Building a Debian\Snort based IDS

Jason Weir – jason.weir@nhrs.org –8/19/2014

This document installs Debian Wheezy 7.6, Snort 2.9.6.2, Barnyard2-1.13, PulledPork 0.7.0 and BASE 1.4.5.

Document Roadmap:
1. Install OS and base software
2. Install Snort pre-requisites - libpcap, libdnet, and daq
3. Install, configure & test Snort
4. Setup MySQL database
5. Install & configure Barnyard
6. Configure Apache & PHP
7. Install, configure and test BASE
8. Startup script for Snort & Barnyard
9. Keep rules up to date with Pulledpork
10. What I left out

1. Install OS and base software

This document assumes 2 network cards with eth0 being the management interface and eth1 being the collector interface.

Get Debian here: http://www.debian.org/distrib/netinst. I used the small CD version. Burn the iso and boot the CD.

Choose the default options (or as appropriate for your site), when you get to the "Software Selection" screen, unselect all options to get a bare minimum install. After the install finishes, the CD ejects and the system will reboot. Log back in as root.

# apt-get update && apt-get -y install ssh vim – This is so we can connect via SSH and copy\paste to the terminal.

Dotdeb.org maintains packages of mysql and php more current than the Debian repository - do the following so apt can use them.

```
# vi /etc/apt/sources.list
```

Add the following lines:

```
deb http://packages.dotdeb.org wheezy all
deb-src http://packages.dotdeb.org wheezy all
```

Install the dotdeb GnuPG key:

```
# cd /usr/src && wget http://www.dotdeb.org/dotdeb.gpg
# cat dotdeb.gpg | apt-key add -
```

Apt will require input – for example MySQL will ask for you to enter a “root” password for the MySQL server. Make it secure and don’t forget it.

```
# apt-get update && apt-get -y install apache2 apache2-doc autoconf automake bison ca-certificates ethtool flex g++ gcc gcc-4.4 libapache2-mod-php5 libcrypt-ssleay-perl libmysqlclient-dev libnet1 libnet1-dev libpcre3 libpcre3-dev libphp-adodb libssl-dev libtool libwww-perl make mysql-client mysql-common mysql-server ntp php5-cli php5-gd php5-mysql php-pear sendmail sysstat usbmount
```

Disable ”Large Receive Offload” and ”Generic Receive Offload” on the collector interface

```
# vi /etc/rc.local
```

Add before “exit 0”

```
ethtool --offload  eth1  rx off  tx off
ethtool -K eth1 gso off
ethtool -K eth1 gro off
```

2. Install Snort pre-requisites - libpcap, libdnet, and DAQ

Install libpcap:

```
# cd /usr/src && wget http://www.tcpdump.org/release/libpcap-1.6.1.tar.gz
# tar -zxf  libpcap-1.6.1.tar.gz && cd libpcap-1.6.1
# ./configure --prefix=/usr && make && make install
```

Install libdnet:

```
# cd /usr/src && wget http://libdnet.googlecode.com/files/libdnet-1.12.tgz
# tar -zxf libdnet-1.12.tgz && cd libdnet-1.12
# ./configure --prefix=/usr --enable-shared && make && make install
```

Install daq:

```
# cd /usr/src && wget https://www.snort.org/downloads/snort/daq-2.0.2.tar.gz
# tar -zxf   daq-2.0.2.tar.gz && cd daq-2.0.2
# ./configure && make && make install
```

Update the shared library path

```
# echo >> /etc/ld.so.conf /usr/lib
# echo >> /etc/ld.so.conf /usr/local/lib && ldconfig
```

3. Install, configure & test Snort

```
# cd /usr/src && wget http://labs.snort.org/snort/2962/snort.conf
# wget https://www.snort.org/downloads/snort/snort-2.9.6.2.tar.gz
# tar -zxf snort-2.9.6.2.tar.gz && cd snort-2.9.6.2
# ./configure --enable-sourcefire && make && make install
# mkdir /usr/local/etc/snort /usr/local/etc/snort/rules /var/log/snort /var/log/barnyard2 /usr/local/lib/snort_dynamicrules
# touch /usr/local/etc/snort/rules/white_list.rules /usr/local/etc/snort/rules/black_list.rules /usr/local/etc/snort/sid-msg.map
# groupadd snort && useradd -g snort snort
# chown snort:snort /var/log/snort /var/log/barnyard2
# cp /usr/src/snort-2.9.6.2/etc/*.conf* /usr/local/etc/snort
# cp /usr/src/snort-2.9.6.2/etc/*.map /usr/local/etc/snort
# cp /usr/src/snort.conf /usr/local/etc/snort
```

```
# vi /usr/local/etc/snort/snort.conf
```

Change these lines:
Line #45 - **ipvar HOME_NET 172.26.12.0/22** – make this match your internal (friendly) network
Line #48 - **ipvar EXTERNAL_NET !$HOME_NET**
Line #104 - **var RULE_PATH ./rules**
Line #109 - **var WHITE_LIST_PATH ./rules**
Line #110 - **var BLACK_LIST_PATH ./rules**
Line #293 - add this to the end after "decompress_depth 65535" **max_gzip_mem 104857600**
Line #521 - add this line **output unified2: filename snort.log, limit 128**
Line #543 - delete or comment out all of the "**include $RULE_PATH**" lines except "**local.rules**"

```
# vi /usr/local/etc/snort/rules/local.rules
```

Enter a simple rule like this for testing:

```
alert icmp any any -> $HOME_NET any (msg:"ICMP test"; sid:10000001; rev:1;)
```

Now we can start and test snort.

```
# /usr/local/bin/snort -A console -q -u snort -g snort -c /usr/local/etc/snort/snort.conf -i eth0
```

Ping the management IP address from another machine, alerts should be printed to the console like this:

```
02/09-11:29:43.450236  [**] [1:10000001:0] ICMP test [**] [Priority: 0] {ICMP} 172.26.12.1 -> 172.26.12.2
02/09-11:29:43.450251  [**] [1:10000001:0] ICMP test [**] [Priority: 0] {ICMP} 172.26.12.2 -> 172.26.12.1
```

If so congrats – you have Snort working… Use ctrl-c to kill snort.

4. Install & configure Barnyard2

```
# cd /usr/src && wget https://github.com/firnsy/barnyard2/archive/master.tar.gz
# tar -zxf master.tar.gz && cd barnyard2-*
# autoreconf -fvi -I ./m4 && ./configure --with-mysql --with-mysql-libraries=/usr/lib/i386-linux-gnu && make && make install
# mv /usr/local/etc/barnyard2.conf /usr/local/etc/snort
# cp schemas/create_mysql /usr/src
```

```
# vi /usr/local/etc/snort/barnyard2.conf
```

Line #27 change to **/usr/local/etc/snort/reference.config**
Line #28 change to **/usr/local/etc/snort/classification.config**
Line #29 change to **/usr/local/etc/snort/gen-msg.map**
Line #30 change to **/usr/local/etc/snort/sid-msg.map**
Line #227 change to **output alert_fast**

At the end of the file add this line:

```
output database: log, mysql, user=snort password=<mypassword> dbname=snort host=localhost
```

5. Setup the MySQL server

```
# mysql -u root -p  #You will be prompted to enter the password you created during installation.
mysql> create database snort;
mysql> grant CREATE, INSERT, SELECT, DELETE, UPDATE on snort.* to snort@localhost;
mysql> SET PASSWORD FOR snort@localhost=PASSWORD('mypassword');  # set user password different from "root" password
mysql> use snort;
mysql> source /usr/src/create_mysql
mysql> show tables;  # you should see the list of new tables you just imported.
mysql> exit
```

Now start snort and barnyard with these commands:

```
# /usr/local/bin/snort -q -u snort -g snort -c /usr/local/etc/snort/snort.conf -i eth0 &
# /usr/local/bin/barnyard2 -c /usr/local/etc/snort/barnyard2.conf  -d /var/log/snort -f snort.log -w /usr/local/etc/snort/bylog.waldo -C
/usr/local/etc/snort/classification.config &
```

Again ping the management IP address from another machine

This command shows that barnyard is correctly inserting events into the database:

```
# mysql -uroot -p -D snort -e "select count(*) from event" # you will be prompted to enter root password again
```

6. Configure Apache & PHP

```
# cp /etc/apache2/sites-available/default-ssl /etc/apache2/sites-enabled
# vi /etc/php5/apache2/php.ini
```

Line #452 – change line to read - **error_reporting = E_ALL & ~E_NOTICE**

```
# a2enmod ssl
# pear config-set preferred_state alpha && pear channel-update pear.php.net
# pear install --alldeps Image_Color2 Image_Canvas Image_Graph
# /etc/init.d/apache2 restart
```

7. Install and configure BASE

```
# cd /usr/src && wget http://sourceforge.net/projects/secureideas/files/BASE/base-1.4.5/base-1.4.5.tar.gz
# tar -zxf base-1.4.5.tar.gz && cp -r base-1.4.5 /var/www/base
# chmod 777 /var/www/base
```

(just for now)

Open a browser and go to: https://192.168.1.13/base (or whatever the management IP is) .

Click Continue, choose English
Path to adodb: /usr/share/php/adodb
Click Continue
Database Name: snort
Database Host: localhost
Database Port: leave blank
Database User Name: snort
Database Password: mypass

Put in values for the authentication system and click submit.
Click "create baseag" which extends the DB to support BASE.

Continue to step 5 to login.
You should see a number next to unique alerts – click on that and you should see alerts like this:

Snort Alert [1:10000001:0] – the test rule we created above

If you see alerts in BASE – Congrats – everything is working as it should be.

8. Startup script for snort & barnyard

```
# vi /etc/init.d/snortbarn
```

Paste the following into the file:

```
#! /bin/sh
#
### BEGIN INIT INFO
# Provides:          snortbarn
# Required-Start:    $remote_fs $syslog mysql
# Required-Stop:     $remote_fs $syslog
# Default-Start:     2 3 4 5
# Default-Stop:      0 1 6
# X-Interactive: true
# Short-Description: Start Snort and Barnyard
### END INIT INFO

. /lib/init/vars.sh
. /lib/lsb/init-functions
```

```
mysqld_get_param() {
        /usr/sbin/mysqld --print-defaults | tr " " "\n" | grep -- "--$1" | tail -n 1 | cut -d= -f2
}

do_start()
{
        log_daemon_msg "Starting Snort and Barnyard" ""
        # Make sure mysql has finished starting
        ps_alive=0
        while [  $ps_alive -lt 1 ];
        do
        pidfile=`mysqld_get_param pid-file`
        if [ -f "$pidfile" ] && ps `cat $pidfile` >/dev/null 2>&1; then ps_alive=1; fi
        sleep 1
        done

        /sbin/ifconfig eth1 up
        /usr/local/bin/snort -q -u snort -g snort -c /usr/local/etc/snort/snort.conf -i eth1 &
        /usr/local/bin/barnyard2 -q -c /usr/local/etc/snort/barnyard2.conf -d /var/log/snort -f snort.log -w /usr/local/etc/snort/bylog.waldo -C
/usr/local/etc/snort/classification.config 2> /dev/null &
        log_end_msg 0
        return 0
}

do_stop()
{
        log_daemon_msg "Stopping Snort and Barnyard" ""
        kill $(pidof snort) 2> /dev/null
        kill $(pidof barnyard2) 2> /dev/null
        log_end_msg 0
        return 0
}

case "$1" in
  start)
        do_start
        ;;
  stop)
        do_stop
        ;;
  restart)
        do_stop
        do_start
        ;;
    *)
        echo "Usage: snort-barn {start|stop|restart}" >&2
        exit 3
        ;;
esac
exit 0
------------------------------------------------------------------------------------------
```

Make it executable and create the startup symlinks.

```
# chmod +x /etc/init.d/snortbarn
# insserv -f -v snortbarn
```

Snort & Barnyard will now start automatically at boot.

9. Keep your rules up to date with pulledpork

I encourage you to look at the professional rules available at http://www.emergingthreatspro.com and http://www.snort.org

```
# cd /usr/src && wget https://pulledpork.googlecode.com/files/pulledpork-0.7.0.tar.gz
# tar -zxf  pulledpork-0.7.0.tar.gz && cd pulledpork-0.7.0
# cp pulledpork.pl /usr/local/bin && cp etc/*.conf /usr/local/etc/snort
```

```
# vi /usr/local/etc/snort/pulledpork.conf
```

To use the Sourcefire VRT Certified Rules, go to snort.org, register for an account and get an “oinkcode”, this will allow you to download their Registered User rule set.

Line 19: enter your “oinkcode” where appropriate or comment out the line
Line 26: enter your “oinkcode” where appropriate or comment out the line

Line 27: uncomment to use the Emerging Threats rule set
Line 131: change to: **distro=Debian-6-0**
Line 139: Comment out Blacklist
Lines 194-197: Uncomment

```
# echo pcre:fwsam >> /usr/local/etc/snort/disablesid.conf  # disables all block (fwsam) rules
```

Run pulledpork

```
# chmod +x /usr/local/bin/pulledpork.pl
# /usr/local/bin/pulledpork.pl -c /usr/local/etc/snort/pulledpork.conf -T –l
```

You should now see local.rules and snort.rules in /usr/local/etc/snort/rules.

Clean Up:

```
# rm /var/www/index.html
# chmod 755 /var/www/base
# pkill snort && pkill barnyard2
# rm -rf /var/log/snort/* /var/log/barnyard2/*
# vi /usr/local/etc/snort/rules/local.rules – Comment out the test rule
# vi /usr/local/etc/snort/snort.conf – Line 542: add:  include $RULE_PATH/snort.rules
```

Plug a span port or tap into eth1 and restart snort

```
# /etc/init.d/snortbarn restart
```