# WEEK 1

* Introduction to NIDS and Snort

# P'tçWhat is NIDS (Network Intrusion Detection System)?

A **Network Intrusion Detection System (NIDS)** is a **security tool** that monitors network traffic in real-time to detect:

* + Malicious activity
  + Suspicious behavior
  + Policy violations
  + Attack signatures (e.g., port scans, malware, exploits)

C˛\* **How it works:**

* + It captures packets from a network interface.
  + It analyzes them using rules, heuristics, or machine learning.
  + If suspicious activity is detected, it logs or alerts the administrator.

- '▸´f× **What is Snort?**

**Snort** is one of the most widely used **open-source NIDS tools**, developed by **Martin Roesch**

and now maintained by **Cisco**.

\_5†— \_5 **Snort can function as:**

* + A **Packet Sniffer** (like Wireshark)
  + A **Packet Logger** (stores network traffic)

# A Real-time Intrusion Detection System

•˙Q **Snort Features:**

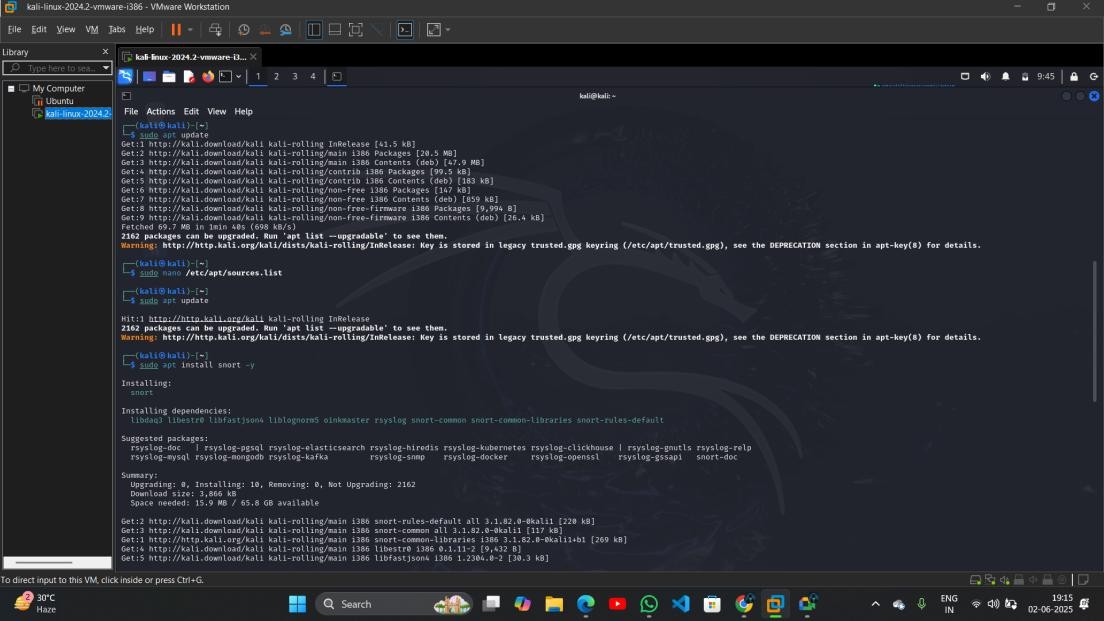
# Feature Description

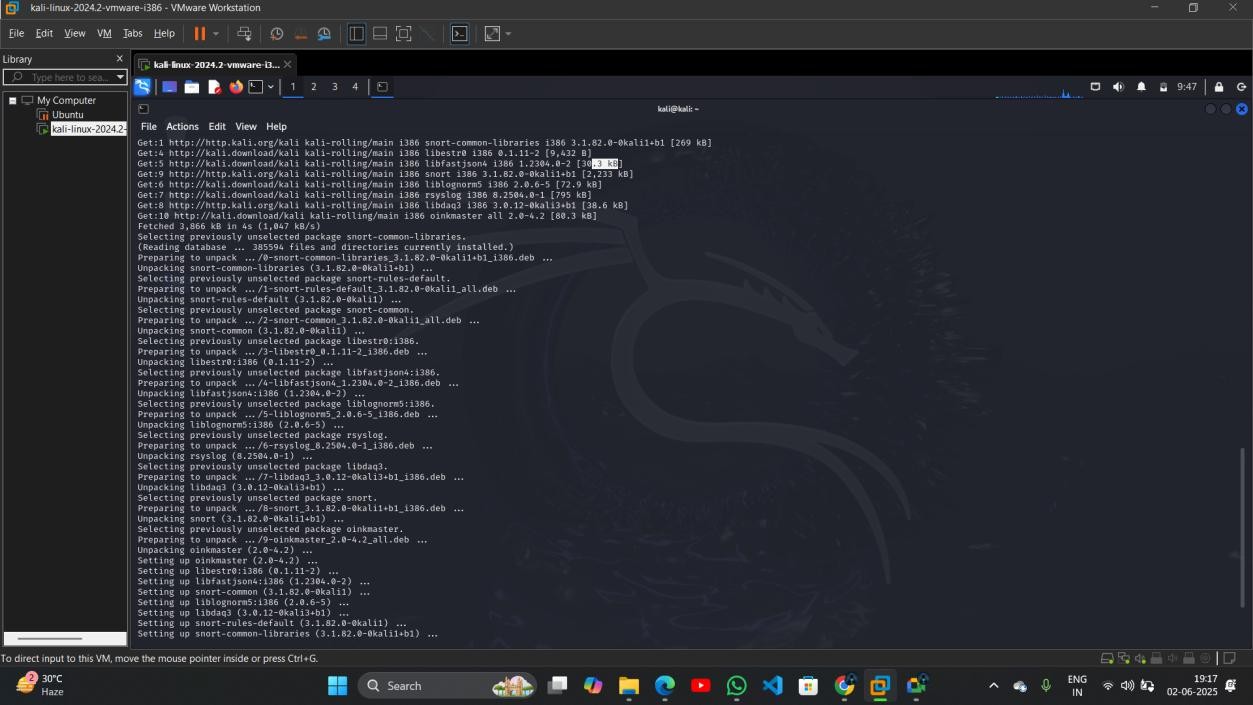
Signature-based Detection Detects known threats using pre-defined rule sets Protocol Analysis Inspects network protocols (TCP, UDP, ICMP, etc.) Logging & Alerting Logs suspicious activity or sends alerts

Custom Rule Creation Users can write their own detection rules Real-time Traffic Analysis Works on live traffic from network interfaces

* Install Linux (Ubuntu/Kali) Done
* Install and verify Snort

1. sudo apt update
2. sudo apt install snort





* + Basic Linux command-line navigation

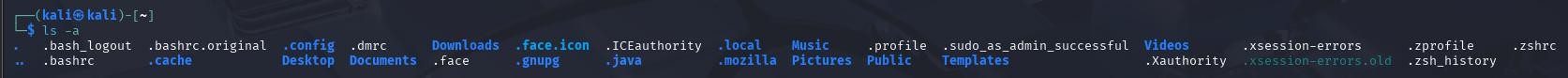
1. pwd: Show current working directory



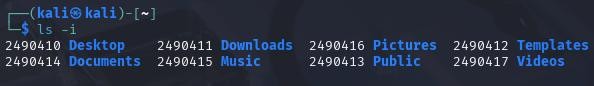
1. ls: List files in current folder



1. ls -a: Show hidden files



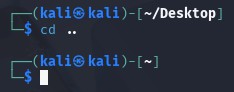
1. ls -l: Show detailed file list



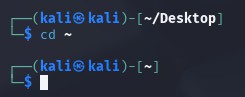
1. cd folder/: Change directory



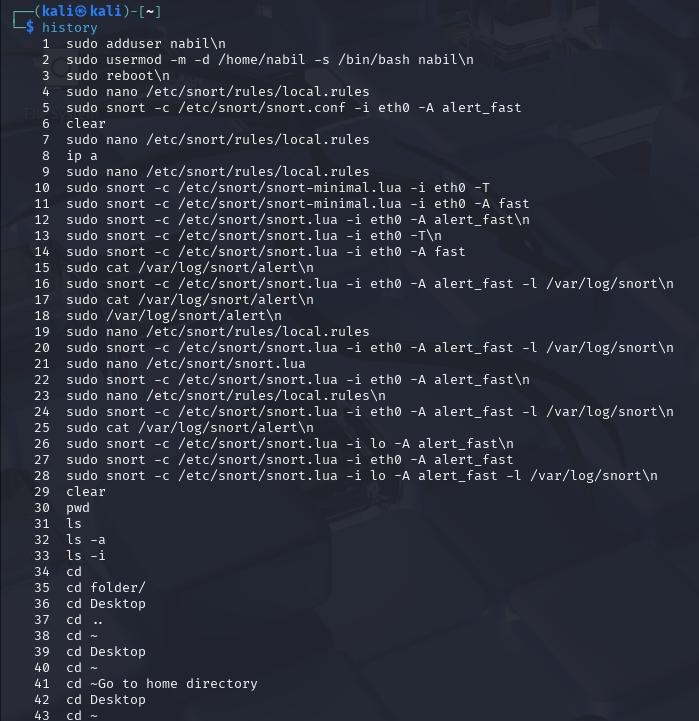
1. cd .. : Go up one directory



1. cd ~ : Go to home directory



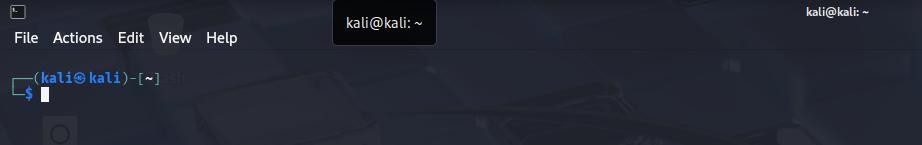
1. history: Show command history



1. clear: Clear terminal screen



1. exit: Exit terminal session



After enter the command ‘exit’ the terminal shutdown and we see desktop interface

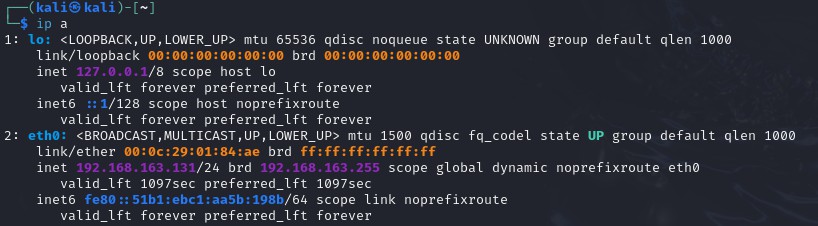


# WEEK 2

* + Identify active network interface Commands:

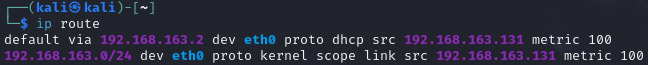
# Method 1: Using ip a (recommended)

ip a



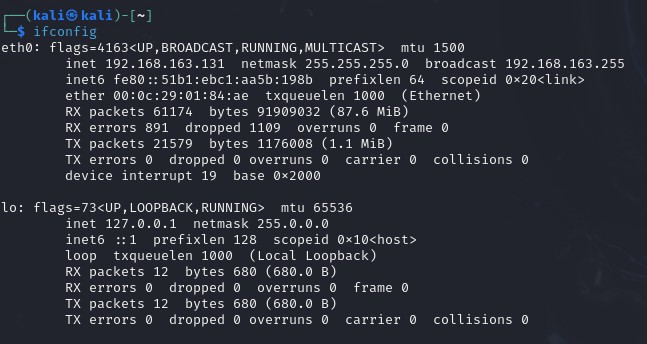
# Method 2: Using ip route

ip route



# Method 3: Using ifconfig (legacy)

Ifconfig



# Method 4: With nmcli (if using NetworkManager)

nmcli device status

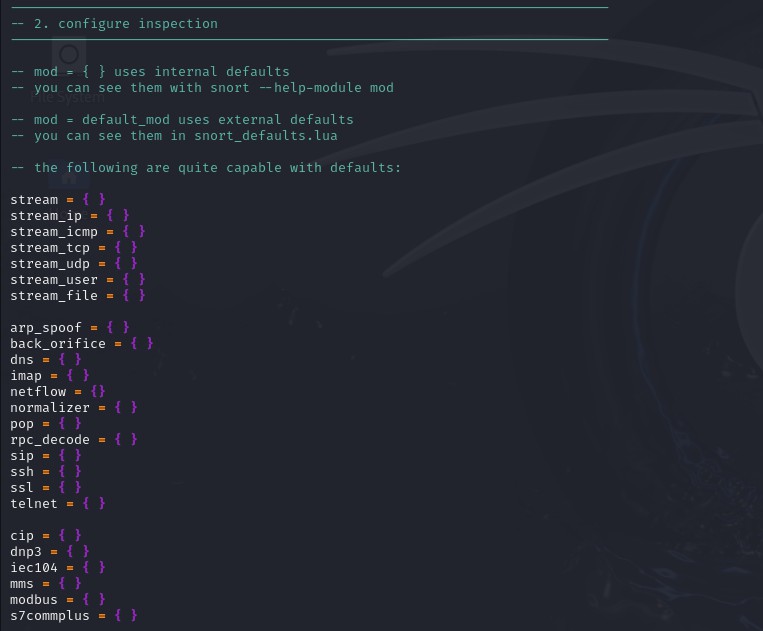
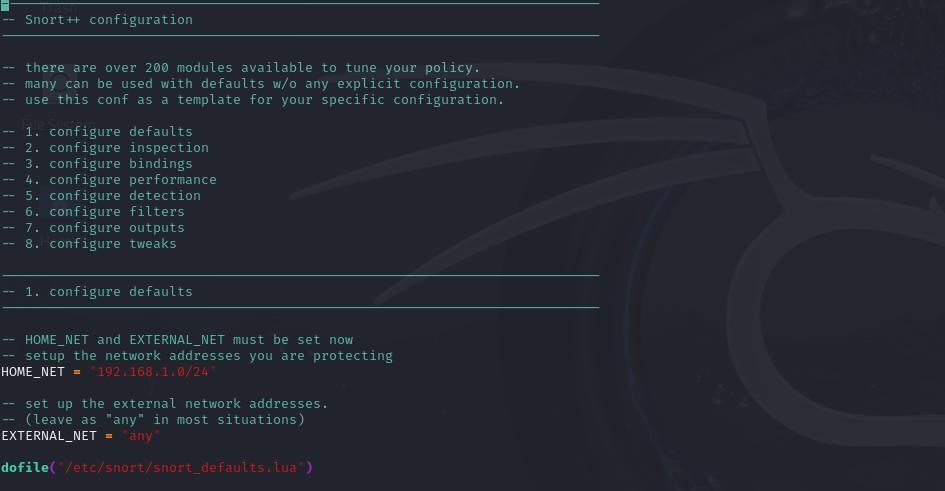


# Configure Snort with monitored IP range

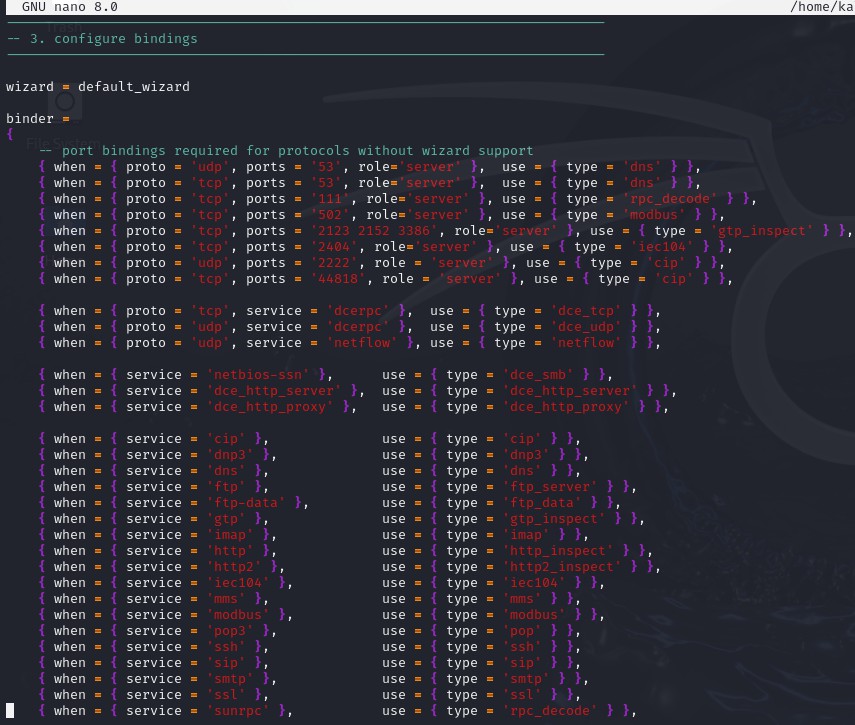
**1. Define your monitored IP range in your Lua config**

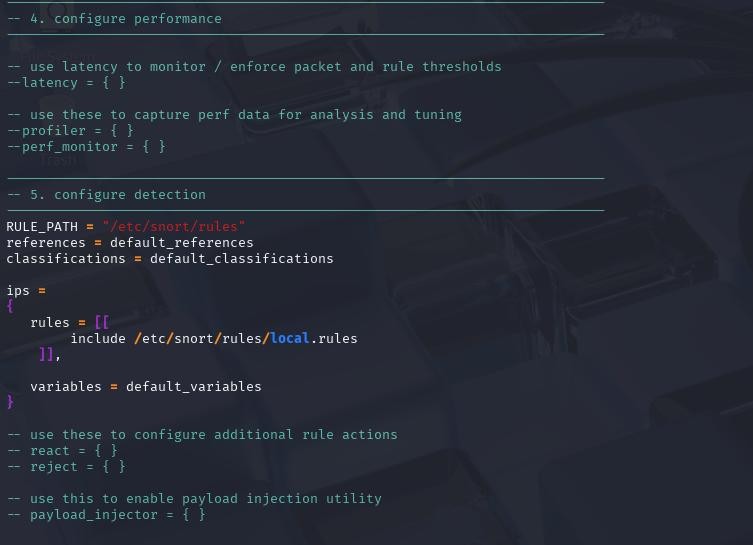
Open your Snort config file (/home/kali/snort.lua): nano /home/kali/snort.lua



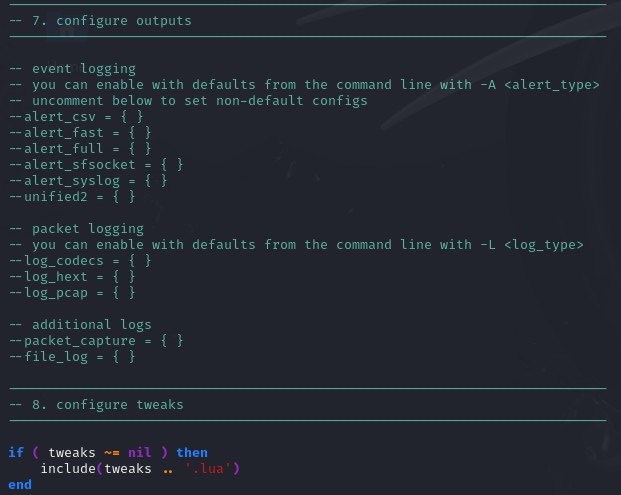












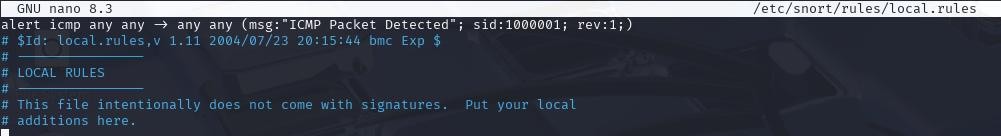
Edit local.rules file:

sudo nano /etc/snort/rules/local.rules

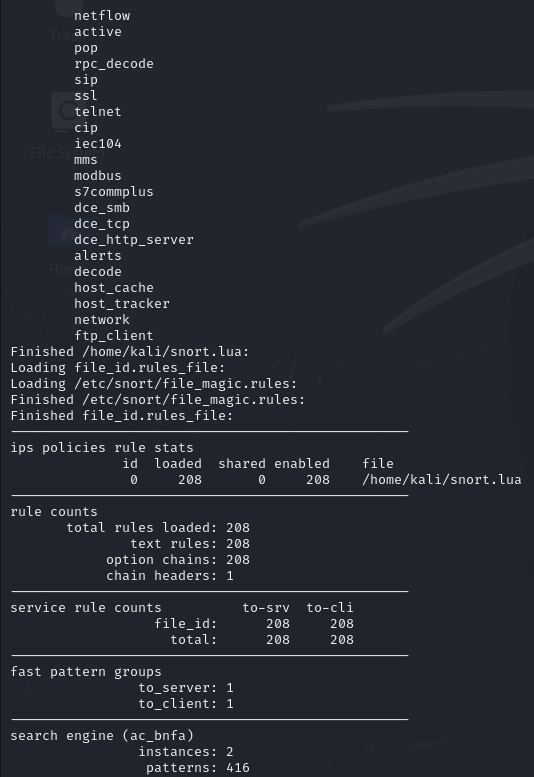
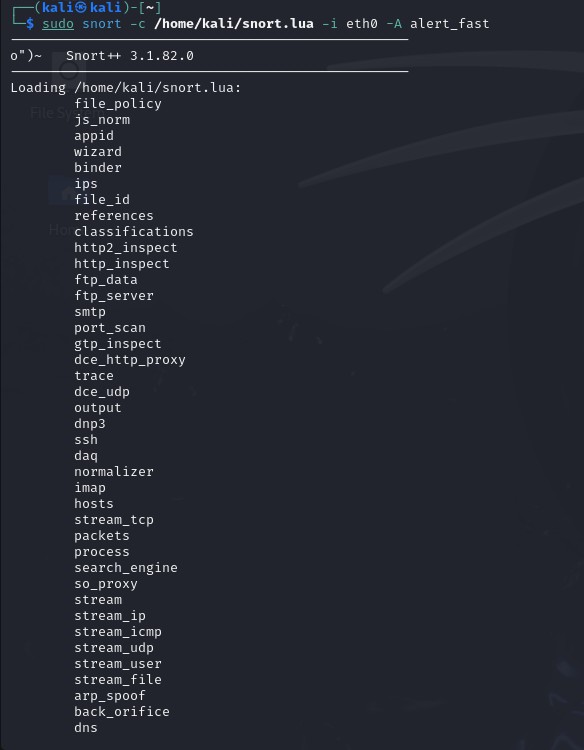


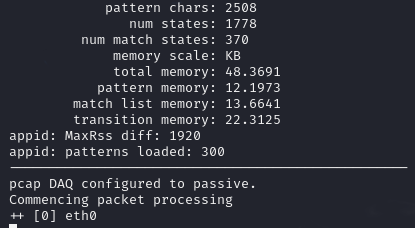
Add this rule to detect all ICMP packets:

# alert icmp any any -> any any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)

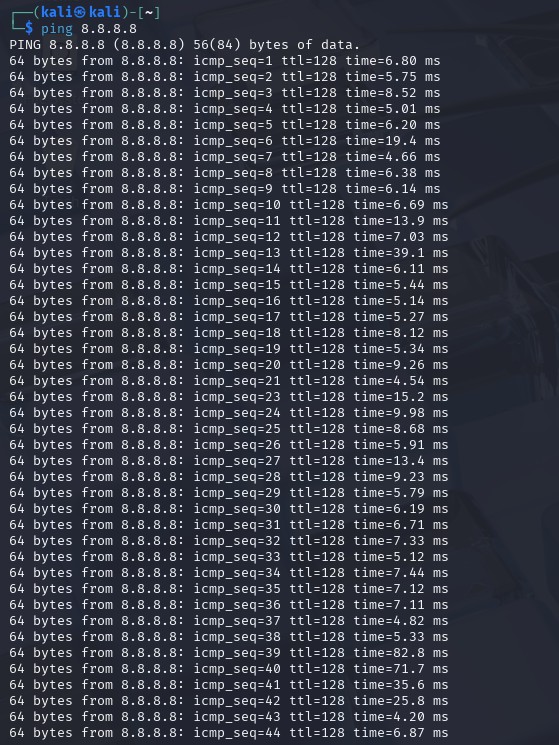
****

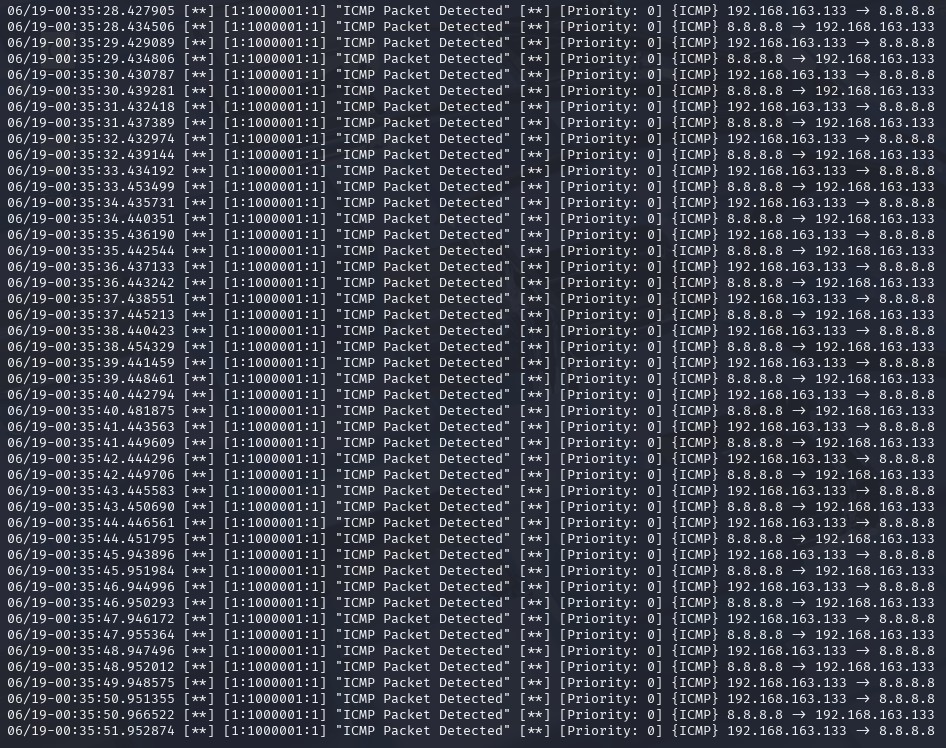
sudo snort -c /etc/snort/snort.lua -i eth0 -A alert\_fast



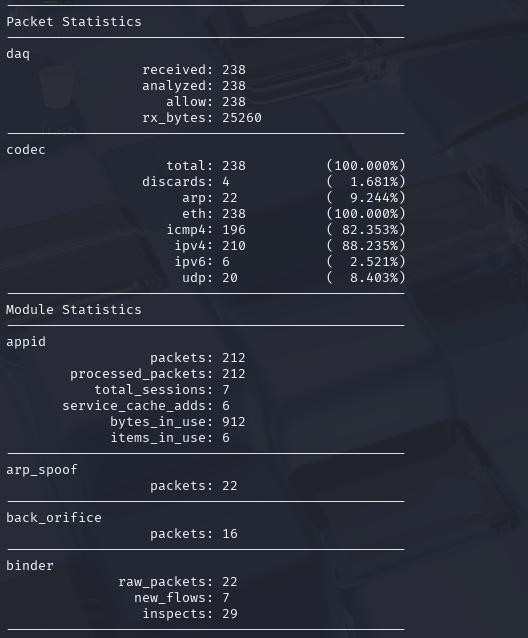


Now open new terminal tab and ping: Ping 8.8.8.8

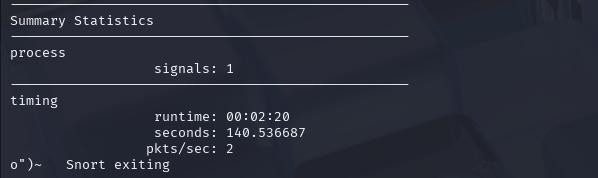


After Pinging, Go back where you started snort you’ll detect Packets:

After killing or stopping the process:

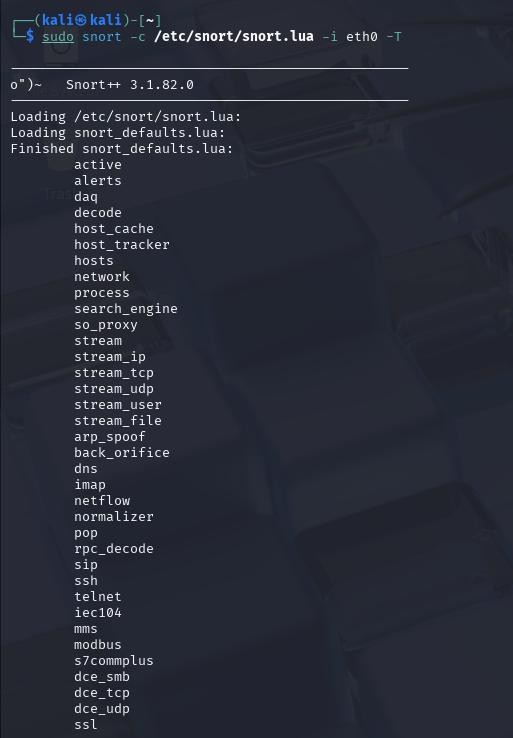


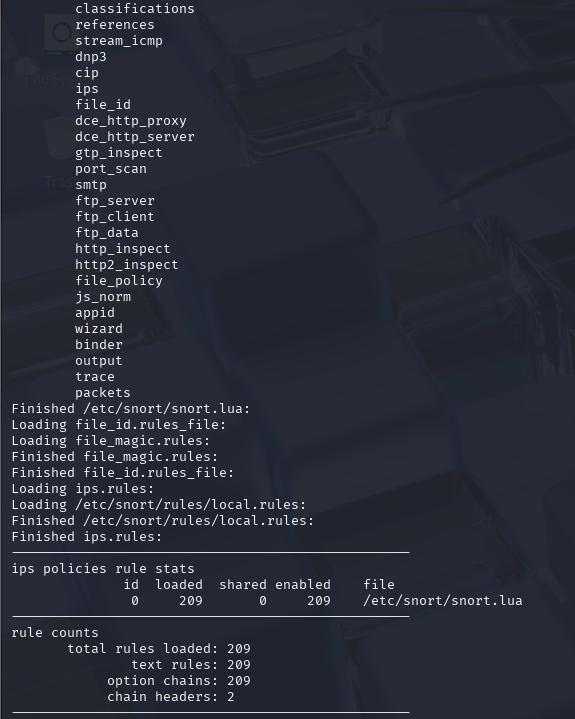


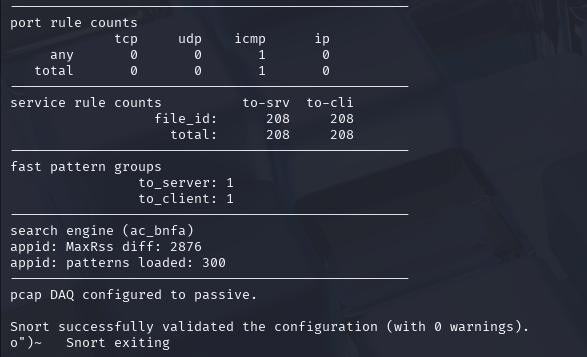


# Run Snort in detection mode

sudo snort -c /etc/snort/snort.lua -i eth0 -T

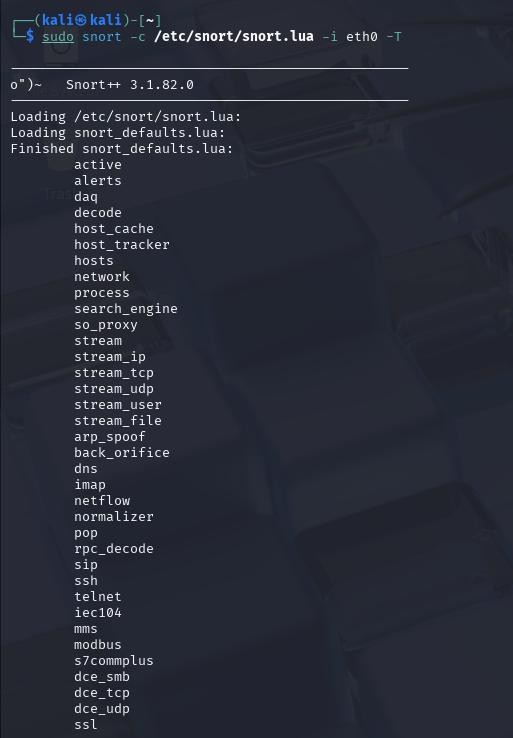


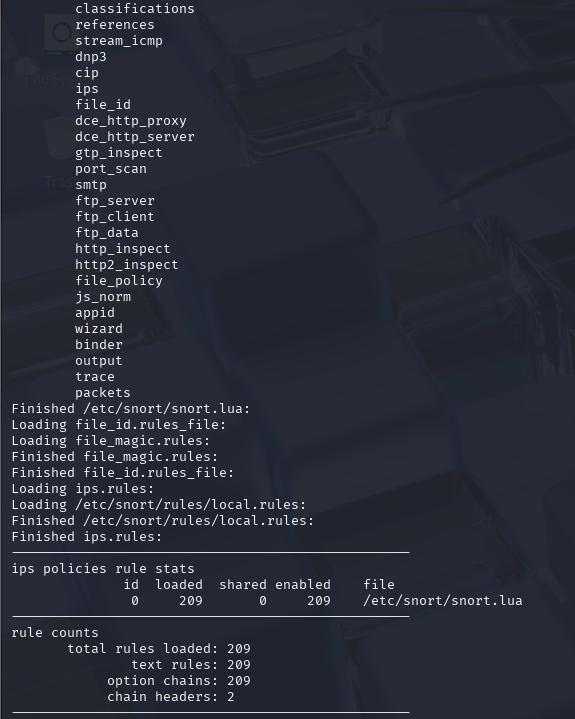


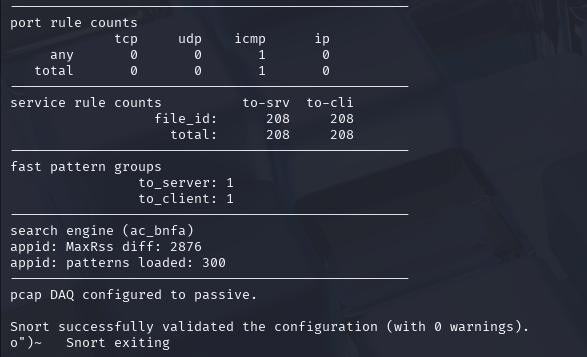


# Monitor live traffic and alerts

sudo snort -c /etc/snort/snort.lua -i eth0 -T



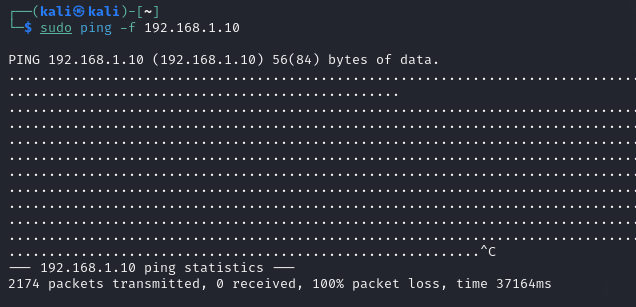




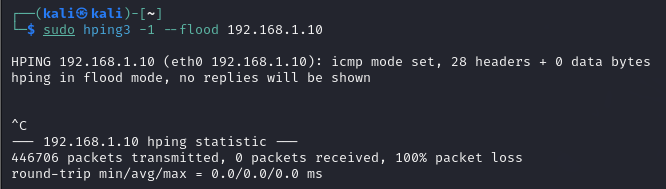
# Week 3

* **Simulate attacks (e.g., ping flood)**

1. Basic Ping Flood (Using ping) sudo ping -f 192.168.1.10



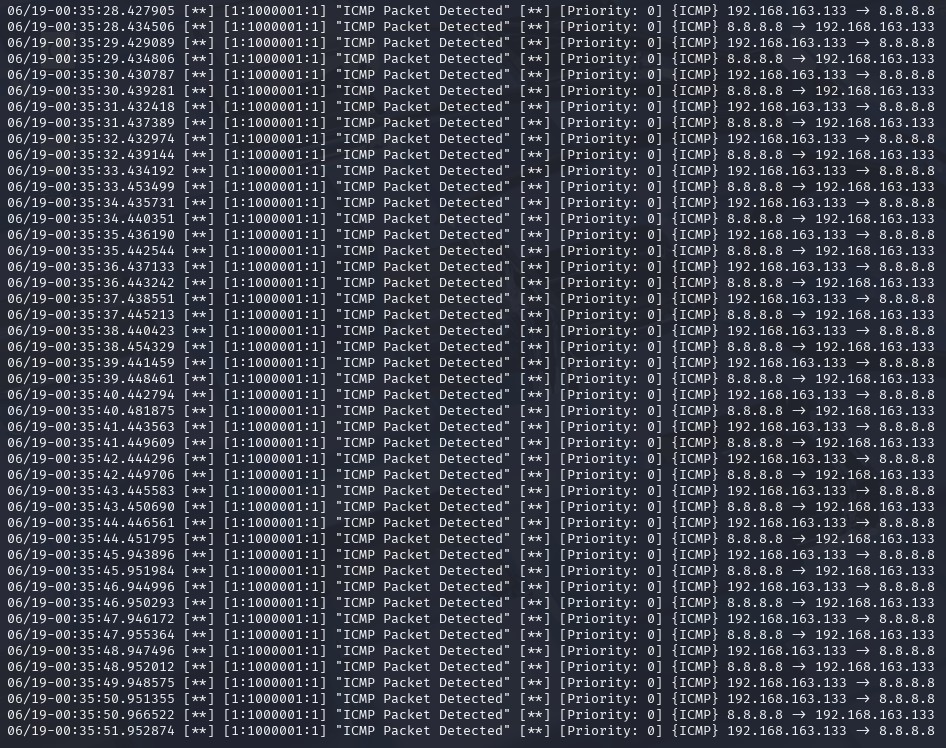
1. Advanced Ping Flood (Using hping3) sudo hping3 -1 --flood 192.168.1.10



Use this for learning:

* + How firewalls react
  + How to detect ICMP floods
  + How to create signatures for IDS/IPS (e.g., Snort or Suricata)

# Observe Snort alerts

****

* **Understand Snort alert formats**

Example in our case:

06/19-00:35:28.427905 [\*\*] [1:1000001:1] "ICMP Packet Detected" [\*\*] [Priority: 0]

{ICMP} 192.168.163.133 -> 8.8.8.8

|  |  |
| --- | --- |
| **Field** | **Description** |
| **Timestamp** | 06/19-00:35:28.427905 – The date and time when the alert was triggered. Format: MM/DD-HH:MM:SS.milliseconds. |
| **Alert Markers** | [\*\*] – Visual separators to distinguish alert sections. |
| **Rule Metadata** | [1:1000001:1] – This consists of:   * 1: Generator ID (GID), indicates Snort itself triggered the alert. * 1000001: Signature ID (SID), uniquely identifies the rule. * 1: Rule revision number. |   | **Message** | "ICMP Packet Detected" – The alert message defined in the msg: field of the rule. |  | **Priority** | [Priority: 0] – Indicates severity (0 = lowest). Priority is set manually in the rule or inferred. |  | **Protocol** | {ICMP} – The detected packet's protocol. |  | **Source → Destination** | 192.168.163.133 -> 8.8.8.8 – The IP addresses of the packet's origin and destination. | |

# Review alert logs

Review in our case:

06/19-00:35:28.427905 [\*\*] [1:1000001:1] "ICMP Packet Detected" [\*\*] [Priority: 0]

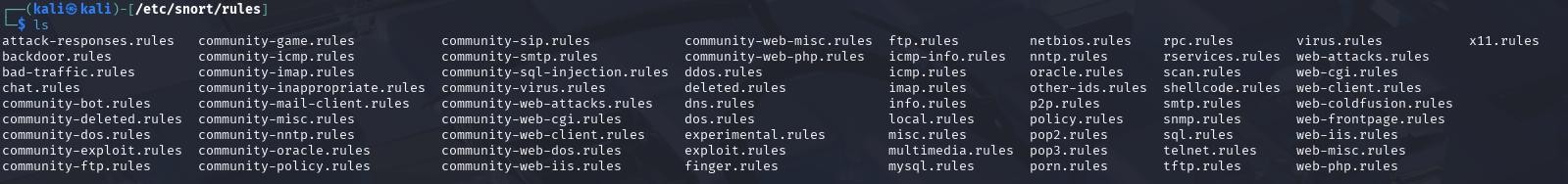
{ICMP} 192.168.163.133 -> 8.8.8.8

This log tells you:

* + A Snort rule was triggered by **ICMP traffic**
  + The alert was logged on **June 19 at 00:35:28**
  + The **source IP** was your machine (192.168.163.133)
  + The **destination** was Google DNS (8.8.8.8)

**WEEK 4**

* **Explore default Snort rules and structure**

****

* **Learn rule components (actions, protocols, etc.)**

Rule:

alert icmp any any -> any any (msg:"ICMP Packet Detected"; sid:1000001; rev:1;)

|  |  |  |
| --- | --- | --- |
| **Component** | **Value** | **Description** |
| **Action** | alert | Tells Snort to generate an alert when this rule matches traffic |
| **Protocol** | icmp | Matches ICMP packets (used in ping, traceroute, etc.) |
| **Source IP** | any | Matches traffic from any source IP |
| **Source Port** | any | ICMP doesn't use ports, but format requires this |
| **Direction** | -> | Matches traffic from source to destination |
| **Destination IP** | any | Matches traffic to any destination IP |
| **Destination Port** | any | Placeholder, ICMP does not use ports |
| **Options** | (msg:"ICMP Packet Detected"; sid:1000001; rev:1;) | Rule-specific metadata and message |

# Prepare a basic report with screenshots on configuration and alerts

The screenshots for the following tasks have already been included in their relative sections in this report:

1. Identifying active network interface
2. Configuring Snort with the monitored IP range.
3. Running Snort in detection mode.
4. Monitoring live traffic and alerts.
5. Simulating attacks such as ping and ping flood.
6. Observing Snort alerts.
7. Understanding Snort alert formats.
8. Reviewing Snort alert logs.

# What I got to learn?

Using Snort as an Intrusion Detection System (IDS), I was able to obtain hands-on experience in network security monitoring. Important lessons learnt include:

1. Setting up and installing Snort on a Linux computer.
2. Finding and keeping an eye on active network interfaces.
3. Creating and evaluating unique Snort rules with appropriate syntax and organisation.
4. Using traffic simulation (ICMP, ping flood, etc.) to set off alarms.
5. Examining log files and Snort alert analysis.
6. Using key Linux commands for log analysis and configuration.

My knowledge of network traffic analysis and real-time intrusion detection has improved as a result of this experience.

# Summary – Month 1

It was found for Month 1 that installation, configuration, and testing of Snort were undertaken successfully. One custom ICMP detection rule was created, and alerts were generated using the ping and hping3 tools. Snort was started in detection mode, and the logs were reviewed for verification of rule efficacy.

While understanding and explaining the rule structure and alert format, screenshots were provided for all major steps, including rule setup, configuration, traffic simulation, and alert output.

All Month 1 milestones were achieved, thus setting a firm base for further developments and testing of the IDS.