

Complete Sguil Installation Guide On OpenBSD 3.6 [20050113]

About the sguil

The analyst console for Network Security Monitoring which based mainly on snort and other small tools which works the best on their own function.

By combining all the tools, it manages to run the analyzing process more smoothly and accurate and makes network security monitoring possible. The claim of all other commercial product/IDS about perfect anomaly detection will never be achieved without interaction of human being. Sguil developers know the fact of it and offer the real time monitoring gui instead of web base gui so that analyst can intercept datas and events directly. Thanks to the wonderful developers of sguil and other small tools.

Sguil Diagram

The diagram visualizes the current structure of sguil which I downloaded from <http://sguil.sourceforge.net/diagram.txt>.

My setup will totally follow the structure of it which having mysql database, sguil sensor, and sguil server separated.

Personal Thinking - When come to implementation, my principle is Firewall and IDS supposed to be deployed securely so I choose OpenBSD. For web server and etc, freebsd and debian seems better choice, however I might say that all of that depends on what distro you familiar with the most and what kind of environments you are playing around with :).

OS choice - OpenBSD 3.6 [Free, Functional and Secure Operating System]

OpenBSD installation is not my concern in this document since the installation guide in www.openbsd.org is good enough. Worthless for me to reinvent the wheel.

About the OpenBSD kernel patching, check out my guide at <http://www.iosn.net/Members/platypus/guides/openbsdstablepatching.sxw>.

This installation guide is best to follow sequentially since the installation of packages and sources have dependencies issues.

Partitioning Strategy

I prefer the Richard's way instead of sguil team, I'm not following the snort_data convention but nsm. So please check out the documentation that written by Richard. This document is mainly refer to his freebsd sguil installation guide.

I have written this documentation while inspired most by his document(Sguil Installation Guide v 0.5.3_02).

Once again, thanks to Richard.

Below it's the partitioning for my system,if you have installed OpenBSD before, guess not hard for you to follow.

Filesystem	Size	Used	Avail	Capacity	Mounted on
/dev/wd0a	501M	40.3M	436M	8%	/
/dev/wd0h	501M	12.0K	476M	0%	/home
/dev/wd0i	3.0G	2.0K	2.8G	0%	/nsm
/dev/wd0d	250M	6.0K	238M	0%	/tmp
/dev/wd0g	2.0G	952M	961M	50%	/usr
/dev/wd0e	1006M	29.1M	927M	3%	/var

Note: Partition your swap based on your memory.

These are the three systems that I have used to install mysql db server, sguil sensor and sguil server.

Mysql database server

hostname = mysqlldb

ip address = 192.168.1.33

Sguil sensor

hostname = sensor

ip address = 192.168.1.44

Sguil server

hostname = center

ip address = 192.168.1.55

My network interface is rl0, you can find out through ifconfig -a command. Remember lo0 refers to loopback.

Feel free to follow the convention above :).

Before you start your sguil installation, in order to have smooth process, please verify all the steps below.

If you don't have dns in your network, you have to configure it in this way.

For example, mysql database host configuration would be

Configure Hostname

```
shell>echo "mysqlldb" > /etc/myname
```

Configure hostname resolution

Edit this two lines in the /etc/hosts using vi editor to feed your environment too

```
::1 mysqlldb.fqdn mysqlldb  
127.0.0.1 mysqlldb.fqdn mysqlldb
```

Then

```
shell>echo "192.168.1.44 sensor.fqdn sensor" >> /etc/hosts
```

```
shell>echo "192.168.1.55 center.fqdn center" >> /etc/hosts
```

Note: Remember to change you fqdn based on your domain. For example, if your domain is microhard.com so what you do is change it to mysqlldb.microhard.com :P.

Configuring static ip address

```
shell>echo "inet 192.168.5.33 255.255.255.0 NONE" > /etc/hostname.r10
```

Verifying configuration without restart your machine.

```
shell>sh /etc/netstart
```

Note: If you have DNS server, all the steps above are not critical. Follow the steps above for the sensor and sguil server host too.

Warning: For security purpose, you are required to shut down all the services which is not needed to secure your box such as sendmail service, however this is not included in this document, check out rc.conf indeed.

Important: To standardize the installation, please download all the source tarball to /usr/local/src directory. you are also required to create a user sguil for your system.

MySQL SERVER INSTALLATION(11th JAN 2005)

Desired OpenBSD package

- mysql-client-4.0.20 multithreaded SQL database (client)
- mysql-server-4.0.20 multithreaded SQL database (server)
- p5-DBD-mysql-2.9004 MySQL drivers for the Perl DBI
- p5-DBI-1.43 unified perl interface for database access
- p5-Net-Daemon-0.38 extension for portable daemons
- p5-PIRPC-0.2018 module for writing rpc servers and clients

Desire Source tarball

-sguil-server-0.5.3.tar.gz

Note: You only need the sql script which comes together with the sguil server source.

```
shell>setenv PKG_PATH ftp://ftp.openbsd.org/pub/OpenBSD/3.6/packages/i386/
```

```
shell>pkg_add ${PKG_PATH}mysql-server-4.0.20.tgz
```

Note: All the other packages mentioned above will be fetched while you install mysql server directly through internet since they are required to install mysql server. By default after you install the mysql package from openbsd, mysql server only allow localhost connection.

Remember don't set the password for root using mysqladmin, please ignore the command that being asked to run when you finish mysql server installation. However run the commands below.

```
shell>mysql -e "CREATE DATABASE sguildb"
```

Download the sguil server source

```
shell>cd /usr/local/src
```

```
shell>ftp http://ovh.dl.sourceforge.net/sourceforge/sguil/sguil-server-0.5.3.tar.gz
```

Untar the sguil server source

```
shell>tar xvzf /usr/local/src/sguil-server-0.5.3.tar.gz
```

Change directory to /usr/local/src/sguil-0.5.3/server/sql_scripts/

```
shell>cd /usr/local/src/sguil-0.5.3/server/sql_scripts/
```

```
shell>mysql -D < create_sguildb.sql
```

```
shell>mysql -e "GRANT INSERT, SELECT, UPDATE on sguildb.* to sguil@localhost"
```

```
shell>mysql -e "GRANT INSERT, SELECT, UPDATE on sguildb.* to sguil@fqdn"
```

```
shell>mysql -e "SET PASSWORD for sguil@localhost=password('secureone')"
```

```
shell>mysql -e "SET PASSWORD for sguil@fqdn=password('secureone')"
```

```
shell>mysql -e "SET PASSWORD for root@localhost=password('secureone')"
```

```
shell>mysql -p -e "SET PASSWORD for root@fqdn=password('secureone')"
```

```
shell>mysql -p -e "FLUSH PRIVILEGES"
```

Note: Fqdn stands for fully qualified domain name, for people who don't know what it is, google it.

To Allow remote connection from 192.168.1.0/24 or from any host to OpenBSD mysql database.(In case you deploy separate database server)

```
mysql> GRANT ALL PRIVILEGES ON sguildb.* TO 'sguil'@'192.168.1.%'  
IDENTIFIED BY 'secureone' with GRANT option;
```

```
mysql> GRANT ALL PRIVILEGES on sguildb.* TO 'sguil'@'%fqdn' BY 'secureone'  
with GRANT option;
```

```
mysql> GRANT ALL PRIVILEGES ON sguildb.* TO 'sguil'@'%' IDENTIFIED BY  
'secureone' with GRANT option;
```

Now you will have password secureone for root and sguil, replace secureone with any secure password. No root allows to connect remotely but locally.

The mysql database server for sguil is completed now :).

SGUIL SENSOR INSTALLATION

Desired OpenBSD package

- libnet-1.0.2a raw IP packet construction library
- mysql-client-4.0.20 multithreaded SQL database (client)
- gettext-0.10.40p1 GNU gettext
- libiconv-1.9.1 character set conversion library
- wget-1.8.2 retrieve files from the 'net via HTTP and FTP

Desired Source tarball

- sguil-sensor-0.5.3.tar.gz
- snort-2.3.0RC2.tar.gz
- barnyard-0.2.0.tar.gz
- pcre-5.0.tar.gz
- sancp-1.6.1.tar.gz
- tcl8.4.9-src.tar.gz
- oinkmaster-1.1.tar.gz

Create a needed directory

```
shell>mkdir /usr/local/src
```

To create directory /usr/local/etc and /usr/local/etc/snort at the same time

The /usr/local/etc/snort is used to store all the snort and barnyard related configuration files

```
shell>mkdir -p /usr/local/etc/snort
```

```
shell>cd /nsm
```

Note:Substitute the sensor to your sensor name

```
shell>mkdir sensor
```

```
shell>cd sensor
```

Create this four directories under /nsm/sensor

```
shell>mkdir dailylogs portscans rules sancp
```

Note:I don't create a ssn_logs directory since I use sancp, tweak it to feed your need.

To change the owner for the /nsm partition

```
shell>chown -R sguil.sguil /nsm
```

```
shell>chown -R sguil.sguil /var/log/snort
```

```
shell>chown -R sguil.sguil /usr/local/etc/snort
```

To install mysql client

```
shell>pkg_add ${PKG_PATH}mysql-client-4.0.20.tgz
```

Finish for mysql client :).

To install libnet

I recommend opensbsd package for this since it is the latest version.

```
shell>pkg_add ${PKG_PATH}libnet-1.0.2a.tgz
```

Finish for libnet :).

To install wget

```
shell>pkg_add ${PKG_PATH}wget-1.8.2.tgz
```

Finish for wget :).

Snort Installation

Download the latest stable source first,

```
shell>cd /usr/local/src
```

```
shell>ftp http://www.snort.org/dl/snort-2.3.0RC2.tar.gz
```

```
shell>ftp http://ovh.dl.sourceforge.net/sourceforge/sguil/sguil-sensor-0.5.3.tar.gz
```

Untar the source

```
shell>tar xvzf snort-2.3.0RC2.tar.gz
```

```
shell>tar xvzf sguil-sensor-0.5.3.tar.gz
```

```
shell>cd snort-2.3.0RC2/src/preprocessors
```

Backup this two files

Note: If you plan to use sancp to collect session data, patching for spp_stream4.c is not necessary.

```
shell>cp spp_portscan.c spp_portscan.c.bak
```

```
shell>cp spp_stream4.c spp_stream4.c.bak
```

Copy the snort patches from sguil sensor source

```
shell>cp /usr/local/src/sguil-0.5.3/sensor/snort_mods/2_1/* ./
```

```
shell>patch spp_portscan.c < spp_portscan_sguil.patch
```

```
shell>patch spp_stream4.c < spp_stream4_sguil.patch    #this is optional
```

Now we can proceed to compile snort, however please remember to have libnet installed first since we want to have support for flexible response.

```
shell>cd /usr/local/src/snort-2.3.0RC2
```

```
shell>./configure --enable-flexresp
```

```
shell>make
```

```
shell>make install
```

To store all the sensor rules in /nsm/sensor/rules, copy all the rules from snort source to the desired location

```
shell>cp /usr/local/src/snort-2.3.0RC2/rules/* /nsm/sensor/rules/
```

Copy the snort configuration file snort.conf to the desired directory

```
shell>cp /usr/local/src/snort-2.3.0RC2/etc/snort.conf /usr/local/etc/snort/
```

Change some of the variables to match your environment using vi editor, in my case

1. Set the rules path:

```
var RULE_PATH /nsm/sensor/rules
```

2. Enable the portscan preprocessor:

```
preprocessor portscan: $HOME_NET 4 3 /nsm/sensor/portscan sensor
```

3. Enable log_unified output:

```
output log_unified: filename snort.log, limit 128
```

4. Unless you want to use stream4 to capture session data, else the changes below can be ignored.

```
preprocessor stream4: detect_scans, disable_evasion_alerts, keepstats db  
/nsm/sensor/ssn_logs
```

5. Make other modifications to work best with your network. For example to monitor my internal LAN

```
var HOME_NET 192.168.1.0/24
```

Note: Please don't install snort from OpenBSD package since you need to patch it.

Some of the files in /usr/local/src/snort-2.3.0rc2/etc are being shared by snort and barnyard, so we should copy it to /nsm/sensor/rules

```
shell>cd /usr/local/src/snort-2.3.0RC2/etc/
```

Copy six files below to the /nsm/sensor/rules

1. classification.config
2. gen-msg.map
3. reference.config
4. sid-msg.map
5. threshold.conf
6. unicode.map

After you copied the file, make a symlink so that more easier to update rules and these 6 files using Oinkmaster.

```
shell>ln -s /nsm/sensor/rules/classification.config /usr/local/etc/snort/classification.config
```

```
shell>ln -s /nsm/sensor/rules/gen-msg.map /usr/local/etc/snort/gen-msg.map
```

```
shell>ln -s /nsm/sensor/rules/reference.config /usr/local/etc/snort/reference.config
```

```
shell>ln -s /nsm/sensor/rules/sid-msg.map /usr/local/etc/snort/sid-msg.map
```

```
shell>ln -s /nsm/sensor/rules/threshold.conf /usr/local/etc/snort/threshold.conf
```

```
shell>ln -s /nsm/sensor/rules/unicode.map /usr/local/etc/snort/unicode.map
```

Barnyard Installation

```
shell>cd /usr/local/src
```

```
shell>ftp http://www.snort.org/dl/barnyard/barnyard-0.2.0.tar.gz
```

```
shell>tar xvzf barnyard-0.2.0.tar.gz
```

```
shell>cd barnyard-0.2.0
```

Note: Richard has mentioned in his document that perhaps we need to modify the configure script for BSD system, however in my case I don't modify anything.

When running the configure script, use this syntax

```
shell>./configure --enable-mysql
```

```
shell>make
```

```
shell>make install
```

Copy the barnyard.conf to /usr/local/etc/snort

```
shell>cp /usr/local/src/barnyard-0.2.0/etc/barnyard.conf /usr/local/etc/snort/
```

To tell barnyard how to collect alert data from snort and insert it into mysql database, make the following changes to the barnyard.conf.

1. Configure the sensor hostname

```
config hostname: sensor
```

2. Configure the network interface

```
config interface: rl0
```

3. Configure the sguil output plugin

```
output sguil: mysql, sensor_id 0, database sguildb, server mysqldb, user sguil,\  
password xxxxxxxx, sguil_host center, sguil_port 7736
```

I have sguild server running on center, mysql database server on mysqldb and since I have only one sensor, so the sensor_id is 0 by default. If I have another sensor so what I need to do is creating a mysql entry for it using the syntax below.

```
mysql>INSERT into sensor set sid='2', hostname='sensor2';
```

Warning: Change the password to a secure one is highly recommended.

Oinkmaster installation

Oinkmaster is a perl script which help you to manage and update your snort ruleset.

```
shell>cd /usr/local/src
```

```
shell>ftp http://internap.dl.sourceforge.net/sourceforge/oinkmaster/oinkmaster-1.1.tar.gz
```

```
shell>cd oinkmaster-1.1
```

```
shell>cp oinkmaster.conf /usr/local/etc/snort
```

```
shell>cp oinkmaster.pl /usr/local/bin/
```

Normally you don't need any changes for oinkmaster.conf if you use snort-2.3.0RC2 or current.

Tcl Installation

```
shell>cd /usr/local/src
```

```
shell>ftp http://puzzle.dl.sourceforge.net/sourceforge/tcl/tcl8.4.9-src.tar.gz
```

```
shell>tar xvzf tcl8.4.9-src.tar.gz
```

```
shell>cd tcl8.4.9-src.tar.gz
```

```
shell>./configure
```

```
shell>make
```

```
shell>make install
```

This should be done quickly without any errors. Then create a symlink for the tclsh8.4

```
shell>ln -s /usr/local/bin/tclsh8.4 /usr/local/bin/tclsh
```

Sancp Installation

Downloading sancp

```
shell>cd /usr/local/src
```

```
shell>ftp http://www.metre.net/files/sancp-1.6.1.tar.gz
```

To install it,

```
shell>tar xvzf sancp-1.6.1.tar.gz
```

```
shell>cd sancp-1.6.1
```

```
shell>make
```

Then you will have sancp in your directory, just copy it to the /usr/local/bin

```
shell>cp sancp /usr/local/bin/
```

After that, copy sancp.conf from sguil distribution to /usr/local/etc/snort

```
shell>cp /usr/local/src/sguil-0.5.3/sensor/sancp/sancp.conf /usr/local/etc/snort/
```

Configure the HOME_NET variable for sancp.conf so that it matches your internal network, in my case

```
HOME_NET 192.168.1.0/24
```

Sensor_agent.tcl

Make the modification to the sensor_agent.conf in the /usr/local/src/sguil-0.5.3/sensor directory

1. Define the sguil server

```
set SERVERHOST center
```

2. Define the sguil sensor

```
set HOSTNAME sensor
```

3. Define the log directory

```
set LOG_DIR /nsm
```

4. To disable the session data collection using stream4 preprocessor

```
set S4_KEEP_STATS 0
```

5. Enabling the collection of session data using SANCP

```
set SANCP 1
```

Log_packets.sh

This shell script collects and manages full content data collected from a second instance of snort running as packet logger. Make the following changes to /usr/local/src/sguil-0.5.3/sensor/log_packets.sh

1. Define the sensor

```
HOSTNAME="sensor"
```

2. Define the log directory

```
LOG_DIR="/nsm"
```

3. Define the network interface

```
INTERFACE="rl0"
```

4. Uncomment the following line to let snort run as user sguil

```
OPTIONS="-u sguil -g sguil -m 122"
```

Cron

The log_packet.sh scripts should be run using the crontab.

```
shell>crontab -e
```

Add this line at the end of file

```
00 0-23/1 * * * /usr/local/src/sguil-0.5.3/sensor/log_packets.sh restart
```

This will complete the sguil sensor setup. Do not enable any components yet till the rest of architecture is up.

SGUILD SERVER INSTALLATION

Desired OpenBSD package

-mysql-client-4.0.20 multithreaded SQL database (client)
-p0f-1.8.3 passive OS fingerprinting tool
-tcpflow-0.21 tool for capturing data from TCP connections

Desired Source tarball

-sguil-server-0.5.3.tar.gz
-snort-2.3.0RC2.tar.gz

Note: You need snort source since you need to copy it's rules file into /nsm/rules/sensor, the other option is you can download only the rules file from snort.org.

TCL/TK suite

-tcl8.4.9-src.tar.gz
-tk8.4.9-src.tar.gz
-tcllib-1.7.tar.gz
-tclx8.3.5-src.tar.gz
-tls1.5.0-src.tar.gz
-myqltcl-2.51.tar.gz

Note: Tk is not required if you configure it with --enable-tk=NO when compiling Tclx.

Create a needed directory for the sguil server

```
shell>mkdir /nsm/archive #Used to hold pcap data retrived in response to user queries
```

```
shell>mkdir -p /nsm/rules/sensor #Create a directory for each separate sensor's rules
```

Change the ownership of the directory

```
shell>chown -R sguil.sguil /nsm
```

For mysql client, p0f and tcpflow installation,

```
shell>setenv PKG_PATH ftp://ftp.openbsd.org/pub/OpenBSD/3.6/packages/i386/
```

```
shell>pkg_add ${PKG_PATH}mysql-client-4.0.20.tgz
```

```
shell>pkg_add ${PKG_PATH}p0f-1.8.3.tgz
```

```
shell>pkg_add ${PKG_PATH}tcpflow-0.21.tgz
```

Finish installation for them.

To install pcre(not from OpenBSD package)

I recommend you compile the pcre instead of using openbsd package since you won't have any problem at all.

```
shell>cd /usr/local/src
```

```
shell>ftp ftp://ftp.csx.cam.ac.uk/pub/software/programming/pcre/pcre-5.0.tar.gz
```

```
shell>tar xvzf pcre-5.0.tar.gz
```

```
shell>cd pcre-5.0
```

```
shell>./configure
```

```
shell>make
```

```
shell>make install
```

Finish for pcre :).

Install Tcl and TK, tcllib, tclx, mysqltcl, tcltls on OpenBSD(8th JAN 2005)

Source File require

tcl8.4.9-src.tar.gz	http://tcl.sourceforge.net/
tk8.4.9-src.tar.gz	http://tcl.sourceforge.net/
tcllib-1.7.tar.gz	http://tcllib.sourceforge.net/
tclx8.3.5-src.tar.gz	http://tclx.sourceforge.net/
mysqltcl-2.51.tar.gz	http://www.xdobry.de/mysqltcl/
tls1.5.0-src.tar.gz	http://tls.sourceforge.net/

Check out all the File Distribution links from the url above in order to download the source to /usr/local/src.

I have tried to use the package for tcl and tk however have no success in compiling tclx and mysqltcl since too many errors and I'm not coder, so I decide to take some times to compile from source and it seems work properly.

Note: Don't use mysqltcl-3.01 version which is the latest, it only compatible with mysql 4.1 or later.

Tcl and Tk installation is very straight forward, just untar the source and run into the directory, running

```
shell>./configure
```

```
shell>make
```

```
shell>make install
```

So it will be done without error.

For tcllib,

```
shell>tar xvzf tcllib-1.7.tar.gz
```

```
shell>cd tcllib-1.7
```

```
shell>./configure
```

```
shell>make install
```

To install tclX

Don't try it if you install tcl and tk using the openbsd package, you will end up with nothing. Anyone who able to do so please send me the how-to. The other things that you need in order to install tclx from source is tcllib. Again, don't use the openbsd package.

```
shell>tar xvzf tclx8.3.5-src.tar.gz
```

```
shell>cd tcl8.3.5
```

```
shell>cd unix
```

```
shell>./configure
```

```
shell>make
```

```
shell>make install
```

To install mysqлтcl-2.51 for openBSD-3.6

Before you start installing it, remember to make a symlink for libmysqlclient.so

```
shell>cd /usr/local/src
```

```
shell>tar xvzf mysqлтcl-3.01.tar.gz
```

```
shell>cd mysqлтcl-2.51
```

```
shell>ln -s /usr/local/lib/mysql/libmysqlclient.so.12.0 /usr/local/lib/libmysqlclient.so
```

```
shell>env CC=gcc ./configure -with-mysql-include=/usr/local/include/mysql -with-  
mysql-lib=/usr/local/lib/
```

If you proceed without error like i do, then just go ahead.

```
shell>make
```

```
shell>make install
```

Note: I have tried whether mysqltcl-3.01 works with me or not but failed since in the mysqltcl website, it has mentioned that version 3.01 only compatible with mysql-4.1 or later. I only have mysql 4.0.20 so I give up on it.

For tcltls installation on openBSD-3.6

```
shell>tar xvzf tls1.5.0-src.tar.gz
```

```
shell>cd tls1.5
```

```
shell>./configure --with-tcl=/usr/local/lib --with-tcl-include=/usr/local/include --with-ssl-  
dir=/usr/
```

```
shell>make
```

```
shell>make install
```

This should work out without error.

To check whether you have install correctly,

```
shell>/usr/local/bin/tclsh
```

So you will be in tcl shell which is either % or tclsh>

```
tclsh>package require Tclx  
8.3
```

```
tclsh>package require mysqltcl  
2.51
```

```
tclsh>package require tls  
1.50
```

When you try to check the `tclx`, `tcltls` and `mysqltcl` which is required, it will show its version, that means you have installed correctly :)

Sguild Configuration

Download the sguild server source

```
shell>cd /usr/local/src
```

```
shell>ftp http://ovh.dl.sourceforge.net/sourceforge/sguil/sguil-server-0.5.3.tar.gz
```

```
shell>vi sguil-server-0.5.3/server/sguild.conf
```

Make changes to the `sguild.conf` to match your environment

1. Define rules path

```
set RULESDIR /nsm/rules
```

2. Define mysql database user

```
set DBUSER sguil
```

3. Define mysql database password

```
set DBPASS xxxxxxxx
```

The `xscriptd` functions are now configured within `sguild.conf`, so set the variable below.

1. Define pcap data storage that base on user queries

```
set LOCAL_LOG_DIR /nsm/archive
```

2. Define tcpflow executable path

```
set TCPFLOW "/usr/local/bin/tclflow"
```

3. Define p0f executable path

```
set P0F_PATH "/usr/local/bin/p0f"
```

While in the `/usr/local/src/sguil-0.5.3/server` directory, create a `sguil` user who will access the sguild server shortly

```
shell>./sguild -c sguild.conf -u sguild.users -adduser sguil
```

To encrypt the communication between the sguil client and server, please check out the Encryption section in the installation guide of Richard. It totally works on OpenBSD.

Now Sguil server is done.

Note: For sguil client, please refer to Sguil Client section by Richard too, I don't recommend you install sguil client on OpenBSD and better use other OS as your choice for ease of installation.

Now we have finished the sguil server installation, bingo :).

Starting All Components

Following what has been recommended by Richard, it's better to start each components in a separate terminal instead of running in daemon for the first time to understand the operation of sguil.

Mysql Database Server

```
shell>/usr/local/bin/mysqld_safe &
```

Note: You can configure mysql to start everytime system reboot, check it out at www.openbsd.org.

Sguil Server

Log in as user sguil,

Running Sguil

```
shell>cd /usr/local/src/sguil-0.5.3/server
```

```
shell>cd /usr/local/src/sguil-0.5.3/server
```

```
shell>./sguild -c sguil.conf -u sguil sguil.users -O /usr/local/lib/libtls150.so.1.0 -C /usr/local/etc/snort
```

-C specify the directory contain sguil.pem and sguil.key

-O specify the Openssl enable using path to tls

-c specify the sguil configuration file(sguil.conf)

-D run it in daemon mode

Note: After you familiar with sguild's behaviour, you can run it in daemon mode using the -D option

Sguil Sensor

Running Barnyard

Log in as user sguil,

```
shell>cd /usr/local/etc/snort
```

```
shell>barnyard -c barnyard.conf -d /nsm/sensor -g gen-msg.map -s sid-msg.map -f snort.log -w -waldo.file
```

-c specify configuration file

-d specify spool file directory

-g read the gen-msg.map file from

-s read the sid-msg.map file from

-f Use <base> as the base unified filename

-w Enable bookmarking using <file>

Running Sensor_agent.tcl

As user sguil,

```
shell>cd /usr/local/src/sguil-0.5.3/sensor
```

```
shell>./sensor_agent.tcl
```

Running Snort

As user root,

```
shell>ifconfig rl0 -arp up
```

```
shell>snort -u sguil -g sguil -c /usr/local/etc/snort/snort.conf -U -l /nsm/sensor -m 122 -A none -i rl0
```

-u specify to run as user

-g specify to run as group

-c specify the configuration file

-U use UTC for timestamps

-l log to directory

-m Set umask

-A Set alert mode #we choose none since we use barnyard

-i specify the network interface to use

Running Sancp

As user root,

```
shell>/usr/local/bin/sancp -d /nsm/sensor/sancp -i rl0 -u sguil -g sguil -c /usr/local/etc/snort/sancp.conf > /var/log/sancp.log
```

-d specify directory for output file

-i specify network interface to use

-u specify to run as user

-g specify to run as group

-c specify the sancp configuration file

Running Log_packets.sh

As user root,

```
shell>cd /usr/local/src/sguil-0.5.3/sensor
```

```
shell>./log_packets.sh start
```

Running oinkmaster

Make sure you have wget installed to get oinkmaster working.

```
shell>cd /usr/local/bin
```

```
shell>chmod 755 /usr/local/bin/oinkmaster.pl
```

```
shell>./oinkmaster.pl -C /usr/local/etc/snort/oinkmaster.conf -c -q -o /nsm/sensor/rules
```

- C Specify the configuration file
- c Careful mode (dry run) - check for changes but do not update anything
- q Quiet mode - no output unless changes were found
- o specify the directory which store snort rules
- b backup your existing rule file to the specific directory

If you want to update it directory, ignore the -c option.

Finally all done, congrate to yourself and get a break from the shell. At this moment, you can connect to the server using sguil client :).

Note: For sguil client, use the latest source which is sguil-client-0.5.3.tar.gz(BSD/*nix) or sguil-client-0.5.3.zip(WINS).

Future thought

I do wish sguil team offers sguil as port/package in OpenBSD to simplify installtion.

Reference

http://sguil.sourceforge.net/sguil_guide_latest.txt by Richard Bejtlich

Author

lee a.k.a as geek00L, you can reach me at cslee[at]misprai.mine.nu or snort-gui.