C41 Technical Report
C41-0008R-2003

Guide to the Secure Configuration of Solaris 8

Operating Systems Division UNIX Team
of the
Systems and Network Attack Center (SNAC)

Dated: 09 October 2003
Version 1.0

National Security Agency
9800 Savage Rd. Suite 6704
Ft. Meade, MD 20755-6704

SNAC.Guides@nsa.gov

This page is intentionally left blank

Warnings

- Do not attempt to implement any of the settings in this guide without first testing in a non-operational environment.
- This document is only a guide containing recommended security settings. It is not meant to replace well-structured policy or sound judgment. Furthermore this guide does not address site-specific configuration issues. Care must be taken when implementing this guide to address local operational and policy concerns.
- The security changes described in this document only apply to the Solaris 8 Operating System and should not be applied to any other operating system.
- The recommendations in this guide were written for SPARC based systems. Some scripts may need to be modified to work on x86 based systems.
- SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE EXPRESSLY DISCLAIMED. IN NO EVENT SHALL THE CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

Acknowledgements

This document is based closely upon the Center for Internet Security's (CIS) *Solaris Benchmark* , without which this guide would not be possible. We would like to thank all of the team members that participated in the development of the CIS *Solaris Benchmark* guide.

Trademark Information

Table of Contents

The CIS Solaris Benchmark was used as a baseline for this document. The following key notates differences from the CIS guide. The CIS benchmark can be found at http://www.cisecurity.org.

**	Additional information added to CIS discussion
#	Section scheduled for removal from CIS document
+	Script modified from CIS form
++	Recommendation modified from CIS form
>	New Section

ABSTRACT

This document provides additional security measures beyond those specified in the Center for Internet Security *Solaris Benchmark*. The document was developed to provide system administrators with steps to create a more secure Solaris 8 operating environment running on a SPARC processor.

The document is written to give a detailed step-by-step description on how to secure a system running Solaris 8. Guidance is provided on how to set up the partitions, apply the latest recommended patches, and configure system settings. While the CIS *Solaris Benchmark* consists of security actions for both Solaris 8 and Solaris 9, the additional information provided by the National Security Agency (NSA) only applies to Solaris 8. Many of the steps in this document will need to be repeated on a regular basis to maintain system security and all of the steps should be reviewed if the system is upgraded for any reason. **This document should be read in the order presented since some sections build upon previous sections.**

The information in the CIS document is the collaborative work of several agencies, including the NSA, colleges and company representatives. The NSA configuration guide includes information found on several computer security related sites, various Solaris 8 documents and work-related experience.

How to Use This Document

Shaded Items
Systems deployed as desktop workstations typically have different security expectations than systems deployed as network servers. 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.

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.

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 "copy-and-paste" operation. The "copy-and-paste" operation applies to all sections with the exception of sections containing red shaded variables <os>, <ver>, x.x.x.x etc. The red shaded variables denote instances where the system administrator must input the appropriate information

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/ftpusers /etc/hosts.equiv /etc/inittab /etc/issue \
/etc/.login /etc/motd /etc/pam.conf /etc/passwd /etc/profile \
/etc/rmmount.conf /etc/shadow /etc/shells /etc/syslog.conf \
/etc/system /etc/vfstab /etc/default/cron /etc/default/ftpd \
/etc/default/inetinit /etc/default/init /etc/default/login \
/etc/default/sendmail /etc/default/telnetd /etc/inet/inetd.conf \
/etc/dfs/dfstab /etc/ssh/ssh*_config /.rhosts /.shosts \
/etc/cron.d/*.allow /etc/cron.d/*.deny /etc/dt/config/Xaccess \
/etc/dt/config/Xservers /etc/dt/config/*/sys.resources \
/etc/init.d/nfs.server  /etc/dt/config/*/Xresources \
/etc/init.d/perf /etc/init.d/inetsvc /etc/init.d/syslog; do
[ -f $file ] && cp $file $file-preNSA
done
```

This page is intentionally left blank

1 Patches and Additional Software

1.1 Partition hard drive to compartmentalize data.

Action (Solaris 8):

Keeping their uses in mind, create the following partitions during the install process. After installation the partitions are very difficult to resize so plan ahead. The number of configurable partitions is limited to seven on a SPARC platform and nine on the Intel platform.

These directory names are commonly used and the partitions should be created accordingly.

/ — for everything not explicitly covered by the following partitions; once installed, very little is added to this directory.

swap — most systems have RAM of adequate size so swap is used infrequently; a good rule is to make swap equivalent to RAM size unless you anticipate large loads, which would lead you to setting this to 1.5 times fast memory for standard applications (e.g. `Ls`,`lp`, `vi`, etc.). The swap partition is mounted as `/tmp`.

/opt — for third party software; software is most frequently added here as new applications and tools are marketed so make this partition sufficiently large to accommodate new software.

/usr/local — for local workstation software (e.g. open source software like `perl`, gnu tools, etc.)

/var — for logging; when using BSM, logging data can grow quite quickly so make sure this partition is sufficiently large in size.

The following partitions have suggested names that may be changed as desired. These directories may or may not be needed, depending on the function the machine serves.

/var/spool/mqueue — for local queuing of mail before sending; remember to avoid using the same name for this directory as the directory used by the mail server.

/export/home — each user should have an adequate amount of space for the work they are doing; estimate the number of users and plan accordingly.

/anonftp/incoming — if anonymous `ftp` upload is allowed, make the writable directory its own partition.

Once the partitions are created and installed, set the permissions for these directories as recommended in this guide.

Discussion:

Partitioning data will help security in a number of ways, including: protecting against a denial-of-service system failure by users filling their home directories or by logs filling up, making it easier to manage space and back-up routines, protecting against NFS weaknesses, and making it easier to protect data and prevent unauthorized changing of data by separating it into its own partition.

You must already have a plan for what size you will need each partition to be. This involves knowing what the system will be used for and by whom. Since there is no easy way to redistribute disk space, the OS must be reinstalled in order to re-partition.

1.2 Apply latest OS patches

Action (Solaris 7 and later):

1. Download Sun Recommended Patch Cluster into `/tmp` (Sun Recommended Patch Clusters can be obtained from ftp://sunsolve.sun.com/pub/patches/ -- look for files named <osrel>_Recommended.zip, where <osrel> is the Solaris OS release number).

2. Execute the following commands:

```
cd /tmp
unzip -qq *_Recommended.zip
cd *_Recommended
./install_cluster -q
```

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. Note that in addition to installing the Solaris Recommended Patch Clusters as described above, administrators may wish to also check the `Solaris<osrel>.PatchReport` file (available from the same FTP site as the patch clusters) for additional security, Y2K, or functionality patches that may be required on the local system. Administrators are also encouraged to check the individual `README` files provided with each patch for further information and post-install instructions. Automated tools for maintaining current patch levels are also available, such as the Solaris Patch Manager tool (for more info, see http://www.sun.com/service/support/sw_only/patchmanager.html).

During the cluster installation process, administrators may ignore individual patch installs that fail with either return code 2 (indicates that the patch has already been installed on the system) or return code 8 (the patch applies to an operating system package which is not installed on the machine). If a patch install fails with any other return code, consult the patch installation log in `/var/sadm/install_data`.

Note that Section 6.1 below recommends mounting the `/usr` file system read-only. When applying patches to a system that has already been secured according to the steps in this document, the read-only setting on `/usr` will cause patch installs to fail. Please refer to the Discussion section in Section 6.1 for information on making the file system writable before applying patches.

1.3 Install TCP Wrappers

Action (Solaris 8 and earlier):

1. Download pre-compiled TCP Wrappers software package from ftp://ftp.sunfreeware.com/pub/freeware/<proc>/<osrel>/ (here <proc> is the processor type--"sparc" or "intel"-- and <osrel> is the Solaris version number of your system, e.g. "5.8", etc.). The file name will be slightly different depending on the version of the software and the OS release, e.g. `tcp_wrappers-7.6-sol8-sparc-local.gz.` If the site is using IPv6, the `tcp_wrappers_ipv6-7.6-sol8-sparc-local.gz` file should be downloaded.

   Note that the gzip compression utilities must be installed in order to install the TCP Wrappers software package. The `gzip` utilities are included with the Solaris OS as of Solaris 8 (though the local site may have chosen not to install these utilities as part of their standard install image). Pre-compiled binaries for various Solaris releases may be obtained from the URL given above, where the package name would again be something like `gzip-1.3.5-sol8-sparc-local` (depending on the current version number of the `gzip` software and the OS revision). Use the command "`/usr/sbin/pkgadd d gzip-*-local all`" to install the `gzip` software from this package file after downloading.

2. Install package:

```
/usr/local/bin/gunzip tcp_wrappers-*-local.gz
pkgadd -d tcp_wrappers-*-local all
```

3. Remove package file after installation:

```
rm -f tcp_wrappers-*-local
```

4. Create `/etc/hosts.allow`:

```
echo "ALL: <net>/<mask>, <net>/<mask>, ..." > /etc/hosts.allow
```

   where each <net>/<mask> combination (for example, "192.168.1.0/255.255.255.0") represents one network block in use by your organization.

5. Create `/etc/hosts.deny`:

```
echo "ALL: ALL " >/etc/hosts.deny
```

6. Modify `inetd.conf`:

```
cd /etc/inet
awk '($3 ~ /^(udp|tcp)/) && \
     ($6 != "internal") \
     { $7 = $6; $6 = "/usr/local/bin/tcpd" }; \
     { print }' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
chown root:sys inetd.conf
chmod 444 inetd.conf
```

Action (Solaris 9):

1. Create `/etc/hosts.allow`:

```
echo "ALL: <net>/<mask>, <net>/<mask>, ..." > /etc/hosts.allow
```

where each <net>/<mask> combination (for example, "192.168.1.0/255.255.255.0") represents one network block in use by your organization.

2. Create `/etc/hosts.deny`:

```
echo "ALL: ALL" > /etc/hosts.deny
```

3. Modify `inetd.conf`:

```
cd /etc/inet
awk '($3 ~ /^(udp|tcp)/) && \
     ($6 != "internal") \
     { $7 = $6; $6 = "/usr/sfw/sbin/tcpd" }; \
     { print }' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
chown root:sys inetd.conf
chmod 444 inetd.conf
```

Discussion:

TCP Wrappers allow the administrator to control what hosts have access to various network services based on the IP address of the remote end of the connection. TCP Wrappers also provide logging information via `Syslog` about both successful and unsuccessful connections. TCP Wrappers are generally triggered out of `/etc/inet/inetd.conf`, but other options exist for "wrappering" non-`inetd`-based software (see the documentation provided with the source code release).

Solaris 9 now includes the TCP Wrappers distribution as part of the operating system (assuming the administrator has installed the `SUNWtcpd` software package).

1.4 Install random number generator

Actions (Solaris 9):

No actions needed

Actions (Solaris 8):

Install one of two kernel patches:
112438-01 (for Sparc)
112439-01 (for x86)
Note: The system must be rebooted for the patches to take effect

```
init 6
```

Actions (Solaris 7 and earlier):

Install `prngd`

1. Obtain appropriate `prngd` package from http://www.sunfreeware.com.

2. Install the package

```
mv packagefile /var/spool/pkg
cd /var/spool/pkg
pkgadd -d packagefile
```

3. Configure the software

3.1 Copy, modify, and link the startup file

```
cp /usr/local/doc/prngd/contrib/Solaris-7/prngd /etc/init.d
cd /etc/init.d
sed 's/sbin/bin/; s/\/opt//; s/prngd-socket/egd-pool/' \
prngd > prngd.new
mv prngd.new prngd
chmod 0744 prngd
chown root:sys prngd
ln -s /etc/init.d/prngd /etc/rc0.d/K30prngd
ln -s /etc/init.d/prngd /etc/rc1.d/K30prngd
ln -s /etc/init.d/prngd /etc/rc2.d/S20prngd
ln -s /etc/init.d/prngd /etc/rcS.d/K30prngd
```

3.2 Set up the configuration file

```
mkdir /usr/local/etc/prngd
cp /usr/local/doc/prngd/contrib/Solaris-7/prngd.conf.solaris-7 \
/usr/local/etc/prngd/prngd.conf
chown root:root /usr/local/etc/prngd/prngd.conf
chown root:root /usr/local/etc/prngd
```

3.3 Establish initial seed value

```
cat /var/adm/* > /usr/local/etc/prngd/prngd-seed
init 6
```

Discussion:

Older versions of Solaris (pre-Solaris 9) did not ship with built-in random number generators. This lack of a readily available entropy pool leads to significant delay upon startup for encryption programs such as SSH. To remedy this delay, it is recommended that either a kernel patch to create an entropy pool (Solaris 8) or the freely-available `prngd` daemon (Solaris 7 and earlier) be installed.

1.5 Install SSH

Action (Solaris 8):

The following instructions are for compiling and installing `ssh` on a development machine. This machine will need to have the developer tools of `make` and `gcc` available as well as the `gzip` utility from the Companion CD. Once the following software is compiled with the necessary options, it can be packaged for distribution and placed on other machines that do not have or need the development tools. Where the term ver.num is shown, replace it with the appropriate version number of the downloaded software being referenced.

Note that the `/etc/issue` and `/etc/motd` must be configured as recommended in Section 7.10.

0. Before installing the following software, make sure you have completed instructions on installing a random number generator and rebooted your system.

1. Gather the necessary files into the `/opt` directory. The directory should not be world-writable.

   a)
   Download the latest version of `md5` to the `/opt` directory from http://sunsolve.sun.com/md5.
   Install `md5`.

```
cd /opt
zcat /opt/md5.tar.Z | tar xvf -
chown root:bin /opt/md5
chmod 755 /opt/md5
mkdir /usr/local/bin
ln -s /opt/md5/md5-sparc /usr/local/bin/md5-sparc
```

   b)
   Download the latest version of `zlib` (1.1.4 or greater) and the file containing its MD5 hash to the `/opt` directory from http://www.gzip.org/zlib. Compare the MD5 hash number with a hash of the gzipped file to ensure a match. The following `grep` command should print a line indicating the MD5 hash was correctly found.

```
cd /opt
grep `/usr/local/bin/md5-sparc zlib-ver.num.tar.gz | cut -f 4 -d “ “` \
zlib-ver.num.tar.gz.md5
```

   Ensure that the following patch is applied to the Sparc system to prevent attacks related to `zlib` vulnerabilities.

```
showrev -p | grep 112611-01
```

c)
Download the OpenSSL package, version number 0. 9.6 or greater and the file containing its MD5 hash to the `/opt` directory from http://www.openssl.org. Compare the MD5 hash number with a hash of the gzipped file to ensure a match. The following `grep` command should print a line indicating the MD5 hash was correctly found.

```
cd /opt
grep `/usr/local/bin/md5-sparc openssl-ver.num.tar.gz | \
cut -f 4 -d " "` openssl-ver.num.tar.gz.md5
```

d)
Download the latest version of OpenSSH to the `/opt` directory from http://www.openssh.org.

2. Install the following programs you have downloaded. The administrator must insure that `/usr/ccs/bin` in the `$PATH` for `make` and also `/opt/sfw/bin` for `gcc` to work correctly.

a) `zlib`

```
cd /opt
/usr/bin/gunzip -c zlib-ver.num.tar.gz | tar xvf -
chown -R root:bin zlib-ver.num
chmod 755 zlib-ver.num
cd /opt/zlib-ver.num
./configure
/usr/ccs/bin/make
/usr/ccs/bin/make test
/usr/ccs/bin/make install prefix=/usr/local
```

b) openssl (at least version 9.6j or 9.7b or above)

```
cd /opt
/usr/local/bin/gunzip -c openssl-ver.num.tar.gz | tar xvf -
chown -R root:bin openssl-ver.num
chmod 755 openssl-ver.num
cd /opt/openssl-ver.num
./Configure solaris-sparcv8-gcc --prefix=/usr/local --
openssldir=/usr/local/openssl
/usr/ccs/bin/make
/usr/ccs/bin/make test
/usr/ccs/bin/make install
```

c) OpenSSH

```
cd /opt
/usr/local/bin/gunzip -c openssh-ver.num.tar.gz | tar xvf -
chown -R root:bin openssh-ver.num
chmod 755 openssh-ver.num
cd /opt/openssh-ver.num
./configure --sysconfdir=/usr/local/OpenSSH --with-pam \
--with-tcp-wrappers --with-last-log=/var/adm/lastlog \
--without-privsep-user --without-privsep-path --without-prngd \
--without-rand-helper
/usr/ccs/bin/make
/usr/ccs/bin/make install
```

3. Edit OpenSSH server configuration file `sshd_config`

```
touch /etc/issue
cd /usr/local/OpenSSH
nawk '/#Banner/                  { sub(/^#/,""); $2 = "/etc/issue" }; \
      /#HostbasedAuthentication/ { sub(/^#/,""); $2 = "no" }; \
      /#IgnoreRhosts/            { sub(/^#/,""); $2 = "yes" }; \
      /#IgnoreUserKnownHosts/    { sub(/^#/,""); $2 = "yes" }; \
      /#KeepAlive/               { sub(/^#/,""); $2 = "yes" }; \
      /#LoginGraceTime/          { sub(/^#/,""); $2 = "120" }; \
      /#PasswordAuthentication/  { sub(/^#/,""); $2 = "yes" }; \
      /#PermitEmptyPasswords/    { sub(/^#/,""); $2 = "no" }; \
      /#PermitRootLogin/         { sub(/^#/,""); $2 = "no" }; \
      /#PrintLastLog/            { sub(/^#/,""); $2 = "yes" }; \
      /#PrintMotd/               { sub(/^#/,""); $2 = "yes" }; \
      /#Protocol/                { sub(/^#/,""); $2 = "2" }; \
      /#RhostsRSAAuthentication/ { sub(/^#/,""); $2 = "no" }; \
      /#UseLogin/                { sub(/^#/,""); $2 = "no" }; \
      /#UsePrivilegeSeparation/  { sub(/^#/,""); $2 = "no" }; \
      /#RhostsAuthentication/    { sub(/^#/,""); $2 = "no" }; \
      /#StrictModes/             { sub(/^#/,""); $2 = "yes" }; \
       { print }' sshd_config > sshd_config.new
echo “AllowTCPForwarding yes” >> sshd_config.new
mv sshd_config.new sshd_config
chown root:sys sshd_config
chmod 600 sshd_config
```

4. Establish RC scripts using a simple script from http://sunfreeware.com

```
cd /etc/init.d
cat << END_SCRIPT >> opensshd
#!/bin/sh
pid=\`/usr/bin/ps -e | /usr/bin/grep sshd | \
/usr/bin/sed -e 's/^  *//' -e 's/ .*//'\`
case "\$1" in
    'start')
      /usr/local/sbin/sshd
      ;;
    'stop')
      if [ "\${pid}" != "" ]
      then
            /usr/bin/kill \${pid}
      fi
      ;;
    *)
      /usr/bin/echo "usage: /etc/init.d/opensshd {start|stop}"
      exit 1
      ;;
esac
exit 0
END_SCRIPT
chmod 744 /etc/init.d/opensshd
chown root:sys /etc/init.d/opensshd
ln /etc/init.d/opensshd /etc/rc3.d/S25opensshd
ln /etc/init.d/opensshd /etc/rcS.d/K30opensshd
ln /etc/init.d/opensshd /etc/rc0.d/K30opensshd
ln /etc/init.d/opensshd /etc/rc1.d/K30opensshd
ln /etc/init.d/opensshd /etc/rc2.d/K40opensshd
```

5. Start OpenSSH. Reboot the system to start the OpenSSH daemon automatically or manually start the opensshd by typing the following command.

```
/etc/init.d/opensshd start
```

Discussion:

OpenSSH is a popular free distribution of the standards-track SSH protocols. However, compilation of OpenSSH is complicated by the fact that it is dependent upon several other freely-available software libraries which also need to be built before OpenSSH itself can be compiled. Until an official distribution source can be set up, these files will be downloaded from the Internet directly.

Though OpenSSH has the capability for privilege separation, a feature that contributes to the security of the system through use of an unprivilege process, the functionality with Pluggable Authentication Module (PAM) on a Solaris system is still not robust so it is not currently recommended.

For more information on building OpenSSH from source, see http://www.openssh.org. Sun also publishes information on building OpenSSH for Solaris as part of its Blueprints series (see http://www.sun.com/solutions/blueprints/0103/openSSH.pdf).

1.6 Install NTP

Action (Solaris 8):

NTP server information:

Note: Enter the correct ip address for your site.

1. Create the `ntp` configuration file

```
cat << END_SCRIPT > /etc/inet/ntp.conf
# subnet
restrict x.x.x.0 mask 255.255.255.0 notrust nomodify notrap
# ip address of this system's time server
restrict x.x.x.x noquery nomodify notrap
# ip address of this system's time server
server x.x.x.x key 2
enable auth
# Add drift file if necessary
driftfile /var/ntp/drift
keys /etc/inet/ntp.keys
trustedkey 1 2
END_SCRIPT
chown root:root /etc/inet/ntp.conf
chmod 600 /etc/inet/ntp.conf
```

2. Create the `drift` file

```
touch /var/ntp/drift
chown root:root /var/ntp/drift
chmod 600 /var/ntp/drift
```

3. Key setup

> Note: The following steps assume that a key file already exists on the system. If an `ntp.key` file does not exist, the file will be created in the following steps.

```
cat <<END_SCIRPT >> /etc/inet/ntp.keys
#keyid key_type key_value
1    M       keypass1
2    M       keypass2
END_SCRIPT
chown root:root /etc/inet/ntp.keys
chmod 600 /etc/inet/ntp.keys
```

4. Start ntp daemon

```
/etc/init.d/xntpd start
```

NTP client information:
Note: Please enter the correct ip address for your site.
1. Create the ntp configuration file

```
cat << END_SCRIPT > /etc/inet/ntp.conf
# ip address of time server created above or known network time server
restrict x.x.x.x noquery nomodify notrap
server x.x.x.x key 1
enable auth
# Add drift file if necessary
driftfile /var/ntp/drift
keys /etc/inet/ntp.keys
trustedkey 1
END_SCRIPT
chown root:root /etc/inet/ntp.conf
chmod 600 /etc/inet/ntp.conf
```

2. Create the drift file

```
touch /var/ntp/drift
chown root:root /var/ntp/drift
chmod 600 /var/ntp/drift
```

3. Key setup

Note: The following steps assume that a key file already exists on the system.

```
cat <<END_SCRIPT >> /etc/inet/ntp.keys
#keyid key_type key_value
1    M        keypass1
END_SCRIPT
chown root:root /etc/inet/ntp.keys
chmod 600 /etc/inet/ntp.keys
```

4. Start `ntp` daemon

```
/etc/init.d/xntpd start
```

Discussion:

It is important for the computer system to maintain correct time. Especially if databases or auditing tools are running on the system. The drift file is used to store the time difference between the local clock and the network clock. Because the value is stored on the system, it does not have to be recalculated every time synchronization occurs. The `drift` file should be used if multiple servers are listed in the `ntp.conf` file.

The key file information above is an example. These keys are used to compute the digital signatures for the NTP transaction. The key file must limit read permissions because it contains authorization data. The keyid can range from 1 to 4294967295 but must not be 0 (zero). Each key number must be unique. There must be a space between the keyid and the key_type. The key_value field, shown as keypass1 above, should be an arbitrary string of up to eight characters.

The keyid and associated key_value must be known to both the server and the client attempting to access the server. If the correct key information is not provided, time synchronization will not take place. The key information should be transferred to each client in the most secure manner possible. For example, the key information can be put on a disk and the system administrator can load the keys on each system. If `ssh` is used, the keys can be transferred over the network. NTP version 4 has a built in key distribution process. Information about this process can be found in the NTP version 4 documentation.

Additional information on how to configure a NTP server and client can be obtained from http://www.sun.com/solutions/blueprints/0701/NTP.pdf.

2 Minimize `inetd` Network Services

2.1 Disable standard services

Action (Solaris 8):

```
cd /etc/inet
for svc in time echo discard daytime chargen fs dtspc \
    exec comsat talk finger uucp name xaudio; do
    awk "(\$1 == \"$svc\") { \$1 = \"#\" \$1 }; {print}" \
    inetd.conf >inetd.conf.new
    mv inetd.conf.new inetd.conf
done
for svc in 100068 100146 100147 100150 100155 100221 \
    100232 100235 rstatd rusersd sprayd walld; do
    awk "/^$svc\\// { \$1 = \"#\" \$1 }; { print }" \
    inetd.conf >inetd.conf.new
    mv inetd.conf.new inetd.conf
done
for svc in printer shell login telnet ftp tftp; do
    awk "(\$1 == \"$svc\") { \$1 = \"#\" \$1 }; {print}" \
    inetd.conf >inetd.conf.new
    mv inetd.conf.new inetd.conf
done
for svc in 100083 100229 100230 100242 \
    100234 100134 kerbd rquotad; do
    awk "/^$svc\\// { \$1 = \"#\" \$1 }; { print }" \
    inetd.conf >inetd.conf.new
    mv inetd.conf.new inetd.conf
done
chown root:sys inetd.conf
chmod 444 inetd.conf
```

Discussion:

The stock /etc/inet/inetd.conf file shipped with Solaris contains many services which are rarely used or which have more secure alternatives. Indeed, after enabling SSH (see Section 1.5) it may be possible to completely do away with all inetd-based services, since SSH provides both a secure login mechanism and a means of transferring files to and from the system. In fact, the actions above will disable all standard services normally enabled in the Solaris inetd.conf file.

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 last two loops in the "**Action**" section above. 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.

Note: Sections 2.2 through 2.7, 2.9 and 2.10 have been moved to Appendix B. These sections enable tools that decrease system security. These tools should only be enabled if there is a mission-critical need.

2.8 Only enable CDE-related daemons if absolutely necessary

Question:

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

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

Action (Solaris 8):

```
sed 's/^#100083/100083/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:

The rpc.ttdbserverd process supports many tools and applications in Sun's CDE windowing environment, but has historically been a major security issue for Solaris systems. If this service is enabled, it is vital to keep up to date on vendor patches. Never enable this service on any system which is not well protected by a complete network security infrastructure (including network and host-based firewalls, packet filters, and intrusion detection infrastructure).

2.11 Minimize inetd.conf file

Action (Solaris 8):

```
mv inetd.conf inetd.conf.complete
grep -v '^#' inetd.conf.complete > inetd.conf
chown root:sys inetd.conf
chmod 444 inetd.conf
```

Discussion:

With so much of the stock inetd.conf file devoted to comments, it can be very difficult to see when new services have been added to the file. Also, automated exploit scripts have been known to automatically re-enable inetd-based services by removing the comment character from disabled entries. By reducing the inetd.conf file to only the "active" service entries, these exploit scripts are thwarted and make it much easier for administrators to audit the file for new service entries that may have been added.

Note that the original inetd.conf file is saved as inetd.conf.complete. Should the administrator wish to enable additional services in the future, they can simply edit the inetd.conf.complete file and then re-run the grep command line given above.

2.12 Disable multicasting and routing discovery

Question:

Is there a mission-critical reason to run multicasting network services at this site?

If the answer is no, proceed with the actions below.
Note: If ipfilters will be used (see Section 4.7), then skip this step.

Action (Solaris 8):

```
awk '/Setting default IPv4 interface for multicast/  {$1 = "#"$1}; \
     /add net 224/      {$1 = "#"$1}; \
     /add -interface/   {$1 = "#"$1}; \
     { print }' /etc/init.d/inetsvc > /etc/init.d/inetsvc.new
mv /etc/init.d/inetsvc.new /etc/init.d/inetsvc
chown root:sys /etc/init.d/inetsvc
chmod 744 /etc/init.d/inetsvc
```

Discussion:

Solaris 8 supports multicasting by default. By disabling multicasting, router discovery can not be performed. An alternate method for disabling router discovery is listed below.

```
touch /etc/notrouter
chown root:sys /etc/notrouter
chmod 644 /etc/notrouter
```

2.13 Disable IPv6

Question:

Is IPv6 in use at this site?

If the answer is no, proceed with the actions below.

Action (Solaris 8):

1. Remove all IPv6 hostname information

```
cd /etc
rm hostname6.*
```

2. Remove all IPv6 TCP and UDP information from `inetd.conf`

```
awk '(( $3 == "tcp6" || $3 == "udp6" ) && ( $1 != "^#" )) \
     { $1 = "#"$1}; \
     { print }' /etc/inet/inetd.conf > /etc/inet/inetd.conf.new
mv /etc/inet/inetd.conf.new /etc/inet/inetd.conf
chown root:sys /etc/inet/inetd.conf
chmod 444 /etc/inet/inetd.conf
```

Discussion:

Although IPv6 provides added security to the network layer, it is not widely used. If the system is configured to handle IPv6 and it is not being used, IPv6 related services and interfaces should be disabled.

2.14 Enable encrypted remote administration if necessary

Action (Solaris 8):

Enable X Graphical User Interface for administration if necessary

On the machine that is to be administered, the following commands must be issued locally as `root`. Ensure that no `ssh` sessions are active before beginning.

```
/etc/init.d/opensshd stop
cd /usr/local/OpenSSH
nawk '/#X11Forwarding/  { sub(/^#/,""); $2 = "yes" }; \
     { print }' sshd_config > sshd_config.new
mv sshd_config.new sshd_config
chown root:sys sshd_config
chmod 600 sshd_config
/etc/init.d/opensshd start
```

Discussion:

Remote administration must be done over an encrypted channel to protect against information or control leakage. OpenSSH is an appropriate communication encryption tool to use for remote administration. The box that is to be administered remotely must first be configured locally to allow X11 forwarding.

3 Minimize Boot Services

3.1 Disable `login:` prompts on serial ports

Action (Solaris 8):

```
cd /etc
grep -v /usr/lib/saf/sac inittab > inittab.new
mv inittab.new inittab
chown root:sys inittab
chmod 644 inittab
```

Discussion:

Disabling the `login`: prompt on the system serial device makes it more difficult for unauthorized users to attach modems, terminals, and other remote access devices to these ports.

Note that this action may safely be performed even if console access to the system is provided via the serial ports, because the `login`: prompt on the console device is provided through a different mechanism.

3.2 Set daemon `umask`

Action (Solaris 8 and later):

The `CMASK` parameter in `/etc/default/init` should be at least 022 (note that this is the default setting for Solaris 8 and later).

Action (Solaris 7 and earlier):

```
echo "umask 022" > /etc/init.d/umask.sh
chmod 744 /etc/init.d/umask.sh
for dir in /etc/rc?.d
do
     ln -s /etc/init.d/umask.sh $dir/S00umask.sh
done
```

Discussion:

The system default `umask` should be set to at least 022 in order to prevent daemon processes 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.

3.3 Turn on `inetd` tracing, disable `inetd` if possible

Action (Solaris 8):

```
cd /etc/init.d
grep -v /usr/sbin/inetd inetsvc > newinetsvc
cat <<'EONewInetd' >> newinetsvc
lines=`grep -v '^#' /etc/inet/inetd.conf 2>/dev/null | \
     wc -l | sed 's/ //g'`
if [ "\$lines" != "0" ]; then
     /usr/sbin/inetd -s -t &
fi
EONewInetd
cp inetsvc inetsvc.old
mv newinetsvc inetsvc
chown root:sys inetsvc
chmod 744 inetsvc
rm -f /etc/rc2.d/S72inetsvc
ln -s /etc/init.d/inetsvc /etc/rc2.d/S72inetsvc
```

Discussion:

If the actions in Section 2 of this benchmark resulted in no services being enabled in `/etc/inet/inetd.conf`, then the revised boot script created here will prevent the `inetd` daemon from even being started. If `inetd` is running, it is a good idea to make use of the "tracing" (`-t`) feature of the Solaris `inetd` that logs information about the source of any network connections seen by the daemon. This information is logged via `Syslog` and by default Solaris systems deposit this logging information in `/var/adm/messages` with other system log messages. Should the administrator wish to capture this information in a separate file, simply modify `/etc/syslog.conf` to log `daemon.notice` messages to some other log file destination.

In addition to the information provided by `inetd` tracing, the popular free 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.

3.4 Prevent Syslog from accepting messages from network

Question:

Is this machine a log server, or does it need to receive Syslog messages via the network from other systems?

If the answer to both parts of the question is no, proceed with the actions below.

Action (Solaris 8):

```
awk '$1 ~ /syslogd/ && !/-(t|T)/ { $1 = $1 " -t" }; \
     { print }' /etc/init.d/syslog >/etc/init.d/newsyslog
cp /etc/init.d/syslog /etc/init.d/syslog.old
mv /etc/init.d/newsyslog /etc/init.d/syslog
chown root:sys /etc/init.d/syslog
chmod 744 /etc/init.d/syslog
rm -f /etc/rc2.d/S74syslog
ln -s /etc/init.d/syslog /etc/rc2.d/S74syslog
```

Discussion:

By default the system logging daemon, `syslogd`, listens for log messages from other systems on network port 514/udp. Unfortunately, the protocol used to transfer these messages does not include any form of authentication, so a malicious outsider could simply barrage the local system's `Syslog` port with spurious traffic--either as a denial-of-service attack on the system, or to fill up the local system's logging file so that subsequent attacks will not be logged.

Note that it is considered good practice to set up one or more machines as central "log servers" to aggregate log traffic from all machines at a site. However, unless a system is set up to be one of these "log server" systems, it should not be listening on 514/udp for incoming log messages.

3.5 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 no, proceed with the actions below.

Action (Solaris 8 and later):

```
cd /etc/default
cat <<END_DEFAULT >sendmail
MODE=
QUEUEINTERVAL="15m"
END_DEFAULT
chown root:sys sendmail
chmod 744 sendmail
```

Action (Solaris 7 and earlier):

```
mv /etc/rc2.d/S88sendmail /etc/rc2.d/.NOS88sendmail
cd /var/spool/cron/crontabs
crontab -l > root.tmp
echo '0 * * * * /usr/lib/sendmail -q' >> root.tmp
crontab root.tmp
rm -f root.tmp
```

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 after disabling the `-bd` option on the local mail server on Solaris 9 (or any system running Sendmail v8.12 or later) it is also necessary to modify the `/etc/mail/submit.cf` file. Find the line that reads "`D{MTAHost}localhost`" and change `localhost` to the name of the appropriate mail server for the organization. This will cause email generated on the local system to be relayed to that mail server for further processing and delivery.

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/.

3.6 Disable boot services if possible

Question:

Is this machine a network boot server or Jumpstart server?

If the answer to both parts of the question is no, then perform the action below.

Action (Solaris 9):

```
mv /etc/rc3.d/S16boot.server /etc/rc3.d/.NOS16boot.server
```

Action (Solaris 8 and earlier):

```
cd /etc/init.d
awk '/tftpboot/,/;;/ { if ($1 != ";;") next }
     { print }' nfs.server > newnfs.server
cp nfs.server nfs.server.old
mv newnfs.server nfs.server
chown root:sys nfs.server
chmod 744 nfs.server
rm -f /etc/rc3.d/S15nfs.server
ln -s /etc/init.d/nfs.server /etc/rc3.d/S15nfs.server
```

Discussion:

If the/tftpboot directory exists (see Appendix B Section 2.5), the in.rarpd and rpc.bootparamd services will be enabled. These services are designed to assist machines and devices that need to download their boot images over the network from some central server. However, the system may be running TFTP and have a /tftpboot directory but not be acting as a boot server (for example, many sites use TFTP to back up configuration files from their network routers). in.rarpd and rpc.bootparamd should only be enabled if the machine is actually going to be acting as a boot server.

3.7 Disable other standard boot services

Action (Solaris 8):

Note: Please refer to the chart in the discussion section before disabling the red-highlighted boot services.

```
cd /etc/rc2.d
for file in S72autoinstall S85power S89bdconfig \
     S73cachefs.daemon S93cacheos.finish S40llc2 S47pppd \
     S47asppp S70uucp S72slpd S75flashprom S80PRESERVE \
     S89PRESERVE S90wbem S94ncalogd S95ncad; do
     [ -s $file ] && mv $file .NO$file
done
cd /etc/rc3.d
for file in S77dmi S80mipagent; do
     [ -s $file ] && mv $file .NO$file
done
cd /etc/rc2.d
for file in S73nfs.client S74autofs S71rpc \
     S72directory S71ldap.client S80lp S80spc S92volmgt \
     S91afbinit S91ifbinit S99dtlogin S42ncakmod; do
     [ -s $file ] && mv $file .NO$file
done
cd /etc/rc3.d
for file in S90samba S15nfs.server S13kdc.master S14kdc \
     S50apache S76snmpdx S34dhcp; do
     [ -s $file ] && mv $file .NO$file
done
```

Discussion:

Renaming these scripts in the system boot directories will effectively disable a wide variety of infrequently used subsystems. The scripts are merely renamed (rather than removed outright) so that the local administrator can easily "restore" any of these files if they discover a mission-critical need for one of these services. Not all of the scripts listed above will exist on all systems (some are only valid for certain releases, others only exist if certain OEM vendor software is installed). Note also that vendor patches may restore some of the original entries in the /etc/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 chart below can be used to determine if the red highlighted boot scripts above should be disabled by the system administrators.

Filename	***Purpose***
/etc/rc2.d/S71rpc	- Starts network service `rpcbind` daemon - Used by NIS & NIS+ configuration, key services, XSun services - Required to run CDE
/etc/rc2.d/S74autofs	- Start `automount` daemon - Used for automounting and to locate directories
/etc/rc2.d/S90wbem	- Configures Web-Based Enterprise Management Services - Needed for System Management Console
/etc/rc2.d/S91afbinit	- Configures any graphic frame buffers or graphic accelerators - Needed for system with Elite3D graphics - Needed for X Window
/etc/rc2.d/S91ifbinit	- Configures any graphic frame buffers or graphic accelerators - Needed for system with Expert3D (IFB) graphics
/etc/rc2.d/S92volmgt	- Starts the `vold` daemon - Neede to mount cdroms and floppy disks
/etc/rc2.d/S99dtlogin	- Starts the CDE desktop login process, dtlogin - Needed for logging in using CDE
/etc/rc3.d/S15nfs.server	- Starts the NFS server daemons `nfsd`, `mountd` and `nfslogd` - Needed to mount NFS systems
/etc/rc3.d/S76snmpdx	- Starts snmp daemon - Needed by Solstice Enterprise Agents dmispd and snmpXdmid

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 last two loops in the "**Action**" section above. 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.

3.8 Only enable Windows-compatibility servers if absolutely necessary

Question:

Does this machine provide authentication, file sharing, or printer sharing services to systems running Microsoft Windows operating systems?

If the answer to any part of the question listed above is yes, proceed with the actions below.

Action (Solaris 9):

```
mv /etc/rc3.d/.NOS90samba /etc/rc3.d/S90samba
```

Discussion:

Solaris 9 now includes the popular Open Source Samba server for providing file and print services to Windows-based systems. This allows a Solaris system to act as a file or print server 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.

3.9 Only enable NFS server processes 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 (Solaris 8):

```
mv /etc/rc3.d/.NOS15nfs.server /etc/rc3.d/S15nfs.server
```

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" and "`nosuid`" where appropriate. For more information consult the `share_nfs` manual page.

3.10 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, proceed with the actions below.

Action (Solaris 8):

```
mv /etc/rc2.d/.NOS73nfs.client /etc/rc2.d/S73nfs.client
mv /etc/rc2.d/.NOS74autofs /etc/rc2.d/S74autofs
```

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.

3.11 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 runs a GUI or GUI-based administration tool*
- *This machine is a network boot server or Jumpstart server*
- *The machine runs a third-party software application which is dependent on RPC support (examples: FlexLM License managers, Veritas, Solaris DiskSuite)*

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

Action (Solaris 8):

```
mv /etc/rc2.d/.NOS71rpc /etc/rc2.d/S71rpc
```

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 you are unsure whether or not a particular third-party application requires RPC services, consult with the application vendor.

3.12 Only enable Kerberos server daemons if absolutely necessary

Question:

Is this system a Kerberos Key Distribution Center (KDC) for the site?

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

Action (Solaris 9):

```
mv /etc/rc3.d/.NOS13kdc.master /etc/rc3.d/S13kdc.master
mv /etc/rc3.d/.NOS14kdc /etc/rc3.d/S14kdc
```

Discussion:

Solaris 9 includes greater support for the Kerberos authentication system. In particular, the Kerberos server daemons have been bundled with the core operating system. However, if the site is not using Kerberos or if this machine is not configured as one of the site's Kerberos servers, there is no reason to enable this service.

3.13 Only enable directory server if absolutely necessary

Question:

Is this system an LDAP directory server for this site?

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

Action (Solaris 9):

```
mv /etc/rc2.d/.NOS72directory /etc/rc2.d/S72directory
```

Discussion:

Solaris 9 has included the iPlanet Directory Server product as part of the operating system. However, this service only needs to be running on the machines that have been designated as LDAP servers for the organization. If the machine is an LDAP server, the administrator is encouraged to search the Web for additional documentation on LDAP security issues. A Secure LDAP Directory Services paper will be available at http:/www.sun.com/blueprints in the Fall of 2003.

3.14 Only enable the LDAP cache manager if absolutely necessary

Question:
Is the LDAP directory service in use at this site, and is this machine an LDAP client?

If the answer to both parts of the question listed above is yes, proceed with the actions below.

Action (Solaris 8 or later):
```
mv /etc/rc2.d/.NOS71ldap.client /etc/rc2.d/S71ldap.client
```

Discussion:
Clearly, if the local site is not currently using LDAP as a naming service, then there is no need to keep LDAP-related daemons running on the local machine.

3.15 Only enable the printer daemons 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, proceed with the actions below.

Action (Solaris 8):
```
mv /etc/rc2.d/.NOS80lp /etc/rc2.d/S80lp
mv /etc/rc2.d/.NOS80spc /etc/rc2.d/S80spc
```

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 these services. The UNIX print service has generally had a poor security record--be sure to keep up-to-date on vendor patches. The administrator may wish to consider converting to the LPRng print system (see http://www.lprng.org/) which was designed with security in mind and is widely portable across many different UNIX platforms.

3.16 Only enable the volume manager if absolutely necessary

Question:

Is there a mission-critical reason why CD-ROMs and floppy disks should be automatically mounted when inserted into system drives?

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

Action (Solaris 8):

```
mv /etc/rc2.d/.NOS92volmgt /etc/rc2.d/S92volmgt
```

Discussion:

The Solaris volume manager automatically mounts CD-ROMs and floppy disks for users whenever a disk is inserted in the local system's drive (the `mount` command is normally a privileged command which can only be performed by the superuser). 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 your network. The malicious programs and data could be used by an unauthorized user to gain root access on the system.

3.17 Only enable GUI login if absolutely necessary

Question:

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

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

Action (Solaris 2.6 and later):

```
mv /etc/rc2.d/.NOS99dtlogin /etc/rc2.d/S99dtlogin
mv /etc/rc2.d/.NOS91afbinit /etc/rc2.d/S91afbinit
mv /etc/rc2.d/.NOS91ifbinit /etc/rc2.d/S91ifbinit
```

Discussion:

The X Windows-based CDE GUI on Solaris systems has had a history of security issues. Never run any GUI-oriented service or application on a system unless that machine is protected by a strong network security infrastructure.

Note that the `S91afbinit` and `S91ifbinit` scripts enable support for high-end frame buffer devices--these will not be required if this system is not running a GUI.

3.18 Only enable Web server 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, proceed with the actions below.

Action (Solaris 8 and later):

```
mv /etc/rc3.d/.NOS50apache /etc/rc3.d/S50apache
mv /etc/rc2.d/.NOS42ncakmod /etc/rc2.d/S42ncakmod
```

Discussion:

Even if this machine is a Web server, the local site may choose not to use the Web server provided with Solaris in favor of a locally developed and supported Web environment. If the machine is a Web server, the administrator is encouraged to search the Web for additional documentation on Web server security. A good starting point is the http://httpd.apache.org/docs-2.0/misc/security_tips.html. The Center for Internet Security will be publishing an Apache Web Server Benchmark at http://www.cisecurity.org.

3.19 Only enable SNMP 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, proceed with the actions below.

Action (Solaris 2.6 and later):

```
mv /etc/rc3.d/.NOS76snmpdx /etc/rc3.d/S76snmpdx
```

Discussion:

If SNMP is used to monitor the hosts on the network, it is recommended that the default community string used to access data via SNMP be changed. On Solaris systems, this parameter can be changed by modifying the `system-group-read-community` parameter in `/etc/snmp/conf/snmpd.conf`.

SNMP is shipped with a default community string of "public" or "private". If the default community string is set to the string of "private", an unauthorized user will have access to remotely read and modify parameters. If the default community string is set to "public", an unauthorized user will have read access to network management information.

The community string should be changed to prevent access to the system parameters by an unauthorized user. The SNMP community string needs to be hard to guess, like passwords. It should include a combination of letters, numbers, special characters and have a minimum length of six characters. Even if community string is changed, it should be noted that SNMP versions 1 and 2 use the community string unencrypted for authentication.

3.20 Only enable DHCP server if absolutely necessary

Question:

Does this machine act as a DHCP server for the network?

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

Action (Solaris 9):

```
mv /etc/rc3.d/.NOS34dhcp /etc/rc3.d/S34dhcp
```

Discussion:

DHCP is a popular protocol for dynamically assigning IP addresses and other network information to systems on the network (rather than having administrators manually manage this information on each host). However, if this system is not a DHCP server for the network, there is no need to be running this service.

3.21 Disable BIND

Question:

Is there a mission-critical reason to run a DNS Server on this system?

If the answer is no, proceed with the actions below:

Action (Solaris 8):

```
1. cd /etc/init.d
cat << END_NAMED > named.script
/if \[ -f \/usr\/sbin\/in.named/ {
     s!if \[ -f!#if \[ -f!
}
/starting internet domain name server/ {
     s!echo!#echo!
}
/\/usr\/sbin\/in.named &/ {
     s!/!#/!
     n
     s!^fi!#fi!
}
END_NAMED
chown root:sys named.script
chmod 744 named.script
```

2. Run the script to change the `inetsvc` file

```
sed -f named.script inetsvc > inetsvc.new
mv inetsvc.new inetsvc
chown root:sys inetsvc
chown 744 inetsvc
```

3. Stop then restart the service

```
/etc/init.d/inetsvc stop
/etc/init.d/inetsvc start
```

Discussion:

BIND can be used by attackers to gather information about the network. If the system is not the DNS server, the `bind` daemon should not be running. If the `named` daemon must be running, the latest version of `bind` should be installed on the system. Additional precautions should be taken to run `bind` securely.

3.22 Disable `nscd`

Question:

Is this system a DNS client or running the Basic Security Module?

If the answer to both parts of the question is no, proceed with the actions below:

Action (Solaris 8):

```
mv /etc/rc2.d/S76nscd /etc/rc2.d/.NOS76nscd
```

Discussion:

The Name Service Cache Daemon maintains a database containing commonly used Domain Named Service (DNS) lookup information such as passwords, groups and hosts.

This service is needed if the system has the Basic Security Module (BSM) or DNS enabled. If BSM or DNS are not used, it is recommended that Name Service Cache Daemon be disabled. If BSM or DNS are used, the nscd daemon must be running.

3.23 Use RMTMPFILES to clear /var/tmp

Question:

Is there a mission-critical reason why files in /var/tmp should not be removed?

If the answer is no, proceed with the actions below:

Action (Solaris 8):

```
cd /etc/init.d
sed 's/^exit/#exit/' RMTMPFILES > RMTMPFILES.new
mv RMTMPFILES.new RMTMPFILES
chown root:sys RMTMPFILES
chmod 744 RMTMPFILES
```

Discussion:

/var/tmp contains information about su, syslogd and the xntp key. The information obtained in this directory can be used to gain access to the system. For example, the su information will show who has access to su to root. When the steps listed above are taken, all the files listed in /var/tmp are removed except Ex* files. The Ex* files are created by using the vi command. Ex* files are removed through the use of the /etc/init.d/PRESERVE.

4 Kernel Tuning

4.1 Disable core dumps

Action (Solaris 8):

```
cat <<END_CFG >>/etc/system
* Prevent core dumps
set sys:coredumpsize = 0
END_CFG
```

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. If you need to allow developers to obtain core files, use the coreadm man page to investigate the coreadm utility. The coreadm utility is available on Solaris 7 and 8 systems.

4.2 Enable stack protection

Action (Solaris 2.6 and later):

```
cat <<END_CFG >> /etc/system
* Attempt to prevent and log stack-smashing attacks
set noexec_user_stack = 1
set noexec_user_stack_log = 1
END_CFG
```

Discussion:

Buffer overflow exploits have been the basis for many of the recent highly publicized compromises and defacements of large numbers of Internet connected systems. Many of the automated tools in use by system crackers exploit well-known buffer overflow problems in vendor-supplied and third-party software. Enabling stack protection prevents certain classes of buffer overflow attacks and is a significant security enhancement.

4.3 Restrict NFS client requests to privileged ports

Action (Solaris 8):

```
cat <<END_CFG >>/etc/system
* Require NFS clients to use privileged ports
set nfssrv:nfs_portmon = 1
END_CFG
```

Discussion:

Setting this 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.

4.4 Modify network parameters

Action (Solaris 8 and later):

```
cat <<END_SCRIPT >/etc/init.d/netconfig
#!/sbin/sh
ndd -set /dev/ip ip_forward_src_routed 0
ndd -set /dev/ip ip6_forward_src_routed 0
ndd -set /dev/tcp tcp_rev_src_routes 0
ndd -set /dev/ip ip_forward_directed_broadcasts 0
ndd -set /dev/tcp tcp_conn_req_max_q0 4096
ndd -set /dev/tcp tcp_ip_abort_cinterval 60000
ndd -set /dev/ip ip_respond_to_timestamp 0
ndd -set /dev/ip ip_respond_to_timestamp_broadcast 0
ndd -set /dev/ip ip_respond_to_address_mask_broadcast 0
ndd -set /dev/arp arp_cleanup_interval 60000
ndd -set /dev/ip ip_ire_arp_interval 60000
ndd -set /dev/ip ip_ignore_redirect 1
ndd -set /dev/ip ip6_ignore_redirect 1
END_SCRIPT
chown root:root /etc/init.d/netconfig
chmod 744 /etc/init.d/netconfig
ln -s /etc/init.d/netconfig /etc/rc2.d/S69netconfig
```

Action (Solaris 7 releases and earlier):

```
cat <<END_SCRIPT >/etc/init.d/netconfig
#!/sbin/sh
ndd -set /dev/ip ip_forward_src_routed 0
ndd -set /dev/ip ip_forward_directed_broadcasts 0
ndd -set /dev/tcp tcp_conn_req_max_q0 4096
ndd -set /dev/tcp tcp_ip_abort_cinterval 60000
ndd -set /dev/ip ip_respond_to_timestamp 0
ndd -set /dev/ip ip_respond_to_timestamp_broadcast 0
ndd -set /dev/ip ip_respond_to_address_mask_broadcast 0
ndd -set /dev/arp arp_cleanup_interval 60000
ndd -set /dev/ip ip_ire_flush_interval 60000
ndd -set /dev/ip ip_ignore_redirect 1
END_SCRIPT
chown root:root /etc/init.d/netconfig
chmod 744 /etc/init.d/netconfig
ln -s /etc/init.d/netconfig /etc/rc2.d/S69netconfig
```

Discussion:

Note: A new script is created in the action listed above. The `S69netconfig` script will be executed at boot time to reconfigure various network parameters. For a more complete discussion of these parameters and their effect on the security of the system, see: http://www.sun.com/solutions/blueprints/1200/network-updt1.pdf

4.5 Modify additional network parameters

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

If the answer to both parts of the question is no, then perform the action below.

Action (Solaris 8 and later):

```
cat <<END_SCRIPT >> /etc/init.d/netconfig
ndd -set /dev/ip ip_forwarding 0
ndd -set /dev/ip ip6_forwarding 0
ndd -set /dev/ip ip_strict_dst_multihoming 1
ndd -set /dev/ip ip6_strict_dst_multihoming 1
ndd -set /dev/ip ip_send_redirects 0
ndd -set /dev/ip ip6_send_redirects 0
END_SCRIPT
```

Action (Solaris 7 and earlier):

```
cat <<END_SCRIPT >>/etc/init.d/netconfig
ndd -set /dev/ip ip_forwarding 0
ndd -set /dev/ip ip_strict_dst_multihoming 1
ndd -set /dev/ip ip_send_redirects 0
END_SCRIPT
```

Discussion:

For a more complete discussion of these parameters and their effect on the security of the system, see the URL noted in the previous item.

4.6 Use better TCP sequence numbers

Action (Solaris 2.6 and later):

```
cd /etc/default
awk '/^TCP_STRONG_ISS/ { $1 = "TCP_STRONG_ISS=2" }; \
     { print }' inetinit > inetinit.new
mv inetinit.new inetinit
chown root:sys inetinit
chmod 444 inetinit
```

Discussion:

Setting this parameter in `/etc/default/inetinit` causes the system to use a better randomization algorithm for generating initial TCP sequence numbers. This makes remote session hijacking attacks more difficult, as well as any other network-based attack that relies on predicting TCP sequence number information.

4.7 Setup host based firewalls

Action (Solaris 8):

1. Download pre-compiled version of `gcc-3.2.2-sol8-sparc-local` from http://www.sunfreeware.com. Place the file in the `/opt` directory.

Note: In order to compile `ipfilters` source code, a compiler capable of creating a 64-bit executable must be used. GCC versions 2.95.5 and later can be used to create 64-bit executables.

2. Install package:

```
cd /opt
pkgadd -d gcc-3.2.2-sol8-sparc-local all
```

3. Download `ip_fil3.4.28.tar.gz` from http://coombs.anu.edu.au/~avalon/ip-filter.html. Place the file in the `/opt` directory.

4. Execute the following commands:

```
gunzip ip_fil3.4.28.tar.gz
tar xvf ip_fil3.4.28.tar
```

5. Install `ip_fil3.4.28`
Note: A kernel loadable module (`/etc/rc2.d/S65ipfboot`) is created during the `ipfilters` installation.

```
PATH=/usr/local/bin:/usr/ccs/bin:$PATH; export PATH
cd ip_fil3.4.28
```

Note: Create `ipfilter` binaries

```
make solaris
cd SunOS5
```

Note: Build the `ipf.pkg` package

```
make package
```

6. Turn on `ipfilter`

```
cat << END_SCRIPT >> /etc/rc.conf
ipfilter_enable="YES"
ipfilter_rules="/etc/opt/ipf/ipf.conf"
ipfilter_flags="-E"
END_SCRIPT
```

7. Set up filter rules:

Note: Use appropriate interface in place of hme0

```
cat << END_SCRIPT > /etc/opt/ipf/ipf.conf
# block all but localhost access
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port = 898  # web-based enterprise management
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port = 3852 # sunscreen gui
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port = 3853 # sunscreen remote admin
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port = 5981 # java browser
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port = 5987 # web-based enterprise management
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port 5999 >< 6005 # Xserver
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from any to any port = 8888 # answerbook

# block all but local network
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from !x.x.x.0/24 to any port = 111  # rpcbind
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from !x.x.x.0/24 to any port = 587  # mail submission
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from !x.x.x.0/24 to any port = 2049 # nfsd
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from !x.x.x.0/24 to any port = 2099 # rmi
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from !x.x.x.0/24 to any port = 4045 # lockd
block return-rst in log first level auth.warn quick on hme0 proto tcp \
from !x.x.x.0/24 to any port 32767 >< 32901# rpc services

block return-icmp(port-unr) in log first level auth.warn quick on \
hme0  proto udp from !x.x.x.0/24 to any port = 111 # rpcbind
block return-icmp(port-unr) in log first level auth.warn quick on \
hme0 proto udp from !x.x.x.0/24 to any port = 161  # snmpdx
block return-icmp(port-unr) in log first level auth.warn quick on \
hme0 proto udp from !x.x.x.0/24 to any port = 514  # syslog
block return-icmp(port-unr) in log first level auth.warn quick on \
hme0 proto udp from !x.x.x.0/24 to any port = 2049 # nfsd
block return-icmp(port-unr) in log first level auth.warn quick on \
hme0 proto udp from !x.x.x.0/24 to any port = 2099 # rmi
block return-icmp(port-unr) in log first level auth.warn quick on \
hme0 proto udp from !x.x.x.0/24 to any port = 4045 # lockd
block return-icmp(port-unr) in log first level auth.warn quick on \
hme0 proto udp from !x.x.x.0/24 to any port 32767 >< 32901 # rpc svcs
END_SCRIPT
chown root:sys /etc/opt/ipf/ipf.conf
chmod 644 /etc/opt/ipf/ipf.conf
```

8. Configure router information

```
route add x.x.x.x localhost 0  # default router
```

9. Add the following to /etc/syslog.conf

```
printf "local0.info;local0.err;local0.debug\t\t/var/log/ipflog\n" \
>> /etc/syslog.conf
```

10. Create `/var/log/ipflog`

```
touch /var/log/ipflog
chown root:sys /var/log/ipflog
chmod 600 /var/log/ipflog
```

11. Reboot the system

Note: The `syslog` daemon will be restarted when the system is rebooted.

```
init 6
```

Discussion:

In some environments, services that should ideally be disabled must remain open due to operational necessity. Thus, care should be taken to prevent unauthorized or insecure access to these services. In the case of services spawned by `inetd`, the TCP Wrappers daemon, discussed previously, is used to perform this access control. Not all services are spawned by `inetd` and some of these services do not have the means to prevent unauthorized access. Therefore it is recommended to use a host based firewall to limit access to a machine's services.

The firewall configuration given above is for the `ipfilter` firewall. In this configuration, some ports are blocked outright so that only the local machine can connect to them. Access to other ports, however, is granted to any machine on a local subnet. Access to other ports not specifically mentioned is assumed to be blocked by TCP Wrappers or a service-specific access control mechanism. For all the ports blocked above, the firewall will log all incoming access attempts and respond to the request as if the port were not open.

The `ipfilter` firewall was chosen because it compiles and runs on both Sparc and x86 platforms, for all versions of Solaris (32- and 64-bit). Further more, modern versions of the firewall software contain support for IPv6 firewall rules.

4.8 Set routing policies/configuration

Question:

Is your machine acting as a router or does it need to perform router discovery?

If the answer is yes, proceed with the actions below to set up static routing.

Action (Solaris 8):

Note: x.x.x.x must be replaced with the address appropriate for your network.

```
echo  x.x.x.x   > /etc/defaultrouter
chown root:sys /etc/defaultrouter
chmod 644 /etc/defaultrouter
```

Discussion:

The `defaultrouter` file is used to provide a default network route for the machine. Its presence also prevents the router discovery daemon, `in.rdisc`, from starting at boot time.

Note: DHCP-published routes supersede the router found in `/etc/defaultrouter`.

5 Logging

The items in this section cover enabling various different forms of system logging in order to keep track of activities 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 in Section 1.6, at http://www.ntp.org and at http://www.sun.com/solutions/blueprints/0701/NTP.pdf.

5.1 Capture messages sent to syslog AUTH facility

Action (Solaris 8):

1) Remove the following lines from `/etc/syslog.conf`

```
cd /etc
awk '/err;kern.notice/  { $1 = "#"$1 }; \
     /err;kern.debug/   { $1 = "#"$1 }; \
     /alert;kern.err/   { $1 = "#"$1 }; \
     /user.alert/ { $1 = "#"$1 }; \
     /user.emerg/ { $1 = "#"$1 }; \
     { print }' syslog.conf > syslog.conf.new
mv syslog.conf.new syslog.conf
chown root:sys syslog.conf
chmod 644 syslog.conf
```

2) Create a script to add the following new information to `/etc/syslog.conf`

```
cat << END_SCRIPT > /etc/syslogadd
#!/bin/sh
printf "auth.err\t\t\t\t\t/dev/console
*.err;auth.notice;kern.debug\t\t\tifdef(\\\`LOGHOST', /
var/adm/messages, @loghost)
kern.info\t\t\t\t\tifdef(\\\`LOGHOST', /var/log/kernlog, @loghost)
user.info\t\t\t\t\tifdef(\\\`LOGHOST', /var/log/userlog, @loghost)
mail.info\t\t\t\t\tifdef(\\\`LOGHOST', /var/log/maillog, @loghost)
daemon.info\t\t\t\t\tifdef(\\\`LOGHOST', /var/log/daemonlog, @loghost)
auth.info\t\t\t\t\tifdef(\\\`LOGHOST', /var/log/authlog, @loghost)
cron.info\t\t\t\t\tifdef(\\\`LOGHOST', /var/log/cronlog, @loghost)\n"\
>> syslog.conf
END_SCRIPT
chown root:sys syslogadd
chmod 744 syslogadd
```

3) Create log files

```
cd /var/log
touch kernlog userlog maillog daemonlog cronlog authlog
chown root:sys kernlog userlog maillog daemonlog cronlog authlog
chmod 600 kernlog userlog maillog daemonlog cronlog authlog
```

4) Run the syslogadd script

```
/etc/syslogadd >> /etc/syslog.conf
```

5)Restart the syslog daemon

```
/etc/init.d/syslog stop
/etc/init.d/syslog start
```

Discussion:

The original configuration file for `syslog` does not log AUTH messages to any files. AUTH messages should be logged to keep track of who logs into the system. The remote log host name should be added to `/etc/hosts` so the remote host name will always be resolved, even if the DNS server is down. The remote log host should be listed in `/etc/hosts` as the log host. A `cron` job can be set up using the `grep` command to separate the two systems' information in `/var/log/authlog`.

5.2 Capture `FTP` and `inetd` connection tracing info

Action (Solaris 8):

```
printf "daemon.debug\t\t\t\t\t/var/log/connlog\n" >> /etc/syslog.conf
touch /var/log/connlog
chown root:root /var/log/connlog
chmod 600 /var/log/connlog
/etc/init.d/syslog stop
/etc/init.d/syslog start
```

Discussion:

If the FTP service is enabled on the system, Appendix B Section 2.3 also enables the "debugging" (`-d`) and connection logging (`-l`) flags to track FTP activity on the system. Similarly, the tracing (`-t`) option to `inetd` was enabled in Section 3.3. All of this information is logged to Syslog, but the Syslog daemon must be configured to capture this information to a file.

The `connlog` file should be reviewed and archived on a regular basis. A sample script for archiving log files is provided as Appendix A to this document. Solaris 9 systems include the `logadm` utility for archiving log files.

5.3 Create `/var/adm/loginlog`

Action (Solaris 8):

```
touch /var/adm/loginlog
chown root:sys /var/adm/loginlog
chmod 600 /var/adm/loginlog
cd /etc/default
awk '/SYSLOG_FAILED_LOGINS=/ \
     { $1 = "SYSLOG_FAILED_LOGINS=0" }; \
     { print }' login >login.new
mv login.new login
chown root:sys login
chmod 444 login
```

Discussion:

If the `loginlog` exists, the file `/var/adm/loginlog` will capture failed login attempt messages (this file does not exist by default). Starting with Solaris 8, administrators may also modify the `SYSLOG_FAILED_LOGINS` parameter in `/etc/default/login` to control how many login failures are allowed before log messages are generated--if set to zero then all failed logins will be logged.

The `loginlog` file should be reviewed and archived on a regular basis. A sample script for archiving log files is provided as Appendix A to this document. Solaris 9 systems include the `logadm` utility for archiving log files.

5.4 Turn on cron logging

Action (Solaris 8):

```
cd /etc/default
awk '/CRONLOG/ { $1 = "CRONLOG=YES" }; \
     { print }' cron > cron.new
mv cron.new cron
chown root:sys cron
chmod 444 cron
```

Discussion:

Setting the `CRONLOG` parameter to YES in `/etc/default/cron` causes information to be logged for every `cron` job that gets executed on the system. Log data can be found in `/var/cron/log` and this file should be reviewed on a regular basis.

5.5 Enable system accounting

Action (Solaris 8):

```
cat <<END_SCRIPT > /etc/init.d/newperf
#!/sbin/sh
/usr/bin/su sys -c \
"/usr/lib/sa/sadc /var/adm/sa/sa\`date +%d\`"
END_SCRIPT
mv /etc/init.d/newperf /etc/init.d/perf
chown root:sys /etc/init.d/perf
chmod 744 /etc/init.d/perf
rm -f /etc/rc2.d/S21perf
ln -s /etc/init.d/perf /etc/rc2.d/S21perf
/usr/bin/su sys -c crontab <<END_ENTRIES
0,20,40 * * * * /usr/lib/sa/sa1
45 23 * * * /usr/lib/sa/sa2 -s 0:00 -e 23:59 -i 1200 -A
END_ENTRIES
```

Discussion:

System accounting gathers baseline system data (CPU utilization, disk I/O, etc.) every 20 minutes. The data may be accessed with the `sar` command (see `man sar` for more information), or by reviewing the nightly report files named `/var/adm/sa/sar*`. Once a normal baseline for the system has been established, unauthorized activity (password crackers and other CPU-intensive jobs, and activity outside of normal usage hours) may be detected due to departures from the normal system performance curve.

Note that this data is only archived for one week before being automatically removed by the regular nightly `cron` job. Administrators may wish to archive the `/var/adm/sa` directory on a regular basis to preserve this data for longer periods.

5.6 Enable kernel-level auditing

Action (Solaris 8):

1) Enable Basic Security Module (BSM)

```
echo y | /etc/security/bsmconv
```

Note: The "y" is used to answer the following question. "Shall we continue with the conversion now? [y/n]"

2) Configure the classes of events to log

```
mkdir -p /var/log/auditlog
mkdir -p /opt/log/auditlog
cd /etc/security
cat << END_PARAMS > audit_control
dir:/var/log/auditlog
flags: lo,ad,ex,fm,-fw,-fc,-fd,na
naflags: lo,ad,ex,fm,-fw,-fc,-fd
minfree:20
/usr/sbin/auditconfig -setpolicy -cnt,argv,arge
# location for log overflow
dir:/opt/log/auditlog
END_PARAMS
```

3) Create a `root` cronjob to force new audit logs daily

```
cd /var/spool/cron/crontabs
crontab -l > root.tmp
echo '0 * * * * /usr/sbin/audit -n' >> root.tmp
crontab root.tmp
rm -f root.tmp
```

4) Reboot system

Note: If `L1-A` is needed, please enable it before the system is rebooted. (See information provided in the Discussion section)

```
init 6
```

5) See CIS Appendix A for log rotation script

Discussion:

Auditing gathers system data about logins and logouts, administrative actions, `exec` system calls, etc. Although auditing may cause some performance degradation, in the event system intrusion does occur, the information obtained from the audit logs will provide very valuable forensic evidence.

When BSM is enabled, the startup scripts for `L1-A` and `vold` are disabled. The `L1-A` feature allows the system administrator to halt the systems. If `L1-A` is needed, comment out the line containing "`abort_enable=0`" in `/etc/system`. The `vold` daemon is used for volume management services. If `vold` is needed, move `/etc/security/spool/S92volmgt` to `/etc/rc2.d/S92volmgt`. If the `minfree` value is reached, the system will begin logging the auditing information in the secondary directory if one is listed.

5.7 Setup Role-Based Access Control

Action (Solaris 8):

1. Set up the audit account role for monitoring the audit logs

- Add `audit` account to `/etc/passwd` file

Note: The following entry should be placed after the `root` entry

```
useradd -d / -g 1 -o -u 0 -s /sbin/sh audit
```

- Add `audit` account information to `/etc/shadow`

```
pwconv
```

- Add entry in `/etc/security/audit_user` to turn off auditing for the `audit` account

```
echo "audit:no:all" >> /etc/security/audit_user
```

- Set password for `audit` account

```
passwd audit
```

2. Make the `audit` account a role

- Add the following line to `/etc/user_attr`

```
echo "audit::::type=role;auths=solaris.audit.;profiles=Audit Control,\
Audit Review" >> /etc/user_attr
```

3. Assign users to the `audit` role

- Add the following line to `/etc/user_attr`

Note: username is the name of the desired `audit` role user. This step should be repeated for each user that needs access to `audit` role.

```
echo "username::::roles=audit;type=normal" >> /etc/user_attr
```

Discussion:

Role Based Access Control (RBAC) assign's user privileges based on least privilege and separation of duty. RBAC allows a system administrator to assign individuals to roles based on their job function. A user can use the "`su`" command to switch to an assigned role.

Note: According to the Basic Security Module Guide, the `audit` account should be placed directly under the `root` entry in the `/etc/passwd` file.

5.8 Confirm permissions on system log files

Action (Solaris 8):

```
chown root:sys /var/log/syslog /var/log/authlog \
/var/adm/loginlog
chown root:root /var/cron/log /var/adm/messages
chmod go-wx /var/log/syslog /var/adm/messages
chmod go-rwx /var/log/authlog /var/adm/loginlog \
/var/cron/log
cd /var/adm
chown root:bin utmpx
chown adm:adm wtmpx
chmod 644 utmpx wtmpx
chown sys:sys /var/adm/sa/*
chmod go-wx /var/adm/sa/*
dir=`awk -F: '($1 == "dir") { print $2 }' \
     /etc/security/audit_control`
chown root:root $dir/*
chmod go-rwx $dir/*
```

Discussion:

It is 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.

Note that sites using the `runacct` script for generating billing reports and other data from the system process accounting logs will notice that the script incorrectly sets the mode on the `wtmpx` file to 664 (adds the "group writability" bit). The local site may wish to "`chmod g-w /var/adm/wtmpx`" after running the `runacct` script. Additional information about how to use `runacct` can be found on SUN `runacct man` page.

6 File/Directory Permissions/Access

6.1 Add 'logging' option to root file system

Action (Solaris 8 and later):

```
awk '($4 == "ufs" && $3 == "/") \
     { $7 = "remount,logging" }; \
     { print }' /etc/vfstab >/etc/vfstab.new
mv /etc/vfstab.new /etc/vfstab
chown root:sys /etc/vfstab
chmod 664 /etc/vfstab
```

Discussion:

A corrupted root file system is one mechanism that an attacker with physical access to the system console can use to compromise the system. By enabling the "logging" option on the root file system, it is much more difficult for the root file system to become corrupted at all, thwarting this particular type of attack. Note that the administrator may also wish to add the "logging" option to other `ufs` type file systems in `/etc/vfstab`. This will help the system to reboot faster in the event of a crash at the cost of some disk overhead (up to a maximum of 64MB per partition) for the file system transaction log file.

6.2 Add 'nosuid' option to /etc/rmmount.conf

Action (Solaris 8 and later):

This action does not usually need to be performed on Solaris 8 and later systems, as it is the default configuration for these platforms..

Action (Solaris 7 and earlier):

```
fs=`awk '($1 == "ident") && ($2 != "pcfs") \
     { print $2 }' /etc/rmmount.conf`
echo mount \* $fs -o nosuid >> /etc/rmmount.conf
```

Discussion:

Removable media is one method by which malicious software can be introduced into the system. By forcing these file systems to be mounted with the "nosuid" option, the administrator prevents users from bringing set-UID programs into the system via CD-ROMs and floppy disks.

6.3 Configure vold.conf to allow users access to CDs only

Action (Solaris 8):

```
awk '($2 == "floppy" || $2 == "/dev/diskette[0-9]*" || $4 == "floppy")\
     {$1 = "#"$1}; { print}'  /etc/vold.conf > /etc/vold.conf.new
mv /etc/vold.conf.new /etc/vold.conf
chown root:bin /etc/vold.conf
chmod 444 /etc/vold.conf
```

Discussion:

Users can use removable media, such as floppy disk, to insert malicious code on the system. By preventing regular users from having access to the floppy drive, there is less of a chance that an exploit will be loaded on the system. Only the root user will be allowed to mount floppy drives.

Note: If a user has access to CD burners, the threat of the user loading an exploit on the system still exist.

6.4 Use full path names in `/etc/dfs/dfstab` file

Action (Solaris 8):

```
cd /etc/dfs
awk '($1 == "share") { $1 = "/usr/sbin/share" }; \
     { print }' dfstab >dfstab.new
mv dfstab.new dfstab
chmod 644 dfstab
chown root:sys dfstab
```

Discussion:

The commands in the `dfstab` file are executed via the `/usr/sbin/shareall` script at boot time, as well as by administrators executing the `shareall` command during the uptime of the machine. It seems prudent to use the absolute pathname to the `share` command to protect against an exploits stemming from an attack on the administrator's `$PATH` environment, etc.

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

Action (Solaris 8):

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

Discussion:

This ensures the correct ownership and access permissions for these files.

6.6 Verify world-writable directories have their sticky bit set

Action (Solaris 8):

Administrators who wish to obtain a list of world-writable directories may execute the following commands:

```
for part in `awk '($4 == "ufs" || $4 == "tmpfs") \
     { print $3 }' /etc/vfstab`
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.

6.7 Find unauthorized world-writable files

Action (Solaris 8):

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

```
for part in `awk '($4 == "ufs" || $4 == "tmpfs") \
     { print $3 }' /etc/vfstab`
do
     find $part -xdev -type f -perm -0002 -print
done
```

Discussion:

Data in world-writable 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.

6.8 Find unauthorized SUID/SGID system executables

Action (Solaris 8):

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

```
for part in `awk '($4 == "ufs" || $4 == "tmpfs") \
     { print $3 }' /etc/vfstab`
do
     find $part -xdev -type f \
     \( -perm -04000 -o -perm -02000 \) -print
done
```

Discussion:

The administrator should take care to ensure that no rogue set-UID programs have been introduced into the system. Information on the set-UID and set-GID applications that normally ship with Solaris systems can be found at http://ist.uwaterloo.ca/security/howto/

6.9 Run fix-modes

Action (Solaris 8):

1. Download the pre-compiled fix-modes software from ftp://ftp.CISecurity.org/pub/pkgs/Solaris/fix-modes.tar.Z

2. Unpack the software and run the fix-modes program

```
zcat fix-modes.tar.Z | tar xf -
cd fix-modes
fix-modes
```

Discussion:

The fix-modes software corrects various ownership and permission issues with files throughout the Solaris OS file systems. This program should be re-run every time packages are added to the system, or patches are applied. Administrators may wish to run the tool periodically out of cron.

Note that the actions above recommend using a pre-compiled version of fix-modes. For sites that wish to build the tool themselves from source, the source code is available from ftp://ftp.science.uva.nl/pub/solaris/fix-modes.tar.gz.

One of the most common methods for an attacker to elevate his privileges is one which an average user has no need to use (e.g. reconfiguring a video card). Thus, many set-UID and set-GID executables can have their SUID/SGID bits removed without any appreciable difference in system usability.

7 System Access, Authentication, and Authorization

7.1 Remove .rhosts support in /etc/pam.conf

Action (Solaris 2.6 or later):

```
cd /etc
grep -v rhosts_auth pam.conf > pam.conf.new
mv pam.conf.new pam.conf
chown root:sys pam.conf
chmod 644 pam.conf
```

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. 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. These services are typically run on ports 512 through 514. Other services may share these port numbers. 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 (Solaris 8):

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

Discussion:

The `/.rhosts`, `/.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.

7.3 Create `/etc[/ftpd]/ftpusers`

Note: The `/etc/ftpusers` file is shipped with Solaris 8 by default. Solaris 9 is also shipped with this file, but the file's path name has been changed to `/etc/ftpd/ftpusers` in this release.

Action (Solaris 8):

```
for user in root daemon bin sys adm lp uucp nuucp \
      smmsp listen nobody noaccess nobody4
do
     echo $user >> /etc/ftpusers
done
chown root:root /etc/ftpusers
chmod 600 /etc/ftpusers
```

Discussion:

`ftpusers` contains a list of users who are not allowed to access the system via FTP. 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. Consider also adding the names of other privileged or shared accounts which may exist on your system such as user `oracle` and the account under which your Web server process runs.

7.4 Create `/etc/shells`

Action (Solaris 8):

```
echo /sbin/sh > /etc/shells
for shell in sh csh ksh bash tcsh zsh; do
     [ -f /bin/$shell ] && echo /bin/$shell >> /etc/shells
     [ -f /usr/bin/$shell ] && \
     echo /usr/bin/$shell >> /etc/shells
done
chown root:root /etc/shells
chmod 644 /etc/shells
```

Discussion:

`/etc/shells` contains a list of valid login shells for user account entries in `/etc/passwd`. If `/etc/shells` does not exist, then any program is valid as a user shell in `/etc/passwd`. It's best to restrict `/etc/shells` to a list of known good shell programs provided by Sun or installed locally by the System Administrator.

Note that it may be necessary to add other locally-installed shells to the `/etc/shells` file (for example if `bash` is installed in `/opt` or `/usr/local/bin`), depending on the needs of a particular organization.

7.5 Prevent remote XDMCP access

Action (Solaris 2.6 or later):

```
mkdir -p /etc/dt/config
cat <<EOXaccess > /etc/dt/config/Xaccess
!*
!*  CHOOSER BROADCAST
EOXaccess
chown root:sys /etc/dt/config/Xaccess
chmod 755 /etc/dt/config
chmod 644 /etc/dt/config/Xaccess
```

Discussion:

The standard GUI login provided on most UNIX systems can act as a remote login server to other devices (including X terminals and other workstations). Access control is handled via the `Xaccess` file--by default under Solaris, this file allows any system on the network to get a remote login screen from the local system. This default behavior can be overwritten in the `/etc/dt/config/Xaccess` file.

7.6 Prevent X server from listening on port 6000/tcp

Action (Solaris 9):

```
if [ -f /etc/dt/config/Xservers ]; then
     file=/etc/dt/config/Xservers
else
     file=/usr/dt/config/Xservers
fi
awk '/Xsun/ && !/^#/ \
     { print $0 " -nolisten tcp"; next }; \
     { print }' $file > $file.new
cp $file.new /etc/dt/config/Xservers
chown root:sys /etc/dt/config/Xservers
chmod 444 /etc/dt/config/Xservers
chown root:bin /etc/dt/config
chmod 755 /etc/dt/config
chown root:bin /usr/dt/config
chmod 755 /usr/dt/config
```

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.

7.7 Set default locking screensaver timeout

Action (Solaris 2.6 and later):

```
for file in /usr/dt/config/*/sys.resources; do
     dir=`dirname $file | sed s/usr/etc/`
     mkdir -p $dir
     echo 'dtsession*saverTimeout: 10' >> $dir/sys.resources
     echo 'dtsession*lockTimeout: 10' >> $dir/sys.resources
done
```

Discussion:

The default timeout is 30 minutes of keyboard/mouse inactivity before a password-protected screen saver is invoked by the CDE session manager. The above action reduces this default timeout value to 10 minutes, though this setting can still be overridden by individual users in their own environment.

7.8 Restrict `at/cron` to authorized users

Action (Solaris 8):

```
cd /etc/cron.d
rm -f cron.deny at.deny
echo root > cron.allow
echo root > at.allow
chown root:root cron.allow at.allow
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 (e.g., the `cron` jobs running as user `sys` for system accounting tasks--see Section 5.5). `cron.allow` only controls administrative access to the `crontab` command for scheduling and modifying `cron` jobs.

7.9 Remove empty `crontab` files and restrict file permissions

Action (Solaris 8):

```
cd /var/spool/cron/crontabs
for file in *
do
     lines=`grep -v '^#' $file | wc -l | sed 's/ //g'`
     if [ "$lines" = "0" ]; then
      rm $file
     fi
done
chown root:sys *
chmod 400 *
```

Discussion:

The system `crontab` files are accessed only by the `cron` daemon (which runs with `root` 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.

7.10 Create appropriate warning banners

Action (for Solaris 2.6 and later):

```
echo "Authorized uses only.  All activity may be monitored and
reported."  > /etc/bannertext
chmod 644 /etc/bannertext
eeprom oem-banner="`cat /etc/bannertext`"
eeprom oem-banner\?=true
cd /etc
cat /etc/bannertext > motd
cat /etc/bannertext > issue
echo "BANNER=\"`cat bannertext` \\\n \\\n\"" > default/telnetd
echo "BANNER=\"`cat bannertext` \\\n \\\n\"" > default/ftpd
for file in /usr/dt/config/*/Xresources
do
     dir=`dirname $file | sed s/usr/etc/`
     mkdir -p $dir
     if [ ! -f $dir/Xresources ]; then
      cp $file $dir/Xresources
     fi
     echo "Dtlogin*greeting.labelString: `cat /etc/bannertext`" >>
$dir/Xresources
     echo "Dtlogin*greeting.persLabelString: `cat /etc/bannertext" >>
$dir/Xresources
done
chown root:sys /etc/motd /etc/dt/config/*/Xresources
chown root:root /etc/issue
chmod 644 /etc/motd /etc/issue /etc/dt/config/*/Xresources
chown root:sys /etc/default/telnetd /etc/default/ftpd
chmod 444 /etc/default/telnetd /etc/default/ftpd
```

Discussion:

Presenting some sort of statutory warning message at login time may 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 from attackers attempting to target specific attacks at a system. Guidelines published by the US Department of Defense require that warning messages include 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 in compliance with local statutes, and that use of the system implies consent to such monitoring. Clearly, the organization's local legal counsel and/or site security administrator should review the content of all messages before the above modifications are made.

Note that if TCP Wrappers are being used to display warning banners for various `inetd`-based services, it is important that the banner messages be formatted properly as not to interfere with the application protocol. The `Banners.Makefile` file provided with the TCP Wrappers source distribution (available from ftp.porcupine.org as well as http://www.sunfreeware.com) contains shell commands to help produce properly formatted banner messages.

7.11 Restrict `root` logins to system console

Action (Solaris 8):

```
cd /etc/default
awk '/CONSOLE=/ { print "CONSOLE=/dev/null"; next }; \
     { print }' login > login.new
mv login.new login
chown root:sys login
chmod 444 login
```

Discussion:

Setting the `CONSOLE` variable to `/dev/null` prevents `root` logins from the console. Administrators will have to log into the system as themselves and then '`su`' to `root`. If the system is in single user mode, the user will be allowed to log in as `root`.

The administrator should access the system via a privileged 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 a problems.

7.12 Limit number of failed login attempts

Action (Solaris 8):

```
cd /etc/default
if [ "`grep RETRIES= login`" ]; then
     awk '/RETRIES=/ { $1 = "RETRIES=3" }
        { print }' login > login.new
     mv login.new login
     chown root:sys login
     chmod 444 login
else
     echo RETRIES=3 >> login
fi
```

Discussion:

The RETRIES parameter is the number of failed login attempts a user is allowed before being disconnected from the system and having to re-initiate their login session. Setting this number to a reasonably low value helps discourage brute force password guessing attacks.

7.13 Set EEPROM security-mode and log failed access

Action (Solaris 8 SPARC-based systems):

```
eeprom security-#badlogins=0
cd /var/spool/cron/crontabs
crontab -l >root.tmp
echo "0 0,8,16 * * * /usr/bin/logger -p auth.info \
\`/usr/sbin/eeprom security-#badlogins\`" >> root.tmp
crontab root.tmp
rm -f root.tmp
eeprom security-mode=command
```

Discussion:

After entering the last command above, the administrator will be prompted for a password. This password will be required to authorize any future command issued at boot-level on the system (the `ok' or `>' prompt) except for the normal multi-user boot command (i.e., the system will be able to reboot unattended). This helps prevent attackers with physical access to the system console from booting off some external device (such as a CD-ROM or floppy) and subverting the security of the system.

Note that the administrator should write down this password and place the password in a sealed envelope in a secure location (note that locked desk drawers are typically not secure). If the password is lost or forgotten, simply run the command "`eeprom security-mode=none`" as root to erase the forgotten password, and then set a new password with "`eeprom security-mode=command`".

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 (Solaris 8):

```
passwd -l sys
passwd -l daemon
for user in adm bin lp smmsp nobody noaccess \
     uucp nuucp smtp listen nobody4; do
     passwd -l $user
     /usr/sbin/passmgmt -m -s /dev/null $user
done
```

Discussion:

Accounts that are not being used by regular users should be locked. Not only should the `password` field for the account be set to an invalid string, but also the shell field in the `/etc/password` file should contain 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 Assign `noshell` for system accounts

Action (Solaris 8):

1. Create `noshell` script

```
cat <<END_SCRIPT > /sbin/noshell
#!/sbin/s
#
# This script was taken from JASS noshell program
#

trap "" 1 2 3 4 5 6 7 8 9 10 12 15 19

PATH=/usr/bin:/usr/sbin:$PATH; export PATH

HOSTNAME="`/bin/uname -n`"
USER="`/bin/id | /bin/awk '{print \$1}'`"

/bin/logger -i -p auth.err \
  "Attempted access by \${USER} on host \${HOSTNAME}"

wait
exit
END_SCRIPT
chown root:root /sbin/noshell
chmod 744 /sbin/noshell
```

2. Stop then restart `syslog` daemon

```
/etc/init.d/syslog stop
/etc/init.d/syslog start
```

Discussion:

If the system passwords were locked in a previous step, the `noshell` script will not work for those accounts. If accounts are not locked or have a password setting "`no passwd; setuid only`", the shell can be set to use `/sbin/noshell` which will cause a error to appear in `/var/log/syslog`. The script will log all attempts to switch user to a system account. The script listed above is taken from the SUN JASS script for `noshell`.

Note: The `noshell` script should not be used on the `root` account.

8.3 Set account expiration parameters on active accounts

Action (Solaris 8):

```
logins -ox |awk -F: '($1 == "root" || $1 == “audit” || $8 == "LK")
     { next }
     { $cmd = "passwd" }
     ($11 <= 0 || $11 > 91)  { $cmd = $cmd " -x 91" }
     ($10 < 7)               { $cmd = $cmd " -n 7" }
     ($12 < 28)              { $cmd = $cmd " -w 28" }
     ($cmd != "passwd")      { print $cmd " " $1 }' \
> /etc/NSAupd_accounts
/sbin/sh /etc/NSAupd_accounts
rm -f /etc/NSAupd_accounts
cat <<EO_DefPass > /etc/default/passwd
MAXWEEKS=13
MINWEEKS=1
WARNWEEKS=4
PASSLENGTH=6
EO_DefPass
```

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` and `audit` accounts) to force password changes every 91 days (13 weeks), and then prevent password changes for seven days (one week) thereafter. Users will begin receiving warnings 28 days (4 weeks) 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).

These are recommended starting values, but sites may choose to make them more restrictive depending on local policies. Note that due to the fact that `/etc/default/passwd` sets defaults in terms of number of weeks (even though the actual values on user accounts are kept in terms of days), it is probably best to choose interval values that are multiples of 7.

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

Action (Solaris 8):

The following command should return no lines of output

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

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 are no longer required on Solaris systems, but may exist in files that have been imported from other platforms. 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 there are no accounts with empty password fields

Action (Solaris 8):

The following command should return no lines of output

```
logins -p
```

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*".

8.6 Verify that no UID 0 accounts exist other than root and audit

Action (Solaris 8):

The following command should return only the words root and audit.

```
logins -o | awk -F: '($2 == 0) { print $1 }'
```

Discussion:

Any account with UID 0 has superuser privileges on the system. The only superuser account on the machine should be the root and audit accounts, and they should be accessed by logging in as an unprivileged user and using the su command to gain additional privileges.

Finer granularity access control for administrative access can be obtained by using the freely-available sudo program (http://www.courtesan.com/sudo/) or Sun's own Role-Based Access Control (RBAC) system. For more information on Solaris RBAC, see http://wwws.sun.com/software/whitepapers/wp-rbac/.

8.7 Disallow '.' or group/world-writable directory in `root $PATH`

Action (Solaris 8):

```
for dir in `logins -ox | \
    awk -F: '($1 == "root") { print $6 }'`
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:

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

8.8 Set user home directories to mode 750 or more restrictive

Action (Solaris 8):

```
for dir in `logins -ox | \
     awk -F: '($8 == "PS" && $1 != "root" && $1 != "audit") \
     { print $6 }'`
do
     chmod 750 $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.

8.9 Disallow group/world-writable user dot-files

Action (Solaris 8):

```
for dir in `logins -ox | \
    awk -F: '($8 == "PS") { print $6 }'`
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:

Group or 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.

8.10 Change user's `.forward` file to mode 600

Action (Solaris 8):

1. Create script to check for `.forward` file in home accounts

```
cat <<END_SCRIPT > /etc/forward
#!/bin/sh
for userhome in \`awk -F: '(\$7 != "/sbin/sh" && \
    \$7 != " " && \$7 != "/usr/lib/uucp/uucico" && \
    \$6 != "/" && \$6 != "/var/adm" && \$6 !~/usr/)\
    { print \$6 }' /etc/passwd\`
do
    if [ -f \$userhome/.forward ]; then
      /bin/logger -i -p user.info \
          "Changed the .forward permission for \$userhome"
      ls -al \$userhome/.forward > forwardls.new
      for username in \`awk '(\$1 != "-rw-------") \
          { print \$3 }' forwardls.new\`
      do
          chmod 600 \$userhome/.forward
          mail -m .forward \$username < /etc/permchange
      done
      rm forwardls.new
    else
      echo ".forward file does not exist for \$userhome"
    fi
done
END_SCRIPT
chown root:sys /etc/forward
chmod 700 /etc/forward
```

2. Create the email message to send to users

```
echo "The permissions on the .forward file for this account were \
changed by an administrator." > /etc/permchange
chown root:sys /etc/permchange
chmod 744 /etc/permchange
```

3. Add the following line to `/etc/syslog.conf`

```
printf “user.info\t\t\t\t\t/var/log/forward\n” >> /etc/syslog.conf
```

4. Create `/var/log/forward`

```
touch /var/log/forward
chown root:sys /var/log/forward
chmod 600 /var/log/forward
```

5. Stop then restart `syslog` daemon

```
/etc/init.d/syslog stop
/etc/init.d/syslog start
```

6. Run the forward script

```
/etc/forward
```

Discussion:

The `.forward` file should not be world or group writable. If the `.forward` file is world/group-writable, an attacker could use this file to embed scripts on the system that may contain exploits. The exploit can then be used to gain `root` access.

8.11 Remove user `.netrc` files

Action (Solaris 8):

```
for dir in `logins -ox | \
     awk -F: '($8 == "PS") { print $6 }'`
do
     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.

8.12 Set default UMASK for users

Action (Solaris 8):

```
cd /etc/default
awk '/UMASK=/ { $1 = "UMASK=077" }; \
     { print }' login > login.new
mv login.new login
chown root:sys login
chmod 444 login
echo UMASK=077 >> ftpd
echo umask 077 >> /etc/profile
echo umask 077 >> /etc/.login
```

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 his/her 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.

8.13 Set "mesg n" as default for all users

Action (Solaris 8):

```
echo mesg n >> /etc/profile
echo mesg n >> /etc/.login
```

Discussion:

"mesg n" blocks attempts to use the write or talk commands to contact the user at their terminal, but has the side effect of slightly strengthening permissions on the user's tty device. Since write and talk are no longer widely used at most sites, the incremental security increase is worth the loss of legacy functionality.

8.14 Change root's home directory

Action (Solaris 8):

```
su root
mkdir /root
mv /.* /root/.
passmgmt -m -h /root root
passmgmt -m -h /root audit
chmod 700 /root
```

Discussion:

Changing `root`'s home directory (as well as `audit`'s) aids in system administration as well as provides a small obfuscation to someone who attempts to gain unauthorized access to the `root` account. The system administrator's personal files should be kept in `/root` so as to provide a clear separation of what files are and are not part of the system software. A benefit is that these private files and their contents will not be visible to non-root users. This change of home directory could also serve to confuse any automated script that assumes root access begins with the "`/` " directory.

Note: This change may confuse already configured programs such as Netscape. Either use these programs from a non-root user or delete configuration files and reinitialize the program when logged in as `root`.

8.15 Setup user file quotas

Action (Solaris 8):

1. Set up UFS file system(s) for quotas (where mount_point is the file system on which quotas are to be set).

```
cd /mount_point
touch quotas
chown root:root quotas
chmod 600 quotas
cd /etc
awk '($3 == "mount_point" && $4 == "ufs") \
     { $7 = $7 ",rq"; print; next; } \
     { print; }' vfstab > vfstab.new
mv vfstab.new vfstab
chown root:sys vfstab
chmod 664 vfstab
edquota -t mount_point
fs mount_point blocks time limit = number time_unit, files time limit = number time_unit
```

2. Establish and enable quotas for users, where proto_user is the prototype user for other users

```
edquota proto_user
fs mount_point blocks (soft=soft_lim, hard=hard_lim) inodes (soft=soft_lim2, hard=hard_lim2)
edquota -p proto_user user_1 user_2
quotacheck -v -a
quotaon -v file_system_1 file_system_2
```

3. View user quota usage

```
repquota -v -a
```

Discussion:

Quotas are established to prevent user files from consuming all available hard drive disk space. Only the `root` user can create or edit quotas. The hard limit is the absolute maximum amount a user can consume and once it is reached, the user cannot create new files, edit old files, compile programs, etc. The soft limit is the maximum that the administrator would prefer. Once the soft limit is exceeded the system warns the user and starts the grace period, usually between 5 and 9 days. During this time, the user is still able to perform file operations that exceed the soft limit but not the hard limit. When the grace period ends, the soft limit is enforced as a hard limit. Disk quotas should be enforced on file systems used for mail (ex. `/var/spool/mail`), user home directories (ex. `/export/home`), and temporary files (ex. `/tmp`). The administrator must choose which file systems need quotas, the appropriate soft time limit (no more than two weeks), which users should have quotas enforced, and the appropriate soft and hard limits. See `man edquota` for explanation of red colored variables.

Appendix A: Log Rotation Script

```
#!/bin/ksh

# rotate -- A script to roll over log files
# Usage: rotate /path/to/log/file [mode [#revs] ]

FILE=$1
MODE=${2:-644}
DEPTH=${3:-4}

DIR=`dirname $FILE`
LOG=`basename $FILE`
DEPTH=$(($DEPTH - 1))

if [ ! -d $DIR ]; then
echo "$DIR: Path does not exist"
exit 255
fi
cd $DIR

while [ $DEPTH -gt 0 ]
do
OLD=$(($DEPTH - 1))
if [ -f $LOG.$OLD ]; then
mv $LOG.$OLD $LOG.$DEPTH
fi
DEPTH=$OLD
done

if [ $DEPTH -eq 0 -a -f $LOG ]; then
mv $LOG $LOG.0
fi

cp /dev/null $LOG
chmod $MODE $LOG

/etc/init.d/syslog stop
/etc/init.d/syslog start
```

Appendix B:

2.2 Only 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 action below.

Action (Solaris 8):

```
cd /etc
sed 's/^#telnet/telnet/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

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 (see http://www.openssh.com/) provide encrypted network logins and should be used instead.

2.3 Only 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 scp?

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

Action (Solaris 8):

```
awk '!/^#ftp/ { print }
     /^#ftp/ { $1 = "ftp"; print $0 " -d -l" }' \
inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

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 Section 7.3.

Note that if the FTP daemon is left on, it is recommended that the "debugging" (-d) and connection logging (-l) flags also be enabled to track FTP activity on the system. Information about FTP sessions will be logged via `Syslog`, but the system must be configured to capture these messages. For further configuration information, see Section 5.2. If `ftp` is not needed to transfer files, turn it off. Ftp sends the password in the clear.

If FTP must used, enable logging and use `/etc/ftpusers` to identify which users are not allowed to use `ftp`. Also, if users must write to the server, make sure users can only write to a specific area of the disk.

2.4 Only 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 (Solaris 8):

```
sed 's/^#shell/shell/; s/^#login/login/' \
inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

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 7.1 for additional security-related configuration settings.

2.5 Only 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 either part of this question is yes, proceed with the actions below.

Action (Solaris 8):

```
sed 's/^#tftp/tftp/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
mkdir p m 711 /tftpboot
chown root:root /tftpboot
```

Discussion:

TFTP is typically used for network booting of diskless workstations, X-terminals, and other similar devices (TFTP is also used during network installs of systems via the Solaris Jumpstart facility). 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 Only enable printer service if absolutely necessary

Question:

Is this machine a print server for your network?

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

Action (Solaris 2.6 and later):

```
sed 's/^#printer/printer/' inetd.conf >inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:

`in.lpd` provides a BSD-compatible print server interface. Even machines that are print servers may wish to leave this service disabled if they do not need to support BSD-style printing.

2.7 Only enable rquotad if absolutely necessary

Question:

Is this system an NFS file server with disk quotas enabled?

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

Action:

```
sed 's/^#rquotad/rquotad/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:

rquotad allows NFS clients to enforce disk quotas on file systems that are mounted from the local system. If your site does not use disk quotas, then you may leave the rquotad service disabled.

2.9 Only enable Solaris Volume Manager daemons if absolutely necessary

Question:

Is the Solaris Volume Manager GUI administration tool required for the administration of this system?

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

Action (Solaris 9 systems or systems which have the Solaris Volume Manager or Solaris DiskSuite products installed):

```
sed 's/^#100229/100229/;
     s/^#100230/100230/;
     s/^#100242/100242/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:

The Solaris Volume Manager (formerly Solaris DiskSuite) provides software RAID capability for Solaris systems. This functionality can either be controlled via the GUI administration tools provided with the operating system, or via the command line. However, the GUI tools cannot function without several daemons enabled in inetd.conf. Since the same functionality that is in the GUI is available from the command line interface, administrators are strongly urged to leave these daemons disabled and administer volumes directly from the command line.

2.10 Only enable Kerberos-related daemons if absolutely necessary

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 (Solaris 2.6 and later):

```
sed 's/^#kerbd/kerbd/;
      s/^#100134/100134/;
      s/^#100234/100234/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:

With the release of Solaris 8, Kerberos support has been added to Solaris. However, Kerberos may not be in use at all sites. For more information on Kerberos see http://web.mit.edu/kerberos/www/.

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/

Sun Microsystems

Patches and related documentation:
ftp://sunsolve.sun.com/pub/patches/

Sun Patch Manager tool:
http://www.sun.com/service/support/sw_only/patchmanager.html

Kernel (ndd) settings for network-layer security:
http://www.sun.com/software/solutions/blueprints/1200/network-updt1.pdf

Role-Based Access Control (RBAC) white paper:
http://wwws.sun.com/software/whitepapers/wp-rbac/

OpenSSH white paper:
http://www.sun.com/solutions/blueprints/0701/openSSH.pdf

NTP white paper:
http://www.sun.com/solutions/blueprints/0701/NTP.pdf

Other Miscellaneous Documentation

Various documentation on Solaris security issues:
http://ist.uwaterloo.ca/security/howto/

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/
http://www.deer-run.com/~hal/dns-sendmail/DNSandSendmail.pdf

Software

Pre-compiled software packages for Solaris:
http://www.sunfreeware.com/
ftp://ftp.cisecurity.org/

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

TCP Wrappers source distribution:
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/

fix-modes (free tool to correct permissions and ownerships in the Solaris OS):
ftp://ftp.science.uva.nl/pub/solaris/fix-modes.tar.gz

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