Build Securely a Shadow/Snort Sensor Step-by-Step Powered by Slackware Linux

By Guy Bruneau, GSEC, GCIA, GCUX

Version 4.5 – 15 September 2004

This configuration process is used to deploy Shadow/Snort sensors powered by the Slackware Linux (GNU) operating system. This setup was developed for sensors using IDE or SCSI drives. The full installation using this setup is ~160 MB in size and provides no remote services except through Secure Shell and Webmin for remote management. The Snort sensor logs are process via **Barnyard** backend processing.

This process doesn't address how to setup a Shadow Analysis station. More information available at: http://www.nswc.navy.mil/ISSEC/CID/

This installation CD contains a precompiled package that contains Apache/ACID/Mysql database ready to install right on the sensor to collect the Snort events or another server to aggregate multiple Snort sensor events. The database is built with two tables; the live database is snort and the archive is called snortarchive. See the ACID installation section to install this package.

This installation CD has a web management tool called Webmin used to securely manage a Shadow/Snort sensor via a web browser through SSL and has its own web server built-in. Additional information about Webmin is available at: http://www.webmin.com/. In addition, a Snort plugin by MSB Networks (http://msbnetworks.net/snort/) has been adapted and included in Webmin to fully manage the Snort sensor (config file, plugins, ruleset, etc) to ease remote management of the sensor. See the Webmin section to correctly configure this package.

I have included on this CD in the rel_note section, the complete instructions to build a complete Linux Apache/ACID/Mysql database on a separate server to manage several sensors. Mark Rupright also supplied a Windows equivalent for those who prefer using a Windows console.

This setup includes both Shadow and Snort IDS sensor as well as Network Grep (Ngrep) as an additional tool to be combined with Shadow or use in standalone mode for those who want additional analysis flexibility with this multipurpose platform. **This package is built for a sensor that contains 2 NIC cards** (eth0 = control and eth1 = Shadow, Snort or Ngrep) and included in the files section the necessary packages to turn the sensor back to a single NIC.

This installation is divided into tree parts: the installation of the OS, the configuration of the Shadow sensor and the configuration of the Snort sensor. In order to make Snort signature updates more flexible, I have included the oinkmaster script written by Andreas Östling available at http://www.algonet.se/~nitzer/oinkmaster/ .

The Snort sensor includes the Bleeding Edge Malware rules which are updated daily via Oinkmaster. This rules are available at: http://www.bleedingsnort.com/

The Shadow/Snort ISO image powered by the Slackware Linux OS can be downloaded at: http://www.whitehats.ca/downloads/ids/shadow-slack/shadow.iso

The MD5 signature for the ISO image is available at:
http://www.whitehats.ca/downloads/ids/shadow-slack/shadow.md5

I invite you to read the release notes on the CD, which contains additional information not contained in this document.

Important: Before you start, make sure you are disconnected from the network until the sensor has been securely configured.

Partitioning the drive:

- Boot on the system using the Slackware CD-ROM.

To partition the drive, login as root and run ***cfdisk /dev/hda*** (IDE drive) or ***cfdisk /dev/sda*** (SCSI drive). If this isn't a new drive, delete the old partitions before starting.

- hda1: / = 500 MB (Select new, select primary, size is 500, beginning, bootable)
- hda2: SWAP = 128 MB or same amount as the RAM (Select Pri/Log Free Space, new, primary, size is 128, beginning)
- Change hda2 to swap by selecting *type 82*
- hda3 (Select Pri/Log Free Space, new, primary, remainder of disk if you are going to only setup a Shadow sensor. *Otherwise, reserve at a minimum 500 MB for Snort for the hda4 partition*)
- *hda4 (Select Pri/Log Free Space, new, primary, remainder of disk for Snort)*
- Select Write to save the new settings to disk
- Select *Quit* to exit

Now that you have partitioned the drive, and saved your setting, you are ready to setup the Operating System.

- Run setup
- Select addswap
- Continue with installation: yes
- Select Linux installation partition
    - /dev/hda1 (format, reiserfs - default)
    - /dev/hda3 (format, reiserfs - default)
    - Select mount point for /dev/hda3: /LOG
    - /dev/hda4 (format, reiserfs - default) → **Optional if using Snort**
    - Select mount point for dev/hda4: /usr/local → **Optional if using Snort**

  - Select add none and continue with setup
- Select *continue* to go to the SOURCE section
- Select *1* to install from a Slackware CD-ROM
- Select *auto* to scan automatically for the CD-ROM
- Make sure the CD is in the CD-ROM drive and select OK
- Select *Yes* on continue

- Install Shadow/Snort installation CD which only shows 4 packages:

A, AP, D, N

- Select *OK* to continue and go to the *INSTALL* section
- Select install everything (full)
- Install a Linux kernel from the CD-Rom
- Select default kernel
- Select *bare.i* if using IDE or *scsi.s* or other SCSI depending of your drive
- Make a boot disk for recovery (LILO)
- After the boot disk, choose continue with the configuration
- Skip modem configuration (*no modem*)
- Install LILO and select expert
  - Select Begin, at the blank prompt press enter, select *standard*, install to *MBR* confirm location to install lilo (select default @/dev/hda) and none
  - Add Linux and choose the root partition (i.e. /dev/hda1)
  - Use *Linux* as a partition name
  - Install LILO
- Configure the network with your settings
- Probe the network card
- When the network card has been probed, it will ask if the settings are correct
- Setup the hardware clock
- Setup the root password
- Exit setup
- Reboot (reboot at the prompt)
- Manually eject the CD-ROM
- Log back into the sensor as root
- Delete residual mail *rm /var/spool/mail/root*

Note: Run **pkgtool** and **remove** the ***pcmcia*** package if not using a laptop, remove the ***ngrep*** package you are not planning to use it in combination with Shadow, remove the ***logger (Shadow) and ngrep*** packages if only using Snort or remove the ***snort, ngrep and wget*** packages if only using Shadow.

Configure SSH TCP Wrappers in the following way:

- vi /etc/hosts.allow
- Add in the TCP Wrappers file which host(s) are allowed to connect to the sensor

```
sshd: 192.168.3. \
                    192.168.2.6 \
                    .site.ca
```

- The /etc/hosts.deny has been configured to deny ALL (ALL: ALL) by default

Configure iptables firewall (rc.firewall)

Note: You need a firewall (iptables) to allow the sensor to be as invisible as possible. Use the firewall supplied with this installation or create your own.

O- Configured the firewall located in */etc/rc.d* directory or create your own
O- Edit the firewall scrip and change to variable according to your site
O- *chmod 755 /etc/rc.d/rc.firewall* to enable the firewall
O- To start the firewall now do: */etc/rc.d/rc.firewall* at the prompt
O- Check the firewall policy with the following command: iptables -L
O- The firewall will start upon the next reboot

Setting up the NICs if undetected during the installation

Note: A list of the NIC kernel modules is in the /etc/rc.d/rc.modules file. I recommend using an Intel PCI card for the Shadow/Snort packet collection. If the NIC card detection setup fail, the card can be manually added to this script. The cards will be loaded in the order they are listed in this configuration file.

Setting up NIC modules (example)

```
vi /etc/rc.d/rc.netdevice

# RealTek 8129/8139 to communicate with the Management station (eth0)

/sbin/modprobe rtl8139oo

# Intel EtherExpress Pro/100 PCI support used to collect packets (eth1):

/sbin/modprobe eepro100
```

Webmin Configuration

Webmin is a secure remote IDS manager. The IDS can be remotely managed via an SSL browser to manage Apache, the logs, the entire Snort IDS including its plugins, the rules, etc. It is quite versatile and very easy to use for those who prefer using a GUI to manage the sensor.

After you log in Webmin, to manage the Snort sensor selects the **Servers** icon. You then are going to see Snort IDS Admin eth0 to eth4. The eth0 and eth1 are the only two-interface preconfigured Module Config.

Configuring Webmin

You need to change the default account password before making this system operational. The default account is **admin** and the default password is **admin**. Change the default admin account password the following manner. At the sensor command line console do:

```
/usr/local/webmin/changepass.pl /etc/webmin admin newpassword
```

The Webmin service can be started and stopped this way:

```
/etc/webmin/start
/etc/webmin/stop
```

Access is via SSL this way:

https://yourIPaddress:10000

A FAQ is available on the Webmin site at:

http://www.webmin.com/faq.html

Snort IDS configuration

Snort Version 2.2.0 (Build 30)
12 August 2004

If you are going to enable the firewall on the sensor, don't forget to add the IP of the ACID database in the rc.firewall script if the sensor is going to be sending the events to a remote ACID database.

To install ACID on a remote console, check the installation the document acid_mysql.pdf in the rel_notes directory on the shadow CD. The installation notes is for ACID, MySQL and Snort running on the same platform in **single mode** or **distributed mode** with ACID/mysql on one platform and multiple Snort sensors distributed across the enterprise reporting to a central Snort console. Snort is configured to automatically use **Barnyard** and through its output log_unified function.

Note: You can install the acid-1.2-i386-1.tgz available on the CD which has been expressly built to run on this sensor. Check the section below called **Install the ACID ready to use package** on how to configure the package.

This Snort binary has been pre-built with the following options:

--with-mysql	Support for MySQL
--enable-smbalerts	SMB alerting capability via Samba
--enable-sourcefire	Enable Sourcefire specific build options
--enable-flexresp2	Flexible response on hostile connection attempts
--enable-perfmonitor	Enable perfmonitor preprocessor
--enable-linux-smp-stats	Enable statistics reporting through proc

To uninstall Snort on the IDS platform do:

- Run pkgtool and select remove packages from other directories
- Select Snort to remove the package

The necessary files have been installed into the /usr/local/snort directory. This version contains the latest signature libraries from http://www.snort.org as of the above date.

The Snort binary is compiled with the "-static" option and the sensor runs in a charrooted jail by default in the /usr/local/snort. Thanks to GJ Hagenaars' contribution, the Snort sensor is now controlled by a startup script located in /etc/rc.d/rc.snort.

- Snort configuration files for multiple interfaces are located in /usr/local/snort/etc and each interfaces has a snort.internal.conf or snort.external.conf. Edit each of these files to reflect your settings.
- Snort configuration files for multiple interfaces are located in /usr/local/snort/etc and each interfaces should be listed in snort.internal.nic and snort.external.nic.

The eth0 interface is configured by default to snort.internal.nic. Edit each of these files to reflect your settings.

- The Snort unified logs are saved in /usr/local/snort/log/eth(0-6). Default installation will contains eth0 and eth1 directory.
- Test the configuration by running the ./**check_snort_eth0** and **check_snort_eth1** script. If the this is successful, start the sensor
- **/etc/rc.d/rc.snort start** script to starts the sensor

Oinkmaster

You should read the following document on how to configure oinkmaster to download the signature updates. ftp://ftp.it.su.se/pub/users/andreas/oinkmaster/docs/README

I have included a troubleshooting script in the /usr/local/snort called check_snort_eth0 and check_snort_eth1 to ensure the rules are configured correctly.

IMPORTANT

By default, the sensor will not process any pass rules. In order for the sensor to correctly process pass rules, you must add the **–o** switch in the /etc/rc.d/rc.snort script to change the rule order from Alert, Log, Pass to **Pass, Alert, Log**. Be very careful that your pass rules will not disable other rules in the ruleset.

Example

```
$SNORTBIN/snort \
        -d \
        -o \
        -I \
        -g snort \
        -u snort \
        -i $NIC \
        -c $SNORTETC/snort.${POLICY}.conf \
        -l $SNORTLOG/$NIC \
        -D
```

Install the ACID ready to use package

mount /mnt/cdrom
cd /mnt/cdrom/acid
run pkgtool

Select **Current**	Install packages from the current directory
Package **acid-1.4-i486-1.tgz**	Shows up for installation
Select acid and press enter	To complete the installation

Setting up the database and its users

If mysql database is not started, start it this way

```
/usr/local/mysql/bin/mysqld_safe user=mysql &
```

Change the mysql root password

```
/usr/local/mysql/bin/mysqladmin –u root password ‘your-new-password-for-sql_user-root’
```

Default password for dba and snort = password

```
mysql –p                                                (root password set earlier)
\u mysql                                                (User mysql)
DELETE FROM user WHERE User=’’;                         (2 single quotes)
DELETE FROM user WHERE Password=’’;                     (2 single quotes)
GRANT ALL PRIVILEGES ON *.* TO dba@localhost IDENTIFIED BY
‘make_a_password_for_user_dba’;
GRANT INSERT,SELECT,DELETE ON snort.* TO snort@localhost \
IDENTIFIED BY ‘make_a_password_for_user_snort’;
\q                                                      (Quit mysql)
```

Note: When entering the passwords for the sql_user dba & snort, ensure it gets enclosed in single quotes or you will get an error.

Configuring acid

vi /usr/local/apache/conf/httpd.conf and change for the following:

ServerName = your_server_name (X2)
ServerAdmin = root@your_server_name

vi /usr/local/apache/htdocs/acid_conf.php and change for the following:

$alert_dbname:	MySQL database name where the alerts are stored	***snort***
$alert_host:	host where the database is stored	***localhost***
$alert_port:	port where the database is stored	***3306***
$alert_user:	username into the database	***root***
$alert_password:	password for username	***whatever_u_asassigned***

$archive_dbname:	***snortarchive***
$archive_host:	***localhost***
$archive_port:	***3306***
$archive_user:	***root***
$archive_password:	***whatever_u_asassigned***

<u>Note</u>: You must use the values you chose while creating the mysql database above. The default password set in the acid_conf.php file is **SnortBox**.

Start up Apache

/usr/local/apache/bin/apachectl startssl

Configure snort for your site

vi /usr/local/snort/etc/snort.internal.conf
vi /usr/local/snort/etc/snort.external.conf

This is optional

If you want to send the logs/alarms to other places as well, uncomment any of these lines:

output alert_syslog: LOG_AUTH LOG_ALERT	(Send alarms to a syslog server)
output alert_full: alert.full	(Output all the data information)
output alert_smb: workstation.list	(Sending alerts to a workstation)
output log_unified: filename snort.log, limit 128	(Sending to Barnyard)

<u>Note</u>: If using Barnyard unified format, skip Local Sensor/Database configuration and go to Barnyard configuration section.

Local Sensor/Database (All on one line)**:**

```
output database: alert, mysql, dbname=snort user=snort host=localhost  \
password=usersnortpassword  sensor_name=meaningful_name_for_host
```

- cd /usr/local/snort
- Test the configuration by running the ./**check_snort_eth0** and **check_snort_eth1** script. If the this is successful, start the sensor

Change the Apache password

Apache has access control build in and a default username of ***cyber*** and a default password of ***admin***. The default password must be changed to access ACID. To change the default password do the following:

- cd /usr/local/apache/auth
- htpasswd passwd ***cyber***
- Enter new password when prompted

If you want to add a new account, repeat the same procedures as above but you will need to edit the group file and add the name in it. The current group file only has the cyber account in it and to add another name, do the following:

- Edit the group file with vi and add the new name separated with a space:
- snort: cyber guy
- Save the group file and the new account has now access the ACID Cyber Threat Portal.

Reboot the server

Test your ACID database access

https://yoursite//

Generating a new Apache SSL certificate

In order to generate a new Apache web server certificate you can follow the following instructions or you can go to the sensor and **run the /root/new_Apache_certificate script** that will take you through the entire process.

To build a new Apache site certificate using the supplied script, as root do:

```
/root/new_Apache_certificate
```

This is based on information provided by Adrien de Beaupre.

Follow these instructions to manually build a new Apache site certificate. As root on the sensor do:

1- Create the private key.

```
/usr/bin/openssl genrsa -des3  -out server.key 1024
```

2- Create a copy of the key that doesn't require a password for apache to load.

```
/usr/bin/openssl rsa -in server.key -out server.pem
```

3- Create a request for a certificate

```
/usr/bin/openssl req -new -key server.key -out server.csr
```

4- Enter pass phrase for server.key

You are about to be asked to enter information that will be incorporated into your certificate request. What you are about to enter is what is called a Distinguished Name or a DN.

There are quite a few fields but you can leave some blank. For some fields there will be a default value, If you enter '.', the field will be left blank.

- Country Name (2 letter code) [AU]:CA
- State or Province Name (full name) [Some-State]:Ontario
- Locality Name (eg, city) []:Ottawa
- Organization Name (eg, company) [Internet Widgits Pty Ltd]: Seeker
- Organizational Unit Name (eg, section) []:Cyber Division
- Common Name (eg, YOUR name) []: www.erin.ca
- Email Address []:admin@erin.ca

5- Sign the certificate with the server key (self-signed), valid for 600 days.

```
/usr/bin/openssl x509 -req -days 600 -in server.csr -signkey server.key –out server.crt
```

6- Copy server.key and server.pem to /usr/local/apache/conf/ssl.key/

7- Copy server.crt to /usr/local/apache/conf/ssl.crt/

8- Copy server.csr to /usr/local/apache/conf/ssl.csr/

9- Edit /usr/local/apache/conf/httpd.conf and change server.key to server.pem.

10- Stop and start apache

```
/etc/rc.d/rc.httpd restart
```

Configuring Barnyard to send ACID database

Barnyard Version 0.2.0 (Build 32)
15 September 2004

Barnyard unified logging output processor is used with this system to process all the data from the Snort sensor. The default scripts are setup to process the unified logs to binary logs format and can also be configured to process MySQL database logging as well as syslog. In order to use Barnyard with MySQL, you must configure the output process as per the instructions below.

This Barnyard binary has been pre-built with the following options:

```
--with-mysql                Support for MySQL
--with-mysql-includes
--with-mysql-libraries
```

The output processor is enable with **log_unified** filename snort.log, limit 128 in the **snort.external.conf** and **snort.internal.conf** (only if you are monitoring this interface) to log to the /usr/local/snort/log/eth* directory.

Before proceeding with this step, make sure you have setup your Snort database account, which is part of the previous section in "Setting up the database and its users". Configure Barnyard to forward data from a sensor to the ACID database in the following manner:

```
vi /usr/local/barnyard/etc/barnyard.eth0.conf
vi /usr/local/barnyard/etc/barnyard.eth1.conf
```

```
# set the hostname (currently only used for the acid db output plugin)
config hostname: snorthost

# set the interface name (currently only used for the acid db output plugin)
config interface: eth0

# Converts data from the dp_log plugin into standard pcap format
# Argument: <filename>

output log_pcap

# Available as both a log and alert output plugin.  Used to output data into
# the db schema used by ACID

output log_acid_db: mysql, database snort, server localhost, user snort, password password, detail full
```

- Save the file and restart Barnyard

```
/etc/rc.d/rc.barnyard restart
```

- Ensure the Barnyard processes are running

```
ps –aef |grep barnyard
```

Note: If you are going to create some local rules (i.e. local-eth1.rules) you MUST include the SID and the SID name in the /usr/local/snort/rules/local-sid.map. These must match the information put in the rule file as follow:

```
9001||Local task rule
9002||TCP connections to TCP 3127
```

Configuring Local Shadow Console

In order to run a local console on the same box as the sensor, you must install the ACID package because requires an Apache web server in order to access the hourly logs. If you decide you still want to run the Shadow console but do not want to run the ACID database on the sensor, install ACID and remove the MySQL database as follow:

```
rm –r /usr/local/mysql
```

Edit the /etc/rc.d/rc.local file and comment out the mysql database startup command

```
echo "Starting mysql database..."
/usr/local/mysql/bin/mysqld_safe --user=mysql &
```

For the log file to be processed, you need to enable the cronjobs under the Shadow account. To do so, do the following:

As shadow do (These cronjob must be enabled):

```
su - shadow
crontab –e
```

Uncomment the conjob time and change the *sensor name* to the name of your system

```
# Transfer the Shadow log from /LOG/RAW/gmt to /LOG/shadow
5 * * * * /usr/local/SHADOW/fetchem.pl -l sensor name

# Collect statistics each nights
1 0 * * * /usr/local/SHADOW/stats/do_daily_stats.pl –l sensor name

# Cleanup the directory on the Sensor to prevent data lost
17 1 * * * /usr/local/SHADOW/cleanup.pl -l sensor name
```

Now you will need to configure the Shadow filter for your site

- cd /usr/local/SHADOW/filters
- mv shadow to *sensor name*
- cd *sensor name*

Edit using vi the following filters:

ip.filter
icmp.filter
tcp.filter
udp.filter

```
Change the net 172.21 to the correct network address.
```

Configure site file

su - shadow
cd /usr/local/SHADOW/sites
cp shadow.ph sensor_name.ph (Where sensor_name is your sensor's correct name)
vi sensor_name.ph

Change all of the following variables:

$SITE="*change shadow to sensor name*"
$SENSOR="*change shadow to sensor name*"
$WEB_SERVER="*change shadow to sensor name*"
our @LOCAL_IP = ("*Assign real IP*")
our @LOCAL_DOMAIN=("*Assign real IP*")

Setting shadow.conf

As root user do:

vi /etc/shadow.conf

Modify the following line:

Change all of the following variables to reflect the correct information

our %SENSOR_LABELS = *change shadow to sensor name*
our @SENSORS = ("*change shadow to sensor name*")
our $DEFAULT_SENSOR = "*change shadow to sensor name*"
our $SHADOW_FILE_SERVER= "*Change shadow.erin.ca to sensor name*"
our @LOCAL_IP = ("*Assign real IP*")
our @OBF_IP = ("*Assign obfuscated IP*")

Test your configuration in debug mode

1. chown -R shadow:shadow /LOG
2. su - shadow
3. cd /usr/local/SHADOW
4. ./fetchem.pl --loc sensor_name --debug (press enter to run the last hour log)
5. more /tmp/fetchem.log (This file will show whether there was any errors and where the error is)
6. If you had some errors fix them and run through steps 3 and 4
7. When it shows success go to the shadow website
8. https://IP_address/shadow/

Ngrep Version 1.40.1

5 March 2003

Configure Ngrep in the following way (Pre-configured with this installation):

I would like to thank Alex Arndt for supplying the scripts in this section to make them part of this sensor. In this section, you can run ngrep in the same manner as a Shadow sensor in real-time to monitor as in the example below, someone successfully gaining access to a workstation and running Directory of C.

- cd /usr/local/NGREP/sensor and make the following changes:

    - vi **template.ph** and verify the following settings:

```
$FILTER = "'[C|c][+][D|d][I|i][R|r][+][C|c]|Directory of [C|c]'";
```

The start_ngrep.pl and stop_ngrep.pl scripts are used by cron to rollover a new file hourly. Check the rc.local section to start ngrep when the sensor start and the cron section to see the rollover configuration.

- cd /usr/local/NGREP

You can use **ngreppm.pl** to download the files to a remote analysis station.

Note: Adapt the filter to the keyword being searched.

Ngrep Pattern Search script

A new script is now in the /root called ***ngrep_pat_search.pl*** which can be used to search multiple days. The format is as follow:

```
./ngrep_pat_search.pl –b YYYYMMDDHH –e YYYYMMDDHH –p PATTERN –i BPF filter
```

-b Beginning day and hour
-e Ending day and hour
-p Pattern to search. Ex: "kazaa"
-i BPF Filter. Ex: "tcp port 80"

It is configured with the following ngrep options:

-t Print a timestamp in the form of YYYY/MM/DD HH:MM:SS.UUUUUU everytime a packet is matched
-I Ignore the case of the expression
-I Read pcap_dump back into ngrep

This script match keywords and BPF filters. Here are some examples:

Print all the packets on TCP port 80 that contains http

```
./ngrep_pat_search.pl –b YYYYMMDDHH –e YYYYMMDDHH –p “http” –i “tcp port 80”
```

Print all the packets on TCP port 80

```
./ngrep_pat_search.pl –b YYYYMMDDHH –e YYYYMMDDHH –p “ ” –i “tcp port 80”
```

Print all the packets that contains the string http

```
./ngrep_pat_search.pl –b YYYYMMDDHH –e YYYYMMDDHH –p “http” –i “ ”
```

Note: The output can be redirected to a file by using **>** ***filename***

Shadow Command Line Pattern Search script

A new script is now in the /root called ***pat_search.pl*** which can be used to search multiple days. The format is as follow:

```
./pat_search.pl –n –b YYYYMMDDHH –e YYYYMMDDHH –p PATTERN
```

-n	Do not resolve addresses (faster)
-b	Beginning day and hour
-e	Ending day and hour
-p	Pattern to search in BPF format. Ex: “tcp port 27374”

It is configured with the following tcpdump options:

-X	Display HEX and ASCII data
-v	Verbose
-s 0	Read back the maximum snaplen on each packets

The output can be redirected to a file by using > ***filename***

If you would like to install the Shadow console, you should read the following document from the Shadow Team at:

http://www.nswc.navy.mil/ISSEC/CID/SHADOW-1.8-Install.pdf

Background Information About this Setup

How to manually mount a CD-ROM or diskette

To manually mount the CD-ROM do:

```
mkdir /cdrom                      (create cdrom directory)
mount /dev/hdc /cdrom -t iso9660  (mount the cdrom)
umount /cdrom                     (un-mount the cdrom)
```

The cdrom maybe hdb, hdc or hdd depending where it has been installed in the computer. To find out which device is the CD-ROM, do *dmesg |more*

To manually mount the floppy do:

```
mkdir /floppy                     (create floppy directory)
mount /dev/fd0 /floppy -t vfat    (mount the floppy)
umount /floppy                    (un-mount the floppy)
```

Operating System Patches

The Slackware web site should be monitored for any new patches that should be applied on the selected packages at Annex A. The site is http://www.slackware.com

The security list is available at:

http://www.slackware.com/security/list.php?l=slackware-security&y=2004

Slackware Patch Maintenance Script

http://128.173.184.249/slackupdate/

Patches can be maintained and downloaded by running the /root/slackupdate.sh script. This script will check for any package that are available for update and saves them in /tmp/slackupdate. To install the patch updates as follow:

```
cd /tmp/slackupdate
telinit 1
upgradepkg <patch>.tgz
lilo      (Must be run only if upgraded IDE kernel)
telinit 3
```

Note with other kernels: If you are using a kernel other than IDE, you must download the new kernel that is normally a bzImage and copy it to /boot/vmlinuz and run lilo before you reboot to ensure the new kernel will be used.

Open Secure Shell (openssh) is part of the installation on this CD

Note: Secure Shell is normally used to transfer data between the monitoring station and the sensor. The instructions on how to setup an analysis station are available at the NSWC site. The site is at: http://www.nswc.navy.mil/ISSEC/

Configure Shadow in the following way (Pre-configured with this installation):

- cd /usr/local/SHADOW/sensor and make the following changes:

    - vi gmt.ph and verify the following settings:

    * Decide whether you want to use local time or GMT (default GMT)
    * $LOGPROG = "/usr/sbin/tcpdump" (or its exact location)
    * $PROGPAR = "-i eth1" (or eth0 if using a single card as the traffic collector)
    * $GZIPPROG = "/bin/gzip" (or its exact location)
    * $LOGDIR = "/LOG/RAW/gmt" (if not using default, change it here)

<u>Note</u>: All of the Shadow files are owned by the shadow user and part of the shadow group. The account has been locked. In order to use this account, run "*passwd shadow*" to add your own user password.

The message of the day changed to reflect more proactive security (Pre-configured with this installation):

- vi /etc/motd

```
***************************************************************
This is a controlled access system.
This station is monitored at all times.
Only authorized users may connect
***************************************************************
```

- cp /etc/motd /etc/issue

Update rc.local to start local applications:

O- vi /etc/rc.d/rc.local and add the following services

```
#!/bin/sh
#
# /etc/rc.d/rc.local:  Local system initialization script.
#
# Put any local setup commands in here:
# Starting eth1
```

```
echo "Fire up eth1 now to start Shadow collection..."
/sbin/ifconfig eth1 promisc -arp
/sbin/ifconfig eth1 up
#
echo "Starting mysql database..."
/usr/local/mysql/bin/mysqld_safe --user=mysql &
#
echo "Starting shadow sensor..."
/usr/local/SHADOW/sensor/start_logger.pl gmt
#
# Uncomment any of these to start Network Grep sensor
#
#echo "Starting Network Grep sensor..."
#/usr/local/NGREP/sensor/start_ngrep.pl kazaa
#/usr/local/NGREP/sensor/start_ngrep.pl gnutella
#/usr/local/NGREP/sensor/start_ngrep.pl dir_c
#
echo “Starting Webmin…”
/etc/webmin/start
#
echo "Starting firewall..."
/etc/rc.d/rc.firewall
#
echo "Starting Snort sensor..."

if [ -x /etc/rc.d/rc.snort ]; then
  . /etc/rc.d/rc.snort start
  . /etc/rc.d/rc.barnyard start
fi
#
# All done
```

Update cronjob to start Shadow, update time, and cut new logs each hour (Pre-configured on Shadow CD):

Crontab running as Root

```
# Sync with a time server on a daily basis

17 23 * * * /usr/sbin/ntpdate time-a.nist.gov
18 23 * * * /sbin/hwclock --systohc

# Cut a new Shadow log on a hourly basis
0 * * * * /usr/local/logger/SHADOW/sensor_driver.pl gmt > /dev/null 2>1&

# Restart snort every night at midnight after updated Snort
# signatures have been downloaded
```

```
#
5 0 * * * /etc/rc.d/rc.snort restart > /dev/null 2>1&

# Cut a new ngrep log on a hourly basis
# You can uncomment any of these examples or create your own in the
# /usr/local/NGREP/sensor directory using the template.ph

#0 * * * * /usr/local/NGREP/sensor/ngrep_driver.pl kazaa > /dev/null 2>1&
#0 * * * * /usr/local/NGREP/sensor/ngrep_driver.pl gnutella > /dev/null 2>1&
#0 * * * * /usr/local/NGREP/sensor/ngrep_driver.pl dir_c > /dev/null 2>1&
# Check that sshd is running

0,5,10,15,20,25,30,35,40,45,50,55 * * * * /usr/local/sbin/chk-sshd.pl
```

Crontab running as Snort

```
# This crontab runs at 1 am to updated the Snort signatures using the oinkmaster.pl
# script and merge the new rules into the rules directory.
#
0 0 * * * /usr/local/snort/rc.snortupdate
```

Note: During the first reboot, you will notice some errors in directory /LOG/RAW/gmt with files sniff and sensor date. This is normal as those files do not exist yet and the sensor is creating them.

O- Log in as root
O- Run *ps -aef* and verify the services running. (See picture 1)
O- Run *netstat -at* and verify the active connections (See picture 2). ssh should be the only service listening for remote login.
O- Check Annex D for a NMAP port reconnaissance probe confirming ssh is the only available service.
O- cd /LOG/RAW/gmt and verify the sensor is collecting. Do an *ls -l* and look for the hourly file that looks like this: tcp.20010312.gz with 0 bytes.
O- The sensor is ready to be connected to the network.

Slackware Linux security files

/etc/inetd.conf	Daemon configuration file
/etc/rc.d/rc.S	Start up script for single user mode
/etc/rc.d/rc.M	Start up script for Multi user mode
/etc/rc.d/rc.local	Start up script for multiple NIC
/etc/issue	Change to reflect something other than Linux version
/etc/motd	Change Message of the Day
/etc/rc.d/rc.firewall	Setup firewall
/var/adm/messages	General log file
/var/adm/syslog	Syslog file
/usr/local/SHADOW	Shadow directory files
/usr/local/NGREP	Network Grep sensor files
/etc/rc.d/rc.snort	Snort startup script
/usr/local/snort	Snort directory files

References

Securing Linux – A Survival Guide for Linux Security
SANS Institute

Intrusion Detection: Shadow Style step-by-step guide, Version 1.2.2
SANS Institute 1998

Installing Shadow
http://www.nswc.navy.mil/ISSEC/CID/SHADOW-1.8-Install.pdf

Shadow step-by-step Intrusion Detection using TCPdump
http://www.nswc.navy.mil/ISSEC/CID/shadow.ppt

Snort IDS
http://www.snort.org

Oinkmaster
http://www.algonet.se/~nitzer/oinkmaster/

Webmin
http://www.webmin.com/

MSB Networks
http://msbnetworks.net/snort/

SlackUpdate
http://128.173.184.249/slackupdate/

ACID
http://www.andrew.cmu.edu/~rdanyliw/snort/snortacid.html

MySQL Windows Control Center
http://www.mysql.com/downloads/mysqlcc.html