# Suricata

For this project you will be required to install suricata. I started with my ubuntu1404 image and cloned it. Suricata is an IDS/IPS. First, I installed the following packages:

```
apt-get install libpcre3 libpcre3-dbg libpcre3-dev build-essential libpcap-dev   \
        libyaml-0-2 libyaml-dev pkg-config zlib1g zlib1g-dev \
        make libmagic-dev libjansson-dev liblz4-dev
```

Then, download the suricata tar file.

```
wget https://www.openinfosecfoundation.org/download/suricata-5.0.2.tar.gz
```

Install rust

```
curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

Set default python version

```
sudo update-alternatives --install /usr/bin/python python /usr/bin/python2.7 1
sudo update-alternatives --config python
```

Might have to adjust some path stuff. Logout and log back in and make sure that cargo is in your path somewhere. (I.e. `/home/<user>/.cargo/bin`)

Extract suricata and run:

```
./configure --prefix=/usr --sysconfdir=/etc
sudo make
sudo make install-full
```

We can now run suricata by something like:

```
/usr/bin/suricata -c /etc/suricata/suricata.yaml -i eth0
```

## EDITS

Now for some detailed edits.

Edit `/etc/suricata/rules/test.rules` to have the following:

```
alert icmp any any -> $HOME_NET any (msg:"ICMP connection attempt"; sid:1000002; rev:1;)
alert tcp any any -> $HOME_NET 23 (msg:"TELNET connection attempt"; sid:1000003; rev:1;)
```

This gives us some sample rules that will cause an alert to trigger.

Now edit `/etc/suricata/suricata.yaml`. Find the `rule-files` section and add:

```
- test.rules
```

Note that in the previous file the values for HOME_NET. Leave them be.

## NAT

In our setup, we are going to have traffic flow through our suricata box. It will perform NAT functionality for us. In order to do this, we will make it listen on 2 `interfaces`. (These instructions are for pre-Ubuntu 18.04). Ubuntu 18.04 is a little different.

Begin my making sure your normal networking works. Now, edit your `/etc/network/interfaces` to include something like following, but DON'T DELETE or CHANGE your current iface.

```
iface ens18:0 inet static
  address 10.100.1.1
  netmask 255.255.255.0
```

```
    #the gateway address should be the public ip address of this NAT machine
    gateway 144.38.193.207
```

My primary interface was ens18. Your is PROBABLY something different. Change to reflect what your is. You can use the same address and netmask since we are on separate networks. See if you can bring the interface up `ifup ens18:0`. If you get errors, first check to see what the output of `ip a | grep 10.100.1.1` is. If the ip address is listed, then everything worked ok.

Use the following commands to enable NAT. Note that my primary interface was ens18 (public), my private interface (as shown above) was ens18:0. YOURS WILL BE DIFFERENT. Make changes to reflect that.

```
    iptables -t nat -A POSTROUTING -o ens18 -j MASQUERADE
    iptables -A FORWARD -i ens18:0 -o ens18 -m state --state RELATED,ESTABLISHED -j ACCEPT
    iptables -A FORWARD -i ens18:0 -o ens18 -j ACCEPT
```

We also need to tell the suricata system that it should forward packets. You need to make sure the contents of `/proc/sys/net/ipv4/ip_forward` is a 1. If it isn't, change it by doing something like `echo 1 > /proc/sys/net/ipv4/ip_forward`. To make this change persist you should edit `/etc/sysctl.conf` and add `net.ipv4.ip_forward = 1`.

Start suricata by doing something like this:

```
    /usr/bin/suricata -D -c /etc/suricata/suricata.yaml -i ens18:0
```

Create another linux machine that is on the same PRIVATE network as above. It need not be on the public network. It will not do NAT. The gateway for this machine will be `10.100.1.1`. Make sure this machine can ping out.

After successfully doing a ping, you should be able to see your alert fire, by looking on suricata machine:

```
    tail -f /var/log/suricata/fast.log
```

Once it is working right, you can kill suricata and change every reference of `eeth0` to `ens18:0` (or whatever your nic is) in `/etc/suricata/suricata.yaml`. And then `service suricata start`. Double check it still can log.

## Some more tests

From client machine:

- use `apt` to install something (or upgrade)
- use curl to send an emerging threats user agent string through it (i.e. something like `curl -A REKOM www.google.com`.
- use wget to create a tcp connection to an ip listed in `tor.rules`
- Look at rules and see if you can get others to fire (without actually generating malicious traffic)

It would be interesting to see if it can detect the rootkit traffic/RAT tool we installed a few weeks ago.

## TO pass off

Take a screenshot of your `fast.log`.