# Attack Identification Using Snort

Practical Assignment—Lab Report

. . .

1. **What is snort?**

Snort is an open-source network intrusion detection and prevention system (IDS/IPS). It monitors network traffic for malicious activity and can block potential attacks.

**What are snort 'custom rules'?**

A "Snort custom rule" refers to a user-defined rule created within the Snort intrusion detection/prevention system that allows network administrators to monitor for specific, customized patterns of network traffic that might indicate a potential security threat, not covered by the default Snort rule sets, enabling them to tailor their network protection to their unique environment and needs by defining specific actions to take when a match is found, like generating an alert or blocking the traffic.

The aim of this practical assignment was to manually configure snort and create a custom rule-set to identify and generate alerts for 4 possible attacks namely :

1. *ICMP Ping of Death Attack*
2. *Suspicious FTP Connection Request Attack (SYN Request)*
3. *Suspicious IP Address Connection Attack*
4. *Blacklisted IP Address Connection Attack*

**Task 1 : Install & Configure SNORT along with all the required libraries**

On using the following command we were able to ensure that our snort.conf (configuration file) had been successfully installed.

```
sudo snort -T -c /etc/snort/snort.conf
```

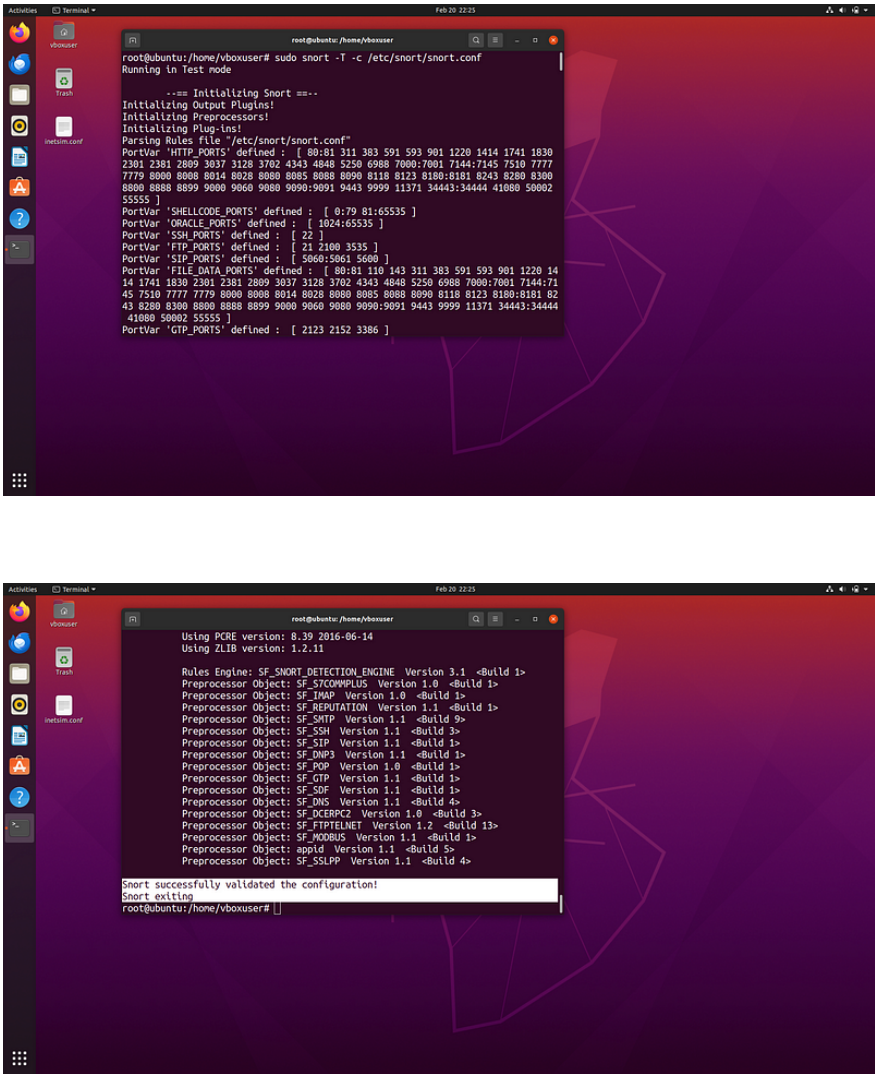


Fig 1.1 — Snort configured successfully

Post this, we made changes within the configuration file to include paths to various custom created directories so that snort can access our rules from there.

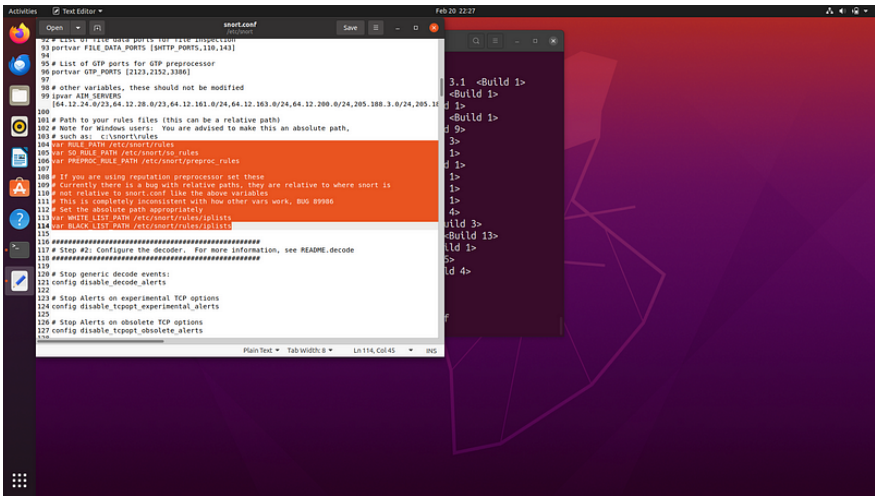

Fig 1.2 — Modified snort.conf file

**Task 2 : Create a Custom Rule Set in SNORT to identify (generate Alert) the Ping of Death Attack.**

This entails that when another machine (potentially the attacker) generates a ping request to this (host) machine's IP address as follows

```
ping <IP Address of host>
```

Then an alert must be generated on the host machine saying "ICMP Ping Detected." We created a custom rule for the same in the following file /etc/snort/rules/local.rules

```
sudo gedit /etc/snort/rules/local.rules
```

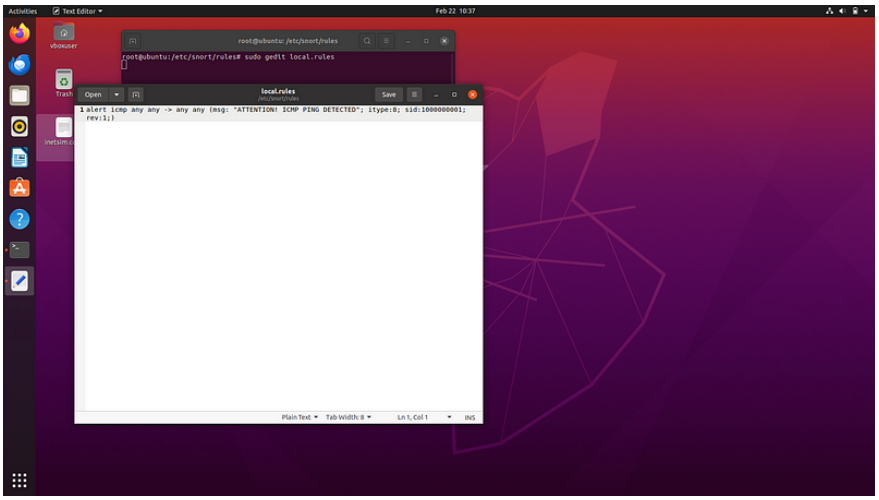


Fig 2.1 — Creating the rule

Explanation of rule keywords :

"alert" — the action that needs to be performed by snort

"icmp" — the targeted protocol

"any any" — any Source IP and port and any Destination IP and port

"->" — unidirectional flow

"msg" — the alert message to be generated

"itype" — the itype:8 rule will always fire regardless of it's position in the rule file because it is the more unique rule

"sid" — unique identifier for the rule

"rev" — indicates that a rule is in its first revision

In order to execute the rule we shall write the following command on the host machine

```
sudo snort -A console -q -c /etc/snort/rules/loc
```

And the following command on the attacker machine

```
ping 192.168.57.5
```

The following images display the execution of the attack.

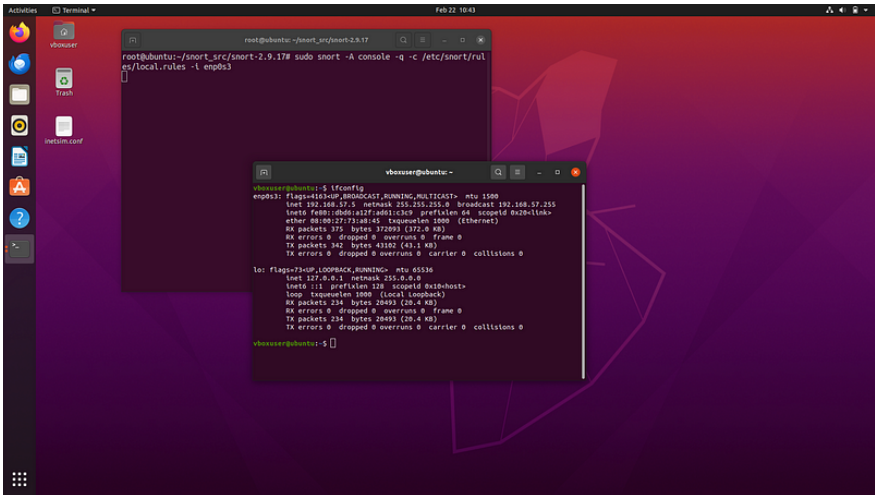


Fig 2.2 — Finding the IP of host machine

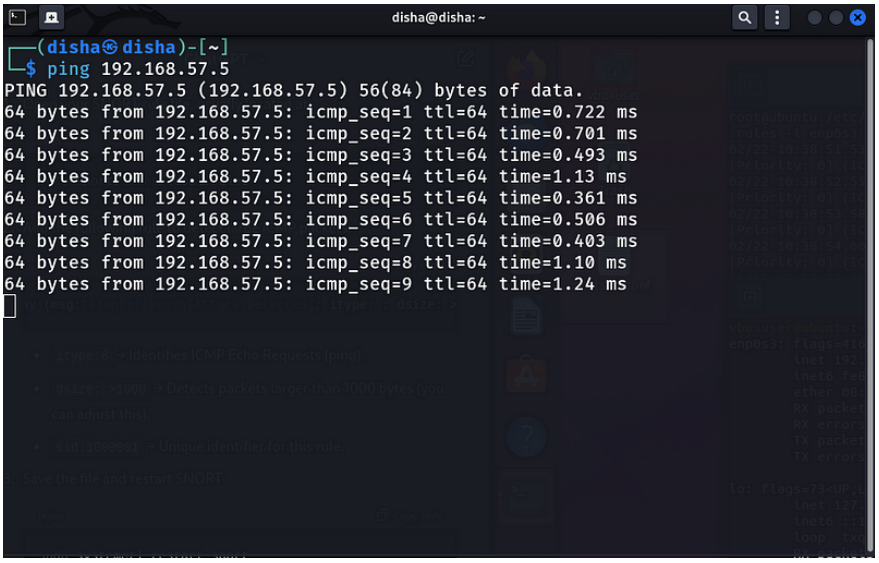


Fig 2.3 — Ping alert generated from attacker machine

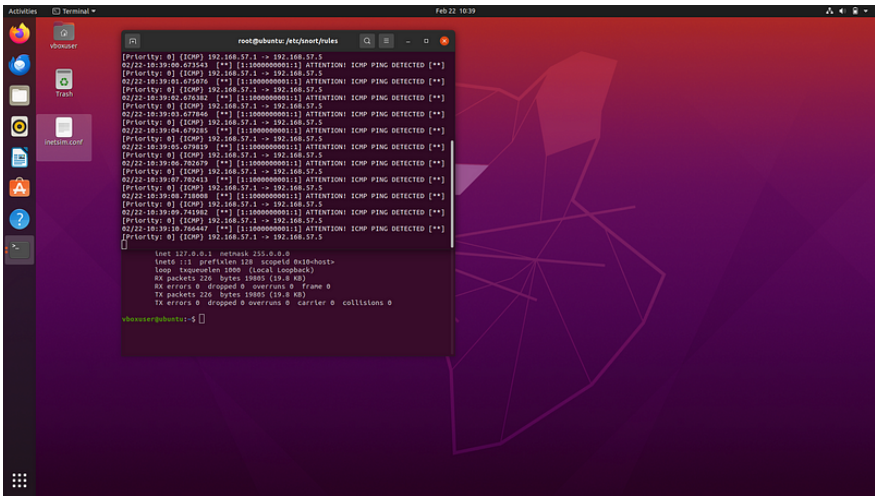

Fig 2.4 — Alert message generated

**Task 3 : Create a Custom Rule Set in SNORT to identify (generate Alert) the FTP Connection Request.**

The aim of this task is to generate an alert on the host machine if a FTP connection request (usually the presence of a SYN flag) is detected from the attacker machine.

This is the custom rule that I created for the same:

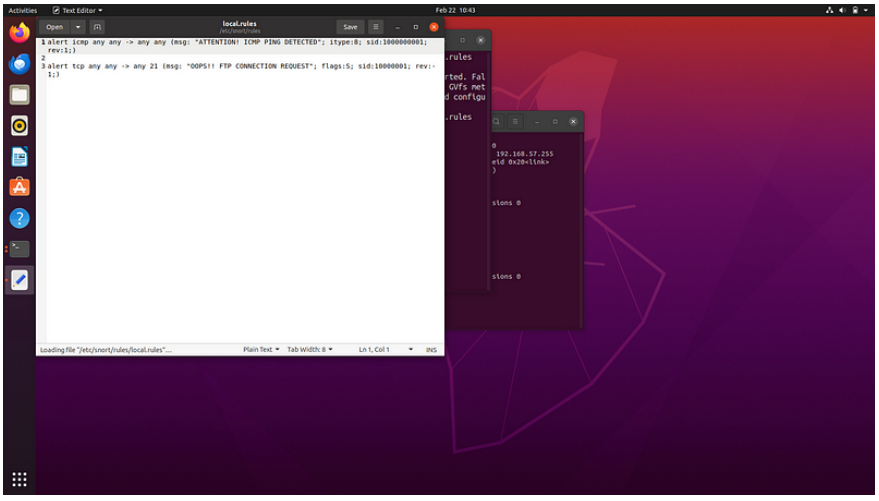

Fig 3.1 — FTP Connection rule

Explanation of rule keywords :

"alert" — the action that needs to be performed by snort

"tcp" — the targeted protocol

"any any" — any Source IP and port and any Destination IP and port

“21”—port for FTP requests

“flag”—S stands for SYN request

“->”—unidirectional flow

“msg”—the alert message to be generated

“sid”—unique identifier for the rule

“rev”—indicates that a rule is in its first revision

On running the following command on the host machine

```
sudo snort -A console -q -c /etc/snort/rules/sno
```

And the following command on the attacker machine

```
ftp 192.168.57.5
```

We get the following output

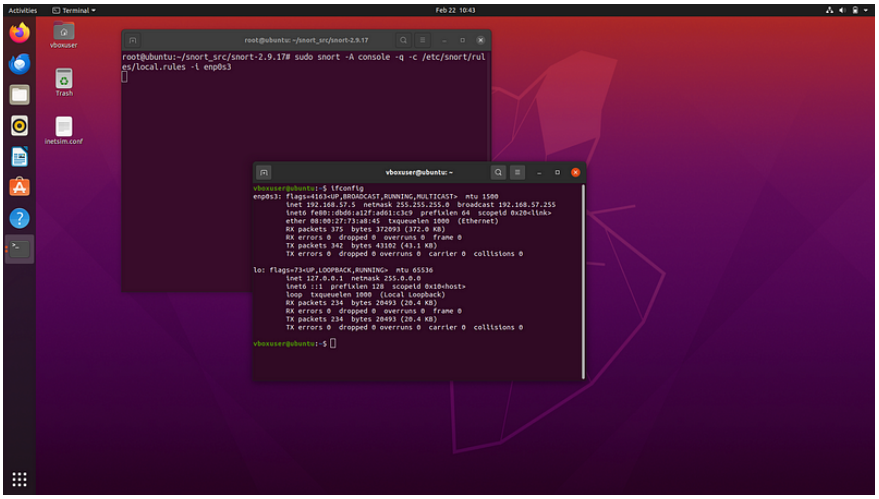


Fig 3.2—Finding the IP of the host machine


Fig 3.3 — Generating ftp connection request from attacker machine

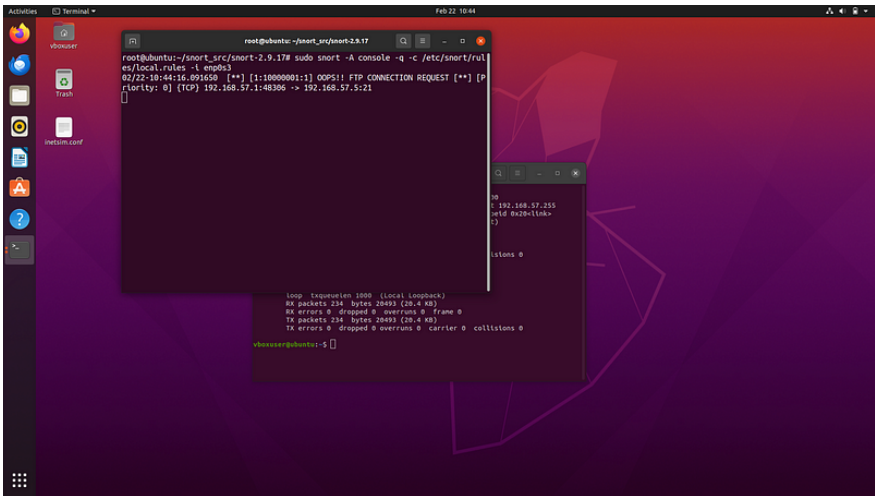


Fig 3.4 — Alert generated on host machine

**Task 4 : Create a Custom Rule Set in SNORT to identify (generate Alert) for a request from one specific IP to your Home Network.**

The aim of this task is to generate an alert if a specified IP Address (not blacklisted) makes a connection request to the host's IP Address. For this I have written the following custom rule.

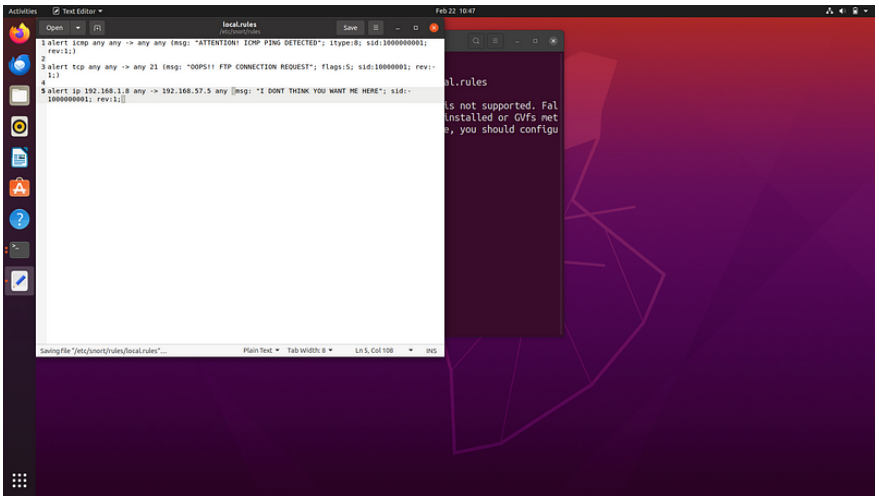


Fig 4.1 — Custom rule for IP request alert

Explanation of rule keywords :

"alert" — the action that needs to be performed by snort

"ip" — referring to IP address rule

"any any" — any Source IP and port and any Destination IP and port

"192.168.1.8" — attacker machine's IP address

"192.168.57.5" — host machine's IP address

"->" — unidirectional flow

"msg" — the alert message to be generated

"sid" — unique identifier for the rule

"rev" — indicates that a rule is in its first revision

Then we run the following command to start snort console and simulate the attack.

```
sudo snort -A console -q -c /etc/snort/rules/loc
```

On the attacker machine we can run any of the following commands

```
nc -zv 192.168.57.5 80
telnet 192.168.57.5 80
curl http://192.168.57.5
```

The following images show the simulation of the attack :

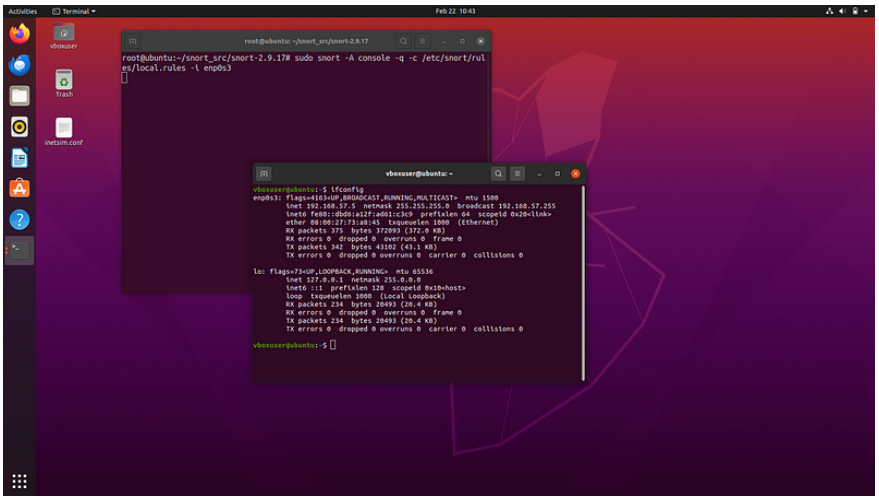

Fig 4.2 — Finding the IP of the host machine

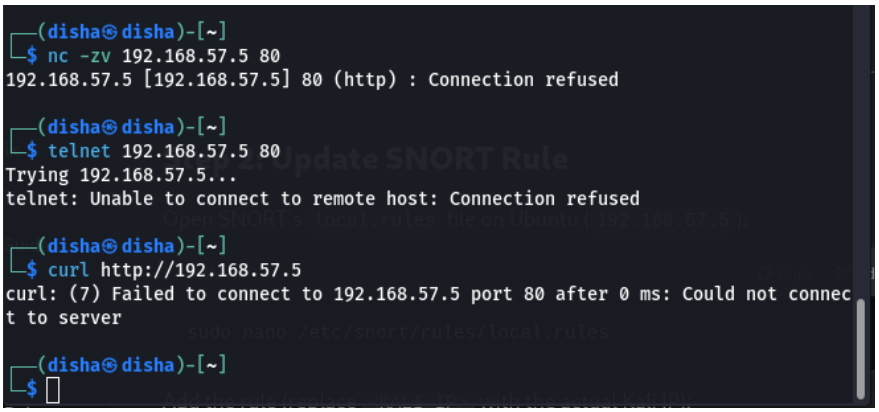


Fig 4.3 — Commands run on attacker machine

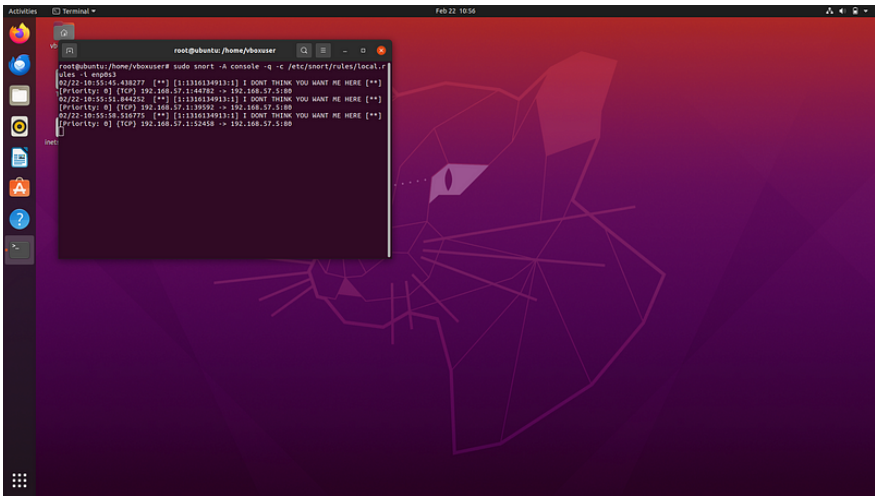


Fig 4.4 — Alert generated on host machine

**Task 5 : Create a Custom Rule Set in Snort to identify (generate Alert) for a request from a Blacklisted IP.**

The aim of this task is to generate an alert if a specified IP Address (blacklisted) makes a connection request to the host's IP Address. For this I have written the following custom rule.

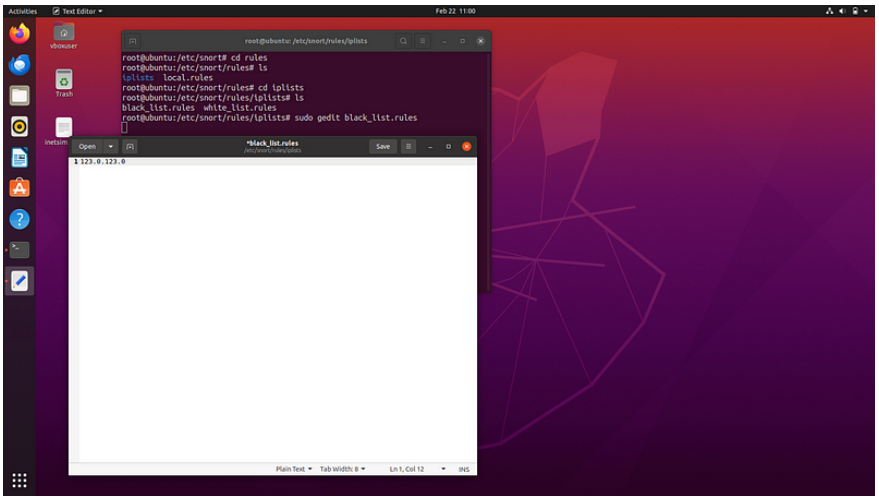


Fig 5.1 — Adding random IP to black_list.rules

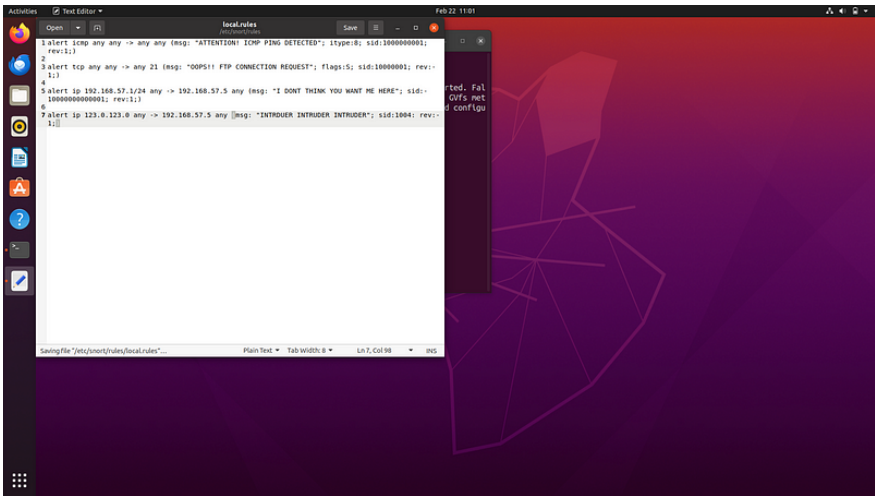


Fig 5.2 — Creating custom rule

Explanation of rule keywords :

"alert" — the action that needs to be performed by snort

"ip" — referring to IP address rule

"any any" — any Source IP and port and any Destination IP and port

"123.0.123.0" — random blacklisted IP address

"192.168.57.5" — host machine's IP address

"->" — unidirectional flow

"msg" — the alert message to be generated

"sid" — unique identifier for the rule

"rev" — indicates that a rule is in its first revision

We run the following command on the host machine

```
sudo snort -A console -q -c /etc/snort/rules/loc
```

And the following command on the attacker machine

```
sudo hping3 -S -p 80 -a 123.0.123.0 192.168.57.5
```

The above command pings the host machine by forging its identity and acting like “123.0.123.0”

The following images show the simulation of the attack

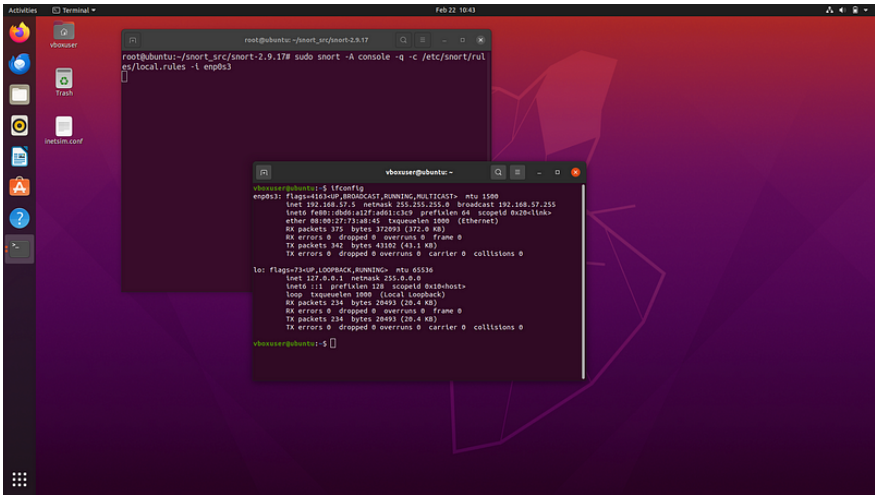


Fig 5.3 — Finding the IP of the host machine

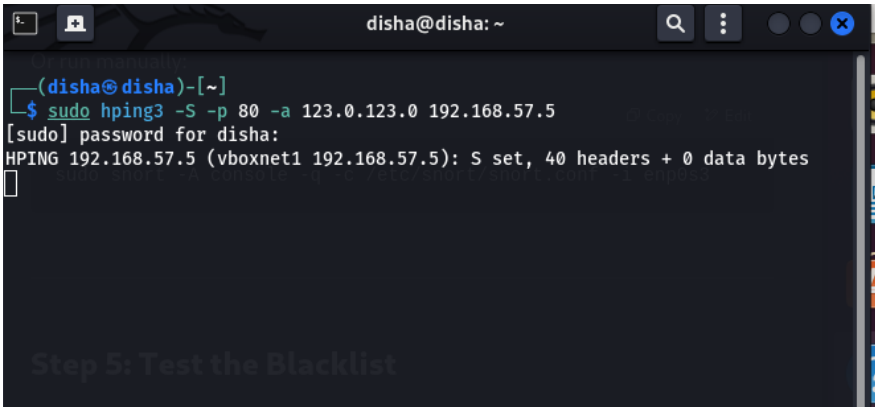


Fig 5.4 — Pinging the host machine from the attacker’s end

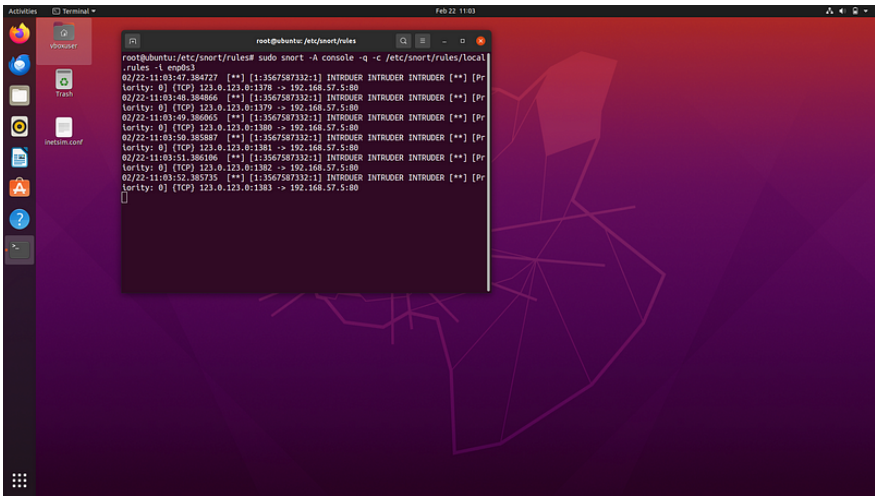


Fig 5.5 — Alert generated on host machine

**Conclusion:**

This practical assignment gave me an understanding of how snort rules function and intricate details of the configuration. During the installation procedure for snort, I encountered many errors such as “compatible with libpcap > 1.0 version” and “missing rpc headers”. Solving these errors made me interact with snort on a much more functional level. Add to this, the process of creating and executing custom rule-sets!