Global Information Assurance Certification Paper

Securing Unix Step by Step

-- Building a Secure Apache Server on Solaris 8

GCUX Practical Assignment Version 1.9

Baoqing Ye

September, 2002

Table of Contents

4. Routine Patching and Updating

Check Configuration

1. Operating System and Administrative Practices

1.1 Network Connection
1.2 Authentication
1.3 Logging and Audit
1.4 Integrity Checking
1.4.Backup

2. Apache Server

2.1 Server Identity
2.2 Access Control
2.3 Audit Logs

Appendix

References

Font usage in this document:

Procedure descriptions and explanations:
Times New Roman, font size 12

```
Commands, executed from command line:
Courier New, font size 12
```

```
Code or output:
Courier New, font size 10
```

[R#]: Reference number (see the list of reference in the last page of the document)
Superscript, font size 12

The purpose of this paper is to give a detailed step-by-step description on how to secure a Unix (Solaris) system, named WEBSERV, which provides secure web services used within a company wide intranet.

System Description:

Hardware environment:

Platform Description: Sun UltraSparc Enterprise 2
CPU: 296 Mhz
Memory: 256 MB
Hard Drive: 8G
The system will be physically located in a locked secure office.

Software Environment:

Operating System environment:

Solaris 8 (07/01):
Minimum disk space requirement: 1.6GB
OS installation is from a read-only media CD-ROM, completed by one single person (i.e. myself) with installation steps logged.

Web Server Environment:

Apache 1.3.26 is chosen as the web server.
SSL add-on: apache-1.3.26+ssl-1.48

Other Software to Hardening the WEBSERV:

PGP: An encryption tool to provide confidentiality.
TCP Wrappers: A method to provide service access control for logging connections
OpenSSL: provides privacy and data integrity between two communication parties
OpenSSH: provides encrypted channels to thwart network-level attacks.
Tripwire: file integrity protection
Fix-modes: sets appropriate permissions on files, overwriting insecure default settings.
NMAP: A security scanner to detect open ports in a system.

Networking environment:

WEBSERV will provide web services available to company wide intranet to be accessed by thousands of employees and consultants.
The system is protected by a company firewall along with an edge router.

Risk Analysis of the System

WEBSERV will be accessed within a company wide intranet. The potential intrusion or penetration risks come from either legitimate employees (inner hackers or crackers), or outsider crackers who have gained the access to the intranet with authorized IP addresses to access the WEBSERV.

The following are the *key security concerns* that may lead to compromising WEBSERV and its services:

1. Possible security vulnerabilities in the Solaris operating system, including improper system configuration.
2. Possible Apache Web Server Vulnerabilities
3. Security implications of web applications
4. Possible security problems in third party software.

The primary *potential threats* to the system are:

1. Break-in to the WEBSERV and modify system configuration or confidential information.
2. Thefts of private information or confidential data by unauthorized users if WEBSERV is compromised.
3. Compromise of the WEBSERV, which is then used as a relay or zombie for further attacking to other systems
4. Compromise of the system that can cause denial or interruption of WEBSERV services.

The WEBSERV shall satisfy the following *security requirements*:

The primary goal of the WEBSERV is to provide on-line services for internal employees. There are two-fold security requirements. First, the employee information collected and stored on-line is not considered private for internal legitimate employees, but is confidential to unauthorized outsiders. This requirement can be satisfied through security features including access control by both edge routers and WEBSERV itself. Second, The WEBSERV shall provide continuous and correct on-line service without being compromised.

The following mechanisms are used to satisfy the two major security goals above:

1. The WEBSERV shall be physically secured and kept in an isolated room with controlled access.
2. Secure the OS platform so as to thwart the potential breach into the WEBSERV.
3. The apache web server must be secured and protected
4. All web service applications must be security bug/holes free.
5. Additional software will be necessary to hardening the WEBSERV platform. All third party application should be checked to eliminate known security vulnerabilities.

The step-by-step installation and ongoing maintenance in the following sections will further detail how to substantiate the above security schemes.

Required *Services* run on WEBSERV:

1. General on-line web services, including department homepages and employee information.
2. Web education services, including on-line exams for training purposes.

Access Control requirements:

1. Only company employees have the access to the on-line services.
2. Employees must log on to the WEBSERV from a host in the company intranet.

Step by Step Guide

1. OS Installation

1.1 Initial installation

The initial installation of the Solaris operating system is in an isolated environment without network connection, booted from Solaris8 (07/01) CD-ROM to start with a clean system image.
The following are parameters required during the installation process:

Parameters	**Answer/Input**
Networked?	Networked [1]
DHCP	No
Hostname	WEBSERV.domain.name [2]
IP address	10.0.0.1 [3]
Part of subnet?	Yes
Netmask	255.255.255.0
Enable Ipv6?	Yes [4]
Kerberos Security?	No
Name Service	None [5]
Default router	10.0.0.10 [6]
Specify time zone by	Geographic region
Geographic Region	United States / Eastern
Date and Time	2002-07-28 17:41
Root Password	Pass$W0rd [7]
Power Management	Off
Proxy	Direct connection to the Internet
Specify Media	CD
Select type of Install	Custom install

Select software location	North America
Select system locale	English (United States, ISO8859-1)
Select Products	Solaris 8 Documentation English Solaris 8 Software 2 of 2 Computer system supplement CD
Additional product	None
64-bit selection	Yes
Select Solaris Software Group	Core[8]
Disk Selection	c0t0d0

Note:
[1] Select "Networked" even though the machine is currently disconnected
[2] Domain name is for the presentation purpose only. The real domain name is not unveiled for the company information confidentiality purpose
[3] The IP address is the presentation purpose only.
[4] Choosing IPv6 is for a more comprehensive compatibility purpose, it is back compatible with IPv4.
[5] Name service will be configured in a later time after installation
[6] The default router IP address is for the presentation purpose only.
[7] The password is for the presentation purpose only.
[8] The selection can be adjusted based on individual needs from five options: "Entire Plus OEM / Entire / Developer / End User / Core"; For example, "Entire Plus OEM" can be chosen if further tailoring work of customizing packages is done

Disk partition strategy is determined by system space availability, service requirement, and personal preference and experiences.
The partition for the WEBSERV is designed as below:

c0t0d0s0	/	500MB
c0t0d0s1	Swap	601MB
c0t0d0s3	/usr	3333MB
c0t0d0s4	/opt	3283MB
c0t0d0s5	/var	811MB
c0t0d0s7	/export/home	100MB

After the initial installation, reboot the machine.

1.2. Post Install and Network Configuration

Perform the following steps for network configuration [R9]:
1) `vi /etc/defaultrouter`,
-- Confirm the correctness of the IP address of the system's default router
2) `touch /etc/notrouter`

-- Disable IP forwarding and prevent `in.routed` and `in.rdiscd` from starting from starting at boot time.

3) `vi /etc/resolv.conf`

Edit the file to add DNS information:

```
search <domain.name>[1]
nameserver <dns.server.name.#>[1]
```

note [1]: Insert you domain name and DNS server name here.

4) `vi /etc/nsswitch.conf`

Change the "hosts" specification as "`hosts: files dns`"

5) `reboot`

1.3. Adding / confirming additional packages

1) `cd /var/sadm/pkg/`

2) Install the following packages after initial installation:

SUNWter, SUNWaccr, SUNWaccu, SUNWlibC, SUNWdoc, SUNWman.

Note:

SUNWter: Extensive terminfo database entries describing capabilities of terminals and pseudo-terminals.

SUNWaccr: Utilities for accounting and reporting of system activity

SUNWaccu: Utilities for accounting and reporting of system activity

SUNWlibC: Sun Workshop Compilers Bundled libC

SUNWdoc: on-line documentation

SUNWman: on-line manual pages

Insert Solaris 8 Software CD 1/2

```
mount -r -F hsfs /dev/dsk/c0t6d0s0 /cdrom
cd /mnt/Solaris_8/Product
pkgadd -d . SUNWlibC
```

Similarly, from Software CD 2/2,

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

From Documentary CD,

Pkgadd –d . SUNWdoc SUNWman

1.4 Installing Patches

Perform the following steps to install SUN patches:

1) Download the package to another machine, which is not currently networked with WEBSERV.

FTP to the vendor SUN's patch site:

```
ftp sunsolve.sun.com (anonymous/[user@domain.name])
cd pub/patches
get 8_Recommended.README
```

```
get 8_Recommended.zip
```

2) Transfer the package `8_Recommended.zip` to a CD or other removable media.
3) Copy the package to `/var/tmp` in WEBSERV from the media.
4) Unpack the patch cluster:

```
unzip –qq 8_Recommended.zip
```

Note:

–qq: unzip in a quiet mode, only report errors.

5) Install the package

```
cd 8_Recommended
./install_cluster –q –nosave
```

note: –q: install in a quiet mode.

--nosave: do not use this option if you intent to save an image of un-patched codes.

6) Remove patch cluster

```
cd /var/tmp
rm –rf /var/tmp/8_Recommended
```

7) `reboot`

2. Hardening OS

2.1Modifying Boot Services

The more open services are offered, the riskier a system will be. It is necessary to disable services that are not needed in WEBSERV.

a) Since the web service is the only service required in the WEBSERV system, we disable all services listed in `/etc/inetd.conf` by commenting out the lines starting with each service's name.

```
cd /etc
vi inetd.conf
```

The following is an example of commenting out `telnet` service.

Original:

```
telnet stream tcp6   nowait  root  /usr/sbin/in.telnetd  in.telnetd
```

Modified:

```
#telnet stream tcp6   nowait  root  /usr/sbin/in.telnetd  in.telnetd
```

Following the same commenting procedure for other services.

b) Preventing WEBSERV from serving or mounting file systems via NFS without administrator interventions [R9]. NFS is a big security hole.

This can be achieved by renaming NFS related links as follows.

```
cd /etc/rc2.d
for file in S73nfs.client S74autofs *cache*
do
  mv $file .NO$file
done
```

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

c) Disable RPC-based services [R9], which otherwise will also pose security risks.

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

d) Disabling Sendmail start-up script to thwart popular Sendmail attacks.

```
cd /etc/rc2.d
mv S88sendmail .NO88sendmail
```

2.2 Configure Kernel Parameters

1) Create new `/etc/init.d/netconfig` script to defend against various denial-of-service attacks targeted the TCP, IP and ARP layer

a) `vi /etc/init.d/netconfig`

b) Add the following lines in the file:

```
#!/sbin/sh
#Increase TCP connection queue value against SYN Flood attacks
ndd -set /dev/tcp tcp_conn_req_max_q0 8192
# Decrease TCP connection time-out value to 1 min against SYN Flood
ndd -set /dev/tcp tcp_ip_abort_cinterval 60000
# Disable unicast timestamp request, which is not essentially necessary
ndd -set /dev/tcp ip_respond_to_timestamp 0
# Disable timestamp broadcast requests(or attacks)
ndd -set /dev/tcp ip_respond_to_timestamp_broadcast 0
# Disable responding to address mask broadcast to reduce DoS risks
ndd -set /dev/tcp ip_respond_to_address_mask_broadcast 0
#Ignore ICMP redirect errors, which can be forged by DoS attackers
ndd -set /dev/tcp ip_ignore_redirect 1
# Disable the sending of redirect errors (only routers need to)
ndd -set /dev/tcp ip_send_redirects 0
# Disable Source Routed Forwarding to thwart source routing attacks
ndd -set /dev/tcp ip_forward_src_routed 0
# Disable direct broadcast to thwart SMURF alike DoS attacks
ndd -set /dev/tcp ip_forward_directed_broadcasts 0
# Disable IP forwarding to reduce network reach-ability from attackers
ndd -set /dev/tcp ip_forwarding 0
# Enable Strict Destination Multihoming to ignore packets sent to an
interface from which it did not arrive
ndd -set /dev/tcp ip_strict_dst_multihoming 1
# Thwart ARP attacks by reducing ARP table refreshing time to 1 minute
ndd -set /dev/arp arp_cleanup_interval 60000
#Thwart ARP attacks by reducing IP routing table refreshing time to 1 m
ndd -set /dev/ip ip_ire_arp_interval 60000
# Prevent machines from responding to pings sent to the LAN broadcast
address; purpose: prevent from being DoS attack (e.g.Smurf) amplifier
ndd -set /dev/ip ip_respond_to_echo_broadcast 0
```

Note: `ndd` is a command to examine or set TCP/IP drivers.

c) Save the file
d) Change ownership/permission
```
chown root:root /etc/init.d/netconfig
chmod 744 /etc/init.d/netconfig
```
e) Create link:
```
cd –s /etc/init.d/netconfig /etc/init.d/S69netconf
```

2) Edit /etc/system file to defend against buffer-overflow attacks, over-sized core files, and privileged port restriction for NFS clients [R9]:
```
* prevent stack-smashing attacks
set maxuprc = 128
* prevent the creation of a core file
set sys:coredumpsize = 0
* require NFS client requests to originate from privileged ports
set nfssrv:nfs_portmon = 1
```

3) Force system to use a better TCP sequence number generation algorithm to defend against man-in-the-middle alike attacks [R9].
Edit file `/etc/default/inetinit`, set `TCP_STRONG_ISS=2`.
4) `reboot`

2.3 Removing Files

a) Remove NFS-related configuration files so that administrator can know when NFS services are re-enabled [R9].
```
rm /etc/auto_* /etc/dfs/dfstab
```
b) Remove empty crontab files:
```
cd /var/spool/cron/crontabs
rm adm lp
```

2.4 File System Configuration

Edit `/etc/vfstab` file on the partition features to protect critical file systems:

a) Mount non-root ufs systems `nosuid` to prevent set-UID programs executing:
```
/dev/dsk/c0t0d0s5 /dev/rdsk/c0t0d0s5 /var ufs 1 no nosuid,logging
/dev/dsk/c0t0d0s4 /dev/rdsk/c0t0d0s4 /opt ufs 2 yes nosuid,logging
```
Note: This settings will prevent rogue set-UID programs from showing up in `/var` and `/opt` partitions. The logging option helps prevent file system inconsistencies that can slow or abort the system boot process.

b) Mount the root file system with the logging option
```
/dev/dsk/c0t3d0s0 /dev/rdsk/c0t3d0s0 / ufs 1 no remount,logging
```

c) [optional] Mount `/usr` read-only.
This operation is optional since it can be performed later on after additional packages are installed to the `/usr` partition.

```
/dev/dsk/c0t0d0s3 /dev/rdsk/c0t0d0s3 /usr ufs 1 no ro
```

Mounting `/use` read-only can protect the partition frin Trojan horse programs.

2.5 Logging

Logging is a very important step to trace potential intrusion attempts. Perform the following steps to harden the system logging function:
a) Add the following line to /etc/syslog.conf so that authentication related information will be logged into `the /var/log/authlog` file.

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

b) Confirm the existence of file `/var/log/authlog`, and set the access mode to root readable/writable only.

```
chmod 600 /var/log/authlog
```

c) Create `/var/adm/loginlog` to capture failed logins [R9]

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

d) Install the following log rotation script [R9] on /opt/bin:

```
#!/bin/ksh
# rotate: a script to roll over log files
# Usage: rotate /path/to/log/file [ mode [#revs] ]

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

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

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

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

```
      DEPTH=$OLD
done

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

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

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

e) Perform the following steps to retain logs every 4 weeks:

```
crontab –e root
```

Add the following lines to root's crontab:

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

f) Confirm the file `/etc/default/cron` to read

```
CRONLOG=YES
```

g) Edit `/etc/init.d/perf` and uncomment the lines that cause a marker to be placed in the system accounting logs when machine boots.

See Appendix A for the revised `/etc/init.d/perf` file.

h) Perform the following steps to configure crontab for user sys:

```
crontab –e sys
```

Edit the crontab file [R9] so that system accounting data will be captured every 20 minutes and daily reports written to /var/adm/sa. The data will be overwritten on a monthly cycle:

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

2.6. Enable Kernel Level Auditing BSM

Basic Security Module (BSM) is Sun's auditing function that can provide the administrator with a detailed report of all system activity.
Following the steps below to enable this function:
1) Enable BSM by running script `bsmconv`:

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

2) Configure the `/etc/security/audit_control` file as shown below, so that failed login-log-off, administrative actions will be audited. A warning will be generated when only 20% left for available logging space.

```
dir:/var/audit
flags:lo,ad,+fm,+fc
naflags:lo,ad
```

```
minfree:20
```

Note: The configuration above is not necessarily suitable for all circumstances; different configurations are desirable for different machines and services.

3) Configure the `/etc/security/audit_user`:
Because `/etc/security/audit-control` file has defined the events to be audited, comment out default configurations for root –

```
root:lo:no
```

4) Edit roots' crontab by executing command:

```
crontab –e root
```

Add the following line to force new audit log files to be started every hour:

```
0 * * * * /usr/sbin/audit –n
```

5) Set the `cnt` policy or set up an audit administration account
a) In the event of an audit trail overflow, the `cnt` policy must be enabled, which allows further system further functioning. The `cnt` policy ensure not to suspend process when audit resources are exhausted; instead, to drop audit records and keep a count of the number of records.

```
/etc/security/audit_starup:
auditcofig -setpolicy +cnt
```

b) If the `cnt` policy is not enabled, an account must be available that can work without further being audited. To setup such an account, follow the steps below:
In the `/etc/passwd`, add the following entry:

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

To add a corresponding entry into the /etc/shadow file:

```
pwconv
```

In the /etc/security/audit_user file, add the following entry to turn off auditing for the account:

```
audit:no:all
```

set the password for the new account using passwd:

```
passwd <password>
```

```
6) reboot
```

2.7 Authentication

Robust authentication is one important step in the procedure of securing a system. Password administration is among the most obvious and critical in defending against various attacks.

a) Turn on the EEPROM security functionality so that a password will be prompted before any PROM level commands are executed [R9]:

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

b) Edit file `/etc/default/login`:
Uncomment the `UMASK` line. Set `TIMEOUT` to 60 seconds.
Set `SYSLOG_FAILED_LOGINS` to 0 to log all failed login attempts.

c) To increase security level, turn on the Password Aging functionality by editing file `/etc/default/passwd`:

```
#ident      "@(#)passwd.dfl   1.3   92/07/14 SMI"
# Set the password aging threshold to maximum six weeks, warning
threshold four weeks. Password length must be eight characters minimum.
MAXWEEKS=6
MINWEEKS=4
WARNWEEKS=4
PASSLENGTH=8
```

2.8 Access Control

Access control on strict password administration on non-root users, removal of insecure channels (e.g. `.rhost, hosts.equiv`) against remote attacks, file permission restrictions, and adequate network security mechanisms through properly setting up nearby routers, are important steps in securing the WEBSERV system.

1) Make /dev/null the shell for non-root users in `/etc/passwd` using `passmgmt` command [R9].

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

2) Remove `.rhosts` support through authentication management (`pam.conf`) [R9]

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

3) Create empty files to thwart remote attacks [R9]:

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

4) Perform the following steps to allow only `root` to run `crontab` and `at` commands [R9].

```
cd /etc/cron.d
rm –f cron.deny at.deny
```

```
echo root > cron.allow
echo root > at.allow
chown root:root cron.allow at.allow
chmod 400 cron.allow at.allow
```

5) Install `fix-modes` script to set appropriate permissions on files, overwriting default permissions that could many files insecure.
FixModes is a script that tries to make Solaris Operating Environment file modes more secure. It does this by removing group and world write permissions of all files, devices, and directories listed in `/var/sadm/install/contents`, with the exception of those listed in `exceptions.h`
a) Download fix-mode package from the following web site to a removable media:
http://www.sun.com/solutions/blueprints/tools
Transfer the package to `/opt/local` in WEBSERV.
b) `cd /opt/local/FixModes`
c) Run the fix-mode shell script from the command line:

```
sh fix-modes
```

6) Provide adequate network security
a) Confirm that local network administrators have properly configured ingress/egress filtering in edge routers to block spoofed packets
The ingress filtering rules should at least reject packets from outside of the intranet domain to access WEBSERV, and reject packets from outside domains with intranet domain IP addresses (inward IP spoofing).
The egress filtering rules should disallow internal packets out of the intranet, which IP addresses appear to be out of the intranet boundary (outward IP spoofing).
b) Confirm that local network administrators have properly configured edge routers' parameters to alleviate `smurf` and other denial-of-service attacks, using available technologies
c) Confirm that local network administrators not to grand outside users out of company to access the WEBSERV in edge routers.

2.9. Statutory Warnings

1) Displaying warning messages to login users is highly recommended to claim crackers' liability for breaching into an unauthorized system.
Refer to Appendix B [R9] for a sample text for the file `/etc/issue` and `/etc/motd`.

2) Create banner messages for telnet users through editing file `/etc/default/telnetd`:
`BANNER="Authorized uses only. All access may be logged.\n"`
Note: `Telnet` service has been turned off in WEBSERV. This is for instruction purposes only.

3) Create banner messages for ftp users by editing file `/etc/default/ftpd`:

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

Note: `FTP` service has been turned off in WEBSERV. This is for instruction purposes only.

4) Setup access control on the banner files, so that only root can have write privilege on the login warning-message and root owns all the banner files [R9].

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

5) Set boot-level warning message [R9]

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

3. Hardening OS by Installing Third Party Software

3.1 Installing gcc

GNU-GCC is a commonly used compiler compatible for most third-party software installations.
Compiling a package through source codes is preferred when certain configuration parameters need to be set up manually. Installing binary code directly does not provide this kind of flexibility.

1) Download GCC package to a machine currently not connected to WEBSERV:

```
ftp mirrors.xmission.com
cd sunfreeware/sparc/8
get gcc-3.1-sol8-sparc-local
```

2) Transfer the package to a CD or other removable media
3) Copy the package from the media to `/opt/local` in WEBSERV
4) Unpack the package:

```
pkgadd -d gcc-3.1-sol8-sparc-local
```

5) (This is not a step to be executed immediately.)
After all add-on software compilation and configuration are completed, remove the GCC package. The purpose is to reduce the risk that crackers might be able to build malicious codes through this tool close at hand if WEBSERV is compromised.

3.2 Install PGP

PGP is an encryption tool that can be used to provide confidentiality. It can be used to check the integrity of software by verifying their signatures, which is a commonly adopted approach in software transfers from remote locations.

1) Download PGP to a machine currently not connected to WEBSERV.
Go to PGPi web site http://www.pgpi.org, download current PGP package PGPcmdln-6.5.8.SolPkg_FW.tar.gz

2) Copy the package to a CD or other removable media

3) Transfer the package to /opt/local/PGP6.5.8 in WEBSERV from the media.
Install PGP:

```
gunzip -c PGPcmdln-6.5.8.SolPkg_FW.tar.gz | tar -xvf -
pkgadd -d PGPcmfln-6.5.8.SolPkg_FW
```

4) Check installation:

```
pkginfo -l PGP
```

The status should be "completely installed"

3.3 Install TCP Wrappers

TCP Wrappers provides a cross-platform uniform method for logging connections to various system services, service access, and a means of booby-trapping various network services.

1) Obtain TCP wrappers source code and save to a CD or other removable media:

```
ftp ftp.porcupine.org
cd pub/security
get tcp_wrappers_7.6-ipv6.1.tar.gz
```

2) Transfer the package to /opt/local in WEBSERV
Unpack the package:

```
cd /opt/local
gunzip -c tcp_wrappers_7.6-ipv6.1.tar.gz | tar xf -
cd tcp_wrappers_7.6-ipv6.1
```

3) Edit Makefile

```
vi Makefile
```

Make the following modifications in the Makefile:

a) REAL_DAEMON_DIR = /usr/sbin

b) set CC=gcc for target sunos5

c) Uncomment TLI = -DTLI
-- Turning on TLI (transport-level interface) support

d) set RANLIB = echo
-- Setting object that ranlib will be run on

e) set LIBS = -lsocket –lnsl
-- Specifying the networking libraries

f) set BUGS = -DGETPEERNAME_BUG –DBROKEN_FGETS –DSOLARIS_24_GETHOSTBYNAME_BUG
-- Working around system bugs

g) Uncomment NETGROUP= -DNETGROUP
-- Turning on netgroup host access-control
h) Uncomment STYLE = -DPROCESS_OPTIONS
-- Turning on language extensions
i) set FACILITY = LOG_AUTH
-- Changing the default disposition of the logfile records
j) Uncomment DOT = -DAPPEND_DOT
-- Reducing DNS load

4) Build the TCP wrappers:

```
make sunos5
```

5) Install files [R9] to /usr/local:

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

6) Dry-run the following programs:
a) Run `tcpdchk` to identify the most common problems in the wrapper and inetd configuration files.

```
tcpchk
```

A blank output means that the TCP wrapper is installed properly.
b) Run `tcpdmatch` to examine how the wrapper would react to specific requests for service. Command line format:

```
tcpdmatch [-d] [-i inet_conf] daemon[@host] user[@host]
```

where, -d: use allow/deny files in current directory
-i: location of inetd.conf file
c) Test `safe_finger` command
See the following sample command,

```
safe_finger 10.0.0.2
```

which generated the following output:

```
Login name: 10.0.0.2   In real life: ???
```

7) Edit the `/etc/inetd.conf` file to insert the TCP wrapper daemon `tcpd`. Sample partial code for the `inetd.conf` with `tcpd` inserted is attached in Appendix C.
(Note, even though current services in inetd.conf have been disabled, this approach is still recommended for possible future usages).

8) Set up allow and deny list by editing `/etc/host.allow` and `/etc/host.deny` files. Detailed code will not be demonstrated here for confidentiality purpose. The template could look like:

```
/etc/hosts.allow:
ALL:<net1>/<mask1>, ..., <netN>/<maskN>
/etc/hosts.deny:
ALL: ALL: /usr/bin/mailx -s "%s: connection attempt from %a" \
AlertUser@domain.name
```

Here is how the filtering rule works with the hosts.allow|hosts.deny:
If an accessing host ∈ {hosts.allow}, then accept.
else if the host ∈ {hosts.deny}, then reject.
else accept.

9) Set permissions on allow and deny files:

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

10) Send a 'kill –HUP" to the `inetd` process to make the change effective.

3.4 Installing OpenSSL

SSL provides privacy and data integrity between two communication applications, mainly on top of layer 4. Installation of OpenSSL is for the preparation of building OpenSSH, which can effectively eliminate eavesdropping, connection hijacking, and other network-level attacks.

1) Confirm the existence of SUNWzlib package.
Note:
If it does not exist, follow the steps below to add the package:
a) Insert Solaris 8 Software CD 2/2
b) `mount –r –F hsfs /dev/dsk/c0t6d0s0 /cdrom`
c) `cd /mnt/Solaris_8/Product`
`pkgadd –d . SUNWzlib`

2) Installing OpenSSL
a) Download OpenSSL to a machine not currently connected to WEBSERV:

```
ftp mirrors.xmission.com
cd sunfreeware/sparc/8
get openssl_0.9.6.d-5018-sparc-local.gz
```

Transfer the file to a removable media, such as a CD , disk, or tape.
b) Transfer the package to /opt/local in WEBSERV from the media.
Unpack the gzip file:
`gunzip openssl_0.9.6.d-5018-sparc-local.gz`
c) Install the OpenSSL package:

```
pkgadd -d openssl_0.9.6.d-5018-sparc-local
```

3.5.Installing and Configuring OpenSSH

1) Installing OpenSSH

a) Download OpenSSH to a machine currently not connected with WEBSERV, from web site http://www.openssh.com; select the portable version openssh-3.4p1

b) Burn the package to a CD or other media

c) Copy the package to `/opt/local` in WEBSERV from the media.

Unpack the package:

```
gunzip -c openssh-3.4p1.tar.gz | tar -xvf -
cd openssh-3.4p1
```

d) Set the environment:

```
CFLAGS="-I/usr/local/include"; export CFLAGS
LDFLAGS="-L/usr/local/lib"; export LDFLAGS
```

e) Build the OpenSSH:

```
./configure -prefix=/usr/local -with-tcp-wrappers \
     -with-ssl-dir=/usr/local/ssl/lib
make
```

Note: options in configure –

`prefix`: indicate installation location (/<prefix>/{bin,etc,lib,sbin})

`with-tcp-wrappers`: enable TCP Wrappers support (/etc/hosts.allow|deny)

`with-ssl-dir`: allows to specify where the openSSH libraries are installed

f) Install OpenSSH:

```
make install
```

2) Configuring OpenSSH

a) Create the /etc/sshd_config file for the SSH server. See a sample configuration file in Appendix D (to reduce security risks, only protocol 2 is chosen).

b) Set appropriate file permissions on configuration files

```
cd /etc
chown root:root sshd_config
chmod 600 sshd_config
```

c) Create an /etc/init.d/sshd script for starting the SSH server at boot time. A sample is attached in Appendix E [R9].

d) Create links for sshd startup script in /etc/rc2.d to that in /etc/init.d, which starts running right after syslog has been activated and can receiving logging messages:

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

3) Start SSH daemon

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

3.6. Installing Tripwire

The purpose of installing Tripwire is for file integrity protection. Tripwire is a utility tool that compares a designated set of files and directories against information stored in a previously generated database. When run against system files on a regular basis, any changes in critical system files will be visible, and appropriate damage control measures can be taken.

1) Obtaining Tripwire software:
Tripwire commercial version can be purchased through www.tripwire.com.
Or, download a trial version or academic version from http://www.tripwiresecurity.com.
Save the package to a CD or other removable media.

2) Transfer the package to /opt/local in WEBSERV: /opt/local/tw_ASR_1.3.1_src.
cd /opt/local/tw_ASR_1.3.1_src

3) Modify Makefile:
```
INSTALL= /usr/ucb/install
HOSTNAME= "uname -n"
```

4) Modify /opt/local/tw_ASR_1.3.1_src./include/config.h file:
Confirm the following statement is included in the ./include/config.h file:
```
#include "../configs/conf-svr4.h"
```
Confirm the definition of CONFIG_PATH and DATABASE_PATH:
```
#define CONFIG_PATH     "usr/local/bin/tw"
#define DATABSE_PATH    "var/tripwire"
```
(Or: redefine the two path by your choice)
Confirm the definition of CONFIG_FILE and DATABASE_FILE:
```
#define CONFIG_FILE     "tw.config"
#define DATABSE_FILE    tw.db @
```
(Or: redefine the two file names by your choice)

5) Customize ./config/tw.conf based on your needs for file protection.
```
cp ./configs/tw.conf.sunos5 /usr/local/bin/tw/tw.config
```

6) Perform the following steps to configure and install the package:
```
make
mkdir /usr/man/man8
make install
make clean
make clobber
```

7) Test Tripwire:
```
make test
```

8) Create initialization database:
```
tripwire -initilize
```
This will create a datafile file ./database/tw.db_WEBSERV
```
mv ./database/tw.db_WENSERV /var/tripwire
```

9) Make a copy of the tw.db_WEBSERV to a removable media.
Here is an example of copying the initial database to a tape:
```
tar cvf /dev/rmt/0 ./database
```

3.7.Installing NMAP

NMAP is a network exploration tool and security scanner that can be used to detect open ports in a system. It can be used to check whether unnecessary services/ports have been properly closed.

1) Download NMAP to a machine to be used to scan WEBSERV
```
ftp ftp.sunfreeware.com
cd pub/freeware/sparc/8
get nmap-2.54BETA28-sol8-sparc-local.gz
```

2) Install NMAP
```
pkgadd -d nmap-2.54BETA28-sol8-sparc-local.gz
```

4. Installing and Hardening Apache Web Server

Apache Web Server is the major service provided by the WEBSERV system, which must be ensured to be free of security vulnerabilities.

4.1Installing "patch"

Newer version of "`patch`" is required for the Apache Server installation.
1) Obtaining `patch` package:
Go to the website http://www.sunfreeware.com and download `patch` package for Solaris 8: patch-2.5.4-sol8-sparc-local.gz to a removable media.
Transfer the package to /opt/local in WEBSERV.
```
2) cd /opt/local
3) gunzip patch-2.5.4-sol8-sparc-local.gz
4) pkgadd -d patch-2.5.4-sol8-sparc-local
```
5) New patch version is installed in `/usr/local/bin`

4.2 Installing Apache SSL Server

1) Download Apache from http://www.apache.org, current version: apache_1.3.26.tar.gz, to a removable media.
Transfer the package to /opt/local in WEBSERV.

Unpack the package:

```
cd /opt/local
gunzip -c apache_1.3.26.tar.gz | tar -xvf -
```

2) Apply patches and add SSL

a) Obtaining apache secure server

Go to website http://www.apache.org

Download `apache_1.3.26+ssl_1.48.tar.gz` to a removable media.

Transfer the package to `/opt/local/apache_1.3.26 in WEBSERV`

b) `cd /opt/local/apache_1.3.26`

c) Unpack the package:

gunzip –c apache_1.3.26+ssl_1.48.tar.gz | tar –xvf –

d) Modify `./FixPath` file so that the right `patch` version is used:

/usr/local/bin/patch

Run `./FixPath` to fix up the OpenSSL paths.

e) Proceed with the following Apache configuration:

```
./configure -prefix=/opt/apacheSSL \
    -enable-module=auth_dbm \
    -enable-module=rewrite \
    -enable-module=usertrack \
    -enable-module=vhost_alias \
    -disable-module=info \
    -disable-module=status \
    -disable-module=autoindex \
    -disable-module=userdir
```

The meaning of modules in configure options:

`auth_dbm`: allows for basic HHTP authentication and stores the username/password pairs in a DMB type file.

`rewrite`: controls URL redirection and manipulation

`usertrack`: helps track users who make malicious requests

`vhost_alias`: provides support for dynamically configured mass virtual hosting

`info`: gives server information

`status`: gives server information

`autoindex`: creates directory listing information when an index page is missing

`userdir`: allows user directories within the web site

f) Hiding Server Identity

Modify the default HHTP Response Header parameter for the web server by modifying `/opt/local/apache_1.3.26/src/include/httpd.h`:

```
#define SERVER_BASEPRODUCT  "My server"
#define SERVER_BASEREVISION "hidden"
```

g) Build the server:

```
make
```

```
make install
```

3) Configure WEBSERV

Edit `/opt/apacheSSL/conf/httpsd.conf` to configure the SSL WEBSERV:

```
Port 443
Listen 443
SSLCertificateKeyFile /opt/local/ssl/certs/cert.cert
SSLCertificateFile /opt/local/ssl/certs/cert.key
SSLCACertificateFile /opt/local/ssl/certs/certCA.key
SSLEnable
SSLCacheServerPath /opt/apacheSSL/bin/gcache
SSLCacheServerPort 8889
SSLCacheServerRunDir /tmp
SSLSessionCacheTimeout 300
```

4.3. Securing Apache Server

1) Access Control

a) Create user/group [R12] for the web server

```
groupadd webadmin
groupadd webdev
groupadd webserv
useradd –d "/opt/apacheSSL/htdocs" –g webserv –c "Web Server" –m webusr
```

Create a quota for the web account `webusr` so that new files will not be allowed to be created:

```
touch /opt/quotas
edquota webusr
```

Change the quota settings to `hard=1`.

Lock down the account `webusr` so that the account can not be logged in:

```
Usermod –s .bin/false webusr
```

Edit the user&group parameters within the `/opt/apacheSSL/conf/httpd.conf` file so that the HTTP requests will not be executed with `root` privileges:

```
User webusr
Group webserv
```

Change Ownership/permissions on Directories and files:

```
chgrp webadmin /opt/apacheSSL/conf
chmod –R 770 /opt/apacheSSL/conf
chgrp webdev /opt/apacheSSL/htdocs
chmod –R 775 /opt/apacheSSL/htdocs
chgrp webdev /opt/apacheSSL/cgi-bin
chmod –R 775 /opt/apacheSSL/cgi-bin
chgrp webadmin /opt/apacheSSL/logs
chmod –R 770 /opt/apacheSSL/logs
chgrp webadmin /opt/apacheSSL/bin
chmod –R 770 /opt/apacheSSL/bin
```

b) Permissions on server root Directories.

Make sure critical server root directories are write-able only by root:

```
cd /opt/apacheSSL
chgrp 0 . bin conf logs
cd ./bin
chgrp 0 httpsd
chmod 511 httpsd
```

Confirm that the above files and directories are owned by root. Allowed access right to `/opt/apacheSSL`, and `bin | conf |logs` files in those directories are 755.

c) Protecting Server Files:

Confirm the following code in `/opt/apacheSSL/conf/httpsd.conf`, so as to stop users from setting up `.htaccess` files which can override security features configured:

```
<Directory />
 AllowOverride None
</Direcroty>
```

To prevent clients from walking through the entire file system [R8], add the following block to the `/opt/apacheSSL/conf/httpsd.conf`:

```
<Directory />
   Order Deny,Allow
   Deny from all
</Directory>
```

For the same reason, add the following line:

```
UserDir disabled root
```

d) Disallow executing system commands and following symbolic links.

Add the following options in `httpsd.conf` file:

```
<Directory "/opt/apacheSSL/htdocs">
  …
  Options IncludesNoexec SymLinksIfOwnerMatch
  …
</Directory>
```

e) Disallow access to sensitive files

Add the following lines to httpsd.conf:

```
<Directory "/opt/apacheSSL/cgi-bin">
<FilesMatch (guestbook\.cgi|rwwwshell\.cgi)>
  order allow,deny
  deny from all
</FilesMatch>
</Directory>
```

Note: file names above are for example only. Do adjust based on individual server needs.

2) Audit Logs

Edit `/opt/apacheSSL/conf/httpsd.conf` file:
a) Define the verbosity of information in the error log:
```
LogLevel notice
```
b) Log access attempts
```
CustomLog /opt/apacheSSL/logs/httpsd_access_log combined
```
c) Log Error messages
```
ErrorLog /opt/apacheSSL/logs/httpsd_error_log
```

3) Misc

a) Hiding server identity [R12]
Editing `httpsd.conf` file:
```
ServerSignature Off
```

b) Preventing Denial of Service
Modify default values in the httpsd.conf file to thwart DoS attacks:
```
Timeout 60
StartServers 10
MinSpareServers 10
MaxSpareServers 20
```

c) Preventing Buffer Overflow
Add the following lines in the httpsd.conf file to prevent from buffer overflow of HTTP requests:
```
LimitRequestBody 51200
LimitRequestFields 40
LimitRequestFieldsize 1000
LimitRequestline 4000
```

d) Signature based Intrusion detection
Utilizing Mod-Rewrite feature [R12] to identify suspicious HTTP requests that contains illegal characters, which will be redirected to `403.cgi`.
Add the following lines in httpsd.conf:
```
RewriteEngine On
RewriteLog /opt/apacheSSL/logs/ids.log
RewriteLogLevel 9
RewriteRule
\.\.|\.\.\.|\+|\/\/|\[|\]|\#|\%|\^|\&|\(|\)|\{|\}|\;|\,|<|<<||\||\!|\`|
>|>>|\@ - [F]
RewriteRule
/\.[a-z0-
9|/bin|/sbin|/etc|httpd\.conf|httpsd_access_log|httpsd_error_log - [F]
```

e) Error Handling
Editing error pages to non-default appearance.
In httpsd.conf:

```
ErrorDocument 401 /opt/apacheSSL/cgi-bin/401.cgi
```
The same goes to other error pages such as 400, 403, 405, 413, 414, 500, 501.

5. Backup

Backup procedure is mandatory for disaster recovery. If a security breach is suspected in WEBSERV, backups can be used to compare potentially damaged OS files or mission critical applications against existing clean version.
Tape is an ideal storage medium because it is capable of storing high capacities of information for a relatively low cost. Tape is also perfect for archival because cartridges can be stored off-site for enhanced data security.
Follow the steps below to perform necessary back-ups:
1) Shutdown the WEBSERV system
2) Turn off power
3) Connect tape driver to the WEBSERV host; Connect tape driver power cable; set the jumper to 4
4) Turn on tape driver power; insert a tape into the driver
5) Turn on WEBSERV system
6) Login as root
7) Modify `/kernel/drv/st.conf` (as root):

```
cd /kernel/drv/st.conf
cp st.conf st.conf.old
```

a) Uncomment line:

```
Tape-config-list=
```

b)Directly below the following existing line:

```
# "HP     C1539A",    "HP DDS-2 4mm DAT",     "HP_DAT",
```

Add the following line:

```
"HP      C1537A",    "HP DDS3 4mm DAT",      "HP_DAT";
```

(note: there are six spaces between "HP" and "C1537A")

c) Directly below the following existing line:

```
#HP_DAT   =     1,0x34,0,0x19679,1,0x0,0;
```

Add the following line:

```
HP_DAT    =     1,0x34,0,0x9639,4,0x00,0x8c,0x8c,0x8c,1;
```

d) Save the file

```
8) halt
9) boot -r
```

10) Check tape drive status:

```
mt -f /dev/rmt/0 status
```

Execute the commands again till

```
sense key(0x6)= no additional sense
```

11) Back up all ufs file systems to the tape (tape1):

```
mt -f /dev/rmt/0 rewind
for dir in / /usr /var /opt
do
```

```
        ufsdump 0f /dev/rmt/0n $dir
    done
    mt -f /dev/rmt/0 rewoffl
```

Repeat the steps above to back up to tape2.

12) Write protect tape1 and tape2

13) Store tape1 locally and tape2 off-site; Both tapes are stored in a secure place only accessible by one person (in this case, me)

6. Physical Security

Physical security is one issue that can be easily ignored but very important. No matter how many secure software and setups are configured, it can be all in vain if the machine is put in an open guest room.

The following approaches have been used to protect our WEBSERV system.
1) Place the WEBSERV host in a locked room with access controlled by one person (i.e. me in this case)
2) Provide temperature and humidity controls to avoid damages to the system.
3) Connect a UPS with warning system when the power in the UPS is about to be exhausted.
4) Ensure the layout of the machine in the room so that the keyboard is absolutely not accessible from windows or other vantage points.

Ongoing Maintenance

After setting up the protecting shield, we must be persistent with ongoing maintenance procedures and rules.
Logging and auditing procedures monitor the system's overall health. Integrity checking plays critical rule in spotting suspicious changes in file systems to detect potential intrusions. Backup is a conventional routine in disaster discovery either by attacks or accidents. Continues patching and updating OS and applications are also a must to protect from "new" vulnerabilities.

1. Logging and Auditing

1.1.Logging
Logging activities without checking on them on regular basis is just like to set up an alarm system for deaf people. The following procedures should be followed.

1) Setup log checking type and schedule
Critical alert should be report to the administrator through e-mail, such as the connection attempt logged by the Tcpwrapper.

User `mailx` to check mails for root for Tcpwrapper alert.
Common warning logging data for failed operations should be checked at daily basis.

2) Log files should be checked regularly:
`/var/log/authlog`:
-- system events including login attempts, failed `su` attempts, reboots
`/var/adm/loginlog`:
-- failed logs
`/var/adm/messages`:
-- system warning messages
`/var/adm/sulog`:
-- logs each time the `su` command is used to change the users' privileges to those of another
`/var/cron/log`:
-- cron log
`/var/adm/sa`:
-- system account data daily reports
Note: You might want to use log processing tools to check logs regularly for unusual activities (e.g. LogSentry in the TriSentry suite by Psionic.)

3) Use user commands to check system status logged regularly or on-demand
`last`
-- display login and logout information about users and terminals
`logins`
-- see the names of those who have logins on your system

4) Cleaning/rotating/saving old log files regularly
Use either script to rotate log files periodically (see section 2.4 in the above "Step by Step" part), or manually remove all files when designated file system being eaten up. If necessary, compress log files and store into tapes.

1.2.Auditing

Auditing enables an administrator to detect potential security breaches. It shall be performed in daily basis.
1) Audit schedule
The BSM provided by SUN is the major auditing tool used for WEBSERV.
Audit is scheduled by setting proper configurations in the `cron` file as discussed in `StepbyStep` section 2.5, where an audit report is generated every hour.

2) Dedicated partition for audit data
Ideally, it is recommended to dedicate a partition for the audit data with sufficient space.

In WEBSERV, audit data is stored in `/var` partition, specifically to the directory `/var/audit`.

3) Monitoring auditing reports
a) To monitor the audit trail:

```
praudit <logdatafile>
```

Note: <logdatafile> is a audit log file generated by BSM.
b) Due to the size of the auditing reports, extracted reports can be generated targeting specific features an administration is particularly interested.
The following gives an example that login-logout audit report is extracted, from all audit reports in defulat directory `/etc/security/audit/localhost/files`, and saved in `/var/audit/auditsummary` directory with a file name which suffix is "logins"

```
mkdir /var/audit/auditsummary
cd /var/audit/auditsummary
auditreduce -c lo -O logins
praudit <yyyymmddhhmm1,yyyymmddhhmm2>.logins
```

4) Reduce or compress, and store auditing data
Auditing data can be reduced to smaller compact reports by using the command:

```
auditreduce <options> <files>
```

As an example, the following commands compress the daily report on August 8, 2002 and saved in `/etc/security/summary.dir`:

```
auditreduce -O daily.summary -b 20020808 -c lo
compress *daily.summary
mv *daily.summary /etc/security/summary.dir
```

Compressed auditing data will be stored in tapes in regular basis through scheduled backup.

2. Integrity Checking

Regular integrity checking is important to detect potential intrusions on file systems.

1) Setup integrity checking schedule
Keep database up-to-date every 2-3 days.
In our case, the database is checked or/and updated every Tuesday and Friday at 5:30pm.
Note: specific schedules should be determined by individual needs based on the type of existing services and applications.

2) Perform regular integrity checking
On scheduled date, run Tripwire in Integrity Checking mode:

```
tripwire
```

Or, run tripwire in Interactive mode:

```
tripwire –interactive
```

Note: running tripwire in Interactive mode can prompt whether you want to update the database.

3) Tripwire database update on-need:

If a single file or a set of files has changed purposely, the database can be updated explicitly.

```
tripwire –update [dir/filename]
```

3. Backup Routine

Persistent and adequate backup can avoid severe damages due to accidental or intentional causes.

1) Determining the types of files to be backed up and back up size

Major file systems will be backed up periodically. Using one 2-24GB tape, all critical file systems, including sensitive data files, can be backed up considering the size of WEBSERV hard disk, with a full, incremental, or differential backup mechanism.

2) Determining the backup schedule

A modified grandfather-father-son (GFS) backup scheme is chosen for backup and tape rotation schedule.

a) Obtain 14 tapes and label them as follows:

4 daily tapes (sons) labeled “Monday” through “Thursday”;
4 weekly tapes (fathers) labeled “week1” through “week4”;
6 monthly tapes (grandfathers) labeled with month and year.

b) Beginning on a Friday (6:00pm), perform a full backup on the “week1” tape. Store the tape off site (or on site of your choice):

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

c) Beginning on the following Monday, perform a differential backup (or incremental backup of your choice) on the “Monday” through “Thursday” tapes. Store the tapes on site.

```
mt –f /dev/rmt/0 rewind
for dir in / /usr /var /opt
do
   ufsdump 1f /dev/rmt/0n $dir
done
```

```
mt -f /dev/rmt/0 rewoffl
```

d) On Friday, perform another full backup on the "week2" tape.

e) Continue with this rotation schedule until the last business day of the month No matter what day of the week it is, perform a full backup on the first month (grandfather) tape. Label the current date and store it off site.

3) Determining the tape retirement schedule

Based on the tapes' rated service live to determine a schedule for retirement. For the DDS3 tape drive connected to our WEBSERV, a tape should retire when the attention light in the tape drive continuously flashes amber even though the drive head has just been cleaned.

4. Routine Patching and Updating

Routinely updating OS, server and third party software is an important step to ensure the WEBSERV robust.

1) Patching OS

Check and patch the Solaris operating system regularly:
ftp://sunsolve.sun.com/pub/patches
Subscribe SUN security mailing list so that the most update patching information will be informed when they are available.

2) Patching Apache server

Check and download apache sites for patches regularly:
http://www.apache.org/dist/httpd/patches/
and http://www.apache.org/dist/httpd/contrib/patches/
Also check announcement and other important updates periodically:
http://httpd.apache.org/
Subscribe Apache security mailing list so that the patching information will be updated and informed promptly.

3) Third party application updates and patching

All third party software installed should be updated and patched regularly by checking relevant home pages.

Check Configuration

1. Operating System and Administrative Practices

1.1. Network Connection

Use NMAP tool to confirm all unnecessary services have been closed.
The following is an nmap screen output:

```
nmap 10.0.0.1
```

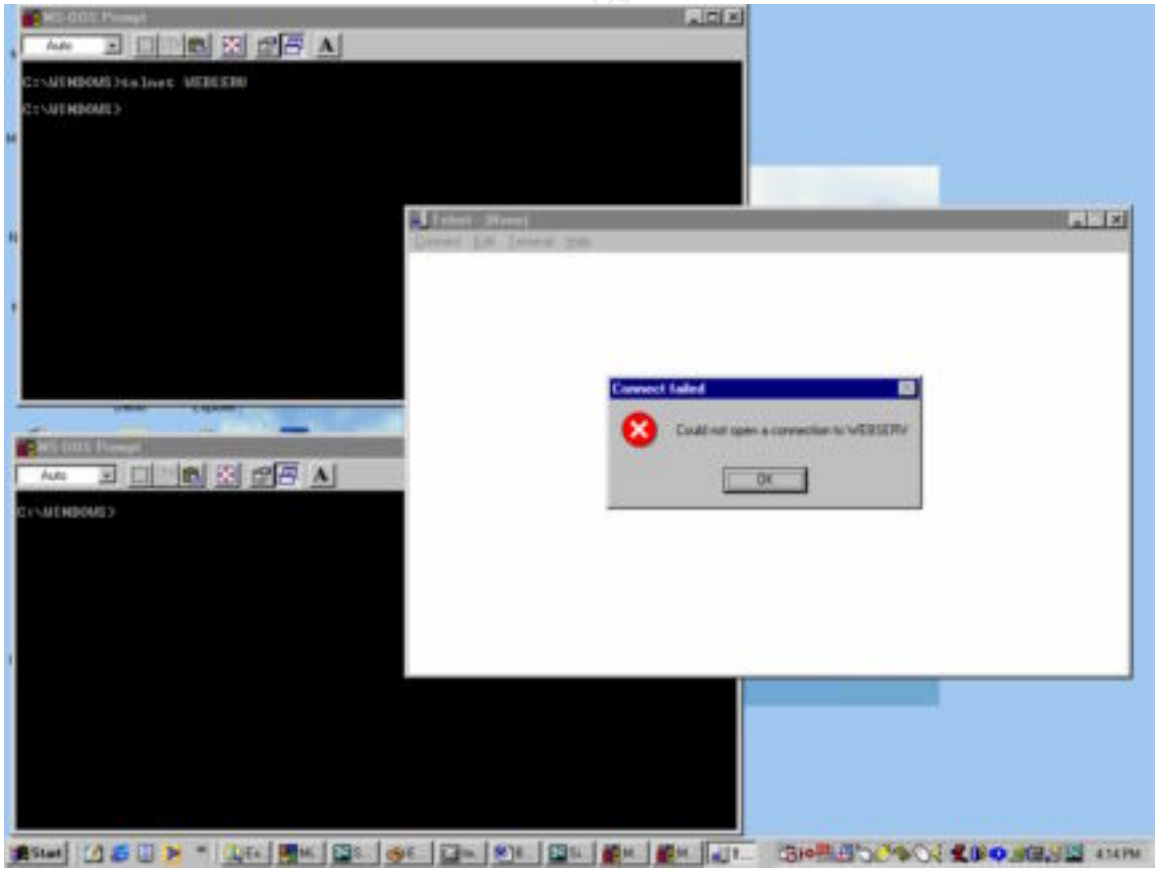
```
Starting nmap V. 2.54BETA28 ( www.insecure.org/nmap/ )
Interesting ports on (10.0.0.1):
(The 1532 ports scanned but not shown below are in state: closed)
Port       State       Service
22/tcp     open        ssh
80/tcp     open        sun-answerbook

Nmap run completed -- 1 IP address (1 host up) scanned in 1 second
```

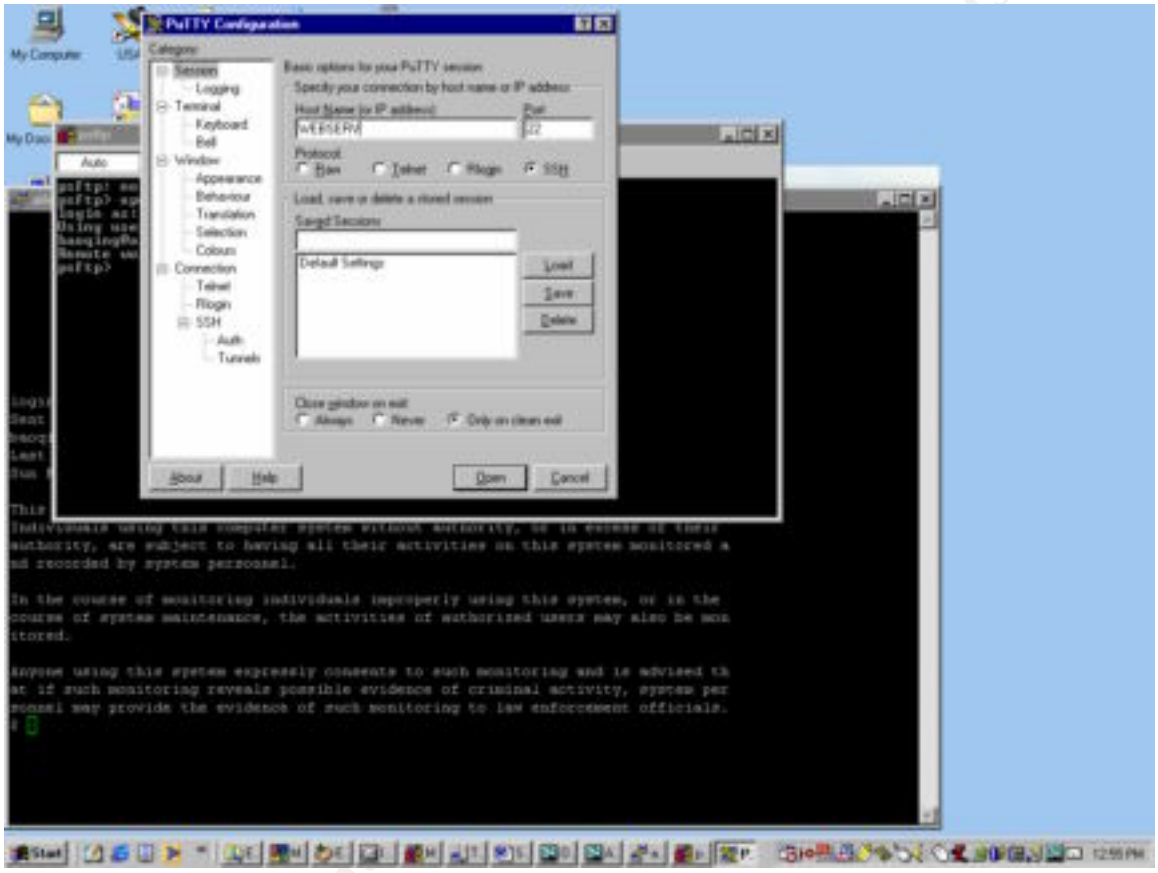
The output confirms that only necessary services (ssh, http) for WEBSERV are currently open.

1.2. Authentication

Clear-text password transmission through telnet, ftp, rlogin, rsh has been disabled. The following is a screen output of a failed `telnet` attempt to WEBSERV.



User authentication through passwords transmission is encrypted using ssh.
See the following ssh output generated through a PuTTy ssh client connection from a Windows platform to WEBSERV.



Security banners are displayed when users log in to the WEBSERV:

```
login as: user1
Sent username "user1"
User1@WEBSERV's password:
Last login: Tue Sep 17 16:24:35 2002 from 10.0.0.2

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

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

Anyone using this system expressly consents to such monitoring and is
advised that if such monitoring reveals possible evidence of criminal
```

```
activity, system personnel may provide the evidence of such monitoring
to law enforcement officials.
$
```

1.3.Logging and Audit

a) Checking logs
File `/var/log/authlog` provides authentication information, useful for debugging and detecting malicious accesses when suspicions arise.
The following is a sample output of the `authlog` file:

```
Aug 28 12:49:15 WEBSERV sshd[4322]: [ID 800047 auth.info] Accepted
password for
 User1 from 10.0.0.2 port 1289 ssh2
Aug 28 12:49:15 WEBSERV sshd[4322]: [ID 800047 auth.info] subsystem
request for sftp
Aug 28 13:41:59 WEBSERV sshd[3912]: [ID 800047 auth.info] Received
SIGHUP; restarting.
Aug 28 13:49:39 WEBSERV sshd[4374]: [ID 800047 auth.info] Server
listening on 0.0.0.0 port 22.
Aug 28 13:50:38 WEBSERV sshd[4376]: [ID 800047 auth.info] Accepted
password for user1 from 10.0.0.2 port 1358 ssh2
Aug 28 13:50:38 WEBSERV sshd[4376]: [ID 800047 auth.info] subsystem
request for sftp
Aug 28 13:50:38 WEBSERV sshd[4376]: [ID 800047 auth.error] error:
subsystem: cannot stat sftp-server: No such file or directory
Aug 28 14:36:22 WEBSERV sshd[4374]: [ID 800047 auth.info] Received
SIGHUP; restarting.
Aug 28 14:36:22 WEBSERV sshd[4394]: [ID 800047 auth.info] Server
listening on 0.0.0.0 port 22.
Aug 28 15:25:51 WEBSERV in.rshd[4552]: [ID 947420 auth.warning] refused
connect from localhost
Aug 28 15:25:51 WEBSERV sshd[4560]: [ID 947420 auth.warning] refused
connect from localhost
Aug 29 11:47:24 WEBSERV sshd[5334]: [ID 800047 auth.info] Accepted
password for user1 from 10.0.0.2 port 1250 ssh2
Aug 29 11:47:25 WEBSERV sshd[5334]: [ID 800047 auth.info] subsystem
request for sftp
```

File /var/log/syslog contains system operation logs, which is also useful in debugging system health.
File /var/adm/sa/sar* contains system usage information. A sample output is included in Appendix F.

b) Checking audit results
BSM creates audit results continuously.
Use the merging and reducing tool `auditreduce` to generate targeted information.

e.g. the command below to extract administrative actions performed in 08/15/2002:

```
auditreduce -d 20020815 -c ad -O admins
praudit 20020815040000.20020815192000.admins
```

The following shows the last several records generated by the above commands:

```
header,85,2,sysinfo(2),,Thu 15 Aug 2002 03:01:55 PM EDT, + 300001897
msec
argument,1,0x202,cmd
subject,root,root,other,root,other,7441,485,0 0 WEBSERV
return,success,85
header,85,2,sysinfo(2),,Thu 15 Aug 2002 03:01:55 PM EDT, + 300001897
msec
argument,1,0x202,cmd
subject,root,root,other,root,other,7441,485,0 0 WEBSERV
return,success,85
header,85,2,sysinfo(2),,Thu 15 Aug 2002 03:01:55 PM EDT, + 400000965
msec
argument,1,0x202,cmd
subject,root,root,other,root,other,7442,485,0 0 WEBSERV
return,success,85
header,85,2,sysinfo(2),,Thu 15 Aug 2002 03:01:55 PM EDT, + 400000965
msec
argument,1,0x202,cmd
subject,root,root,other,root,other,7442,485,0 0 WEBSERV
return,success,85
header,73,2,getaudit_addr(2),,Thu 15 Aug 2002 03:20:00 PM EDT, +
420004753 msec
subject,sys,root,sys,root,sys,7463,294,0 0 0.0.0.0
return,success,0
header,95,2,cron-invoke,,Thu 15 Aug 2002 03:20:00 PM EDT, + 430004001
msec
subject,sys,sys,sys,sys,sys,7463,294,0 0 0.0.0.0
text,crontab-job
text,/usr/lib/sa/sa1
return,success,0
```

1.4.Integrity Checking

Tripwire is used to check WEBSERV file system integrity.
The following is an output after running the following command:

```
Tripwire
```

```
# tripwire
Tripwire(tm) ASR (Academic Source Release) 1.3.1
File Integrity Assessment Software
(c) 1992, Purdue Research Foundation, (c) 1997, 1999 Tripwire
Security Systems, Inc. All Rights Reserved. Use Restricted to
Authorized Licensees.
### Phase 1:   Reading configuration file
```

```
### Phase 2:   Generating file list
tripwire: /.cshrc: No such file or directory
### Phase 3:   Creating file information database
### Phase 4:   Searching for inconsistencies
###
###                    Total files scanned:           36233
###                          Files added:             0
###                          Files deleted:           0
###                          Files changed:           8
###
###                    Total file violations:         8
###
changed: -rw-r--r-- root          198 Aug  7 15:25:28 2002 /.profile
changed: -rw-r--r-- root            0 Aug  9 16:06:38 2002
/etc/dfs/sharetab
changed: -rw-r--r-- root         7338 Aug 15 14:37:42 2002
/etc/inet/inetd.conf
changed: -rw-rw---- root           54 Aug 15 19:00:01 2002
/etc/security/audit_data
changed: -rw-r--r-- root         2303 Jan  5 18:58:37 2000 /etc/ttydefs
changed: -rw-rw-r-- root            0 Jul 11 18:26:50 2002
/etc/dumpdates
changed: -r-------- root          533 Aug 15 17:21:06 2002 /etc/shadow
changed: -rw-r--r-- root            0 Jul 12 17:08:24 2002 /etc/rmtab
### Phase 5:   Generating observed/expected pairs for changed files
###
### Attr        Observed (what it is)         Expected (what it should
be)
### ===================================================================
/.profile
       st_size: 198                           182
      st_mtime: Wed Aug  7 15:25:28 2002      Mon Jul 22 14:16:39 2002
      st_ctime: Wed Aug  7 15:25:28 2002      Mon Jul 22 14:16:39 2002
    md5 (sig1): 2eKoYsjz0XW7QBEv7sclZ4        1yb1Jr.6:eXm5SImck9ks1
 snefru (sig2): 12SUxjXSxgeCrMtD:BrsLp        1Tj7EXqXRC0OHzdj8VEGVf

/etc/dfs/sharetab
      st_mtime: Fri Aug  9 16:06:38 2002      Fri Jul 19 17:13:32 2002
      st_ctime: Fri Aug  9 16:06:38 2002      Fri Jul 19 17:13:32 2002

/etc/inet/inetd.conf
       st_size: 7338                          7335
      st_mtime: Thu Aug 15 14:37:42 2002      Fri Jul 19 17:28:27 2002
      st_ctime: Thu Aug 15 14:37:42 2002      Fri Jul 19 17:28:27 2002
    md5 (sig1): 03GJ2eHDioSe3ocLRBrMfx        2ksT2Jnp5ls:BmbqOoE26b
 snefru (sig2): 2oDQ7riYeP4Of06CKbuJzv        2dkmhCpEARAjuocR9rCZxX

/etc/security/audit_data
      st_mtime: Thu Aug 15 19:00:01 2002      Thu Aug 15 18:00:01 2002
      st_ctime: Thu Aug 15 19:00:01 2002      Thu Aug 15 18:00:01 2002
    md5 (sig1): 04KSHOhftuWNSAxfW6eTz2        1isH3C:JdpA7aI6xAWECGP
 snefru (sig2): 0FYbT9:Nd15lRelA9RTEmY        18atqac3FDZIr8AwTEkVvf

/etc/ttydefs
```

```
    md5 (sig1): 0                               1uytCCz5bfTZ59pSkCrgNT
 snefru (sig2): 0                               0EQsIRBAPaGA6SUkDzb1w0

/etc/dumpdates
    md5 (sig1): 0                               3K7OpPZm2o1Ec02PZi:49:
 snefru (sig2): 0                               265.DcLce163Vq:qkbljwY

/etc/shadow
        st_gid: 3                               1
       st_size: 533                             527
      st_mtime: Thu Aug 15 17:21:06 2002        Fri Jul 12 17:01:27 2002
      st_ctime: Thu Aug 15 17:21:06 2002        Fri Jul 12 17:01:27 2002
    md5 (sig1): 0                               0uBgTS6mKCXw.F3rn4pbgt
 snefru (sig2): 0                               2zBCo8cNiFZNfCxggmr6Wy

/etc/rmtab
    md5 (sig1): 0                               3K7OpPZm2o1Ec02PZi:49:
 snefru (sig2): 0                               265.DcLce163Vq:qkbljwY
```

Confirm that the files flagged above were modified by WEBSERV administrators. The file integrity is ensured.

If the interactive or update mode of tripwire is performed, database can be updated to the most current status. For example,

```
tripwire --interactive
```

After new database is generated, update the new database to a removable media. The following gives a sample command on backing up to a tape.

```
tar cvf /dev/rmt/0 ./database
```

1.5.Backup

Scheduled DDS3 tape backups are mandatory.
The following is an output sample after executing shown backup and associate commands:

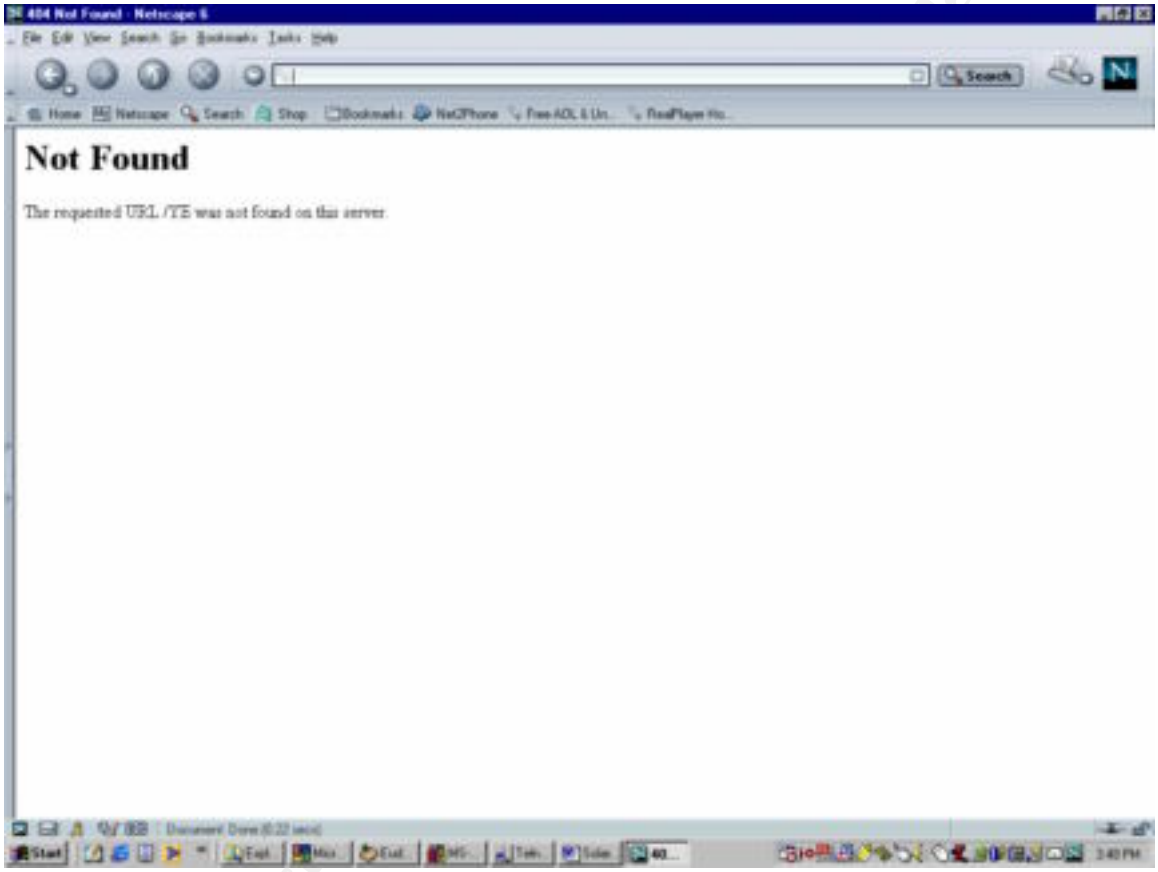
```
#mt -f /dev/rmt/0 status
HP DDS3 4mm DAT loader tape drive:
   sense key(0x6)= Unit Attention   residual= 0   retries= 0
   file no= 0   block no= 0
#mt -f /dev/rmt/0 status
HP DDS3 4mm DAT loader tape drive:
   sense key(0x0)= No Additional Sense   residual= 0   retries= 0
   file no= 0   block no= 0
mt -f /dev/rmt/0 rewind
# for dir in / /usr /var /opt
> do
> ufsdump 0f /dev/rmt/0n $dir
> done
  DUMP: Writing 32 Kilobyte records
```

```
DUMP: Date of this level 0 dump: Fri 16 Aug 2002 03:09:21 PM EDT
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdsk/c0t0d0s0 (WEBSERV:/) to /dev/rmt/0n.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 155764 blocks (76.06MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 155646 blocks (76.00MB) on 1 volume at 1818 KB/sec
DUMP: DUMP IS DONE
DUMP: Writing 32 Kilobyte records
DUMP: Date of this level 0 dump: Fri 16 Aug 2002 03:10:07 PM EDT
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdsk/c0t0d0s3 (WEBSERV:/usr) to /dev/rmt/0n.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 2510864 blocks (1226.01MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 85.14% done, finished in 0:01
DUMP: 2510846 blocks (1226.00MB) on 1 volume at 1805 KB/sec
DUMP: DUMP IS DONE
DUMP: Writing 32 Kilobyte records
DUMP: Date of this level 0 dump: Fri 16 Aug 2002 03:21:48 PM EDT
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdsk/c0t0d0s5 (WEBSERV:/var) to /dev/rmt/0n.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 708604 blocks (346.00MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 708542 blocks (345.97MB) on 1 volume at 1575 KB/sec
DUMP: DUMP IS DONE
DUMP: Writing 32 Kilobyte records
DUMP: Date of this level 0 dump: Fri 16 Aug 2002 03:25:35 PM EDT
DUMP: Date of last level 0 dump: the epoch
DUMP: Dumping /dev/rdsk/c0t0d0s4 (WEBSERV:/opt) to /dev/rmt/0n.
DUMP: Mapping (Pass I) [regular files]
DUMP: Mapping (Pass II) [directories]
DUMP: Estimated 973156 blocks (475.17MB).
DUMP: Dumping (Pass III) [directories]
DUMP: Dumping (Pass IV) [regular files]
DUMP: 973118 blocks (475.16MB) on 1 volume at 2008 KB/sec
DUMP: DUMP IS DONE
```

2. Apache Server

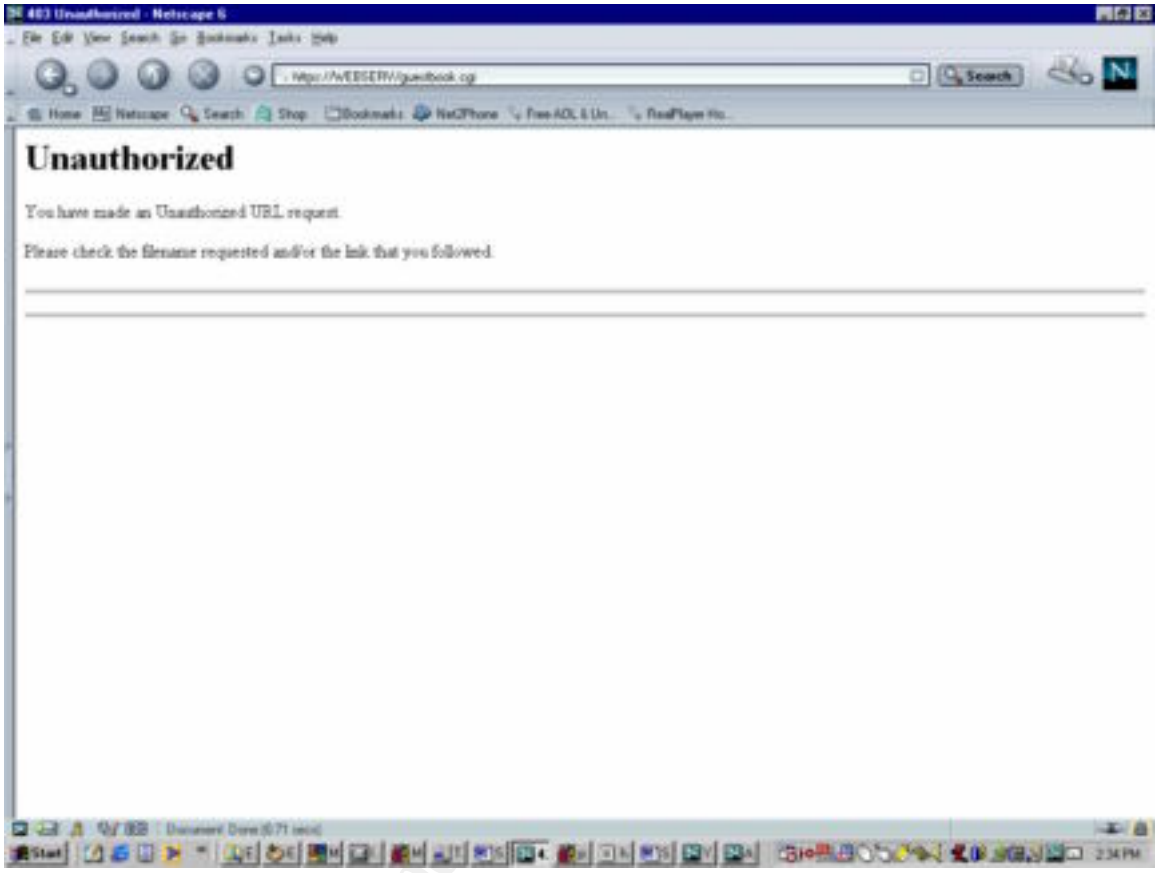
2.1 Server Identity

In order to hide WEBSERV software identity, the server signature has been turned off. The following screen output shows a simple "Not Found" 404 error page when an invalid web page request is denied, which does not disclose Apache brand name and its version number.
http request: http://WEBSERV/YE
where file "YE" does not exist.



2.2 Access Control

a) Forbidden files
Directories or files not allowed to access will display warning messages when an attempt is made.
The following is a 403 user defined Unauthorized output screen when an attempt to access `https://WEBSERV/guestbook.cgi` is made.
(Note: for testing purposes, `/opt/ApacheSSL/htdocs/guestbook.cgi` has been set as not-allowed to access in the `httpsd.conf` file.)



When a user sends suspicious requests, the user and its request signature will be captured. The following is a report sample when an attempt to access a forbidden file was made. https://WEBSERV/guestbook.cgi

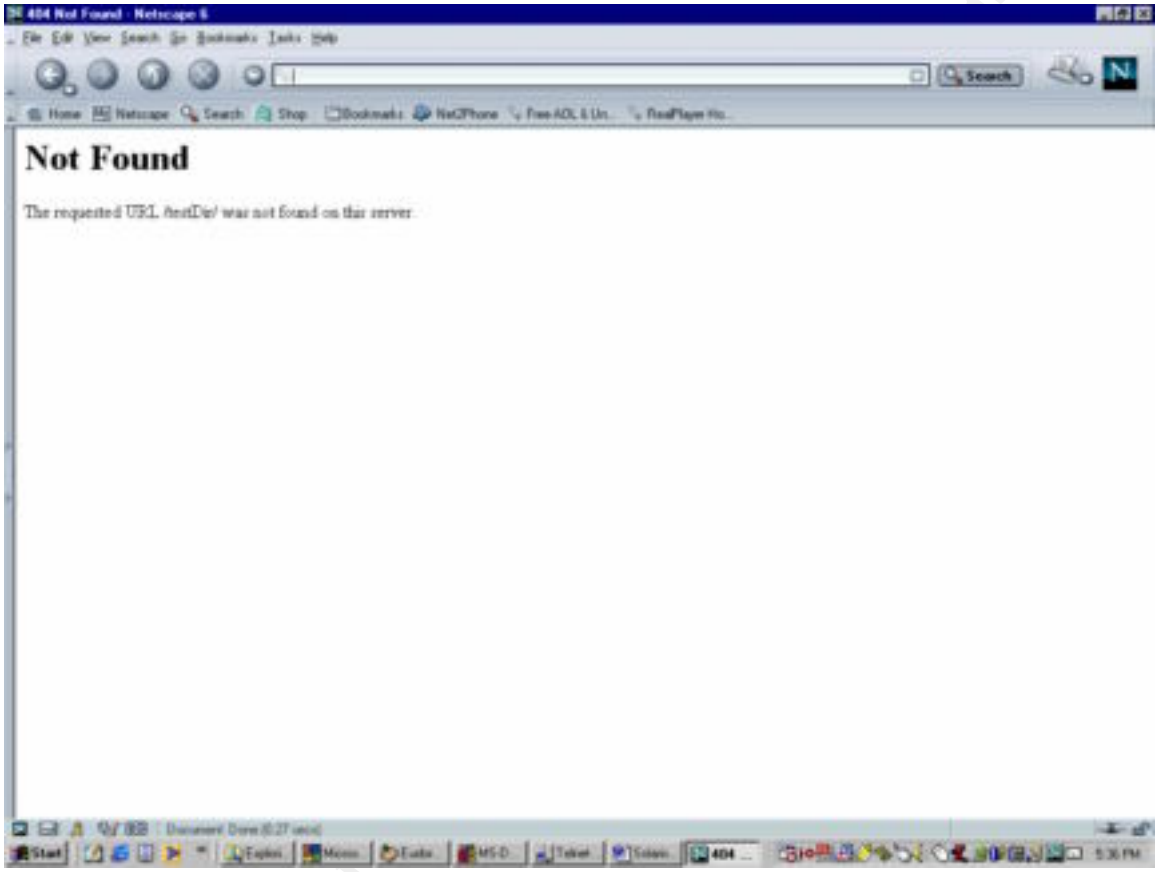
```
Subject: 403 Alert

There was An Illegal Access Attempt On:
Thu Aug 29 14:27:09 EDT 2002

Web Server - WEBSERV:443
Attacking Host - 10.0.0.2
Request Method - GET
URL Request -
User Agent - Mozilla/5.0 (Windows; U; Win98; en-US; rv:0.9.4.1)
Gecko/20020508 Netscape6/6.2.3
```

b) Do not allow automatic creation of directory listing when an index page is missing.

The following page is a screen output when an ~/testDir is requested, where 404 error message is issued.
Http request: https://WEBSERV/testDir



2.3 Audit Logs
a) Access logs
The access log provides information on when the WEBSERV was visited and by whom.
The following is an output sample from `/opt/apacheSSL/logs/httpsd_access_log`:

```
10.0.0.2 - - [23/Aug/2002:17:08:23 -0400] "GET / HTTP/1.1" 200 1456
10.0.0.2 - - [23/Aug/2002:17:08:24 -0400] "GET /apache_pb.gif HTTP/1.1"
200 2326
10.0.0.2 - - [23/Aug/2002:17:09:00 -0400] "GET / HTTP/1.1" 200 1456
10.0.0.2 - - [23/Aug/2002:17:09:01 -0400] "GET /apache_pb.gif HTTP/1.1"
200 2326
10.0.0.3 - - [28/Aug/2002:18:18:30 -0400] "GET / HTTP/1.1" 200 1456
```

```
10.0.0.3 - - [28/Aug/2002:18:18:30 -0400] "GET /apache_pb.gif HTTP/1.1"
200 2326
10.0.0.2 - - [28/Aug/2002:18:18:46 -0400] "GET / HTTP/1.1" 200 1456
10.0.0.2 - - [28/Aug/2002:18:18:47 -0400] "GET /apache_pb.gif HTTP/1.1"
200 2326
10.0.0.3 - - [28/Aug/2002:18:23:58 -0400] "GET / HTTP/1.1" 200 1456
10.0.0.3 - - [28/Aug/2002:18:25:16 -0400] "GET / HTTP/1.1" 200 1456
10.0.0.2 - - [28/Aug/2002:18:26:37 -0400] "GET /cgi-bin HTTP/1.1" 404
213
```

b) Error log

The error log provides information useful for debugging WEBSERV or detecting suspicious activities.

The following is a sample from `/opt/apacheSSL/logs/httpsd_error_log`:

```
[Wed Aug 28 18:18:17 2002] [info] removed PID file
/opt/apacheSSL/logs/httpsd.pid (pid=4880)
[Wed Aug 28 18:18:17 2002] [notice] caught SIGTERM, shutting down
[Wed Aug 28 18:18:23 2002] /opt/apacheSSL/bin/gcache started
[Wed Aug 28 18:18:23 2002] [notice] My Server/hidden Ben-SSL/1.48
(Unix) configured -- resuming normal operations
[Wed Aug 28 18:18:23 2002] [info] Server built: Aug 22 2002 11:58:36
[Wed Aug 28 18:18:23 2002] [notice] Accept mutex: fcntl (Default:
fcntl)
[Wed Aug 28 18:18:30 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:18:30 2002] [debug] apache_ssl.c(1926): CIPHER is RC4-
MD5
[Wed Aug 28 18:18:30 2002] [debug] buff.c(314): read returned 338
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
[Wed Aug 28 18:18:30 2002] [debug] buff.c(314): read returned 262
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
[Wed Aug 28 18:18:46 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:18:46 2002] [debug] apache_ssl.c(1926): CIPHER is RC4-
MD5
[Wed Aug 28 18:18:46 2002] [debug] buff.c(314): read returned 517
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
[Wed Aug 28 18:18:47 2002] [debug] buff.c(314): read returned 571
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
gcache.c:152: failed assertion `n == 1'
[Wed Aug 28 18:21:26 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:21:26 2002] [error] SSL_accept failed
[Wed Aug 28 18:21:26 2002] [notice] child pid 4931 exit signal Abort
(6)
[Wed Aug 28 18:22:38 2002] [info] removed PID file
/opt/apacheSSL/logs/httpsd.pid (pid=4927)
[Wed Aug 28 18:22:38 2002] [notice] caught SIGTERM, shutting down
[Wed Aug 28 18:22:45 2002] /opt/apacheSSL/bin/gcache started
[Wed Aug 28 18:22:45 2002] /opt/apacheSSL/bin/gcache started
[Wed Aug 28 18:22:45 2002] [notice] My Server/hidden Ben-SSL/1.48
(Unix) configured -- resuming normal operations
```

```
[Wed Aug 28 18:22:45 2002] [info] Server built: Aug 22 2002 11:58:36
[Wed Aug 28 18:22:45 2002] [notice] Accept mutex: fcntl (Default:
fcntl)
[Wed Aug 28 18:23:06 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:23:06 2002] [error] SSL_accept failed
[Wed Aug 28 18:23:58 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:23:58 2002] [debug] apache_ssl.c(1926): CIPHER is RC4-
MD5
[Wed Aug 28 18:23:58 2002] [debug] buff.c(314): read returned 208
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
[Wed Aug 28 18:24:32 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:24:32 2002] [error] SSL_accept failed
[Wed Aug 28 18:25:16 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:25:16 2002] [debug] apache_ssl.c(1926): CIPHER is RC4-
MD5
[Wed Aug 28 18:25:16 2002] [debug] buff.c(314): read returned 338
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
[Wed Aug 28 18:26:37 2002] [debug] apache_ssl.c(379): Random input
/dev/urandom(1024) -> 1024
[Wed Aug 28 18:26:37 2002] [debug] apache_ssl.c(1926): CIPHER is RC4-
MD5
[Wed Aug 28 18:26:37 2002] [debug] buff.c(314): read returned 524
rwstate=1 state=3 rstate=240 cren=0 aren=0 accept=1
[Wed Aug 28 18:26:37 2002] [error] [client 132.197.164.233] File does
not exist: /opt/apacheSSL/htdocs/cgi-bin
```

Appendix A: /etc/init.d/perf

```
#!/sbin/sh
#
# Copyright (c) 1984, 1986, 1987, 1988, 1989 AT&T.
# All rights reserved.
#
# THIS IS UNPUBLISHED PROPRIETARY SOURCE CODE OF AT&T
# The copyright notice above does not evidence any
# actual or intended publication of such source code.
#
# Copyright (c) 1997 by Sun Microsystems, Inc.
# All rights reserved.
#
#ident      "@(#)perf.sh      1.7   97/12/08 SMI"

# Uncomment the following lines to enable system activity data
gathering.
# You will also need to uncomment the sa entries in the system crontab
# /var/spool/cron/crontabs/sys.  Refer to the sar(1) and sadc(1m) man
pages
# for more information.

 if [ -z "$_INIT_RUN_LEVEL" ]; then
      set -- `/usr/bin/who -r`
      _INIT_RUN_LEVEL="$7"
      _INIT_RUN_NPREV="$8"
      _INIT_PREV_LEVEL="$9"
 fi

 if [ $_INIT_RUN_LEVEL -ge 2 -a $_INIT_RUN_LEVEL -le 4 -a \
     $_INIT_RUN_NPREV -eq 0 -a \( $_INIT_PREV_LEVEL = 1 -o \
     $_INIT_PREV_LEVEL = S \) ]; then

      /usr/bin/su sys -c "/usr/lib/sa/sadc /var/adm/sa/sa`date +%d`"
 fi
```

Appendix B /etc/issue

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

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

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

Appendix C Partial /etc/inetd.conf

```
# ftp daemon with TCP wrapper
#ftp  stream      tcp6  nowait      root  /usr/local/sbin/tcpd
      in.ftpd
#telnet
#telnet     stream      tcp6  nowait      root  /usr/local/sbin/tcpd
      in.telnetd
#
# named #
name  dgram udp   wait  root  /usr/local/sbin/tcpd    in.tnamed
#
# Shell, login, and finger
#
#shell      stream      tcp   nowait      root  /usr/local/sbin/tcpd
      in.rshd
#shell      stream      tcp6  nowait      root  /usr/local/sbin/tcpd
      in.rshd
#login      stream      tcp6  nowait      root  /usr/local/sbin/tcpd
      in.rlogind
#finger     stream      tcp6  nowait      nobody
      /usr/local/sbin/tcpd    in.fingerd
```

Appendix D /etc/sshd_config

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

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

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

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

Appendix E /etc/init.d/sshd

```
#!/sbin/sh

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

Appendix F /var/adm/sa/sar**

A sample code (extracted) of sar28 (Aug. 28) log is attached below:

```
SunOS WEBSERV 5.8 Generic_108528-15 sun4u    08/28/02

00:00:00    %usr    %sys    %wio   %idle
00:20:00       0       0       0     100
00:40:01       0       0       0     100
01:00:00       0       0       0     100
01:20:00       0       0       0     100
01:40:00       0       0       0     100

Average        0       0       0     100

00:00:00   device        %busy   avque   r+w/s  blks/s  avwait  avserv

00:20:00   fd0               0     0.0       0       0     0.0     0.0
           sd0               0     0.0       0       2     0.0    22.8
           sd0,a             0     0.0       0       1     0.0    16.3
           sd0,b             0     0.0       0       0     0.0     0.0
           sd0,c             0     0.0       0       0     0.0     0.0
           sd0,d             0     0.0       0       0     0.0    74.1
           sd0,e             0     0.0       0       0     0.0     0.0
           sd0,f             0     0.0       0       0     0.0    16.5
           sd0,h             0     0.0       0       0     0.0     0.0
           sd6               0     0.0       0       0     0.0     0.0
           st4               0     0.0       0       0     0.0     0.0

Average    fd0               0     0.0       0       0     0.0     0.0
           sd0               0     0.0       0       2     0.0    21.7
           sd0,a             0     0.0       0       1     0.0    16.4
           sd0,b             0     0.0       0       0     0.0     0.0
           sd0,c             0     0.0       0       0     0.0     0.0
           sd0,d             0     0.0       0       0     0.0   116.0
           sd0,e             0     0.0       0       0     0.0    15.4
           sd0,f             0     0.0       0       1     0.0    18.1
           sd0,h             0     0.0       0       0     0.0     0.0
           sd6               0     0.0       0       0     0.0     0.0
           st4               0     0.0       0       0     0.0     0.0

00:00:00 runq-sz %runocc swpq-sz %swpocc
00:20:00
00:40:01
01:00:00     1.0       0
01:20:00

Average      1.7       0

00:00:00 bread/s lread/s %rcache bwrit/s lwrit/s %wcache pread/s
pwrit/s
```

```
00:20:00       0       0      99       0       0      53       0
0
00:40:01       0       0     100       0       0      26       0
0
01:00:00       0       0     100       0       0      48       0
0
01:20:00       0       0     100       0       0      20       0
0

Average        0       0     100       0       0      51       0
0

00:00:00 swpin/s bswin/s swpot/s bswot/s pswch/s
00:20:00    0.00     0.0    0.00     0.0      62
00:40:01    0.00     0.0    0.00     0.0      62
01:00:00    0.00     0.0    0.00     0.0      63

Average     0.00     0.0    0.00     0.0      70

00:00:00 scall/s sread/s swrit/s  fork/s  exec/s rchar/s wchar/s
00:20:00     103       8       8    0.00    0.00    3094    2956
00:40:01     103       8       8    0.00    0.00    3069    2950
01:00:00     104       8       8    0.01    0.01    3103    2950
01:20:00     103       8       8    0.00    0.00    3069    2950

Average      119       9       9    0.01    0.01    3302    3030

00:00:00  iget/s namei/s dirbk/s
00:20:00       0       1       0
00:40:01       0       1       0
01:00:00       0       1       0

Average        0       1       0

00:00:00 rawch/s canch/s outch/s rcvin/s xmtin/s mdmin/s
00:20:00       0       0       0       0       0       0
00:40:01       0       0       0       0       0       0
01:00:00       0       0       0       0       0       0
01:20:00       0       0       0       0       0       0
01:40:00       0       0       0       0       0       0
02:00:00       0       0       0       0       0       0

Average        0       0       4       0       0       0

00:00:00  proc-sz    ov  inod-sz    ov  file-sz    ov   lock-sz
00:20:00   86/3898    0 9535/16884    0  547/547     0    0/0
00:40:01   86/3898    0 9535/16884    0  547/547     0    0/0
01:00:00   86/3898    0 9536/16884    0  547/547     0    0/0


00:00:00   msg/s  sema/s
00:20:00    0.00    0.00
00:40:01    0.00    0.00
01:00:00    0.00    0.00
```

```
01:20:00    0.00    0.00

Average     0.00    0.00

00:00:00  atch/s  pgin/s ppgin/s  pflt/s  vflt/s slock/s
00:20:00    0.02    0.00    0.00    0.20    0.39    0.00
00:40:01    0.02    0.00    0.00    0.15    0.30    0.00
01:00:00    0.04    0.00    0.00    0.25    0.45    0.00

Average     0.09    0.00    0.00    0.60    1.13    0.00

00:00:00  pgout/s ppgout/s pgfree/s pgscan/s %ufs_ipf
00:20:00     0.00     0.00     0.00     0.00     0.00
00:40:01     0.00     0.00     0.00     0.00     0.00
01:00:00     0.00     0.00     0.00     0.00     0.00

Average      0.00     0.00     0.00     0.00     0.00

00:00:00 freemem freeswap
00:20:00    9788  1386303
00:40:01    9789  1386326
01:00:00    9795  1387354
01:20:00    9789  1386314

Average     9314  1377466

00:00:00 sml_mem   alloc  fail  lg_mem   alloc  fail  ovsz_alloc  fail
00:20:00 5095616 4095201     0 64004096 50109440     0     4202496
0
00:40:01 5095616 4095201     0 64004096 50109440     0     4202496
0
01:00:00 5095616 4095969     0 64004096 50110552     0     4202496
0

Average   366626  298899     0 4597665 3631862     0      300178     0
```

References

1. *Solaris 8 7/01 Sun Hardware Platform Guide,* Sun Microsystems, Inc.
2. *Unix System Administration Handbook, 2nd & 3rd Edition,* Evi Nemeth, Garth Snyder, Scott Seebass, Trent R. Hein, Prentice Hall PTR
3. *Unix – The Complete Reference,* Kenneth H. Rosen, Gouglas A. Host, James M. Farber, Richard R. Rosinski, Macgrill Hill, 1999
4. *The Basic Backup Guide,* Exabyte Corporation, 2002
5. *Advanced Installation Guide, Solaris 8 7/01,* Sun Microsystems, Inc.
6. *Solaris Handbook for Sun Peripherals,* Sun Microsystems, Inc.
7. *http://www.tripwiresecurity.com*
8. *http://www.apache.org/docs/misc/security_tips.html*
9. *Solaris Security Step by Step, Ver2.0,* The SANS Institute, 2001
10. *SunSHIELD Basic Security Module Guide,* Sun Microsystems, Inc., February 2000
11. *Auditing Networks, Perimeters and Systems,* SANS Institute, 2002
12. *Securing Apache Step by Step,* Ryan C. Barnett, http://www.giac.org/GCUX.php
13. *http://www.openssh.com*