Build Securely Snort with Sguil Sensor Step-by-Step Powered by Slackware Linux

By Guy Bruneau, GSEC, GCIA, GCIH, GCUX, GCFA
Version 6.3 – 9 April 2009

INTRODUCTION 3

ALL-IN-ONE DATABASE/SENSOR SUMMARY INSTALLATION 4
DATABASE SUMMARY INSTALLATION 4
SENSOR SUMMARY INSTALLATION 5
CLIENT CONFIGURATION 6
Sguil sguil.conf update 6
CLIENT ACCESS TO DATABASE 7

INSTALLATION, CONFIGURATION AND PARTITIONING THE DRIVE 7

DATABASE AND SENSOR 7
Large Disk RAID Array Installation 8
Note on MySQL Drive size estimate: 8
INSTALL THE SOFTWARE 9
TCP WRAPPER - SSH 10
IPTABLES FIREWALL 10
SETTING UP NIC MODULES (EXAMPLE) 11
MOUNTING USB DRIVE 11
SNORT SENSOR MEMORY LIMITATION 12

WEBMIN CONFIGURATION 13

CONFIGURING WEBMIN 13
ACCESS IS VIA SSL THIS WAY: 13
A FAQ IS AVAILABLE ON THE WEBMIN SITE AT: 13

CONFIGURING BARNYARD TO SEND TO SGUIL DATABASE 14

CONFIGURE BARNYARD TO FORWARD DATA TO SGUIL DATABASE 14
ENABLE DATA INSERT INTO SGUIL DATABASE 14
CONFIGURE SSL BETWEEN A SENSOR AND A SGUIL SERVER 15
LOCAL SID RULE MAPPING 15

SHADOW HOURLY WEB REPORTS 16

NOW YOU WILL NEED TO CONFIGURE THE SHADOW FILTER FOR YOUR SITE 16
TEST YOUR CONFIGURATION IN DEBUG MODE 16

NEW PROCEDURES FOR SNORT OINKMASTER UPDATES 17

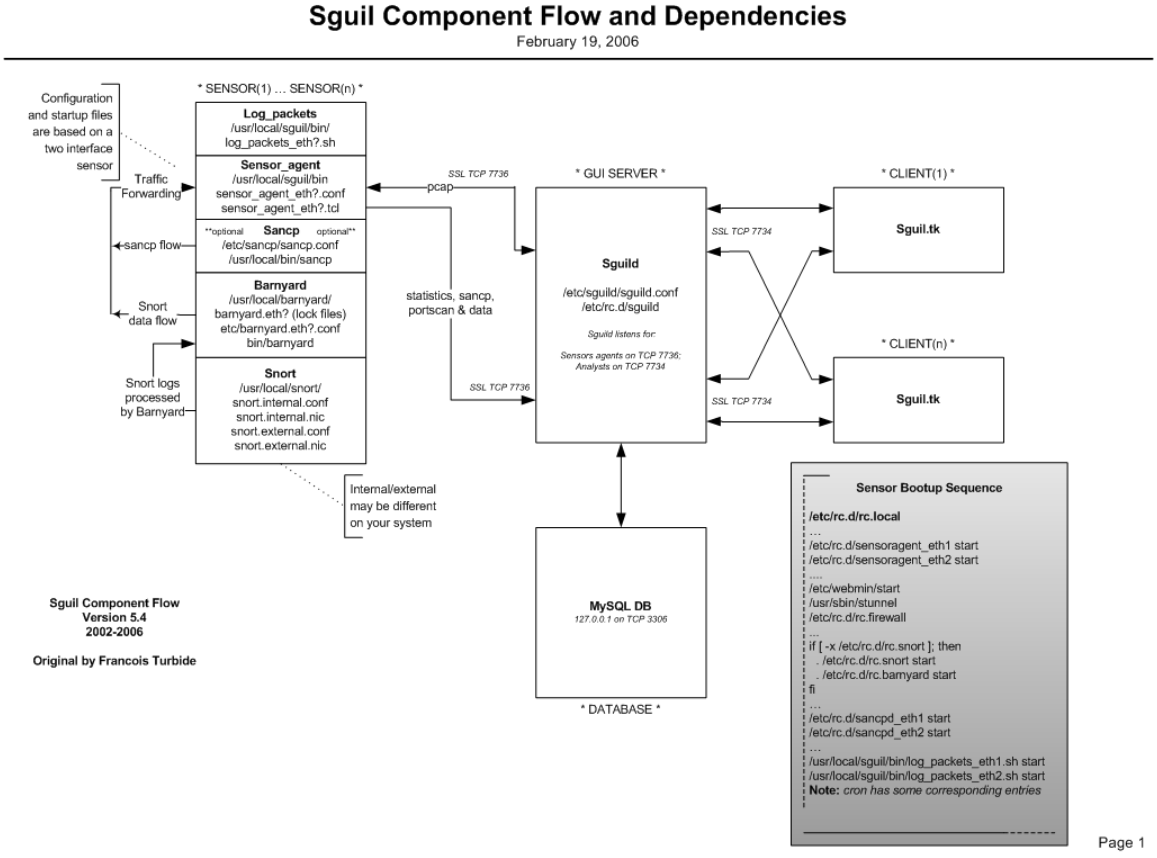
OINKMASTER 17
REGISTER WITH SNORT TO GET AN ACCOUNT 17
PATH 18
SNORT IDS MAIN SCREEN 19
EDIT OINKMASTER 19

SNORT WITH SGUIL FILES AND SCRIPTS 20

SANCP CUSTOM QUERIES 22

SOURCE AND DESTINATION PORT SEARCH 22
SOURCE AND DESTINATION PORT SEARCH FOR ONLY SID 2 22
SINGLE DAY SOURCE AND DESTINATION IP SEARCH FOR SID 2 ONLY 22

SPECIFIC SOURCE IP 22
SPECIFIC SENSOR (SID=2) AND SOURCE PORT 22
SNORT IDS CONFIGURATION 23
NOTE ABOUT SGUIL 23
SGUIL SERVER TROUBLESHOOTING 25
SGUIL WEB REPORTS 27
SETTING BOND0 NETWORKING 27
SETTING BOND0 NETWORKING FOR VLANS 27
SETTING UP NTOP FOR TRAFFIC MONITORING 28
BACKGROUND INFORMATION ABOUT THIS SETUP 30
OPERATING SYSTEM PATCHES 30
SLACKWARE PATCH MAINTENANCE SCRIPT 30
NETWORK GREP (NGREP) CONFIGURATION 35
REFERENCES 36



Introduction

This configuration process is used to deploy Snort sensors with the information managed through a Sguil console 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 ~400 MB in size and provides no remote services except through Secure Shell and Webmin for remote management of the sensor and the server. The Snort sensor logs are process via **Barnyard** backend processing.

This installation contains three separate and ready to use Sguil packages that contain all the necessary files to install Sguil as a sensor only (sensguil package), database only (sguildb package), or all-in-one systems (sguil package). Sguil contains some very useful analysis function such as the Security Analyst Network Connection Profiler (sancp) which collects statistical network traffic information, it has a script to log all the packets in pcap format, it uses tcpflow and p0f to get TCP session transcripts, it uses the Passive Asset Detection System (PADS) to collect banners on host services it sees to be used for correlation and uses Wireshark for in-depth packet analysis.

There is a sguil.pdf document that explains how to setup Sguil and the Sguil client on a Windows workstation. Additional information including console snapshots can be viewed at: http://sguil.sourceforge.net.

This installation has a web management interface called Webmin which is used to remotely manage the MySQL database and Snort sensor via a SSL enabled web browser. Additional information about Webmin is available at: http://www.webmin.com/. In addition, a Webmin Snort plugin to fully manage the Snort sensor (config file, plugins, oinkmaster, ruleset, etc.) to ease remote management of the sensor. See the Webmin section to correctly configure this package.

I recommend using **a minimum of 1GB of RAM** to function efficiently. If you have less than that, uncomment one of the memory options in the Snort configuration file located in /usr/local/snort/etc/snort.external.conf to ensure the sensor function properly. The most common memory option used is either search-method ac-bnfa or lowmem for systems with 512 MB of RAM. **This package is built for a sensor that contains 2 NIC cards** (eth0 = control and eth1/bond0 = Snort).

In order to make Snort signature updates more flexible, I have included the oinkmaster script written by Andreas Östling available at http://oinkmaster.sourceforge.net/. The Snort sensor includes the Emerging Threats rules also updated daily via Oinkmaster. The rules are available at: http://www.emergingthreats.net/rules/. You should review the /usr/local/snort/oinkmaster.conf file for its configuration.

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.

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

All-in-one Database/Sensor Summary Installation

- Boot from install CD with correct kernel. Default is huge26smp.s which should work with all systems.
- Partition the drive(s) as per example
- Run setup and follow Install the Software
- After the installation completed, at the Slackware Linux Setup screen, select <Cancel>
- cd /cdrom/sguil
- Run **pkgtool**
- Select Current and install the package
- Remove CD (**eject**)
- Type **reboot** to restart system
- Review sguil.pdf documents for Sguil server/client installation
- Login system
- Edit with vi /usr/local/etc/pads.conf and change ***network*** variable to correct range
- Change MySQL root password (default blank)
  - *mysqladmin –u root password 'your-new-password-for-sql_user-root'*
- Change Webmin admin password
- */usr/local/webmin/changepass.pl /etc/webmin admin newpassword*
- Add a Sguil user for remote console access
  - /etc/sguild/sguild –adduser guy [Enter]
- Webmin access https://systemIP:10000 (default is user: admin, password: admin)
- Configure TCP Wrappers for SSH access using Webmin (Servers, TCP Wrappers)
- Configure the sensor through Webmin (Servers, Sguil Sensor Controls, Barnyard eth1 config) to change **shadow** to real sensor name and remove # output sguil.
- In Sguil Sensor Controls, *Restart all Sensor Services*
- See Client Configuration to access the database with a Windows workstation
- See Shadow Hourly Web Reports to configure Shadow like reports

Database Summary Installation

- Boot from install CD with correct kernel. Default is huge26smp.s which should work with all systems.
- Partition the drive(s) as per example
- Run setup and follow Install the Software
- After the installation completed, at the Slackware Linux Setup screen, select <Cancel>
- cd /cdrom/sguildb

- Run **pkgtool**
- Select Current and install the package
- Remove CD (**eject**)
- Type **reboot** to restart database server
- Review sguil.pdf documents for Sguil server/client installation
- Login system
- Change MySQL root password (default blank)
  - *mysqladmin –u root password 'your-new-password-for-sql_user-root'*
- Change Webmin admin password
- */usr/local/webmin/changepass.pl /etc/webmin admin newpassword*
- Add a Sguil user for remote console access
  - /etc/sguild/sguild –adduser guy [Enter]
- Webmin access https://systemIP:10000 (default is user: admin, password: admin)
- Configure TCP Wrappers for SSH access using Webmin (Servers, TCP Wrappers)
- See Client Configuration to access the database with a Windows workstation

Sensor Summary Installation

- Boot from install CD with correct kernel. Default is huge26smp.s which should work with all systems.
- Partition the drive(s) as per example
- Run setup and follow Install the Software
- After the installation completed, at the Slackware Linux Setup screen, select <Cancel>
- cd /cdrom/sensguil
- Run **pkgtool**
- Select Current and install the package
- Remove CD (**eject**)
- Type **reboot** to restart sensor
- Review sguil.pdf documents for Sguil server/client installation
- Login system
- Edit with vi /usr/local/etc/pads.conf and change ***network*** variable to correct range
- Change Webmin admin password
- */usr/local/webmin/changepass.pl /etc/webmin admin newpassword*
- Webmin access https://systemIP:10000 (default is user: admin, password: admin)
- Configure TCP Wrappers for SSH access using Webmin (Servers, TCP Wrappers)
- Configure the sensor through Webmin (Servers, Sguil Sensor Controls, Barnyard eth1 config) to change **shadow** to real sensor name and remove # output sguil.
- Configure sensor to log data into MySQL as per "Configure SSL Between a Sensor and a Sguil Server section" (Servers, Sguil Sensor Controls, Sensor Agent eth1 config file) and change **127.0.0.1** to correct database server IP.
- In Sguil Sensor Controls, *Restart all Sensor Services*
- See Shadow Hourly Web Reports to configure Shadow like reports

Client Configuration

Download the Windows Sguil client at:
http://sourceforge.net/project/showfiles.php?group_id=71220
Unpack in C:\sguil-0.7.0

Download Windows Active TCL at: http://www.activestate.com/Products/ActiveTcl/
Install at c:\tcl

Download Windows TLS libraries at: http://tls.sourceforge.net
Unpack in C:\tcl\lib

Download Wireshark for Windows at: http://www.wireshark.org
Install at C:\Wireshark

If you prefer Firefox instead of Explorer, download at: http://www.mozilla.org/
Install at C:\Firefox

Sguil sguil.conf update

Edit c:\sguil-0.7.0\client\sguil.conf → Configure according to your Windows settings

```
# Change SERVERHOST to the correct IP or servername
set SERVERHOST 192.168.30.4

#  Enable Ext DNS
set EXT_DNS 1

# Define the external nameserver to use. OpenDNS list 208.67.222.222 and 208.67.220.220
set EXT_DNS_SERVER 208.67.222.222

# Define a list of space separated networks (xxx.xxx.xxx.xxx/yy) that you want to use the OS's
#resolution for.
set HOME_NET "192.168.0.0/16 10.0.0.0/8"

# Path to wireshark (ethereal) win32 example
set WIRESHARK_PATH "c:/wireshark/wireshark.exe"

# Where to save the temporary raw data files on the client system You need to remember to
# delete these yourself.
# win32 example
set WIRESHARK_STORE_DIR "c:/tmp"

# Favorite browser for looking at sig info on snort.org win32 example (IE)
set BROWSER_PATH c:/progra~1/intern~1/iexplore.exe              or
set BROWSER_PATH "c:/firefox/firefox.exe"

# Mailserver to use for emailing alerts
set MAILSERVER mail.example.com

# Default From: address for emailing
set EMAIL_FROM foo@example.com
```

Note: The TLS libraries are used to encrypt the session between the Windows client and the database server.

Create the C:\TMP directory to store Wireshark files if it doesn't already exist

Client Access to Database

The client can access the database at this point by executing the sguil.tk. However, sguil.tk must be associated with the "wish application" before it will start.

c:\sguil-0.7.0\client\sguil.tk

Installation, Configuration and Partitioning the Drive

Drive partitioning can be done in multiple ways. This is an example that can be used if you have two drives and you are installing both the database and server on the same system:

Database and Sensor

Drive One

/	Minimum of 750 MB	
swap	Minimum of 512 MB	
/usr/local	Remainder of drive	(Contains MySQL DB and Snort)

Drive Two

/LOG	Whole drive	(Contains the tcpdump Sguil logs)

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

To partition the drive, login as root and run ***cfdisk /dev/hda*** (IDE drive), ***cfdisk /dev/sda*** (SCSI drive) or ***cfdisk /dev/cciss/c0d0*** (Raid drive). If this isn't a new drive, delete the old partitions before starting.

- hda1: / = 750 MB (Select new, select primary, size is 750, beginning, bootable)
- hda2: SWAP = 512 MB or same amount as the RAM (Select Pri/Log Free Space, new, primary, size is 512, beginning)
- Change hda2 to swap by selecting *type 82*
- hda3 (Select Pri/Log Free Space, new, primary, for MySQL and Snort)
- *hda4 (Select Pri/Log Free Space, new, primary, remainder of disk for pcap logs)*
- Select Write to save the new settings to disk
- Select *Quit* to exit

Large Disk RAID Array Installation

It is possible to install this system on a large disk array. This process has been tested with an HP Server DL 580 G5 using HP Storageworks MS 70 using 4 shell containing each 25 drive @ 146GB on a Smart Array P800/512 BBWC Controller.

- Configure the first logical drive with 2 disks RAID 1+0
- Install the base OS with the RAID drive as per Database and Sensor configuration (above)
- Do not install the Sguil components
- Reboot the system

Configure rc.S

- cd /etc/rc.d
- vi rc.S
- Find "Mount usbfs if we're not using hotplug" and add the following line:
  /usr/sbin/partprobe (save and exit)

Configure Large Disk Array

- Execute parted /dev/cciss/c0d1 (second disk in the array)
- mklabel *gpt* <enter>
- p <enter> (This shows the disk geometry in megabytes or terabytes
- *mkpart*
  Partition type? [primary] primary, logical or extended
  File system type? [ext2]? ext2
  Start? 0
  End? 13.5tb (put size here from printout)
- q <enter>
- *mkfs.xfs /dev/cciss/c0d1p1*
- *sfdisk -l* (list partitions)

Add the partition to the /etc/fstab

- vi /etc/fstab
- mkdir /test
- mount /dev/cciss/c0d1p1 /test
- cd /LOG
- mv * /test
- Add a new line after the last /dev/cciss as follow:
  /dev/cciss/c0d1p1 /LOG xfs defaults 1 2 (Save and exit)
- reboot
- mount /mnt/cdrom /mnt/cdrom
- cd /mnt/cdrom/sensguil
- pkgtool (Install current package)
- reboot

The various Linux file systems are debatable. XFS is considered better with large files (/LOG) . ReiserFS and ext3 are made more for desktop.

Note on MySQL Drive size estimate:

Recommend no more than about 250GB for the MySQL data. To keep ~500,000,000 rows of SANCP data and ~2,500,000 alerts, a MySQL database will need ~180GB of disk space.

Install the Software

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
- Check swap partitions for bad blocks (It's a choice here): yes or no
- Swap partition configured
- Select Linux installation partition
    - /dev/hda1 (format, ext3 - default)
    - /dev/hda3 (format, ext3 - default)
    - /dev/hda4 (format, ext3 - default)
- Select mount point for /dev/hda3: /usr/local → **Needed for Snort/MySQL**
- Select mount point for /dev/hda4: /LOG → **Needed for Sguil pcap logs**
    - 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 or DVD drive
- Make sure the CD or DVD is in the CD-ROM drive and select OK
- Select *Yes* on continue
- Scan for the CD or DVD drive

- Install Snort with Sguil from the installation CD which only shows 5 packages:

A, AP, D, N, TCL

- 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 you booted with
- Select *huge26.s* for kernels that don't require hyper treading or multiple CPUs
huge26SMP.s for kernel that require hyper threading or multiple CPUs.
- Make a boot disk for recovery (LILO) or skip it
- After the boot disk, choose continue with the configuration
- Skip modem configuration (*no modem*)
- Enable Hotplug/Udev subsystem at boot? *Yes*
- Install LILO and select *expert*
    - Select Begin, at the blank prompt press enter, select *default*, install to *MBR* confirm location to install lilo (select default @/dev/hda, /dev/sda or /dev/cciss/c0d0) and none
    - Add Linux and choose the root partition (i.e. /dev/hda1, /dev/sda1 or /dev/ccisss/c0d0p1)
    - 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
- Confirm the network setup
- Confirm startup services to run (use default), select *Enter*
- Setup the hardware clock
- Setup the root password
- After the installation completed, at the Slackware Linux Setup screen, select <*Cancel*>
- Depending of type of installation: cd /cdrom/sguil (DB and sensor combo) or sguildb (DB only) or sensguil (sensor only)
- Run **pkgtool**
- Select Current and install the package
- Remove CD (**eject**)
- 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*

TCP Wrapper - SSH

Configure SSH TCP Wrappers via Webmin or in the following way:

- vi /etc/hosts.allow (Webmin, Servers, TCP Wrappers)
- 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

IPTables Firewall

Configure iptables firewall (rc.firewall)

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

O- Configure the firewall located in */etc/rc.d* directory or create your own
O- Edit the firewall script 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 at this time execute: */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 fails, 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)

If the sensor is using two different NICs, it will only detect the first one it sees and you will need to add the NIC module for the second one to the rc.netdevice file. You can look into the /etc/rc.d/rc.modules file for the NIC module and test it at the command line to see if it loads as follows:

```
modprobe eepro1000
dmesg
```

After dmesg displays its output, you should see the card loaded at the end of the dmesg output. If it loaded correctly, then add it to the rc.netdevice file to ensure it loads when the sensor or server reboots. If the NIC you are using doesn't appear in the rc.modules list, look at the list in /lib/modules/2.6.x/kernel/drivers/net where 2.6.x (x = kernel revision number) and repeat the above process until you find the right module to load your card.

You should have two modules listed. The order they are listed is the order they will load.

```
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

# Targa 3 10/100/1000 card (eth2):
/sbin/modprobe tg3
```

Mounting USB Drive

To mount a USB drive with this OS, plug in the USB drive and do the following:

```
dmesg |grep sda, sdb, sdc or sdd
mount /dev/sda? /mnt/hd        Where ? = the partition and usually 1
cd /mnt/hd                     You can copy or move files from this directory
umount /usb                    When done with the USB drive
```

Note: the device can be sda, sdb, sdc, sdd and usually partition 1 (i.e. sda1)

Snort Sensor Memory Limitation

You can limit the amount of memory used by the sensor by uncommenting one of the memory options in /usr/local/snort/etc/snort.external.conf (eth1) or snort.internal.conf (eth2) or using Webmin under Server to make the change. Suggested configuration for low memory and high performance is:

config detection: search-method ac-bnfa

Webmin Configuration

Webmin is a secure remote sensor and console manager. For example, the IDS can be remotely managed via an SSL enabled browser to manage MySQL, Sguil, the logs, the entire Snort IDS including its plugins, the rules, the configuration files, the restart the sensor services, view a daily Sguil report, etc. It is quite versatile and very easy to use for those who prefer using a GUI to manage their sensor.

After you log into Webmin, to manage the server and the Snort sensor is located in the **Servers** section. You then are going to see Snort IDS Admin eth1 to eth4, etc. The eth1 and eth2 are the only two-interface preconfigured in Webmin.

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

Configuring Barnyard to send to Sguil database

Barnyard Version 0.2.0 (Build 32)
6 December 2005

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-tcl Support for TCL

Configure Barnyard to Forward Data to Sguil Database

The Snort 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.

The default account is Sguil MySQL database account is *sguil* and the password is *password*. These can be changed if you desire but the whole sensor is configured to work with this account with the following minor configuration changes. These changes can be done at the command line or via Webmin in the Server tab. Configure Barnyard to forward data from a sensor to the Sguil database in the following manner:

vi /usr/local/barnyard/etc/barnyard.eth1.conf (External interface)
vi /usr/local/barnyard/etc/barnyard.eth2.conf (Internal interface)
vi /usr/local/barnyard/etc/barnyard.bond0.conf (External bond0 interface)

set the hostname (used for the sguil db output plugin)
config hostname: *shadow*

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

output log_pcap

Enable Data Insert into Sguil Database

Use this configuration if using Sguil

output sguil: agent_port 7735 (eth1/bond0) or output Sguil: agent_port 7745 (eth2)

- Save the file and restart Barnyard

Note: Barnyard connects to the database via the snort_agent_eth1 or snort_agent_bond0 startup scripts. The snort_agent configuration files are located in /etc/sguil in the snort_agent_eth1.conf and sensor_agent_bond0.conf files in the *SET BY_PORT 7735* and require no other configurations.

/etc/rc.d/rc.barnyard restart

- Ensure the Barnyard processes are running

ps –aef |grep barnyard

Configure SSL between a Sensor and a Sguil Server

If your configuration includes a sensor and a remote database server, you must configure the sensor to know where the database server is located. To allow the sensor to communicate with a Sguil database server, follow these steps:

- Log into Webmin
- Select the Server tab
- Select Sguil Sensor Controls
    - Select Snort Agent eth1 config file or Snort Agent bond0 config file
    - Select PADS Agent eth1 config file
    - Select PCAP Agent eth1 config file
    - Select SANCP Agent eth1 config file
- Modify "**set SERVER_HOST 127.0.0.1**" and change the IP to the Sguil Server database IP and save changes
- Restart **Restart snort_agent_eth1** or **Restart snort_agent_bond0**

Local sid Rule Mapping

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 and can be done via Webmin. These must match the information put in the rule file as follow:

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

Shadow Hourly Web Reports

Note: This should be configured only on an external sensor and the reports can only be viewed on the sensor. 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 for external (eth1 NIC) or internal (eth2 NIC) to run the hourly reports saved in Webmin, Servers/ Sguil Reports

Process Sguil logs
3 * * * * /usr/local/SHADOW/fetchem.pl -l *external*
10 * * * * /usr/local/SHADOW/fetchem.pl -l *internal*

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

- cd /usr/local/SHADOW/filters
- cd *external or internal*

In Webmin, edit using vi the following filters (to configure see *ShadowFilters.pdf*):

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

Change the net 172.21 to the correct network address.

Test your configuration in debug mode

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

New procedures for Snort Oinkmaster Updates

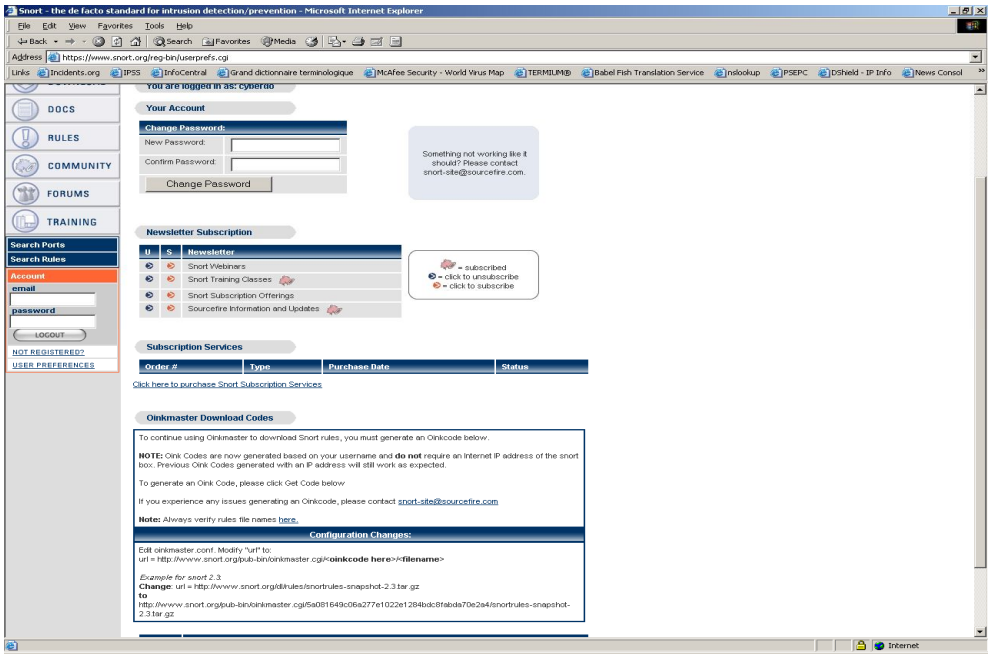
Oinkmaster

You should read the FAQ on how to configure oinkmaster to download the signature updates. http://oinkmaster.sourceforge.net/

Register with Snort to get an account

https://www.snort.org/pub-bin/register.cgi

You will receive your account information via e-mail
Login the site as per e-mail and password
Change your password if you want
In order for Oinkmaster to download the updated rules, you must generate a site string using the Get Code at the bottom of the page



As per the instructions at the bottom of the page, copy the new download path with your sting in it into oinkmaster.conf

Path (code is invalid, it is just an example):

url = http://www.snort.org/pub-bin/oinkmaster.cgi/**5a081649c06a277e1022e1284bdc8fabda70e2a4**/snortrules-snapshot-Current.tar.gz

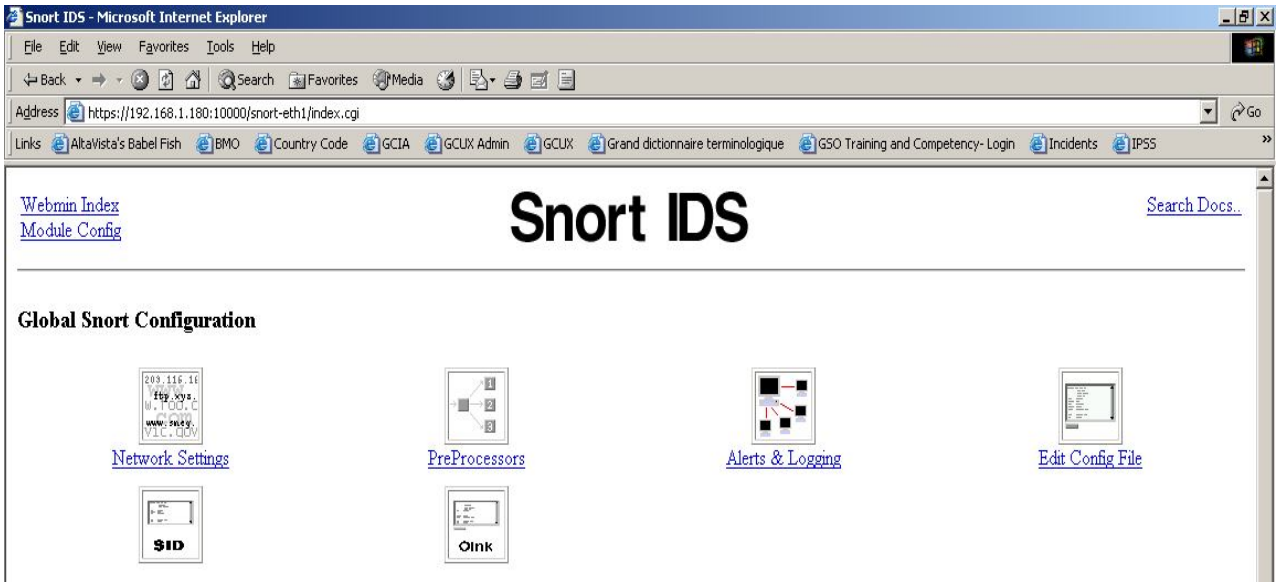
- The bold portion is where your code goes.
- The file is located at: /usr/local/snort
- The oinkmaster.conf file can be updated using vi or with Webmin
- If using Webmin, goto Servers, Snort IDS Admin eth1, select Oink
- Find url = http://www.snort.org/dl/rules/snortrules-snapshot-2.3.tar.gz

Change it to:

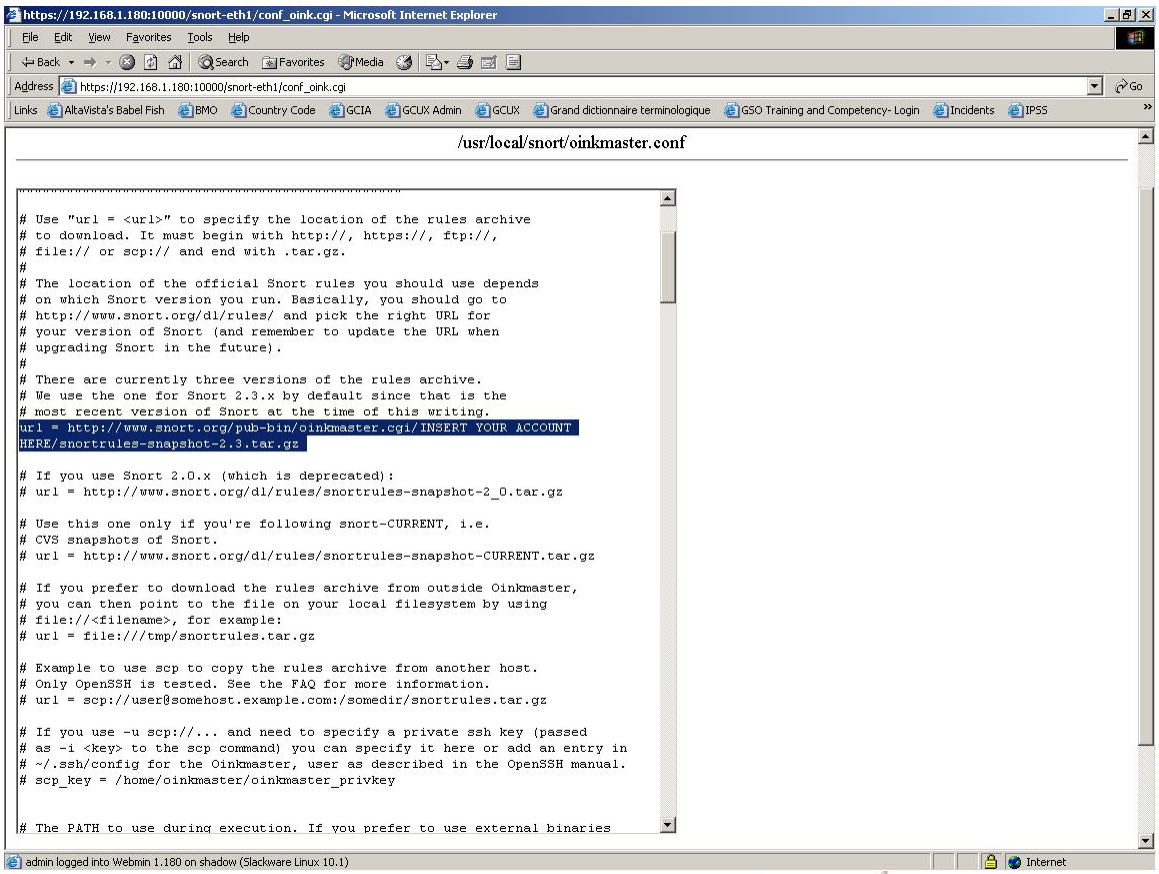
url = http://www.snort.org/pub-bin/oinkmaster.cgi/**INSERT YOUR ACCOUNT HERE**/snortrules-snapshot-Current.tar.gz

Webmin screenshots below

Snort IDS Main Screen



Edit Oinkmaster



Snort with Sguil Files and Scripts

/etc/rc.d	All system start/stop scripts
/etc/rc.d/rc.K	Kill all system script
/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.snort	Snort start/stop script
/etc/rc.d/rc.barnyard	Barnyard start/stop script
/etc/rc.d/rc.sguild	Sguil database start/stop script
/etc/rc.d/sancpd_eth1	sancp start/stop script for eth1
/etc/rc.d/pads_agent_eth1	Sguil PADS agent start/stop script to connect to database
/etc/rc.d/pcap_agent_eth1	Sguil pcap agent start/stop script to connect to database
/etc/rc.d/sancp_agent_eth1	Sguil sancp agent start/stop script to connect to database
/etc/rc.d/snort_agent_eth1	Sguil snort agent start/stop script to connect to database
/etc/rc.d/rc.mysqld	MySQL database script
/etc/rc.d/rc.netdevice	NIC module loading script
/etc/rc.d/rc.local	Script for all other configuration
/etc/issue	Banner message
/etc/motd	Banner message
/etc/rc.d/rc.firewall	Setup firewall
/var/adm/messages	General log file
/var/adm/syslog	Syslog file
/var/run	Various server pid files
/usr/local/snort	Snort directory files
/usr/local/snort/etc	Snort configuration scripts
/usr/local/snort/rules	Snort rules
/usr/local/snort/log	Snort log directory
/usr/local/snort/bin	Snort sensor binary
/usr/local/barnyard	Barnyard directory files (including Barnyard lock file)
/usr/local/barnyard/etc	Barnyard configuration scripts
/usr/local/barnyard/log	Barnyard log directory
/usr/local/barnyard/bin	Barnyard binary
/usr/local/etc/pads.conf	Pads configuration file
/etc/sguild	Sguil server configuration scripts
/etc/sguil	Sguil sensor agents configuration scripts.
/etc/sguild/incident_report.tcl	Daily Sguil web report output in Webmin/Servers
/usr/local/sguil/bin	Sguil logging scripts (log_packets_eth1.sh)
/usr/local/sguil/archive	Sguil pcap archives from sensors (delete regularly)
/LOG/external	Sguil log directory (eth1)
/LOG/external/dailylogs	Sguil daily logs collected by log_packets_eth1.sh
/LOG/external/portscans	Snort portscan dump directory
/LOG/external/sancp	sancp log dump directory
/LOG/external/pads	PADS log dump directory
/LOG/internal	Sguil log directory (eth2)
/usr/local/mysql	MySQL database

/usr/local/SHADOW	Shadow directory files
/LOG/RAW/gmt	Shadow log files
/usr/local/NGREP	Network Grep sensor files
/root/sguil_pcap.sh	Sguil pcap files search script for /LOG/external/dailylogs
/root/shadow_save.sh	Shadow pcap files search script for /LOG/RAW/gmt
/root/epoch.pl	Convert Snort pcap files epoch time to YYYYMMDDHH

sancp custom queries

SID 1 – shadow
SID 2 – Snort1

Source and Destination Port Search

WHERE sancp.start_time > '2005-01-05' AND (sancp.src_port = '11768') OR (sancp.dst_port = '11768') LIMIT 500

Source and Destination Port Search for Only SID 2

WHERE sancp.sid= 2 AND sancp.start_time > '2005-01-20' AND (sancp.src_port = '15118') OR (sancp.dst_port = '15118') LIMIT 500

Single Day Source and Destination IP Search for SID 2 only

WHERE sancp.sid=2 AND sancp.start_time > '2005-01-15' AND sancp.end_time < '2005-01-16' AND (sancp.src_ip = INET_ATON('192.168.158.186') OR sancp.dst_ip = INET_ATON('192.168.158.186')) LIMIT 500

Specific Source IP

WHERE sancp.start_time > '2005-03-15' AND (sancp.src_ip = INET_ATON('192.168.8.89')) LIMIT 500

Specific Sensor (sid=2) and Source Port

WHERE sancp.sid=2 AND sancp.start_time > '2005-03-30' AND (sancp.src_port='53') LIMIT 500

Snort IDS configuration

Snort Version 2.8.4 (Build 26)
9 April 2009

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

Note about Sguil: You can install any of the 3 packages (sguil, sguildb and sensguil) available on the CD which has been expressly built to run on this sensor. Check the document called **sguil.pdf** in the rel_note on how to configure each of the packages. Snort is configured to automatically use Barnyard through its output log_unified function.

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

--enable-sourcefire	Enable Sourcefire specific build options
--enable-stream4udp	Enable UDP session tracking in Stream4
--enable-dynamicplugin	Enable to dynamically load plugin, rules lib, detection eng
--enable-perfprofiling	Enable preprocessor and rule performance profiling
--enable-timestats	Enable TimeStats functionality
--enable-linux-smp-stats	Enable statistics reporting through proc
--enable-flexresp	Flexible response on hostile connection attempt
--enable-aruba	Enable Aruba output plugin
--enable-gre	Enable GRE and IP in IP encapsulation support
--enable-memory-cleanup	Enable memory cleanup upon exist
--enable-targetbased	Enable support in Stream, Frag and rules
--enable-mpls	Enable MPLS support

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 interface should be listed in snort.internal.nic and snort.external.nic. The eth2 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(1-5). Default

installation will contain eth0 and eth1 directory.

- Test the configuration by running the ./**check_snort_eth1** and **check_snort_eth2** script. If the this is successful, start the sensor
- **/etc/rc.d/rc.snort start** script to starts the sensor

Sguil Server Troubleshooting

Note: All these checks must be done as root

1- If you are unable to login the Sguil server via the Sguil client, check to see if the Sguil daemon are running on the server. Execute the following command:

ps –aef| grep tclsh

There should be 3 lines showing the services running. If you only see one, kill that service before restarting the Sguil daemon. Execute the following and repeat the previous command:

pkill tclsh

The Sguil database should have 2 ports open to provide access to the client and the sensor. Execute the following command:

netstat –an | grep 773*

You should see TCP port 7734 used by the client and TCP port 7736 used by the sensor (i.e. snort, sancp, pads, pcap agent service). If one is missing, restart the Sguil daemon:

/etc/rc.d/sguild stop
/etc/rc.d/sguild start

2- If the Sguild daemon won't start, check if the MySQL database is running:

ps –aef | grep mysqld

If the database isn't running, restart the database:

/etc/rc.d/mysqld stop
/etc/rc.d/mysqld start

3- If you suspect the data from a sensor is not going into the database, you can stop the Sguil service and manually observe the traffic. Do the following to watch the traffic in realtime:

/etc/rc.d/sguild stop
/etc/sguild/sguild

If there are any errors, Sguil is going to crash where the error occurs. If no errors are observed, use CTRL-C to stop and restart the service.

The database has a cronjob that should be cleaning the database logs on the first of each month at 1 am. If for some reasons that Sguil daemon does not restart, it is possible the database has not been cleaned and you may want to run the script to see if this will solve the issues. The script will stop the Sguil daemon if running, keep the past 30 days of logs and restart the Sguil Daemon. Run the following script and check to see if the Sguil daemon is running:

/etc/sguild/cleansguil.php
ps –aef | grep tclsh

However, if you get the following error:

```
root@sensor:/etc/sguild# ./sguild
pid(2182)  Loading access list: /etc/sguild/sguild.access
pid(2182)  Sensor access list set to ALLOW ANY.
pid(2182)  Client access list set to ALLOW ANY.
pid(2182)  Adding AutoCat Rule: ||ANY||ANY||ANY||ANY||ANY||ANY||tag:
Tagged Packet||1
pid(2182)  Connecting to 127.0.0.1 on 3306 as sguil
pid(2182)  MySQL Version: version 4.1.13-log
pid(2182)  SguilDB Version: 0.11
pid(2182)  Creating event MERGE table.
pid(2182)  Creating tcphdr MERGE table.
pid(2182)  Creating udphdr MERGE table.
pid(2182)  Creating icmphdr MERGE table.
pid(2182)  Creating data MERGE table.
ERROR: loaderd: You appear to be using an old version of the sguil Database schema
that does not support the MERGE sancp table. Please see the CHANGES document for
more information.
```

If you get the above error, you can execute this script:

/root/scripts/fix_mysql.sh

If the script doesn't exist, execute the following commands:

```
/etc/rc.d/rc.sguil stop
mysql –usguil –psguil –h 127.0.0.1 –D sguildb
mysql> DROP TABLES sancp, event, tcphdr, udphdr, icmphdr, data;
```

That will remove the MERGE tables only. Your data will still be there
and the MERGE tables will get recreated when sguild starts.

```
/etc/rc.d/rc.sguil start
Login the client
```

Sguil Web Reports

The server containing the database creates by default a daily web report available through Webmin. Login Webmin in the report section https://server:10000/reports/ and look under Sguil_Reports. These reports are generated at 1200 daily on the traffic reported on the previous day. The reports are based on the events processed under the 7 incidents categories. Since all the events for the previous day won't be processed before the report is automatically generated, you can manually reprocess the previous' day report with the following command:

/etc/sguil/incident_report.tcl --start yesterday --end today

You can manually generate reports based on dates as well. Here is an example to create a monthly report for the previous month.

/etc/sguil/incident_report.tcl --start "2006-09-01" --end today "2006-10-01"

Setting bond0 networking

In order to setup eth1 and eth2 as bond0, you need to copy the following scripts:

This will enable all the bond0 scripts:
- cp /etc/rc.d/rc.local_bond0 /etc/rc.d/rc.local

- vi /usr/local/snort/etc/snort.external.nic
- Change the eth1 to bond0 and save the changes

- vi /usr/local/snort/etc/snort.internal.nic
- Delete eth2 from this file (dd and Shift ZZ). Save and exit

Change the log_packet script configuration in the root cron:

- crontab –e
- Commend out the log_packets script for eth1
- Uncomment the log_packets script for bond0

Reboot the system to enable bond0 interface

Setting bond0 networking for VLANs

This solution is used to remove the vlan IDs from the traffic and bound it together under one interface. This example removes the vlan tags and merges the traffic with regular traffic. It removes the VLAN tags from vlan 502 and 202 and merge the traffic with regular traffic. Use all the bond0 scripts instead of eth1 scripts for your sensor to work

properly.

Starting bond0

echo "Fire up bond0 now to start vlan collection and remove tags..."
/sbin/vconfig add eth1 502
/sbin/vconfig add eth1 202
/sbin/modprobe bonding
/sbin/ifconfig bond0 promisc -arp up
/sbin/ifenslave bond0 eth1 eth1.502 eth1.202

Setting up Ntop for Traffic Monitoring

If you only want to run Ntop on a harden platform, use the default installation without Sguil and enable Ntop as describe below.

Before you start ntop to monitor the network traffic passing through the sensor, you need to execute the following command to setup your admin password:

ntop -A (Enter your admin password here)

Edit the rc.local file on the sensor only or the all-in-one sensor:
vi /etc/rc.d/rc.local

Uncomment the last two lines to start ntop on the next reboot as a daemon. If you are using the default configuration, the default port is 3000.

To manually start ntop and configure /etc/rc.d/rc.local according to your network, do:

ntop -W 443 -w 0 -i eth1,eth2 -M -m 192.168.25.0/24 -p /usr/local/etc/ntop/service.list -d

-w	To start non SSL ports such as port 80
-W	Start SSLv2 port on port 443
-i	Configure list of interfaces separated by coma
-m *range*	Used to define the local subnet “192.168.1.0/24,192.168.25.0/24”
-M	Should be used if you are monitoring more than one interface
-p	Contains the list of ports that are showing as graphic on the main page
-d	To daemon mode

Important Note: Enable SSL 2 in your browser before you login the ntop webserver

https://sensorIP

After you login, Configure -> Admin -> Startup Options and login with your newly created admin account and password.

The default capture interface is set to eth1 in order to capture the same data as the sensor sees and restart ntop to enable the correct interface. Usually rebooting the sensor and login ntop. See the man pages for any other configuration alternative.

Enjoy!

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=2008

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:

telinit 1
cd /tmp/slackupdate
upgradepkg <patch>.tgz
(Do not apply kernel modules or new kernels patches. System will fail if applied)
telinit 3

Note with kernels: Do not apply any kernel or modules because this system has a custom kernel and modules and if those supplied by Slackware are applied, the system will fail.

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

# Starting eth2
#echo "Fire up eth2 now to start Shadow collection..."
#/sbin/ifconfig eth2 promisc -arp
#/sbin/ifconfig eth2 up

# This solution is used with network taps to agregate the
# date (send and receive) into one logical interface.
# Starting bond0
#echo "Fire up bond0 now to start Shadow collection..."
#/sbin/modprobe bonding
#/sbin/ifconfig bond0 promisc -arp up
#/sbin/ifenslave bond0 eth1
#/sbin/ifenslave bond0 eth2

# This solution is used to remove the vlan IDs from the
# traffic and bound it together under one interface.
# This example remove the vlan tags and merge the traffic
# with regular traffic.
# Starting bond0
#echo "Fire up bond0 now to start vlan collection and remove tags..."
#/sbin/vconfig add eth1 502
```

```
#/sbin/vconfig add eth1 202
#/sbin/modprobe bonding
#/sbin/ifconfig bond0 promisc -arp up
#/sbin/ifenslave bond0 eth1 eth1.502 eth1.202

#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 Stunnel..."
#/usr/sbin/stunnel

echo "Starting firewall..."
/etc/rc.d/rc.firewall

echo "Starting Sguild server..."
/etc/rc.d/rc.sguild start

echo "Starting Sancp system for eth1..."
/etc/rc.d/sancpd_eth1 start

echo "Starting PADS for eth1..."
/usr/local/bin/pads -c /usr/local/etc/pads.conf -D

#echo "Starting Sancp system for bond0..."
#/etc/rc.d/sancpd_bond0 start

echo "Starting Sguil packet logger for eth1..."
/usr/local/sguil/bin/log_packets_eth1.sh start

#echo "Starting Sguil packet logger for bond0..."
#/usr/local/sguil/bin/log_packets_bond0.sh start

echo "Starting all Sguil services..."
/etc/rc.d/sancp_agent_eth1 start
/etc/rc.d/snort_agent_eth1 start
/etc/rc.d/pads_agent_eth1 start
/etc/rc.d/pcap_agent_eth1 start
```

```
echo "Starting Snort sensor and Barnyard output processor..."

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

# echo "Starting Ntop..."
# /usr/local/bin/ntop -W 443 -w 0 -i eth1 -p /usr/local/etc/ntop/service.list -d

# Turn off PC Speaker
/sbin/rmmod pcspkr

# All done.
```

Update cronjob to start various Sguil components, update time, and cut new logs each hour (Pre-configured on CD and relative of what is installed):

Crontab running as Root

```
# Internet date
17 23 * * * /usr/sbin/ntpdate time-a.nist.gov > /dev/null 2>1&
18 23 * * * /sbin/hwclock -- systohc > /dev/null 2>1&

# Clean and optimize Database Sguil Tables on the 1st of the month
0 2 1 1-12 * /etc/sguild/cleansguil.php > /dev/null 2>1&

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

5 0 * * * /etc/rc.d/rc.snort stop > /dev/null 2>1&
5 0 * * * /etc/rc.d/rc.barnyard stop > /dev/null 2>1&
7 0 * * * /etc/rc.d/rc.snort start > /dev/null 2>1&
7 0 * * * /etc/rc.d/rc.barnyard start > /dev/null 2>1&

# Cut a new Sguil log on a hourly basis
# The log_packet_eth2.sh must be enabled if using two cards for the IDS
0 * * * * /usr/local/sguil/bin/log_packets_eth1.sh restart > /dev/null 2>1&
#0 * * * * /usr/local/sguil/bin/log_packets_eth2.sh restart > /dev/null 2>1&
#0 * * * * /usr/local/sguil/bin/log_packets_bond0.sh restart > /dev/null 2>1&

# Cut a new Sguil log on a hourly basis
# The log_packet_eth2.sh must be enabled if using two cards for the IDS
0 * * * * /usr/local/sguil/bin/log_packets_eth1.sh restart > /dev/null 2>1&
#0 * * * * /usr/local/sguil/bin/log_packets_eth2.sh restart > /dev/null 2>1&
#0 * * * * /usr/local/sguil/bin/log_packets_bond0.sh restart > /dev/null 2>1&
```

```
# Create a softlink to the logs in /LOG/external/dailylogs/$date
# to be used with the sguil_pcap.sh script
1 * * * * /root/epoch_eth1.pl >/dev/null 2>1&
#1 * * * * /root/epoch_eth2.pl >/dev/null 2>1&

# Sguil Daily incident report
# This report is saved in /usr/local/webmin/reports/Sguil_Reports
0 12 * * * /etc/sguild/incident_report.tcl --start yesterday --end today > /dev/null 2>1&

# Create Sguil Report web page
15 * * * * /usr/local/webmin/reports/ExplorerIndex.pl > /dev/null 2>1&
```

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
```

Network Grep (Ngrep) configuration

Ngrep Version 1.42
5 January 2005

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***

References

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/

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

Chaosreader to replay tcpdump traffic in HTML
http://users.tpg.com.au/bdgcvb/chaosreader.html