/tripwire_asr/ http://www.tripwire.org/

sudo (provides fine-grained access controls for superuser activity): http://www.courtesan.com/sudo/

Nessus (free remote security scanner): http://www.nessus.org/

Common UNIX Printing System (CUPS): http://www.cups.org/