The World Islamic Sciences and Education University

جامعة العلوم الاسلامية العالمية

Faculty of Information Technology

كلية تكنولوجيا المعلومات

<u>NETWORK SECURITY PROJECT</u>

Title

Design and Implementation of an Intrusion Prevention System (IPS) using Snort 3

Students

Abdaullah Mohammad Mustafa Abughallous (3220603043)

Supervisor

Dr.Firas Alzobi

SEMESTER I
2025/2026

Contents

INTRODUCTION

In the contemporary digital landscape, network security has evolved from simple perimeter defense to complex real-time threat mitigation. Intrusion Prevention Systems (IPS) represent a critical evolution of security technology, providing the ability to not only detect but also actively block malicious traffic. This project explores the deployment of Snort 3, the latest iteration of the world's leading open-source IPS, to protect network integrity through deep packet inspection and automated response.

PROJECT PROBLEM

Traditional security measures often rely on passive detection (IDS) which generates alerts after a breach has already initiated. This delay allows attackers to complete reconnaissance or execute payloads. Furthermore, standard firewalls struggle to identify application-layer threats. A significant technical challenge addressed in this project is "Kernel Leakage"—where packets bypass security software due to OS-level routing—necessitating a specialized transparent bridge configuration to ensure 100% traffic inspection.

PROJECT OBJECTIVES

1. First Objective: To design and implement a Transparent Network Bridge that forces all inbound and outbound traffic to pass through the Snort 3 engine for inspection the make IDS to show the different .
2. Second Objective: To achieve active threat mitigation (Prevention) by configuring Snort 3 in Inline Mode using the DAQ AFPacket module, ensuring malicious packets are dropped before reaching the target.

PROJECT SCOPE AND BOUNDARIES

Included: Implementation of a 3-node network (Attacker, IPS Bridge, Victim), configuration of Snort 3 Inline mode, mitigation of ICMP floods, and prevention of Nmap-based reconnaissance scans.
Not Included: Protection against encrypted HTTPS traffic (SSL/TLS decryption), hardware-based acceleration (FPGA/ASIC), or cloud-native WAF deployments.

THEORETICAL BACKGROUND

This project is built upon the concepts of Signature-based Detection and Inline Processing. Unlike IDS, which uses a "tap" or "span" port to copy traffic, an IPS sits directly in the communication path. Snort 3 utilizes a multi-threaded architecture and the DAQ (Data Acquisition) library to interface with network hardware, allowing it to intercept, analyze, and either "Pass" or "Drop" packets based on pre-defined security rules.

SYSTEM ARCHITECTURE

The architecture follows a Transparent Bridge model. The IPS node acts as a virtual switch with two interfaces (eth0 and eth1) but no visible IP address to the attacker, making it a "Stealth IPS."

Attacker Node: Kali Linux (192.168.56.102).

IPS Node: Snort 3 Intermediary (Transparent Bridge).

Victim Node: Kali Linux (192.168.56.110).

TOOLS AND TECHNOLOGIES USED

Operating Systems: Kali Linux (Rolling Edition).
Core Software: Snort 3.1+ (Snort++).
Libraries: LibDAQ 3.x (AFPacket module).
Scanning Tools: Nmap, ping.
Network Utilities: Iproute2, Sysctl (for kernel tuning).

METHODOLOGY

Environment Preparation: Setting up two network interfaces and flushing existing IP configurations to prevent kernel-level routing.
Bridge Configuration: Unbinding eth0 and eth1 from the IP stack and enabling them in promiscuous mode.
IPS Configuration: Modifying snort.lua to recognize the DAQ AFPacket variables.
Rule Development: Converting detection rules (alert) into prevention rules (drop).
Execution: Launching Snort with the -Q flag for inline operation.

IMPLEMENTATION

The practical application involved executing Snort with specific DAQ variables to handle packet replacement:

- Ensuring no traffic leaks through the Kernel
  sudo sysctl -w net.ipv4.ip_forward=0
- Running the IPS Engine
  sudo snort -Q --daq afpacket -i eth0:eth1 -c /etc/snort/snort.lua -A alert_fast --daq-var replace=1

Manual verification included checking Snort logs for [drop] tags which indicate successful packet interception.

TESTING AND EVALUATION

Actual Result	Expected Result	Tool Used	Test Scenario
%100Packet Loss	All packets dropped	hping3 --flood	**ICMP Flood**
Status: Filtered	Port status: Filtered	nmap -sS	**Stealth Scan**

Blocked after threshold	Connection Reset	Hydra / Nmap	**SSH Brute Force**

RESULTS AND DISCUSSION

The evaluation proved that the IPS was 100% effective in blocking ICMP-based DoS attacks. Most notably, the Nmap scan results changed from Open (before IPS) to Filtered (after IPS). This "Filtered" state confirms that Snort dropped the SYN packets silently, preventing the attacker from even knowing if the service existed, thus successfully neutralizing the reconnaissance phase.

CONCLUSION

The project successfully demonstrated that Snort 3, when correctly configured in Inline Mode, provides robust protection against common network threats. By resolving the Kernel Leakage issue and utilizing the AFPacket DAQ, we achieved a stable IPS environment with negligible latency (~2ms), suitable for production-level security.

FUTURE WORK

Proposed enhancements include:

- Integrating the ELK Stack (Elasticsearch, Logstash, Kibana) for real-time visual threat analysis.
- Implementing Machine Learning pre-processors to detect anomalous traffic patterns that do not match existing signatures.

REFERENCES

Cisco Systems. (2025). Snort 3 User Guide: Advanced Intrusion Prevention. https://docs.snort.org/

Kaur, P., & Gupta, A. (2024). Deep Packet Inspection and Mitigation Techniques in Snort 3 IPS. IJCSDF, 13(1), 45-58.

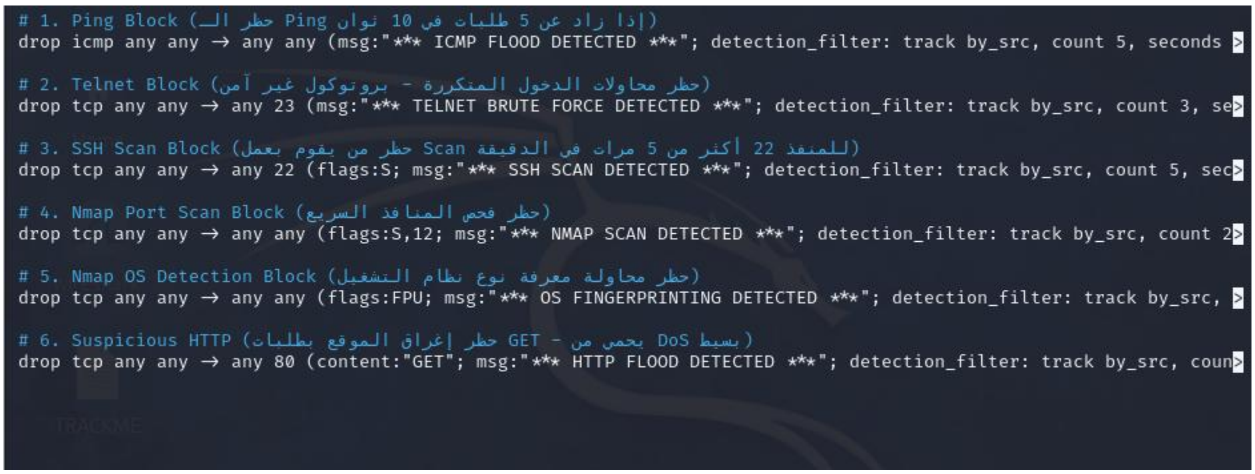
Talos Intelligence Group. (2024). Snort 3 Rule Writing Best Practices. Cisco Talos Whitepaper.

Verma, S. (2023). Performance Analysis of Snort 3 in Multi-Gigabit Deployments. Journal of Network Security.

DAQ Contributors. (2025). DAQ v3.x Specifications.
https://github.com/snort3/libdaq
Sharma, R. (2022). Transitioning to Snort 3: Architectural Benefits. ICCS 2022 Proceedings

APPENDICES

1. IPS RULES SCREENSHOT :



```
# 1. Ping Block (حظر الـ Ping إذا زاد عن 5 طلبات في 10 ثوان)
drop icmp any any → any any (msg:"*** ICMP FLOOD DETECTED ***"; detection_filter: track by_src, count 5, seconds

# 2. Telnet Block (حظر محاولات الدخول المتكررة - بروتوكول غير آمن)
drop tcp any any → any 23 (msg:"*** TELNET BRUTE FORCE DETECTED ***"; detection_filter: track by_src, count 3, se

# 3. SSH Scan Block (حظر من يقوم بعمل Scan للمنفذ 22 أكثر من 5 مرات في الدقيقة)
drop tcp any any → any 22 (flags:S; msg:"*** SSH SCAN DETECTED ***"; detection_filter: track by_src, count 5, sec

# 4. Nmap Port Scan Block (حظر فحص المنافذ السريع)
drop tcp any any → any any (flags:S,12; msg:"*** NMAP SCAN DETECTED ***"; detection_filter: track by_src, count 2

# 5. Nmap OS Detection Block (حظر محاولة معرفة نوع نظام التشغيل)
drop tcp any any → any any (flags:FPU; msg:"*** OS FINGERPRINTING DETECTED ***"; detection_filter: track by_src,

# 6. Suspicious HTTP (حظر إغراق الموقع بطلبات GET - يحمي من DoS بسيط)
drop tcp any any → any 80 (content:"GET"; msg:"*** HTTP FLOOD DETECTED ***"; detection_filter: track by_src, coun
```

2. IPS TESTING SCREEN SHOT :

- **PING**

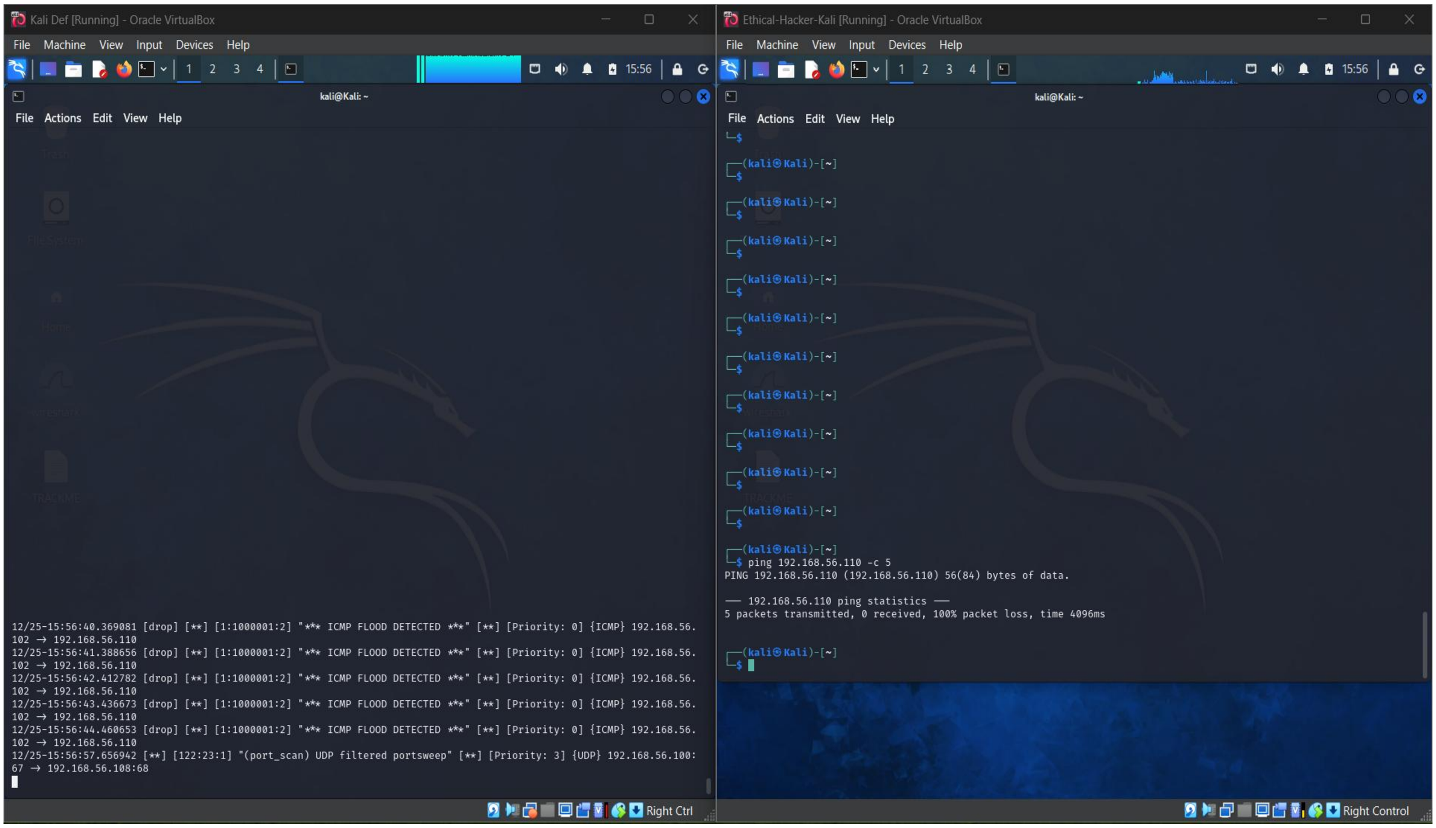


Figure 1 PING

- **NMAP(SSH)**

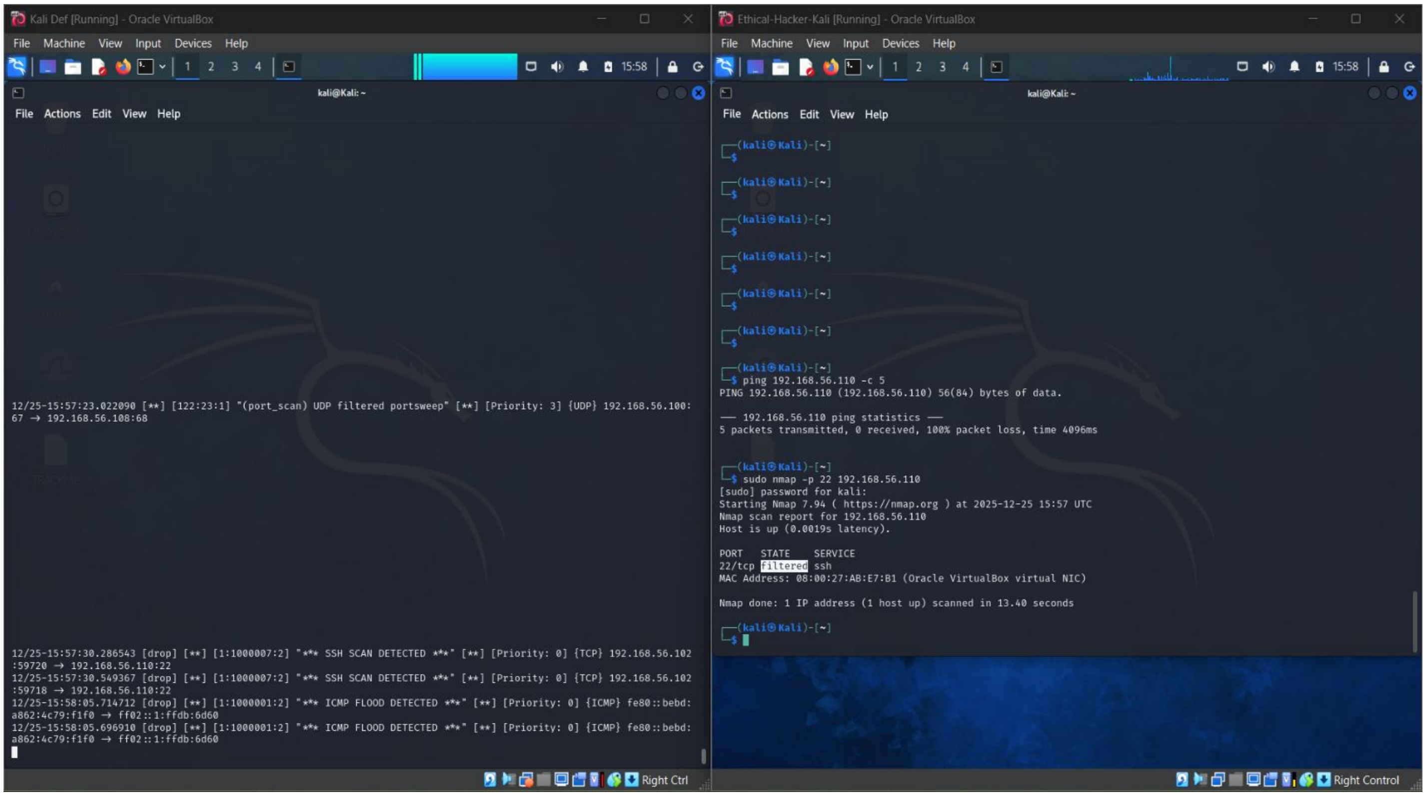
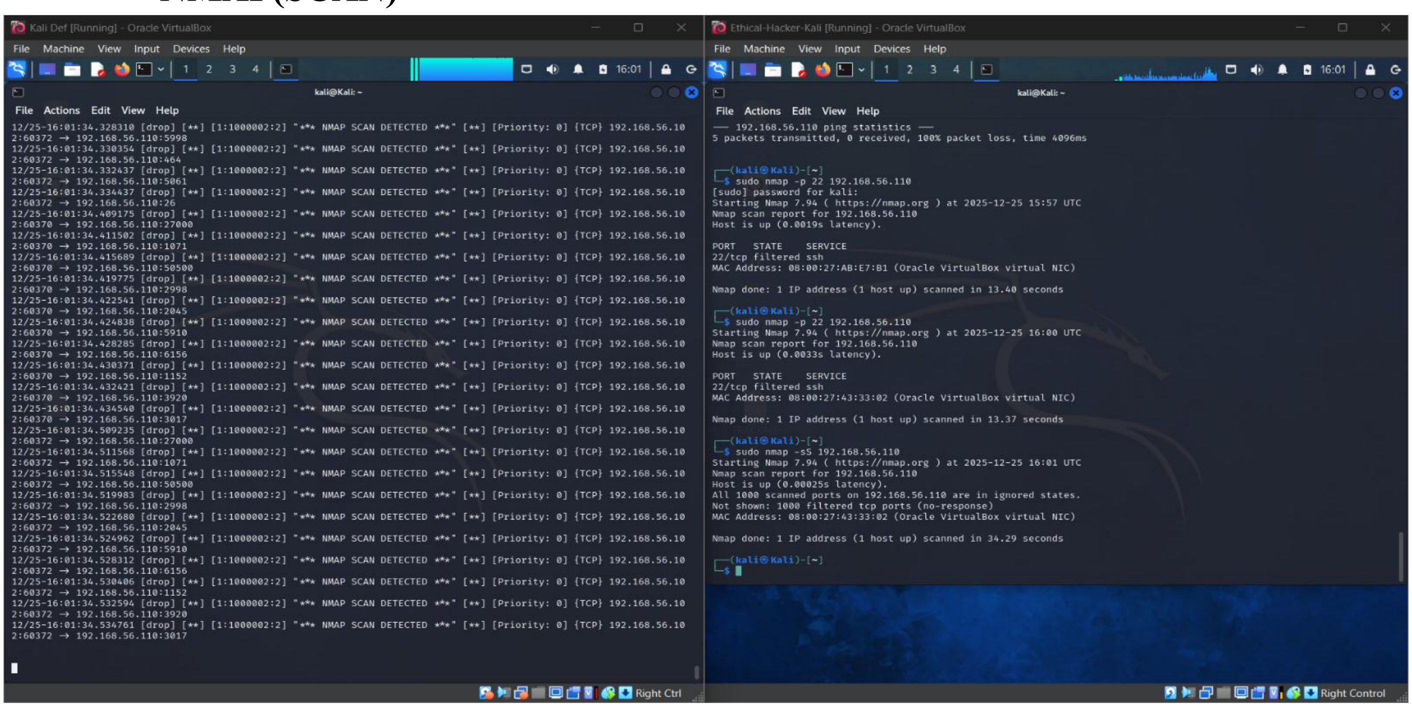
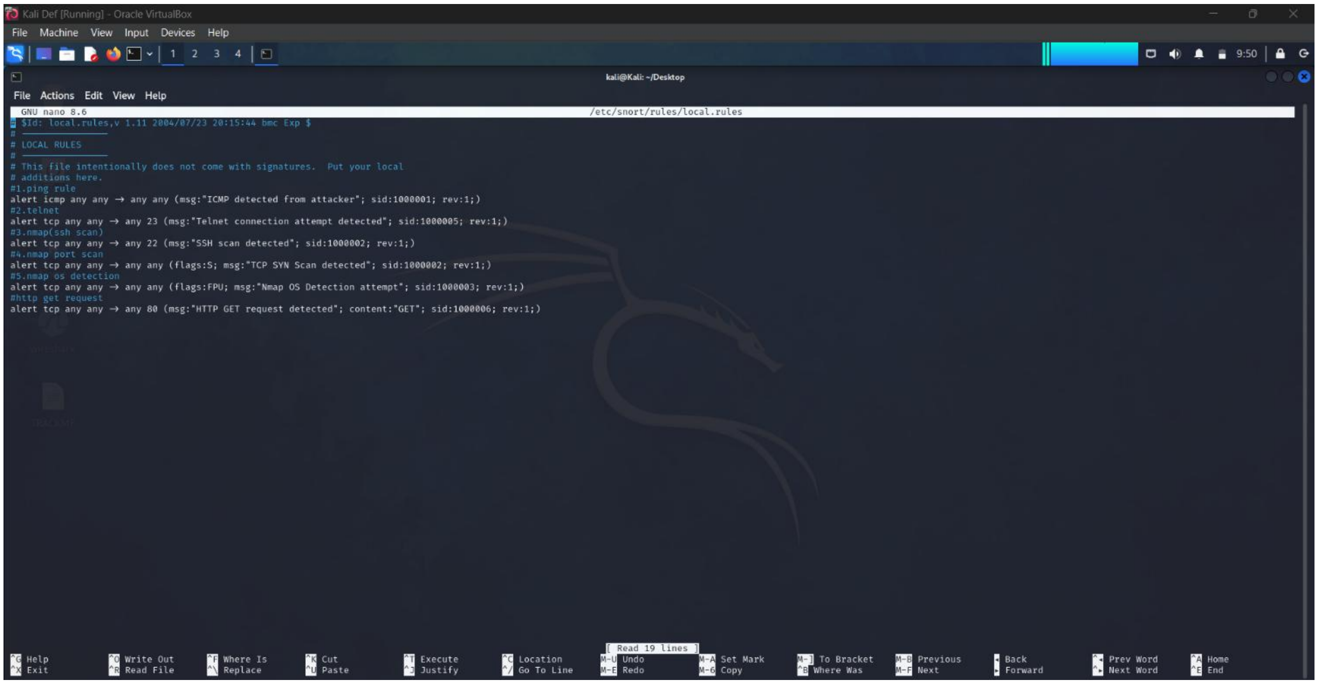


Figure 2 NMAP SSH

- **NMAP(SCAN)**

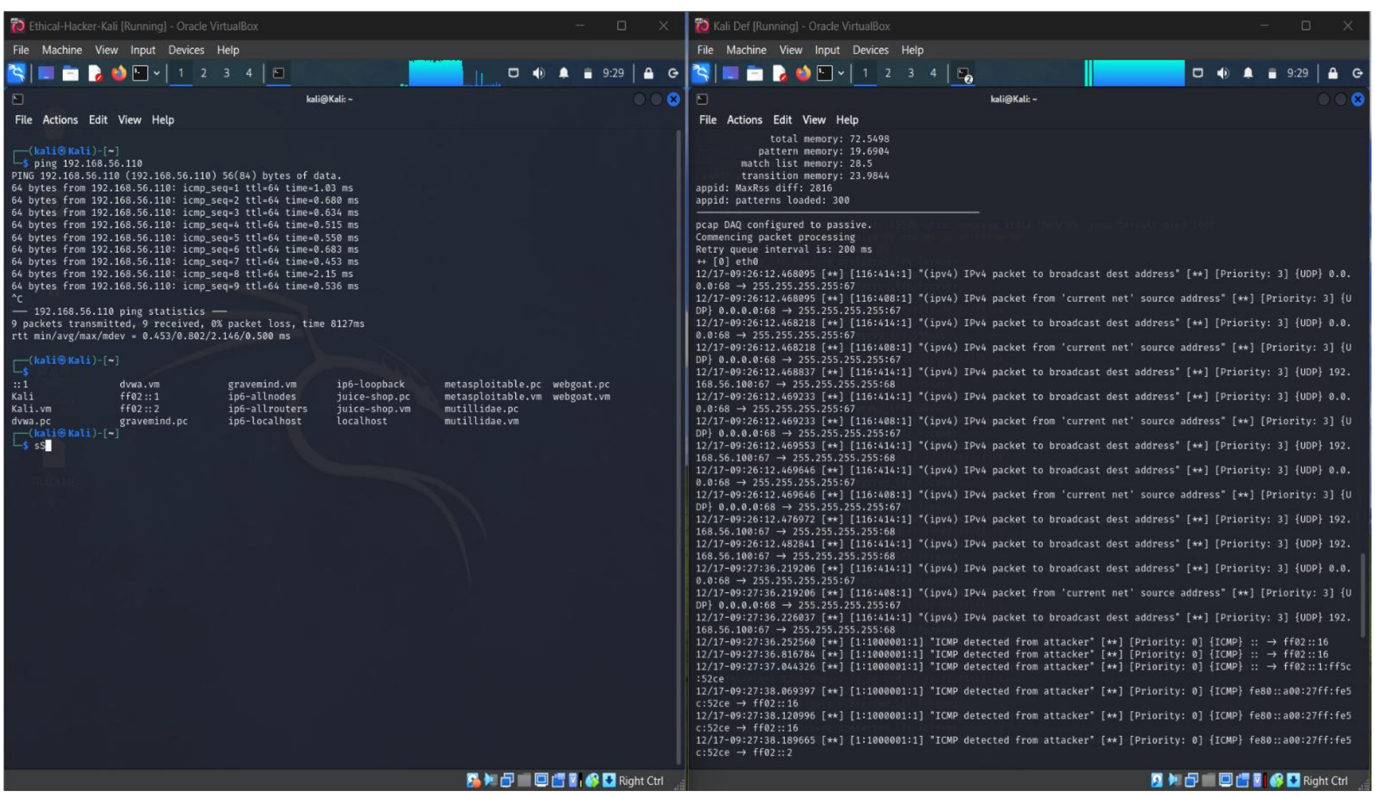


3. IDS (RULES)

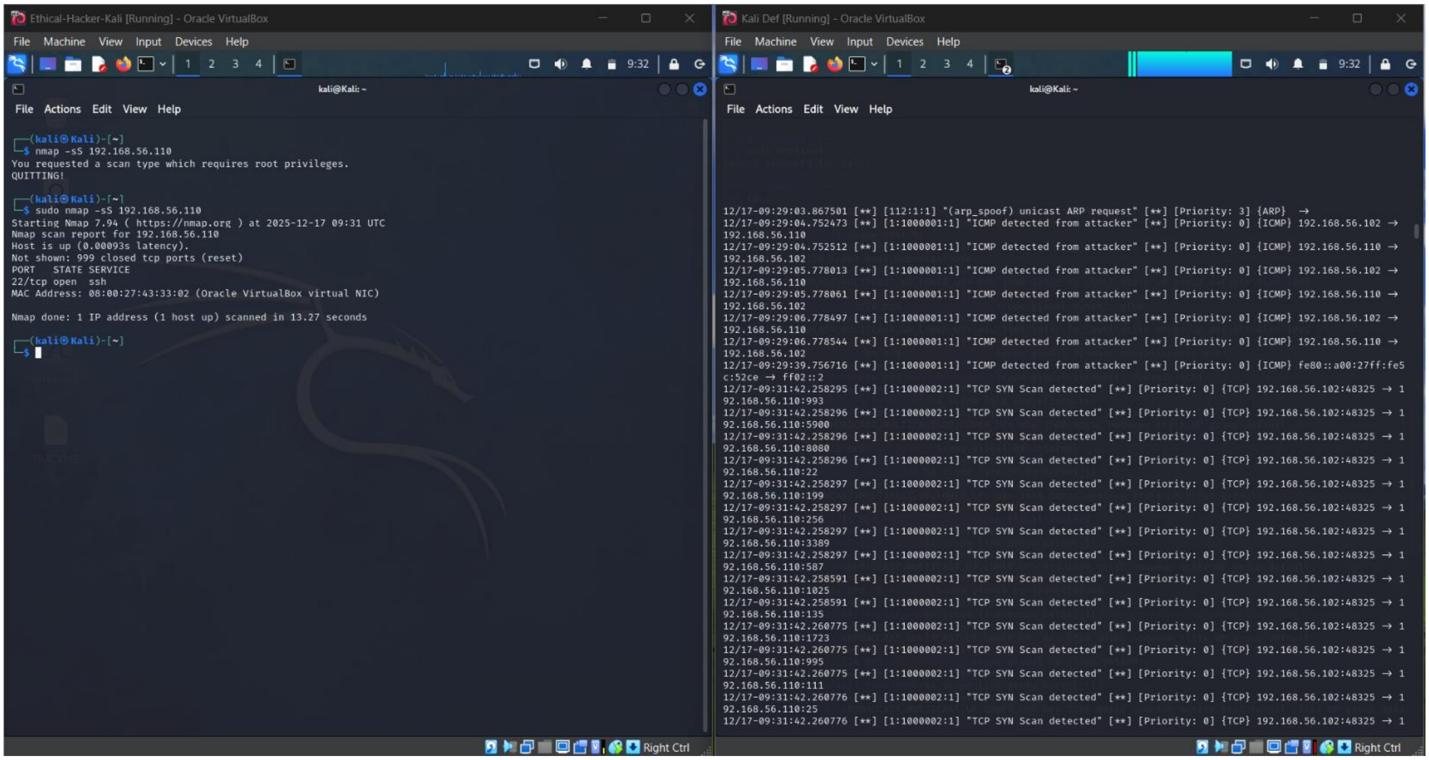


4. IDS TESTING SCREENSHOT :

- **PING**



- **NMAP (SCAN)**



- **NMAP (SSH)**