Network IDS Proof of Concept (PoC) Report

Intern_Id: 170 & 284 [Raj shedge, Siddhi Aadekar]

1. Objective

The goal of this PoC is to implement and demonstrate a lightweight Network Intrusion Detection System (NIDS) using Python and Scapy on Kali Linux. The IDS detects common malicious behaviors such as ICMP floods, TCP SYN floods, and classic scan techniques (SYN, NULL, FIN).

2. Environment Setup

The PoC was implemented on Kali Linux with a Python virtual environment.
Required packages include: scapy, pytest, tcpdump, tshark, and nmap.

Setup steps:
1. Create and activate virtual environment:
$ python3 -m venv nids-venv $
source nids-venv/bin/activate

2. Install dependencies:
$ pip install scapy pytest
$ sudo apt install -y tcpdump tshark nmap jq

3. Project files:
- mini_nids.py (IDS implementation)
- test_detectors.py (unit tests)
- run_demo.sh (automated demo)
- report_template.md (report draft)

3. Detection Logic

- ICMP: Detect echo requests/replies. Raise ICMP_FLOOD if requests exceed threshold in slidingwindow.
- TCP SYN: Track per-source SYN counts. Flag SYN_FLOOD if exceeding threshold.
- SYN Scan: If a source contacts many unique ports within a window, flag SYN_SCAN_SUSPECTED.
- NULL/FIN Scans: Detect abnormal TCP flag combinations.
- Half-Open Connections: Track incomplete TCP 3-way handshakes, flagHALF_OPEN_CONNECTION after timeout.

4. Demonstration

Two PCAPs were created to demonstrate detection:
1. normal.pcap – Benign traffic (light ping, basic connections).

2. attack.pcap – Malicious-like traffic generated with nmap (SYN, NULL, FIN scans) and pingfloods.

Commands used:
- Normal: $ sudo tcpdump -i lo -w normal.pcap 'tcp or icmp'
- Attack: $ sudo tcpdump -i lo -w attack.pcap 'tcp or icmp' &
$ sudo nmap -sS -p 1-200 localhost
$ sudo nmap -sN -p 1-200 localhost
$ sudo nmap -sF -p 1-200 localhost
$ ping -c 20 127.0.0.1

IDS Run:
$ python3 mini_nids.py --pcap normal.pcap --log-file normal_alerts.jsonl
$ python3 mini_nids.py --pcap attack.pcap --log-file attack_alerts.jsonl

5. Results

PCAP	Expected Alerts
normal.pcap	ICMP_ECHO_REQUEST, TCP_SYN
attack.pcap	ICMP_FLOOD, SYN_FLOOD_SUSPECTED, SYN_SCAN_SUSPE NULL_S

6. Observations

- The IDS successfully flagged benign vs malicious PCAPs differently. - False positives may occur under heavy legitimate load (e.g., monitoring systems, port scanners). - Thresholds should be tuned per environment to reduce noise.

7. Next Steps

- Add detection for XMAS scans, UDP scans.
- Persist alerts to a database and visualize via a dashboard.
- Integrate with real-time response mechanisms (e.g., firewall rules). - Extend testing to larger PCAPs and live networks.