Name: Anmol

Tripathi

**Roll No.: 138** 

Batch:T23

## LAB ASSIGNMENT No. 11

**Aim:** Installing snort, configuring it in Intrusion Detection mode and writing rules for detecting pinging activity.

**Lab Outcome Attained:** LO6

## **Theory:**

Steps to Install snort and configure it in Intrusion Detection Mode.

- 1. Check the name of the interface using command if config.
- 2. Install snort in ubuntu machine using command sudo apt-get install snort
- 3. While installing the snort, name of the interface will be asked on which snort is supposed to listen. Enter the interface name observed in step 1.
- 4. Run the command *sudo gedit /etc/snort/snort.conf* . This opens snort configuration file.
- 5. Make following changes to configuration file.
  - **a.** ipvar HOME\_NET **192.168.0.0/24** (in section 1)
- 6. Open new terminal. Open <a href="ftp.rule">ftp.rule</a> file in it by typing the command <a href="sudo">sudo</a> gedit /etc/snort/rules/ftp.rules (optional)
- 7. Open new terminal and type the command *sudo snort -T -c* /*etc/snort/snort.conf -i enp3s0* to validate that all rules are there.

We use the

- -T flag to test the configuration file,
- -c flag to tell Snort which configuration file to use, and -i to specify the interface that Snort will listen on.
- 8. Type the command *sudo snort -A console -q -u snort -g snort -c*

/etc/snort/snort.conf -i enp3s0 (to start snort in NIDS mode)

We use the

-A console The 'console' option prints fast mode alerts to stdout

-q Quiet mode. Don't show banner and status report.

-u snort Run Snort as the following user after startup

-g snort Run Snort as the following group after startup

-c /etc/snort/snort.conf The path to our snort.conf file

- -i enp3s0 The interface to listen on (change to your interface if different)
- 9. Now go to kali linux machine.
- 10. Type command *nmap* 192.168.0.107 on it to start port scanning of ubuntu machine and observe the output in terminal where snort is started in detection environment.

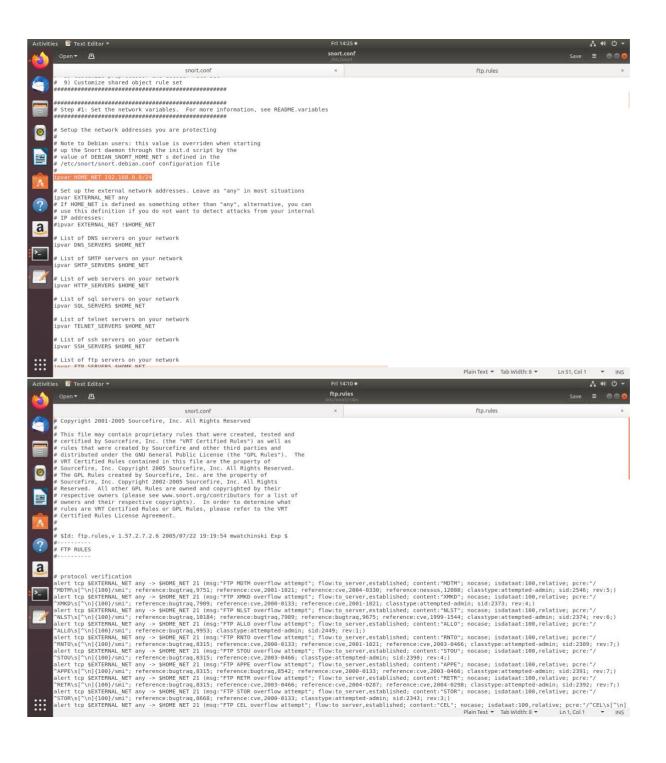
When you execute this command, you will not initially see any output. Snort is running, and is processing all packets that arrive on eth0 (or whichever interface you specified with the -i flag). Snort compares each packet to the rules it has loaded (in this case our single ICMP Ping rule), and will then print an alert to the console when a packet matches our rule.

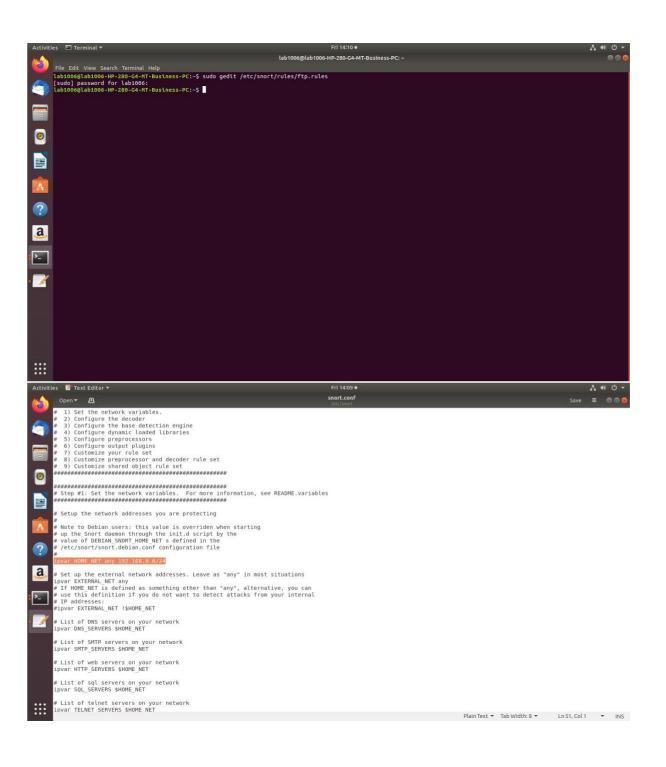
- 11. Then try pinging ubuntu machine by typing the command *ping*192.168.0.107 and observe the output in terminal where snort is started in detection mode.
- 12. Adding rule for detecting ping activity performed by another machine:
  - a. In ubuntu machine, type the following command to create a file called local.rules : sudo gedit /etc/snort/rules/local.rules

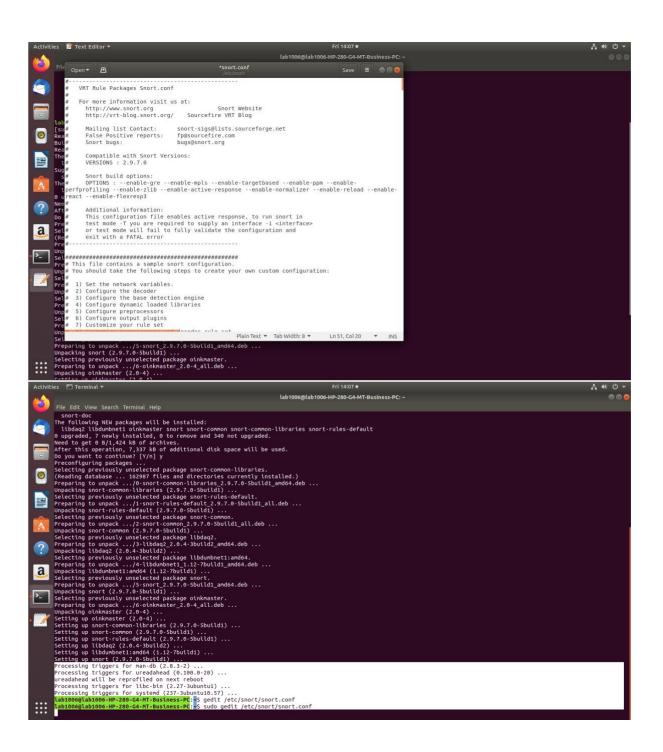
- b. Write the following rule in it: alert icmp any any -> \$HOME\_NET any (msg:"ICMP test detected"; GID:1; sid:10000001; rev:001; classtype:icmp-event;)
- c. Save the local rules file.
- d. Comment the following lines in configuration file (snort.conf) of snort: icmp.rules and icmp-info.rules
- e. Add the local.rules file in section 7 of configuration file of snort by writing: include \$RULE\_PATH local.rules
- f. Validate the changes made in snort.conf file by writing the command in terminal: sudo snort -T -c /etc/snort/snort.conf -i enp3s0
- g. Set the snort in Intrusion Detection Mode by typing the command:
  sudo snort -A console -q -u snort -g snort -c /etc/snort/snort.conf i enp3s0
- h. Now from kali machine ping the ubuntu machine and see the alert generated.
- i. Observe the difference between the alerts generated when icmp.rules and icmp-info.rules are used and when local.rules is used to detect the ping activity.

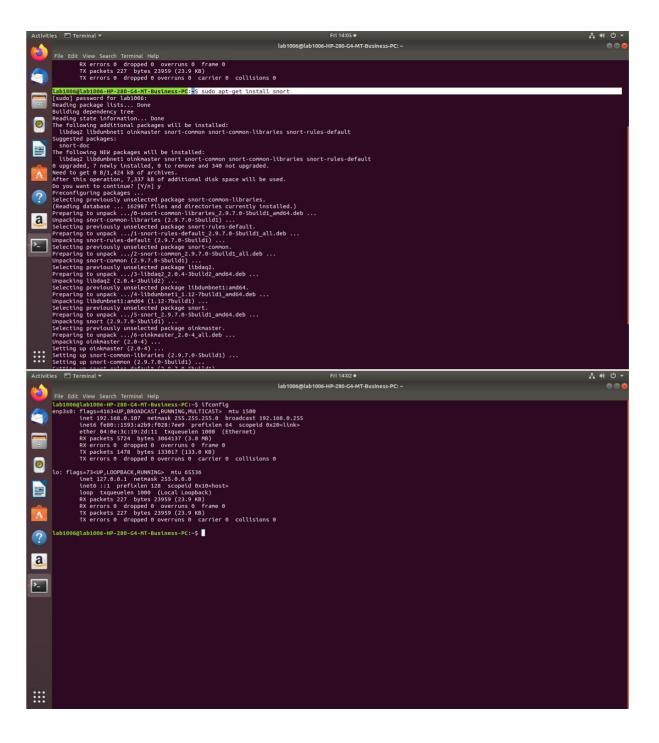
Reference Link for Demo: <a href="https://www.youtube.com/watch?v=iBsGSsbDMyw">https://www.youtube.com/watch?v=iBsGSsbDMyw</a>

## **Output:**









```
lab1006@lab1006-HP-280-G4-MT-Business-PC:
    File Edit View Search Terminal Help
Lab10068[lab1006-IP-280-G4-MT-Business-PC:-$ sudo snort -T -c/etc/snort/snort.conf -i enp3s0
[sudo] password for lab1006:
lunning in Test mode
Running in Test mode

--== Initializing Short ==--

Initializing Output Plugins!

Initializing Preprocessors!

Initializing Plugins!

Parsing Rules file "/etc/short/short.conf"

EBROR: /etc/short/short.conf(51) Missing argument to HOME_NET

Fatal Error, Quitting.

Labitoge(abitoo-HP-280-Gd-MT-Business-PC:-S

Labitoge(abitoo-HP-280-Gd-
  Partializing Snort ==--
Initializing Snort ==--
Initializing Plugins!
Initializing Preprocessors!
Initializing Preprocessors!
Initializing Plugins!
Initializing Plugins!
Initializing Plugins!
Initializing Plugins!
Initializing Plugins!
Pure Initializing Pure Initi
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            lab1006@lab1006-HP-280-G4-MT-Business-PC: ~
                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   000
         File Edit View Search Terminal
State Denstty : 10.6%
Patterns : 5055
Match States : 3855
Match States : 17.00
Potterns : 0.51
Match Lists : 1.02
DFA : 1 byte states : 14.05
4 byte states : 0.00
              Number of patterns truncated to 20 bytes: 1039 ]
cap DAQ configured to passive.
cquiring network traffic from "enp3s0".
                                                               --== Initialization Complete ==--
           ... -*> Snortl <*.

o" )- Version 2.9.7.6 GRE (Build 149)

"" By Martin Reesch & The Snort Team: http://www.snort.org/contact#team
Copyright (C) 2014 (1sco and/or its affiliates. All rights reserved.
Copyright (C) 1998-2013 Sourcefire, Inc., et al.
Using libpcap version 1.8.1

Using PCRE version: 8.39 2016-06-14

Using ZLIB version: 1.2.11
                                                                                  Using ZLIB version: 1.2.11

Rules Engine: SF_SNORT_DETECTION_ENGINE Version 2.4 <Build 1>
Preprocessor Object: SF_SMTP Version 1.1 <Build 9>
Preprocessor Object: SF_GMTP Version 1.1 <Build 9>
Preprocessor Object: SF_DMP3 Version 1.1 <Build 1>
Preprocessor Object: SF_DMP3 Version 1.1 <Build 1>
Preprocessor Object: SF_DERPC2 Version 1.0 <Build 3>
Preprocessor Object: SF_STP Version 1.1 <Build 4>
Preprocessor Object: SF_STP Version 1.1 <Build 1>
Preprocessor Object: SF_FMTPLINET Version 1.2 <Build 1>
Preprocessor Object: SF_FMTPLINET Version 1.2 <Build 13>
Preprocessor Object: SF_FMOPBUS Version 1.1 <Build 3>
Preprocessor Object: SF_SMOPBUS Version 1.1 <Build 1>
Preprocessor Object: SF_SCEP Version 1.1 <Build 4>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 1>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 4>
Preprocessor Object: SF_SSLPP Version 1.1 <Build 1>
Preprocessor Object: SF_SDPP Version 1.1 <Build 1>
              nort successfully validated the configuration!
nort exiting
ab1006@lab1006-HP-280-G4-MT-Business-PC:-$ []
```

Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds lab1006@lab1006-HP-280-G4-NT-Business-PC:-\$

```
F Oa: u<a.rmevv i.ii zee e7zJ z8:*>:4 Om cp i t0CA£ RUNS
```

# This file intentionally does not come with signatures. Put your local

 Open ▼
 A
 \*saort.conf
 Save ■ □□□

 /etc/prior:
 Save ■ □□□

ci ncl ude \$Rurc PAin/ r ie dent ry . rules

¥ T nc \.ude \$RULE PATN / F z \ e mul I T med za . r u\.e s

#include \$RULE\_PATH/file-pdf.rules include \$RULE\_PATH/finger.rules

#include \$RULE\_PATH/icmp-info.rules #include \$RULE PATH/icmp.rules

include sRULE PATH/info.rules #include sRULE\_PATH/malware-backdoor.rules #include SRULE\_PATH/malware-cnc.rules #include sRULE\_PATH/malware-other.rules #include sRULE\_PATH/malware-tools.rules

include SRULE PATH/netbios.rules

include \$RULE\_PATH/oracle.rules #include \$RULE\_PATH/os-linux.rules #include \$RULE\_PATH/os-other.rules

#include \$RULE\_PATH/policy-multimedia.rules

include SRULE\_PATH/pop3.rules

```
18/10/62 ibbl 10/04-iip-26/0
19/66-144-88:20.237384
19/66-144-88:20.684974
19/66-144-88:23.684974
19/66-144-88:23.684911
19/66-144-88:23.684911
19/66-144-88:24.1589765
19/66-144-88:24.1589691
19/66-144-88:24.685913
19/66-144-88:25.695930
19/66-144-88:25.695930
19/66-144-88:26.719633
10/66-144-88:27.743395
10/66-144-88:27.743395
10/66-144-88:28.767752
10/66-144-88:28.767753
                                                Caught Int-Signal
60lab1006-HP-280-G4-MT-Business-PC:-$
```

Conclusion: In conclusion, this assignment involved the installation and configuration of Snort, a powerful Intrusion Detection System. By following the step-by-step instructions, we successfully installed Snort, edited its configuration file, and executed rules to detect ICMP activities. This hands-on experience enhanced our understanding of network security and IDS functionality.