NETWORK IDS

Exploration and Proof of concept (POC)

Prathamesh Arun Kamble

17.08.2025
Intern Id-195

🔍 Tool Name : Network IDS

📌 Project Overview

This project implements a lightweight Network Intrusion Detection System (IDS) designed to analyze network traffic (live or from PCAP files) and detect common suspicious activities.

The IDS is built in Python using the Scapy library and runs directly from the terminal. It provides alerts for potential reconnaissance and exploitation attempts.

🔎 What is This Project?

This project is a lightweight Network Intrusion Detection System (NIDS) built using Python and Scapy. It monitors network traffic (live or from PCAP files) and raises alerts when it detects suspicious activity.

It focuses on early detection of attacks and scanning behavior that often happen before a bigger cyberattack.

🎯 Objective

To create a simple, extensible IDS capable of detecting:

ICMP flood/ping sweep activity

TCP SYN scan attempts

Half-open connections

Basic scanning patterns (NULL, FIN, repeated port probing)

⚙️ Technical Stack

Language: Python 3

Library: Scapy (for packet parsing and analysis)

Platform: Linux / Termux / Windows (with Python + Scapy installed)

🚀 How It Works

The IDS script (network_ids.py) performs packet inspection to identify malicious or unusual activity:

1. Reads Network Traffic

Works on live capture or stored .pcap files.

Uses scapy for packet analysis.

2. Detects Suspicious Patterns

ICMP Ping Floods → too many ICMP Echo Requests from one source.

TCP SYN Floods / Half-Open Connections → repeated SYN packets without completion.

Port Scanning Activity → many connection attempts to multiple ports.

3. Generates Alerts

Alerts are printed/logged with details (source IP, type of suspicious behavior, packet count).

Helps analysts quickly identify reconnaissance or DoS attempts.

🧪 Test Cases

🔹 Test 1: ICMP Flood Detection

Input: PCAP with >10 ICMP Echo Requests from same source.

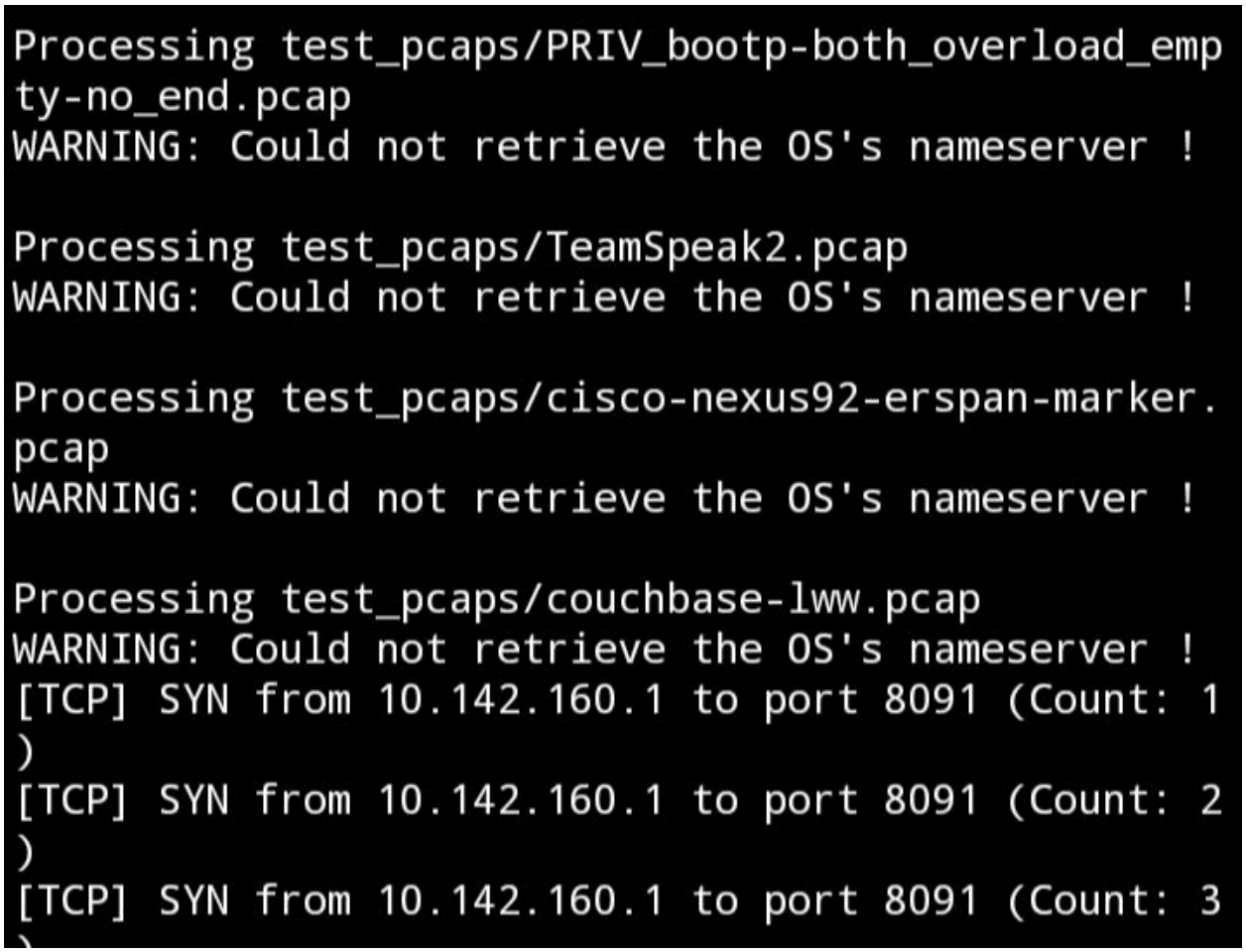
Expected Result:

```
Count: 11)
[ALERT] Possible ICMP flood from 192.168.203.132!
[ICMP] Ping from 10.10.10.109 to 192.168.203.132 (
Count: 11)
[ALERT] Possible ICMP flood from 10.10.10.109!
[ICMP] Ping from 192.168.203.132 to 10.10.10.109 (
Count: 12)
[ALERT] Possible ICMP flood from 192.168.203.132!
[ICMP] Ping from 10.10.10.109 to 192.168.203.132 (
Count: 12)
[ALERT] Possible ICMP flood from 10.10.10.109!
[ICMP] Ping from 192.168.203.132 to 10.10.10.109 (
Count: 13)
[ALERT] Possible ICMP flood from 192.168.203.132!
[ICMP] Ping from 10.10.10.109 to 192.168.203.132 (
Count: 13)
[ALERT] Possible ICMP flood from 10.10.10.109!
[ICMP] Ping from 192.168.203.132 to 10.10.10.109 (
Count: 14)
```

- Test 2: TCP SYN Scan

Input: PCAP with multiple SYN packets to different ports.

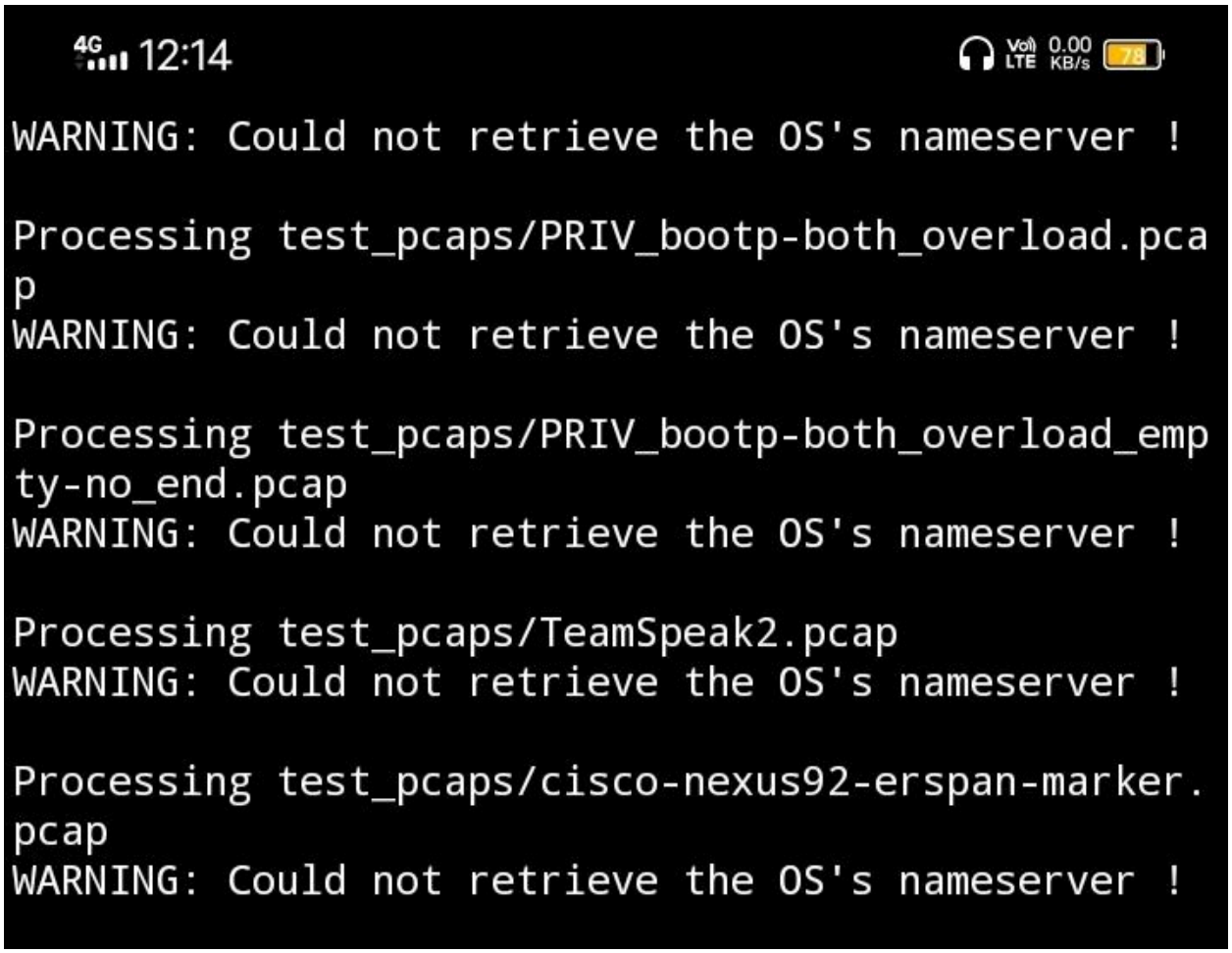
Expected Result:



Test 3: Normal Traffic

Input: PCAP with standard browsing traffic.

Expected Result:



🛡️ Why is This Useful in Cybersecurity?

✔ Attack Detection – Detects common reconnaissance activities (scans, ping sweeps) and early-stage denial-of-service attempts.

✔ Forensics – Useful for analyzing .pcap files from incident response or CTF challenges.

✔ Lightweight & Educational – A simple IDS for learning how network traffic is analyzed, how attackers operate, and how defenders detect them.

✔ Foundation for Bigger IDS – Can be extended with signature-based rules, anomaly detection, or integration into SIEM systems.

📘 Example Use Cases

Lab/CTF Environment – Analyze attack traffic from penetration testing exercises.

SOC Training – Teach interns/juniors how to identify scans/floods.

Research & Development – Extend this script into more advanced IDS/IPS systems.

Packet Analysis – Quick way to find malicious activity inside a captured .pcap.

🔮 Future Enhancements

Log alerts into JSON/CSV for SIEM integration.

Add support for detecting DNS tunneling or HTTP anomalies.

Build a web-based dashboard for real-time visualization.

Apply ML-based anomaly detection.

✅ PoC Conclusion

This PoC demonstrates that the IDS can:

Parse network traffic efficiently using Scapy.

Detect common reconnaissance techniques like ping sweeps and SYN scans.

Serve as a foundation for extending into a more advanced IDS with logging, visualization, and real-time monitoring features.