SNORT - A BASIC SETUP

Snort is an NIDS system (see IDS for an overview), it relies on signature files to check for bad packets. It is an excellent product, exceptionally useful, runs on most major platforms and is -most importantly- free. Setting up an NIDS on your servers is a good idea as you can see if you are receiving any bad packets, but is is an even better idea to install and NIDS on a machine through which all your traffic must pass (for example a firewall), this way you can see what attacks are attempted on your network, even if your firewall does stop them. In this article I will be going through setting up snort on a linux server.

What do I need?
Firstly, you will go course need to download snort, you can get it at *www.snort.org*, and you will very probably need the *libnet* package which you can find in a package at a place like *www.rpmfind.net* or you can get the source from *www.packetfactory.net*. Once you have this software you will be able to begin installing.

Starting with Libnet
Chances are that you will need to install libnet, if you downloaded a package -such as a .rpm file- you can use your package management system to install it. Something like..

rpm -U -v libnet<version>.rpm

If you downloaded the source, then you will need to extract it to a folder. Once you go to that folder you will have to..

- *cp ./makfiles/linux.mak ./port mak* (I am assuming a linux server is our base for the snort system)
- *make*
- *make install*

Installing Snort
Compiling for a basic snort system is a easy process. Extract the source to a directory, change to that directory..

- *./configure*
- *make*
- *make install*

Now we need to set it up so that we can run it. So when need to create a home directory where the rules and the configuration files will live, something like */admin/snort*. Once you have this you need to copy everything from the *<snort_source>/etc* folder to this home folder. You will have copied a sample *conf* file, *.map* files and some other files. Now we need our rules. I would suggest getting three different sets..

- The official rules from *www.snort.org* (to download any rules from *www.snort.org* you will need to register, but its free)
- The community rules from *www.snort.org*
- The bleeding snort ruleset from *www.bleedingsnort.com*

Once you have these files extract them into the home folder (in the example it was */admin/snort*) this will give you a *<snort_home>/rules* directory. You will now be able to use the rules listed there. And that is what we will now do, open up the *snort.conf file* in your favorite editor..

- set the *RULE_PATH* variable to where your rules are (ie, */admin/snort/rules*)
- at the end of the file, add-in or uncomment all the rules you want to use (ie, *include $RULE_PATH/bleeding-malware.rules*)
- you may need to create a */var/log/snort* directory

Lastly, I generally create a startup script that I can put into the *rc.loca*l file or any other startup file I want. A simple startup script could look like this..

```
LNIC=eth1
HOME=/admin/snort
SNORT=/usr/local/bin/snort
CONF=/admin/snort/snort.conf
cd $HOME
# the -D is to make to run in the background
# the -c secifies the conf file
# the -C dumps character data
# the -i tells it what interface to watch
# the -I adds the interface name to the alerts
# the -p does not put the interface into promiscous mode
# the -q is for quiet running
# the -z is for connection checking
# the -U is for a timestamp
$SNORT -D -c $CONF -C -i $LNIC -I -p -q -z -U
```

Checking your Logs
Well, if you now have snort running you are going to want to see what packets it does not like. A handy program for that is the *SnortSnarf* perl script which outputs nice html files which can viewed via a web server. If you use this script you will need to download it and install the *Time::JulianDay* perl module if you do not have it. You can use *perl -MCPAN -e shell* to do this if you have internet access from your server. Once the perl script can run it is as simple as..

snortsnarf.pl /var/log/alerts

or if you want a full listing of the runtime options you can use..

snortsnarf.pl -usage

Or if you are feeling up to it, you can write your own script to go through the log files. For example here is a script I use on a box that does not have perl or a web server installed - for various reasons. Be warned, this script is not pretty or going to win any scripting prizes, it was a quick hack for the report I wanted. Feel free to use it or modify it at your own risk.

```
HOME=/admin/snort/custom
LOG=output.alert
NIC=eth1

##get a baseline
cat /var/log/snort/alert | grep -A 2 eth1 | grep -v Classi > $HOME/$LOG

##give log start date
START=`cat $HOME/$LOG | head -2 | grep -e "->" | cut -f 1 -d "."`
echo "log starts at $START"

##total number of alerts
TOTAL=`cat $HOME/$LOG | grep $NIC | wc -l`
echo "total alerts = $TOTAL"
echo ""

##all unique packet matches
echo "all unique matches are.."
cat $HOME/$LOG | grep $NIC |   sort | uniq
echo ""

##a count of all unique packet matches with source and destination
echo "match count.."
for x in `cat $HOME/$LOG | grep $NIC |   sort | uniq | cut -f 3 -d "[" | cut -f 1 -d "]"`
 do
   CNT=`cat $HOME/$LOG | grep -e "$x" | wc -l`
   NAME=`cat $HOME/$LOG | grep -e "$x" | head -1`
   echo "$NAME had $CNT total matches"
   echo "..match sources.."
   cat $HOME/$LOG | grep -A 2 -e "$x" | grep -e "->" | gawk '{ print $2 }' | cut -f 1 -d ":" | sort | uniq
   echo "..match targets.."
   cat $HOME/$LOG | grep -A 2 -e "$x" | grep -e "->" | gawk '{ print $4 }' | cut -f 1 -d ":" | sort | uniq
   echo ""
 done

##top 10 source of packet matches with a count
echo ""
echo "top sources of matches.."
for y in `cat $HOME/$LOG | grep -e "->" | gawk '{ print $2 }' | cut -f 1 -d ":" | sort | uniq | head -10`
 do
  SCNT=`cat $HOME/$LOG | grep -e "->" | gawk '{ print $2 }' | grep $y | wc -l`
  echo "$y had $SCNT matches"
 done

##top 10 destinations of packet matches with a count
echo ""
echo "top destinations of matches.."
for k in `cat $HOME/$LOG | grep -e "->" | gawk '{ print $4 }' | cut -f 1 -d ":" | sort | uniq | head -10`
 do
  DCNT=`cat $HOME/$LOG | grep -e "->" | gawk '{ print $4 }' | grep $k | wc -l`
  echo "$k had $DCNT matches"
 done

##end date of the log
echo ""
END=`cat $HOME/$LOG | tail -2 | grep -e "->" | cut -f 1 -d "."`
echo "log ends at $END"
```

Final Words
This was just a basic snort setup, but even just this will be very informative and useful. Please play around and experiment with snort as it is capable of a lot more. Do not be scared of the alerts you get, go through them, investigate, and if need be chase them up. As always, have fun and learn lots.