



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Securing Tru64 Unix
A Guide for the GIAC Agency

Bobbi Spitzberg
GCUX, Version 1.8
December 27, 2001

© SANS Institute 2000 - 2002, Author retains full rights.

Overview

Executive Summary

The plan for securing any host begins with determining how the host will be used. The primary use of the machine described in this paper will be as the Tru64 Unix host in a Security Laboratory. The purpose of the Security Lab is to test proof of concept for mitigation strategies for risk management in the infrastructure for the GIAC Agency, Department of Health and Human Services. It will also be used as a workstation for accessing a Major Application as defined by OMB Circular A-130, Appendix III.

The operating system used on the backend database servers in the production environment is Tru64 Unix. The version of Tru64 Unix chosen for the lab machine is 5.1A, the most recent version released. This paper will serve as a guide for securing the production and development servers as well as providing secure access to the both these environments.

The security decisions made in this guide are based on general Security Best Practices as taught in the SANS Institute Track 6 (Securing Unix), recommendations from Compaq Corporation for Tru64 Unix [see Appendix S], specific Agency and Department security policies, and OMB Circular A-130, Appendix III.

Using the HHS system designation process, the Major Application was designated a Security Level 3. Specific security requirements are assigned to each Security Level. These security requirements include limiting unsuccessful logins, locking out the terminal, physical security and the establishment of automated auditing. Please see Appendix R for the Matrix of Minimum Security Safeguards.

The SANS Institute/FBI Twenty Most Critical Internet Security Vulnerabilities are also addressed. There is commonality between these and the Federal Regulations for securing Automated Information Systems. Overlap occurs for password controls, contingency planning (backups), and written audit logs.

This guide will present detailed information that is specific to Tru64 Unix. Many guides exist for Solaris and Linux. This is the first comprehensive, step-by-step guide for securing Tru64 Unix that has submitted as a practical exam for the SANS Institute GIAC GCUX

certification. Therefore, emphasis will be given to the specifics for that operating system.

Hardware

The workstation being secured is a Compaq Alphaserver DS10 with the following hardware specifications:

physical memory = 256.00 megabytes.

COMPAQ AlphaStation XP900 617 MHz

DEC TULIP (10/100) Ethernet interface, hardware address 00-10-64-30-18-57

DEC TULIP (10/100) Ethernet interface, hardware address 00-10-64-30-18-56

2 9 gigabyte internal disk drives

1 12 G/24 G DAT tape drive

Physical Security

The host is located in an office. The access to the office area is restricted to employees with special access cards. Guards are posted at the entrances to the building and movement of equipment requires official property passes. The office does not have a lock. A move to larger, more secure area is anticipated early in the spring.

There is currently no UPS (Uninterruptible Power Supply) attached to the machine. The host is connected to a power surge protector.

The physical security will be increased when the office space has been renovated. We hope to set up a secure lab area at that time.

Risk Analysis

The key security concerns driving the decisions made in configuring this host relate to its role as a laboratory machine for a Level 3, Major Application and as a workstation authorized to access that application. We will, therefore, emphasize strong access controls, encrypted authentication, and good automatic audit logging. Another priority is limiting administrative access to only what is needed.

Since this machine has access through the application firewall to the production servers, a successful compromise of this machine gains

access to the production machines through the firewall. To reduce the risk of a compromise, the machine will be shutdown at the end of the day. Changing the root password or other compromises will be minimized because of the protection offered by securing the SRM console. Remote access will be severally limited by TCP wrappers. No remote root access will be allowed.

Because of its gateway function, this host needs to be very secure. We are planning on installing a hardware firewall in front of this machine similar to those protecting the production servers.

© SANS Institute 2000 - 2002, Author retains full rights

OS Installation – Preparation

Securing the SRM console

Before beginning the full installation, the host name and IP address were registered with the local DNS server. [Fictitious network addresses and domains will be used in this paper.] The hostname is sansproj.giac.dhhs.gov. The IP address is 172.29.126.142. The subnet mask is 255.255.254.0 and the gateway is 172.29.126.1. An Ethernet cable was connected to tu0, but not connected to the Internet. The machine is not to be connected to the Internet until it was hardened sufficiently to risk that exposure.

The next step is to secure the SRM console. As described in the Security Best Practices Evaluated Configuration paper cited in the References, there are two modes for the console – secure and user mode. User mode allows access to all SRM commands. Secure mode only allows access to the following commands – *start*, *continue*, *boot (only for the stored parameters)*, and *login*. In secure mode, you cannot change the boot default device or boot from the CD-ROM without first successfully logging in.

```
>>> set secure
```

```
Secure is not set. Please set the password.
```

```
>>> set password
```

```
Please enter the password:
```

```
Please enter the password again:
```

```
>>> login
```

```
Please enter the password:
```

```
>>>
```

The password must be between 15 and 30 alphanumeric characters.

Before we can boot from the OS distribution CD-ROM, we have to determine the address of the CD-ROM drive.

```
>>> show devices (If the password has not been yet been entered,  
Console is secure. Please login.
```

```
>>> login
```

```
Please enter the password:
```

```
>>> show devices
```

```
dka0.0.0.14.0      DKA0      (first internal disk)
```

```
dkb100.1.0.14.0  DKB100  (second internal disk)
...
dqb0.0.1.13.0    DQB0    the CD-ROM
...
```

Firmware installation

Place the firmware CD that was distributed with the Operating System CDs in the CD-ROM. A new version of firmware is always distributed with an OS release. The firmware should be upgraded before the OS is installed or upgraded.

```
>>> boot dqb0
```

The firmware update utility automatically determines the system type and model. The instructions on the screen should be followed. The firmware update should not be interrupted under any circumstances. Once it has completed the server is reset automatically. We are now ready to begin the installation of the OS.

SRM Boot Flag Settings

Before beginning the installation of the operating, certain console variables should be set. The machine must be halted in order to do this, if it isn't already.

```
>>> set boot_osflags ""      [Boot to single-user mode]
>>> set auto_action halt     [To return the machine to console prompt
after a system crash or power failure during installation. For a
production server, this would be set to "restart" after installation is
complete.] We will not change these settings after the installation is
complete for the lab machine.
```

Planning Disk Layout and AdvFS Domains

The installation for 5.x makes it easier to separate out /var from /usr, so we will select that option during the installation process. It is also a Best Practice to separate /tmp from the root domain and /var/tmp from /var. By placing these mount points in a separate domain, we are reducing the probability of host downtime because the root partition (if /tmp is left in /) or /var fills up. If /var fills up, the process that log to /var will stop, unless they have alternate places to log or other ways of handling this

full fileset situation. The layout (sizes approximate) for the 2 9G disks will be:

Mount point	Size	Partition	Domain/fileset
/	1G	/dev/disk/dsk0a	root_domain#root
swap RAM)	512MB	/dev/disk/dsk0b	swap (2 times)
/usr	2G	/dev/disk/dsk0g	usr_domain#usr
/var	2G	/dev/disk/dsk0h	var_domain#var
/usr/local	2G	/dev/disk/dsk0e	local_domain#local
/tmp	1G	/dev/disk/dsk0f	tmp_domain#tmp
/home	2G	/dev/disk/dsk1e	home_domain#home
/var/tmp	1G	/dev/disk/dsk1f	vartmp_domain#vartmp
/var/log	1G	/dev/disk/dsk1h	varlog_domain#varlog

The flexibility of AdvFS allows us to easily add or remove space to the file domains if needed. We have left sufficient space on the second disk drive to reallocate disk space for any partition that needs it.

Network Related Information

HOSTNAME	IP address/(mask, gateway)	Function
sansproj.giac.dhss.gov	172.29.126.142 (255.255.254.0,172.29.126.1)	The host being configured.
ns.giac.dhss.gov	172.29.128.251	Primary internal nameserver
ns2.giac.dhss.gov	172.29.64.1	Secondary nameserver
ntp1.giac.dhss.gov	172.29.157.45	Primary NTP server
ntp2.giac.dhss.gov	172.29.158.48	Secondary NTP server

Operating System Installation

A full installation will be done. The network interfaces have already been configured. We will select system subsets in addition to the mandatory subsets. These include C2 security support (OSFC2SEC520 and OSFXC2SEC520), OSFDCMTEXTxxx – to enable password triviality checks, OSFINCLUDE520 – system header files needed for the installation of security tools. These will be selectively installed after the installation of the mandatory subsets. Please see Appendix A for a list of the subsets installed. The mandatory subsets are in bold italics. Since we will be doing some development on this machine as part of the lab,

more subsets will be installed than there would be on a production server.

Place the Tru64 Unix version 5.1A Operating System, Volume 1 CD in the CD-ROM. (There is no volume 2.) At the chevron prompt (>>>), boot from the same device that was used for booting the firmware disk. (Login to the SRM console if you have not already done so.) The boot process can take several minutes. The time required depends on the hardware complexity of the system.

On systems with graphical displays, a window asking you to choose the language for the remainder of the installation will be displayed once the system has successfully rebooted. The choices are United States English, Chinese, or Japanese. After the language is selected, the Installation Welcome window is displayed. [See Appendix B]

If you prefer not to use the graphical user interface (gui), select the “Quit” option from the file menu. You will then exit from this user interface. You will be at a root prompt. The command to proceed with the command line interface is given below:

```
# restart nogui
```

We will assume that the gui is used for the installation. The process is the same independent of whether the gui or command line interface is used.

The next dialog box is the Host Information Dialog Box. [See Appendix C.]

The hostname can be 2 to 63 alphanumeric characters. Fully qualified hostnames can contain a maximum of 254 characters. They must begin with a letter. Consult with the site administrator to ensure that the hostname conforms to the site’s standards.

Proceed to set the time and date. The date is of the form *mm dd [cc]yy*. The time is entered as *hh mm* where *hh* is entered using the 24-hour-clock format. Use the pull down menus for setting area and location. This is what sets the time zone.

After clicking on “Next”, you will proceed to the Set Root Password Dialog Box. [Appendix D.] The root password must be between 6 and 16 characters and should contain a combination of upper and lower case letters. A minimum of one of the first characters must be a number, a special character, or an upper case letter. This password will be changed later as part of the Enhanced Security installation.

We will now be choosing the Operating System Subsets to be installed. [Appendix E contains the Software Selection Window.] It is important not to install more subsets than is needed. The number one vulnerability on the SANS Top Twenty list is the default installation of operating systems and applications. We do not want unnecessary services and ports available on the system for exploitation.

There are two possible approaches at this point. We could proceed with a custom installation. If the Customize option is selected, be sure to click on “Edit List” or you will get the Mandatory subsets without having the opportunity to select any optional subsets. Optional subsets included at this point would be Perl, AdvFS, LSM, and the Kernel Debugging Tools. Some of the X demos are quite useful. We have used `cpuinfo` in the past to very nicely show CPU utilization.

An alternate approach would be to do the mandatory installation and manually add and delete subsets after the installation of the mandatory subsets completes. Instructions for doing this will be presented later in this paper.

The next selection is to choose the options that are to be built into the kernel. Choose customize. [See Appendix G] These options will be presented after the system reboots following the completion of this selection process. The kernel option choices depend on the OS subsets that were chosen in the previous step.

The next step is to select the file system layout. [See Appendix H.] A recommendation for laying out the disks has already been discussed. Choose “Customize File System Layout.” This gives you the opportunity to configure the disks and file systems as you choose. Since we will need to relabel the disks, select “Edit Partitions from the “Custom File System Layout” dialog box. [Appendix I.] It is best if an experienced Tru64 System Administrator labels the disk. This can be done by selecting the Unix shell at the beginning of the installation and labeling the disk using the `disklabel` command. If this method is used then it is not necessary to edit the disk partitions. This is the recommended procedure for novice users. Experienced Tru64 Unix System Administrators will be able to edit the disk partitions as part of the Full OS Installation procedure.

Separate `/var` from `/usr` and select `dsk0` and the `h` partition for `/var`. Select `dsk0` and the `g` partition for `/usr`. Root is always on the `a` partition and we want it to be on `dsk0`. The file system type is AdvFS for all. After the Operating System has been installed, we will separate `/tmp`

from /, /var/tmp from /var, /var/log from /var, /usr/local from /usr and create a home domain for user home directories.

Swap is on the b partition on dsk0.

Click Next when the custom configuration is complete.

We now have the opportunity to check the choices we have made. When satisfied that the Summary Box reflects your choices accurately, select Finish. You may choose to redo any previous settings by selecting it on the Summary Box. [See Appendix J.]

Click OK in the Ready to Begin Installation Box. [See Appendix K.] The full installation procedure will then begin. File systems are created first. The software subsets will be loaded, followed by a reboot and the software configuration phase. This will take about a half hour for the system we are using.

Since we elected to customize kernel components, the kernel build will not proceed automatically as it would if the mandatory or full installation options had been selected. We have chosen to customize the kernel build in order not to install code beyond what is essential for this machine. A kernel option menu will appear after the system reboots.

Kernel Option Selection	
1	System V Devices
2	NTP V3 Kernel Phase Lock Loop (NTP_TIME)
3	Kernel Breakpoint Debugger (KDEBUG)
4	Packetfilter driver (PACKETFILTER)
5	IP-in-IP-Tunneling (IPTUNNEL)
6	IP Version 6 (IPv6)
7	Point-to-Point Protocol (PPP)
8	STREAMS pckt module (PCKT)
9	Data Link Bridge (DLPI V2.0 Service Class 1)
10	X/Open Transport Interface (XTISO, TIMOD, TIRDWR)
9	ISO 9660 Compact Disc File System (CDFS)
10	X/Open
11	DVDFS
12	CDFS
13	Audit
14	ACL Subsystems
15	None of the above
16	Help
17	Display all options again

Enter your choices.

Choices (for example, 1 2 4-6) [14]: 1-6, 12-14

After the kernel is rebuilt, the system automatically reboots. When we login as root, the System Setup Window will appear. We will exit from it at this time in order to do some preliminary housekeeping.

Post Installation customizations

Separating out /tmp, /var/tmp, /var/log, /usr/local, and user home directories

As discussed previously, we will place /tmp, /var/tmp, /var/log, /home, and /usr/local in their own domains (virtual volumes). Compaq recommends that /tmp and /var/tmp be separated from / and /usr. We will carry that recommendation further by separating them out into their own domains.

```
# mkfdmn /dev/disk/dsk0f tmp_domain
# mkfdmn /dev/disk/dsk1f vartmp_domain
# mkfdmn /dev/disk/dsk0e local_domain
# mkfdmn /dev/disk/dsk1e home_domain
# mkfdmn /dev/disk/dsk1h varlog_domain
```

```
# mkfset tmp_domain tmp
# mkfset vartmp_domain vartmp
# mkfset local_domain local
# mkfset home_domain home
# mkfset varlog_domain varlog
```

We now edit /etc/fstab.

```
# cp -p /etc/fstab /etc/fstab.orig
# vi /etc/fstab
[insert the following lines in the file after var_domain#var /var advfs
rw 0 2]
tmp_domain#tmp /tmp advfs rw 0 2
vartmp_domain#vartmp /var/tmp advfs rw 0 2
local_domain#local /usr/local advfs rw 0 2
home_domain#home /home advfs rw 0 2
```

```
varlog_domain#varlog /var/log advfs rw 0 2
```

Miscellaneous preparations

Save copies of some files before we change them.

```
#mkdir /etc/save.orig
```

Copy the listed files into that directory preserving permissions. For example,

```
# cp -p /etc/ftpusers /etc/save.orig
```

The files and directories we will save are: audit_events, ftpusers, motd, rc3.d, auth.system.default, ftpusers, passwd, securetty, auth.system.devassign, group, profile, /etc/shells, cron.deny, inetd.conf, rc2.d, and skel.login.

Before taking the system down to single user mode, verify that we will be prompted for root's password before we are actually allowed to be in single mode. This is a new security feature with 5.x.

```
# rcmgr get SECURE_CONSOLE [null response if not defined, otherwise YES]
# YES
```

If there was a null response,

```
# rcmgr set SECURE_CONSOLE YES
# rcmgr get SECURE_CONSOLE
# YES
# shutdown now
```

Enter root password at prompt. Verify that the new domains were mounted correctly.

```
# mount -a
# mount
```

We will now manually install the subsets needed for enhanced security and to build the tools we will be adding later. Place the Operating Systems Volume 1 in the CD-ROM.

```
# mount -r /dev/disk/cdrom0 /mnt
# cd ALPHA/BASE
# setld -l . OSFC2SEC520 OSXC2SEC520 OSFCDMTEXT520
OSFINCLUDE520
```

[Enhanced security, support for triviality checking, and standard headers for tools compilation]
Select all subsets.

We could elect to install other subsets at this time as well if we chose the mandatory subsets with the intent of manually installing subsets at this time. When the subsets have been installed, we reboot the system.

Customizing the setup

When the system reboots, select Custom setup from the System Setup window. The Custom Setup will then be displayed. [Appendix M.]

License Installation

Select License Manager. Delete the default USR license. Select “Edit” and “add”. Fill in the fields from the PAK on the license template displayed. We will enter the OSF-BASE, OSF-USR, OSF-SVR, ADVFS-UTILITIES and LSM licenses. Many of the features of both AdvFS and LSM that formerly required licenses have been incorporated in the base OS. However, many of the advanced features still require a license.

Network and NTP

We will setup the network even though we are still not connected to the network.

Both tu0 and tu1 are displayed when we start the network setup. Highlight tu0 and select “CONFIGURE”. Enter hostname, subnet mask, IP address and gateway. Select gated as the routing service, no DHCP, no rwho. Do not restart the network at this time. We have not yet connected the host to the Internet.

We will now configure the DNS client. We will be using the local nameservers for the GIAC agency. Our local domain is giac.dhhs.gov. Please enter the nameservers from the Network Related Information table we created earlier. Yes, we want the nameserver information in /etc/hosts. The hostname resolution order is localHostFile, DNS database,NIS. This corresponds to the following entry in /etc/svc.conf -- hosts=local,bind,yp. The search order of the domains is giac.dhhs.gov, dhhs.gov. Exit from this setup.

We will now set up NTP. It is important from a security perspective to have NTP configured. If there is a penetration of the system, it is important to have accurate timestamps on files. There are legal issues

associated with having accurate timestamps on system logs, particularly for coordinating log information from multiple systems.

Select NTP on the Custom Setup box. We will configure the machine as an NTP client. Select “add servers and peers V3.” Enter the NTP from the Network Related Information table. Select yes on the next screen. Since we are not currently on the network, don’t start the daemon now.

Since we are setting up the network, this might be a good time to open up another terminal window and tune the kernel with the recommended setting for avoiding SYN attacks. This recommendation comes from CERT® Advisory CA-2000-21 Denial-of-Service Vulnerabilities in TCP/IP Stacks. To lessen the probability of a successful SYN-ACK attack you essentially increase the size of the queue resources Tru64 UNIX will need for all connections, and since many of the SYN-ACK attacks don't form a complete connection, they get timed out and deleted. Setting the value of the parameters sominconn and somaxconn to 65535 will harden Tru64 UNIX against the SYN attacks. This change can be made using the following command:

```
# /sbin/sysconfig -r socket sominconn=65535  
# /sbin/sysconfig -r socket somaxconn=65535
```

Select Security from the Setup GUI so we can configure Enhanced Security. Select “enhanced”. Configure “Break-in detection and evasion options.” We want to log terminal logins, successful and unsuccessful login, maximum unsuccessful attempts will be set to 5. We will then elect to trim the authorization monthly without loss of data. Establish the cron job to run at 12:15 AM on the first of the month, i.e. 15 0 1 * *. Enable Access Control Lists (ACLs) and disable segment sharing. As recommended by Compaq we will ensure that segment sharing is disabled. Because of the way the page table is shared for shared libraries, normal file system permissions are not sufficient to protect against unauthorized reading. The text part of the library, not the data segment is shared. In order to avoid any unauthorized sharing, segment sharing must be disabled.

A new root password must now be set. The enhanced security will take effect at the next reboot of the system.

Account management

Hardening user accounts

Edit /usr/skel/.login, /etc/profile, /and /usr/skel/ .cshrc to make sure that the umask is 027. Add /usr/bin/Rsh (restricted shell) and /usr/bin/false to /etc/shells.

The GIAC agency security policy requires that password expire after 6 months. It also requires users to select strong passwords. Federal government security policy mandates further password controls such as limiting the number of unsuccessful attempts and protecting the secrecy of passwords. The SANS Institute lists poor or non-existent passwords as the second of the Top Twenty vulnerabilities.

We will use edauth to edit the trusted database files -- /etc/auth/system/default – the system default database and the two terminal control databases -- /etc/auth/system/ttys.db and /etc/auth/system/devassign.

To edit the system default database, enter the following command:
edauth -dd default.

We will be changing the following values in this file.

Password aging. The agency policy is that passwords expire after 6 months. We will also set a password lifetime of 9 months. This will give the user 3 months to reset the expired password after it has expired. After a total of 9 months, the user account expires. The depth of the password history file is set to 9. Time values given in seconds.

- u_exp=15638400
- u_life=23328000
- u_pwdepth#9

Password robustness. The agency guidelines recommend that passwords be at least 6 characters. The default for this system will be a minimum password length of 8 characters and a maximum length of 20 characters. We will enable triviality checks.

- u_minlen#8
- u_maxlen#20
- u_minchg#86400 [minimum time between password changes – 1 day]
- u_restrict [triviality checking]
- u_policy

Do not allow null passwords.

- u_nullpw@

Locking accounts. Accounts will be locked when they are initialing set up by the system administrator. The account will be unlocked when the system administrator has contacted the user and the user has been given the initial password for the account. Initial passwords are pre-expired. An account will be locked after 5 consecutive unsuccessful tries. Terminals will be locked after 10 consecutive unsuccessful attempts. The delay between unsuccessful login attempts will be set to 2 seconds as a defense against password cracking tools. If successful authentication is not completed within the login timeout interval, the login is aborted.

- u_maxtries#5
- u_lock [new accounts locked]
- t_maxtries#10
- t_logdelay#2
- t_login_timeout#300

Limiting Root Access

Other changes we will be making to the trusted databases include specific changes to limit access to the console as well as limiting root logins. Only root and the user *bobbi* will be allowed to login at the console.

```
# edauth -dv console
console:\
    :v_devs=/dev/console:v_type=terminal:v_users=root,bobbi:chkent:
```

```
# edauth -dv :0
\:0|\:0.0:\
    :v_devs=\:0,\:0.0:v_type=xdisplay:v_users= root,bobbi:chkent:
```

The secure terminal database file, `/etc/securettys`, controls root logins for all security levels. The default file only allows root logins at the console, from the X-display, and from `tty00`. We will, therefore, not change this file.

Giving the root account remote login capability is a huge vulnerability. We are also considering preventing root login at the console.

Removing unnecessary services

Many unnecessary services are started when the system boots and from `inetd`. One of the Top Twenty vulnerabilities is a large number of open ports. The more ports that are open, the more opportunities exist for people to find ways to connect to the system. So before connecting to the

Internet we need to remove unnecessary services from /etc/inetd.conf and prevent unneeded services from starting at boot.

All services except for cfmgr and suitjd can be removed from /etc/inetd.conf. We will edit /etc/inetd.conf to prevent services from starting from inetd. To do this, edit inetd's configuration file by place a number sign (#) as the first character in the line for everything except cfmgr (configuration management server) and suitjd which are used by the OS. Examples of configuration management requests are requests to configure, reconfigure, query, or unconfigure a subsystem. SysMan, the tool for managing the system, uses suitjd. Since we are not yet connected to the Internet, we do not have to HUP inetd.

We also need to prevent unnecessary services from starting from the start scripts in /sbin/rc?.d.

Disable all startup files for services that are not needed from /sbin/rc2.d and /sbin/rc3.d. The list below is partially drawn from the Tru64 Security Guide at geocities. We will prevent services from starting by changing the capital 'S' in the name of the script to a lowercase 's'. The following startup files should not be disabled:

In /sbin/rc2.d

S00savecore	S05paging	S10recpasswd	S35streams
S19security	S06mfsmount	S25enlogin	S20sia

In /sbin/rc3.d

S00inet	S00.50ip6host	S21audit	S59lsm
S30rmtmpfiles	S08startlmf	S25preserve	S45xntpd
S60mott	S90ws	S09syslog	S11route
S55inetd	S63write	S12gateway	S10binlog
S27sia	S57cron	S80crashdc	S22passwd

Also, rename the corresponding kill scripts in /sbin/rc0.d and /sbin/rc2.d.

Patching the Operating System

The machine is now secure enough that we can risk connection to the Internet to download any patches to the OS. Connect the Network Interface Card to the Internet port.

```
# shutdown -r now
```

When the machine reboots, verify that we are successfully online by doing a nslookup, netstat -nr, netstat -a, date, etc. commands.

```
# mkdir /usr/local/patch
# cd /usr/local/patch
```

Any patches for the version of the OS we are installing will be found at ftp1.support.compaq.com. The only patch currently available for Tru64 Unix is rpc.ttdserverd, a buffer overflow security patch. Even though we do not plan to run rpc services, we will still download and install this patch. A jumbo patch kit is expected to be released in January.

Change to the following directory public/unix/v5.1a.
mget t64v51assb*. This will download the patch, readme file and tarball.
Verify that checksum.

```
# more CHECKSUM
t64v51assb-c0000800-11707-er-20010928.tar 49656 1670
# sum t64v51assb-c0000800-11707-er-20010928.tar
49656 1670 t64v51assb-c0000800-11707-er-20010928.tar
```

The checksum validates the tarball was downloaded successfully.

```
# tar -xvf t64v51assb-c0000800-11707-er-20010928.tar
# shutdown now
```

Enter root's password to enter single user mode.

```
# mount -a
# cd /usr/local/patch/patch_kit
# ./dupatch
```

Follow the instructions from dupatch. It is not necessary to rebuild the kernel with this patch kit. After the patch installation is complete, reboot to multi-user mode.

Eliminating and restricting user accounts

Compaq recommends that the following accounts be locked – daemon, bin, sys, nobody. These can be locked from the dxaccounts gui or by using edauth.

```
# edauth daemon
```

Using the editor set up for root, insert `u_lock` into the entry for `daemon`, save the file and exit the editor. Repeat the process for the other accounts we will lock.

Add `/usr/bin/Rsh` and `/usr/bin/false` as valid shells by editing `/usr/shells` and inserting them in that file. We will then use `dxaccounts` through the System Setup Checklist to change the shells for `uucp` and `adm` to `/usr/bin/Rsh` (the restricted shell).

Adding user accounts

Previous to version 5.x, Compaq did not recommend using `useradd`, `usermod`, and `userdel` for managing user accounts with Enhanced Security. Only `dxaccounts` could be used. That is no longer true. We can now use `dxaccounts` or the 3 user commands or `sysman` account management to add the desired accounts on this host.

We will first add several groups for our users. We will be setting up 3 functional classes of users:

- System administrator – will have the ability to obtain root privilege through `su` initially and Compaq's Division of Privileges (`dop`). We will set up one account like this.
- General users – these accounts will be used to test our security policies for general users
- Restricted users – we will use these to test how we can restrict accounts before applying security policies on the development and production servers in our agency. Group and user names cannot exceed 8 characters.

Group name	Group id (gid)
admins	2000
testers	12000
restrict	20000

Before adding users via the GUI interface, let's add the groups we need from the above table.

```
# groupadd -g 2000 admins
# groupadd -g 12000 testers
# groupadd -g 20000 restrict
```

We can now set up the defaults for adding users. The following command establishes the default group, home directory and shell for new users. The second command displays the template that will be used for all new accounts.

```
# usermod -D -g testers -d /home -s /usr/bin/ksh [set defaults]
# usermod -D [display defaults]
Minimum User ID      = 12
Next User ID         = 1200
Maximum User ID      = 4294967293
Duplicate User ID    = 0
Use Hashed Database  = 0
Max Groups Per User  = 32
Base Home Directory  = /home
Administrative Lock   = 1
Primary Group        = testers
Skeleton Directory   = /usr/skel
Shell                = /usr/bin/ksh
Inactive Days        = 0
Expire Date          = Never
```

We will add the following accounts initially. Other accounts will be added later as needed.

User name	uid	Primary group	Shell	Home directory	User identification
bobbi	5497	admins	ksh	/home/bobbi	Bobbi
testusr1	5600	testers	ksh	/home/testers1	Jim
testusr2	5700	testers	ksh	/home/testusr2	Mike
ruser1	6100	restrict	Rsh	/home/ruser1	Sidney

```
# useradd -c Bobbi -u 5497 -g admins -m -p bobbi
# useradd -c "restricted user" -u 6100 -g restrict -m -p ruser1
.....
#
```

These commands will add the users and create the home directories. The default shell and the home directory structure have been established previously when we set the defaults. We will be prompted to enter a password (the -p flag) and to verify the password entered.

Miscellaneous security tightening

/etc/ftpusers

Even though we are not going to run the ftp daemon, we will set up the ftpd security file to reject remote logins to local user accounts. The

commands below can be combined into a script to run through root's crontab to keep this file current. The frequency needed to update the file depends on how often users are added.

```
# cp -p /etc/ftpusers /etc/save.orig/ftpusers
# rm /etc/ftpusers
# cat /etc/passwd | cut -f1 -d: > /etc/ftpusers
# chmod 600 /etc/ftpusers
```

Monitoring suid and sgid programs, hidden files

Set up a directory in /usr/local that will house baseline information for comparison.

```
# mkdir /usr/local/secsave
```

Create a baseline of setuid and setgid programs.

```
# /bin/find / -type f \( -perm -4000 -o -perm -2000 \) \
    -exec ls -ldb {} \; > /usr/local/secsave/setuid.list
```

Make a list of the hidden files on the system.

```
# /bin/find / \( -name '.*' ! -name . ! -name .. \) \
    -print > /usr/local/secsave/dotted.files
```

Eliminate .rhosts , /etc/hosts.equiv, .netrc

As recommended by the policies of the agency's CIO, we will eliminate /.rhosts, and /.netrc /. The reference is an internal document for agency use only.

```
# mkdir /.rhosts
# touch /.rhosts/x
# chmod 0 /.rhosts/x
# chmod 0 /.rhosts
```

Repeat the above for /.netrc and /etc/hosts.equiv and verify results.

```
# ls -ld /.rhosts
d----- 2 root    system    8192 Dec 17 22:05 /.rhosts
# ls -l /.rhosts
total 0
----- 1 root    system    0 Dec 17 22:05 x
# ls -ld /.netrc
```

```
d----- 2 root    system    8192 Dec 17 22:08 /.netrc
# ls -l /.netrc
total 0
----- 1 root    system      0 Dec 17 22:08 x
```

Secure libraries and message file

Libraries can be used as an attack. Compaq recommends disabling segment sharing, which we have already done when we configured Enhanced Security. Compaq and Securing Unix Track 6 notes also recommend verifying the ownership and permissions for libraries.

```
# ls -l /usr/shlib/*.so
# chown -h root /usr/shlib/*.so

# chmod 600 /var/adm/messages
```

Secure terminal ports

Terminal ports should only be readable by the owner. Modify the remote login shell file by using the code suggested by the Tru64 Unix Security manual.

The following should be added to /etc/profile file:

```
case "$TERM" in
none) ;;
*) /usr/bin/setacl -b '/usr/bin/tty' ;;
esac
```

Add this to the /etc/csh.login file:

```
if ($?TERM) then
if ("$TERM" != "none") then
/usr/bin/setacl -b '/usr/bin/tty'
endif
endif
```

Securing cron and at jobs

The following was suggested by the Tru64 Security Guide at www.geocities.com. It is important to note that the instructions in this guide are for version 4.x. In version 5.x, /usr/lib/cron is a CDSL.

To create the allow follows:

```
# echo "root" > /usr/lib/cron/cron.allow
# echo "bobbi" >> /usr/lib/cron/cron.allow
```

```
# chown root /usr/lib/cron/cron.allow [Suggested in the paper, but not
necessary for the system we are setting up since this are the permissions
already.]
```

```
# chmod 600 /usr/lib/cron/cron.allow
```

```
# cp -p /usr/lib/cron/cron.allow /usr/lib/cron/at.allow
```

To create the deny files

```
# cat /etc/passwd | cut -f1 -d: | grep -v root | grep -v bobbi \
>/usr/lib/cron/cron.deny
```

```
# chown root /usr/lib/cron/cron.deny
```

```
# chmod 600 /usr/lib/cron/cron.deny
```

```
# cp -p /usr/lib/cron/cron.deny /usr/lib/cron/at.deny
```

File access controls

As recommended by Hal Pomeranz in the [Securing Unix Track](#), we will modify the way /usr is mounted. We do not want any files in the /usr partition comprised. We will change the way /usr is mounted to ro. This will require a reboot to take effect.

Edit /etc/fstab and change the entries for /usr and /home to the following:

```
usr_domain#usr /usr advfs ro 0 2
```

Warning Banners

It is strongly recommended that a warning banner be included on all government computers. The sample warning banner is included at the end of this paper in Appendix Q.

Create an /etc/issue file containing the sample warning banner using your favorite editor. We will also replace the original /etc/motd since it contains the OS version.

```
# cp -p /etc/motd /etc/motd.orig
```

Create a /etc/motd.new with a welcome message of your choosing.

```
# cat /etc/issue /etc/motd.new > /etc/motd
```

Auditing

We will set up the Basic Audit Configuration. As we become more familiar with the auditing subsystem and managing the disk space that it uses, we will augment this initial setup.

The tasks involved in setting up auditing involve:

- Defining the configuration
- Determining what is audited
- Generating audit reports
- Manage the disk space used by the audit logs
- Archiving the audit logs

The following table maps system commands (root privileges required) to tasks. The information in the table has been obtained from the man pages for the commands as well as the Tru64 Unix Security manual.

auditconfig	Defines the configuration used for auditing
auditmask	Gets and sets audit masks
audgen	Generates an audit record and places it in the audit log
auditd	Audit daemon. Will display information about the auditing configuration. Also administers audit subsystem data storage. Configures the audit subsystem.
audit_tool	Presents information from the binary audit log in a readable format.

The security subsets OSFC2SEC520 and OSFXC2SEC520 (for X support) must be configured in the kernel. We have already included them for our system. We will setup auditing from the command line without using a GUI tool.

```
# sysman auditconfig
...
                                Audit Configuration on sansproj.giac.dhhs.gov :
Welcome
Welcome to the audit configuration utility.
PLEASE NOTE:
- This utility erases previous audit subsystem configurations.
- For hosts in a cluster, modifications made with this utility
are clusterwide.
Do you wish to configure the audit subsystem? Yes

(The tool will then go on to explain that the audit subsystem is
configured in 2 parts. The first part configures the audit logs –
location, the action to be taken if the logs fill up, how long you
want the logs on the system. The second part involves the
selection of events to be audited.)
```

Select [OK]

Select the default location of `/var/audit/auditlog`.

There are 5 possible actions if the audit logs fill up -- suspend auditing until space becomes available, change audit data location according to `/etc/sec/auditd_loc`, overwrite the current log, terminate auditing, or halt the system.

Select changing the audit data local by using the arrow keys and pressing the enter key when that choice is highlighted. Tab to [Next] and then enter.

We are then given a choice on how long to keep the audit logs. We will be monitoring their growth and write our own scripts to back them up and remove old logs as our experience grows. We therefore select “forever” as the amount of time to keep them. Tab to [Next] and hit enter.

The remaining choices for part one of the setup involve choosing where audit information will be logged. We again select the default – syslog and also indicate that remote systems cannot log to this host.

We will proceed to part 2 to select the events to be audited.

We select a standard auditing profile for this initial auditing configuration of our lab machine. As recommended by Compaq, we will select to audit trusted events. Tab to [Next] and hit enter. A list of the events to be audited for the profile we selected then appear. This seems like a good list for us to use initially. We therefore proceed to the next selection which lists the files to be audited. We proceed to the next selection. We have finished setting up auditing through the `auditconfig` tool. Please see Appendix N for the complete list.

We now edit `/etc/sec/auditd_loc` in order to specify the locations for files if the audit log space fills up. We set up this secondary location in `/var/log` since this is in a separate domain.

```
# mkdir /var/log/audit
# chmod 700 /var/log/audit
# ls -ld /var/log/audit
drwx----- 2 root  system   8192 Dec 23 18:35 /var/log/audit
```

Edit /etc/sec/auditd_loc to include this path.

To verify the audit setup we have just completed –

```
# auditd -w
```

Audit data and msgs:

```
-l) audit data destination          = /var/audit/auditlog.sansproj.001
-c) audit console messages         = syslog
```

Network:

```
-s) network audit server status (toggle) = off
-t) connection timeout value (sec)      = 4
```

Overflow control:

```
-f) % free space before overflow condition = 10
-o) action to take on overflow           = change to next auditlog location
```

We will run our initial reports on login data. As our experience with the auditing subsystem grows, we can add and/or delete the events audited and the files audited. We might also elect to set up /var/audit in its own fileset with a soft and/or hard limit set or in its own domain to protect other processes that write to /var.

We might also initially just audit trusted events. This is a recommendation from Compaq.

We plan to compress audit logs regularly and copy the compressed logs to tape. After determining that the tape backups of the audit logs are indeed valid, the backed up logs will be removed from the system. These tapes will be kept in a fireproof safe in a secure area.

We will run the script [Appendix P] suggested by Compaq to extract information from the audit log daily from cron.

```
# crontab -e
insert the following line into root's crontab
0 4 * * * /var/adm/local/login.extract
```

save the crontab and quit out of the editor.

Sendmail

Sendmail historically has had a large number of vulnerabilities. Version 8.9.3 and above are more secure and more stable. However, since this machine is not a mail relay, there is no need for it to run sendmail as a

daemon (-bd) We will only run sendmail periodically to empty the mail queue.

To find the version of sendmail that is shipped with v5.1A.

```
# echo \$(Z | /usr/sbin/sendmail -bt -d0  
Version 8.9.3  
....
```

Although this version is usable from a security perspective, we will rebuild sendmail from more recent source. Sendmail can be found at ftp.sendmail.org. It is located in the directory /pub/sendmail. Download [sendmail.8.11.6.gz](http://ftp.sendmail.org/pub/sendmail.8.11.6.gz). After you have built sendmail, change the configuration so that you can run it as a null client. The OSTYPE will be osf1 and the FEATURE ('nullclient', mailhub'). Since we have disabled sendmail from running automatically, we will need to set up a job in crontab to run to empty the mail queue periodically. We will empty the queue every hour at 15 past the hour. The crontab entry would therefore be:

```
15 * * * * /usr/lib/sendmail -q.
```

Installation of important security tools

We will be installing tcp_wrappers and OpenSSH (and zlib and OpenSSL – needed for the ssh installation.) We will only allow wrapped ssh connections to this host. We will connect to the Internet briefly to download these tools. Compaq provides compiled code for both ssh and TCP wrappers. We choose not to use these because we want to do our own configurations. We also want IPv6 support with wrappers (we have included IPv6 support in the kernel).

TCP wrappers

TCP wrappers is an excellent tool for controlling and logging network access. Access to this host will be limited to only 2 IP addresses initially. Until the lab obtains a small firewall, TCP wrappers will be the means for denying access from other than these 2 hosts.

Download from ftp.porcupine/pub/security/tcp_wrappers_7.6-ipv6.1.tar.gz.

```
# gunzip tcp_wrappers_7.6-ipv6.1.tar.gz | tar xvf -  
# cd tcp_wrappers_7.6-ipv6.1
```

Edit the Makefile to have the correct location for the standard system daemons (REAL_DAEMON_DIR=/usr/sbin) and to log to the auth log rather than mail log (FACILITY= LOG_AUTH).

```
# make alpha
```

After wrappers has been built successfully, move the binaries to /usr/local/etc.

```
# mv tcpd tcpdchk tcpdmatch try-from safe-finger moduli /usr/local/etc
```

We will wrap ftp and telnet in /etc/inetd.conf for the moment. After ssh has been installed, we will comment out these lines.

```
ftp  stream tcp  nowait root  /usr/local/etc/tcpd  ftpd
telnet stream tcp  nowait root  /usr/local/etc/tcpd  telnetd
```

To complete the TCP wrappers set up, we need to configure /etc/hosts.allow and /etc/hosts.deny. The rules in /etc/hosts.allow are checked first and if no match is found there then /etc/hosts.deny is checked. The first rule that matches causes an exit. If the match is in /etc/hosts.allow, the connection is allowed. If the match is in /etc/hosts.deny, the connection is denied. If no match is found in either file or there are no files, the connection is allowed. Our security policy for this host is very simple, we will deny all access except for ssh from two specific hosts. To test our setup we will allow ALL services from the two trusted IP addresses. After ssh has been installed, we will change "ALL" to sshd in /etc/hosts.allow.

```
/etc/hosts.deny
ALL: ALL
```

```
/etc/hosts.allow
ALL: 192.168.231.95, 192.168.127.186
```

Make sure that permissions for these files are set appropriately.

```
# chmod 640 /etc/hosts.allow /etc/hosts.deny
# ls -l /etc/hosts.*
-rw-r----- 1 root  system    161 Dec 17 23:26 /etc/hosts.allow
-rw-r----- 1 root  system    107 Dec 24 21:03 /etc/hosts.deny
```

The TCP wrappers logs all connections to the log facility that is specified in the Makefile. Both successful and unsuccessful connections are logged.

Let's test it before proceeding.

[The following is from auth.log]

```
Dec 24 21:03:31 sansproj sshd[1105]: refused connect from  
192.168.45.14
```

```
Dec 24 21:05:02 sansproj sshd[1108]: Accepted password for bobbi \  
from 192.168.127.186 port 3547 ssh2
```

In order to make sure that tcpd.h and libwrap.a will be found during the installation of ssh, we will copy them to /usr/local/lib.

```
cp -p tcpd.h /usr/local/lib  
cp -p libwrap.a /usr/local/lib
```

Secure Shell – ssh

As described by SSH Communications Security Corporation, ssh is a mechanism for secure remote access over the inherently insecure Internet. Ssh provides security with strong authentication of both the client and the server. It uses public key cryptography with strong encryption methods. Telnet and other timesharing services allow passwords to be transmitted in clear text. With the secure shell, the whole session is encrypted.

There is really very little similarity between version 1 and 2. Ssh version 2 is actually 3 protocols – establishment of the ssh connection, user authentication, and a transport protocol.

Compaq does have a free version of ssh available for download for Tru64 Unix. We have chosen not to use this so we can configure ssh with tcp wrappers.

We will install OpenSSH. We also need OpenSSL and zlib as well as libwrap.a [installed with TCP wrappers] to complete the ssh installation.

This table gives the information we need to obtain the necessary tools.

Tool	ftp site	Version
zlib	ftp.info-zip.org/pub/infozip/zlib	1.1.3
ssl	ftp.openssl.org/source	0.9.6b
ssh	ftp.openbsd.org/pub/OpenBSD/OpenSSH/portable	3.0.2p1

Build zlib.

```
# gunzip zlib-1.1.3.tar.gz | tar -xvf -
# cd zlib-1.1.3
```

Read the README file and Makefile.

```
# ./configure
# make test
# make install
```

Build OpenSSL.

```
# gunzip openssl-0.9.6b.tar.gz | tar -xvf -
# cd openssl-0.9.6b
```

Read the README and Makefile files.

```
# ./configure
# make
# make test
# make install
```

We are now ready to install ssh itself. We will select 3 options to be used with the configure script. We will be compiling ssh with TCP wrappers support

(--with-tcp-wrappers). We want to prevent the ssh from using rsh if a remote machine is not running ssh (--without-rsh). We will not be running ssh as suid (--disable-suid-ssh0).

Build ssh.

```
# gunzip openssh-3.0.2p1.tar.gz | tar -xvf -
# cd openssh-3.0.2p1
```

Let's install it.

```
# ./configure --with-tcp-wrappers --without-rsh --disable-suid-ssh
```

OpenSSH has been configured with the following options:

User binaries: /usr/local/bin

System binaries: /usr/local/sbin

Configuration files: /usr/local/etc

Askpass program: /usr/local/libexec/ssh-askpass

Manual pages: /usr/local/man/manX

PID file: /var/run

```
sshd default user PATH:
/usr/bin:/bin:/usr/sbin:/sbin:/usr/local/bin
Random number collection: Builtin (timeout 200)
Manpage format: man
PAM support: no
KerberosIV support: no
Smartcard support: no
AFS support: no
S/KEY support: no
TCP Wrappers support: yes
MD5 password support: no
IP address in $DISPLAY hack: no
Use IPv4 by default hack: no
Translate v4 in v6 hack: no

Host: alphaev67-dec-osf5.1
Compiler: cc
Compiler flags: -g
Preprocessor flags: -I/usr/local/ssl/include
Linker flags: -L/usr/local/ssl/lib
Libraries: -lwrap -lz -lsecurity -ldb -lm -laud -lcrypto

WARNING: you are using the built-in random number collection service.
Please read WARNING.RNG and request that your OS vendor includes
/dev/random in future versions of their OS.
```

We verify that TCP wrappers support is active. It is interesting that the version number is displayed as 5.1 rather than 5.1A. We have verified that the CDs used to install the OS clearly state the version to be 5.1A.

The only settings we change from the original configuration file for the daemon is the ListenAddress to be the IP address of the host. We feel this adds a bit more security. We also disallow root logins.

```
# diff sshd_config sshd_config.orig
11d10
< ListenAddress 172.29.126.142
32,33c31
< #PermitRootLogin yes
< PermitRootLogin no
---
> PermitRootLogin yes
71c69
```

We are not going to use X11 forwarding at the present time. There are no changes that need to be made to the client configuration file at this time.

Let's make sure that all the files we need to run ssh are in /usr/local/etc with the correct permissions.

```
# ls -l /usr/local/etc
total 365
-rwxr-x--- 1 root  system 29312 Dec 16 09:02 safe_finger
-rw-r----- 1 root  system 1088 Dec 16 10:48 ssh_config
-rw-r----- 1 root  system 1050 Apr 3 2001 ssh_config.orig
-rw----- 1 root  system 668 Dec 16 11:33 ssh_host_dsa_key
-rw-r----- 1 root  system 618 Dec 16 11:33 ssh_host_dsa_key.pub
-rw----- 1 root  system 543 Dec 16 11:33 ssh_host_key
-rw-r----- 1 root  system 347 Dec 16 11:33 ssh_host_key.pub
-rw----- 1 root  system 883 Dec 16 11:33 ssh_host_rsa_key
-rw-r----- 1 root  system 238 Dec 16 11:33 ssh_host_rsa_key.pub
-rw-r----- 1 root  system 2256 Dec 16 10:01 ssh_prng_cmds
-rw-r----- 1 root  system 2039 Dec 16 10:36 sshd_config
-rw-r----- 1 root  system 1988 Sep 20 19:15 sshd_config.orig
-rwxr-x--- 1 bin   bin   53856 Dec 16 09:02 tcpd
-rwxr-x--- 1 root  system 62432 Dec 16 09:02 tcpdchk
-rwxr-x--- 1 root  system 62752 Dec 16 09:02 tcpdmatch
-rwxr-x--- 1 root  system 42096 Dec 16 09:02 try-from
```

Move the sshd binaries to /usr/local/sbin. While still in the source directory, make the necessary keys.

```
# make host-key
.....
```

```
Your identification has been saved in /usr/local/etc/ssh_host_key.
Your public key has been saved in /usr/local/etc/ssh_host_key.pub.
The key fingerprint is:
a5:bd:08:44:b3:0c:44:05:09:e4:6a:01:f0:09:2a:27
root@sansproj.giac.dhhs.gov
Generating public/private dsa key pair.
Your identification has been saved in /usr/local/etc/ssh_host_dsa_key.
Your public key has been saved in
/usr/local/etc/ssh_host_dsa_key.pub.
The key fingerprint is:
1d:2f:51:28:66:10:6f:03:fe:77:92:f6:a0:eb:eb:30
root@sansproj.giac.dhhs.gov
Generating public/private rsa key pair.
```

Your identification has been saved in /usr/local/etc/ssh_host_rsa_key.
Your public key has been saved in /usr/local/etc/ssh_host_rsa_key.pub.
The key fingerprint is:
0c:ee:6d:d7:47:d8:60:73:c4:f7:c4:7b:25:df:80:bf
root@sansproj.giac.dhss.gov

One last task to complete the installation of ssh. We need a start/stop script in /sbin/init.d. The following is an adaptation of the sample script from the course notes on Securing Unix – SANS Institute Track 6.5 – Solaris Practicum by Hal Pomeranz.

```
#!/bin/sh

# SSH2 start/stop script

case $1 in
  start)
    # Start the daemon
    echo "Starting ssh service"
    if [ -x /usr/local/sbin/sshd -a -f /usr/local/etc/sshd_config
]; then
        /usr/local/sbin/sshd -f /usr/local/etc/sshd_config
    fi
    ;;
  stop)
    # Stop the ssh daemon
    echo "Shutting down the SSH service"
    PID=`cat /var/run/sshd.pid`
    if [ -z "$PID" ]
    then
        echo "SSH daemon not running"
    else
        /usr/bin/kill ${PID} 1> /dev/null 2>&1
    fi
    ;;
  *)
    echo "Usage: $0 {start|stop}"
esac
exit 0
```

Set the permissions appropriately and set up the link in /sbin/rc3.d.

```
# chmod 750 /sbin/init.d/sshd
# ln -s /sbin/init.d/sshd /sbin/rc3.d/S85ssh
```

```
# ln -s /sbin/init.d/sshd /rc0.d/K15ssh
```

Start ssh and validate that it is working properly. Then edit `/etc/hosts.allow` to only allow sshd connections. Disallow ftp and telnet connections in `/etc/inetd.conf`.

Division of Privileges

Tru64 Unix, as discussed in the Tru64 Unix Security manual, has a proprietary feature for distributing superuser administrative privileges among multiple users. The concept behind the dop utility is that particular classes of administrative tasks can be assigned to specific users or classes of users. This leads to access limitations to the root account itself in a similar way to the freeware tool sudo. The goal is to limit access to root's password while providing a secure mechanism for the sharing of administrative tasks that require superuser privileges.

Related administrative tasks have been organized into logical groupings. Appendix O is a table listing the various roles for the division of privileges.

Here is the AccountManagement privilege that can be used to do the tasks listed for Accounts on the sysman menu.

- Accounts [accounts]
Manage local users [users]
Manage local groups [groups]
Manage NIS users [nis_users]
Manage NIS groups [nis_groups]

What is sysman? The following is from the `sysman_intro` manual page.

DESCRIPTION
SysMan is a suite of applications for managing Tru64 UNIX systems. The SysMan applications provide a simple, easy to use, graphical user interface (GUI) for common system management tasks including installation and configuration.

Sysman has a command line interface. It can therefore be easily used from a non-graphics terminal.

We will experiment with dop on the lab machine to see if it will meet our needs for delegation of roles. At first glance, it might work well in a larger shop than ours where the system administration tasks can be

easily divided to match specific administrative roles. In our environment, each system administrator performs a wide variety of administrative tasks; so full root privileges are needed for all administrative roles.

Division of Privileges is audited by auditd with our present auditing configuration. The example below is for a password change for user testusr3 done by user bobbi, our system administrator. We will use sysman dopconfig to give bobbi AccountManagement privileges.

```
$ dop accounts
```

No password is needed for dop. The command above takes you to the sysman menu for AccountManagement. The user testusr3 was selected and given a new password.

Here is the audit information for the password change using this mechanism.

AUID:RUID:EUID	PID	RES/(ERR)	EVENT
5497:0:0	1873	0x0	auth_event (Local Protected Password Database modified by dxaccounts User Entry testusr3: MODIFICATION. Old value for u_pwd: * New value for u_pwd: 4DBj.I3daOD1EjuXDrY9qNmo)
5497:0:0	1878	0x0	auth_event (Local Protected Password Database modified by dxaccounts User Entry testusr3: MODIFICATION. Old value for u_pwd: 4DBj.I3daOD1EjuXDrY9qNmo New value for u_pwd: tncucFgKjLkJMNOzwpvQy9s2)
5497:0:0	1886	0x0	auth_event (root su bobbi)

Notice the reference to dxaccounts even though the interface used was the sysman command line interface.

We will evaluate the dop facility. Once the user has been set up, no password is needed. There is flexibility in the commands that the user has within a role because “actions” can be added and deleted. We do have need for users to have privileges to issue commands as user other than root. There are occasions when it is not appropriate for these users to have the password for those particular accounts. Sudo could be used for these instances. We will evaluate the need for having both and how each fits in our environment.

Ongoing maintenance

Backups

The third SANS Top Twenty Vulnerabilities is “non-existent or incomplete backups.” Appendix S contains a simple backup script that will be used regularly to backup all files on the system. The capability of the DAT (12G/24G) tapes is sufficient to back up all file systems on our 2 9-gigabyte disks.

A good backup strategy is essential in any security plan. Backups provide a means to restore a compromised system whether the system was compromised by a hardware failure, a natural disaster or a security breach.

Backups alone are not sufficient, however. They must be tested regularly by doing restores to ensure their integrity. No backup policy is complete without incorporating regular restore testing.

The backup script will be run weekly and after any major modification to the system.

Our backup tapes are stored in a fireproof safe.

Integrity checks

As recommended by Compaq, the commands `fverify` and `authck` will be run regularly.

The command `authck` checks the internal consistency of the authentication databases. It is run without arguments. From the `authck` manual page,

If `authck` did not detect any inconsistencies, it exits with a status of 0 (zero). If the user is not authorized, `authck` exits with a status of 1. If the user specifies the wrong argument syntax, `authck` exits with a status of 2. Otherwise, `authck` exits with status equal to the number of inconsistencies found.

The following, from the Security manual, explains the role of `fverify` in integrity checking.

Fverify – The fverify program reads subset inventory records from standard input and verifies that the attributes for the files on the system match the attributes listed in the corresponding records. Missing files and inconsistencies in file size, checksum, user ID, group ID, permissions, and file type are reported.

Patches

It is important to be aware of all patches to the system as they become available, particularly security patches. The best way to do this is to subscribe to the mailing lists for both security patches and for the OS in general. To do this, use the Compaq web site patch mailing lists and follow the directions for subscribing to security and Tru64 Unix patches. The URL is: <http://www.support.compaq.com/patches/ mailing-list.shtml>.

Logs

Some of the logs will continue to be checked manually, e.g. /var/adm/messages. We will investigate installing a log checking tool. We also want to investigate how the Event Manager can be used to help notice us of security related problems.

The auditing script sends us mail when it is run, but we still have to scrutinize it manually.

Testing the configuration

Password customizations

An important part of our security policy concerns passwords. Let's verify some of the settings.

- Test that new accounts are initially locked and have pre-expired passwords
 - Set up a new account – testusr4
 - Login as testusr4

- The following is the verification that the account is locked initially. The information was obtained by running the script to look at the audit log.

```
-1:0:0      4051  -1      auth_event ( testusr4 account lockout
denies account access locked_out_acct_es reason(s) for lockout are
Account has explicit administrative lock. )
```

- Use edauth to remove the administrative lock
 - # edauth testusr4 – remove administrative lock by changing the u_lock entry to be u_lock@
 - # edauth -g testusr4
 testusr4:u_name=testusr4:u_id#22222:u_pwd=Lcr5fZdLLX2wo.S5LJ7EXq2o:u_succhg#1009463823:\ :u_unsucchg#1009463831:u_pwdict=mYzQ0CfNidNYQ :u_oldcrypt#0:u_suclog#1009463823:\ :u_suctty=pts/3:u_unsuctty=INET#capu-dsl-128-231-150-95.net.nih.gov:u_unsuclog#1009463361:u_lock@:\ :chkent:
 - attempt to login again

```
Last successful login for testusr4: NEVER
Last unsuccessful login for testusr4: Thu Dec 27 09:29:21 EST 2001
from capu-dsl-128-231-150-95.net.nih.gov
There have been 1 access failures for your account.
```

Your password has expired.

Old password:

```
Last successful password change for testusr4: Thu Dec 27 09:28:36 EST
2001
```

```
Last unsuccessful password change for testusr4: NEVER
```

Do you want (choose one option only):

- 1 Pronounceable passwords generated for you
- 2 A string of characters generated for you
- 3 A string of letters generated for you
- 4 To pick your password

Select ONE item by number: 1

You have selected:

Pronounceable passwords generated for you

Generating random pronounceable password for testusr4.
The password, along with a hyphenated version, is shown.

(Password generation will be a bit slow.)

Hit <RETURN> or <ENTER> until you like the choice.

When you have chosen the password you want, type it in.

Note: type your interrupt character or `quit' to abort at any time.

Password: bludfomcuhehekta Hyphenation: blud-fom-cu-he-hek-ta

Enter password:

Password: vequefvihegnircebaf Hyphenation: ve-quef-vi-heg-nirc-eb-af

Enter password:

Password: manhajoidwuwatejsi Hyphenation: man-haj-oid-wu-wat-ej-si

Enter password:

Password: lenuvpotil Hyphenation: len-uv-pot-il

- One of our policies established that new passwords couldn't be changed for 24 hours.

```
sansproj.giac.dhhs.gov> whoami
```

```
testusr4
```

```
sansproj.giac.dhhs.gov> passwd
```

```
Password not changed: minimum time between changes has not elapsed.
```

- Minimum password length – attempt to enter a password less than 8 characters

```
sansproj.giac.dhhs.gov> passwd
```

```
Old password:
```

```
Last successful password change for bobbi: Mon Dec 24 12:06:05 EST  
2001
```

```
Last unsuccessful password change for bobbi: Thu Dec 27 09:49:15 EST  
2001
```

Do you want (choose one option only):

- 1 Pronounceable passwords generated for you
- 2 A string of characters generated for you
- 3 A string of letters generated for you
- 4 To pick your password

Select ONE item by number: 4

You have selected:

To pick your password

Password must be from 8 to 80 characters long.

Illegal password, try again.

New password:

- Triviality checks – verify that existing group names and user names cannot be used and that dictionary words cannot be used

```
sansproj.giac.dhhs.gov> passwd
```

Old password:

Last successful password change for bobbi: Mon Dec 24 12:06:05 EST 2001

Last unsuccessful password change for bobbi: NEVER

Do you want (choose one option only):

- 1 Pronounceable passwords generated for you
- 2 A string of characters generated for you
- 3 A string of letters generated for you
- 4 To pick your password

Select ONE item by number: 4

You have selected:

To pick your password

New password:

`restrict' is (or looks too much like) a group name to be a password.

Illegal password, try again.

New password:

`password' is an English word, and passwords may not be.

Illegal password, try again.

New password:

Password not changed: user stopped program.

Connectivity checks

- Can we get to our host from the allowed hosts?

```
Dec 25 21:25:32 sansproj sshd[1374]: Accepted password for bobbi  
from 172.29.150.95 port 4420 ssh2
```

- Can we get to our host from unauthorized hosts?

```
Dec 24 20:59:01 sansproj sshd[1101]: refused connect from
```

```
prd1.giac.dhhs.gov
```

- Can root login remotely?

```
Dec 27 10:28:57 sansproj sshd[4272]: ROOT LOGIN REFUSED  
FROM 172.29.150.95  
Dec 27 10:28:57 sansproj sshd[4272]: Failed password for root  
from 172.29.150.95 port 1383 ssh2
```

Interesting attempts logged

While checking the logs we discovered some break-in attempts.

```
-1:0:0      972  (err 2)      auth_event ( ftp Anonymous ftp  
account missing ftpd )  
-1:0:0      15424 (err 2)      auth_event ( (unknown) argv[0]=-ca-  
ol-angers-17-86.abo.wanadoo.fr: Invalid account )  
-1:0:0      15426 (err 2)      auth_event ( (unknown) argv[0]=-  
AMarseille-101-1-2-214.abo.wanadoo Invalid account )  
-1:0:0      15438 (err 2)      auth_event ( anonymous Anonymous  
ftp account missing ftpd )  
-1:0:0      15439 (err 2)      auth_event ( anonymous Anonymous  
ftp account missing ftpd )  
-1:0:0      15440 (err 2)      auth_event ( anonymous Anonymous  
ftp account missing ftpd )
```

Verify that we cannot write in /usr

First, we will verify that /usr is mounted correctly.

```
# mount  
root_domain#root on / type advfs (rw)  
/proc on /proc type procfs (rw)  
usr_domain#usr on /usr type advfs (ro)
```

...

Now, we attempt to create a file.

```
# touch /usr/bin/TESTIT  
touch: /usr/bin/TESTIT cannot create
```

The Future

We plan on installing host-based intrusion detection software and working with ACLs. We also want to run password cracking software regularly.

We also will install a firewall to better protect the lab host. This host will be used to test out log checking scripts. Logs will be checked manually until the suite of automated tools are in place.

Security is a dynamic process. No host can ever be completely hardened against attack. We will continue to test out mechanism to identify and mitigate the most serious risks.

© SANS Institute 2000 - 2002, Author retains full rights

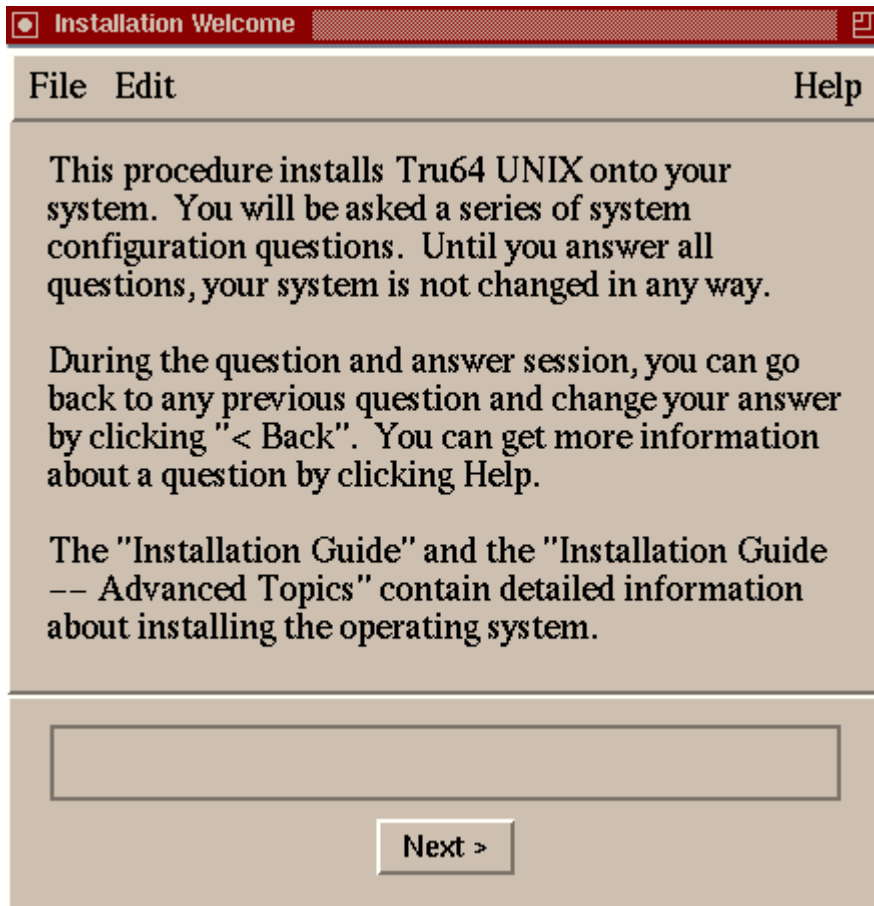
Appendices

Appendix A – Installed subsets

Subset	Status	Description
-----	-----	-----
ESVFBIN200	installed	Modified and New Commands and APIs (Extended System V Functionality)
ESVFMAN200	installed	Man pages for the Modified and New Commands and APIs(Extended System V Functionality)
FSFGZIPSRC520	installed	GNU Gzip Source (GNU Source)
FSFINDENTSRC520	installed	GNU Indent Source (GNU Source)
FSFM4V14520	installed	"m4 Sources" (GNU Source)
FSFPERL520	installed	perl Sources (GNU Source)
FSFRCSSRC520	installed	GNU Revision Control System Source (GNU Source)
OSFACCT520	installed	System Accounting Utilities (System Administration)
OSFADVFS520	installed	AdvFS Commands (System Administration)
OSFADVFSBIN520	installed	AdvFS Kernel Modules (Kernel Build Environment)
OSFADVFSBINOBJECT520	installed	AdvFS Kernel Objects (Kernel Software Development)
OSFADVFSDAEMON520	installed	AdvFS Daemon (System Administration)
OSFBASE520	installed	Base System (- Required -)
OSFBIN520	installed	Standard Kernel Modules (Kernel Build Environment)
OSFBINCOM520	installed	Kernel Header and Common Files (Kernel Build Environment)
OSFBINOBJECT520	installed	Standard Kernel Objects (Kernel Software Development)
OSFC2SEC520	installed	Enhanced Security (System Administration)
OSFCDEDT520	installed	CDE Desktop Environment (Windowing Environment)
OSFCDEMANOP520	installed	Ref Pages: CDE Development (Reference Pages)
OSFCDEMANOS520	installed	Ref Pages: CDE Admin/User (Reference Pages)
OSFCDEMIN520	installed	CDE Minimum Runtime Environment (Windowing Environment)
OSFLINET520	installed	Basic Networking Services (Network-Server/Communications)
OSFCMPLR520	installed	Compiler Back End (Software Development)
OSFDCMT520	installed	Doc. Preparation Tools (Text Processing)
OSFDCMTEXT520	installed	Doc. Preparation Tools Extensions (Text Processing)
OSFEXER520	installed	System Exercisers (System Administration)
OSFHBASE520	installed	Base System - Hardware Support (- Required -)
OSFHBBIN520	installed	Hardware Kernel Modules (Kernel Build Environment)
OSFHBBINCOM520	installed	Hardware Kernel Header and Common Files (Kernel Build Environment)
OSFHBBINOBJECT520	installed	Hardware Kernel Objects (Kernel Software Development)
OSFIMXE520	installed	Compaq Management Agents Version 2.1b(System Administration)
OSFINCLUDE520	installed	Standard Header Files (Software Development)
OSFJAVA520	installed	Java 1.1.8-10 Environment (General Applications)
OSFKBDPCXAL520	installed	PCXAL Keyboard Support (Windowing Environment)
OSFKTOOLS520	installed	Kernel Debugging Tools (System Administration)
OSFLSMBASE520	installed	Logical Storage Manager (System Administration)
OSFLSMBIN520	installed	Logical Storage Manager Kernel Modules(Kernel Build Environment)

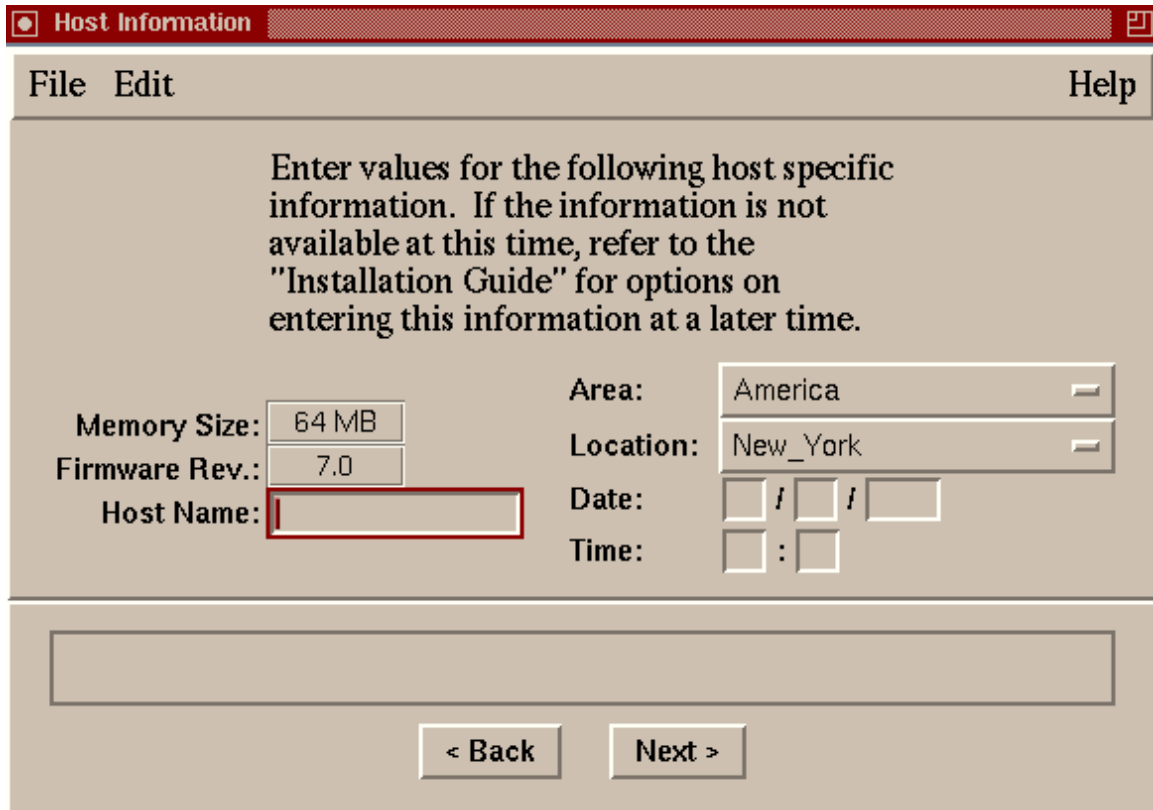
OSFLSMX11520	installed	Logical Storage Manager GUI (System Administration)
OSFMANOP520	installed	Ref Pages: Programming (Reference Pages)
OSFMANOS520	installed	Ref Pages: Admin/User (Reference Pages)
OSFMITFONT520	installed	X Fonts (Windowing Environment)
OSFNETCONF520	installed	Basic Networking Configuration
Applications (System Administration)		
OSFNETSCAPE520	installed	Netscape Communicator V4.76 (Windows Applications)
OSFNFS520	installed	NFS(tm) Utilities (Network-Server/Communications)
Server/Communications)		
OSFOEMBASE520	installed	Tru64 UNIX Base System (- Required -)
OSFPAT00000031520	installed	Patch Tools (- Required -)
OSFPATC0000800520	installed	Patch: Fixes problem reported in Patches)
SSRT0767U.(Common Desktop Environment (CDE)		
OSFPERL520	installed	perl 5.6.0 Runtime (General Applications)
OSFSER520	installed	X Servers Base (Windowing Environment)
OSFSERPC520	installed	X Servers for PCbus (Windowing Environment)
OSFSERVICETOOLS520	installed	Service Tools (System Administration)
OSFSYSMAN520	installed	Base System Management Applications and Utilities (System Administration)
OSFTCLBASE520	installed	Tcl Commands (General Applications)
OSFTKBASE520	installed	Tk Toolkit Commands (General Applications)
OSFX11520	installed	Basic X Environment (Windowing Environment)
OSFXADMIN520	installed	Graphical System Administration
Utilities (System Administration)		
OSFXADVFS520	installed	AdvFS Graphical User Interface (System Administration)
Administration)		
OSFXC2SEC520	installed	Enhanced Security GUI (System Administration)
OSFXDEMOS520	installed	Demo X Applications (Windows Applications)
OSFXSYSMAN520	installed	Graphical Base System Management Utilities(System Administration)

Appendix B – Welcome screen



© SANS Institute

Appendix C – Host Information Dialog Box



The image shows a dialog box titled "Host Information" with a menu bar containing "File", "Edit", and "Help". The main area contains instructions: "Enter values for the following host specific information. If the information is not available at this time, refer to the 'Installation Guide' for options on entering this information at a later time." Below this are several input fields: "Memory Size:" with a text box containing "64 MB"; "Firmware Rev.:" with a text box containing "7.0"; "Host Name:" with an empty text box; "Area:" with a dropdown menu showing "America"; "Location:" with a dropdown menu showing "New_York"; "Date:" with three separate boxes for day, month, and year; and "Time:" with two boxes for hour and minute. At the bottom are two buttons: "< Back" and "Next >".

Host Information

File Edit Help

Enter values for the following host specific information. If the information is not available at this time, refer to the "Installation Guide" for options on entering this information at a later time.

Memory Size: 64 MB

Firmware Rev.: 7.0

Host Name:

Area: America

Location: New_York

Date: / /

Time: :

< Back Next >

© SANS Institute 2000 - 2002

Appendix D

Set Root Password Dialog Box

Set root Password

File Edit Help

Enter a password to use as the root (superuser) password. Be sure to remember this password, because it is needed to log in as the user "root" following installation. The password must be at least six characters long, and at least one of the first eight characters must be a number, special character or uppercase letter.

Enter root Password:

Verify Password:

< Back Next >

© SANS Institute 2000 - 2002

Appendix E - Software Selection Dialog Box



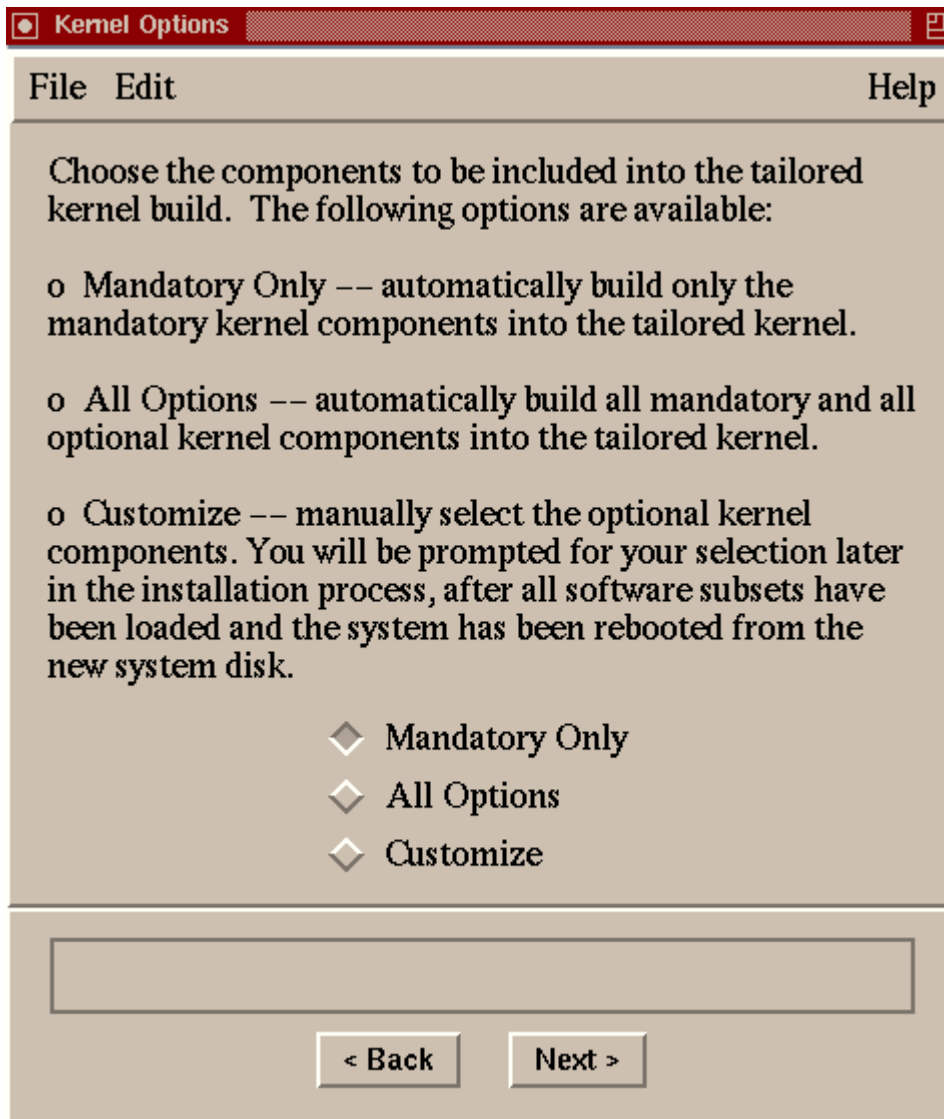
© SANS

Appendix F – Software Selection Dialog Box



© SANS

Appendix G – Kernel Options Dialog Box



© SANS

Appendix H – Select File System Layout Dialog Box



© SANS

Appendix I – Custom File Layout Dialog Box

Custom File System Layout

File Edit Help

For each file system, select the disk, partition on that disk, and file system type.

File System Layout

Use LSM: Yes

	Disk	Partition	Type
root	dsk1	a (128 MB)	AdvFS
/usr	dsk1	g (700 MB)	AdvFS
/var	dsk2	h (981 MB)	UFS
swap1	dsk1	b (196 MB)	
swap2	dsk2	b (196 MB)	

Identify Disk... Edit Partitions...

LSM Options

Disk	Private Region
dsk1	d (2 MB)
dsk2	d (2 MB)

< Back Next >

© SANS

Appendix J – Installation Summary Box

Tru64 UNIX T5.0-23 (Rev. 861.3) Installation Summary

Review this summary of the information you have entered, and make any necessary changes. When everything is correct, press the Finish button to let the installation proceed.

General Information

Memory Size: 64 MB Area: America

Firmware Rev.: 7.0 Location: New_York

Host Name: mysystem Date: 12 / 17 / 1999

Change root Password... Time: 12 : 50

Software Subsets

- Mandatory Only Show List...
- All Software Show List...
- Customize Edit List...

Country Support: English (US)

Kernel Options

- Mandatory Only
- All Options
- Customize

File System Layout

Use LSM: Yes

	Disk	Partition	Type
root	dsk1	a (128 MB)	AdvFS
/usr	dsk1	g (700 MB)	AdvFS
/var	dsk2	h (981 MB)	UFS
swap1	dsk1	b (196 MB)	
swap2	dsk2	b (196 MB)	

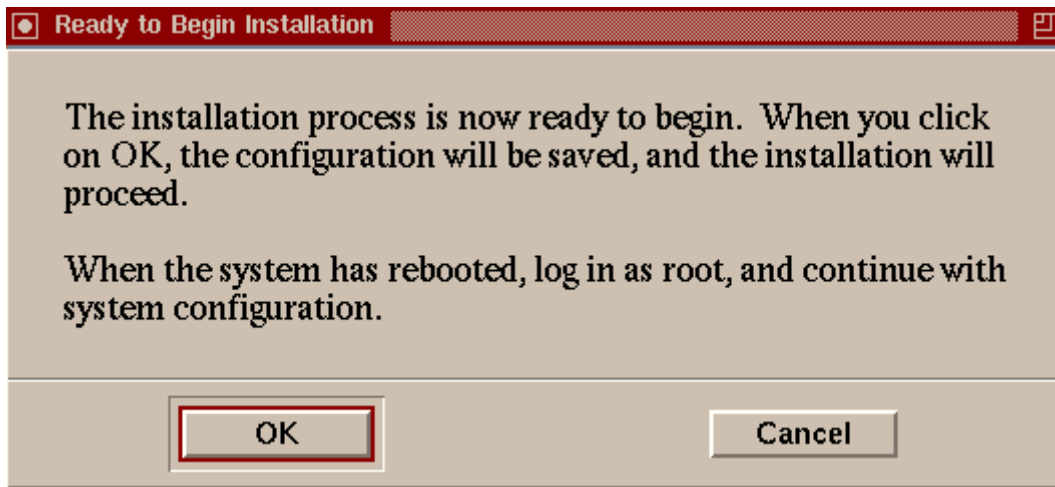
Identify Disk... Edit Partitions...

LSM Options

Disk	Private Region
dsk1	d (2 MB)
dsk2	d (2 MB)

Finish Reset Shell Window Quit Help

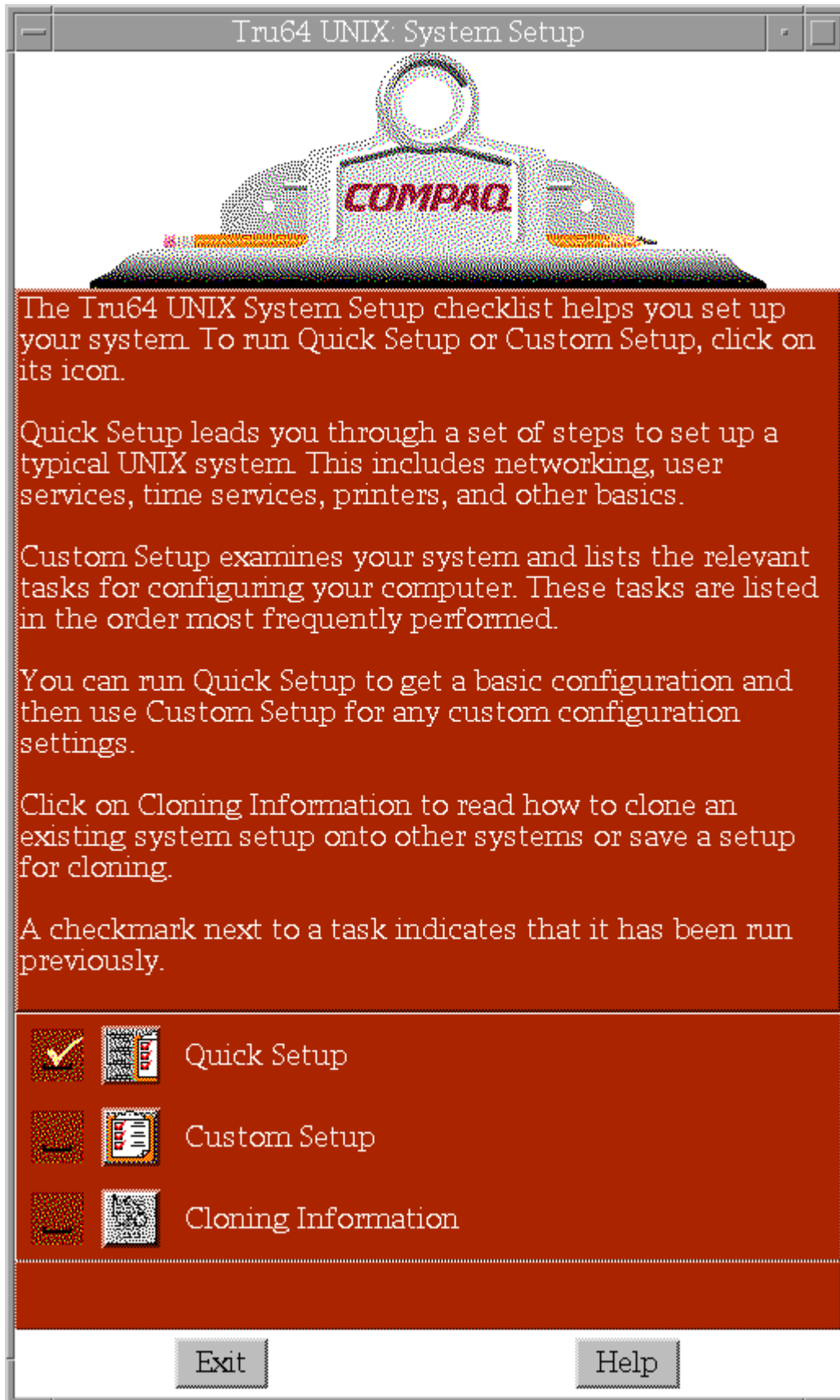
Appendix K – Ready to Begin Installation Dialog Box



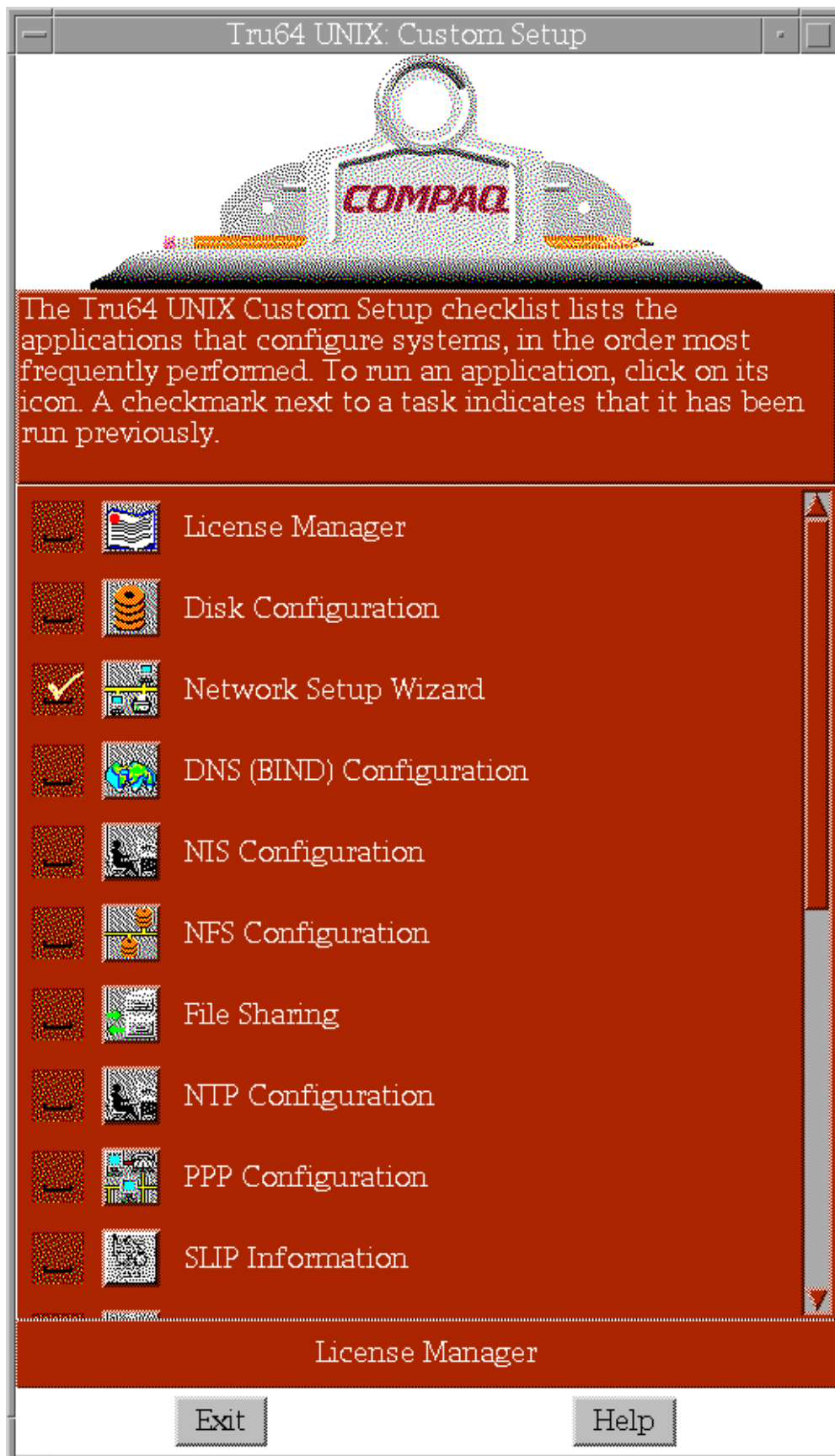
© SANS Institute 2000 - 2002, Author

Appendix L – System Setup Window

© SANS Institute 2000 - 2002, Author retains full rights.



Appendix M – Custom Setup Dialog Box



Appendix N – Audited events - /etc/sec/audit_events

[events that are not audited have been deleted to save space]

more /etc/sec/audit_events

! Audited system calls:

mknod	succeed fail
mount	succeed fail
umount	succeed fail
setuid	succeed fail
setlogin	succeed fail
reboot	succeed fail
revoke	succeed fail
chroot	succeed fail
sethostname	succeed fail
settimeofday	succeed fail
setreuid	succeed fail
setregid	succeed fail
truncate	succeed fail
ftruncate	succeed fail
setgid	succeed fail
shutdown	succeed fail
adjtime	succeed fail
sethostid	succeed fail
setsid	succeed fail
setdomainname	succeed fail
exportfs	succeed fail
alternate setsid	succeed fail
swapon	succeed fail
utc_adjtime	succeed fail
security	succeed fail
uadmin	succeed fail
audcntl	succeed fail
setsysinfo	succeed fail
! Audited trusted events:	
audit_start	succeed fail
audit_stop	succeed fail
audit_setup	succeed fail
audit_suspend	succeed fail
audit_log_change	succeed fail
audit_log_creat	succeed fail
audit_xmit_fail	succeed fail
audit_reboot	succeed fail
audit_log_overwrite	succeed fail
audit_daemon_exit	succeed fail
login	succeed fail
logout	succeed fail
auth_event	succeed fail
audgen8	succeed fail

Appendix O - Table of Division of Privileges Roles

Network Configuration	Storage Configuration
Mail Management	Storage Management
Mail Configuration	Process Management
Printer Management	File Management
Printer Configuration	Power Management
Distributive Printing	Keyboard Configuration
Advanced Printing	Security
Event Management	Host Management
Event Configuration	Account Management
Superuser CDE Configuration	CDE Configuration
Cluster Configuration	Software Management
Cluster Monitoring	SysMan Accounts, including local users, local groups, NIS users, NIS groups

Appendix P – Sample script for Tru64 Unix Security manual to extract login and logout information from the audit logs

```
# more /var/adm/local/login.extract
#!/usr/bin/ksh -ph
# Script to return summary of login/logout activities on the
# system since the last time it was run.
export PATH=/usr/sbin:/usr/bin:/usr/ccs/bin:/sbin
# where this script should run
Bdir=/var/adm/local
# where to find audit log files
Adir=/var/audit
Ofile="{Bdir}/lasttime"
Nfile="{Bdir}/newtime"
Afile="{Bdir}/lastdata"
Tfile="{Bdir}/lastmsg"
Events="-e trusted_event"
umask 077
# ensure the output format we need from date.

export LANG=C LC_ALL=C
export TZ=:UTC
if [ ! -f "{Ofile}" ]
then
print 700101000001 > "{Ofile}"
touch -t 197001010000.01 "{Ofile}"
fi
date +%y%m%d%H%M%S > "{Nfile}"
curfile=$(auditd -q)
```

```
auditd -dx
```

```
sleep 20 # give time for compression of the old log
while [ -f "$curfile" -a -f "$curfile.Z" ] || [ -f "$curfile" \
-a -f "$curfile.gz" ]
do
sleep 2 # wait some more
done
: > "${Afile}"
for af in $(find "$Adir" -name "auditlog.*" -newer "${Ofile}" \
-print | sort)
do
audit_tool -b -t "${Ofile}" -T "${Nfile}" >> \
"${Afile}" -o -Q $Events "${af}" 2>/dev/null
# the suppressed errors are for the {un,}compressed messages
done
TZ=:localtime

if [ -s "${Afile}" ]
then
audit_tool -B -Q "${Afile}" > "${Tfile}"
if [ -s "${Tfile}" ]
then
Mail -s 'login/out audit summary' root < "${Tfile}"
fi
fi
mv -f "${Nfile}" "${Ofile}"
rm -f "${Afile}"
```

Appendix Q – Sample Banner

Warning Notice!

This is a U.S. Government computer system, which may be accessed and used only for authorized Government business by authorized personnel. Unauthorized access or use of this computer system may subject violators to criminal, civil, and/or administrative action.

All information on this computer system may be intercepted, recorded, read, copied, and disclosed by and to authorized personnel for official purposes, including criminal investigations. Such information includes sensitive data encrypted to comply with confidentiality and privacy requirements. Access or use of this computer system by any person, whether authorized or unauthorized, constitutes consent to these terms. There is no right of privacy in this

system.

Appendix R – Government Security Safeguards Matrix

[The last 2 columns have been cut to conserve space for the table.]

Exhibit III-A: Matrix of Minimum Security Safeguards

Explanation: This matrix is used to identify a minimum set of safeguards, by security level, which should be implemented to protect AISs, AIS facilities, and/or ITUs.¹

Justification for non-implementation of these safeguards should be based on the results of a formal risk analysis (and cost-benefit) study.

Directions: Scan the Xs and Os beneath each security level designation. An X means that the security safeguard listed to the left is a requirement. An O means that the security safeguard is optional.

	Level 4 High Sensitivity/ Criticality, Nat'l Security	Level 3 High Sensitivity/ Criticality
1. Ensure that a complete and current set of documentation exists for all operating systems.	X	X
2. Require use of current passwords and log-on codes to protect sensitive AIS data from unauthorized access.	X	X
3. Establish procedures to register and protect secrecy of passwords and log-on codes, including the use of a nonprint, nondisplay feature.	X	X
4. Limit the number of unsuccessful attempts to access an AIS or a database.	X	X
5. Develop means whereby the user's authorization can be determined. (This may include answerback capability.)	X	X
6. Establish an automated audit trail capability to record user activity.	X	X
7. Implement methods, which may include the establishment of encryption, to secure data being transferred between two points.	X	X
8. Ensure that the operating system contains controls to prevent unauthorized access to the executive or control software system.	X	X

¹OMB Circular A-130 requires formal risk analyses for sensitive AISs, AIS facilities, and ITUs. Required security safeguards may change as a result of the risk analysis.

		Level 4 High Sensitivity/ Criticality, Nat'l Security	Level 3 High Sensitivity/ Criticality
9.	Ensure that the operating system contains controls that separate user and master modes of operations.	X	X
10.	Record occurrences of nonroutine user or operator activity (such as unauthorized access attempts and operator overrides) and report to the organizational ISSO.	X	X
11.	Ensure that the operating system provides methods to protect operational status and subsequent restart integrity during and after shutdown.	X	X
12.	Install software feature(s) that will automatically lock out the terminal if it is not used for a predetermined period of lapsed inactive time, for a specified time after normal closing time, or if a password is not entered correctly after a specified number of times.	X	X
13.	Ensure that the operating system contains controls to secure the transfer of data between all configuration devices.	X	0
14.	Establish controls over the handling of sensitive data, including labeling materials and controlling the availability and flow of data.	X	X
15.	Require that all sensitive material be stored in a secure location when not in use.	X	X
16.	Dispose of unneeded sensitive hard copy documents and erase sensitive data from storage media in a manner which will prevent unauthorized use.	X	X
17.	Prepare and maintain lists of persons authorized to access facilities and AISs processing sensitive data.	X	X
18.	Establish procedures for controlling access to facilities and AISs processing sensitive data.	X	X
19.	Furnish locks and other protective measures on doors and windows to prevent unauthorized access to computer and support areas.	X	X
20.	Install emergency (panic) hardware on "Emergency Exit Only" doors. Ensure that emergency exits are appropriately marked.	X	X
21.	Specify fire-rated walls, ceilings, and doors for construction of new computer facilities or modifications of existing facilities.	X	X
22.	Install smoke and fire detection systems with alarms in the computer facility. When feasible, connect all alarms to a control alarm panel within the facility and to a manned guard station or fire station.	X	X
23.	Install fire suppression equipment in the computer facility, which may include area sprinkler systems with protected control valves and/or fire extinguishers.	X	X
24.	Provide emergency power shutdown controls to shut down AIS equipment and air conditioning systems in the event of fire or other emergencies. Include protective covers for emergency controls to prevent accidental activation.	X	X
25.	Provide waterproof covers to protect computers and other electronic equipment from water damage.	X	X
26.	Establish a fire emergency preparedness plan to include training of fire emergency response teams, development and testing of an evacuation plan, and on-site orientation visits for the		

	Level 4 High Sensitivity/ Criticality, Nat'l Security	Level 3 High Sensitivity/ Criticality
local fire department.	X	X
27. Secure communication lines.	X	X
28. Conduct Tempest testing of operating system.	X	0
29. Ensure that all requirements of NSDD-145 (National Security Decision Directive) are met.	X	0
30. Establish detailed risk management program.	X	X
31. Establish Computer Systems Security Plans for sensitive systems.	X	X
32. Conduct formal risk analyses.	X	X
33. Establish employee security awareness and training programs.	X	X
34. Maintain accurate inventory of all hardware and software.	X	X
35. Establish security review and certification program.	X	X
36. Establish contingency plan.	X	X
37. Establish emergency power program.	X	X
38. Ensure that all personnel positions have been assigned security level designations.	X	X
39. Conduct periodic security level designation reviews.	X	X
40. Ensure that all personnel, including contractors, have received appropriate background investigations.	X	X
41. Maintain a list of all personnel, including contractors, who have been approved for 6C (High Risk Public Trust), 5C (Moderate Risk Public Trust), 4C (Top Secret, requiring special security considerations), 3C (Top Secret), and 2C (Secret or Confidential) risk level positions.	X	X

Appendix S – Backup Script

```
#!/bin/ksh
# preliminary script to backup system filesets to 4mm tape
# Bobbi Spitzberg
# This script is for v5.1a

VDUMP=/sbin/vdump
VRESTORE=/sbin/vrestore
ADMIN="bobbi"
DEV=/dev/ntape/tape0
LOG=/usr/local/sys/backup/backup.log
# get date
MMDDYY=`date +"%m%d%y"`
FSYSLIST="/ /usr /var /usr/local /home"

# Check to see if tape is in drive ?
mt -f $DEV rewind
status=$?
if [ $status != "0" ]
then
    echo "$0 : invalid tape"
    mail $ADMIN >> EOF
Backup failed $MMDDYY
EOF
    exit
fi

# backups
# should redirect output of vdump to log file
for fs in $FSYSLIST
do
    TIME=`date +"%H:%M"`
    $VDUMP -NC0uf $DEV $fs
    echo "$MMDDYY $TIME $fs vdump" >> $LOG
done

#rewind tape
mt -f $DEV rewind

#backup log directory
BDIR=/usr/local/sys/backup
echo "Creating restore list of files"
# read tape
```

```

for fs in $FSYSLIST
do
  if [ $fs = "/" ]
  then
    fs="root"
  fi
  fsr=`echo $fs | tr -d '/'`
  echo "listing $fsr"
  $VRESTORE -t -f $DEV > $BDIR/$fsr.vdump.$MMDDYY
done

# rewind the tape
mt -f $DEV rewind
#mt -f $DEV unload

```

Appendix T – Some Compaq Security Recommendations

- Ensure that the /tmp /var/tmp and /var/spool directories are on a file system other than that of the root (/) and /usr directories.
- The ability to verify the integrity of the trusted computing base (TCB) is met by running the fverify and authck commands periodically as determined by your site 's security policy.
- Select either user-chosen or machine-generated passwords and configure as follows:
 - For user-chosen passwords (u_pickpw field in the /etc/auth/system/default file),set the minimum length to 8 characters (u_minlen#8 and the maximum length to 80 characters (u_maxlen#80 .
 - For machine-generated passwords (no u_pickpw field in the /etc/auth/system/default file),set the minimum length to 0 characters (u_minlen#0 and the maximum length to 10 characters (u_maxlen#10 .The value of 0 for minimum length causes Tru64 UNIX to use the *Green Book* algorithm to generate passwords.
- Ensure that null passwords cannot be used (u_nullpw@)
- Set the password expiration time to 180 days (u_exp#15724800
- Set the account lifetime set to 360 days (u_life#31449600
- Set the depth of the password history file to 9 (u_pwdepth#9
- Set the number of tries to enter a password before locking the account to 5(u_maxtries#5
- Set new accounts to be locked (u_lock
- Set the maximum number of login attempts before the terminal is locked to 10 (t_maxtries#10
- Set the delay between attempted logins to 2 seconds (t_logdelay#2
- Select triviality checks (u_restrict and site password restrictions

(u_policy

Use the Account Manager (dxaccounts) or the edauth program to change the default settings.

The libraries on your system can be used in an attack. Secure the libraries as follows:

- Disable segment sharing by answering yes when prompted by seconfig
- Verify that the permissions are correct (no write access except for the owner) and that the ownership is root on shared libraries (/usr/shlib/*.so, including any linked target files). Use the ls -lL command for this procedure.

Configure user accounts as follows:

- Using the provided default templates, create account templates that reflect your site's security policy.
- Set the umask in the /usr/skel/.login file. (Compaq recommends a value of 027.)
- Designate a restricted shell (Rsh for users where appropriate).
- Verify that each user has a valid entry path (login shell) on the system. Users can be placed directly into an application by executing the application from the user's /home/.profile or from the entry in the /etc/passwd file or as a start point for the user with the execution of a startup program.

Before the audit subsystem kernel option can be configured, it needs to be included for the kernel build. Use the sysman auditconfig utility to configure the audit subsystem any time after the kernel build. Compaq recommends that you configure and run audit as follows:

- Use the default location for audit logs (/var/audit/auditlog.nnn). For overflow protection, put the audit logs on a file system other than root (/and /usr
- Establish an alternate location for audit logs to provide for an overflow of audit log data by editing the /etc/sec/auditd_loc file.
- Send auditd messages to the console (/dev/console).
- Set the audit mask to audit trusted_events and to log the name of a user (as described in your site policy) who attempts to log in to an invalid account.

If you are starting the audit daemon from the command line, use the following command:

```
# /sbin/init.d/audit start
```

Because root access must be carefully controlled and monitored, make sure the following conditions are met:

- That all passwords are changed after a system installation or after support vendors have had access to your machine.
- That the root password is changed before vendor access is granted to prevent exposure of your password generation methodology.
- That the single-user password feature is enabled. See the sulogin 8) reference page.

- That using the su command to become root is logged by audit.
- That the /var/spool/cron/crontabs files are accessible only by root or the owner.
- That root access is restricted to certain devices for login or that users must use the su command to access the root account. See the securetty(4) reference page for more information.
- The logins for the system-supplied UIDs are limited (setting the u_lock field) where appropriate. The following table provides the restrictions recommended by Compaq:

UID Recommended login Status

root	Restricted
daemon	Not allowed
bin	Not allowed
sys	Not allowed
uucp	Restricted
nobody	Not allowed
adm	Restricted
lp	Not allowed

© SANS Institute 2000 - 2002, Author retains full rights.

References

OMB Circular A-130, 8 February, 1996. URL:

<http://www.whitehouse.gov/omb/circulars/a130/text/a130.html>

Sil, "Minimizing Denial of Service Attacks", 13 July, 2001. URL:

<http://www.antioffline.com/stoppingdos.html>

"CERT® Advisory CA-2000-21 Denial-of-Service Vulnerabilities in TCP/IP Stacks", 4 December 2000. URL:

<http://www.cert.org/advisories/CA-2000-21.html>

Department of Health and Human Services, "Automated Information Systems Security Program Handbook" Release 2.0, May 1994. URL:

<http://irm.cit.nih.gov/policy/aissp.html>

"Tru64 Security Guide".

URL:http://www.geocities.com/sabernet_net/papers/Tru64.html

http://www.tru64unix.compaq.com/docs/best_practices/BP_EVAL/bp_eval.pdf
f Evaluated configuration document

Compaq Computer Corporation. "Tru64 Unix: Evaluated Configuration, May 2001. URL:

<http://www.tru64unix.compaq.com/internet/security/sshdwn.html>

SSH Communications Security Corp. URL:

http://www.ssh.com/products/ssh/administrator24/SSH_Secure_Shell.html

Compaq Computer Corporation, "Tru64 Unix:Security", June, 2001. URL:

http://www.tru64unix.compaq.com/docs/base_doc/DOCUMENTATION/V51A_HTML/ARH95DTE/TITLE.HTM

SANS Institute. "The Twenty Most Critical Internet Security Vulnerabilities (Updated):The Experts' Consensus", 15 November 2000. URL:

<http://www.sans.org/top20.htm>

SANS Institute, Solaris Security:Step-by-Step, Version 2.0, Copyright 2001.

Pomeranz, Hal and Lee Brotzman, Securing Unix Systems, July, 2001, SANS Institute.