

Snort Use Notes

V2 – Fixed a couple typos noted in class

For full snort information, see <http://snort.org>.

Important Snort Files

- `/var/log/snort` – Snort log and alert files. You can use `ethereal` or `tcpdump` to look at the captured packets in the log files. The alerts are in `ascii`.
- `/etc/snort/snort.conf` – Default configuration files. Controls the built in controls and loads the rule files. Use this to tune the signatures you are scanning for.
- `/etc/snort/rules` – Signature rule files.

Running Snort

You can run snort in several ways: as a sniffer, as an out-of-band NIDS, or inline.

Options to run as a sniffer:

- `snort -v` or `snort -dv` or `snort -dve`: Vary the amount of detail captures.
- `Snort -dve -l ./log -h 192.168.100.0/24`: Specify a log file and the home network. Default home network can also be specified in `snort.conf` file.

Options to run as NIDS

- `snort -dve c /etc/snort/snort.conf`: Use default log, Run from the snort config file. Still capture packets.

Run in inline mode. First you need to set the device to run in transparent bridge mode

- `brctl addbr br0`
- `brctl addif br0 eth0`
- `brctl addif br0 eth1`
- `ifconfig eth0 up`
- `ifconfig eth1 up`
- `ifconfig br0 up`

At this point the device should be bridging. Now we need to set the iptables to pass all packets to the queue to be processed by snort

- `modprobe ip_queue` – Load the kernel support for QUEUES. The QUEUE is a means to pass packets from kernel to user space.
- `iptables -I FORWARD -o br0 -j QUEUE`

No traffic will pass until we turn on snort

- `snort -Qc /etc/snort/snort.conf`

The bridging commands were gleaned from the more general `rc.firewall` script distributed with the honey net project (<http://www.honeynet.org/tools/dcontrol/rc.firewall>). This script also supports a L3 routed mode which they call NAT mode. The benefit of operating in a bridge or L2 mode is that you can insert the snort box without adjusting your address distribution or routing logic. From the honeynet perspective, it makes it harder for the attacker to notice that you might have an interception point.

Things to Try

Replay some of the CCDC logs in the root home directory: `snort -pcap-single=<logfile name> -c /etc/snort/snort.conf`. Look at the alerts. Can you find the matching rules. What do the alerts mean?

Try running `nmap` through or past the snort box. Do you get alerts? Try using `ftp` to get some interesting system files.

Try to write a signature that differentiates good and bad magic8 packets.

If you are working on an inline device try changing some of the signatures to drop packets rather than just alerting.