Snort IDS Project Report

Title: Installing and Configuring Snort on Ubuntu Server (VirtualBox Lab)

Author: Rishav K Thapa

Date: 7/6/2025

Introduction

This project demonstrates how to install and configure Snort, an open-source Intrusion Detection System (IDS), on an Ubuntu Server using VirtualBox. Snort monitors traffic for suspicious activity and logs alerts based on predefined rules. This is part of a blue team security lab where real attacks are simulated using Nmap to test detection capabilities.

Objective

- Learn how to install and configure Snort on a Linux server.
- Simulate real attacks using tools like Nmap.
- Detect and analyze traffic with Snort alerts.
- Build a reusable IDS lab environment for future learning and experimentation.

Tools Used

- Ubuntu Server 22.04 (VirtualBox)
- Snort 2.9.15.1
- Nmap (for scanning and attack simulation)
- nano (for editing config files)
- VirtualBox (for creating virtual lab)

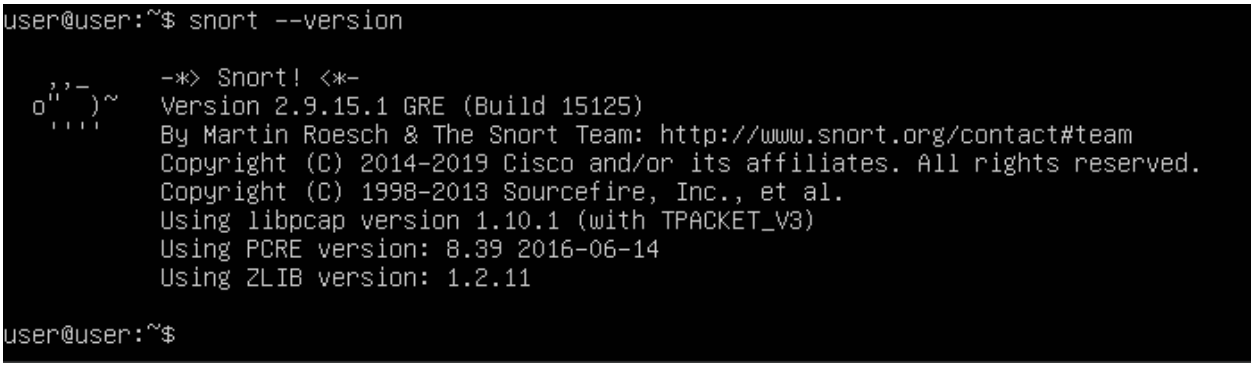
Lab Deployment Overview

In this demonstration, we set up Snort IDS on an Ubuntu Server running inside VirtualBox. The server was configured with a static IP, and Snort was installed, tuned with the correct HOME_NET, and then tested using basic Nmap scanning techniques from a Kali Linux machine. The goal was to verify if Snort could detect and log malicious or suspicious network traffic in a controlled lab environment.

Step 1: Install Snort

- sudo apt update
- sudo apt install snort -y

Step 2: Check Snort Version



- snort --version

Step 3: Verify Snort Files

```
user@user:~$ cd /etc/snort/
user@user:/etc/snort$ ls
attribute_table.dtd    file_magic.conf   rules              threshold.conf
classification.config  gen-msg.map       snort.conf         unicode.map
community-sid-msg.map  reference.config  snort.debian.conf
user@user:/etc/snort$ rano snort.conf
```

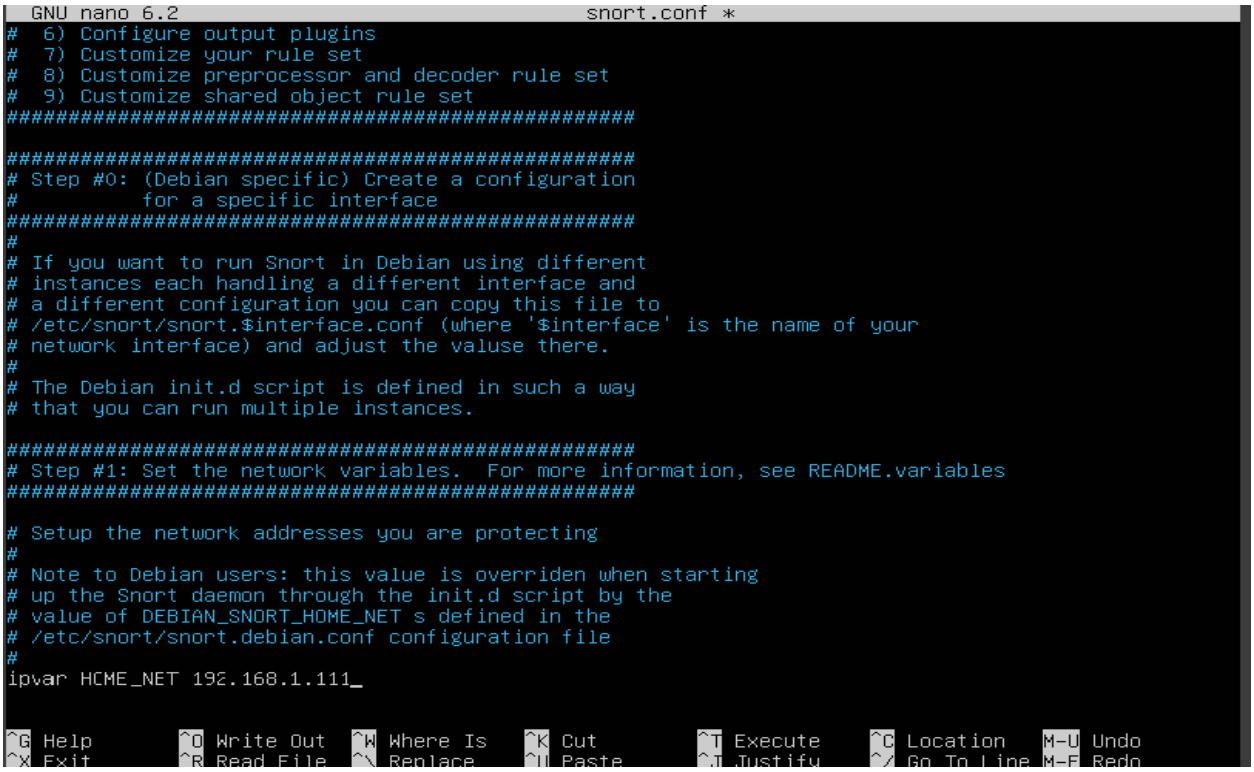
- cd /etc/snort
- ls

Important files and directories:

- ✓ snort.conf
- ✓ classification.config
- ✓ reference.config
- ✓ threshold.conf
- ✓ rules/
- ✓ unicode.map
- ✓ snort.debian.conf

Step 4: Configure HOME_NET in snort.conf

- sudo nano /etc/snort/snort.conf
- ipvar HOME_NET 192.168.1.111



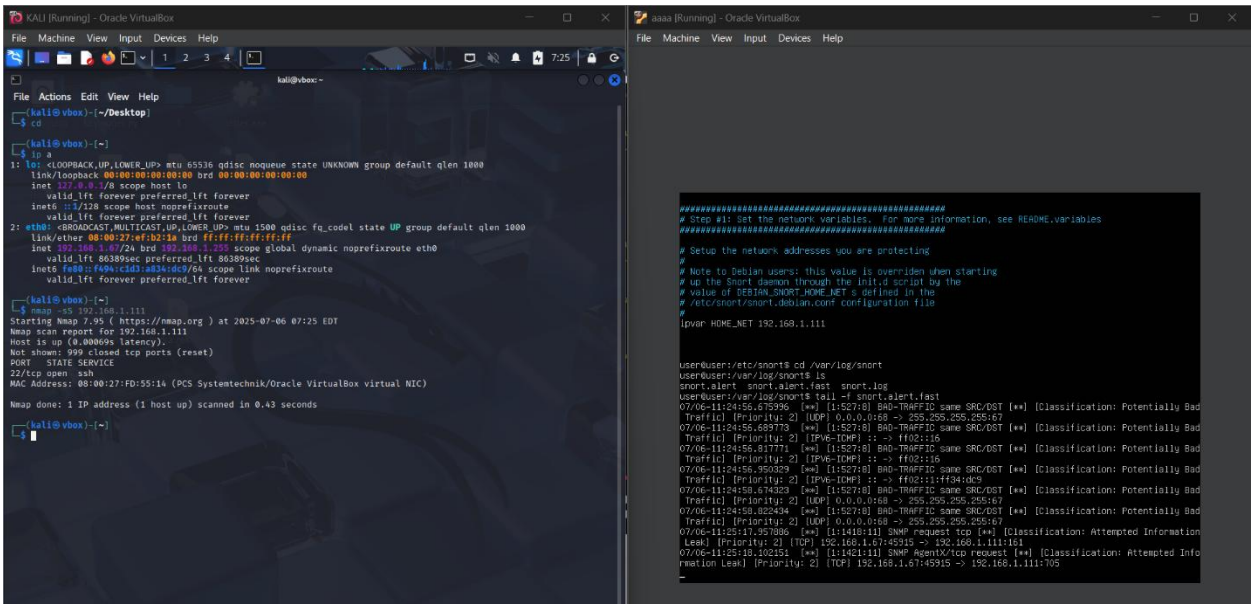
Step 5: Start Snort and Monitor Traffic

On Ubuntu server:

- cd /var/log/snort
- ls
- tail -f snort.alert.fastnmap -sS 192.168.1.111

On kali machine:

nmap -sS 192.168.1.111



Snort successfully captured and logged various types of network activity, including suspicious traffic triggered by simulated Nmap scans. The alerts were generated in real time and stored in the snort.alert.fast log file, confirming that Snort was actively monitoring the interface and applying its detection rules. This validates that our configuration is working correctly, and the IDS is functioning as expected within the virtual lab environment.

Results & Observations

- Snort successfully detected simulated attacks from Nmap.
- Alerts for "BAD-TRAFFIC", SNMP scanning, and UDP broadcasts were captured.
- Confirmed Snort is properly monitoring the interface and logging alerts based on traffic patterns.

Problems Faced

- Snort failed to start initially due to an incorrect interface name.
- Need to manually fix the HOME_NET variable.
- Required enabling the correct NIC (eth0) in VirtualBox settings.
- Rules needed to be adjusted for fine-tuned detection.

Future Enhancements

- Add custom Snort rules for known threats.
- Integrate Snort with ELK Stack or Wazuh for log visualization.
- Set up email alerts or syslog forwarding.
- Automate rule updates and configuration backups.

Conclusion

This project helped me understand how Snort detects network intrusions and how to set up a working IDS environment inside a virtual lab. The process also involved troubleshooting configurations and simulating real-world attacks. This forms a strong foundation for further work in blue teaming and network security analysis.

References

- https://www.snort.org/
- https://ubuntu.com
- https://nmap.org
- https://www.kali.org