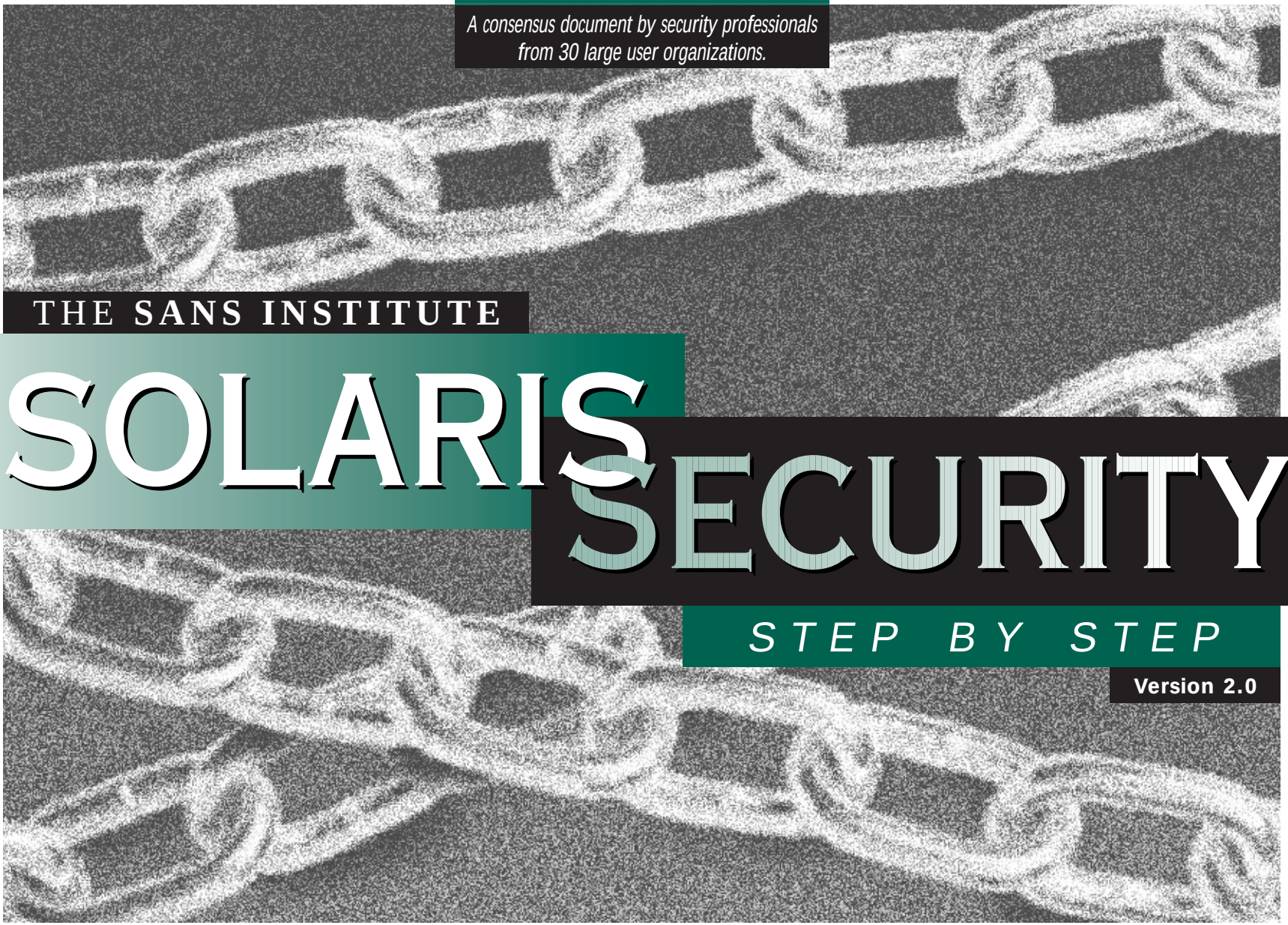
A Survival Guide for Solaris Security

A consensus document by security professionals from 30 large user organizations.



THE SANS INSTITUTE

SOLARIS SECURITY

STEP BY STEP

Version 2.0

SOLARIS STEP BY STEP SECURITY

VERSION 2.0

This document is based entirely on the real-world experiences of the editor and an amazing team of reviewers. The procedures described in this document apply equally well to all Solaris versions from Solaris 2.5.1 through Solaris 8 (and most of the steps are known to work on Solaris releases as early as 2.3). Testing was conducted using Solaris on the Sparc platform, but these procedures should work on x86 and PowerPC-based systems.

The document is designed to be a step-by-step procedure for starting from scratch on an unused Solaris system and turning it into a platform capable of supporting a wide variety of services with high levels of security. Many of the procedures outlined in this document are equally valid, however, for systems already in production. In particular, the majority of Steps 1.4 and onward can be applied to any Solaris system. The document does not describe the security implications of any particular service in great detail. Please consult relevant security material before deploying services on top of this platform.

INTRODUCTION

The SANS Institute enthusiastically applauds the work of these professionals and their willingness to share the lessons they have learned and the techniques they use.

Brian L. Birkinbine
Sean Boran, SecurityPortal
David Brumley, Stanford Computer Security
Donald G. Brunder, Ph.D., The University of Texas Medical Branch at Galveston
Jozo Capkun, Komoko Services Ltd.
Jami M. Carroll, BTG, Inc.
Sweth Chandramouli, ServerVault, Inc.
Pete Clare, Telstra Saturn, New Zealand
Jason M. Frey
Sanjay Gowda, Williams Communications
Daniel Harrison, Loudcloud Inc.
Simon Horgan, Internet Security Systems
Frank Hui, Unix Server Engineering, Pharmacia Corp
Shawn Laemmrich, Michigan Tech University.
Israel Lawson, GMAC Insurance Online, Inc.
Rob Marchand, VoiceGenie Technologies
Chris McDonald, CISSP, U.S. Army Research Laboratory
Juan R Mendez, Veridian Information Solutions
Thomas Nau, Universities Computer Center, Ulm, Germany
Edward Nichols, Qualcomm Inc.
Dan Nienhaus, Nienhaus Consulting
Reg Quinton, University of Waterloo, Waterloo, Ontario, Canada
Steve Remsing, Dow Chemical Company
Bill Royds, Department of Canadian Heritage, Hull, Quebec, Canada
Marc Schaming, EOST Strasbourg
Shane Tully, Qantas Airways Limited, Sydney, Australia
Pekka Viitasalo, CISSP
Dennis Villagomez, Systems Engineer
Cecil Whitaker, NSWCDD
Laurie Zirkle, Virginia Tech

In some cases, third-party software tools are required to complete various steps. The target platform is specifically installed without the compilers and other programming tools required to build this third-party software from source code. It is therefore necessary that there be another reasonably secure machine available which is capable of building the software that is needed. For some steps, this machine will need a copy of the `gzip` *utilities from the Free Software Foundation (see* `http://www.fsf.org/gnulist/production/gzip.html` *for more information, or install package* `SUNWgzip` *under Solaris 8). Obviously some transfer mechanism is also required (see note below regarding the system build environment). If these procedures are going to be repeated often, it may be expedient to create a CD-ROM with all of the required third-party tools and packages.*

IMPORTANT TO NOTE: *Be careful of the environment in which the machine is being built. During the OS install process, the network interfaces for the host are live, yet no work has been done to secure the platform. It is entirely possible that attackers can subvert the system before it can be secured— making any additional security largely useless. If possible, build the systems on a physically isolated ("air-gapped") network or without any network connection at all. Build machines in a locked or otherwise secure area away from where they will be installed for production use (to prevent other staff from "helpfully" connecting the machine to its production network). Have a single person complete this installation process per machine and log or track each installation step. All installed software (both the operating system and any third-party tools) should be installed from read-only media and that media clearly labeled and securely stored for any future audit needs.*

IMPORTANT:
Updates will be issued whenever a change in these steps is required, and new versions will be published periodically. Errata for the current version of the guide can be found at `http://www.sans.org/solaris_errata.htm`.
All comments and suggestions related to the current or future editions of this guide should be directed to `solaris@sans.org`.

This edition was drafted and edited by Hal Pomeranz, Deer Run Associates

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

CONTENTS

STEP 1
Basic OS Installation

STEP 1.1 BOOT-TIME CONFIGURATION

PROBLEM: *During the process of booting from CD-ROM for the initial OS install, the administrator is prompted for local host configuration information. Note that for Solaris 8, the install process varies slightly from the order presented below.*

- Step 1.1.1. Boot from most current Solaris OS CD-ROM
- Step 1.1.2. Enter host name
- Step 1.1.3. Select “Networked” (even if machine is currently disconnected)
- Step 1.1.4. Enter IP address
- Step 1.1.5. Select “None” for name service
- Step 1.1.6. Enter appropriate netmask information
- Step 1.1.7. Select time zone
- Step 1.1.8. Verify that the date/time presented by the system is correct

STEP 1.2 MINIMAL OS INSTALLATION

PROBLEM: *Modern Unix systems contain a huge variety of programs that, while useful, significantly reduce the security of the host platform. Install the smallest operating system image provided by Solaris which meets the business requirements for the system. For Internet-connected platforms such as Web and FTP servers, install only the “Core System Support” image. User desktops may need other packages which contain CDE, programming tools and include files, etc.*

- Step 1.2.1. Choose “Initial” install (not upgrade) to start with a clean system image.
- Step 1.2.2. Configure the machine as a “Standalone” server. The machine should not be dependent upon resources from other machines (which could be compromised or shut down).
- Step 1.2.3. Select “Core System Support”

CAVEAT:
Administrators may wish to reserve a 5MB partition on each disk if there's a possibility that Sun's Online Disk Suite (ODS) product may be installed on this system. Veritas Volume Manager requires two free partitions to encapsulate the root drive.

- Step 1.2.4. Lay out file system on disks. At a minimum, the administrator should create four partitions: `/`, `/usr`, `/var`, and an additional `/local` file system for non-Sun applications and data. Additional disks, file systems, etc. may be added at the discretion of the administrator.

NOTE: The Solaris 8 "Core System Support" cluster (32-bit only) will "fit" in under 110MB but additional space is desired for logging, third-party applications, data, etc. Leave a great deal of room in `/var` for log files (possibly putting them in a separate partition). Systems which require substantial third-party software may need a separate `/opt` or `/usr/local` partition.

- Step 1.2.5. Do not choose to mount any remote file systems
- Step 1.2.6. Select "Reboot" after install and begin installation

STEP 1.3 POST INSTALL/NETWORKING CONFIGURATION

PROBLEM: *Additional steps require getting the Sun Recommended Patch Cluster and other third-party software onto the machine. If the machine is physically disconnected from production networks, some sort of portable media will be required to get this (and other files in later steps) onto the host. If these files are obtained over the network (even though this makes the host vulnerable to attack), perform the following configuration steps to make the machine "play" on the network. Even if the machine is currently physically disconnected and will not be downloading files via the network, go ahead and perform these steps.*

CAVEAT:
Dynamic routing may be required instead of static default routing for some applications. Get `gated` (`http://www.gated.org/`) in this case.

- Step 1.3.1. Set root password as appropriate
- Step 1.3.2. Create an `/etc/defaultrouter` file containing the IP address of the system's default router.
- Step 1.3.3. Create `/etc/notrouter` to disable IP forwarding and prevent `in.routed` and `in.rdiscd` from starting at boot time

```
touch /etc/notrouter
```

- Step 1.3.4. Create `/etc/resolv.conf` with appropriate local information.

- Step 1.3.5. Modify `/etc/nsswitch.conf` and change the appropriate line to read

```
hosts: files dns
```

- Step 1.3.6. `reboot`

CAVEAT:
Administrators may wish to keep a small list of the hosts this machine trusts or must communicate with in the machine's `/etc/inet/hosts` file. This helps protect against DNS spoofing at the cost of maintaining multiple copies of the same information. Generally, entries in the hosts file should contain both the fully qualified host name (listed first) and the unqualified `host` name of the machine.

STEP 1.4 ADDING ADDITIONAL PACKAGES

PROBLEM: *Some useful tools for easing administration and enhancing the security of the system are not installed as part of the "Core System Support" cluster. The administrator must install these packages manually from the OS media. Note that Solaris 8 is now shipped on two CD-ROMs: packages are split between the two disks.*

- Step 1.4.1. Verify that the OS media is still in the drive, or re-insert the first OS media CD-ROM.

CAVEAT:
Depending upon the hardware architecture of the system, the disk device for the OS media may be different from that shown above. Use the appropriate disk device for the machine.

- Step 1.4.2. Mount the OS media on `/mnt`

```
mount -r -F hsfs /dev/dsk/c0t2d0s0 /mnt
```

- Step 1.4.3. `cd /mnt/Solaris_*/Product`
- Step 1.4.4. Add the terminfo database and system accounting related packages

```
pkgadd -d . SUNWter SUNWaccr SUNWaccu
```

NOTE: Administrators may wish to add the `SUNWntpr` and `SUNWntpu` packages to load the NTP server software for time synchronization (these packages are available for Solaris 2.6 and later). `SUNWscpu` may be added to install the Berkeley compatibility tools. `SUNWlibC`, `SUNWdoc`, and `SUNWman` can be added to install the on-line manual pages. Any additional packages should be scrutinized carefully and their security implications considered before the package is installed on the system.

STEP 1.5 INSTALLING PATCHES

PROBLEM: *Between the time the OS CD was created and the time the machine is installed, Sun has discovered a number of functionality and security-related bugs. Administrators must install the Recommended Patch Cluster appropriate for the current OS on the machine. Administrators should not install other patches unless specifically directed to do so by Sun.*

- Step 1.5.1. *[Certain releases of Solaris 2.6]* Remove any dependencies on `/usr/xpg4/bin/grep` (not installed as part of the "Core System Support" image) from the patchadd script:

```
cd /usr/sbin
mv patchadd patchadd-orig
sed s/\\/xpg4// patchadd-orig > patchadd
chown root:bin patchadd
chmod 555 patchadd
```

- Step 1.5.2. Download latest Recommended Patch Cluster from

```
ftp://sunsolve.sun.com/pub/patches/<osrel>_Recommended.zip
```

where `<osrel>` is the version of the OS that is being installed, e.g. 7 or 8.

CAVEAT:
For Solaris 2.6 and earlier, the patch cluster names are `<osrel>_Recommended.tar.Z`, rather than `.zip` files.

- Step 1.5.3. Use some mechanism to get this file into `/var/tmp` on the machine
- Step 1.5.4. Unpack Patch Cluster

```
cd /var/tmp
unzip -qq <osrel>_Recommended.zip
```

CAVEAT:
For Solaris 2.6 and earlier, use the command `zcat <osrel>_Recommended.tar.Z | tar xf -` to unpack the patch cluster.

CAVEAT:
The Sun Recommended Patch Cluster may not include all current security patches for the system. Administrators are encouraged to review the Patch Report file for their OS (available from the same location as the patch cluster file) or use the `patchdiag` command to locate and install additional security patches.

■ Step 1.5.5. Use install script

```
cd <osrel>_Recommended
./install_cluster -q -nosave
```

NOTE: If a patch installation fails with return code 8, then that patch applies to a package which has not been installed on the system. Other patches may fail with return code 2— these have already been applied to the OS image loaded from CD-ROM.

■ Step 1.5.6. Remove patch cluster from `/var/tmp`

```
cd /var/tmp
rm -rf /var/tmp/<osrel>_Recommended*
```

■ Step 1.5.7. `reboot`

STEP 2
OS Modification

STEP 2.1 PURGING BOOT DIRECTORIES OF UNNECESSARY SERVICES

PROBLEM: *Solaris starts many services at boot time which are dangerous or simply not useful. By renaming links in the* `/etc/rc*.d` *directories, the administrator prevents these processes from starting but make it easy to recreate the links in the event that one of these services must be invoked in the future (the new link names begin with a "." so they don't show up in the normal output of the* `ls` *command).*

- Step 2.1.1. `cd /etc/rc2.d`
- Step 2.1.2. Rename "auto configuration" related links

```
for file in S30sysid.net S71sysid.sys S72autoinstall
do
        mv $file .NO$file
done
```

CAVEAT:
Moving these files turns off Solaris' automatic reconfiguration features. Allowing any root user to easily reconfigure the system's network parameters is probably not a good idea.

- Step 2.1.3. Rename NFS-related links

```
for file in S73nfs.client S74autofs *cache*
do
        mv $file .NO$file
done
mv /etc/rc3.d/S15nfs.server /etc/rc3.d/.NOS15nfs.server
```

CAVEAT:
This makes the system unable to serve or mount file systems via NFS without administrator intervention. NFS is a huge security hole on any system.

- Step 2.1.4. Rename RPC related links

```
mv S71rpc .NOS71rpc
```

CAVEAT:
Renaming this files disables CDE, network information services such as NIS and NIS+, as well as certain commercial software (e.g., Legato Networker), and will impact NFS operations.
RPC-based services generally perform limited authentication and are a significant security risk.

NOTE: If the system must continue running with RPC services enabled, consider installing Wietse Venema's version of `rpcbind`, available from `ftp://ftp.porcupine.org/pub/security/rpcbind_2.1.tar.gz`.

■ Step 2.1.5. Disable `nscd`

```
mv S76nscd .NOS76nscd
```

NOTE: Some versions of the Netscape Navigator and Netscape HTTP Proxy as well as the Darwin Quicktime software will not function if `nscd` is disabled on the system. If this system is going to be a user desktop, administrators may wish to leave `nscd` enabled.

■ Step 2.1.6. *[Solaris 8]* Disable LDAP cache manager

```
mv S71ldap.client .NOS71ldap.client
```

NOTE: Do not perform this step if this machine will be an LDAP client. Consult relevant vendor documentation.

■ Step 2.1.7. Rename Sendmail start-up script

```
mv S88sendmail .NOS88sendmail
```

CAVEAT:
The host will now be unable to receive mail or act as a mail server. Electronic mail can still be sent from this host (see Step 2.10 for further information). Sendmail attacks are still popular mechanisms for gaining control of a system.

■ Step 2.1.8. Rename `expreserve` initiation script

```
mv S80PRESERVE .NOS80PRESERVE
```

CAVEAT:
`expreserve` is the program which recovers `vi` buffers when the system is rebooted. This program has historically had security problems which are probably fixed at this point.

STEP 2
OS Modification

STEP 2.2 NEW AND MODIFIED BOOT SERVICES

PROBLEM: *In order to disable certain services, it is necessary to install modified forms of the standard Solaris boot scripts. Rather than modifying the installed boot scripts (which could be later overwritten by patch installs or upgrades), the administrator should create new scripts in* `/etc/init.d` *and make or recreate the appropriate links in the* `/etc/rc?.d` *directories.*

- Step 2.2.1. *[Solaris 7 and earlier]* Create scripts to set default `umask` for system processes [1]

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

CAVEAT:
It is critical that the script names end in `.sh` or the `umask` command will not take effect on other script invocations.

NOTE: Starting with Solaris 8, the `CMASK` parameter in `/etc/default/init` controls the default umask for processes spawned by `init`.

- Step 2.2.2. Install the file shown in Appendix A as `/etc/init.d/newinetsvc`.
- Step 2.2.3. Replace the link to `/etc/init.d/inetsvc` in `/etc/rc2.d` with a link to the `newinetsvc` script

```
rm -f /etc/rc2.d/S72inetsvc
ln /etc/init.d/newinetsvc /etc/rc2.d/S72newinetsvc
chmod 744 /etc/init.d/newinetsvc
chown root:root /etc/init.d/newinetsvc
```

CAVEAT:
This replacement script disables DHCP, multicast routing, and `inetd` (and may end up disabling other services in future versions of Solaris). As a result of disabling `inetd`, administrators and users will be unable to `telnet`, `rlogin`, or otherwise access the machine over the network until `sshd` is installed according to Step 3.

- Step 2.2.4. *[Solaris 7 and later]* Make a copy of the `devfsadm` script in `/etc/init.d`

```
cp /etc/init.d/devfsadm /etc/init.d/newdevfsadm
chmod 744 /etc/init.d/newdevfsadm
chown root:root /etc/init.d/newdevfsadm
```

STEP 2
OS Modification

- Step 2.2.5. *[Solaris 7 and later]* Modify the `/etc/init.d/newdevfsadm` script and comment out the invocations for `devfsadmd` and `devfseventd`.
- Step 2.2.6. *[Solaris 7 and later]* Replace the link to the `devfsadm` script in `/etc/rcS.d`

```
rm -f /etc/rcS.d/S50devfsadm
ln -s /etc/init.d/newdevfsadm /etc/rcS.d/S50newdevfsadm
```

CAVEAT:
This will disable hot-pluggable hardware support on enterprise-class systems. Do not perform this step on systems with hot-pluggable hardware.

- Step 2.2.7. *[Solaris 8]* Make a copy of the `syslog` script in `/etc/init.d`

```
cp /etc/init.d/syslog /etc/init.d/newsyslog
chmod 744 /etc/init.d/newsyslog
chown root:root /etc/init.d/newsyslog
```

- Step 2.2.8. *[Solaris 8]* Modify the `newsyslog` script and add the `-t` flag to the `syslogd` invocation. The new line in the script should read

```
/usr/sbin/syslogd -t >/dev/msglog 2>&1 &
```

- Step 2.2.9. *[Solaris 8]* Replace the link to the syslog script in `/etc/rc2.d`

```
rm -f /etc/rc2.d/S74syslog
ln -s /etc/init.d/newsyslog /etc/rc2.d/S74syslog
```

CAVEAT:
Enabling the `-t` flag to `syslogd` causes the daemon to stop listening on UDP port 514 for messages from other hosts (though messages generated by processes on the local system will still be logged). Do not perform this step on a machine which is acting as a central loghost for other systems.

STEP 2 *OS Modification*

STEP 2.3 CONFIGURING KERNEL PARAMETERS

■ Step 2.3.1. Create new `/etc/init.d/netconfig` script to configure various network parameters [2] [6] [8] [9]

```
cat <<END_SCRIPT >/etc/init.d/netconfig
#!/sbin/sh
ndd -set /dev/tcp tcp_sack_permitted 2
ndd -set /dev/tcp tcp_conn_req_max_q0 8192
ndd -set /dev/tcp tcp_ip_abort_cinterval 60000
ndd -set /dev/ip ip_respond_to_timestamp 0
ndd -set /dev/ip ip_respond_to_timestamp_broadcast 0
ndd -set /dev/ip ip_respond_to_address_mask_broadcast 0
ndd -set /dev/ip ip_ignore_redirect 1
ndd -set /dev/ip ip_send_redirects 0
ndd -set /dev/ip ip_forward_src_routed 0
ndd -set /dev/ip ip_forward_directed_broadcasts 0
ndd -set /dev/ip ip_forwarding 0
ndd -set /dev/ip ip_strict_dst_multihoming 1
ndd -set /dev/arp arp_cleanup_interval 60000
ndd -set /dev/ip ip_ire_flush_interval 60000
END_SCRIPT
```

CAVEAT:
Starting with Solaris 8, the parameter `ip_ire_flush_interval` is called `ip_ire_arp_interval`. Make the appropriate change to the last line above on these systems. Setting the `tcp_sack_permitted` parameter is only required on Solaris 7— prior OS releases did not support this flag, and 2 is the default setting for Solaris 8 (note that setting `tcp_sack_permitted` to 2 may be cause problems when the machine attempts to connect to older Xyplex terminal servers). Similarly, `tcp_conn_req_max_q0` does not exist prior to Solaris 2.5.1 (and Solaris 2.5.1 machines must have recent versions of patches `103582` and `103630` installed to access this parameter).

NOTE: Administrators may also wish to add `ndd -set /dev/ip ip_respond_to_echo_broadcast 0` to prevent machines from responding to pings sent to the LAN broadcast address. Responses to broadcast pings can be helpful to local network administrators but can also leave the machine open to being used as an amplifier for Smurf and other denial-of-service type attacks. Pings from outside of an organization (particularly broadcast pings) should be blocked by that organization's firewall or network perimeter devices.

■ Step 2.3.2. Set ownership/permissions on `netconfig` script

```
chown root:root /etc/init.d/netconfig
chmod 744 /etc/init.d/netconfig
```

STEP 2
OS Modification

■ Step 2.3.3. Create link to `netconfig` script in `/etc/rc2.d`

```
cd /etc/rc2.d
ln -s ../init.d/netconfig S69netconfig
```

NOTE: The `netconfig` script should be run **immediately after** the `S69inet` script.

■ Step 2.3.4. *[Solaris 2.6 and later]* Prevent and log certain types of buffer overflow attacks by adding the following lines to `/etc/system` [5]

```
* Attempt to prevent and log stack-smashing attacks
set noexec_user_stack = 1
set noexec_user_stack_log = 1
```

CAVEAT:
Configuring these variables may rarely cause some poorly written software (including some revisions of the Netscape Navigator software) to misbehave. However, this step does provide significant additional security and should not be skipped without substantial reasons.

■ Step 2.3.5. Limit user resource consumption by adding the following lines to `/etc/system` [8]

```
* Set various parameters to more reasonable values
set maxuprc = 128
set sys:coredumpsize = 0
```

CAVEAT:
The last line prevents any process on the system from creating a core file. If a users needs to obtain a core file for debugging, the administrator will have to remove (or comment out) the last line above and reboot the system. On Solaris 7 and later systems, administrators may wish to investigate the `coreadm(1M)` utility for managing core files.

■ Step 2.3.6. Require NFS client requests to originate from the privileged port range by adding the following lines to `/etc/system`

```
* Require NFS clients to use privileged ports
set nfssrv:nfs_portmon = 1
```

NOTE: For Solaris 2.5 and earlier, the above line should read "`set nfs:nfs_portmon = 1`".

■ Step 2.3.7. Reboot the system in order to update kernel configuration

STEP 2.4 CLEANING HOUSE

PROBLEM: *Certain files should now be removed or simplified to assist in system auditing. For example, NFS-related configuration are removed files so the administrator can know when somebody has re-enabled NFS services on the machine.*

- Step 2.4.1. Remove NFS-related configuration files

```
rm /etc/auto_* /etc/dfs/dfstab
```

- Step 2.4.2. Remove empty `crontab` files

```
cd /var/spool/cron/crontabs
rm adm lp
```

- Step 2.4.3. `rm /etc/inet/inetd.conf /etc/inetd.conf`

STEP 2.5 FILE SYSTEM CONFIGURATION

PROBLEM: *The OS binaries in `/usr` should be protected from being replaced with trojan horse programs. Administrators should also attempt to stop rogue set-UID programs from showing up in other directories or on removable media by mounting file systems with the `nosuid` option. Unfortunately, the root file system cannot be mounted `nosuid` since `nosuid` also implies `nodev`. Administrators may also wish to investigate using AIDE (`http://www.cs.tut.fi/~rammer/aide.html`) or Tripwire (`http://www.tripwire.org`) to further monitor the integrity of their file systems.*

- Step 2.5.1. Mount `/usr` read-only in `/etc/vfstab`

```
/dev/dsk/c0t3d0s4 /dev/rdsk/c0t3d0s4 /usr  ufs 1 no ro
```

CAVEAT:
`/usr` may be remounted in read-write mode with the command `mount -o remount,rw /usr` but can only be set to read-only mode again with a reboot. Watch for suspicious reboots on the system because they may be a sign that somebody has modified a file in `/usr`.

■ Step 2.5.2. Mount other non-root ufs file systems `nosuid` to prevent set-UID programs executing here

```
/dev/dsk/c0t3d0s5 /dev/rdsk/c0t3d0s5 /var  ufs 1 no nosuid
/dev/dsk/c0t3d0s6 /dev/rdsk/c0t3d0s6 /local ufs 2 yes nosuid
```

NOTE: For Solaris 7 and later, administrators may wish to add the `logging` option to all writable UFS file systems (the last column of the lines above would read "`nosuid,logging`"). File system logging helps prevent file system inconsistencies which can slow or abort the system boot process.

■ Step 2.5.3. *[Solaris 8]* Mount the root file system with the `logging` option (the `remount` option also needs to be specified so that the other options to the file system take effect)

```
/dev/dsk/c0t3d0s0 /dev/rdsk/c0t3d0s0 /  ufs 1 no remount,logging
```

NOTE: A corrupted root file system can give an unprotected root shell to an attacker who has console access to the system.

■ Step 2.5.4. *[Solaris 7 and earlier]* Add the following lines to `/etc/rmmount.conf` [8]

```
mount hsfs -o nosuid
mount ufs -o nosuid
```

NOTE: This is the default for Solaris 8.

STEP 2
OS Modification

STEP 2.6 ADDITIONAL LOGGING

PROBLEM: *By default, Solaris does not capture syslog events sent to* `LOG_AUTH`. *This information is very useful since it contains information on unsuccessful login attempts, successful and failed* `su` *attempts, reboots, and a wealth of other security-related information. System accounting can also be used to provide interesting information about system usage so that abnormal patterns can be detected. Administrators may wish to investigate the freely available* `sudo` *software (*`http://www.courtesan.com/sudo/`*) which can capture much more information about commands run as a privileged user (as well as providing a higher level of security over the standard* `su` *command).*

CAVEAT:
The white space between the two columns must be tabs or the file will not be parsed properly. Also note that all events of info severity and higher will be logged to this file. Administrators may also wish to send these logs to a different machine as well so they can have a copy to compare against in case of a break in.

- Step 2.6.1. Add this line to `/etc/syslog.conf`

```
auth.info     /var/log/authlog
```

- Step 2.6.2. Create `/var/log/authlog`

```
touch /var/log/authlog
chown root /var/log/authlog
chmod 600 /var/log/authlog
```

- Step 2.6.3. Create `/var/adm/loginlog` to capture failed logins

```
touch /var/adm/loginlog
chmod 600 /var/adm/loginlog
chown root:sys /var/adm/loginlog
```

- Step 2.6.4. Install the log rotation script from Appendix E in some directory on the system (e.g., `/usr/local/bin`).

CAVEAT:
Logs will only be retained for four weeks, though the number of weeks kept may be increased by changing the argument in the last column. Administrators may wish to consider some mechanism for permanent storage of this data (e.g., tape archive).

- Step 2.6.5. Using the `crontab` command, add the following lines to root's crontab

```
30 3 * * 0 /usr/local/bin/rotate /var/log/authlog 600 4
35 3 * * 0 /usr/local/bin/rotate /var/adm/loginlog 600 4
```

NOTE: The path name of the `rotate` script depends on where this script was installed in the previous step.

■ Step 2.6.6. Modify `/etc/default/cron` to read

```
CRONLOG=YES
```

NOTE: The `cron` log, `/var/cron/log`, should be reviewed regularly for suspicious behavior.

■ Step 2.6.7. Edit `/etc/init.d/perf` and follow the instructions located there to uncomment the indicated lines which cause a marker to be placed in the system accounting logs when the machine boots.

■ Step 2.6.8. Add the following lines to the `crontab` for user `sys` (use "`crontab -e sys`" to modify this file)

```
0,20,40 * * * * /usr/lib/sa/sa1
45 23 * * * /usr/lib/sa/sa2 -s 0:00 -e 23:59 -i 1200 -A
```

NOTE: System accounting data will now be captured every 20 minutes and daily reports written to `/var/adm/sa`. This data will be overwritten on a monthly cycle; administrators may wish to archive older data to another location for preservation.

STEP 2.7 ENABLE KERNEL-LEVEL AUDITING

PROBLEM: *Sun's Basic Security Module (BSM) auditing functionality can provide the administrator with a detailed report of all system activity. However, the output can consume enormous amounts of disk space and can be cryptic, at best, to review. For more information on configuring, managing, and interpreting BSM audit trails, see the "SunSHIELD Basic Security Module Guide" (*`http://docs.sun.com/ab2/coll.47.8/SHIELD/`*).*

CAVEAT:
Audit logs will not actually be generated until the system is rebooted.

❑ Step 2.7.1. (Advanced) Enable BSM by running the following command

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

❑ Step 2.7.2. *(Advanced)* Configure the `/etc/security/audit_control` file

```
dir:/var/audit
flags:lo,ad,-all,^-fm
naflags:lo,ad
minfree:20
```

NOTE: There is no clear consensus on the "proper" configuration for `/etc/security/audit_control`. Administrators are encouraged to read the relevant vendor documentation and make appropriate choices for their site.

❑ Step 2.7.3. *(Advanced)* Add the following lines to root's crontab to force new audit log files to be started every hour:

```
0 * * * * /usr/sbin/audit -n
```

NOTE: Audit logs can consume enormous amounts of disk space. The administrator is encouraged to compress old audit logs, and consider archiving older logs to another system or to off-line storage (e.g., tape or CD-ROM). Some organizations (particularly classified projects) may have strict guidelines regarding retention of audit data.

❑ Step 2.7.4. (Advanced) Reboot the system to activate audit logging

STEP 2.8 USER ACCESS CONTROL

CAVEAT:
This step may cause patch installs and other automated install scripts which are dependent upon the existence of certain user names to fail. Skipping this step for the sake of convenience should not substantially impact the security of the system. However, the shell for these users should be set to `/dev/null` (see next step) if they remain on the system.

PROBLEM: *Tight user access controls are the first line of defense for most Unix systems. Administrators should carefully inspect all user accounts in their password file and remove or disable unnecessary users. Network address based authentication (`.rhosts`) should be disabled, as well as FTP access for system users; this is a "strength in depth" effort since the system should not be running these services at all. Non-privileged access to the `cron` system should be tightly controlled.*

■ Step 2.8.1. Clean out `/etc/passwd` file

```
for user in uucp nuucp lp smtp listen nobody4
do
      /usr/sbin/passmgmt -d $user
done
```

■ Step 2.8.2. Make `/dev/null` the shell for other non-root users in `/etc/passwd`.

```
for user in adm daemon bin nobody noaccess
do
      /usr/sbin/passmgmt -m -s /dev/null $user
done
```

NOTE: Administrators who wish to log failed login attempts may use the `noshell` program provided with the Titan Security Package. [3]

■ Step 2.8.3. *[Solaris 7 and earlier]* Create `/etc/ftpusers`

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

NOTE: This is the default for Solaris 8.

■ Step 2.8.4. *[Solaris 2.6 and later]* Remove `.rhosts` support from `/etc/pam.conf`

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

■ Step 2.8.5. Create empty files to attempt to thwart remote attacks

```
for file in /.rhosts /.shosts /.netrc /etc/hosts.equiv
do
      cp /dev/null $file
      chown root:root $file
      chmod 000 $file
done
```

■ Step 2.8.6. Only root should be allowed to run the `crontab` and `at` commands

```
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
```

STEP 2.9 STATUTORY WARNINGS

PROBLEM: *It is widely believed that displaying appropriate warning messages when users access a system will assist in prosecuting computer crime cases and defending court challenges related to the machine. Consult local legal counsel for the appropriate statutes and on the wording of all messages.*

■ Step 2.9.1. Create `/etc/issue` and `/etc/motd` files with an appropriate statutory warning. An example of such a warning would be [4]

```
This system is for the use of authorized users only.
Individuals using this computer system without
authority, or in excess of their authority, are
subject to having all of their activities on this
system monitored and recorded by system personnel.

In the course of monitoring individuals improperly
using this system, or in the course of system
maintenance, the activities of authorized users may
also be monitored.

Anyone using this system expressly consents to such
monitoring and is advised that if such monitoring
reveals possible evidence of criminal activity, system
personnel may provide the evidence of such monitoring
to law enforcement officials.
```

- Step 2.9.2. *[Solaris 2.6 and later]* Create an `/etc/default/telnetd` file containing the following line

```
BANNER="Authorized uses only.  All access may be logged.\n"
```

- Step 2.9.3. *[Solaris 2.6 and later]* Create an `/etc/default/ftpd` file containing the following lines

```
BANNER="Authorized uses only.  All access may be logged."
UMASK=022
```

- Step 2.9.4. Set appropriate access controls on all files

```
chown root:sys /etc/motd
chown root:root /etc/issue
chmod 644 /etc/motd /etc/issue
chown root:sys /etc/default/telnetd /etc/default/ftpd
chmod 444 /etc/default/telnetd /etc/default/ftpd
```

- Step 2.9.5. Set boot-level warning message

```
eeprom oem-banner="Authorized uses only.  All access may be logged."
eeprom oem-banner\?=true
```

STEP 2.10 SENDMAIL

PROBLEM: *The `sendmail` program can often be a security issue. Most machines only need to be able to send out email to a relay host, and can therefore run with most Sendmail functionality disabled.*

- Step 2.10.1. Administrators may wish to replace the Solaris Sendmail binary with the Open Source version available from `http://www.sendmail.org/`.

NOTE: Administrators may also wish to consider using an alternate mail transfer agent (MTA) such as QMail (`http://www.qmail.org/`), Postfix (`http://www.postfix.org/`), or Exim (`http://www.exim.org/`) which may be more secure. However, the subsequent configuration steps in this section will not be appropriate for these MTAs. Consult the relevant documentation for the MTA chosen.

■ Step 2.10.2. Use the minimal `/etc/mail/sendmail.cf` file shown in Appendix B.

NOTE: This `sendmail.cf` is only appropriate for machines where local mail delivery is not required. Be sure to change the relay host name `mailhost` to be the name of the appropriate machine for the local site.

CAVEAT:
This minimal `sendmail.cf` uses unqualified sender addresses which may cause problems on relay hosts running Sendmail 8.9 and later. For an alternate mechanism for creating a fairly minimal `sendmail.cf` file, see the `nullclient` feature in the Open Source Sendmail documentation.

■ Step 2.10.3. Add this line to root's crontab

```
0 * * * * /usr/lib/sendmail -q
```

CAVEAT:
Electronic mail will normally flow out of the system immediately but some mail may be queued because the remote destination is unavailable for some period. Since the Sendmail daemon is not running (see Step 2.1.7), the above line is required to flush the mail queue hourly.

STEP 2.11 MISCELLANEOUS

PROBLEM: *Several other minor modifications should be made to the system to enable certain functionality or improve security.*

■ Step 2.11.1. Turn on EEPROM security functionality

```
eeprom security-mode=command
```

CAVEAT:
The administrator will be prompted for a password which must be entered before any PROM level command will be executed. If this password is forgotten, the administrator may be unable to reboot off of CD-ROM or otherwise change boot parameters.

NOTE: The number of failed EEPROM logins can be monitored with the command "`eeprom security-#badlogins`". Administrators may wish to zero this value initially ("`eeprom security-#badlogins=0`") to clear any residual data.

■ Step 2.11.2. Edit `/etc/default/login`. Uncomment the `UMASK` line. Uncomment the `TIMEOUT` line and set the value of `TIMEOUT` to 60 (seconds). For Solaris 8, uncomment the `SYSLOG_FAILED_LOGINS` line and set this value to 0 ("zero", which means log all failed login attempts).

STEP 2
OS Modification

- Step 2.11.3. *[Solaris 2.6 and later]* Disable the `Stop-A` abort sequence by editing `/etc/default/kbd` and setting `KEYBOARD_ABORT=disabled`

CAVEAT:
If the system becomes hung or wedged, the administrator will be forced to perform a hard power down to interrupt the system.

- Step 2.11.4. *[Solaris 2.6 and later]* Edit `/etc/default/inetinit` and set `TCP_STRONG_ISS=2` to cause the system to use a better TCP sequence number generation algorithm

- Step 2.11.5. The administrator may wish to turn on password aging functionality by setting the value of `MAXWEEKS`, `MINWEEKS`, and `WARNWEEKS` in `/etc/default/passwd`.

- Step 2.11.6. Edit `/etc/inittab` and remove the following line

```
sc:234:respawn:/usr/lib/saf/sac -t 300
```

CAVEAT:
This will disable the login prompt on the system serial devices so that modems and terminals will not function. Note that serial console devices will continue to function even if this line is removed.

STEP 2.12 FIX-MODES SCRIPT

PROBLEM: *The default permissions on many files are somewhat insecure.* `fix-modes` *was written by Caspar Dik to correct these permissions for Solaris 2.2 through Solaris 8. Since the target machine does not have any of the compiler tools installed, the administrator will need to complete the first three steps on some other machine.*

- Step 2.12.1. Obtain `fix-modes` software from

```
ftp://ftp.fwi.uva.nl/pub/solaris/fix-modes.tar.gz
```

- Step 2.12.2. Unpack sources

```
mkdir fix-modes
mv fix-modes.tar.gz fix-modes
cd fix-modes
gunzip -c fix-modes.tar.gz | tar xf -
```

- Step 2.12.3. Build software on some other machine with a compiler. Just running the "`make`" command in the `fix-modes` directory should suffice, though if `gcc` is being used to build this software run "`make CC=gcc`".

- Step 2.12.4. Move `fix-modes` distribution to machine being secured. The best approach may be to simply `tar` up the `fix-modes` directory created in Step 2.12.2 and copy the `tar` file over to the target platform.

- Step 2.12.5. Run `fix-modes` shell script from the command line.

```
sh fix-modes
```

STEP 3.1 BUILDING AND INSTALLING THE TCP WRAPPERS SOFTWARE

PROBLEM: *TCP Wrappers allow the administrator to control access to certain services by IP address. OpenSSH can be compiled with TCP Wrappers functionality but only if the TCP Wrappers software is built first. Note that this build and install (as well as the other compiles in this Section) will have to be done on some other system than the secure platform being configured because the secure platform has no compilers or other tools.*

■ Step 3.1.1. Obtain TCP Wrappers source code from

```
ftp://ftp.porcupine.org/pub/security/tcp_wrappers_<vers>.tar.gz
```

NOTE: For Solaris 8, administrators should download the "`-ipv6`" version of TCP Wrappers which supports IPv6 networking.

■ Step 3.1.2. Unpack sources

```
gunzip -c tcp_wrappers_<vers>.tar.gz | tar xf -
cd tcp_wrappers_<vers>
```

■ Step 3.1.3. Modify top-level `Makefile`

```
chmod 644 Makefile
vi Makefile
```

In particular, uncomment the correct value of `REAL_DAEMON_DIR` for the system. Also modify the `FACILITY` variable so all logging goes to `LOG_AUTH`.

■ Step 3.1.4. Build software

```
make sunos5
```

Add `CC=gcc` to the command line above if `gcc` is being used to build the software.

■ Step 3.1.5. Install resulting files in some useful directory

```
mkdir -p /usr/local/sbin /usr/local/include /usr/local/lib
for file in safe_finger tcpd tcpdchk tcpdmatch try-from
do
      /usr/sbin/install -s -f /usr/local/sbin \
            -m 0555 -u root -g daemon $file
done
/usr/sbin/install -s -f /usr/local/include \
      -m 0444 -u root -g daemon tcpd.h
/usr/sbin/install -s -f /usr/local/lib \
      -m 0555 -u root -g daemon libwrap.a
```

NOTE: Administrators may also wish to install the manual pages.

STEP 3.2 BUILDING AND INSTALLING ZLIB

PROBLEM: *Before OpenSSH can be compiled, the Zlib data compression software library must be compiled and installed. Note that Solaris 8 ships with pre-compiled copy of the Zlib software which may be installed by adding package* `SUNWzlib` *to the system (this package is on the second OS CD-ROM).*

■ Step 3.2.1. Download Zlib software from

```
ftp://ftp.freesoftware.com/pub/infozip/zlib/zlib.tar.gz
```

■ Step 3.2.2. Unpack software archive

```
gunzip -c zlib.tar.gz | tar xf -
cd zlib-*
```

■ Step 3.2.3. Run the `configure` script, build the software, and install

```
sh configure
make
make install
```

NOTE: By default the software will be installed under `/usr/local`. This path can be changed by supplying the `--prefix=<dir>` option to the configure script.

STEP 3.3 BUILDING AND INSTALLING OPENSSL

PROBLEM: *OpenSSH requires the Open Source OpenSSL library as well. Note that the auto-configuration script supplied with OpenSSL requires that Perl v5 be installed on the system. Perl ships with Solaris 8, but may need to be compiled and installed on other platforms (for more information see* `http://www.perl.com/`*).*

- Step 3.3.1. Download the OpenSSL software from

```
ftp://ftp.openssl.org/source/openssl-<vers>.tar.gz
```

- Step 3.3.2. Unpack the source archive

```
gunzip -c openssl-<vers>.tar.gz | tar xf -
cd openssl-<vers>
```

- Step 3.3.3. Run the `config` script, build the software, and install

```
sh config
make
make install
```

NOTE: By default the software will be installed under `/usr/local`. This path can be changed by supplying the `--prefix=<dir>` option to the `config` script.

STEP 3.4 BUILDING AND INSTALLING THE OPENSSH SOFTWARE

PROBLEM: *Having built and installed all of the supporting library code, the OpenSSH software can now be compiled.*

- Step 3.4.1. Download source code. Pointers to various FTP sites can be found at `http://www.openssh.com/ftp.html`. Note that Solaris administrators should download the latest "`p`" (portable) release from the "`portable`" subdirectory at the FTP site.

- Step 3.4.2. Unpack sources

```
gunzip -c openssh-<version>.tar.gz | tar xf -
cd openssh-<version>
```

STEP 3
Installing OpenSSH With TCP Wrappers Support

■ Step 3.4.3. Build software

```
setenv CFLAGS -I/usr/local/include
setenv LDFLAGS -L/usr/local/lib
sh configure --prefix=/usr/local --with-tcp-wrappers \
     --without-rsh --disable-suid-ssh
make
```

CAVEAT:
The values of `CFLAGS` and `LDFLAGS` are dependent upon the locations of the TCP Wrappers, Zlib, and OpenSSL libraries and header files from Steps 3.1 through 3.3.

NOTE: There are many, many configure options for OpenSSH: consult the `INSTALL` file for more information. Administrators are strongly encouraged to consider using a one-time password scheme.

■ Step 3.4.4. The `sshd` and `ssh-keygen` binaries should be copied to the secure host and installed in some useful directory (e.g., `/usr/local/bin`). The `ssh_prng_cmds` file should be installed in the location compiled into the OpenSSH binaries (`/usr/local/etc` by default). The administrator may wish to copy other files from the OpenSSH distribution to the secure system, such as the OpenSSH client programs or the `sftp-server` binary.

STEP 3.5 CONFIGURING TCP WRAPPERS AND THE SSH DAEMON

PROBLEM: *Once the OpenSSH software is installed on the host's local drives, the daemon and TCP Wrappers functionality must be configured for proper and secure operations.*

■ Step 3.5.1. Create `/etc/hosts.allow` file for TCP Wrappers. This file might look like:

```
ALL: <net1>/<mask1>, ..., <netN>/<maskN>
```

where `<netx>` is one of the local site networks, and `<maskx>` is the corresponding netmask. Consult the TCP wrappers documentation for further information.

■ Step 3.5.2. Create `/etc/hosts.deny` file for TCP Wrappers:

```
echo 'ALL: ALL: /usr/bin/mailx \
     -s "%s: connection attempt from %a" \
     root@localdomain.com' >/etc/hosts.deny
```

NOTE: Replace the email address `root@localdomain.com` with some appropriate address for the local site.

- Step 3.5.3. Create the `/etc/sshd_config` file for the SSH server. A sample `/etc/sshd_config` file is available in Appendix C.

- Step 3.5.4. Set appropriate file permissions on configuration files

```
cd /etc
chown root:root sshd_config hosts.allow hosts.deny
chmod 600 sshd_config hosts.allow hosts.deny
```

- Step 3.5.5. Generate server key files

```
/usr/local/bin/ssh-keygen -b 1024 -N '' -f /etc/ssh_host_key
/usr/local/bin/ssh-keygen -d -N '' -f /etc/ssh_host_dsa_key
```

NOTE: The path for the `ssh-keygen` binary depends on where the administrator installed the software (see Step 3.4.4). The path names chosen for the `-f` option should match the corresponding path names in the `sshd_config` file (see Step 3.5.3).

- Step 3.5.6. Create an `/etc/init.d/sshd` script for starting the SSH server at boot time. A sample script is available in Appendix D.

- Step 3.5.7. Create link to sshd startup script in `/etc/rc2.d`

```
chmod 744 /etc/init.d/sshd
cd /etc/rc2.d
ln -s ../init.d/sshd S75sshd
```

This causes the SSH daemon to start running right after `syslogd` has been activated and can receive logging messages.

- Step 3.5.8. Start SSH daemon

```
/etc/init.d/sshd start
```

STEP 4.1 MAKE A BACKUP

PROBLEM: *Backups are necessary not only for disaster recovery but also if there is a security incident which requires comparing OS files against a "gold" image. Backups can also be used to spawn new systems with duplicate configurations. The procedure below will make a backup of the current version of the system, but administrators should ensure that a regular backup schedule is followed for all critical systems.*

- Step 4.1.1. Boot the system in single-user mode

```
reboot -- -s
```

- Step 4.1.2. Mount all file systems

```
fsck
mount -a
```

- Step 4.1.3. Back up all ufs file systems to tape or other media twice

```
mt /dev/rmt/0 rewind
for dir in / /usr /var /local
do
      ufsdump 0f /dev/rmt/0n $dir
done
mt /dev/rmt/0 rewoffl
```

Repeat the above steps on new media. Be sure to back up any other file systems that were created when building the machine.

CAVEAT:
The above commands are appropriate for an 8mm type tape device. Consult the applicable documentation for the system's media choice.

- Step 4.1.4. Write protect both tapes.
- Step 4.1.5. Store one tape locally and the other off-site.
- Step 4.1.6. Make sure both tapes are in physically secure locations which can only be accessed by trusted personnel.

STEP 4.2 PHYSICALLY SECURE THE MACHINE [7]

PROBLEM: *It is a fact of life that anybody who can get access to the console of a standard Unix system can get superuser access to the device. Booting from CD-ROM, tape, or other portable media, hard crashing the system to come up in single-user mode and forcing a manual fsck to get a root shell, and outright theft of external drives are all mechanisms for compromise.*

- ■ Step 4.2.1 Place the server in a locked room with access controlled by the administrator. Verify that drop-down ceilings and raised floors do not allow uncontrolled access.
- ❑ Step 4.2.2 (Advanced) Provide electronic access control and recording for the server room.
- ■ Step 4.2.3. Provide temperature and humidity controls sufficient to avoid damage to the equipment. One uninterruptible power supply (UPS) vendor provides an optional attachment that monitors temperature and humidity and can send administrative alerts and emails and can page the system administrator.
- ❑ Step 4.2.4. (Advanced) Provide one or more halon-type automatic fire extinguishers.
- ■ Step 4.2.5. Install a UPS and associated software that enables the server to shut down automatically and safely when the power in the UPS is about to be exhausted.
- ❑ Step 4.2.6. (Advanced) Use surveillance cameras to record who accesses the equipment
- ■ Step 4.2.7. Lock the CPU case and set up a system to ensure the key is protected and yet easily available to the administrator. Make a back-up key and protect it off-site in a secure disaster recovery site or a safety deposit box or similarly protected place. Lock the server down with a cable or in a rack.
- ■ Step 4.2.8. Arrange the room so that the keyboard is hidden from prying eyes at windows or other vantage points.
- ❑ Step 4.2.9. (Advanced) Consider providing additional shielding against electronic interference or eavesdropping.

STEP 4.3 PROVIDE ADEQUATE NETWORK SECURITY

PROBLEM: *Since TCP Wrappers functionality is being used to selectively permit SSH sessions by IP address, local administrators must ensure that outsiders cannot send the machine packets with spoofed source addresses which purport to be from trusted hosts. Outsiders should only have access to the services they absolutely need.*

- Step 4.3.1. Configure nearby routers to block spoofed packets.
- Step 4.3.2. Stop smurfing and other denial-of-service type attacks.
- Step 4.3.3. Only grant outside access to small list of services.

A FINAL WORD

Despite all the good work of careful administrators and well-behaved users, some sites encourage security breaches by assuming all new employees and contracted staff members are honest and stable. If the organization's information assets are valuable, then it makes no sense to give keys to thieves. Conduct detailed background checks on each employee or contractor with root privileges. Require bonding of contractor personnel. Establish a contingency plan in case the current system administrators become unavailable or malicious. If the value of the information being protected is very high (as in law enforcement, financial services, or national security) make the checks extend to a full five years back into the person's history.

REFERENCES

[1] Brad Powell, Dan Farmer, Matt Archibald, “The Titan Security Package”, http://www.fish.com/titan/, From module add-umask.sh

[2] ibid., From modules adjust-arp-timers.sh, disable-ip-holes.sh

[3] ibid., From module disable-accounts.sh

[4] ibid., From module create-issue.sh

[5] ibid., From module fix-stack.sol2.6.sh

[6] Jens Voeckler, “Solaris — Tuning Your TCP/IP Stack”, http://www.rvs.uni-hannover.de/people/voeckler/tune/EN/tune.html

[7] Jesper Johansson and Gene Schultz (ed.), “Windows NT Security Step-by-Step”, SANS Institute, 1999.

[8] Jean Chouanard (et al), YASSP, http://www.yassp.org/

[9] Alex Noordergraaf and Keith Watson, “Solaris Operating Environment Network Settings for Security”, Sun BluePrints OnLine (http://www.sun.com/blueprints/1299/network.pdf), December 1999.

APPENDIX A

REPLACEMENT /ETC/INIT.D/NEWINETSVC SCRIPT

```
#!/sbin/sh

/usr/sbin/ifconfig -au netmask + broadcast +

if [ -f /usr/sbin/in.named -a -f /etc/named.conf ]; then
      /usr/sbin/in.named
      echo "starting internet domain name server."
fi

#mcastif=`uname -n`
#echo "Setting default interface for multicast: \c"
#/usr/sbin/route add -interface -netmask "240.0.0.0" "224.0.0.0" "$mcastif"

# Run inetd in "standalone" mode (-s flag)
#/usr/sbin/inetd -s -t
```

APPENDIX B

MINIMAL /ETC/MAIL/SENDMAIL.CF FILE

```
# Minimal client sendmail.cf

### Defined macros
# The name of the mail hub — PUT APPROPRIATE HOSTNAME FOR LOCAL SITE HERE!!!
DRmailhost

# Define version
V8

# Whom errors should appear to be from
DnMailer-Daemon

# Formatting of the UNIX from line
DlFrom $g $d

# Separators
Do.:%@!^=/[]

# From of the sender's address
Dq<$g>

# Spool directory
OQ/usr/spool/mqueue

### Mailer Delivery Agents
# Mailer to forward mail to the hub machine
Mhub,    P=[IPC], S=0, R=0, F=mDFMuCX, A=IPC $h
# Sendmail requires these, but are not used
Mlocal, P=/dev/null, F=rlsDFMmnuP, S=0, R=0, A=/dev/null
Mprog,   P=/dev/null, F=lsDFMeuP, S=0, R=0, A=/dev/null

### Rule sets — WHITESPACE BETWEEN COLUMNS MUST BE TABS!!!

S0
R@$+          $#error $:                  Missing user name
R$+           $#hub $@$R $:$1             forward to hub

S3
R$*<>$*       $n                          handle <> error address
R$*<$*>$*     $2                          basic RFC822 parsing
```

APPENDIX C

SSH SERVER CONFIG FILE

```
Port 22
ListenAddress 0.0.0.0
Protocol 2,1
SyslogFacility AUTH
LogLevel INFO

PidFile /etc/sshd.pid
HostDSAKey /etc/ssh_host_dsa_key
HostKey /etc/ssh_host_key
KeyRegenerationInterval 900
ServerKeyBits 1024

LoginGraceTime 180
X11Forwarding yes
StrictModes yes
KeepAlive no
UseLogin no
CheckMail no
PrintMotd no

PasswordAuthentication yes
PermitEmptyPasswords no
PermitRootLogin no
IgnoreRhosts yes
RhostsAuthentication no
RhostsRSAAuthentication no
IgnoreUserKnownHosts yes
RSAAuthentication yes
DSAAuthentication yes
```

APPENDIX D

SSH STARTUP SCRIPT

```
#!/sbin/sh

case "$1" in
'start')
      if [ -x /usr/local/sbin/sshd -a -f /etc/sshd_config ]; then
            /usr/local/sbin/sshd -f /etc/sshd_config
      fi
      ;;
'stop')
      kill `cat /etc/sshd.pid`
      ;;
*)
      echo "Usage: $0 { start | stop }"
      ;;
esac
exit 0
```

APPENDIX E

LOG ROTATION SCRIPT

```
#!/bin/ksh

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

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

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

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

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

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

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

/etc/rc2.d/S74syslog stop
/etc/rc2.d/S74syslog start
```

APPENDIX F

OTHER RESOURCES

Many other individuals and organizations have created similar procedures for managing the security of their Solaris systems. Some of these are even available on the Web. Sabernet maintains similar documents on OS hardening for Solaris and several other operating systems (see `http://www.sabernet.net/papers/`). Sean Boran has developed several different hardening procedures for Solaris (`http://www.boran.com/security/sp/Solaris_hardening.html`). Reg Quiton maintains a number of interesting Solaris-related security documents at the University of Waterloo (`http://ist.uwaterloo.ca/security/howto/`). Sun publishes several Security white papers as part of their Sun Blueprints™ series (`http://www.sun.com/blueprints/browsesubject.html#security`). The Solaris Security FAQ at SunWorld Online (`http://www/sunworld.com/sunworldonline/common/security-faq.html`) contains a wealth of Solaris-security related information, including information based on an earlier version of this procedure.

Various systems have been developed to automatically configure different security settings on Solaris systems. This guidebook attempts to remain in loose synchronization with the YASSP toolkit (`http://www.yassp.org/`). Other similar projects include the TITAN Project (`http://www.fish.com/titan/`), and Sun's own JASS toolkit (`http://www.sun.com/blueprints/tools/`). The Bastille Project (`http://bastille-linux.sourceforge.net/`) attempts to perform similar configuration for Linux systems. Note that David Brumley has written a paper which compares the features of YASSP and TITAN against an earlier release of this guide (`http://www.theorygroup.com/Theory/`).

Documentation on various Sun kernel parameters can be found both on Sun's documentation server (`http://docs.sun.com/ab2/coll.707.1/SOLTUNEPARAMREF/`), as well as at Jens Voeckler's excellent site (`http://www.rvs.uni-hannover.de/people/voeckler/tune/EN/tune.html`).

CHECKLIST

STEP 1 BASIC OS INSTALLATION

Step 1.1: Boot-time Configuration	Name of person Responsible	Date Completed	Initials
Action 1.1.1. Boot from most current Solaris OS CD-ROM			
Action 1.1.2. Enter host name			
Action 1.1.3. Select “Networked”			
Action 1.1.4. Enter IP address			
Action 1.1.5. Select “None” for name service			
Action 1.1.6. Enter appropriate netmask information			
Action 1.1.7. Select time zone			
Action 1.1.8. Verify that the date/time presented by the system is correct			

Step 1.2: Minimal OS Installation	Name of person Responsible	Date Completed	Initials
Action 1.2.1. Choose “Initial” install			
Action 1.2.2. Configure the machine as a “Standalone” server			
Action 1.2.3. Select “Core System Support”			
Action 1.2.4. Lay out file system on disks			
Action 1.2.5. Do not choose to mount any remote file systems			
Action 1.2.6. Select “Reboot” after install and begin installation			

Comments:

CHECKLIST

STEP 1 BASIC OS INSTALLATION

Step 1.3: Post Install / Networking Configuration	Name of person Responsible	Date Completed	Initials
Action 1.3.1. Set root password as appropriate			
Action 1.3.2. Create `/etc/defaultrouter`			
Action 1.3.3. Create `/etc/notrouter`			
Action 1.3.4. Create `/etc/resolv.conf`			
Action 1.3.5. Modify `/etc/nsswitch.conf`			
Action 1.3.6. Reboot			

Step 1.4: Adding Additional Packages	Name of person Responsible	Date Completed	Initials
Action 1.4.1. Insert the first OS media CD-ROM			
Action 1.4.2. Mount the OS media			
Action 1.4.3. `cd /mnt/Solaris_*/Product`			
Action 1.4.4. Add the terminfo database and system accounting related packages			

Step 1.5: Installing Patches	Name of person Responsible	Date Completed	Initials
Action 1.5.1. Remove any dependencies on `/usr/xpg4/bin/grep`			
Action 1.5.2. Mount the OS media			
Action 1.5.3. Put the Patch Cluster in `/var/tmp` on the machine			
Action 1.5.4. Unpack Patch Cluster			
Action 1.5.5. Use install script			
Action 1.5.6. Remove patch cluster from `/var/tmp`			
Action 1.5.7. `reboot`			

Comments:

CHECKLIST

STEP 2 OS MODIFICATIONS

Step 2.1: Purging Boot Directories of Unnecessary Services	Name of person Responsible	Date Completed	Initials
Action 2.1.1. `cd /etc/rc2.d`			
Action 2.1.2. Rename "auto configuration" related links			
Action 2.1.3. Rename NFS-related links			
Action 2.1.4. Rename RPC related links			
Action 2.1.5. Disable `nscd`			
Action 2.1.6. Disable LDAP cache manager			
Action 2.1.7. Rename Sendmail start-up script			
Action 2.1.8. Rename `expreserve` initiation script			

Step 2.2: New and Modified Boot Services	Name of person Responsible	Date Completed	Initials
Action 2.2.1. Set default umask for system processes			
Action 2.2.2. Install `/etc/init.d/newinetsvc` script			
Action 2.2.3. Replace the link to `/etc/init.d/inetsvc` in `/etc/rc2.d`			
Action 2.2.4. Make a copy of the `devfsadm` script in `/etc/init.d`			
Action 2.2.5. Modify the `/etc/init.d/newdevfsadm` script			
Action 2.2.6. Replace the link to the `devfsadm` script in `/etc/rcS.d`			
Action 2.2.7. Make a copy of the `syslog` script in `/etc/init.d`			
Action 2.2.8. Modify the `newsyslog` script			
Action 2.2.9. Replace the link to the `syslog` script in `/etc/rc2.d`			

Comments:

CHECKLIST

STEP 2 OS MODIFICATIONS

Step 2.3: Configuring Kernel Parameters	Name of person Responsible	Date Completed	Initials
Action 2.3.1. Create `/etc/init.d/netconfig` script			
Action 2.3.2. Set ownership/permissions on `netconfig` script			
Action 2.3.3. Create link to `netconfig` script in `/etc/rc2.d`			
Action 2.3.4. Prevent and log certain types of buffer overflows			
Action 2.3.5. Limit user resource consumption			
Action 2.3.6. Require NFS client requests to originate from privileged ports			
Action 2.3.7. Reboot the system in order to update kernel configuration			

Step 2.4: Cleaning House	Name of person Responsible	Date Completed	Initials
Action 2.4.1. Remove NFS-related configuration files			
Action 2.4.2. Remove empty `crontab` files			
Action 2.4.3. `rm /etc/inet/inetd.conf /etc/inetd.conf`			

Step 2.5: File System Configuration	Name of person Responsible	Date Completed	Initials
Action 2.5.1. Mount `/usr` read-only in `/etc/vfstab`			
Action 2.5.2. Mount other non-root ufs file systems `nosuid`			
Action 2.5.3. Mount the root file system with the `logging` option			
Action 2.5.4. Add lines to `/etc/rmmount.conf`			

Comments:

CHECKLIST

STEP 2 OS MODIFICATIONS

Step 2.6: Additional Logging	Name of person Responsible	Date Completed	Initials
Action 2.6.1. Modify `/etc/syslog.conf`			
Action 2.6.2. Create `/var/log/authlog`			
Action 2.6.3. Create `/var/adm/loginlog`			
Action 2.6.4. Install the log rotation script from Appendix E			
Action 2.6.5. Add lines to root's `crontab`			
Action 2.6.6. Modify `/etc/default/cron`			
Action 2.6.7. Edit `/etc/init.d/perf`			
Action 2.6.8. Add lines to the `crontab` for user sys			

Step 2.7: Enable Kernel-Level Auditing	Name of person Responsible	Date Completed	Initials
❑ Action 2.7.1. (Advanced) Enable BSM			
❑ Action 2.7.2. (Advanced) Configure the `/etc/security/audit_control` file			
❑ Action 2.7.3. (Advanced) Force new audit log files to be started every hour			
❑ Action 2.7.4 (Advanced) Reboot the system to activate audit logging			

Step 2.8: User Access Control	Name of person Responsible	Date Completed	Initials
Action 2.8.1. Clean out `/etc/passwd` file			
Action 2.8.2. Make `/dev/null` the shell for other non-root users in `/etc/passwd`			
Action 2.8.3. Create `/etc/ftpusers`			
Action 2.8.4 Remove `.rhosts` support from `/etc/pam.conf`			
Action 2.8.5. Create empty files to attempt to thwart remote attacks			
Action 2.8.6. Only root should be allowed to run the `crontab` and `at` commands			

CHECKLIST

STEP 2 OS MODIFICATIONS

Step 2.9: Statutory Warnings	Name of person Responsible	Date Completed	Initials
Action 2.9.1. Create `/etc/issue` and `/etc/motd`			
Action 2.9.2. Create an `/etc/default/telnetd` file			
Action 2.9.3. Create an `/etc/default/ftpd` file			
Action 2.9.4. Set appropriate access controls on all files			
Action 2.9.5. Set boot-level warning message			

Step 2.10: Sendmail	Name of person Responsible	Date Completed	Initials
Action 2.10.1. Optionally install the latest Open Source Sendmail			
Action 2.10.2. Use the minimal `/etc/mail/sendmail.cf` file shown in Appendix B			
Action 2.10.3. Add line to root's `crontab`			

Step 2.11: Miscellaneous	Name of person Responsible	Date Completed	Initials
Action 2.11.1. Turn on EEPROM security functionality			
Action 2.11.2. Edit `/etc/default/login`			
Action 2.11.3. Add line to root's `crontab`			
Action 2.11.4. Modify `/etc/default/inetinit`			
Action 2.11.5. Optionally enable password expiration			
Action 2.11.6. Modify `/etc/inittab`			

Comments:

CHECKLIST

STEP 2 OS MODIFICATIONS

Step 2.12: Fix-Modes Script	Name of person Responsible	Date Completed	Initials
Action 2.12.1. Obtain `fix-modes` software			
Action 2.12.2. Unpack sources			
Action 2.12.3. Build software on some other machine with a compiler			
Action 2.12.4. Move `fix-modes` distribution to machine being secured			
Action 2.12.5. Run `fix-modes` shell script from the command line			

STEP 3 INSTALLING OPENSSH WITH TCP WRAPPERS SOFTWARE

Step 3.1: Building and Installing the TCP Wrappers Software	Name of person Responsible	Date Completed	Initials
Action 3.1.1. Obtain TCP Wrappers source code			
Action 3.1.2. Unpack sources			
Action 3.1.3. Modify toplevel `Makefile`			
Action 3.1.4. Build software			
Action 3.1.5. Install resulting files in some useful directory			

Step 3.2: Building and Installing Zlib	Name of person Responsible	Date Completed	Initials
Action 3.2.1. Download Zlib software			
Action 3.2.2. Unpack software archive			
Action 3.2.3. Run the `configure` script, build the software, and install			

Comments:

CHECKLIST

STEP 3 INSTALLING OPENSSH WITH TCP WRAPPERS SOFTWARE

Step 3.3: Building and Installing OpenSSL		Name of person Responsible	Date Completed	Initials
Action 3.3.1.	Download the OpenSSL software			
Action 3.3.2.	Unpack the source archive			
Action 3.3.3.	Run the `config` script, build the software, and install			

Step 3.4: Building and Installing the OpenSSH Software		Name of person Responsible	Date Completed	Initials
Action 3.4.1.	Download source code			
Action 3.4.2.	Unpack sources			
Action 3.4.3.	Build software			
Action 3.4.4.	Install `sshd` and `ssh-keygen` binaries			

Step 3.5: Configuring TCP Wrappers and the SSH Daemon		Name of person Responsible	Date Completed	Initials
Action 3.5.1.	Create `/etc/hosts.allow` file			
Action 3.5.2.	Create `/etc/hosts.deny` file			
Action 3.5.3.	Create `/etc/sshd_config` file			
Action 3.5.4.	Set appropriate file permissions on configuration files			
Action 3.5.5.	Generate server key files			
Action 3.5.6.	Create `/etc/init.d/sshd` script			
Action 3.5.7.	Create link to sshd startup script in `/etc/rc2.d`			
Action 3.5.8.	Start SSH daemon			

Comments:

CHECKLIST

STEP 4 PUTTING THE SYSTEM INTO PRODUCTION

Step 4.1: Make a Backup	Name of person Responsible	Date Completed	Initials
Action 4.1.1. Boot the system in single-user mode			
Action 4.1.2. Mount all filesystems			
Action 4.1.3. Back up all ufs file systems to tape or other media TWICE			
Action 4.1.4. Write protect both tapes			
Action 4.1.5. Store one tape locally and the other off-site			
Action 4.1.6. Make sure both tapes are in physically secure locations which can only be accessed by trusted personnel			

Step 4.2: Physically Secure the Machine	Name of person Responsible	Date Completed	Initials
Action 4.2.1 Place the server in a locked room with access controlled by the administrator			
❑ Action 4.2.2. (Advanced) Provide electronic access control and recording for the server room			
Action 4.2.3. Provide temperature and humidity controls			
❑ Action 4.2.4. (Advanced) Provide one or more halon-type automatic fire extinguishers			
Action 4.2.5. Install a UPS and associated software			
❑ Action 4.2.6. (Advanced) Use surveillance cameras to record who accesses the equipment			
Action 4.2.7. Lock the CPU case and set up a system to ensure the key is protected and yet easily available to the administrator			
Action 4.2.8. Arrange the room so that the keyboard is hidden from prying eyes at windows or other vantage points			
❑ Action 4.2.9. (Advanced) Consider providing additional shielding against electronic interference or eavesdropping			

CHECKLIST

STEP 4 PUTTING THE SYSTEM INTO PRODUCTION

Step 4.3: Provide Adequate Network Security	Name of person Responsible	Date Completed	Initials
Action 4.3.1. Configure nearby routers to block spoofed packets			
Action 4.3.2. Stop smurfing and other denial-of-service type attacks			
Action 4.3.3. Only grant outside access to small list of services			

FINAL STEP BACKGROUND CHECKS

Final Step: Perform background checks on all persons with Administrator priviledges	Name of person Responsible	Date Completed	Initials

Comments:

CHECKLIST

Comments:

ABOUT THE SANS INSTITUTE

The SANS Institute is a cooperative research and education organization through which system administrators, security professionals, and network administrators share the lessons they are learning. It offers educational conferences and in-depth courses, cooperative research reports, and electronic digests of authoritative answers to current questions.

INFORMATION ON UPCOMING EVENTS

For the most current information on SANS Conferences and Events please see our website at: http://www.sans.org

For more information on these programs, email info@sans.org *or call* **(719) 599-4303.**

See the back cover for other resources from the SANS Institute.

Here are some reasons why security and audit professional and system and network administrators say SANS conferences are the ONLY ones they attend:

"One week at SANS provided me with a year's experience in system administration."

Scottie Swenson, SAIC and the University of Washington

"This conference provided the opportunity to learn from many of the people who are defining the future direction of information technology."

Larry Anderson, Computer Sciences Corp.

"The best aspect of SANS conferences is that they are tailored each year to what I, as an administrator, need to learn. SANS does an excellent job of keeping pace with current technologies, issues and trends."

John Mechalas, Intel

"I am impressed by the smorgasbord of technical sessions, course offerings, and BOF sessions. SANS truly wraps up the key issues facing SYS/NET administrators in ongoing future activities."

Robert Clay, GTE

ADDITIONAL OFFERINGS FROM THE SANS INSTITUTE

COOPERATIVE RESEARCH REPORTS AND PROJECTS

These documents present the results of a series of in-depth, consensus research programs aimed at identifying the most effective, proven approaches to meeting common challenges in security. Each booklet has been shaped by dozens of practitioners from large and small organizations who check and affirm its contents, based on their real-world, in-the-trenches experience. Among the participating organizations are: Merrill Lynch, Ballistic Missile Defense Organization, Andersen Consulting, MITRE, Exxon, Virginia Tech University, Naval Surface Warfare Center, Intel, Global Integrity, and KPMG Peat Marwick. Each is available in paper, as a 6-user license of a PDF, or as an unlimited site license PDF. The current consensus reports are listed to the right:

To order these publications go to www.sans.org, and click on the Bookstore.

ELECTRONIC DIGESTS

SANS NewsBites
A weekly email summary of the dozen most important news articles that have been published on computer security during the past week. Each entry includes brief highlights of the article and a url to allow you to read the whole story if it is still posted. To subscribe, send email to info@sans.org with the subject "subscribe SANS NewsBites."

The Security Alert Consensus
Published weekly and distributed via email, this consensus report provides two unique and important benefits: 1. absolute assurance that it covers all the important new vulnerabilities, and 2. personalization so you get only the material relevant to your systems. The Security Alert Consensus is a joint project of SANS, Network Computing Magazine, and Neohapsis. More than 100,000 people depend on it every week. To subscribe, send email to info@sans.org with the subject "subscribe Security Alert Consensus."

The Windows Security Digest
This digest provides updates to Windows Security: Step-by-Step, plus up-to-date guidance on new Hotfixes and Service Packs that should and should not be implemented. It also summarizes new threats and new bugs found in Windows and its services. To subscribe, send email to digest@sans.org with the subject "subscribe Windows Security Digest."

Windows NT Security: Step-by-Step
Now in its third edition, this booklet lists each of the security challenges that most Windows NT security professionals face and provides the detailed steps necessary, along with unique tips and caveats based on real-world experience, to close the holes or solve the problems. Among the 114 experts who helped make this booklet effective are Dr. Eugene Schultz and Stephen Northcutt.

Securing Linux: Step-by-Step
This publication outlines the specific steps required to tighten security, not only on RedHat's Linux, but also on the related services that are delivered with RedHat.

Computer Security Incident Handling: Step-by-Step
This fascinating guide reflects the experience of dozens of people who collectively have handled over 100 major computer incidents. Everyone should have this guide on hand, if only for its prescription for what to do if you haven't prepared in advance.

Intrusion Detection, Shadow Style: Step-by-Step
The author of the major book on intrusion detection, with the help of a team of America's most experienced intrusion detection analysts, gives you a step-by-step guide to implementing low-cost intrusion detection using proven tools.

The Annual SANS System, Network and Security Administrator Salary Survey
More than 7,000 people participated in the 2000 survey which provides the only authoritative salary information for security and system administration professionals. Tables reflect primary operating system (NT, UNIX, etc.), geographic location, experience, type and size of organization and much more. Also helps answer two management questions: "What does an employer have to do to keep talented security and system administration people?" and "What do those people need to do to get high raises?" (Available only to people who attend the SANS conferences)

THE SANS RESEARCH LIBRARY

Visit http://www.sans.org/infosecFAQ/index.htm for the largest collection of in-depth security research anywhere on the Internet. As many as 25 papers in each of 25 categories ranging from firewalls to intrusion detection, from auditing to policy.

Visit http://www.sans.org/newlook/resources/IDFAQ/ID_FAQ.htm for the definitive list of frequently asked questions (and answers) about intrusion detection.

And visit http://www.sans.org/newlook/resources/index.htm to see the twenty other security knowledge resources SANS members have compiled for the use of the entire community.

THE GLOBAL INCIDENT ANALYSIS CENTER

GIAC is the Global Incident Analysis Center established by the SANS Institute to monitor new attacks and provide immediate analysis and response. GIAC is one of the security community's primary sources of data on new attacks - the data is immediately fed into the GIAC education programs, and distributed to GIAC certified practitioners throughout the world. Please see http://www.sans.org/giac.htm for more information. The GIAC Training and Certification Program is designed to serve the people who are or will be responsible for managing and protecting important information systems and networks. GIAC course specifications were developed through a consensus process and combine the opinions, knowledge, and experience of many of the world's most experienced front-line security and system administrators, intrusion detection analysts, consultants, auditors, and managers. Please see http://www.sans.org/giactc.htm for more information

THE SANS ROADMAP TO NETWORK SECURITY WALL POSTER

Updated twice a year, these posters present top ten lists of answers to common questions: the best security books, the best security web sites, the biggest threats, the vendor contacts, the top tools and more. They are mailed automatically to all Security Alert and NewsBites subscribers and people who attend the Institute conferences.

SOLARIS STEP BY STEP SECURITY VERSION 2.0