



# 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.

# Solaris 7 Installation Checklist

Joseph Harple  
13MAR01

This is the administrative checklist for installing Solaris 7 on a Sparc 5 server for use as a Securid server. The purpose of this checklist is to bring the machine securely up to the point of installing the Securid software. The machine will need to be as secure as possible since the security of our entire network revolves around the service provided by this machine. Only four people will access the machine remotely since all Securid administration will take place through a GUI interface. The Sparc 5 has a 2 GB hard drive with 128 MB of RAM.

This checklist is broken into four sections. The first section is the building of services that need a compiler. The compiler host will need disk space, gcc, and the make utility. I actually build a development Solaris 7 platform first on the final target hardware to build the services I needed. After the services are built, all the important binaries will be tarred up. This tar ball then also can be used on all like system in our enterprise. Section 2 is the configuration of the stripped down Solaris 7 OS on the final target hardware. The third section is the deployment of the services previously built on the newly secured server. The final section is the final configurations.

When you finish a task, please put your initials on the line provided before the task. If you have any problems, please make notes in the comment section at the end of each section. Discuss these problems with the Network Security Engineer on site. At the end, please sign and date. If at any time you have a question, ask the Engineer. Small mistakes can have huge security impacts.

## Section 1: Configuration of Services Needed for the Secure Server

- Move into a directory with space. I prefer /usr/local/src
- mkdir 4sectarget
- Verify that gcc and make are in your path
- cd 4sectarget

### Package Gathering

- ftp [ftp.porcupine.org](ftp://ftp.porcupine.org) (anonymous login)
- get /pub/security/tcp\_wrappers\_7.6.tat.gz
- ftp [ftp.openssl.org](ftp://ftp.openssl.org) (anonymous login)
- get /source/openssl-0.9.6.tar.gz
- ftp [ftp.openbsd.org](ftp://ftp.openbsd.org) (anonymous login)
- get /pub/OpenBSD/OpenSSH/portable/openssh-2.5.1p2.tar.gz
- ftp vic.cc.purdue.edu (anonymous login)
- get /pub/tools/unix/lsof/lsof.tar.Z
- ftp [ftp.freesoftware.com](ftp://ftp.freesoftware.com) (anonymous login)
- get /pub/infozip/zlib/zlib.tar.gz
- ftp [ftp.sendmail.org](ftp://ftp.sendmail.org) (anonymous login)
- get /pub/sendmail/sendmail.8.9.3.tar.gz

\_\_\_ Verify that 6 items are in your directory

### Configure lsof

```
___mkdir lsof
___mv lsof.tar.Z lsof
___cd lsof
___zcat lsof.tar.Z | tar xvf -
___rm lsof.tar.Z
___tar xvf lsof_4.55.tar
___cd lsof_4.55
___./Configure solaris
      Take an inventory-Yes
      Customize-No
___make
___Test ./lsof (ignore first run time errors) You should see everything the machine is
running.
___cp lsof /usr/local/src/4sectarget/
___cd /usr/local/src
___rm -fR lsof
```

### Configuration of zlib

```
___gzip -d zlib.tar.gz
___tar xvf zlib.tar
___rm zlib.tar
___cd zlib-1.1.3
___./configure
___make test
___make install
___cd ..
___rm fR zlib-1.1.3
```

### Configuration of tcp wrappers

```
___gzip -d tcp_wrappers_7.6.tar.gz
___tar xvf tcp_wrappers_7.6.tar
___rm tcp_wrappers_7.6.tar
___cd tcp_wrappers_7.6
___chmod 644 Makefile
___vi Mkaefile
      Uncomment the line: REAL_DAEMON_DIR=/usr/sbin/..
      This line is after the line: #SysV.4 Solaris 2.x OSF AIX
___make sunos5
___mkdir /usr/local/src/4sectarget/tcpwppr
___cp safe_finger /usr/local/src/4sectarget/tcpwppr/
___cp tcpdmatch /usr/local/src/4sectarget/tcpwppr/
___cp tcpd.h /usr/local/src/4sectarget/tcpwppr/
___cp libwrap.a /usr/local/src/4sectarget/tcpwppr/
```

```
__ cp try-from /usr/local/src/4sectarget/tcpwppr/
__ cp tcpd /usr/local/src/4sectarget/tcpwppr/
__ cp tcpdchk /usr/local/src/4sectarget/tcpwppr/
__ Also need to copy these file to the compiler machine inorder to compile ssl and
ssh
    cp safe_finger tcpd tcpdchk tcpdmatch try-from /usr/local/sbin/
    cp tcpd.h /usr/local/include
    cp libwrap.a /usr/local/lib
__ cd ..
__ rm -fR tcp_wrappers_7.6
```

### Configuration of SSL and SSH

```
__ gzip -d openssl-0.9.6.tar.gz
__ tar xvf openssl-0.9.6.tar
__ rm openssl-0.9.6.tar
__ cd openssl-0.9.6
__ ./config
__ make test (go get a beer)
__ make install
__ cd ..
__ gzip -d openssh-2.5.1p2.tar.gz
__ tar xvf openssh-2.5.1p2
__ cd openssh-2.5.1p2
__ ./configure --with-ssl-dir=/usr/local/ssl --sysconfdir=/etc/ssh --with-tcp-
wrappers -- with-ipv4-default --without-rsh -- disable-suid-ssh
__ make
__ make install
__ mkdir /usr/local/src/4sectarget/ssh
__ cd /etc
__ tar cvf /usr/local/src/4sectarget/ssh/essh.tar ssh/.
__ cd /usr/local
__ tar cvf /usr/local/src/4sectarget/ssh/ulssl.tar ssl/.
__ mkdir /usr/local/src/4sectarget/ssh/bin
__ mkdir /usr/local/src/4sectarget/ssh/sbin
__ cd /usr/local/bin
__ cp ssh* sftp /usr/local/src/4sectarget/ssh/bin/
__ cd /usr/local/sbin
__ cp sshd /usr/local/src/4sectarget/ssh/sbin
__ cd /usr/local/src
__ rm fR open*
```

### Configuration of Sendmail

```
__ gzip -d sendmail.8.9.3.tar.gz
__ tar xvf sendmail.8.9.3.tar
__ rm sendmail.8.9.3.tar
__ mkdir /usr/local/src/4sectarget/sendmail
```

```
__ cd sendmail-8.9.3/BuildTools/OS
__ chmod 644 SunOS.5.7
__ vi SunOS.5.7
Change the line: define(`confENVDEF', `-DSOLARIS=20700 ') to
define(`confENVDEF', `-DSOLARIS=20700 -DUSE_VENDOR_CF_PATH')
__ cd ../../src
__ sh Build
__ cd obj.SunOS.5.7.sun4
__ cp sendmail /usr/local/src/4sectarget/sendmail/
__ cd ..../cf/cf
__ more cp clientproto.mc sol_nullclient.mc
__ vi sol_nullclient.mc
    Change:
        OSTYPE(unknown)
        FEATURE(nullclient, mailhost.$m)
    To:
        OSTYPE(`solaris2')
        FEATURE(nullclient, mailhost.$m)
__ ./Build sol_nullclient.cf
__ vi sol_nullclient.cf
    Uncomment the line : #CE root
__ cp sol_nullclient.cf /usr/local/src/4sectarget/sendmail/
__ rm sendmail-8.9.3

__ tar cvf 4sectarget.tar 4sectarget/.
__ ftp 4sectarget.tar to a place that is going to be accessible to the target (secured)
machine.
```

### Comments for the Section:

---

---

---

## \*\*\*\*\* Section 2: Solaris 7 OS Configurations

\_\_ Boot machine with Solaris 7 media (most current)

#### Solaris Interactive Installation:

Enter or select the following configuration information during the install

Select: English: USA (ASCII)

Enter Host name: \_\_\_\_\_ (not fully qualified)

Select Networked: Yes

Enter IP address: \_\_\_\_\_

Select Name Service: **None**

Select Subnet: **Yes**

Enter Subnet mask: \_\_\_\_\_

Select Time zone: **Geographic region**

Select Correct: **Regions, Time Zones**

Verify Time

Select: **Initial**

Select Allocate Client Services: **Continue**

Select Software: **Core System Support**

Select Preserve Data: **Continue**

Select Layout File Systems: **Auto Layout**

Create: /, /opt /usr /var ,and swap

Select File System and Disk Layout: **Customize**

Remove /export/home

The following is an example layout (2 GB hard drive) for a Securid server

/ 100

/var 99

swap 128

/opt 900 (where the Securid software lives)

/usr 800

Select Mount Remote File Systems: **Continue**

Select Profile: **Begin Installation**

Select: **Auto Reboot**

passwd root (make a good password)

vi /.profile

Add the following lines:

EDITOR=vi; export EDITOR

PATH=/usr/bin:/usr/sbin:/usr/local/bin:/usr/local/sbin:.; export PATH

Set default gateway

cd /etc

echo '*ip address of default router*' >defaultrouter

ps -ef | grep route

kill -9 *PID for /usr/sbin/in.routed -q*

route -fn add default *ip address of defaultrouter*

touch notrouter

## Configure DNS

vi /etc/nsswitch.conf

Change the line: hosts: files

To: hosts: files dns

vi /etc/resolv.conf

domain *domain name*

nameserver *ip of nameserver*

nameserver *ip of nameserver*  
Test: ping www (you should receive: [www.domainname](http://www.domainname) is alive)

### **Additional Needed Packages**

```
__cd /dev/dsk (check to see what other drive is listed other than the hard drive,  
mine is c0t6d0s0)  
__mount -r -F hsfs /dev/dsk/c0t6d0s0 /mnt  
__cd /mnt/Solaris_2.7/Product  
Answer: YES to all script questions during package installs  
__pkgadd -d .SUNWter  
__pkgadd -d .SUNWacc*  
__pkgadd -d .SUNWlibC  
__pkgadd -d .SUNWdoc  
__pkgadd -d .SUNWman  
__pkgadd -d .SUNWntpr  
__pkgadd -d .SUNWntpu
```

### **Download the lastest Patch**

```
__mkdir /usr/local/src  
__cd /usr/local/src  
__ftp sunsolve.sun.com (anonymous login)  
    __cd /pub/patches  
    __get 7_Recommended.zip  
__unzip 7_Recommended.zip  
__cd 7_Recommend  
__./install_cluster -nosave  
__cd ..  
__rm -fR 7*  
__reboot when finished
```

### **Removal of Services**

```
__cd /etc/rc2.d  
__rm S30sysid.net  
__rm S71sysid.sys  
__rm S72autoinstall  
__rm S71rpc  
__rm S76nscd  
__rm K28nfs.server  
__rm S73nfs.client  
__rm S74autofs  
__rm S73cachefs.daemon  
__rm S80PRESERVE  
__rm S88sendmail
```

### **Configuring Boot Services**

```
__cd /
```

```

__ echo 'umask 022' >/etc/init.d/umask.sh
__ chmod 744 /etc/init.d/umask.sh
__ chown root:root /etc/init.d/umask.sh
__ cd /etc
__ ln -s init.d/umask.sh rc0.d/S00umask.sh
__ ln -s init.d/umask.sh rc1.d/S00umask.sh
__ ln -s init.d/umask.sh rc2.d/S00umask.sh
__ ln -s init.d/umask.sh rc3.d/S00umask.sh
__ ln -s init.d/umask.sh rcS.d/S00umask.sh
__ Test: ls grep rc* ( check to see if S00umask.sh is in all rc directorties)

__ create /etc/init.d/newinetsvc
      #!/sbin/sh
      /usr/sbin/ifconfig -au netmask + broadcast +
__ chmod 744 /etc/init.d/newinetsvc
__ chown root:root /etc/init.d/newinetsvc
__ rm -f /etc/rc2.d/S72inetsvc
__ ln -s /etc/init.d/newinetsvc /etc/rc2.d/S72newinetsvc
__ rm /etc/inet/inetd.conf
__ rm /etc/inetd.conf
__ rm /etc/auto_*
__ rm /etc/dfs/dfstab
__ cd /var/spool/cron/crontabs
__ rm adm lp
__ cd /etc/init.d
__ cp devfsadm newdevfsadm
__ chmod 744 newdevfsadm
__ chown root:root newdevfsadm
__ vi newdevfsadm
      Comment out the following lines:
      /usr/lib/devfsadm/devfsevevt >/dev/console 2>&1
      //usr/lib/devfsadm/devfsadmd >/dev/console 2>&1
__ cd /etc/rcS.d
__ rm -f S50devfsadm
__ ln -s /etc/init.d/newdevfsadm /etc/rcS.d/S50newdevfsadm

```

## Kernel Parameters

```

__ vi /etc/init.d/inetinit
      Add the following to the end of the file:
      ndd -set /dev/tcp tcp_conn_req_max_q0 8096
      ndd -set /dev/tcp tcp_ip_abort_cinterval 60000
      ndd -set /dev/ip ip_ignore_redirects 1
      ndd -set /dev/ip ip_send_redirects 0
      ndd -set /dev/ip ip_ire_flush_interval 60000
      ndd -set /dev/arp arp_cleanup_interval 60000
      ndd -set /dev/ip ip_forward_src_routed 0

```

```
    ndd –set /dev/ip ip_forward_directed_broadcasts 0
    ndd –set /dev/ip ip_forwarding 0
    ndd –set /dev/ip ip_strict_dst_multihoming 1
    ndd –set /dev/ip ip_respond_to_echo_broadcast 0
    ndd –set /dev/ip ip_respond_to_timestamp 0
    ndd –set /dev/ip ip_respond_to_timestamp_broadcast 0
    ndd –set /dev/ip ip_respond_to_address_mask_broadcast 0
__cd /etc
__vi inittab
    Comment out the following line:
    sc:234:respawn:/usr/lib/saf/sac -t 300
__vi system
    Add the following lines:
    set noexec_user_stack=1
    set noexec_user_stack_log=1
    set maxuprc=128
    set sys:coredumpsize=0
```

## Additional Logging

```
__vi /etc/syslog.conf
    Add the following (use tabs for whitespace):
        auth.info      /var/log/authlog
__touch /var/log/authlog
__chown root /var/log/authlog
__chmod 600 /var/log/authlog
__touch /var/adm/loginlog
__chmod 600 /var/adm/loginlog
__chown root:sys /var/adm/loginlog
__create /usr/local/bin
__vi /usr/local/bin/rotate
    Add the following lines:
#!/bin/ksh
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
_crontab -e
    Add the following lines to the end of the table
    30 3 * * 0 /usr/local/bin/rotate /var/log/authlog 600 4
    35 3 * * 0 /usr/local/bin/rotate /var/adm/loginlog 600 4
_cvi /etc/default/cron
    CRONLOG=YES
_cvi /etc/init.d/perf
    Uncomment the two conditionals
_cvi /var/spool/cron/crontabs/sys
    Uncomment the three enters

```

### User Access Control

```

_c/usr/sbin/passmgmt -d uucp
_c/usr/sbin/passmgmt -d nuucp
_c/usr/sbin/passmgmt -d lp
_c/usr/sbin/passmgmt -d listen
_c/usr/sbin/passmgmt -d nobody4
_c/usr/sbin/passmgmt -m -s /dev/null adm
_c/usr/sbin/passmgmt -m -s /dev/null daemon
_c/usr/sbin/passmgmt -m -s /dev/null bin
_c/usr/sbin/passmgmt -m -s /dev/null nobody
_c/usr/sbin/passmgmt -m -s /dev/null noaccess
_cgrep -v rhost_auth /etc/pam.conf > /etc/pam.new
_cmv /etc/pam.new /etc/pam.conf
_chown root:sys /etc/pam.conf
 chmod 644 /etc/pam.conf

```

### Comments for the Section:

---



---



---

\*\*\*\*\*

## Section 3: Built Services Deployment

### Adding Previously Built Services

```
__get the 4sectarget.tar (place in /usr/local/src)
__tar xvf 4sectarget.tar
__rm 4sectarget.tar
__cd 4sectarget
__cp lsof /usr/bin
__cd sendmail
__cp sendmail /usr/lib
__chmod 6551 /usr/lib/sendmail
__chown root:root /usr/lib/sendmail
__mv sol_nullclient /etc/mail/sendmail.cf
__chmod 444 /etc/mail/sendmail.cf
__chmod g-w /etc /etc/mail
__crontab -e
      Add the following line to the end of the table:
      0 * * * * /usr/lib/sendmail -q
__cd ..tcpwppr
__mkdir /usr/local/sbin
__mkdir /usr/local/include
__mkdir /usr/local/lib
__cp safe_finger /usr/local/sbin/
__cp tcpd /usr/local/sbin/
__cp tcpdchk /usr/local/sbin/
__cp tcpdmatch /usr/local/sbin/
__cp try-from /usr/local/sbin/
__cp tcpd.h /usr/local/include/
__cp libwrap.a /usr/local/lib/
__cd /usr/local/sbin
__chmod 0555 *
__chown root:daemon *
__cd ..include
__chmod 0444 *
__chown root:daemon *
__cd ..lib
__chmod 0555 *
__chown root:daemon *
__touch /etc/hosts.allow
__chown root:root /etc/hosts.allow
__chmod 600 /etc/hosts.allow
__echo 'ssh: Your ip range here' >/etc/hosts.allow
__touch /etc/hosts.deny
__chown root:root /etc/hosts.deny
__chmod 600 /etc/hosts.deny
```

```
__ echo 'ALL:ALL:/usr/bin/mailx -s "%s:connection attempt from %a" admin's  
email'>/etc/hosts.allow
```

```
__ cd /usr/local/src/4sectarget/ssh  
__ cp essh.tar /etc  
__ cp ulssl.tar /usr/local  
__ cd bin  
__ cp * /usr/local/bin  
__ cd ../sbin  
__ cp * /usr/local/sbin  
__ cd /etc  
__ tar xvf essh.tar  
__ cd /usr/local  
__ tar xvf ulssl.tar  
__ cd /etc/ssh/  
__ vi sshd_config
```

Modify to look like this:

```
Port 22  
Protocol 2,1  
ListenAddress 0.0.0.0  
SyslogFacility AUTH  
LogLevel INFO  
Hostkey /etc/ssh/ssh_host_key  
HostKey /etc/ssh/ssh_host_dsa_key  
ServerKeyBits 1024  
Key_regeneration_interval 900  
CheckMail no  
UseLogin no  
PrintMotd no  
KeepAlive no  
PermitRootLogin no  
IgnoreRhosts yes  
RhostsAuthentication no  
RhostsRSAAuthentication no  
RSAAuthentication yes  
PasswordAuthentication yes  
PermitEmptyPasswords no  
StrictModes yes  
UseLogin no  
LoginGraceTime 180  
__ chown root:root sshd_config  
__ chmod 600 sshd_config  
__ vi /etc/init.d/sshd
```

Add the following lines:

```
#!/sbin/sh  
case "$1" in
```

```

‘start’)
    if [-x /usr/local/sbin/sshd -a -f /etc/ssh/sshd_config]; then
        /usr/local/sbin/sshd -f /etc/ssh/sshd_config
    fi
    ;;
‘stop’)
    kill ‘cat /etc/ssh/sshd.pid’
*)
    echo “usage: $0 { start | stop }”
    ;;
esac
exit 0
__chown root:sys /etc/init.d/sshd
__chmod 744 /etc/init.d/sshd
__ln-s /etc/init.d/sshd /etc/rc2.d/S75sshd
__/etc/init.d/sshd start

```

**NOTE :** Since I configured SSH on the same host, I did not need to generate a new set of keys. If you configure the additional services on another host you will need to use the the following commands:

```

ssh-keygen -b 1024 -N ‘’ -f /etc/ssh_host_key
ssh-keygen -d -N ‘’ -f /etc/ssh_host_dsa_key

```

\_\_ Test: netstat -a should only list UDP:syslog idle and TCP: 22 Listen

Comments for the Section:

---



---



---

\*\*\*\*\*

## Section 4: Final Configurations

### Configuring NTP

```

__vi /etc/ntp.conf
Add the following lines
server IPADDRESS
server IPADDRESS
server IPADDRESS

```

Use server and ip address from the site:

<http://www.eeics.udel.edu/~mills/ntp/servers.htm>.

Note: you should get permission from the servers owners before connecting to them. Ntp will start after the next reboot

## **File System Configuration**

- \_\_\_ vi /etc/vfstab
  - \_\_\_\_\_ Change the following mount options for the different slices
  - \_\_\_\_\_ /usr from: - to: ro
  - \_\_\_\_\_ /var from: - to:nosuid
  - \_\_\_\_\_ /opt from: to: nosuid
- \_\_\_ vi /etc/rmmount.conf
  - \_\_\_\_\_ Add the following lines:
  - \_\_\_\_\_ mount hsfs -o nosuid
  - \_\_\_\_\_ mount ufs -o nosuid

## **Backup**

- \_\_\_ Add new media to the tape drive
- \_\_\_ reboot -- -s
- \_\_\_ fsck
- \_\_\_ mount -a
- \_\_\_ mt /dev/rmt/0 rewind
- \_\_\_ ufsdump 0f /dev/rmnt/0n /
- \_\_\_ ufsdump 0f /dev/rmnt/0n /usr
- \_\_\_ ufsdump 0f /dev/rmnt/0n /var
- \_\_\_ ufsdump 0f /dev/rmnt/0n /opt
- \_\_\_ mt /dev/rmt/0 rewoffl
- \_\_\_ Repeat with another new tape
- \_\_\_ Write protect both tapes
- \_\_\_ Store one tape in the storage safe and the other goes to the off site storage

## **Physical Security**

- \_\_\_ Place the machine in one of the secured Data Centers
- \_\_\_ List machine in the Device Data Base
  - Information needed:
  - Location: \_\_\_\_\_
  - Switch it's connect to: \_\_\_\_\_
  - Mod/Port # \_\_\_\_\_
  - Connected to the UPS \_\_\_\_\_
  - Backup Rotation \_\_\_\_\_

## **Comments for the Section:**

---

---

---

\*\*\*\*\*

Sign below that the assigned task above were successfully carried out and the machine is reasonably secured.

**Signature** \_\_\_\_\_ **Date** \_\_\_\_\_

## References

- Acheson, Steve and Pomeranz, Hal. Topics in UNIX Security. SANS Institute, (31 January 2001).
- Brotzman, Lee and Pomeranz, Hal. Running UNIX Applications Securely. SANS Institute, (1 February 2001).
- Brotzman, Lee and Pomeranz, Hal. UNIX Practicum. SANS Institute, (2 February 2001).
- Bishop, Matt. UNIX Security Tools and Their Uses. SANS Institute, 2000.
- Campione, Jeff. “Solaris 8 Installation Checklist”,  
[http://www.sans.org/y2k/practical/Jeff\\_Campione\\_GCUX.htm](http://www.sans.org/y2k/practical/Jeff_Campione_GCUX.htm) (10 March 2001).
- Pomeranz, Hal ed. Solaris Security Step by Step . v. 2.0, SANS Institute, 2000.