I331 Technical Report
I331-007R-2004

Guide to the Secure Configuration of Solaris 9

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

Dated: 16 July 2004
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 9 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.
- Downloaded information and utilities are valid as of 1 July 2004. Newer versions have not been tested for this guide.

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

ABSTRACT

This document provides additional security measures beyond those specified in the Center for Internet Security (CIS) *Solaris Benchmark*. The document was developed to provide system administrators with steps to create a more secure Solaris 9 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 9. 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 multiple versions of Solaris, the additional information provided by the National Security Agency (NSA) only applies to Solaris 9. 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 Items build upon previous Items.**

The information in the CIS document is the collaborative work of several agencies, including the NSA, colleges, and company representatives. The NSA configuration guide is comprised of industry best practices, academic expertise, practical experience, and Solaris 9 documentation.

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.

System Configuration
This guide was tested on an newly configured system with the End User Cluster (SUNWCuser) installed. Several of the items in this guide require installing additional packages that are found in the SUNWCall cluster but not the SUNWCuser cluster. These packages are: SUNWhea (header files), SUNWsprot (make utility), SUNWsprox (SPARCv9 libaries for make utility). For compiling software, the following packages are required in addition to installing gcc: SUNWgcmn, SUNWarc, SUNWarcx (for 64-bit systems), and SUNWbtool. For System Accounting, the following packages are required: SUNWaccu and SUNWaccr. Also, several of the servers enabled in Chapter 3 are only applicable if they were previously installed. These include kerberos, ldap, http, and dhcp servers.

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 **strongly recommended** that administrators make backup copies of critical configuration files that may get modified by various benchmark items. If this step is not performed, then the site may have no reasonable back-out strategy for reversing system modifications made as a result of this document. The script provided in Appendix A of this document will automatically back up all files that may be modified by the actions below, except for the boot scripts manipulated by the various items in Chapter 3 of this document, which are backed up automatically by the individual items in Chapter 3. This guide is intended for configuration of a new system. For older systems, a full backup may be appropriate.

This page is intentionally left blank

1 Patches and Additional Software

1.1 Partition hard drive to compartmentalize data

Action:
Keeping their uses in mind, create the following partitions during the install process. The number of configurable disk slices is limited to seven on a SPARC platform and nine on the Intel platform. However, Solaris 9 allows for soft partitioning which can be used to subdivide disks into as many as 8192 logical volumes. Slices that are used for soft partitions cannot be used for other purposes.

The following disk slices are commonly used and should be created on the system.
`/`
files and directories that make up the operating system; once installed, very little is added to this directory.

`swap`
at a minimum, this should be 512 MB; a good rule is to make swap equivalent to RAM size unless large loads are anticipated, in which case a setting of 1.5 times fast memory is appropriate for standard applications (e.g. `ls, lp, vi,` etc.). The swap partition is typically mounted as `/tmp`.

`/usr`
documentation, system programs, and library routines

These directory names are commonly used and hard or soft partitions should be created accordingly.

`/var`
for logging; when using Basic Seurity Module (BSM), logging data can grow quite quickly so make sure this partition is sufficiently large in size. If using the `savecore` feature, allocate at least twice as much space as there is physical memory for this partition.

`/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.)

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.

The administrator must already have a plan for what size each hard partition must be. This requires knowledge of which software cluster is needed, the system's intended use, and who will use it. Soft partitions can be enlarged after creation if space is needed, as long as space is available on the underlying device. Once enlarged, they cannot be reduced.

Information on planning for disk space can be found in the System Administration Guide: Basic Administration book. Information on creating soft partitions can be found in the Solaris Volume Manager Administration Guide. Both of these can be found on the Sun documentation site `http://docs.sun.com`

1.2 Apply latest OS patches

Action:
1. Download Sun Recommended Patch Cluster into `/var/sadm`.

Sun Recommended Patch clusters can be downloaded via FTP or HTTP from
`http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access`

Patchfinder, Patch Reports, Recommended Patch Cluster README files, and Y2K Patch Clusters can also be accessed from this site.
`ftp://sunsolve.sun.com/patchroot/clusters`
Look for 9_Recommended.zip.

2. Execute the following commands:

```
cd /var/sadm
unzip -qq 9_Recommended.zip
cd 9_Recommended
./install_cluster -q
```

By default, when `./install_cluster` is run, it checks if sufficient disk space exists for the installation of the Patch Cluster. If there is insufficient space, the user can abort the install. The "`-q`" (quiet) option supresses this interactive option. It's recommended that the `CLUSTER_README` file be read for details.

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. In addition to installing the Solaris Recommended Patch Clusters as described above, administrators may wish to also check the Solaris 9 patch report file, `9_patch_report`, (available from the same HTTP 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, PatchPro Interactive, and PatchPro Expert. It's recommended that system administrators research these tools to determine which, if any, should be implemented. For more information on each of these tools, visit the SunSolve Patch Portal (`http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage`). For information on Solaris 9 patch management, including a comparison of patch management tools, see Chapters 15 and 16 of System Administration Guide: Basic Administration, part of the Solaris 9 9/04 System Admistrator Collection (`http://docs.sun.com/app/docs/doc/817-6958`).

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/<cluster name>_log`.

1.3 Install TCP Wrappers

Action:
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. Turn on TCP Wrappers

```
cd /etc/default
awk '/#ENABLE_TCPWRAPPERS/ { $1 = "ENABLE_TCPWRAPPERS=YES" }; \
     { print }' inetd > inetd.new
mv inetd.new inetd
chown root:sys inetd
chmod 444 inetd
pkill -HUP -u 0 -P 1 -x inetd
```

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

The above actions will only provide filtering on standard TCP-based services that are spawned by inetd. To protect UDP and RPC-based services that are spawned by inetd, consider implementing a host-based firewall such as Sun's SunScreen software, which is bundled into the Solaris 9 operating system, or ipfilter (see Item 4.7). See the documentation provided with the TCP Wrappers source code release for information on using TCP Wrappers-style filtering with stand-alone daemons that are not spawned out of inetd.

1.4 Reference system random number generator

Action:
For applications needing random numbers, configure them to use /dev/random or /dev/urandom, as appropriate.

Discussion:
Solaris 9 ships with random number generator devices /dev/random and /dev/urandom. These are preferrable to generators such as prngd, which are not native to the operating system.

1.5 Configure IPsec

IPsec is a network layer protocol that employs a robust set of security mechanisms in order to secure network traffic. IPsec consists of two network packet protocols: the Authentication Header (AH) and the Encapsulating Security Payload (ESP).

Authentication is accomplished by using either the MD-5 or the SHA-1 algorithms to produce an integrity checksum based on the data and the key. The Authentication Header provides strong integrity, data authentication and partial sequence integrity (replay protection).

The Encapsulating Security Payload uses the DES (Data Encryption Standard), 3DES (Triple DES), or AES (Advanced Encryption Standard) encryption algorithms to provide data confidentiality and traffic analysis protection. In addition, the ESP is capable of providing authentication (There is some overlap in the functionality of AH and ESP).

Because IPsec operates on the network layer, it is transparent to network applications and protects all traffic including TCP, UDP, and ICMP.

More informaton on Solaris IPsec can be found in the Solaris System Administration Guide: IP Services. It can be downloaded from the Sun Microsystems website at:
`http://docs.sun.com/db/doc/806-4075`

Additionally, the "IPsec in the Solaris 9 Operating Environment" whitepaper can be downloaded from the Sun Microsystems website at:
`http://www.sun.com/software/solaris/9/whitepapers.html`

Install Solaris 9 Data Encryption Supplement

In order to use the AES encryption algorithm, it is necessary to install the Solaris 9 Data Encryption Supplement. Encryption algorithms DES and 3DES are provided as part of the base Solaris 9 installation.

Action:
0. Before installing the Solaris 9 Data Encryption Supplement, verify that the necessary packages are not already installed. The following command can be used:

```
for ver in SUNWcry SUNWcry64 SUNWcryrx SUNWcryr ;
do echo $ver VERSION `pkginfo -x $ver | sed 1d | awk '{ print $2 }' \
| cut -f1 -d,`; done
```

If each of the four packages are VERSION 11.9.0 or higher, then the AES and encryption algorithm is already installed. If not, then the Solaris 9 Data Encryption Supplement must be downloaded and installed.

1. Download the Solaris 9 Data Encryption Supplement. It can be downloaded from the Sun Microsystems website at:
`http://www.sun.com/download/products.xml?id=3e3af5a6`

2. Once downloaded, install the necessary packages:

```
unzip sol-9-sparc-crypto.zip
pkgadd -d sol-9-sparc-crypto/Encryption_9/sparc/Packages all
```

The "all" option will install all packages available in the Solaris 9 Data Encryption Supplement.

3. Remove the package file after installation:

```
rm -f sol-9-sparc-crypto.zip
rm -rf ./sol-9-sparc-crypto/
```

IPsec Configuration

Solaris IPsec provides various means of protecting network traffic. It can protect all traffic between two hosts, protect individual services, be used as a Virtual Private Network (VPN) and also perform simple packet filtering. The following is a procedure to secure all traffic between two IPv4 hosts using ESP (using its own authentication) with shared keys. In this example, traffic between 10.1.1.2 (testbox1) and 10.1.1.3 (testbox2) will be secured. This procedure is intended to secure traffic between two Solaris 9 hosts. In order to use this procedure on a Solaris 8 host, the Solaris 8 optional encyption packages must be downloaded and installed. They can be downloaded from the Sun Microsystems website at:
`http://www.sun.com/software/solaris/encryption/download.html`

The Solaris 8 optional encryption packages provide the DES and 3DES encryption algorithms for use with IPsec. These algorithms can be used to secure traffic between a Solaris 8 and a Solaris 9 host.

Action:
1. Configure the Security Policy:

For security purposes, this procedure should be carried out when logged in as superuser on the system console.

The following commands should be run to create the security policy file. This can be any file; however, it must have the correct ownership and file permissions (see below). For this example, `/etc/inet/ipsec.pol` will be used.

On the first host (testbox1), run the following commands:

```
cat <<EOF>> /etc/inet/ipsec.pol
{ saddr 10.1.1.2 daddr 10.1.1.3 } apply \
{ encr_algs 3des encr_auth_algs md5 sa shared }
{ saddr 10.1.1.3 daddr 10.1.1.2 } permit \
{ encr_algs 3des encr_auth_algs md5 }
EOF
```

Set the file's ownership and permissions:

```
chown root:root /etc/inet/ipsec.pol
chmod 600 /etc/inet/ipsec.pol
```

The `/etc/inet/ipsec.pol` file will now read as follows:

```
{ saddr 10.1.1.2 daddr 10.1.1.3 } apply { encr_algs 3des encr_auth_algs
md5 sa shared }
{ saddr 10.1.1.3 daddr 10.1.1.2 } permit { encr_algs 3des
encr_auth_algs md5 }
```

The first line specifies the policy for outgoing traffic. It indicates that all traffic with a source IP of 10.1.1.2 and a destination IP of 10.1.1.3 will use the 3DES encryption algorithm, the MD-5 authentication algorithm, and used shared keys. The second line specifies the policy for incoming traffic. It indicates that all traffic with a source IP of 10.1.1.3 and a destination IP of 10.1.1.2 must use the 3DES encryption algorithm and the MD-5 authentication algorithm.

On the second host (testbox2), run the following commands:

```
cat <<EOF>> /etc/inet/ipsec.pol
{ saddr 10.1.1.3 daddr 10.1.1.2 } apply \
{ encr_algs 3des encr_auth_algs md5 sa shared }
{ saddr 10.1.1.2 daddr 10.1.1.3 } permit \
{ encr_algs 3des encr_auth_algs md5 }
EOF
```

Set the file's ownership and permissions:

```
chown root:root /etc/inet/ipsec.pol
chmod 600 /etc/inet/ipsec.pol
```

2. Generate random keys:

The strength of encryption relies on the quality of random key generation. Solaris provides the `/dev/random` pseudo-device to generate random keys for encryption purposes. Four different keys must be generated; one for the AH, one for the ESP and one for each Security Parameters Index (SPI). The Security Parameters Index is a random 32-bit (8 hex digit) number that specifies to the device recieiving the packet which Security Association (SA) to use. This SA contains contains the necessary information on how the receiving device will decrypt the packet.The SPI cannot be encrypted within the packet because the receiving machine must use this value to determine the correct SA to utilize.

To generate the keys for the AH and the ESP, the following command is used:

```
od -x -A n -N 48 </dev/random | sed 's/ //g' \
| awk '{printf("%s\n",$1)}'
```

The output will be 96 (pseudo) random hexadecimal characters similar to:

```
cbf503c4d505d3c8254aa12fe0ef941d
48078bfa312893bbb7b0ac133449f71f
7d8a4f32128d6298f37c3e44057032a2
```

Each algorithm requires a key of a specified length. The number of characters used from the above output is dependent on the algorithm used. Below is a list of the keylengths for each authentication and encryption algorithm:

MD-5	(128 bit)	32 characters
SHA-1	(160 bit)	40 characters
DES	(64 bit)	16 characters
3DES	(192 bit)	48 characters
AES	(128,192,256 bit)	32,48,68 characters

For example, if MD-5 were to be used as the authentication algorithm, the following 32 character string could be used from the output above for the key:

```
48078bfa312893bbb7b0ac133449f71f
```

To generate the keys for each SPI, the following command is used:

```
od -An  -N4 </dev/random | sed 's/ //g' | awk '{printf("%.8s\n",$1)}'
```

The output will be eight (pseudo) random octal characters similar to:

```
06075310
```

The SPI has a keylenth of 8 octal characters. Each SPI must be unique. Run this command once for each unique SPI required. In this example two unique SPIs are required.

Each of these keys, as they must be identical on each host for IPsec to function properly.

3. Configure the Security Association:

Add the following two lines to the security association file. This can be any file, however, it must have the correct ownership and file permissions. For this example, use `/etc/inet/ipsec.sa`.

Run the following command on each host:

```
cat <<EOF>> /etc/inet/ipsec.sa
# From 10.1.1.2 to 10.1.1.3
# SPI: <Unique 8 character SPI>
# Auth Alg: MD-5
# Auth Key: <32 Hex digit MD-5 key>
# Encr Alg: 3des
# Encr Key: <48 Hex digit 3des key>
add esp spi <Unique 8 character SPI> src testbox1 dst testbox2 \
auth_alg md5 encr_alg 3des authkey <32 Hex digit MD-5 key> \
encrkey <48 Hex digit 3des key>
# From 10.1.1.3 to 10.1.1.2
# SPI: <Unique 8 character SPI> //Different SPI from above
# Auth Alg: MD-5
# Auth Key: <32 Hex digit MD-5 key> //same MD-5 key from above
# Encr Alg: 3des
# Encr Key: <48 Hex digit 3des key> //same 3DES key from above
add esp spi <Unique 8 character SPI> src testbox2 dst testbox1 \
auth_alg md5 encr_alg 3des authkey <32 Hex digit MD-5 key> \
encrkey <48 Hex digit 3des key>
EOF
```

Note: In the above action, choose the key sequences from the 96 character hexadecimal ouput from step 2. Also, use the same authentication key throughout as well as the same Encryption key throughout the above action. Use unique SPI keys (total of two in this example) throughout.

On each host, set the file's ownership and permissions:

```
chown root:root /etc/inet/ipsec.sa
chmod 600 /etc/inet/ipsec.sa
```

4. Enable IPsec at boot time:

On each host, add the following to `/etc/init.d/ipsec`:

```
cat <<EOF>> /etc/init.d/ipsec
#!/bin/sh
#Startup script for IPsec.
case "\$1" in
start)
     /usr/sbin/ipsecconf -f
     /usr/sbin/ipseckey flush
     /usr/sbin/ipseckey -f /etc/inet/ipsec.sa
     /usr/sbin/ipsecconf -a /etc/inet/ipsec.pol
     ;;
stop)
     /usr/sbin/ipseckey flush
     /usr/sbin/ipsecconf -f
     ;;
*)
     echo "Usage: \$0 { start | stop }"
     exit 1
     ;;
esac
exit 0
EOF
```

Link the script to the startup directory and set the ownership and file permissions:

```
ln -s /etc/init.d/ipsec /etc/rc2.d/S69ipsec
chown root:sys /etc/init.d/ipsec
chmod 700 /etc/init.d/ipsec
```

IPsec will now be enabled at boot time and protect all traffic between 10.1.1.2 (testbox1) and 10.1.1.3 (testbox2).

Additional information:
To manually start and stop IPsec. Use the following command:

```
/etc/rc2.d/S69ipsec { stop | start }
```

To display the current Security Policy, use the following command:

```
ipsecconf -l
```

To display the current Security Association, use the following command:

```
ipseckey dump
```

Discussion:
This is the manual method for configuration of IPsec. Solaris 9 now employs it's own key management facility: Internet Key Exchange (IKE). This method can also be used to configure IPsec on Solaris 9. See the Sun documentation IPsec and IKE Administration Guide (`http://docs.sun.com/app/docs/doc/817-2694`) for the appropriate Solaris 9 release for more information.

It's recommended that keys be changed regularly to decrease the impact of compromised keys. This can be accomplished by manually editing the new keys into the Security Association file on each machine and restarting the service.

1.6 Configure SSH Server

Action:
The following script is intended to modify the default `sshd_config` file installed with Solaris 9.

```
touch /etc/issue
cd /etc/ssh
/etc/init.d/sshd stop
if [ ! -f /etc/hostname6.* ]; then
  nawk ' /#ListenAddress 0\.0\.0\.0/ { sub(/^#/,"") }; \
         /ListenAddress ::/ { $1 = "#ListenAddress" }; \
     { print }' sshd_config > sshd_config.new
mv sshd_config.new sshd_config
fi

nawk '/#Banner/   { sub(/^#/,""); $2 = "/etc/issue" }; \
      /#IgnoreUserKnownHosts/ { sub(/^#/,""); $2 = "yes" }; \
      /KeyRegenerationInterval/ { $2 = "1800" }; \
      /LoginGraceTime/ { $2 = "60" }; \
      /ServerKeyBits/ { $2 = "1024" }; \
     { print }' sshd_config > sshd_config.new
echo "KbdInteractiveAuthentication no" >> sshd_config.new
mv sshd_config.new sshd_config
chown root:sys sshd_config
chmod 600 sshd_config
/etc/init.d/sshd start
```

Discussion:
Sun developed a version of SSH based on OpenSSH and began bundling it with their Solaris operating system as of the release of Solaris 9. Sun's version has many of the same configuration options as in OpenSSH, though privilege separation, a feature that contributes to the security of the system through use of an unprivileged process, is still not available.

The configuration options above do not include the options AllowGroups, DenyGroups, AllowUsers, and DenyUsers. Any one, but only one, of these options can be used to specify an access control list. It is strongly recommended that one of these options be used to further restrict access to the server to authorized users only. The man pages for sshd_config explains how to specify user or group names with these options.

Though the above Action is specifically for the server, similar options also exist for the ssh client. See the man pages for ssh_config to learn how to set host defaults for ssh.

For 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/blueprints/0404/817-6261.pdf).

1.7 Install NTP

The following configuration is for an NTP client that will function as a local server.

Action :

NTP server information:

1. Create the ntp configuration file

Note: Enter the correct ip address for your site.

```
cat << END_SCRIPT > /etc/inet/ntp.conf
# subnet
#The netmask used in this example is for Class C networks
restrict x.x.x.x 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. The newly added keys will be appended to the end of the current ntp.keys file. If a ntp.keys file does not exist, the file will be created in the following steps.

```
cat <<END_SCRIPT >> /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:

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

```
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.

In some situations, such as a router in Defense Message System (DMS) architecture, it is appropriate to utilize at least two NTP servers. Adjust the action as necessary if more than one NTP server is appropriate.

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

2 Minimize `inetd` Network Services

2.1 Disable standard services

Action:
```
cd /etc/inet
for svc in time echo discard daytime chargen fs dtspc \
    exec comsat talk finger uucp name xaudio \
    netstat ufsd rexd systat sun-dr uuidgen krb5_prop;
```

```
do
   awk "(\$1 == \"$svc\") { \$1 = \"#\" \$1 }; {print}" \
   inetd.conf > inetd.conf.new
   mv inetd.conf.new inetd.conf
done
for svc in 100068 100146 100147 100150 100221 \
   100232 100235 kerbd rstatd rusersd sprayd walld; do
   awk "/^$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 == \"$svc\") { \$1 = \"#\" \$1 }; {print}" \
    inetd.conf > inetd.conf.new
    mv inetd.conf.new inetd.conf
done
for svc in 100083 100229 100230 100242 \
    100234 100134  100155 rquotad 100153; do
    awk "/^$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
pkill -HUP -u 0 -P 1 -x inetd
```

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 Item 1.6) 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.

Most of the remaining actions in this chapter 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. Services colored in red are re-enabled in this in this chapter as needed.

Note: Items 2.2 through 2.7, 2.9 and 2.11 have been moved to Appendix C. These Items 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 CDE on this system?

If the answer to this question is yes, proceed with the Action below.
Note: Workstations must have CDE-related daemons enabled.

Action:

```
cd /etc/inet
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).

Since this service uses Sun's standard RPC mechanism, it is important that the system's RPC portmapper (`rpcbind`) also be enabled when this service is turned on. For more information see Item 3.11, "Only enable other RPC-based services if absolutely necessary."

2.10 Only enable removable media daemon if absolutely necessary

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

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

Action:

```
cd /etc/inet
sed 's/^#100155/100155/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:
This item re-enables the `rpc.smserverd` process that works with the volume manager (see Item 3.16 below) and the CDE file manager application to automatically mount CD-ROMs and floppies when the user inserts the new media into the system's drives (the mount command is normally a privileged command that can only be performed by the superuser). Be aware that allowing users to mount and access data from removable media makes it easier for malicious programs and data to be imported onto your network.

Since this service uses Sun's standard RPC mechanism, it is important that the system's RPC portmapper (`rpcbind`) also be enabled when this service is turned on. For more information see Item 3.11, "Only enable other RPC-based services if absolutely necessary."

2.12 Only enable GSS daemon if absolutely necessary

Question:
Are there any security-related services in use at this site that make use of the GSS API?

Note: In Solaris 9, the GSS daemon is generally needed only when Kerberos is being used to secure NFS.

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

Action:
```
cd /etc/inet
sed 's/^#100234/100234/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:
The GSS API is a security abstraction layer that is designed to make it easier for developers to integrate with different authentication schemes. It is most commonly used in applications for sites that use Kerberos for network authentication, though it can also allow applications to interoperate with other authentication schemes.

Since this service uses Sun's standard RPC mechanism, it is important that the system's RPC portmapper (rpcbind) also be enabled when this service is turned on. For more information see Item 3.11, "Only enable other RPC-based services if absolutely necessary."

2.13 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 Action below.

Note: If `ipfilters` will be used (see Item 4.7), then skip this step. Changes to `/etc/init.d/inetsvc` are likely to be overwritten in a future patch or update. They may need to be reapplied after patching.

Action:

```
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:
By disabling multicasting, router discovery can not be performed. In addition, the following action will disable routing functionality.

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

2.14 Disable IPv6

Question:
Is IPv6 in use at this site?

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

Action:
1. Remove all IPv6 hostname information

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

2. Comment out 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:
If the system is configured to handle IPv6 and it is not being used, IPv6 related services and interfaces should be disabled. Some services, such as time, echo, discard, daytime, and chargen, require tcp6 or udp6 in order to function properly. Commenting these protocols out of the `inetd` file will also eliminate those services' IPv4 functionality.

2.15 Enable encrypted remote administration if necessary

Action:
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/sshd stop
cd /etc/ssh
awk '/X11Forwarding/ { $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/sshd start
```

Discussion:
Remote administration must be done over an encrypted channel to protect against information or control leakage. SSH 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:

```
pmadm -d -p zsmon -s ttya
pmadm -d -p zsmon -s ttyb
```

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.

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:

```
cd /etc/default
awk '/^CMASK=/    { $1 = "CMASK=022" }\
            { print }' init > init.new
mv init.new init
chown root:sys init
chmod 444 init
```

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.

Note: Although this is already the default configuration for Solaris 9, this action serves to reinforce the default or to change the setting back to default in case it has been altered.

3.3 Disable inetd if possible

Note: If inetd is disabled, it may get re-enabled in a future patch. This setting should be checked, and reapplied if necessary after applying patches or updates.

Action:

```
cd /etc/init.d
LINE=`awk '/\/usr\/sbin\/inetd/ && \
      !/\[/ { print }' inetsvc`
if [ -n "$LINE" ]; then
      grep -v /usr/sbin/inetd inetsvc > inetsvc.new
      cat <<'EONewInetd' >> inetsvc.new
      lines=`grep -v '^#' /etc/inet/inetd.conf 2>/dev/null | \
            wc -l | sed 's/ //g'`
      EONewInetd
      echo '[ "$lines" != '0' ] && \c' >> inetsvc.new
      echo $LINE >> inetsvc.new
      mv inetsvc.new inetsvc
fi
chown root:sys inetsvc
chmod 744 inetsvc
```

Discussion:
If the actions in Chapter 2 result in all the `inetd`-based service being disabled, then there is no point in running `inetd` at boot time. The code added to the `inetsvc` boot script will result in `inetd` automatically being restarted at boot time if services are ever enabled in `inetd.conf`. However, it may be necessary to manually start `inetd` if the administrator wishes to enable some of these services without rebooting the system.

3.4 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 Action below.

Action:

```
cd /etc/default
cat <<END_DEFAULT > sendmail
MODE=
QUEUEINTERVAL="15m"
END_DEFAULT
chown root:sys sendmail
chmod 644 sendmail
```

Discussion:

It is possible to run a UNIX system with the Sendmail daemon disabled and still allow users on that system to send email out from that machine. Running Sendmail in "daemon mode" (with the `-bd` command-line option) is only required on machines that act as mail servers, receiving and processing email from other hosts on the network.

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.

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`.

3.5 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:

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

Discussion:

If the`/tftpboot` directory exists (see Appendix C Item 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.

3.6 Disable other standard boot services

Action:
Note: Since the actual number for each start up script may vary (i.e., S74autofs vs. S70autofs), wildcards have been used to match the proper script regardless of number.

```
cd /etc/rc2.d
for file in S*autoinstall S*power S*bdconfig \
     S*cachefs.daemon S*cacheos.finish S*llc2 S*pppd \
     S*asppp S*uucp S*slpd S*flashprom S*PRESERVE \
     S*wbem S*ncalogd S*ncad S*ab2mgr; do
     [ -s $file ] && mv $file .NO$file
done
cd /etc/rc3.d
for file in S*dmi S*mipagent; do
     [ -s $file ] && mv $file .NO$file
done
cd /etc/rc2.d
for file in S*nfs.client S*autofs S*rpc \
     S*directory S*ldap.client S*lp S*spc \
     S*afbinit S*ifbinit S*dtlogin S*ncakmod; do
     [ -s $file ] && mv $file .NO$file
done
cd /etc/rc3.d
for file in S*samba S*nfs.server S*kdc.master S*kdc \
     S*apache S*snmpdx S*volmgt S*dhcp; 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). Also, 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.

Many of the actions in Chapter 3 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 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.

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 Solaris 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/rc3.d/S81volmgt	- Starts the `vold` daemon - Needed 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`

3.7 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 Action below.

Action:
```
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.

3.8 Only enable NFS server processes if absolutely necessary

Question:
Is this machine an NFS file server?

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

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

Discussion:
NFS is frequently exploited to gain unauthorized access to files and systems. 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. If the machine will be an NFS client then the `rpcbind` process must be running (see Item 3.11, "Only enable other RPC-based services if absolutely necessary" below).

3.9 Only enable NFS client processes if absolutely necessary

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

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

Action:

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

Discussion:
While this action disables the standard NFS client processes (`statd` and `lockd`), it is still possible for the superuser to mount remote file systems on the local machine via NFS. Starting with Solaris 9, the administrator can completely disable NFS client access by removing the NFS client software packages (SUNWnfscr, SUNWnfscu, and SUNWnfscx), but these packages will have to be re-installed if NFS is to be re-enabled at a later date.

Other file transfer schemes (such as `rdist` via SSH) can often be preferable to NFS for certain applications, although the use of secure RPC or Kerberos can significantly improve NFS security. If the machine will be an NFS client then the `rpcbind` process must be running (see Item 3.11, "Only enable other RPC-based services if absolutely necessary").

3.10 Only enable `automount` daemon if absolutely necessary

Question:
Are any of the following statments true?

- *The system requires an automount daemon to automatically mount local and/or NFS file systems as needed.*
- *The site uses Sun's SMC graphical administrative interface for system management.*

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

Action:

```
mv /etc/rc2.d/.NOS74autofs /etc/rc2.d/S74autofs
```

Discussion:
The `automount` daemon is normally used to automatically mount NFS file systems from remote file servers when needed. However, the `automount` daemon can also be configured to mount local (loopback) filesystems as well, which may include local user home directories, depending on the system configuration. Sites that have local home directories configured via the `automount` daemon in this fashion will need to ensure that this daemon is running for Sun's SMC graphical administrative interface to function properly.

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*
- *The system requires the Volume Manager* `(vold)`
- *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 Action below.

Action:

```
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, it is best to disable RPC-based tools completely. To clarify 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 Action below.

Action:
```
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 LDAP 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 Action below.

Action :
```
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.

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 Action below.

Action:

```
mv /etc/rc2.d/.NOS71ldap.client /etc/rc2.d/S71ldap.client
```

Discussion:
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 Action below.

Action:

```
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. However, LPRng is not supported by Sun Microsystems.

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 Action below.

Action:

```
mv /etc/rc3.d/.NOS81volmgt /etc/rc3.d/S81volmgt
```

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.

It is also necessary to re-enable the `rpc.smserverd` process for the volume manager to function (see Item 2.10, "Only enable removable media daemon if absolutely necessary".)

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 Action below.

Action :

```
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:
For the Solaris CDE GUI to function properly, it is also necessary to enable the `rpcbind` process (see Item 3.11) and the `rpc.ttdbserverd` process (see Item 2.8) 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.

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 Action below.

Action:

```
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 Action below.

Action:

```
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, 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 Action below.

Action:

```
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 Action below:

Action:
1. Create script to disable Internet domain name server

```
cd /etc/init.d
cat << END_NAMED > named.script
/if \[ -f \/etc\/named.conf/ {
     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
chmod 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. For additional information regarding the installation and configuration of BIND, see:
`http://www.deer-run.com/~hal/dns-sendmail/DNSandSendmail.pdf`

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 Action below:

Action:

```
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 Action below:

Note: This change may be overwritten in a future update or patch and should be reapplied if necessary.

Action:

```
cd /etc/init.d
sed 's/^exit/#exit/' RMTMPFILES > RMTMPFILES.new
mv RMTMPFILES.new RMTMPFILES
chown root:sys RMTMPFILES
chmod 744 RMTMPFILES
rm -f /etc/rc2.d/S05RMTMPFILES
ln -s /etc/init.d/RMTMPFILES /etc/rc2.d/S05RMTMPFILES
```

Discussion:
`/var/tmp` could contain information useful in gaining access to the system. When the steps listed above are taken, all the files in `/var/tmp` are removed at bootup 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 script`.

4 Kernel Tuning

4.1 Restrict core dumps to protected directory

Action:

```
mkdir -p /var/core
chown root:root /var/core
chmod 700 /var/core
coreadm     -g /var/core/core_global_%n_%f_%u_%g_%t_%p \
      -i /var/core/core_per_proc_%n_%f_%u_%g_%t_%p \
      -e log \
      -e global -e global-setid -e process -e proc-setid
```

Discussion:
By default core dump files are world-readable. Yet core dumps, particularly those from set-UID and set-GID processes, may contain sensitive data that should not be viewed by all users on the system. The above action causes all core dumps on the system to be written to a special directory that is only accessible by the superuser. On development workstations, this may make it difficult for developers to obtain core files for debugging without administrative intervention.

Core dumps tend to be large files and the contents of the `/var/core` directory can end up rapidly consuming large amounts of disk space and possibly causing a denial of service attack on the system. It is a good idea to monitor this directory on a regular basis and remove any unneeded core files. If the local site chooses, dumping of core files can be completely disabled with the following command: "`coreadm -d global -d global-setid -d process -d proc-setid`".

4.2 Enable stack protection

Action:

```
if [ ! "`grep noexec_user_stack /etc/system`" ]; then
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
fi
```

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:

```
if [ ! "`grep nfssrv:nfs_portmon /etc/system`" ]; then
cat <<END_CFG >> /etc/system
* Require NFS clients to use privileged ports
set nfssrv:nfs_portmon = 1

END_CFG
fi
```

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:

```
if [ ! -f /etc/init.d/netconfig ]; then
     cat <<END_SCRIPT > /etc/init.d/netconfig
#!/sbin/sh
ndd -set /dev/ip ip_def_ttl= '255'
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_conn_req_max_q 1024
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/ip ip_respond_to_echo_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
ndd -set /dev/tcp tcp_extra_priv_ports_add 6112
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
fi
```

Discussion:
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/security/blueprints/`

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:

```
if [ ! "`grep ip_forwarding /etc/init.d/netconfig`" ]
then
     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
ndd -set /dev/ip ip_respond_to_echo_multicast 0
ndd -set /dev/ip ip6_respond_to_echo_multicast 0
END_SCRIPT
fi
```

Discussion:
For a more complete discussion of these parameters and their effect on the security of the system, see: `http://www.sun.com/security/blueprints/`.

4.6 Use better TCP sequence numbers

Action:

```
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 Set up host based firewalls

Action:

1. Download `libiconv-1.8-sol9-sparc-localgz` and `gcc-3.4.0-sol9-sparc-local.gz` from `http://www.sunfreeware.com`. Place the files 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
gunzip libiconf-1.8-sol9-sparc-local.gz
gunzip gcc-3.4.0-sol9-sparc-local.gz
pkgadd -d libiconv-1.8-sol9-sparc-local all
pkgadd -d gcc-3.4.0-sol9-sparc-local all
```

3. Download `pfil-2.1.2.tar.gz` and `ip_fil4.1.2.tar.gz` (ipfilters depends on pfil) from `http://coombs.anu.edu.au/~avalon/ip-filter.html`. Place the files in the `/opt` directory.

4. Execute the following commands to extract the source:

```
cd /opt
gunzip pfil-2.1.2.tar.gz
gunzip ip_fil4.1.2.tar.gz
tar xvf pfil-2.1.2.tar
tar xvpf ip_fil4.1.2.tar
```

5. Install pfil:

a) Set PATH environment variable

```
      PATH=/usr/local/bin:/usr/ccs/bin:$PATH; export PATH
```

b) Compile the pfil package

```
      cd pfil
      sed 's/S64FLAGS=-xildoff/#S64FLAGS=-xildoff/' Makefile \
            > Makefile.new
      sed 's/#S64FLAGS=-m64/S64FLAGS=-m64/' Makefile.new > Makefile
      CC=gcc make package
```

c) Install the newly-created ipfil package

```
      pkgadd -d /tmp/pfil.pkg all
```

Note: At the time of writing, the version of IP Filter used in this guide was the current version. Later versions may not require the makefile patch in steps 6, a) and 6, b). However, later versions have not been tested for inclusion in this guide.

6. Install `ip_fil4.1.2`

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

   a) Patch the `Makefile`.

   The `Makefile` for Solaris in `ip_fil4.1.2` contains an error and must be patched as follows:

   Make a backup copy of the original `Makefile`

```
cd /opt/ip_fil4.1.2/SunOS5
cp Makefile Makefile.orig
```

   b) Create the patch file (in place of [space] and [tab], insert a single space or tab character, respectively – this is critical for Makefile formatting) and patch the Makefile

```
cat << END_SCRIPT > Makefile.patch
199,200c199,200
<[space]\$(OBJ)/ip_rules.o: \$(TOP)/ip_rules.c \$(TOP)/ip_rules.h
<[space][tab]\$(CC) -I\$(TOP) \$(DFLAGS) -c \$(TOP)/ip_rules.c \
-o \$@
---
>[space]\$(OBJ)/ip_rules.o: \$(OBJ)/ip_rules.c \$(TOP)/ip_rules.h
>[space][tab]\$(CC) -I\$(TOP) \$(DFLAGS) -c \$(OBJ)/ip_rules.c \
-o \$@
306,307c306,314
<[space]\$(OBJ)/ip_rules_u.o: \$(TOP)/ip_rules.c \
\$(TOP)/ip_fil.h \$(TOP)/ip_rules.h
<[space][tab]\$(CC) \$(CCARGS) \$(EXTRA) -c \$(TOP)/ip_rules.c \
-o \$@
---
>[space]\$(OBJ)/ip_rules.c: \$(OBJ)/ipf.exe \
\$(TOP)/tools/ipfcomp.c \$(TOP)/rules/ip_rules
>[space][tab]\$(OBJ)/ipf.exe -cc -nf \$(TOP)/rules/ip_rules
>[space][tab]-/bin/mv -f ip_rules.c \$(OBJ)
>[space]
>[space]\$(TOP)/ip_rules.h: \$(OBJ)/ip_rules.c
>[space][tab]/bin/mv -f ip_rules.h \$(TOP)
>[space]
>[space]\$(OBJ)/ip_rules_u.o: \$(OBJ)/ip_rules.c \
\$(TOP)/ip_fil.h \$(TOP)/ip_rules.h
>[space][tab]\$(CC) \$(CCARGS) \$(EXTRA) -c \
\$(OBJ)/ip_rules.c -o \$@
END_SCRIPT
patch Makefile < Makefile.patch
```

   c) Create the ipfilter binaries

```
cd ..
CC=gcc make solaris
```

   d) Build and install the package

```
cd SunOS5
CC=gcc make package
```

7. Turn on ipfilter

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

8. Set up filter rules

Note: Use appropriate interface in place of hme0. Use `ifconfig -a` to list available network interfaces. Use the appropriate network address in place of the x.x.x

```
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 >< 6064 # 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
```

9. Configure router information

Note: Replace x.x.x.x with the actual IP address for the default router.

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

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

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

11. Create /var/log/ipflog

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

12.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. 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 32- and 64-bit versions of Solaris. Furthermore, modern versions of the firewall software contain support for IPv6 firewall rules.

More information regarding the patch in step 6b can be found at the author's website: `http://blog.graves.com/b2evolution/blogs/blog_a.php?p=590.`

4.8 Set routing policies/configuration

Question:
Is your machine acting as a router or does it need to perform IPv4 router discovery?

If the answer is no, proceed with the Action below to set up static routing.

Action:
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 IPv4 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 chapter cover enabling various different forms of system logging in order to keep track of activities on the system. Tools such as Swatch (`http://swatch.sf.net`) and Logcheck (`http://sourceforge.net/projects/sentrytools/`) can be used to automatically monitor logs for intrusion attempts and other suspicious system behavior. These tools are not officially supported by Sun Microsystems.

In addition to the local log files created by the steps in this chapter, it is also recommended that sites collect copies of their system logs on a secure centralized log server. Not only does centralized logging help sites correlate events that may be occuring on multiple systems, but having a second copy of the system log information may be critical after a system compromise where the attacker has modified local log files on the affected system(s).

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 Item 1.7, at `http://www.ntp.org` and at `http://www.sun.com/security/blueprints`.

5.1 Turn on `inetd` tracing

Action:

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

Discussion:

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`. 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` to some other log file destination (see Item 5.3).

In addition to the information provided by inetd tracing, the popular free PortSentry tool (http://sourceforge.net/projects/sentrytools/) can be used to monitor access attempts on unused ports. Running PortSentry may result in some security testing tools reporting "false positives" for "active" ports that are actually being held by the PortSentry daemon. PortSentry is not officially supported by Sun Microsystems.

5.2 Turn on additional logging for FTP daemons

Action:

```
cd /etc/inet
awk '/in.ftpd/ && !/-d/ { $NF = $NF " -d" }
     /in.ftpd/ && !/-l/ { $NF = $NF " -l" }
     { print }' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
chown root:sys inetd.conf
chmod 444 inetd.conf
```

Discussion:
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. Enabling debugging on the FTP daemon can cause user passwords to appear in clear text form in the system logs, if the user accidentally types in their password at the username prompt.

Information about FTP sessions will be logged via syslog, but the system must be configured to capture these messages. For further information, see Item 5.3, "Capture FTP and inetd Connection Tracing Info" below.

5.3 Capture FTP and inetd connection tracing info

Action:

```
if [ ! "`grep -v '^#' /etc/syslog.conf | \
     grep /var/log/connlog`" ]; then
     echo "daemon.debug\t\t\t\t\t/var/log/connlog" \
     >> /etc/syslog.conf
fi
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, Item 5.2 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 Item 5.1. 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 in Item 5.11.

Note: Syslog message format is subject to change in Solaris patches and updates.

5.4 Capture messages sent to `syslog` `Auth` facility

Action:
1) Edit `/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) Add the following new information to `/etc/syslog.conf`

```
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
```

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) 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`.

5.5 Create /var/adm/loginlog

Action:

```
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). 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 in batches of five.

The `loginlog` file should be reviewed and archived on a regular basis. The `logadm` utility can be used to archive all log files. A sample script for archiving log files is provided in Item 5.11.

5.6 Turn on `cron` logging

Action:

```
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.

Note: Although this is already the default configuration for Solaris 9, this action serves to reinforce the default or to change the setting back to default in case it has been altered.

5.7 Enable system accounting

Action:

```
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:
The system accounting script above 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.

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.

5.8 Enable kernel-level auditing

Action:

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 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 Item 5.11 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/S81volmgt` to `/etc/rc3.d/S81volmgt`. If the `minfree` value is reached, the system will begin logging the auditing information in the secondary directory if one is listed.

Note: The BSM should not be enabled more than once.

5.9 Configure role based access control

The security policy of your organization will determine if all of the following role accounts are needed.

Note: The `roleadd` command will populate the `/etc/passwd, /etc/shadow`, and `/etc/user_attr` files. The `roleadd` command restricts the length of the role name to eight characters. The Primary Administrator role can also be created using the Solaris Management Console.

Action:
1. Set up the audit account role
   The Audit role allows assigned users access to monitor the audit logs. To prevent unauthorized users from gaining access to audit information, only those users who require all of the privileges associated with this role should be assigned this role.
   a) Add `audit` account to `/etc/passwd` file
   Note: The following entry should be placed directly after the root entry

```
audit::0:1:Audit_User:/:/sbin/sh
```

   b) Add `audit` account information to `/etc/shadow`

```
pwconv
```

   c) Set password for `audit` account

```
passwd audit
```

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

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

e) Make the `audit` account a role

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

f) Assign users to the `audit` role

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

```
usermod -R audit  username
```

2. Set up the Primary Administrator role

The Primary Administrator role allows assigned users access to perform administrative tasks. Only those users who require all of the privileges associated with this role should be assigned to this role.

a) Add "Primary Administrator" role

Note: `userid` is the desired user id number for the newly created `PrimAdm` role. `homeaccountdir` is the directory under which individual home directories are strored, such as `/home/`.

```
roleadd -u userid  -o -g sysadmin -d /homeaccountdir/PrimAdm \
-s /bin/pfsh -m -P "Primary Administrator" PrimAdm
```

b) Set up the Primary Administrator password

Note: The newly created role will remain locked until the passwd command is used to assign a password to the account.

```
passwd PrimAdm
```

c) Assign user to the Primary Administrator role

- to assign an existing user to the role of Primary Administrator, perform the following

Note: `username` is the username of an existing user to be assigned to the `PrimAdm` role.

```
usermod -R PrimAdm username
```

- to create a new user account and assign the user to the Primary Administrator role, perform the following

Note: `userid` is the desired user id for the newly created user. `usergrp` is the desired primary group for the newly created user. `/homeaccountdir/` is the directory under which individual home directories are stored such as `/homes/`. `username` is the desired username for the newly created user.

```
useradd -u userid -o -g usergrp -d /homeaccountdir/username -m \
-s /bin/sh -R PrimAdm username
```

3. Set up the System Administrator role

The System Administrator role allows assigned users access to perform "non-security" administrative task; such as, device management, network management, software installations, etc. Only those users who require all of the privileges associated with this role should be assigned to this role.

a) add "System Administrator" role

```
roleadd -u userid -o -g sysadmin -d /homeaccountdir/SystAdm \
-s /bin/pfsh -m -P "System Administrator" SystAdm
```

b) Set up the System Administrator password

Note: The newly created role will remain locked until the passwd command is used to assign a password to the account.

```
passwd SystAdm
```

c) Assign user to the System Administrator role

- if the user account already exists, add user to role by performing the following

```
usermod -R SystAdm username
```

- to create a new user account and assign the user to the Primary Administrator role, perform the following

```
useradd -u userid -o -g usergrp -d /homeaccountdir/username -m  \
 -s /bin/sh -R SystAdm username
```

4. Set up the Operator role

The Operator role allows assigned users access to perform tape backups, tape restores and printer management. To prevent unauthorized users from introducing exploits on the system, only those users who require all of the privileges associated with this role should be assigned to this role.

a) add "Operator" role

```
roleadd -u userid -o -g sysadmin -d /homeaccountdir/TapeOp \
-s /bin/pfsh -m -P "Operator" TapeOp
```

b) Set up the Operator password

Note: The newly created role will remain locked until the passwd command is used to assign a password to the account.

```
passwd TapeOp
```

c) Assign user to the Operator role

- if the user account already exist, add user to role by performing the following

```
usermod -R TapeOp username
```

- to create a new user account and assign the user to the Operator role, perform the following

```
useradd -u userid -o -g usergrp -d /homeaccountdir/username -m \
-s /bin/sh -R TapeOp username
```

5. Restart the name service cache daemon in order for the new roles to take effect.

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

Discussion:
Role Based Access Control (RBAC) assigns 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.

To eliminate the system administrator from logging onto the system as root, root can be made into a role. By creating this role, users will be required to log on as themselves before switching to the root account. Please see Sun documentation (http://docs.sun.com/app/docs/doc/806-4078/6jd6cjs58?a=view) on making the root user into a role.

5.10 Confirm permissions on system log files

Action:

```
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.

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.
Note: Although this is already the default configuration for Solaris 9, this action serves to reinforce the default or to change the setting back to default in case it has been altered.

5.11 Implement automated log rotation

Action:
Modify the `/etc/logadm.conf` file
Note: The time listed after the -P option indicates the last time the log was rotated.

```
cat << END_SCRIPT >> /etc/logadm.conf
/var/log/kernlog -C 4 -p 1m -p 5d \
  -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/userlog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/maillog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/daemonlog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/authlog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/cronlog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/connlog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
/var/log/loginlog -C 4 -p 1m -p 5d \
 -a 'kill -HUP \`cat /var/run/syslog.pid\`'
END_SCRIPT

chown root:sys /etc/logadm.conf
chmod 644 /etc/logadm.conf
```

Modify root `crontab` entry for `logadm`
Note: The time in the `crontab` file should be set for the time in which your organization would like to have `logadm` started.

```
EDITOR=path_to_favorite_editor crontab -e
change logadm line to read "30 6 11 * * /usr/sbin/logadm"
```

Discussion:
System administrators must be aware that the information collected in the logs is important and could pontentially serve as forensic evidence. They must also remember that disk space is limited and every possible step should be taken to preserve it.

The `logadm` and `crontab` commands can be used by system administrators to setup automated log rotation. The `logadm` can be used at the command line to update a log that has become too large before the log is scheduled to be rotated. The `-V` option should be used to validate that the entries in the file are correct when using the `logadm` command to manually edit the `/etc/logadm.conf` configuration file.

6 File/Directory Permissions/Access

6.1 Add 'logging' option to root file system

Action:

```
awk '($4 == "ufs" && $3 == "/" && $7 == "-") \
     { $7 = "logging" }; \
     ($4 == "ufs" && $3 == "/" && $7 !~ /logging/) \
     { $7 = $7",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. However, other sorts of attacks are possible if the attacker has unrestricted physical access to the system. Be sure to keep critical systems in limited access data centers or other restricted facilities.

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:

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

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.

Note: Although this is already the default configuration for Solaris 9, this action serves to reinforce the default or to change the setting back to default in case it has been altered.

6.3 Configure vold.conf to allow users access to CD-ROM only

Action:

```
awk '($2 == "floppy" || $2 == "dev/diskette[0-9]/*" \
      || $4 == "floppy" || $2 == "rmdisk" ) {$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 disks, to insert malicious code on the system. By preventing regular users from having access to the floppy drive and other removable devices, 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.

In Solaris 9, the Volume Manager now allows users access to removable devices, such as DVD-ROMs, jaz and zip drives. The `rmformat` command should be used to format, label, and set read/write protection for the removable devices.

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

6.4 Verify passwd, shadow, and group file permissions

Action:

```
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.

Note: Although this is already the default configuration for Solaris 9, this action serves to reinforce the default or to change the setting back to default in case it has been altered.

6.5 Verify world-writable directories have their sticky bit set

Action:
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.

6.6 Find unauthorized world-writable files

Action:
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.

6.7 Find unauthorized SUID/SGID system executables

Action:
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`. Cryptographic checksums of these files (along with all standard files in the Solaris operating system) can be obtained from the Solaris Fingerprint Database (see `http://sunsolve.sun.com/pub-cgi/fileFingerprints.pl`). Tools for interacting with the Fingerprint Database are available from `http://www.sun.com/blueprints/tools/`.

6.8 Find unowned files and directories

Action:
Administrators who wish to obtain a list of files and directories currently installed on the system where the user or group owner of the file is not listed in the `/etc/passwd` or `/etc/group` files may run the following command:

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

Discussion:
Sometimes when administrators delete users from the system, they neglect to remove all the files owned by those users from the system. A new user who is assigned the deleted user's user ID or group ID may then end up "owning" these files, and thus have more access on the system than was intended. It is a good idea to locate files that are owned by users or groups not listed in the system configuration files, and make sure to reset the ownership of these files to some active user on the system as appropriate.

6.9 Run `fix-modes`

Action:
1. Download the pre-compiled `fix-modes` software from
`http://www.sun.com/software/security/downloads.html`

2. Unpack and install the software

```
uncompress SUNBEfixm.pkg.Z
pkgadd -d SUNBEfixm.pkg all
```

3. Run the fix-modes program.

```
/opt/SUNBEfixm/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`.

The actions above recommend using a pre-compiled version of `fix-modes` supplied by Sun for use with their Solaris Security Toolkit framework. The source code is also available from the same URL. Sun's version of the tool has been specifically modified to avoid well-known problems when running `fix-modes` on SSP systems for the E10K and E15K products.

Exploiting programs that have their SUID and/or SGID bits set is one of the most common methods employed by an attacker for privilege escalation. In most cases these programs are rarely needed by the average user (e.g. video card configuration). 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 Set higher security level for sadmind service

Action:

```
cd /etc/inet
awk '/sadmind /&& !/-S/       { $7 = $7 " -S 2" }
     { print }' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
chown root:sys inetd.conf
chmod 444 inetd.conf
```

Discussion:
The `sadmind` service is the primary daemon that enables the Solaris remote administration framework for distributed system administration tasks. Since the operations allowed by this daemon are extremely powerful, it is best to use the highest security setting available for authorizing client connections. Given the history of significant security issues with `sadmind`, the items of Chapter 2 of this document actually disable the `sadmind` service, so this setting will only take effect if the service is re-enabled in `inetd.conf`.

7.2 Disable "nobody" access for secure RPC

Action:

```
cd /etc/default
awk '/ENABLE_NOBODY_KEYS=/ \
     { $1 = "ENABLE_NOBODY_KEYS=NO" }
     { print }' keyserv > keyserv.new
mv keyserv.new keyserv
chown root:sys keyserv
chmod 444 keyserv
```

Discussion:
The keyserv process stores user keys that are utilized with Sun's secure RPC mechanism. The above action prevents keyserv from using default keys for the "nobody" user, effectively stopping this user from accessing information via secure RPC.

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

Action:

```
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, 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).

7.4 Create `/etc/ftpd/ftpusers`

Action:

```
if [ -d /etc/ftpd ]; then
      file=/etc/ftpd/ftpusers
      for user in root daemon bin sys adm lp uucp nuucp \
            smmsp listen nobody noaccess nobody4
      do
            echo $user >> $file
      done
      sort -u $file > $file.new
      mv $file.new $file
      chown root:root $file
      chmod 600 $file
else
      echo "ftpusers file does not exist"
fi
```

Discussion:
`ftpusers` contains a list of users who *are not* allowed to access the system via FTP. For Solaris 9 this file is `/etc/ftpd/ftpusers`. 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.

7.5 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 Action below.

Action:

```
cd /etc/default
if [ "`grep LOG_FROM_REMOTE= syslogd`" ]; then
     awk '/LOG_FROM_REMOTE=/ \
      { $1 = "LOG_FROM_REMOTE=NO"}
      { print }' syslogd > syslogd.new
     mv syslogd.new syslogd
else
     echo LOG_FROM_REMOTE=NO >> syslogd
fi
chown root:sys syslogd
chmod 444 syslogd
```

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.

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.

7.6 Prevent remote XDMCP access

Action:

```
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. The default Solaris 9 configuration allows any system on the network to get a remote login screen from the local system. This default behavior can be overridden in the `/etc/dt/config/Xaccess` file.

7.7 Prevent X server from listening on port 6000/tcp

Action:

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

Disabling listening on port 6000 for Xservers does have the side effect that it also prevents authorized remote X clients from displaying windows on the local system. However, the forwarding of X events via ssh will still work properly. This is the preferred, more secure method of transmitting data from remote X clients.

7.8 Set default locking screensaver timeout

Action:

```
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
     chown root:sys $dir/sys.resources
     chmod 444 $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.

7.9 Restrict `at/cron` to authorized users

Action:

```
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.

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 Item 5.7, "Enable system Accounting"). `cron.allow` only controls administrative access to the `crontab` command for scheduling and modifying `cron` jobs. Much more effective access controls for `cron` system can be obtained by using Role-Based Access Controls (RBAC).

7.10 Remove empty `crontab` files and restrict file permissions

Action:

```
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.

7.11 Prevent `root` logins to system console

Action:

```
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`.

Anonymous root logins should never be allowed, except on the system console in emergency situations (this is the default configuration for Solaris). At all other times 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 problems.

7.12 Limit number of failed login attempts

Action:

```
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 a 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

Hardware Compatibility:

This action only applies to SPARC-based systems (not Solaris x86 or Solaris PPC).

Action:

```
eeprom security-#badlogins=0
if [ ! "`crontab -l | grep security-#badlogins`" ]; then
     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
fi
eeprom security-mode=command
```

Note: If not prompted for a password, then an EEPROM password has previously been set. To reset the EEPROM password, use the following command

```
eeprom security-password=
```

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 measure helps prevent an attacker with physical access to the system console from subverting the security of the system by requiring authentication when booting off an external device (such as a CD-ROM or floppy disk).

The administrator should write down this password and place the password in a sealed envelope in a secure location (locked desk drawers are typically not secure). If the password is lost or forgotten, simply run the command "eeprom security-password=" as root to reset the forgotten password.

Note: EEPROM security features are available only on Sun SPARC hardware, and not Intel x86-compatible hardware.

8 User Accounts and Environment

The items in this chapter are tasks that the local administrator should undertake on a regular 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 available for free download from http://www.cisecurity.org.

8.1 Block system accounts

Note: The following script will make changes to the /etc/shadow file. It is imperative that this file be backed up first.

Action:

```
passwd -l daemon
for user in bin adm lp uucp nuucp smmsp \
     listen nobody noaccess  nobody4; do
     /usr/sbin/passmgmt -m -s /dev/null $user
done
for user listen nobody noaccess nobody4; do
     passwd -l $user
done
awk -F: '/^(daemon|sys|bin|adm|lp|uucp|nuucp|smmsp):/ \
     { $2="NP" } { print }' /etc/shadow |sed 's/ /:/g' > \
/etc/shadow.new
mv /etc/shadow.new /etc/shadow
chmod 400 /etc/shadow
chown root:sys
```

Discussion:

Accounts that are not being used by regular users should not allow interactive logins. As of Solaris 9, there is a stricter distinction between a locked account and a non-login account. While neither of these types of accounts allow interactive logins, a non-login account can be used to perform tasks such as run a cron job, that a locked account cannot. A non-login account has a password of `NP` and a locked account has a password of `*LK*`. Since there is no interface in Solaris 9 to set a non-login password, the `/etc/shadow` file must be edited directly. Not only should the `password` field for the account be set to an invalid string, but also the shell field in the `/etc/passwd` 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:
Create noshell script

```
cat <<END_SCRIPT > /sbin/noshell
#!/bin/sh
#
# Copyright (c) 2000-2002 by Sun Microsystems, Inc.
# All rights reserved.
#
#ident  "@(#)noshell 1.3     02/12/16     SMI"
#

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

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

HNAME="\`uname -n\`"
UNAME="\`id | awk '{ print $1 }'\`"

logger -i -p auth.crit "Unauthorized access attempt on \
     \${HNAME} by \${UNAME}"

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

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 an 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 Solaris Security Tookit script for noshell.

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

8.3 Verify that there are no accounts with empty password fields

Action:
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. All accounts should have strong passwords or should be locked by a password string of "NP" or "*LK*".

8.4 Set account expiration parameters on active accounts

Action:

```
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. 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.

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

Action:
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.

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

Action:
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://www.sun.com/software/whitepapers/wp-rbac/wp-rbac.pdf`.

8.7 Set default group for root account

Action:

```
passmgmt -m -g 0 root
```

Discussion:
The default group for the root account under Solaris is the "other" group, which may be shared by many other acounts on the system. Changing the default group for the root account helps prevent root-owned files from accidentally becoming acessible to non-privileged users.

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

Action:

```
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.

8.9 Set user home directories to mode 750 or more restrictive

Action:

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

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

8.10 Disallow group/world-writable user dot-files

Action:

```
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.

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

Action:
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 go-rwx \$userhome/.forward
          chmod u-x \$userhome/.forward
          mailx -s .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.

8.12 Remove user `.netrc` files

Action:

```
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.

8.13 Set default `UMASK` for users

Action:

```
cd /etc/default
if [ "`grep UMASK= login`" ]; then
      awk '/UMASK=/ { $1 = "UMASK=077" }
                  { print }' login > login.new
      mv login.new login
else
      echo UMASK=077 >> login
fi

cd /etc
for file in profile .login
do
if [ "`grep umask $file`" ]; then
      awk '$1 == "umask" { $2 = "077" }
            { print }' $file > $file.new
      mv $file.new $file
else
      echo umask 077 >> $file

fi
done
chown root:sys /etc/default/login /etc/profile /etc/.login
chmod 444 /etc/default/login /etc/profile /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.

8.14 Set default UMASK for FTP users

Action:

```
cd /etc/ftpd
if [ "`grep '^defumask' ftpaccess`" ]; then
     awk '/^defumask/   { $2 = "0777" }
                  { print }' ftpaccess > ftpaccess.new
     mv ftpaccess.new ftpaccess
else
     echo defumask 077 >> ftpaccess
fi
chown root:sys ftpaccess
chmod 444 ftpaccess
```

Discussion:
The Solaris 9 FTP daemon is derived from the Washinton University FTP daemon, so the default UMASK value is set in /etc/ftpd/ftpaccess. Please refer to the discussion in Item 8.13 for more information on umask values.

8.15 Set "mesg n" as default for all users

Action:

```
cd /etc
for file in profile .login
do
      if [ "`grep mesg $file`" ]; then
            awk '$1 == "mesg" { $2 = "n" }
                  { print }' $file > $file.new
            mv $file.new $file

      else
            echo mesg n >> $file
      fi
      chown root:sys $file
      chmod 444 $file
done
```

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.

8.16 Change root's home directory

Action:

```
mkdir /root
mv -i /.?* /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`.

8.17 Set up user file quotas

Action:
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 '($4 == "ufs" && $3 == "mount_point" && $7 == "-") \
     { $7 = "rq" }; \
     ($4 == "ufs" && $3 == "mount_point" && $7 !~ /rq/) \
     { $7 = $7",rq"}; \
     { print }' /etc/vfstab > /etc/vfstab.new

mv vfstab.new vfstab
chown root:sys vfstab
chmod 664 vfstab
EDITOR=/usr/bin/vi
export EDITOR
edquota -t mount_point
#The vi editor will be spawned with the line shown below.  Modify the
#corresponding higlighted fields in the editor to meet the block
#and file time limits chosen.
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
#The vi editor will be spawned with the line shown below.  Modify the
#corresponding higlighted fields in the editor to meet the block
#and inode limits chosen.
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
#Activate the quotas previously generated using the following command:
quotaon -v mount_point
```

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 (eg. /var/spool/mail), user home directories (eg. /export/home), and temporary files (eg. /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.

9 Warning Banners

Presenting some sort of statutory warning message prior to the normal user logon 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 the 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 any system modifications are made, as these warning messages are inherently site specific. More information (including citations of relevant case law) can be found at:
http://www.usdoj.gov/criminal/cybercrime/s&sappendix2002.htm.

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.

9.1 Create warnings for physical access services

Action:

```
eeprom oem-banner="Authorized uses only.  All activity \
may be monitored and reported."
eeprom oem-banner\?=true
echo "Authorized uses only.  All  activity may be \
monitored and reported." > /etc/motd
echo "Authorized uses only.  All  activity may be \
monitored and reported." > /etc/issue
chown root:sys /etc/motd
chown root:root /etc/issue
chmod 644 /etc/motd /etc/issue
```

Discussion:

The contents of the `/etc/issue` file are displayed prior to the login prompt on the system's console and serial devices. `/etc/motd` is generally displayed after all successful logins, no matter where the user is logging in from, but is thought to be less useful because it only provides notification to the user after the machine has been accessed.

The OEM banner will be displayed only when the system is powered on. Setting this banner has the side effect of hiding the standard Sun power-on banner, which normally displays the system host ID, MAC address, etc.

9.2 Create warnings for GUI-based logins

Action:

```
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: Authorized uses \
     only.  All activity may be monitored and reported." \
     >> $dir/Xresources
     echo "Dtlogin*greeting.persLabelString: Authorized \
     uses only.  All activity may be monitored and reported." \
     >> $dir/Xresources
done
chown root:sys /etc/dt/config/*/Xresources
chmod 644 /etc/dt/config/*/Xresources
```

Discussion:
The standard graphical login program for Solaris requires the the username to be entered in one dialog box and the corresponding password to be entered in a second, separate dialog. The commands above set the warning message on both to be the same message, but the site has the option of using different messages on each screen. The `Dtlogin*greeting.labelString` is the message for the first dialog where the user is prompted for their username, and `...perslabelString` is the message on the second dialog box.

9.3 Create warnings for `telnet` daemon

Action:

```
cd /etc/default
if [ ! "`grep \"^BANNER=\" telnetd`" ]; then
echo "BANNER=\"Authorized uses only.  All activity may \
be monitored and reported.\\\n\\\n\"" > telnetd
chown root:sys telnetd
chmod 444 telnetd
fi
```

Discussion:
Setting this banner has the side effect of hiding the default `telnet` banner, which advertises the version of the Solaris running on the system.

9.4 Create warnings for FTP daemons

Action:

```
echo Authorized uses only.  All activity may \
be monitored and reported.  > /etc/ftpd/banner.msg
chown root:root /etc/ftpd/banner.msg
chmod 444 /etc/ftpd/banner.msg
```

Discussion:
The FTP daemon in Solaris 9 is based on the popular Washinton University FTP daemon (WU-FTPD), which is an Open Source program widely distributed on the Internet. The procedure for setting the warning banner on Solaris 9 differs from previous releases.

Appendix A: File Backup Script

```
#!/bin/sh
ext=`date '+%Y%m%d-%H:%M:%S'`
for file in /etc/.login             /etc/coreadm.conf       \
            /etc/cron.d/at.allow    /etc/cron.d/at.deny     \
            /etc/cron.d/cron.allow  /etc/cron.d/cron.deny   \
            /etc/default/cron       /etc/default/power      \
            /etc/default/inetd      /etc/defualt/inetinit   \
            /etc/default/init       /etc/default/keyserv    \
            /etc/default/login      /etc/default/passwd     \
            /etc/default/sendmail   /etc/default/syslogd    \
            /etc/default/telnetd /etc/default-sys-suspend   \
            /etc/dt/config/Xaccess                          \
            /etc/dt/config/*/Xresources                     \
            /etc/dt/config/*/sys.resources                  \
            /etc/dt/config/Xservers                         \
            /etc/ftpd/banner.msg    /etc/ftpd/ftpaccess     \
            /etc/ftpusers           /etc/defaultrouter      \
            /etc/hosts.allow        /etc/hosts.deny         \
            /etc/inet/inetd.conf    /etc/inet/ntp.conf      \
            /etc/inet/ntp.keys      /etc/init.d/RMTMPFILES  \
            /etc/init.d/netconfig   /etc/init.d/inetsvc     \
            /etc/init.d/perf        /etc/dfs/dfstab         \
            /etc/issue        /etc/motd   /etc/pam.conf     \
            /etc/passwd       /etc/profile      /etc/shadow \
            /etc/rmmount.conf /etc/security/audit_class     \
            /etc/security/audit_control                     \
            /etc/security/audit_event                       \
            /etc/security/audit_startup                     \
            /etc/security/audit_user      /etc/usr_attr     \
            /etc/logadm.conf  /etc/mail/sendmail.cf         \
            /etc/ssh/sshd_config    /etc/syslog.conf        \
            /etc/system /etc/vfstab /etc/vold.conf
do
      [ -f $file ] && cp -p $file $file-preNSA-$ext
done

mkdir -p -m 0700 /var/spool/cron/crontabs-preNSA-$ext
cd /var/spool/cron/crontabs
tar cf - * | (cd ../crontabs-preNSA-$ext; tar xfp -)
```

Appendix B: Additional Security Notes

The items in this appendix are security configuration settings that have been suggested by several other resources and system hardening tools. However, given the other settings in the benchmark document, the settings presented here provide relatively little incremental security benefit. Nevertheless, none of these settings should have a significant impact on the functionality of the system, and some sites may feel that the slight security enhancement of these settings outweighs the (sometimes minimal) administrative cost of performing them.

None of these settings will be checked by the automated scoring tool provided with the benchmark document. They are purely optional and may be applied or not at the discretion of local site administrators.

SN.1 Enable process accounting at boot time

Action:

```
ln -s /etc/init.d/acct /etc/rc3.d/S99acct
```

Discussion:
Process accounting logs information about every process that runs to completion on the system, including the amount of CPU time, memory, etc. consumed by each process. While this would seem like useful information in the wake of a potential security incident on the system, kernel-level auditing with the "`+argv,arge`" policy (as enabled in Item 5.8) provides more information about each process execution in general (although kernel-level auditing does not capture system resource usage information). Both process accounting and kernel-level auditing can be a significant performance drain on the system, so enabling both seems excessive given the large amount of overlap in the information each provides.

SN.2 Use full path names in /etc/dfs/dfstab file

Action:

```
cd /etc/dfs
awk '($1 == "share") { $1 = "/usr/sbin/share" }; \
     { print }' dfstab > dfstab.new
mv dfstab.new dfstab
chown root:sys dfstab
chmod 644 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. Use the absolute pathname to the `share` command to protect against an exploits stemming from an attack on the administrator's `PATH` environment, etc. However, if an attacker is able to corrupt root's path to this extent, other attacks seem more likely and more damaging to the integrity of the system.

SN.3 Restrict access to power management functions

Action:

```
cd /etc/default
awk '/^PMCHANGEPERM=/   { $1 = "PMCHANGEPERM=-" }
     /^CPRCHANGEPERM=/  { $1 = "CPRCHANGEPERM=-" }
                        { print }' power > power.new
mv power.new power
chown root:sys power
chmod 444 power
```

Discussion:
The settings in `/etc/default/power` control which users have access to the configuration settings for the system power management and checkpoint/resume features. By setting both values to "`-`", configuration changes are restricted to only the superuser. Given that the benchmark document disables the power management daemon by default, the effect of these settings is negligible, but sites may wish to make this configuration change as a "defense in depth" measure.

SN.4 Restrict access to sys-suspend feature

Action:

```
cd /etc/default
awk '/^PERMS=/    { $1 = "PERMS=-" }
                  { print }' sys-suspend > sys-suspend.new
mv sys-suspend.new sys-suspend
chown root:sys sys-suspend
chmod 444 sys-suspend
```

Discussion:
The /etc/default/sys-suspend settings control which users are allowed to use the sys-suspend command to shut down the system. Setting "PERMS=-" means that only the superuser is granted this privilege. A user who is truly motivated to take the system off-line and has physical access to the system can simply remove power from the machine. Granting sys-suspend access may be a more graceful way of allowing normal users to shut down their own machines.

SN.5 Create symlinks for dangerous files

Action:

```
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 Item 7.3). Attackers will often target these files as part of their exploit scripts. By linking these files to /dev/null, any data that an attacker writes to these files is simply discarded (though an astute attacker can still remove the link prior to writing their malicious data). However, the benchmark already disables .rhosts-style authentication in several ways, so the additional security provided by creating these symlinks is minimal.

SN.6 Change default greeting string for Sendmail

Action:

```
cd /etc/mail
awk '/O SmtpGreetingMessage=/ \
     { print "O SmtpGreetingMessage=mailer ready"; next}
     { print }' sendmail.cf > sendmail.cf.new
mv sendmail.cf.new sendmail.cf
chown root:bin sendmail.cf
chmod 444 sendmail.cf
```

Discussion:

The default SMTP greeting string displays the version of the Sendmail software running on the remote system. Hiding this information is generally considered to be good practice, since it can help attackers target attacks at machines running a vulnerable version of Sendmail. However, the actions in the benchmark document completely disable Sendmail on the system, so changing this default greeting string affords no benefit unless the machine happens to be an email server.

Appendix C: High Risk Items

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:

```
cd /etc/inet
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 Item 1.6) 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 Action below.

Action:

```
cd /etc/inet
sed 's/^#ftp/ftp/' 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 Item 7.4.

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 Action below.

Action:
```
cd /etc/inet
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 is unlikely that there is a case where these tools cannot be replaced with SSH (see Item 1.6).

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 Action below.

Action:

```
cd /etc/inet
sed 's/^#tftp/tftp/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
mkdir -p /tftpboot
chown root:root /tftpboot
chmod 711 /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 Action below.

Action:

```
cd /etc/inet
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. 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 Action below.

Action:

```
cd /etc/inet
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 Action below.

Action:

```
cd /etc/inet
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.

Since this service uses Sun's standard RPC mechanism, it is important that the system's RPC portmapper (`rpcbind`) also be enabled when this service is turned on. For more information see Item 3.11, "Only enable other RPC-based services if absolutely necessary."

2.11 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 Action below.

Action:

```
cd /etc/inet
sed 's/^#100134/100134/' inetd.conf > inetd.conf.new
mv inetd.conf.new inetd.conf
```

Discussion:
Kerberos support has been added to Solaris (see Sun's Kerberos site, http://wwws.sun.com/software/security/kerberos/). However, Kerberos may not be in use at all sites. For more information on Kerberos see http://web.mit.edu/kerberos/www/.

Since this service uses Sun's standard RPC mechanism, it is important that the system's RPC portmapper (rpcbind) also be enabled when this service is turned on. For more information see Item 3.11, "Only enable other RPC-based services if absolutely necessary."

References

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

Sun security home:
`http://www.sun.com/security`

Sun security blueprints:
`http://www.sun.com/security/blueprints`

Sun product documentation:
`http://docs.sun.com/prod/solaris`

Patches and related documentation:
`ftp://sunsolve.sun.com/patchroot/clusters/`

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

Solaris Security Toolkit:
`http://www.sun.com/security/jass/`

Pre-compiled `fix-modes` software:
`http://wwws.sun.com/software/security/downloads.html`

Solaris Fingerprint Database:
`http://sunsolve.sun.com/pub-cgi/fileFingerprints.pl`

Sun's Kerberos Information:
`http://wwws.sun.com/software/security/kerberos`

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

OpenSSH white paper, NTP whitepaper, information on kernel (ndd) settings, et al:
`http://www.sun.com/security/blueprints/`

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

Patched Makefile for IP Filter:
`http://blog.graves.com/b2evolution/blogs/blog_a.php?p=590`

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 and Logcheck(port and log monitoring tools):
`http://sourceforge.net/projects/sentrytools`

Swatch (log monitoring tool):
`http://swatch.sourceforge.net`

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

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