Global Information Assurance Certification Paper

Hardening the IRIX Operating System

Michael Evanoff
Creative Technology, Inc.

January 16, 2005

This paper is being submitted to the SANS organization in order for the author to earn the Level Two GIAC certification in Securing Unix. It represents original work except where mentioned in the text. This paper describes steps for securing system running the IRIX operating system and was written for use on our own computers. System administrators will need to review all of the procedures and recommendations carefully in order to ensure that they are appropriate for use on their own systems.

Our system is using Origin servers made by Silicon Graphics, Inc (SGI). The servers will be running the IRIX operating system, which is SGI's version of the UNIX operating system. There are currently no formal procedures for hardening the IRIX operating system. Several people have documented procedures for securing the Solaris operating system. The Solaris operating system is a variant of UNIX System V, written by Sun Microsystems. While many of the security principles outlined in other documentation will apply in general to all UNIX System V operating systems such as IRIX, the details of implementing the procedures will vary according to the manufacturer of the different operating systems. The purpose of this supplement is to cover the IRIX operating system from Silicon Graphics. This supplement is intended for our system only and should be reviewed by each system administrator before implementing it on their systems.

Most hardening steps are accomplished by means of a shell script. The shell scripts are included as Appendix C to this document. While these scripts accomplish the majority of the hardening described in this paper, they can not accomplish all of it. For example, setting the PROM password was not done via a script, and neither was the setting of the root password. The scripts in Appendix C can be used for the hardening of other IRIX systems not related to our system. However, each system administrator should review the scripts against their own system needs to see if the scripts are accomplishing what the administrator intends.

1 Configuring Services

1.1 Disabling Unneeded System Services

There are many different services that are started when IRIX boots up. Thers services are started by individual startup files. The IRIX startup files are in the /etc/init.d directory. Many of the services that are started during the boot process are unnecessary and can cause security holes or can just waste CPU time and network bandwidth. Here is a list of the startup files that are created during a normal IRIX installation. The On/Off column recommends whether these files should be left in place so the services can be turned on, or deleted so that the services will be turned off. They include the following:

File Name	On/Off	Comments
aliases-ip	off	Sets up aliases for IP addresses
announce	on	Displays boot-up status
autoconfig	on	Configures the kernel
availmon	off	The availability monitor is a set of programs to collectively monitor and report the availability of a system and the diagnosis of system crashes.
chkdev	on	Checks for new devices in /dev and loads the modules
configmsg	off	Makes the manpage database and checks for files moved after software install. Turn off after all system software is installed.

cron	on	Starts the cron daemon
disk_patch	off	Loads firmware to disks newly added to the system. Turn off after all system hardware/software is installed.
dlif	off	Software to support diskless workstations
filesystems	on	Mounts and checks file systems
fontserver	on	Turns on the X11 font server
lp	on	Turns on printer spooling
mail	off	Turns on the sendmail program to forward mail
mediad	on	Automounts CDROM and Floppy drives
messagebus	on	Turns on interprocess communication
midi	off	Turns on MIDI sound/joystick support
network	on	Launches all of the network services. Need to edit /etc/inted.conf
netwr_client	off	Provides NetWare client support
ns_admin	off	Launches Netscape Server
ns_fasttrack	off	Launches Netscape Fasttrack Server
pcnfsd	off	Turns on support for NFS for PC clients
perf	off	Turns on some accounting functions
postinst	off	Removes files generated by messy installation programs. Turn off after all system hardware and software is installed.
rtmpfiles	on	Removes temporary files
run_proclaim	off	Sets up DHCP services
savecore	off	Causes computer to save a core dump if it reboots unexpectedly
swap	on	Configures the swap files
sysetup	on	Initializes some system logs
usr	on	Mounts the /usr filesystem
uucp	on	Deletes a few files. Launches no new processes.
videod	on?	Turn of if a DIVO/DIVO-DVC board is installed.
xdm	on	X display manager
xlv	on	Logical disk volume manager

In order to turn the services off, do not delete the actual file in the /etc/init.d directory. Instead, there will be a file in either the /etc/rc0.d or the /etc/rc2.d directory that is a symbolic link to the file in /etc/init.d directory. It is the file in the /etc/rc0.d or /etc/rc1.d that needs to be modified. The files in these two directories that begin with a "S" start new services. To disable service, rename the files in /etc/rc2.d to start with something other than an "S". For example, a file called /etc/rc2.d/S24savecore would be a link to the /etc/init.d/savecore file. By renaming the file to /etc/rc2.d/xS24savecore, the file would not be run as part of the boot process because it now starts with an "x" and not a "S". Renaming the files instead of deleting them allows the system administrator to add the services back in at a later date if it is necessary for the operation of the system.

The script used on the Origin servers to turn off these unnecessary services is included in Appendix C to this document.

Follow this procedure to disable additional selected system services:

1. As root, execute:
   chkconfig appletalk off
   chkconfig autoconfig_ipaddress off
   chkconfig gated off
   chkconfig mrouted off
   chkconfig named off
   chkconfig nds off
   chkconfig netwr_client off
   chkconfig noiconlogin off
   chkconfig nostickytmp off
   chkconfig nocleantmp off
   chkconfig nsd off

/etc/chkconfig nss_fasttrack off
/etc/chkconfig proclaim_server off
/etc/chkconfig proclaim_relayagent off
/etc/chkconfig proxymgr off
/etc/chkconfig quickpage off
/etc/chkconfig rarpd off
/etc/chkconfig rsvpd off
/etc/chkconfig rwho off
/etc/chkconfig vswap off
/etc/chkconfig webface off
/etc/chkconfig verbose on

1.2 Disabling Unneeded Network Services

The "inetd" service handles incoming network requests. For system security, it is necessary to turn off many of the different network protocols by which somone could get access to the system. Services such as finger, bootp and tftp may not be necessary for a particular server and should be turned off. Follow this procedure to disable selected *inetd* services:

1. Edit the file */etc/inetd.conf*, and add the **#** symbol at the beginning of the following lines to comment them out (some may have already been commented out):

```
exec         stream tcp     nowait root /usr/etc/rexecd         rexecd
bootp        dgram  udp     wait   root /usr/etc/bootp          bootp
rstatd/1-3   dgram  rpc/udp wait   root /usr/etc/rpc.rstatd     rstatd
walld/1      dgram  rpc/udp wait   root /usr/etc/rpc.rwalld     rwalld
rusersd/1    dgram  rpc/udp wait   root /usr/etc/rpc.rusersd    rusersd
rquotad/1    dgram  rpc/udp wait   root /usr/etc/rpc.rquotad    rquotad
bootparam/1 dgram  rpc/udp wait   root /usr/etc/rpc.bootparamd bootparam
ypupdated/1 stream rpc/tcp wait   root /usr/etc/rpc.ypupdated  ypupdated
rexd/1       stream rpc/tcp wait   root /usr/etc/rpc.rexd       rexd
```

In other words, they should look like this:

```
#exec         stream tcp     nowait root /usr/etc/rexecd         rexecd
#bootp        dgram  udp     wait   root /usr/etc/bootp          bootp
#rstatd/1-3   dgram  rpc/udp wait   root /usr/etc/rpc.rstatd     rstatd
#walld/1      dgram  rpc/udp wait   root /usr/etc/rpc.rwalld     rwalld
#rusersd/1    dgram  rpc/udp wait   root /usr/etc/rpc.rusersd    rusersd
#rquotad/1    dgram  rpc/udp wait   root /usr/etc/rpc.rquotad    rquotad
#bootparam/1 dgram  rpc/udp wait   root /usr/etc/rpc.bootparamd bootparam
#ypupdated/1 stream rpc/tcp wait   root /usr/etc/rpc.ypupdated  ypupdated
#rexd/1       stream rpc/tcp wait   root /usr/etc/rpc.rexd       rexd
```

If you want details on the services you are disabling, refer to their reference pages. For example, refer to rexecd(1M) for information on the remote execution server, or rexd(1M) for information on the RPC-based remote execution server.

2. Comment out or restrict the following entries in */etc/inetd.conf*:

```
shell  stream tcp nowait root  /usr/etc/rshd    rshd -L
login  stream tcp nowait root  /usr/etc/rlogind rlogind
tftp   dgram  udp wait   guest /usr/etc/tftpd   tftpd -s /usr/local/boot /usr/etc/boot
finger stream tcp nowait guest /usr/etc/fingerd fingerd -S
```

If you comment them out (totally disable them), they should look like this:

```
ftp    stream tcp nowait root  /usr/etc/ftpd    ftpd -l
telnet stream tcp nowait root  /usr/etc/telnetd telnetd
#shell  stream tcp nowait root  /usr/etc/rshd    rshd -L
#login  stream tcp nowait root  /usr/etc/rlogind rlogind
#tftp   dgram  udp wait   guest /usr/etc/tftpd   tftpd -s /usr/local/boot /usr/etc/boot
#finger stream tcp nowait guest /usr/etc/fingerd fingerd -S
```

To be safe, it is best to disable all those services with the comment character as shown above. (Doing so means, however, that the host can only be accessed from the local console.) Of these services, enabling *rshd* is probably the most dangerous, and *tftpd* is almost never required. Regarding *ftpd*, refer to *IRIX Admin: Networking and Mail*. If, however, you must include any of these services, change them as indicated below so that they record a log of their use in the file */var/adm/SYSLOG*:

```
ftp   stream tcp nowait root  /usr/etc/ftpd  ftpd -lll
shell stream tcp nowait root  /usr/etc/rshd  rshd -Lal
tftp  dgram  udp wait   guest /usr/etc/tftpd tftpd -s -l -h /dev/null
```

Note the logging options added to each daemon invocation. (For more information, refer to the reference page for any daemon you modify.)

3. When you have finished making changes to the */etc/inetd.conf* file, write the changes and exit from the editor. The changes take affect after a reboot. If you want to apply them immediately, enter:

```
# killall -HUP inetd
```

4. Test any modified services to be sure they perform as expected.

The script used to turn off the inetd services is included in Appendix C.

1.3 Disabling all EXCEPT root's cron jobs

From the IRIX man pages:

> Crontab(1) - If the file **/etc/cron.d/cron.allow** exists, only users whose names appear in the file are permitted to use *crontab*. This restriction applies to all users, including root. If that file does not exist, the file **/etc/cron.d/cron.deny** is checked to determine if the user should be denied access to *crontab*. If neither file exists, only root is allowed to submit a job. If **cron.allow** does not exist and **cron.deny** exists but is empty, global usage is permitted. The allow/deny files consist of one user name per line.

Accordingly, the file /etc/cron.d/cron.allow should be created with the users "root" and "sys" only.

Remove all files except "root" and "sys" from /var/spool/cron/crontabs

1.4 Disabling NFS file system sharing

If disabling NFS is desired, follow these steps:

Sysconf nfs off
Comment out the following NFS lines from the inetd.conf file

```
mountd/1,3    stream  rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
mountd/1,3   dgram   rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
sgi_mountd/1 stream rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
sgi_mountd/1 dgram  rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
```

2 Network Access Control

2.1 Turning off IP Forwarding

IRIX 6.5.9f and later:
1. As root, execute */usr/sysadm/privbin/configipforwardstate -n -off*
The -n is used to restart the network, so you must be on the system console when executing this command. You could also not use the -n , and then just reboot the system.

IRIX previous to 6.5.9f:
Follow this procedure to turn off automatic IP packet forwarding:
1. As root, edit the file */var/sysgen/master.d/bsd*, changing the value of *ipforwarding* to 0:

Change the line
```
int ipforwarding = 1;
```

to
```
int ipforwarding = 0;
```

2. Save the modified */var/sysgen/master.d/bsd* file and exit from the editor.

3. Run *autoconfig* with the **-f** option:
```
# autoconfig -f
```
This creates a */unix.install* file, which becomes the new */unix* after the system is rebooted.

4. Reboot your system (see reboot(1M)).

5. To verify that IP packet forwarding has been disabled after your system comes back up, use the *netstat* command:
```
# netstat -s -p ip | grep forwarding
```

You should see the following:

```
0 packets forwarded (forwarding disabled)
```

If you do not see this message, repeat steps 1 through 5 until you do. (Be sure that your root filesystem has enough disk space so that the */unix.install* file is being created correctly. See autoconfig(1M) for more information.)

2.2 Block broadcast packets

At this time, there has been no configuration option located to turn off or block broadcast packets on the IRIX system.

2.3 Stop the host from responding to broadcast packets

At this time, there has been no configuration option located to turn off or block broadcast packets on the IRIX system.

2.4 Install TCP Wrappers

A pre-compiled version of tcp_wrappers for IRIX is available from: http://toolbox.sgi.com/TasteOfDT/public/freeware/index-by-alpha.html. It is also included on the SGI Freeware CD's that come with the IRIX operating system. First, nstall TCP Wrappers using the IRIX "inst –f" command. Then run the "configure_tcp_wrappers.sh" script found in the appendix.

The source code is available at: ftp://ftp.porcupine.org/pub/security/index.html

2.5 WU-ftpd

Wuarchive-ftpd, also known as wu-ftpd, is a replacement ftp daemon for Unix systems developed at Washington University by Bryan D. O'Connor. Wu-ftpd is the most popular ftp daemon on the Internet, used on many anonymous ftp sites all around the world.

The latest compiled version of WU-ftpd for IRIX is available at http://toolbox.sgi.com/TasteOfDT/public/freeware/index-by-alpha.html, along with suggestions for configuring it.

WU-ftpd provides extensive functionality for web sites that service a heavy FTP load, and provides anonymous FTP upload and download services. Many systems will likely not have FTP needs that require as sophistocated a system as WU-ftpd provides. The standard IRIX ftp program along with the additional logging capabilities provided by tcp-wrappers should be sufficient for many systems. However, if more difficult FTP needs arise on IRIX system, the documentation that comes with WU-ftpd can be used to install and configure the software on IRIX.

2.6 S/Key

This program is available on IRIX if the need for it arises. The latest compiled version for IRIX is available at http://toolbox.sgi.com/TasteOfDT/public/freeware/index-by-alpha.html. It is also on the SGI freeware CD's that ship with IRIX

S/Key is a procedure for using one time passwords to authenticate access to computer systems. It uses 64 bits of information transformed by the `MD4` algorithm. The user supplies the 64 bits in the form of six English words that are generated by a secure computer. E.g. a pocket sized smart card or a PC/Macintosh, or a machine at work and printed on a sheet of paper. This six-word phrase is then used to answer a specific S/Key challenge. Example use of the S/key program **key**:

```
>key 99 th91334
Enter password:
OMEN US HORN OMIT BACK AHOY
>
```

Skey authentication is often used for internet logins, where passwords are transmitted via insecure means. Because skey uses one-time passwords the threat from passive attacks (snooping the network) is reduced. By default this package only installs the tools used to access an skey-protected system. If you wish to install S/Key authentication on a server you will need to take some additional steps:

1. Install `fw_skey.src.skey` and convince yourself that the privileged code is safe.
2. Install the non-default `fw_skey.sw.skey_server` (and `fw_skey.man.skey_server`) subsystems in this package. If having them suid still makes you uncomfortable you can create a special `skey` group, change **keyinit** and **keyauth** to be sgid (mode 2755) instead of suid, create `/etc/skeykeys` with mode 664, and finally "`chgrp skey`" on all three.
3. Edit `/etc/default/login` to specify **keyauth** as your `SITECHECK` program. Note that sitecheck programs must be executable, owned by `root`, and not writable by anyone else.
4. Optionally create `/etc/skey.access` to specify which networks are permitted to login using regular password authentication.
5. Setup local procedures to ensure that all users with login access to the protected machine have s/key passwords. (You may wish to replace **keyinit** with a script the does **rsh** to the server, and distribute that script to other machines.)

Note: this package is based on the original Bellcore version 1 source from 1994. `OPIE` is a more recent replacement for S/Key.

3 User Accounts

3.1 Password Settings

In order to require passwords for each account and to ensure that the root account can only log in directly at the console device, several changes should be made to the IRIX configuration files. Note that there is a script in Appendix C that performs most of the tasks described in this section.

Add the following lines to the file */etc/default/login*:

```
MANDPASS=YES
CONSOLE=/dev/console
```

Add the following lines to the */etc/default/passwd* file:

```
PASSLENGTH=8
```

From the IRIX login(1) man page:

> **login** reads **/etc/default/login** to determine default behavior. To change the defaults, the system administrator should edit this file. The syntax of the below lines within the **/etc/default/login** file **must not** contain any whitespaces. The examples shown below are login defaults. Recognized values are:
>
> **CONSOLE=*device*** If defined, only allows root logins on the device specified, typically **/dev/console**. This MUST NOT be defined as either **/dev/syscon** or **/dev/systty**. If undefined, root can log in on any device.
>
> **MANDPASS=NO** Like **PASSREQ**, but doesn't allow users with no password to log in.

From the IRIX passwd(1) man page:

> The behavior of the program is influenced by the content of */etc/default/passwd* if this file exists. The file is not supplied with the system, but may be locally created and modified as need be. If the file is not present, the default behaviors described below are followed. The following items are recognized:
>
> **PASSLENGTH=n**
> minimum length of an acceptable password. This defaults to 6, and has a maximum value of 8.

3.2 Unnecessary Accounts

The IRIX system creates several user accounts when it is initially installed. Some of these accounts are necessary and some are not. The unnecessary accounts need to be disabled, and the necessary accounts need to be protected. There is a script in Appendix C for accomplishing the bulk of this work. It is ultimately up to the system administrator, however, to check that all of the accounts are necessary and protected.

Guard access to all the special accounts as you would the *root* account. Either assign passwords to these accounts, or lock them using one of the methods described in "Locking Unused Logins". Following is a list of all the administrative and special accounts on the system and what they are used for:

root	This login has no restrictions, and it overrides all other logins, protections, and permissions. It

	allows you access to the entire operating system. The password for the *root* login should be very carefully protected.
sys	This login has the power of a normal user login over the files it owns, which are in */usr/sr*c. Its login should be disabled.
bin	This login has the power of a normal user login over the files it owns, which are throughout the system. Its login should be disabled.
adm	This login has the power of a normal user login over the files it owns, which are located in */var/ad*m. You may *su* to the adm login. This login should be disabled.
uucp	This login owns the object and spooled data files in */usr/lib/uucp* and */etc/uuc*p.
daemon	This login is the system daemon, which controls background processing. Its login should be disabled.
lp	This login owns the object and spooled data files in */var/spool/l*p. Its login should be disabled unless the system is a print server.
nuucp	This login is used by remote workstations to log into the system and initiate file transfers through */usr/lib/uucp/uucic*o.

Locking Unused Logins

Unused accounts can be locked using the "`passwd -l`" command. For example, if "`passwd -l jones`" is used, the password field entry in the jones account now looks like this:

```
jones:*LK*:3333:10:Jeremiah Jones:/usr/people/jones:/bin/tcsh
```

The second way to lock an account is by editing the password file directly. Change the password field to any string of characters that is not used by the password encryption program to create encrypted passwords. The *passwd* command with the **-l** option uses the string *LK*. You can use other strings to lock accounts. For example, you can use a descriptive phrase such as "LOCKED;" to remind you that the account was deliberately disabled:

```
ralph:LOCKED;:100:1:Ralph P. Cramden:/usr/people/ralph:
```

The semicolon is not used in an encrypted password and causes the account to be locked. The text "LOCKED" is merely to remind you that the account is locked. The following accounts in your default */etc/passwd* file are shipped without passwords. You should create passwords for at least the root account immediately. Several of these accounts should be deleted, such as the guest and demos accounts. Refer to the script in Appendix C for which accounts were deleted from the system.

- `root—Superuser`
- `lp—Print Spooler Owner`
- `nuucp—Remote UUCP User`
- `EZsetup—System Setup`
- `demos—Demonstration User`
- `OutOfBox—Out of Box Experience`
- `guest—Guest Account`
- `4Dgifts—4Dgifts Account`

Caution: Creating passwords on historically open accounts, such as *l*p, may cause certain related applications or operations to fail.

3.3 Assign an invalid shell to disabled accounts

Assign /bin/false to disabled accounts.

A locally compiled "noshell" shell can also be used. Here, */bin/noshell* is a simple C program that warns the user that they do not have access to a shell. Here is the source code for noshell.c:

```
#define Header "You do not have access to a shell.\n"

#define MSG "Please call Academic Computing at (813) 553-9551 if you
have
any questions\n"

void main()
{
        write(1,Header,strlen(Header));
        write(1,MSG,strlen(MSG));
        exit(0);
}
```

To compile the noshell.c application, do the following:

```
cc noshell.c -o /bin/noshell
chown root.daemon /bin/noshell
chmod 711 /bin/noshell
```

Please note that */bin/noshell* must be listed in */etc/shells*. Several flavors of UNIX do not have an */etc/shells* file by default. In those cases, you must create one, listing each of the valid shells available on your system.

A "commercial" quality noshell program is available at:
http://www.nas.nasa.gov/Research/Software/swdescription.html

3.4 Prevent ftp Access with disabled userids

FTP should be disabled on the interfaces that attach to the world or the Internet. If not, the file /etc/ftpusers will be modified in the same way that a Solaris file is modified by adding the following users:

- root
- sys
- bin
- adm
- uucp
- daemon
- lp
- nuucp
- smtp

From the Irix ftpd(1M) man page:

Ftpd authenticates users according to three rules.

1) The user name must be in the password data base, /*etc*/*passwd*, and not have a null password. In this case a password must be provided by the client before any file operations may be performed.
2) The user name must not appear in the file /*etc*/*ftpusers*. However, if the user name is in /*etc*/*ftpusers* followed by the white-space separated keyword ``restrict'', the user is allowed restricted access privileges, as described below.
3) If the user name is ``anonymous'' or ``ftp'', an anonymous ftp account must be present in the password file (user ``ftp''). In this case the user is allowed to log in by specifying any password (by convention this is given as the client user and host name).

3.5 Final Checks

3.6 Good Passworks

The general Solaris guidance is directly applicable to the IRIX operating system and should be used. Passwords should be a minimum of six characters in length and should include characters from several different classes (lower case, capitol, numbers, punctuation). Passwords should not be any word that would be found in a dictionary or a combination of two shorter words. There are a number of web sites and books containing guidance for what makes a good password. I will not attempt to duplicate this material here.

4 Server Security – Installation

4.1 Install a minimum Operating System Configuration

In order to install a minimum operating system configuration on IRIX, a list of the installable packages needs to be consulted, and the individual packages either installed or not installed. Appendix B contains a listing of the all of the packages available on the standard IRIX 6.5 installation CD's and a recommendation to either install or not install the package. This list should be considered a starting point and is subject to change if additional services are necessary.

4.2 Install the Recommended Patch Cluster

The latest patch clusters and security advisories can be found at http://www.sgi.com/support/security/index.html or at http://support.sgi.com/colls/patches/tools/browse/.

At the time of this writing, IRIX 6.5.9.m is the latest IRIX and there is only one patch necessary. IRIX is updated frequently and there are not usually a large number of patches needed if a recent version of the operating system is loaded. According to SGI:

> Unlike IRIX 5.3 through 6.4, there will be no patch sets for the IRIX 6.5 release. Instead, SGI will distribute maintenance fixes and new features for the IRIX 6.5 family using an installation overlay release mechanism. Like patches and patch sets, these are products that contain only the subset of files that have been changed or added since the major release. Unlike patches or patchsets, the files that are replaced are not saved during installation. Therefore, it is not possible to backout an intermediate release without reinstalling the major release of the product.

4.3 Set EEPROM Security Mode and Password

If you wish to set your PROM password from within the Command Monitor, perform the following steps:

1. Log in as root and shut the system down.
2. When you see this message, press the Esc key for the System Maintenance Menu:
`Starting up the system...`
To perform system maintenance instead, press Esc
3. Select option 5 from the System Maintenance Menu to enter the Command Monitor. You see the Command Monitor prompt:
`>>`
4. Type the passwd command and press Enter:
`passwd`
You see the prompt:
Enter new password:

5. Enter the password you want for your system and press Enter. You see the following prompt: Confirm new password:
6. Enter the password again, exactly as you typed it before. If you typed the password the same as the first time, you see the Command Monitor prompt again.Your password is now set. Whenever you access the Command Monitor, you will be required to enter this password.

Ref: IRIX (r) Admin: Backup, Security, and Accounting, Document Number 007-2862-004, Page 83

4.4 Force the use of SU to gain root access

You can restrict root logins to a single device, forcing root users to either use that device or use the su command (thereby leaving a trail in `/var/adm/sulog`). For example, edit `/etc/default/login` to include the following line to restrict root logins to the system console: CONSOLE=/dev/console

Note: Do not name `/dev/syscon` or `/dev/systty` as the device! These devices are the same as `/dev/console`, but login software does not treat them alike.

Ref: IRIX (r) Admin: Backup, Security, and Accounting, Document Number 007-2862-004, Page 94

4.5 Restrict Root's Search Path

The same procedure as outlined in the Solaris guide will be used. The PATH statements in all login files will be looked at.

4.6 Check Files Sourced When Root Logs In

The same procedure as outlined in the Solaris guide will be used.

4.7 Set Root's File Mask

The same procedure as outlined in the Solaris guide will be used. The .profile will be changed to include "umask 022"

4.8 Disable Trusted Host Support

Check to see if there are any `/etc/hosts.equiv` or `$HOME/.rhosts` files. These files can be configured to allow remote access without password protection, and should not be allowed on a firewall host. Refer to `hosts.equiv(4)` for more information.

4.9 Provide a Security Warning Banner

The file /etc/issue will be created according to the instructions for the Solaris system. The scripts in Appendix C accomplish this step.

4.10 Hide OS and Version from Remote Users

The "telnet –h" option will be added to the inetd.conf file to prevent IRIX from displaying the OS and version to incoming telnet connections. The scripts in Appendix C make this change.

5 System Logs

5.1 Restricting Access to Audit Logs

The log files in IRIX are:

/var/adm/sulog
/var/adm/utmpx
/var/adm/utmp
/var/adm/syslog

These will be set so only root has write permission. All other log files in /var/adm will be looked at for appropriate read/write permission.

5.2 Log All su Activity

From the Irix su(1M) man page:

su reads /*etc*/*default*/*su* to determine default behavior. To change the defaults, the system administrator should edit this file. Recognized values are:

SULOG=*file*	# Use *file* as the su log file.
CONSOLE=*device*	# Log successful attempts to su root to *device*.
SUPATH=*path*	# Use *path* as the PATH for root.
PATH=*path*	# Use *path* as the PATH for normal users.
SYSLOG=FAIL	# Log to syslog all failures (SYSLOG=FAIL)
	# or all successes and failures (SYSLOG=ALL).

All attempts to become another user using *su* are logged in the log file /*var*/*adm*/***sulog*** by default.

The following is from the IRIX man page loginlog(4), and it describes the procedure for logging failed attempts at logging in.

> After five unsuccessful login attempts, all the attempts are logged in the file **/var/adm/loginlog**. This file contains one record for each failed attempt. Each record contains the login name, tty specification, and time.
>
> This is an ASCII file. Each field within each entry is separated from the next by a colon. Each entry is separated from the next by a new-line.
>
> By default, **loginlog** does not exist, so no logging is done. To enable logging, the log file must be created with read and write permission for owner only. Owner must be **root** and group must be **sys**.

5.3 Log Incoming Connections for TCP Services

Our system will use tcp-wrapper to log incoming TCP connections. As mentioned in Section 2.4, it is available on the SGI home page. Download the program from SGI and install it using the "inst" program on IRIX. Once tcp-wrappers is installed, it needs to be configured. Refer to Appendix C for the script used to install tcp-wrappers on IRIX.

5.4 Install Tripwire to monitor changes to the system configuration

From the web site: http://toolbox.sgi.com/TasteOfDT/public/freeware/Installable/tripwire-1.2.html

tripwire-1.2: description + notes

tripwire checks file and directory integrity; it is a utility that compares a designated set of files and directories to information stored in a previously generated database. Any differences are flagged and logged, including added or deleted entries. When run against system files on a regular basis, **tripwire** lets you spot changes in critical system files and to immediately take appropriate damage control measures.

To use **tripwire** you will need to create a tw.config file describing what parts of the system to

monitor. You may find /usr/freeware/lib/tripwire/irix-tw.config a useful template, and the README file in that directory a useful guide. The default configuration file directory is /usr/adm/tcheck, and the database directory is /usr/adm/tcheck/databases.

Version 1.2 is the final freeware version; more recent versions can be purchased from Tripwire Security Systems.

After downloading, install tripwire using the "inst" command. Tripwire will install itself in the /usr/freeware directory. After installation, Tripwire needs to be configured. In Appendix C, there is a script for configuring the remaining parts of Tripwire on an IRIX system.

6 File Permissions

6.1 Limit Non-Root Access to System Files and Directories

A program called "fix-modes" was written for the Solaris operating system. Fix-modes restricts the permissions on specific system files to prohibit reading and writing of these files by users that do not need to be able to read and write to them. Unfortunately, this program is extremely Solaris-speciofic and not available for IRIX. The IRIX security manual has the following advice:

Be conservative when establishing or changing permission bit settings on all files and directories. The safest settings do not allow write access, but where this is not possible, it may be possible to limit write access to the owner of the file or directory, or at least just to the owner and the group.

The following files and directories are universally available for read and write access on IRIX as shipped. Depending on your site requirements, you may wish to change the permissions on these files to be more restrictive. See the chmod(1) reference page for a discussion on setting the sticky bit on such directories as */tmp* (this is the IRIX default) to restrict removal and renaming of files.

- */tmp*
- */usr/demos/.xsession*
- */usr/Insight/tmp*
- */usr/Insight/tmp/ebtpriv*
- */usr/Insight/tmp/ebtpub*
- */usr/Insight/tmp/install.insight.log*
- */usr/lib/emacs/maclib*
- */usr/lib/showcase/fonts*
- */usr/lib/showcase/images*
- */usr/lib/showcase/models*
- */usr/lib/showcase/templates*
- */usr/tmp.O*
- */var/spool/locks*
- */var/spool/uucppublic*
- */var/tmp*

Caution: Restricting permissions on historically open directories, such as */tmp*, */usr/tmp.O*, and */var/tmp* (linked to */usr/tmp*), can cause serious malfunctions in many programs, applications, and system utilities that write temporary files on behalf of users in these directories.

The system administrator can run the following command to find any directories on the system that are world read/writeable:

```
find / -local -type d -perm -007 -print 2> /dev/null
```

6.2 Remove sgid Permissions From System Files

The "find" command `find / -perm -2000 -print` will be used to locate the files with the setgid permissions. These files will be reviewed and modified if necessary to remove the setgid attribute. The remaining files will be located and documented. The list of setgid files and directories will be periodically reviewed to ensure that no new files have been added to the system.

The Tripwire program should be used to ensure that none of the sgid files are modified. Note the Tripwire configuration script in Appendix C that performs a "find" command to locate all of the sgid and suid files in order to monitor them.

6.3 Prohibit Execution of setuid Programs

As with Solaris, disks can be mounted using the "nosuid" option in the fstab file. This option should be turned on for all volumes other than / and /usr. A list of all setuid programs will be generated and reviewed using the procedure displayed below. Files that are not necessary will be removed. The remaining files will be located and documented. The list of setuid programs will be periodically reviewed to ensure that no new files have been added to the system.

The IRIX security manual has the following advice:

About Set-UID and Set-GID Permissions

The set user identification (set-UID) and set group identification (set-GID) permissions must be used very carefully. When a user runs an executable file that has either of these permissions, the system gives the user the permissions of the owner of the executable file. You can add these permissions to any executable file with the chmod(1) command. Set-UID and set-GID programs have legitimate uses, but because they are potentially harmful, there should be very few of them on your system. Beware of programs in publicly writable directories (such as */tmp*, */usr/tmp.O*, */var/tmp*, and */usr/spool/uucppublic*) that have the same name as common systems files (such as *vi* and *rm*). One reason the PATH environment variable of the *root* account does not include the current directory (as does the default PATH of most other users) is so that *root* won't accidentally execute such "booby-trap" programs.
System security can be compromised if a user copies another program onto a file with `-rwsrwxrwx` permissions. To take an extreme example, if the ***su*** command has the write access permission allowed for others, anyone can copy the shell onto it and get a password-free version of ***su***.
The following sections provide some example commands that identify files on the system with set-UID permissions. For more information about the set-UID and set-GID bits, see the chmod(1) and chmod(2) reference pages.

Checking for Set-UID Files Owned by root

The following command line lists all set-UID files owned specifically by *root*:

```
find / -user root -perm -4000 -print
```

The results of this command are printed on the screen. All paths are checked starting at /, including all mounted directories. A great number of files will be found. It is up to you to scan these files for any unusual names. One possibility is to direct the output of this program to a file soon after installation and compare the results with later outputs. If this command reports any unusual files, investigate them immediately.
A suspicious file might turn up like this:

```
-r-sr-xr-x 1 root  bin   38836 Aug 10 16:16 /usr/bin/at
-r-sr-xr-x 1 root  bin   19812 Aug 10 16:16 /usr/bin/crontab
-r-sr-xr-x 1 root  bin   27748 Aug 10 16:16 /usr/bin/shl
---s--x--x 1 root  sys   46040 Aug 10 15:18 /usr/bin/ct
-r-sr-sr-x 1 root  bin   33208 Aug 10 15:55 /usr/lib/lpadmin
-r-sr-sr-x 1 root  bin   38696 Aug 10 15:55 /usr/lib/lpsched
---s--x--- 1 root  user  45376 Aug 18 15:11 /usr/jbond/bin/sh
-r-sr-xr-x 1 root  sys   11416 Aug 11 01:26 /bin/mkdir
-r-sr-xr-x 1 root  sys   11804 Aug 11 01:26 /bin/rmdir
-r-sr-xr-x 1 root  bin   12524 Aug 11 01:27 /bin/df
-rwsr-xr-x 1 root  sys   21780 Aug 11 01:27 /bin/newgrp
-r-sr-sr-x 1 root  sys   23000 Aug 11 01:27 /bin/passwd
-r-sr-xr-x 1 root  sys   23824 Aug 11 01:27 /bin/su
```

In this example, the user *jbond* has a personal copy of */bin/sh* and has made it set-UID to *root*. This means that anyone in the group *user* can execute */usr/jbond/bin/sh* and become the superuser.

Checking for Set-UIDs in the root Filesystem

Most new systems will use IRIX's "XFS" file system. On systems using the "EFS" file system, there is extensive guidance in the "IRIX® Admin: Backup, Security, and Accounting: IRIX System Security" document on how to check for these files using the "ncheck" command. For XFS filesystems, use the *find* command:

```
find / -perm -4000 -print
```

Checking Set-UIDs in Filesystems Other Than root

Most new systems will use IRIX's "XFS" file system. On systems using the "EFS" file system, there is extensive guidance in the "IRIX® Admin: Backup, Security, and Accounting: IRIX System Security" document on how to check for these files using the "ncheck" command.

On XFS file systems, the "find" command can be used as shown above. Also, all disks should be mounted with the "nosuid" flag set. This includes locally mounted disks and NFS mounted disks.

7 Security Tools for IRIX

The following tools are available on the SGI Home page, pre-compiled for IRIX:

Tripwire
Tcp-wrappers
S/Key

Other programs such as COPS, portscan and udpscan can be compiled from the source provided by the Agency. Tripwire, tcp-wrappers and S/Key have been discussed already in this document, and there are scripts for configuring Tripwire and tcp-wrappers in Appendix C.

Appendix A References and links

The SGI web site contains a number of documents relating to IRIX hardening. In particular:

http://techpubs.sgi.com/library/tpl/cgi-bin/init.cgi contains a searchable database of all of the technical publications. A search on "IRIX Admin Backup Security Accounting" yields the manual "IRIX Admin: Backup, Security, and Accounting" which is referenced often in this paper.

Another document by Liam Forbes (lforbes@arsc.edu) Arctic Region Supercomputing Center contains general advice for UNIX systems in addition to considerable IRIX-specific information. The paper is located at http://www.arsc.edu/~lforbes/cug/HHPaper.html.

The security programs like S/Key, tcp wrappers and tripwire can all be found at http://toolbox.sgi.com/TasteOfDT/public/freeware/ . This site has precompiled packages that can be installed or uninstalled using the IRIX "inst" command.

Appendix B - A Minimum Installation of IRIX

This table shows all of the software packages that come on the IRIX 6.5 CD's and which packages will be installed initially as part of our secure system. These packages are subject to change, since 3rd party software may require additional IRIX packages to be installed in order for the software to run.

The list of packages was obtained at http://support.sgi.com/6.5/start_here/doc/cd.table.html

Package Name	Description	CD Name	Install?
accessx	Access for Movement-Impaired Users 1.0	APPLICATIONS	No
acrobat	Adobe Acrobat Reader 3.01	APPLICATIONS	No
annotator	IRIS Annotator 1.2	APPLICATIONS	No
appletalk	Xinet Macintosh Connectivity 9.02	APPLICATIONS	No
arraysvcs	Array Services 3.1	APPLICATIONS	No
cms_eoe	Color Management 2.0.5	APPLICATIONS	No
cosmocreate	Cosmo Create Authoring Environment 1.0.3	APPLICATIONS	No
cosmoplayer	Cosmo Player VRML Viewer 1.1 for IRIX 6.3 6.5	APPLICATIONS	No
custlink	CustomerLink Client Software 2.2.3	APPLICATIONS	No
demos	Demonstration Programs 6.5	APPLICATIONS	No
desktop_eoe	IRIX Interactive Desktop 6.5	APPLICATIONS	Yes
desktop_tools	Desktop Tools 6.5	APPLICATIONS	Yes
dps_eoe	Display PostScript/X 2.0.8 based on PostScript Level 2	APPLICATIONS	Yes
dynaweb	Dynaweb (InSight to HTML) 3.1a	APPLICATIONS	No
elec_svcs	Customer Support Services Base Software Rel. 2.2	APPLICATIONS	No
fddivis	FDDIVisualyzer 6.5	APPLICATIONS	No
gateway	Internet Gateway Execution Environment 3.1	APPLICATIONS	No
iforark	LicensePower/iFOR IS4.0.1 ARK	APPLICATIONS	No
iforcrk	LicensePower/iFOR IS4.0.1 CRK	APPLICATIONS	No
imgtools	ImageVision Tools 3.2.1	APPLICATIONS	No
impr_base	Impressario 2.2.5 Base	APPLICATIONS	No
impr_print	Impressario 2.2.5 Print Server	APPLICATIONS	No
impr_rip	Impressario 2.2.5 PostScript Renderer	APPLICATIONS	No
impr_rip_printers	Impressario 2.2.5 Host RIP Printers	APPLICATIONS	No

impr_scan	Impressario 2.2.5 Scanner Software	APPLICATIONS	No
infosearch	Information Searching Execution Environment 6.5	APPLICATIONS	No
InPerson	InPerson Desktop Conferencing 2.2.1	APPLICATIONS	No
insight	InSight Online Doc Viewer 3.1	APPLICATIONS	No
insight_gloss	IRIS InSight Online Glossary 3.0	APPLICATIONS	No
java_eoe	Java Execution Environment 3.1 (Sun JRE 1.1.5)	APPLICATIONS	No
license_eoe	License Tools 3.3	APPLICATIONS	Yes
macromedia	Macromedia Movie Player 1.4.1	APPLICATIONS	No
mbase_client	WebFORCE MediaBase 2.1 - Client	APPLICATIONS	No
media_warehouse	IRIX Interactive Desktop MediaWarehouse 1.2	APPLICATIONS	No
nedit	NEdit V4.0.3i - GUI style editor	APPLICATIONS	Yes
netscape	Netscape Communicator Client 4.05	APPLICATIONS	No
netscape_lite	Netscape Navigator Client 4.05	APPLICATIONS	No
netwr_client	NetWare Client 1.1	APPLICATIONS	No
ns_admin	Netscape Administration Server 2.13	APPLICATIONS	No
ns_fasttrack	Netscape Fasttrack Personal Server 2.01	APPLICATIONS	No
ocs_client	WebFORCE MediaBase 2.1 - OCS Client Execution Only Env	APPLICATIONS	No
outbox	OutBox Personal Web Site 1.5	APPLICATIONS	No
PeoplePages	PeoplePages - The Indigo Magic Phonebook 1.2.1	APPLICATIONS	No
print	Printing Tools Release 1.7.5	APPLICATIONS	No
Register	On-Line Registration 1.4	APPLICATIONS	No
sgips	Adobe Photoshop Performance Package 1.2	APPLICATIONS	No
sgsearch	Fulltext Indexing & Search Environment 2.0	APPLICATIONS	No
showcase	IRIS Showcase 3.4.2	APPLICATIONS	No
sitemgr	SiteMgr - Web Content Administration 1.1	APPLICATIONS	No
sysadmdesktop	IRIX Interactive Desktop System Administration 6.5	APPLICATIONS	Yes
sysmon	Desktop System Monitor 2.1.2	APPLICATIONS	Yes
vlan	VLAN software 1.0 for IRIX 6.5	APPLICATIONS	Yes
websetup	Web Setup and Administration 3.1	APPLICATIONS	No
webviewer	WebViewer library execution only environment 3.0	APPLICATIONS	No
xlators_3d	3D File Translators 1.1.1	APPLICATIONS	Yes
aso	Audio/Serial Option Card Serial Support 1.3	Audio Serial Option	No
aso_audio	Audio/Serial Option Card Audio Support 1.3	Audio Serial Option	No
tooltalk_dev	ToolTalk 1.3	Development Environment Tooltalk	Yes
c++_dev	C++ Headers and Libraries 7.2.1	DEVELOPMENT FOUNDATION	Yes
CaseVision	CASEVision Environment Version 2.6.5	DEVELOPMENT FOUNDATION	No
compiler_dev	Base Compiler Development Environment 7.2.1	DEVELOPMENT FOUNDATION	Yes
c_dev	C Headers and Libraries 7.2.1	DEVELOPMENT FOUNDATION	Yes
ftn77_dev	Fortran 77 Headers and Libraries 7.2.1	DEVELOPMENT FOUNDATION	No
ftn90_dev	Fortran 90 Headers and Libraries 7.2.1	DEVELOPMENT FOUNDATION	No
ftn_dev	Fortran Headers and Libraries 7.2.1	DEVELOPMENT FOUNDATION	No
langtools	Source Code Utilities 1.0	DEVELOPMENT FOUNDATION	Yes
modules	Modules package 2.2.1	DEVELOPMENT FOUNDATION	Yes
ProDev	ProDev WorkShop 2.6 Tutorial	DEVELOPMENT FOUNDATION	No
SpeedShop	Developer Magic SpeedShop 1.3	DEVELOPMENT FOUNDATION	No
WorkShop	Developer Magic WorkShop 2.6.5	DEVELOPMENT FOUNDATION	No
WorkShopMPF	WorkShop Pro MPF CASE products Version 2.8	DEVELOPMENT FOUNDATION	No

cms_dev	Color Management Software 2.0.5 Development	DEVELOPMENT LIBRARIES	No
complib_dev	CHALLENGEComplib 3.1.1	DEVELOPMENT LIBRARIES	No
complib_eoe	CHALLENGEComplib Execution Environment 3.1.1	DEVELOPMENT LIBRARIES	No
dev	Development System 7.2.1	DEVELOPMENT LIBRARIES	No
dmedia_dev	Digital Media Development Environment 6.5	DEVELOPMENT LIBRARIES	No
dvdr	Device Driver 4.1	DEVELOPMENT LIBRARIES	No
gl_dev	Graphics Library Development System 6.5	DEVELOPMENT LIBRARIES	No
ifl_dev	Image Format Library Development Environment 1.2.1	DEVELOPMENT LIBRARIES	No
impr_dev	Impressario 2.2.5 Developer's Kit	DEVELOPMENT LIBRARIES	No
inst_dev	Software Packager 1.5	DEVELOPMENT LIBRARIES	No
irix_dev	IRIX Development Examples 6.5	DEVELOPMENT LIBRARIES	No
ava_dev	Java Development Environment 3.1 (Sun JDK 1.1.5)	DEVELOPMENT LIBRARIES	No
license_dev	License Development Environment 3.3	DEVELOPMENT LIBRARIES	No
motif_books	OSF Motif developer books 1.2.3	DEVELOPMENT LIBRARIES	No
motif_dev	IRIX IM Development Software 6.5 (based on OSF/Motif 1.2.4)	DEVELOPMENT LIBRARIES	No
netscape_dev	Netscape Communicator Developer's Environment 4.05	DEVELOPMENT LIBRARIES	No
ViewKit_dev	ViewKit Development Environment Version 1.5.2	DEVELOPMENT LIBRARIES	No
webviewer_dev	WebViewer library development environment 3.0	DEVELOPMENT LIBRARIES	No
x_books	O'Reilly & Associates Inc. developer books X11R5	DEVELOPMENT LIBRARIES	No
x_dev	X11 Development Environment 3.8 based on X11R6.3	DEVELOPMENT LIBRARIES	No
divo	DIVO Video Execution Environment 1.1 for IRIX 6.5	DIVO Video Execution Environment	Yes
4Dwm	Desktop Window Manager 6.5	FOUNDATION 1	Yes
c++_eoe	Std. Exec. Environ. (C++ Headers & Libraries 7.2.1)	FOUNDATION 1	Yes
compiler_eoe	IRIX Standard Execution Environment	FOUNDATION 1	Yes
desktop_base	IRIX Interactive Desktop Base Software 6.5	FOUNDATION 1	Yes
dmedia_eoe	Digital Media Execution Environment 6.5	FOUNDATION 1	Yes
eoe	IRIX Execution Environment 6.5	FOUNDATION 1	Yes
ftn_eoe	Standard Execution Environment (Fortran Headers and Libraries 7.2.1)	FOUNDATION 1	Yes
insight_base	InSight Online Doc Viewer Base Software 3.1	FOUNDATION 1	Yes
io4prom	IO4prom for 64bit OS systems 6.5	FOUNDATION 1	Yes
ip32prom	Flash PROM for IP32 systems 6.5	FOUNDATION 1	Yes
motif_eoe	IRIX IM Execution Only Environment 6.5 (based on OSF/Motif 1.2.4)	FOUNDATION 1	Yes
tooltalk_eoe	ToolTalk 1.3 Execution Only Environment	FOUNDATION 1	Yes
ViewKit_eoe	ViewKit Execution Environment Version 1.5.2	FOUNDATION 1	Yes
websupport_eoe	WebSupport 1.3 eoe	FOUNDATION 1	Yes
x_eoe	X11 Execution Environment 3.8 based on X11R6.3	FOUNDATION 1	Yes
FDDIXPress	FDDIXPress 6.5	FOUNDATION 2	No
gnu	GNU Software configured & precompiled for IRIX 1.0	FOUNDATION 2	Yes
hwguides_eoe	Hardware Owner's Guides 1.1	FOUNDATION 2	Yes
ifl_eoe	Image Format Library Execution Only Environment 1.2.1	FOUNDATION 2	Yes
il_eoe	ImageVision Library Execution Only Environment 3.2.1	FOUNDATION 2	Yes
inventor_eoe	Inventor Execution Only Environment 2.1.4	FOUNDATION 2	No
isdn_eoe	ISDN Execution Environment 2.0	FOUNDATION 2	No
javascript_eoe	JavaScript library execution only environment 1.0	FOUNDATION 2	Yes
pcp_eoe	Performance Co-Pilot Execution Only Environment 2.0	FOUNDATION 2	No
performer_demo	Performer2.2.1 Demos and Demo Data	FOUNDATION 2	No
performer_eoe	Performer2.2.1 Execution Environment	FOUNDATION 2	No

sgitcl_eoe	SGI Tcl Execution Environment 1.1	FOUNDATION 2	Yes
vfc	Video Format Compiler 1.2	FOUNDATION 2	No
galileo	Galileo Video Execution Environment 6.5	Galileo Video Execution Env	No
impactcomp	IMPACT Compression Execution Environment 6.5	IMPACT Compression Execution Env	No
impactvideo	IMPACT Video Execution Environment 6.5	IMPACT Video Execution Env	No
Patch SG0003131	7.2.1 base compiler runtime environment for IRIX 6.2/6.3/6.4/6.5	INSTALLATION TOOLS	Yes
Patch SG0003139	libmp rollup	INSTALLATION TOOLS	Yes
Patch SG0003140	7.2.1 FORTRAN runtime for IRIX 6.2/6.3/6.4/6.5 - REQUIRED FOR F90	INSTALLATION TOOLS	Yes
roboinst	RoboInst Tools for Automatic Installations 1.0	INSTALLATION TOOLS	Yes
Welcome	Customer Welcome Jun. 98	INSTALLATION TOOLS	Yes
inventor_dev	Inventor 3D Development Toolkit 2.1.4	Inventor	No
inventor_games	Inventor Games Release 1.3	Inventor	No
demos_O2	O2 Demonstration Programs 1.2	O2 Demos	No
demos_octane	OCTANE Demonstration Programs 1.2	Octane Demos	No
evo	OCTANE Personal Video Execution Environment 6.5	OCTANE Personal Video Execution Env	No
nfs	Network File System 6.5	ONC3/NFS version 3	Yes
pcnfsd	PC-NFS 2.0.2 Server Components	ONC3/NFS version 3	No
OutOfBox	O2 OutOfBox Experience version 2.2 O2	Out Of Box Experience	No
react	REACT/PRO IRIX Realtime Extensions Release 3.2	React	No
windview	WindView system visualization for IRIX 1.2	React	No
sirius	Sirius Video Execution Environment 6.5	Sirius Video Execution Env	No
trix_eoe	Trusted IRIX Execution Environment 6.5	Trusted IRIX	No

Appendix C Hardening Scripts

This appendix contains a series of shell scripts for implementing the guidance contained in this document. These scripts are IRIX-specific and system specific. They need to be reviewed by each system administrator before they are run. The scripts are generally Bourne shell scripts and should run on any system. They are not presented in any specific order and should be independent of one another. The only exception to this is the "fix_inetd.sh" script which must NOT be run after the "configure_tcp_wrappers.sh" is run. Otherwise all of the tcp_wrappers will be disabled.

Note that these scripts should be run AFTER the operating system is up and running and after the EZsetup process has been completed. A root password needs to be established prior to running these scripts.

C.1 harden_irix.sh

```
#!/bin/sh
#
# harden_irix.sh
#
# This is the main menu program for performing the IRIX hardening. It
# does not do any hardening of its own. It just calls the other programs.
#
# Mike Evanoff
# Creative Technologies
# mevanoff@creative-tech.com

printmenu() {
```

```
cat<<EndOfMenu

Select one of the following procedures for hardening IRIX

      1) Remove unnecessary accounts and create a shadow password file
      2) Change the default login properties and restrict root login
      3) Configure inetd to turn off unnecessary network protocols
      4) Remove unnecessary programs from the startup directory
      5) Set permissions on log files and remove unnecessary user groups
      6) Remove non-root and sys cron jobs and restrict root and system ftp
      7) Remove rhosts and configure the root login environmental variables
      8) Fix permissions on files and remove some suid/sgid files
      9) Configure the tcp-wrappers program.
      10) Configure the Tripwire program

      0) Exit this Menu

EndOfMenu
}


while [ 1 -eq 1 ]
do

      clear
      printmenu
      echo -n "Enter your choice of options -> "
      read option

      case "$option" in
            1) ./fix_passwd.sh  ;;
            2) ./fix_logins.sh  ;;
            3) ./configure_inetd.sh ;;
            4) ./remove_startups.sh ;;
            5) ./fix_log_and_group_files.sh ;;
            6) ./configure_cron_and_ftp.sh ;;
            7) ./rhosts_and_root_env.sh ;;
            8) ./fix_file_permissions.sh ;;
            9) ./configure_tcp_wrappers.sh ;;
            10) ./configure_tripwire.sh ;;
            0) exit 0 ;;
            *) echo "Unknown option. Please select one of the valid options" ;;

      esac

      echo -n "\nHit Return to continue"
      read junk

done
```

C.2 fix_passwd.sh

```
#!/bin/sh
#
# This script removes unnecessary entries in the password file, and then
```

```
# generates a shadow password file. It also locks out the "lp" account
# so noone can log into the system using that account.
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

# Make a backup of the original password file
echo "backing up original file"
if [ ! -f /etc/passwd.original ]
then
    cp /etc/passwd /etc/passwd.original
fi

# Get rid of the unnecessary accounts
echo "removing old accounts"
cd /etc

/usr/sysadm/privbin/deleteUserAccount -R -l OutOfBox 2> /dev/null
/usr/sysadm/privbin/deleteUserAccount -R -l demos 2> /dev/null
/usr/sysadm/privbin/deleteUserAccount -R -l EZsetup 2> /dev/null
/usr/sysadm/privbin/deleteUserAccount -R -l guest 2> /dev/null
/usr/sysadm/privbin/deleteUserAccount -R -l 4Dgifts 2> /dev/null

echo "removing more old accounts"

egrep -v '(uucp|sgiweb|rfindd)' passwd > tempPasswd
mv tempPasswd passwd

# Lock out the lp account
passwd -l lp

# Create the shadow password file
echo "creating shadow file"

pwconv
```

C.3 fix_logins.sh

```
#!/bin/sh
#
# fix_logins.sh
#
# Set the default login behavior for the system
# Info on these settings can be found in "man login"
# and "man su"
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

touch /var/adm/loginlog
touch /var/adm/sulog

# Set up the /etc/default/login file to require root
# login at the console and to set some other parameters

cat<<EndOfLogin>/etc/default/login
```

```
# Set number of clock ticks per second (do not change!).
HZ=100
#
CONSOLE=/dev/console
PASSREQ=NO
ALTSHELL=YES
MANDPASS=YES
UMASK=022
##TIMEOUT=60
##SLEEPTIME=1
DISABLETIME=20
MAXTRYS=3
LOGFAILURES=3
##IDLEWEEKS=-1
##PATH=/usr/sbin:/usr/bsd:/sbin:/usr/bin:/usr/bin/X11:
SUPATH=/usr/sbin:/usr/bsd:/sbin:/usr/bin:/etc:/usr/etc:/usr/bin/X11
SYSLOG=ALL
INITGROUPS=YES
##SITECHECK=/some-authentication-program
LANG=C
SVR4_SIGNALS=NO
LOCKOUT=0
##LOCKOUTEXEMPT=oper1 niteop

EndOfLogin

# Set up the /etc/default/su to log all su actions

cat<<EndOfSU >/etc/default/su
SULOG=/var/adm/sulog
##CONSOLE=/dev/console
##SUPATH=/usr/sbin:/usr/bsd:/sbin:/usr/bin:/etc:/usr/etc:/usr/bin/X11
##PATH=/usr/sbin:/usr/bsd:/sbin:/usr/bin:/usr/bin/X11:
SYSLOG=ALL

EndOfSU

# Insert the login warning banner in the /etc/issue file

cat<<EndOfIssue >/etc/issue
This is a restricted access system. All activity on this system is subject
to monitoring. If the data collected during monitoring provide evidence of
criminal activity or esceeding priveleges, such evidence may be provided to
the autorities for use in prosecution, administrative, or other adverse
action. By continuing past this, whether you are an authorized user or not,
you expressly consent to this monitoring.

EndOfIssue


cat<<EndOfWarning
*****************************************

WARNING   WARNING   WARNING   WARNING   WARNING   WARNING   WARNING

   The default properties for login have been changed. If you are not
   careful, you may find yourself locked out of this system.
```

```
   Right now, root can ONLY log in via /dev/console. On the SGI Origin
   servers, the serial port on the back is NOT /dev/console. You may need
   to edit /etc/default/login to modify the line

     CONSOLE=/dev/console

    to reflect the actual port that you are logging in on. If in doubt,
    comment out this line and enter it back in later when you know the port.

DO NOT log out until you have done the following. This will ensure that there
is a user that is able to come in on a device other than the console and gain
root access in the event that the root account can not log in directly.

    1) Make sure that there is a user account other than root that can su to
       root.
    2) Double check that the user can telnet via ethernet and can su to root.
    3) Make sure that root has a password and that the non-root user has a
       password. Accounts that do not have passwords can not log in any more.

WARNING   WARNING   WARNING   WARNING   WARNING   WARNING   WARNING   WARNING
*******************************************************************************
EndOfWarning
```

C.4 configure_inetd.sh

```
#!/bin/sh
#
# configure_inetd.sh
#
# This script turns off many of the services that are available under
# the inetd daemon. The server may need to be completely rebooted
# before these changes fully take effect.
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

if [ ! -f /etc/inetd.conf.original ]
then
      cp /etc/inetd.conf /etc/inetd.conf.original
fi

cat<<EndOfInetd >/etc/inetd.conf
# Internet server configuration database
#
# After changing this file, tell inetd to reread it with the command
#     /etc/killall -HUP inetd
#
telnet      stream      tcp   nowait      root  /usr/etc/telnetd  telnetd -h
ftp   stream tcp nowait root  /usr/etc/ftpd  ftpd -lll
#shell stream tcp nowait root  /usr/etc/rshd  rshd -Lal
#tftp  dgram  udp wait   guest /usr/etc/tftpd tftpd -s -l -h /dev/null
#login      stream      tcp   nowait      root  /usr/etc/rlogind  rlogind
#ntalk      dgram udp   wait  root  /usr/etc/talkd          talkd
tcpmux      stream      tcp   nowait      root  internal
```

```
echo  stream      tcp   nowait      root  internal
discard     stream      tcp   nowait      root  internal
chargen     stream      tcp   nowait      root  internal
daytime     stream      tcp   nowait      root  internal
time  stream      tcp   nowait      root  internal
echo  dgram udp   wait  root  internal
discard     dgram udp   wait  root  internal
chargen     dgram udp   wait  root  internal
daytime     dgram udp   wait  root  internal
time  dgram udp   wait  root  internal
sgi-dgl stream    tcp   nowait      root/rcv    /usr/etc/dgld    dgld -IM -
tDGLTSOCKET
#
# RPC-based services
# These use the portmapper instead of /etc/services.
#mountd/1,3    stream  rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
#mountd/1,3   dgram   rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sgi_mountd/1 stream rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sgi_mountd/1 dgram  rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sprayd/1    dgram   rpc/udp wait    root    /usr/etc/rpc.sprayd     sprayd
#
# ToolTalk Database Server
ttdbserverd/1   stream  rpc/tcp wait root ?/usr/etc/rpc.ttdbserverd
rpc.ttdbserverd
#
# TCPMUX based services
#
# Impressario network scanning support
#tcpmux/sgi_scanner stream tcp nowait root   ?/usr/lib/scan/net/scannerd
scannerd
# Printer daemon for passing client requests to lpsched
#tcpmux/sgi_printer stream tcp nowait root   ?/usr/lib/print/printerd printerd

EndOfInetd

/etc/killall -HUP inetd

# Turn off some other services that are not needed

/etc/chkconfig autoconfig_ipaddress off
/etc/chkconfig gated off
/etc/chkconfig mrouted off
/etc/chkconfig named off
/etc/chkconfig nds off
/etc/chkconfig netwr_client off
/etc/chkconfig noiconlogin off
/etc/chkconfig nostickytmp off
/etc/chkconfig nocleantmp off
#/etc/chkconfig nsd off
/etc/chkconfig nss_fasttrack off
/etc/chkconfig proclaim_server off
/etc/chkconfig proclaim_relayagent off
/etc/chkconfig proxymgr off
/etc/chkconfig quickpage off
/etc/chkconfig rarpd off
/etc/chkconfig rsvpd off
/etc/chkconfig rwho off
```

```
/etc/chkconfig vswap off
/etc/chkconfig webface off
/etc/chkconfig verbose on
/etc/chkconfig nfs off

# Turn off ipforwarding on the server
#
#if [ ! -f /var/sysgen/master.c.original ]
#then
#    cp /var/sysgen/master.c /var/sysgen/master.c.original
#fi
#
#sed 's/ipforwarding =  0x1/ipforwarding =  0x0/' /var/sysgen/master.c.original
> /var/sysgen/master.c

# Turn off IP forwarding
/usr/sysadm/privbin/configipforwardstate -off

# Turn off the "Outbox Web Server"
/usr/sysadm/privbin/securesystem -w yes
```

C.5 remove_startups.sh

```
#!/bin/sh
#
# Shell script to turn off unneeded software at system startup.
# No files are deleted from the /etc/rc2.d directory. They are
# simply renamed by adding a lower-case x to the front of the filename.
# If the files do not start with an "S", they will not be executed during
# startup.
#
# System administrators need to review this list carefully to ensure that
# no critical software for their system is being shut down
#
# WARNING: This script will reboot the computer at the end.
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

mv  /etc/rc2.d/S32aliases-ip    /etc/rc2.d/xS32aliases-ip      # aliases-ip
mv  /etc/rc2.d/S99atalk         /etc/rc2.d/xS99atalk           # atalk
mv  /etc/rc2.d/S77atria         /etc/rc2.d/xS77atria           # atria
mv  /etc/rc2.d/S95availmon      /etc/rc2.d/xS95availmon        # availmon
mv  /etc/rc2.d/S88configmsg     /etc/rc2.d/xS88configmsg       # configmsg
mv  /etc/rc2.d/S00disk_patch    /etc/rc2.d/xS00disk_patch      # disk_patch
mv  /etc/rc2.d/S50mail          /etc/rc2.d/xS50mail            # mail
mv  /etc/rc2.d/S98midi          /etc/rc2.d/xS98midi            # midi
mv  /etc/rc2.d/S99netwr_client  /etc/rc2.d/xS99netwr_client    # netwr_client
mv  /etc/rc2.d/S96orbix         /etc/rc2.d/xS96orbix           # orbix
mv  /etc/rc2.d/S99pcnfsd        /etc/rc2.d/xS99pcnfsd          # pcnfsd
mv  /etc/rc2.d/S21perf          /etc/rc2.d/xS21perf            # perf
mv  /etc/rc2.d/S16postinst      /etc/rc2.d/xS16postinst        # postinst
mv  /etc/rc2.d/S75pppstartup    /etc/rc2.d/xS75pppstartup      # pppstartup
mv  /etc/rc2.d/S31proclaim      /etc/rc2.d/xS31proclaim        # run-proclaim
```

```
mv  /etc/rc2.d/S97proxymngr     /etc/rc2.d/xS97proxymngr       # proxymngr
mv  /etc/rc2.d/S90roboinst      /etc/rc2.d/xS90roboinst        # roboinst
mv  /etc/rc2.d/S98rtmond        /etc/rc2.d/xS98rtmond          # rtmond
mv  /etc/rc2.d/S48savecore      /etc/rc2.d/xS48savecore        # Savecore
mv  /etc/rc2.d/S98sdpd          /etc/rc2.d/xS98sdpd            # run-sdpd
mv  /etc/rc2.d/S34snmp          /etc/rc2.d/xS34snmp            # Snmp
mv  /etc/rc2.d/S99qpage         /etc/rc2.d/xS99qpage           # QuickPage
mv  /etc/rc2.d/S35webface       /etc/rc2.d/xS35webface         # Webface
mv  /etc/rc2.d/S60lp          /etc/rc2.d/xS60lp              # lp server
mv  /etc/rc2.d/S61bsdlpr            /etc/rc2.d/xS61bsdlpr          # lpr server




# Reboot the system to make the changes permanent
echo ""
echo "*******************************************"
echo ""
echo "You should now reboot your server to make sure"
echo "that there are no problems with your system"
echo "with all of the rc2.d files disabled"
echo ""
echo "*******************************************"
echo ""
```

C.6 fix_log_and_group_files.sh

```
#!/bin/sh
#
# fix_log_and_group_files.sh
#
# This script makes sure that log files exist and have the correct priveleges.
# It also strips unnecessary groups out of the group file.
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

# Make sure that the log files exist and have the correct owner
# group and permissions.
#
# Here is what it should look like when you are done.
#
# -rw-------   1 adm      adm         9706 Nov  9 15:13 sulog
# -rw-r--r--   1 root     sys          612 Nov  9 15:12 utmp
# -rw-r--r--   1 root     sys         6324 Nov  9 15:12 utmpx
# -rw-r--r--   1 adm      adm         3924 Nov  9 15:12 wtmp
# -rw-r--r--   1 adm      adm        40176 Nov  9 15:12 wtmpx

cd /var/adm
touch sulog
touch utmpx
touch utmp
touch syslog
```

```
chown adm  sulog wtmp wtmpx
chown root utmp utmpx

chgrp sys utmp utmpx
chgrp adm sulog wtmp wtmpx

chmod 600 sulog
chmod 644 wtmp wtmpx utmp utmpx

# From the Irix man pages:
#  By default, loginlog does not exist, so no logging is done.  To enable
#  logging, the log file must be created with read and write permission for
#  owner only.  Owner must be root and group must be sys

touch loginlog
chown root loginlog
chgrp sys loginlog
chmod 600 loginlog

# Remove mail, uucp, guest, demo and other unnecessary groups
# from the group file

cat<<EndOfGroup >/etc/group
sys::0:root,bin,sys,adm
root::0:root
daemon::1:root,daemon
bin::2:root,bin,daemon
adm::3:root,adm,daemon
user::20:
nobody:*:60001:
EndOfGroup

chown root /etc/group
chgrp sys /etc/group
chmod 644 /etc/group
```

C.7 configure_cron_and_ftp.sh

```
#!/bin/sh
#
# configure_cron_and_ftp.sh
#
# This scritp removes cron jobs that are not owned by root or sys.
# It also stops ftp access by root and other system accounts.
#
# Mike Evanoff
# Creative Technologies
# mevanoff@creative-tech.com

# Get rid of all cron jobs except those of root
if [ -f /etc/cron.d/cron.allow ]
then
      mv /etc/cron.d/cron.allow /etc/cron.d/cron.allow.original
fi
```

```
touch /etc/cron.d/cron.allow
chmod 600 cron.allow
echo root > /etc/cron.d/cron.allow
echo sys >> /etc/cron.d/cron.allow

# Get rid of all but the root and sys crontab files
cd /var/spool/cron/crontabs

echo "\n\nYou are about to remove crontab files other than root and sys"
echo "You will be asked to confirm each deletion. Proceed with caution."
echo "\nHit Return when ready"
read line
for file in `ls | egrep -v '^(root|sys)$' `
do
      rm -i $file
done

# Disallow ftp access by root and the system accounts

cat<<EndOfFTP >/etc/ftpusers
root
sys
bin
adm
uucp
daemon
lp
nuucp
smtp

EndOfFTP
```

C.8 fix_file_permissions.sh

```
#!/bin/sh
#
# fix_file_permissions
#
# Following the Solaris guidance for changing permissions on some of the system
# files, here is the list of files to be changed on the IRIX system.
# Many of the files that are suid or sgid on Solaris are not suid/sgid on
# IRIX and therefore do not need to be changed.
#
# Mike Evanoff, Creative Technologies
# mevanoff@creative-tech.com

chmd 500 /usr/bsd/rdist
chmod 400 /usr/bin/snoop
chmod 444 /etc/default/login
chmod 000 /usr/bin/at

# This is a list of all of the SUID programs in /usr /etc and /sbin
# on an IRIX 6.5 system. Only a few are being changed to remove the
# suid properties. More can be removed if necessary.
chmod u-s /usr/bin/at
```

```
#chmod u-s /usr/bin/lp
#chmod u-s /usr/bin/X11/cdheadphone
#chmod u-s /usr/bin/X11/xlock
#chmod u-s /usr/bin/X11/xterm
#chmod u-s /usr/bin/X11/xconsole
chmod u-s /usr/bin/mail
#chmod u-s /usr/bin/lpstat
#chmod u-s /usr/bin/cancel
#chmod u-s /usr/bin/passwd
#chmod u-s /usr/bin/newgrp
#chmod u-s /usr/bin/newproj
#chmod u-s /usr/bin/newsess
#chmod u-s /usr/bin/crontab
#chmod u-s /usr/bsd/rcp
#chmod u-s /usr/bsd/rsh
#chmod u-s /usr/bsd/ordist
#chmod u-s /usr/bsd/rlogin
#chmod u-s /usr/etc/ping
#chmod u-s /usr/etc/route
#chmod u-s /usr/etc/appletalk/psf
#chmod u-s /usr/etc/appletalk/xkas
#chmod u-s /usr/etc/appletalk/xkfs
#chmod u-s /usr/etc/appletalk/xktalk
#chmod u-s /usr/etc/mediad
#chmod u-s /usr/etc/traceroute
#chmod u-s /usr/etc/timedc
#chmod u-s /usr/gfx/setmon
#chmod u-s /usr/lib/iaf/scheme
#chmod u-s /usr/lib/SoftWindows/bin/SoftWindows95
#chmod u-s /usr/lib/SoftWindows/sys.swinconfig
#chmod u-s /usr/lib/print/netprint
#chmod u-s /usr/lib/print/chkicons
#chmod u-s /usr/lib/print/tagprinter
chmod u-s /usr/lib/sendmail
#chmod u-s /usr/lib/InPerson/inpview
#chmod u-s /usr/lib/regview
#chmod u-s /usr/lib/WorkShop/cvconnect
#chmod u-s /usr/lib/addnetpr
#chmod u-s /usr/sbin/cpr
#chmod u-s /usr/sbin/monpanel
#chmod u-s /usr/sbin/iwsh
#chmod u-s /usr/sbin/xwsh
#chmod u-s /usr/sbin/datman
#chmod u-s /usr/sbin/gmemusage
#chmod u-s /usr/sbin/passmgmt
#chmod u-s /usr/sbin/ssplay
#chmod u-s /usr/sbin/startmidi
#chmod u-s /usr/sbin/cview
#chmod u-s /usr/sbin/midisynth
#chmod u-s /usr/sbin/ksyncstat
#chmod u-s /usr/sbin/gr_osview
#chmod u-s /usr/sbin/dmrecord
#chmod u-s /usr/sbin/midikeys
#chmod u-s /usr/sbin/soundtrack
#chmod u-s /usr/sbin/Confidence/cdrom
#chmod u-s /usr/sbin/soundscheme
#chmod u-s /usr/sbin/cdplayer
```

```
#chmod u-s /usr/sbin/mkpts
#chmod u-s /usr/sbin/dmplay
#chmod u-s /usr/sbin/ksyncset
#chmod u-s /usr/sbin/scanners
#chmod u-s /usr/sbin/printers
#chmod u-s /usr/atria/sgi5/etc/db_loader
#chmod u-s /usr/atria/sgi5/etc/dumpers/db_dumper.53
#chmod u-s /usr/atria/sgi5/etc/dumpers/db_dumper.38
#chmod u-s /usr/samba/bin/swat
#chmod u-s /usr/sysadm/bin/rmprivuser
#chmod u-s /usr/sysadm/bin/addprivuser
#chmod u-s /usr/sysadm/bin/rmdefpriv
#chmod u-s /usr/sysadm/bin/addpriv
#chmod u-s /usr/sysadm/bin/adddefpriv
#chmod u-s /usr/sysadm/bin/checkpriv
#chmod u-s /usr/sysadm/bin/rmpriv
#chmod u-s /usr/sysadm/bin/runpriv
#chmod u-s /usr/local/bin/xscreensaver
#chmod u-s /sbin/df
#chmod u-s /sbin/su

# This is a list of all of the SGID programs on the same computer. Only a few
# are being changed to remove the sgid properties. More can be removed if
# necessary.

#chmod g-s /usr/bin/lp
#chmod g-s /usr/bin/X11/xload
chmod g-s /usr/bin/mail
#chmod g-s /usr/bin/passwd
#chmod g-s /usr/bsd/w
chmod g-s /usr/etc/nfsstat
chmod g-s /usr/etc/netstat
#chmod g-s /usr/lib/sa/sadc
#chmod g-s /usr/lib/print/chkicons
#chmod g-s /usr/lib/print/tagprinter
#chmod g-s /usr/lib/expreserve
chmod g-s /usr/sbin/Mail
chmod g-s /usr/sbin/ipcs
#chmod g-s /usr/sbin/osview
chmod g-s /usr/sbin/movemail
#chmod g-s /usr/sbin/bufview
chmod g-s /usr/sbin/mailx
#chmod g-s /usr/sysadm/bin/runpriv
#chmod g-s /sbin/ps
#chmod g-s /sbin/fuser

# Check mounted file systems to see that they have been mounted
# with nosuid

cd /etc
cat<<EndOfFstab

************************************************
File systems need to be mounted with "nosuid". Your
fstab file has been scanned and the following disks
have been mounted without fstab. Only the root and
/usr and swap file systems should be mounted without
```

```
the nosuid option.

You also need to check your /etc/auto_master file to
ensure that the -nosuid option is listed.

This script does not make any changes in these files
for you. You need to make the changes yourself
**********************************************

EndOfFstab

grep -v nosuid /etc/fstab

echo "\n\n*********************************************************\n"
echo "Here is a list of all of your directories that are world-writeable."
echo "Review the list and change the permission on as many as you can"
echo "while still maintaining system functionality"
echo "\n*********************************************************\n"

find / -local -type d -perm -007 -print 2> /dev/null | more
```

C.9 rhosts_and_root_env.sh

```
#!/bin/sh
#
# rhosts_and_root_env.sh
#
# Restrict the root search path, restrict the files "sourced" as part
# of the root login and set the root umask to 022
#
# Mike Evanoff, Creative Technologies Inc
# mevanoff@creative-tech.com

# Restrict Root's Search Path
# The same procedure as outlined in the Solaris guide will be used.
# The PATH statements in all login files will be looked at.

cd /

for file in .login .cshrc .profile
do
      rootpath=`grep "[       ]PATH[      ]" $file`
      if [ ! -z "$rootpath" ]
      then
            echo "\n\n*****************************************\n"
            echo "You have the PATH variable set in the file"
            echo "/$file"
            echo "The offending line is:"
            echo "$rootpath"
            echo "You need to remove this PATH statement. The root"
            echo "path is specified in /etc/default/login and should"
            echo "not be modified by local files this needs to be"
            echo "fixed"
            echo "\n*****************************************\n"
            echo "Hit any key to continue"
```

```
            read junk
      fi
done

# Check Files Sourced When Root Logs In

for file in .login .cshrc .profile
do
      sourced=`egrep '^[      ]*(\.|source)[    ]' $file`
      if [ ! -z "$sourced" ]
      then
            echo "\n\n*****************************************\n"
            echo "You have sourced a file in:"
            echo "/$file"
            echo "The offending line is:"
            echo "$sourced"
            echo "Make absolutely sure you want the root account to"
            echo "read this file as part of the startup process"
            echo "\n*****************************************\n"
            echo "Hit any key to continue"
            read junk
      fi
done

# Set Root's File Mask
# The .profile will be changed to include "umask 022"
cd /
echo "umask 022" >> .profile

# Disable Trusted Host Support
# Check to see if there are any /etc/hosts.equiv or $HOME/.rhosts files.
# These files can be configured to allow remote access without password
protection

echo "\n\nSearching for any .rhosts files on your local system....\n"
for file in `find / -local -name ".rhosts" -print 2>/dev/null`
do
      rm -i $file
done
```

C.10 configure_tcp_wrappers.sh

```
#!/bin/sh
#
# configure_tcp_wrappers.sh
#
# This program finishes the installation of the tcp_wrappers software on an
# IRIX system. The latest tcp_wrappers software can be downloaded from
# http://toolbox.sgi.com/TasteOfDT/public/freeware/index-by-alpha.html
#
# After the software is installed using the "inst" program, it ends up in
# /usr/freeware. This shell script moves the files to a more standard location
# and then configures the inetd.conf file to install the tcp wrappers
#
```

```
# There is a README and an IRIX.README file that comes with the software. Once
this
# script is finished running, there will also be a man page for tcpd and tcpdchk
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

if [ ! -f /usr/freeware/bin/tcpd ]
then
      cat<<EndOfMessage
************************************************************
ERROR

Can not find /usr/freeware/bin/tcpd. It looks like tcp-wrappers
has not been installed on you system yet. You need to install the
program before running this configuration script!

configure_tcp_wrappers is aborting

************************************************************
EndOfMessage
      exit 1
fi

cd /usr/freeware/

# Uncompress the man pages, then pack them. This
# is how Irix likes its man pages

cd catman
find . -name "*.Z" -exec uncompress {} \;
find . -type f -exec pack {} \;

# Put the man pages where they belong, then rebuild the man page
# directory
tar cf - . | (cd /usr/share/catman; tar xvf -)
/usr/lib/makewhatis&

# Move the binary files to /usr/etc where they belong
cd /usr/freeware/bin
tar cf - . | (cd /usr/etc; tar xvf -)

# Move the original daemons to a special subdirectory of the
# /usr/etc directory. This new directory is called "...". This
# directory is compiled into the tcpd binary

cd /usr/etc
mkdir ...
mv telnetd ftpd tftpd rshd rlogind talkd .../

# Configure the inetd.conf file. Because of Irix irregularities, each service
# needs to be commented out, then the inetd service restarted, then the
# service needs to be re-added in its modified form and the inetd service
# restarted again.
#
# This is a stripped-down inetd.conf file with many of the services commented
out.
```

```
# Each sysadmin will need to determine for themselves how best to configure the
# inetd services. The tcp_wrappers will be added to telnet, ftp, shell, tftp,
login
# and ntalk. Only the telnet and ftp services will be activated.

cat<<EndOfInetd1 >/etc/inetd.conf
# Internet server configuration database
#
# After changing this file, tell inetd to reread it with the command
#     /etc/killall -HUP inetd
#
#telnet     stream      tcp   nowait      root  /usr/etc/tcpd     telnetd -h
#ftp   stream tcp nowait root  /usr/etc/tcpd  ftpd -lll
#shell stream tcp nowait root  /usr/etc/tcpd  rshd -Lal
#tftp  dgram  udp wait   guest /usr/etc/tcpd tftpd -s -l -h /dev/null
#login      stream      tcp   nowait      root  /usr/etc/tcpd     rlogind
#ntalk      dgram udp   wait  root  /usr/etc/tcpd          talkd
tcpmux      stream      tcp   nowait      root  internal
echo  stream      tcp   nowait      root  internal
discard     stream      tcp   nowait      root  internal
chargen     stream      tcp   nowait      root  internal
daytime     stream      tcp   nowait      root  internal
time  stream      tcp   nowait      root  internal
echo  dgram udp   wait  root  internal
discard     dgram udp   wait  root  internal
chargen     dgram udp   wait  root  internal
daytime     dgram udp   wait  root  internal
time  dgram udp   wait  root  internal
sgi-dgl stream    tcp   nowait      root/rcv    /usr/etc/dgld     dgld -IM -
tDGLTSOCKET
#
# RPC-based services
# These use the portmapper instead of /etc/services.
#mountd/1,3    stream  rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
#mountd/1,3   dgram   rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sgi_mountd/1 stream rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sgi_mountd/1 dgram  rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sprayd/1    dgram   rpc/udp wait    root    /usr/etc/rpc.sprayd     sprayd
#
# ToolTalk Database Server
ttdbserverd/1   stream  rpc/tcp wait root ?/usr/etc/rpc.ttdbserverd
rpc.ttdbserverd
#
# TCPMUX based services
#
# Impressario network scanning support
#tcpmux/sgi_scanner stream tcp nowait root   ?/usr/lib/scan/net/scannerd
scannerd
# Printer daemon for passing client requests to lpsched
#tcpmux/sgi_printer stream tcp nowait root   ?/usr/lib/print/printerd printerd

EndOfInetd1

# Restart inetd for the first time
/etc/killall -HUP inetd
```

```
# Here is the actual inetd.conf that includes the tcp_wrappers "tcpd" daemon.
# Once this is written to the inetd.conf file, the inetd daemon will be
restarted
# to implement the changes

cat<<EndOfInetd2 >/etc/inetd.conf
# Internet server configuration database
#
# After changing this file, tell inetd to reread it with the command
#     /etc/killall -HUP inetd
#
telnet      stream      tcp   nowait      root  /usr/etc/tcpd     telnetd -h
ftp   stream tcp nowait root  /usr/etc/tcpd  ftpd
#shell stream tcp nowait root  /usr/etc/tcpd  rshd -Lal
#tftp  dgram  udp wait   guest /usr/etc/tcpd tftpd -s -l -h /dev/null
#login      stream      tcp   nowait      root  /usr/etc/tcpd     rlogind
#ntalk      dgram udp   wait  root  /usr/etc/tcpd          talkd
tcpmux      stream      tcp   nowait      root  internal
echo  stream      tcp   nowait      root  internal
discard     stream      tcp   nowait      root  internal
chargen     stream      tcp   nowait      root  internal
daytime     stream      tcp   nowait      root  internal
time  stream      tcp   nowait      root  internal
echo  dgram udp   wait  root  internal
discard     dgram udp   wait  root  internal
chargen     dgram udp   wait  root  internal
daytime     dgram udp   wait  root  internal
time  dgram udp   wait  root  internal
sgi-dgl stream    tcp   nowait      root/rcv    /usr/etc/dgld     dgld -IM -
tDGLTSOCKET
#
# RPC-based services
# These use the portmapper instead of /etc/services.
#mountd/1,3    stream  rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
#mountd/1,3   dgram   rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sgi_mountd/1 stream rpc/tcp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sgi_mountd/1 dgram  rpc/udp wait/lc    root    /usr/etc/rpc.mountd     mountd
#sprayd/1    dgram   rpc/udp wait    root    /usr/etc/rpc.sprayd     sprayd
#
# ToolTalk Database Server
ttdbserverd/1   stream  rpc/tcp wait root ?/usr/etc/rpc.ttdbserverd
rpc.ttdbserverd
#
# TCPMUX based services
#
# Impressario network scanning support
#tcpmux/sgi_scanner stream tcp nowait root   ?/usr/lib/scan/net/scannerd
scannerd
# Printer daemon for passing client requests to lpsched
#tcpmux/sgi_printer stream tcp nowait root   ?/usr/lib/print/printerd printerd

EndOfInetd2

/etc/killall -HUP inetd

# Test the installation of the tcp_wrappers program to make sure that everything
# went well
```

```
cd /usr/etc
./tcpdchk
```

C.11 configure_tripwire.sh

```
#!/bin/sh
#
# configure_tripwire.sh
#
# This script configures the Tripwire utility for use on an IRIX system. Run
# this script after the software has been installed with "inst -f tripwire". The
# files will be installed in /usr/freeware by the "inst" program.
#
# The Tripwire files were downloaded from
#   http://toolbox.sgi.com/TasteOfDT/public/freeware/index-by-alpha.html
#
# Some of the build parameters for the tripwire program supplied by SGI seem
# to be:
#    Database directory = /usr/adm/tcheck/databases
#    Configuration file = /usr/adm/tcheck/tw.config
#    Database file name = /usr/adm/tcheck/databases/tw.db_$HOSTNAME
#
# The list of files to be monitored was based on the generic irix-tw.config
# file that was included with the Tripwire installation package.
#
# Mike Evanoff, Creative Technologies Inc.
# mevanoff@creative-tech.com

if [ ! -f /usr/freeware/bin/tripwire ]
then
cat<<EndOfMessage
************************************************************
ERROR

Can not find /usr/freeware/bin/tripwire. It looks like tripwire
has not been installed on you system yet. You need to install the
program before running this configuration script!

configure_tripwire is aborting

************************************************************
EndOfMessage
exit 1
fi

# Uncompress the man pages, then pack them. This
# is how Irix likes its man pages. Put the man pages
# where they belong, then rebuild the man page directory

cd /usr/freeware/catman
find . -name "*.Z" -exec uncompress {} \;
find . -type f -exec pack {} \;
tar cf - . | (cd /usr/share/catman; tar xvf -)
/usr/lib/makewhatis&
```

```
# Create the tcheck directory if it does not exist

if [ ! -d /usr/adm/tcheck ]
then
      mkdir /usr/adm/tcheck
fi

# Build the configuration file. This is done in two parts. The first part
# is just a generic file that is applied to the standard set of system
# files. The second part finds all of the suid and sgid files and adds
# them to the configuration

CONFIG_FILE=/usr/adm/tcheck/tw.config

cat<<EndOfConfig > $CONFIG_FILE
#  First, root's "home"
=/          L
/.rhosts    R     # may not exist
/.profile   R     # may not exist
/.cshrc           R     # may not exist
/.login           R     # may not exist
/.exrc            R     # may not exist
/.logout    R     # may not exist
/.forward   R     # may not exist
/.netrc           R     # may not exist

# Unix itself
/unix       R

# Now, some critical directories and files
#  Some exceptions are noted further down
/etc        R
/etc/rc0.d  R
/etc/rc2.d  R
/etc/rc3.d  R
/etc/init.d R
/etc/config R
/etc/mtab   L
/etc/motd   L
/etc/rmtab  L
/etc/utmp   L
/etc/wtmp   L
/etc/OLDwtmp     L
/etc/xutmp  L
/etc/group  R     # changes should be infrequent
#  The next line may need to be replaced with /etc/security
#   if C2 is enabled
/etc/passwd L
/dev        L
/usr/etc    R

# Checksumming the following is not so critical.  However,
#  setuid/setgid files are special-cased further down.
/lib        R-2
/bin        R-2
/usr/bin    R-2
```

```
/usr/sbin   R-2
/usr/bsd    R-2
/usr/lib    R-2
/usr/adm    L
/usr/admin  R
/usr/bin/X11      R-2
=/usr       L
=/usr/spool L
/usr/spool/cron         L
/usr/spool/mqueue L
/usr/mail         L

# put entries for uucp if you need them
=/tmp       R-m
=/usr/tmp   R-m

EndOfConfig

# Find all of the SUID and SGID files on the system and add them to the
# configuration files

echo "Finding all SUID/SGID files. Be patient .... "
find /var /usr /etc /bin /sbin \( -perm -4000 -o -perm -2000 \) -print |\
  awk '{print $1, "\tR" }' >> $CONFIG_FILE

chmod 600 $CONFIG_FILE

# Build the database
echo "Building the database. Be patient again ..."
cd /usr/freeware/bin
./tripwire -initialize

# Create the directory to put the database into
if [ ! -d /usr/adm/tcheck/databases ]
then
      mkdir /usr/adm/tcheck/databases
fi
chmod 700 /usr/adm/tcheck/databases

# Move the database into the directory Tripwire expects to find it in

cd /usr/freeware/bin/databases
mv tw.db* /usr/adm/tcheck/databases/

# Create a shell script for running the tripwire utility to look for modified
# files. The output is written to /var/adm/tcheck.

cat<<EndOfCron >/usr/adm/tcheck/run_tripwire.sh
#!/bin/sh
TRIPWIREHOME=/usr/freeware/bin
TRIPCONFIGHOME=/var/adm/tcheck
CONFIGFILE=tw.config
LOGFILE=\`date | awk '{print \$2 "_" \$3 "_" \$6 "_tripwire.log"}' \`
HOST=\`/usr/bsd/hostname\`
\$TRIPWIREHOME/tripwire -c \$TRIPCONFIGHOME/\$CONFIGFILE -d
\$TRIPCONFIGHOME/databases/tw.db_\$HOST > \$TRIPCONFIGHOME/\$LOGFILE 2>&1
```

```
EndOfCron

chmod 700 /usr/adm/tcheck/run_tripwire.sh

# Put an entry into the root cron file to run tripwire every
# night and generate a log file. Restart the cron daemon

echo "0 2 * * * /usr/adm/tcheck/run_tripwire.sh" >>
/var/spool/cron/crontabs/root
kill -HUP `ps -ef | grep cron | awk '{print $2}'`
```

C.12 remove_software.pl

```
#!/usr/sbin/perl
#
# remove_software.pl
#
# This perl script is inteded to assist IRIX system administrators in
# removing unnecessary software. It runs the "versions" command to see what
# is installed and asks the administrator whether or not the software should
# be removed. A new file called remove_unnecessary_software.sh is created
# that contains all of the commands for removing the software. This new file
# needs to be run in order to do the actual software removal.\
#
# Mike Evanoff
# Creative Technologies
# mevanoff@creative-tech.com

open(IN, "versions -b|") || die;
open(OUT, ">remove_unnecessary_software.sh")|| die;

<IN>;<IN>;<IN>;<IN>; #Read the first three header lines

while(<IN>){

      chomp();
      @f = split(' ', $_);

      next if $_ !~ /^[I\s]/; #Skip lines that do not start with an I or a space

      next if $f[1] =~ /^eoe$/; # Dont remove the core operating system

      $desc = join(" ", @f[3 .. $#f]);

      print "Remove $desc (y/[n])? ";
      $response = <>;
      if ($response =~ /[yY]/ ){
            print OUT "echo \"Removing $desc\"\n";
            print OUT "versions -F remove $f[1]\n";
      }

}

print "
```

```
A new file has been created called \"remove_unnecessary_software.sh\".
In order to remove the software packages, you will need to run this
script. Please review the script carefully before you run it to make
sure that this is what you really want to do. Once you are satisfied,
do the following

 chmod +x remove_unnecessary_software.sh
 ./remove_unnecessary_software.sh

";
```