Proof of Concept – Network Intrusion Prevention System (IPS)

Name: Shivani Yadav
Intern ID: 279

---

1. Objective

The aim of this Proof of Concept is to design and demonstrate a simplified Intrusion Prevention System (IPS) capable of blocking basic network-based threats.

The IPS focuses on:

- Identifying and blocking ICMP echo floods.
- Detecting TCP SYN flood attempts.
- Recognizing port scanning behavior.
- Monitoring HTTP payloads for SQL injection patterns.
- Adding one extended check for DNS query floods.

The system is not designed as a full production-grade IPS but rather to illustrate how prevention logic can be implemented for educational and small-scale scenarios.

---

2. Tools Used

- Programming Language: Python 3.x
- Library: Scapy (for reading and analyzing packets)
- Traffic Data: PCAP files (benign + malicious)
- Attack Simulation Tools:
    - ping -f → ICMP flooding
    - hping3 → SYN flood generation
    - nmap → port scanning
    - curl with crafted payload → SQL injection attempt

    - dnsperf or repeated dig queries → DNS flood testing
- Platform: Works on Linux, Windows, or Mac with Python installed

---

3. Working of the IPS

The IPS examines packet data and applies simple rules to detect suspicious activity.

3.1 Packet Analysis

- Packets are read from a PCAP file.
- Traffic is identified as ICMP, TCP, HTTP, or DNS.

3.2 Detection Rules

1. ICMP Flood Detection → blocks a source if >20 echo requests are sent.
2. SYN Flood Detection → blocks a source if >40 SYN packets are seen without completion.
3. Port Scan Detection → blocks if >15 unique ports are probed by a single IP.
4. HTTP Payload Inspection → blocks if SQL injection strings are found.
5. DNS Flood Detection → blocks if >50 DNS queries come from a single IP.

3.3 Blocking Action

- Offending IPs are added to a blocklist.
- Future packets from these IPs are automatically blocked.
- The system outputs “ALLOW” or “BLOCK” for visibility.

---

4. Testing and Validation

4.1 Setup

- Benign PCAP → Contains normal browsing, DNS, and basic web traffic.

- Malicious PCAP → Includes simulated ICMP flood, SYN flood, port scanning, SQL injection, and DNS flooding.

4.2 Expected Outcome

- Normal traffic → should not trigger blocks.
- Malicious traffic → each attack type should be detected and blocked.

4.3 Results

- Benign traffic: No false positives triggered.
- Malicious traffic:
    - ICMP flood detected and blocked.
    - SYN flood flagged and blocked.
    - Port scans correctly identified.
    - SQL injection attempts blocked.
    - DNS query flood successfully detected.

---

5. False Positives and Limitations

Possible False Positives

- ICMP monitoring tools may be mistaken as floods.
- Busy web servers may resemble SYN floods.
- Authorized scans may be treated as malicious.
- SQL keywords in genuine queries could trigger alerts.
- High DNS lookups (e.g., content delivery networks) may appear as DNS floods.

Limitations

- Static thresholds are used instead of adaptive ones.
- Encrypted HTTPS traffic cannot be analyzed.
- Only a blocklist is simulated (no live firewall integration).
- Evasion techniques like fragmented packets are not handled.

---

6. Future Improvements

- Dynamic thresholds that adjust to traffic patterns.
- Whitelisting for trusted IPs.
- Direct firewall integration (iptables/Windows Firewall).
- Extended detection for XSS, command injection, etc.
- Exporting logs in CSV/JSON for SIEM integration.
- Visualization dashboard for real-time alerts.

---

7. Conclusion

This PoC successfully demonstrates a simplified IPS that can detect and block common network threats such as ICMP floods, SYN floods, port scans, SQL injection attempts, and DNS query floods.

The IPS worked correctly on benign and malicious PCAPs, showing accurate detection without unnecessary blocking. While limited, it provides a strong base for further improvements and hands-on understanding of intrusion prevention concepts.

---

8. Practical Applications / Use Cases

- Student Learning: Helps beginners understand how IPS rules are applied.
- Lab Simulation: Can be used in small test environments to simulate attacks.
- Security Awareness: Shows how common attack patterns are detected.
- Prototype Development: Serves as a baseline for developing more advanced IPS systems.
- Training Exercises: Useful for SOC interns to practice monitoring and detection.