Basic Setup

When using Debian or FreeBSD, make sure you enter all commands as root/super-user because for these operating systems it is not possible to use 'sudo' without installing and configuring it first.

Start with creating a directory for Suricata's log information.

```
sudo mkdir /var/log/suricata
```

To prepare the system for using it, enter:

```
sudo mkdir /etc/suricata
```

The next step is to copy classification.config, reference.config and suricata.yaml from the base build/installation directory (ex. from git it will be the oisf directory) to the /etc/suricata directory. Do so by entering the following:

```
sudo cp classification.config /etc/suricata
sudo cp reference.config /etc/suricata
sudo cp suricata.yaml /etc/suricata
```

Note: if you have experience with Snort or have an existing Snort setup, check out the Snort.conf to Suricata.yaml guide.

Auto setup

You can also use the available auto setup features of Suricata:

ex:

```
 ./configure && make && make install-conf
```

The *make install-conf* option will do the regular "make install" and then automatically create/setup all the necessary directories and suricata.yaml.

```
./configure && make && make install-rules
```

The *make install-rules* option will do the regular "make install" and it automatically downloads and sets up the latest ruleset from Emerging Threats available for Suricata.

```
./configure && make && make install-full
```

The *make install-full* option combines everything mentioned above (install-conf and install-rules) - and will present you with a ready to run (configured and set up) Suricata

Setting variables

Make sure every variable of the vars, address-groups and port-groups in the yaml file is set correctly for your needs. A full explanation is available in the Rule vars section of the yaml. You need to set the ip-address(es) of your local network at HOME_NET. It is recommended to set EXTERNAL_NET to !$HOME_NET. This way, every ip-address but the one set at HOME_NET will be treated as external. It is also possible to set EXTERNAL_NET to 'any', only the recommended setting is more precise and lowers the chance that false positives will be generated. HTTP_SERVERS, SMTP_SERVERS, SQL_SERVERS, DNS_SERVERS and TELNET_SERVERS are by default set to HOME_NET. AIM_SERVERS is by default set at 'any'. These variables have to be set for servers on your network. All settings have to be set to let it have a more accurate effect.

Next, make sure the following ports are set to your needs: HTTP_PORTS, SHELLCODE_PORTS, ORACLE_PORTS and SSH_PORTS.

Finally, set the host-os-policy to your needs. See Host OS Policy in the yaml for a full explanation.

```
  windows:[]
  bsd: []
  bsd-right: []
  old-linux: []
  linux: [10.0.0.0/8, 192.168.1.100, "8762:2352:6241:7245:E000:0000:0000:0000"]
  old-solaris: []
  solaris: ["::1"]
  hpux10: []
  hpux11: []
  irix: []
  macos: []
  vista: []
  windows2k3: []
```

Note that bug ~~#499~~ may prevent you from setting old-linux, bsd-right and old-solaris right now.

Rule set management and download.

Rule Management with Oinkmaster

or just download and untar the ruleset in a directory of your choosing (or yaml config setting) from here:
http://rules.emergingthreats.net/open/suricata/

or if you prefer you can download and use a VRT ruleset.

It is recommended to update your rules frequently. Emerging Threats is modified daily, VRT is updated weekly or multiple times a week.

Interface cards

To check the available interface cards, enter:

```
ifconfig
```

Now you can see which one you would like Suricata to use.

To start the engine and include the interface card of your preference, enter:

```
sudo suricata -c /etc/suricata/suricata.yaml -i wlan0
```

Instead of wlan0, you can enter the interface card of your preference.

To see if the engine is working correctly and receives and inspects traffic, enter:

```
cd /var/log/suricata
```

Followed by:

```
tail http.log
```

And:

```
tail -n 50 stats.log
```

To make sure the information displayed is up-dated in real time, use the -f option before http.log and stats.log:

```
tail -f http.log stats.log
```