CSCE 5585: Advanced Network Security

Part A - Attack Vectors and Threat Intelligence: Modern Challenges and Practical Defenses

Cybersecurity Research Overview

Project By:

Group 9

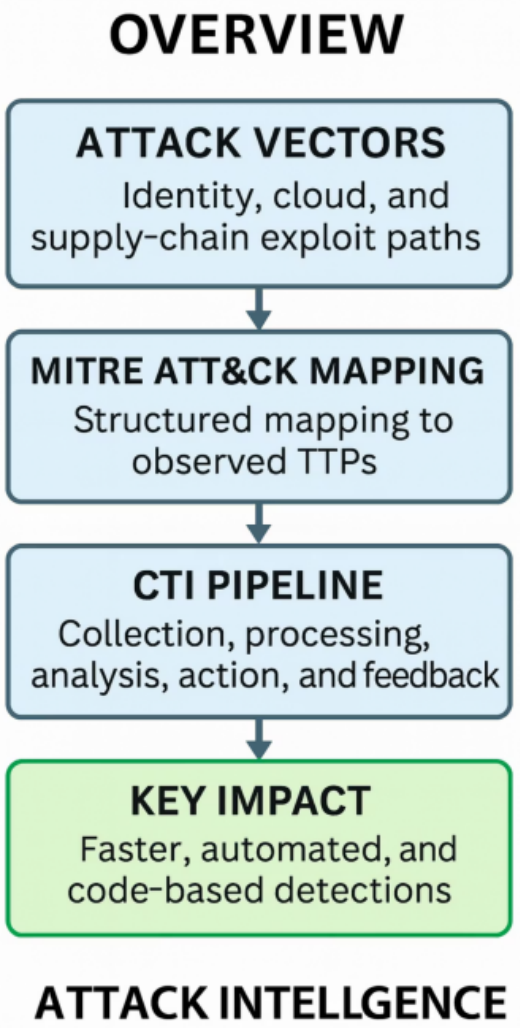
Aparna Singh | Durga Shankar Dalayi | Nikhil Ethamukkala

University of North Texas

Introduction & Problem Statement

Organizations of today are dealing with changing attack vectors which include sophisticated phishing attacks as well as cloud-native privilege abuse and compromise of IOT devices. Static defense methods have proven ineffective in defending against the dynamic nature of an ever-changing attacker.

The evolution of Cyber Threat Intelligence (CTI), is beyond the use of unstructured, ad hoc indicators. CTI now is structured and can be automated for effective mapping of adversaries' behaviors to the MITRE ATT&CK framework for the benefit of a more comprehensive defense strategy



Major Attack Vectors (2020-2025)

Phishing, BEC & Social Engineering

- Credential harvesting
- OAuth abuse
- MFA fatigue
- Deepfake voice
- QR-code phishing

Ransomware & Double/Triple Extortion

- Initial access brokers
- Lateral movement
- Data exfiltration

Supply-Chain Compromise

- Software update channels
- CI/CD credentials
- Third-party components

Cloud-Native Abuse

- Misconfiguration
- Token theft
- IMDS abuse
- Privilege escalation

Web & Application Exploits

- RCEs
- Deserialization
- SSRF
- SQLi

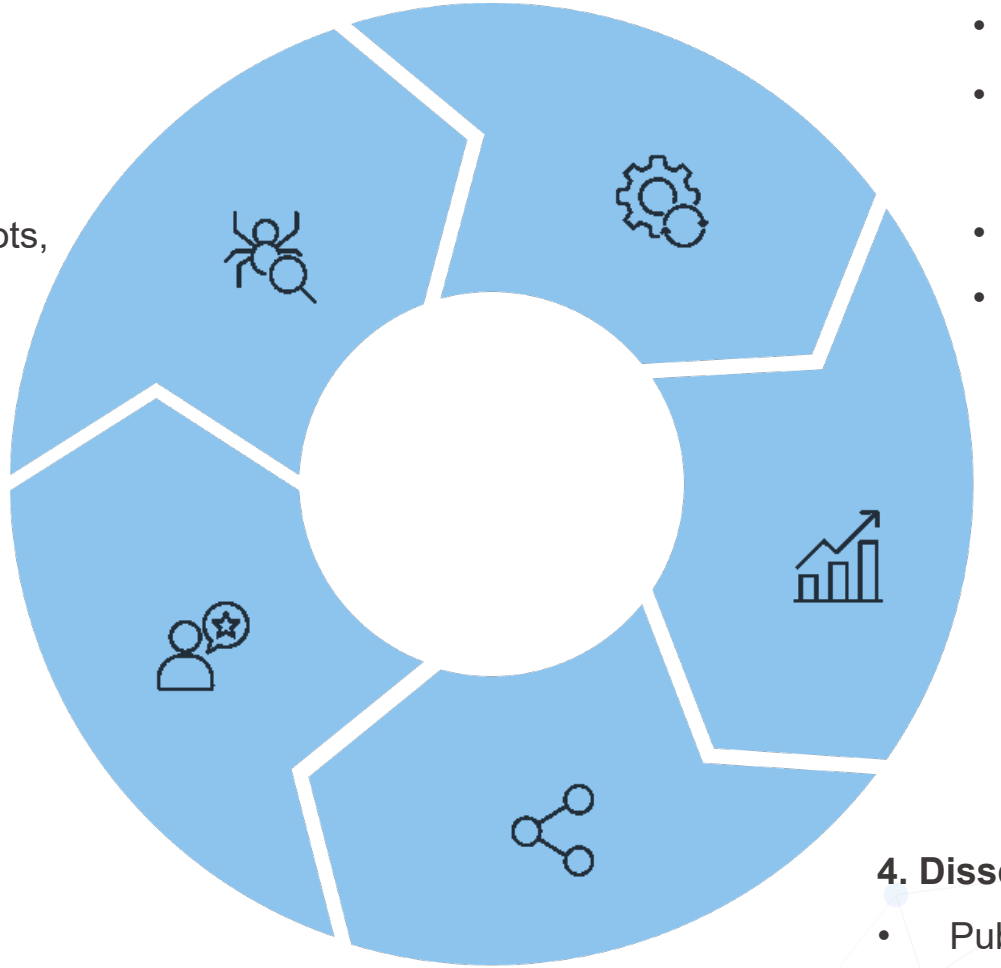
IoT/OT & Edge

- Weak authentication
- Unpatched firmware
- Flat networks

CTI Lifecycle and Pipeline

1. Requirements & Collection

- Define decision-makers and use-cases
- Gather from internal telemetry, honeypots, external OSINT, commercial feeds, ISACs/ISAOs, CISA KEV



2. Processing & Normalization

- Remove duplicates
- Enrich with WHOIS/passive DNS/ASN/geolocation
- Score indicators
- Convert to STIX objects

3. Analysis & Context

- Connect indicators to campaigns, malware families, ATT&CK techniques
- Extract TTPs from reports

4. Dissemination & Action

- Publish to TIP
- Distribute to SIEM/XDR, EDR/NGAV, SOAR, Cloud controls

5. Feedback & Measure

- Monitor SOC analyst feedback
- Track MTTD/MTTR
- Measure ATT&CK coverage

Reference Architecture

CTI-Driven Security Architecture

- **Unified Threat Intelligence:** Centralizes CTI into SIEM, EDR/XDR, SOAR, and cloud tools for ATT&CK-aligned detections.
- **Multi-Source Visibility:** Collects DNS, EDR, identity, email, cloud logs + OSINT & KEV feeds for complete attack coverage.
- **Threat Intelligence Platform (TIP):** Automates STIX/TAXII ingest, IOC scoring/decay, and maps indicators to TTPs.
- **Analytics & Automation:** SIEM correlation + UEBA identity detection + automated SOAR playbooks for fast containment.
- **Cloud & Identity Security:** CSPM/CNAPP, JIT access, workload identities, and secrets scanning reduce cloud attack surface.
- **Outcome:** Faster detection (low MTTD), rapid response (low MTTR), and reduced overall attack blast radius.

Conclusion & Future Directions

Key Findings

- Attack vectors have shifted toward identity and supply-chain leverage while maintaining social engineering as dominant entry point.
- Effective defense operationalizes CTI, converting raw indicators into mapped behaviors, detections-as-code, and repeatable response playbooks.
- Measurable outcomes anchor programs to risk and reveal investment priorities.

Future Work

- Graph-based campaign clustering and infrastructure lineage identification.
- Automated SBOM + VEX integration in build/deploy pipelines.
- Federated/privacy-preserving sharing among peers and ISACs.
- Machine Learning/LLM assistance for report triage and enrichment.
- Transition to memory-safe languages and hardware-backed isolation.

References

1. Tounsi, W., & Rais, H. (2018). *A survey on technical threat intelligence in the age of sophisticated cyber attacks*. Computers & Security, 72, 212–233. https://doi.org/10.1016/j.cose.2017.09.001
2. Wagner, T. D., Mahbub, K., Palomar, E., & Abdallah, A. E. (2019). *Cyber threat intelligence sharing: Survey and research directions*. Computers & Security, 87, Article 101589 https://doi.org/10.1016/j.cose.2019.101589
3. Verizon. (2025). 2025 *Data Breach Investigations Report.* Verizon Enterprise Solutions.

   https://www.verizon.com/business/resources/reports/dbir/
4. European Union Agency for Cybersecurity. (2023, October 19). *ENISA Threat Landscape 2023*.

   https://www.enisa.europa.eu/publications/enisa-threat-landscape-2023
5. National Institute of Standards and Technology. (2024, February 26). *The NIST Cybersecurity Framework (CSF) 2.0*. https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.29.pdf

CSCE 5585: Advanced Network Security

Part B - Secure Network Design and Implementation

Project By:

Group 9

Aparna Singh | Durga Shankar Dalayi | Nikhil Ethamukkala

University of North Texas

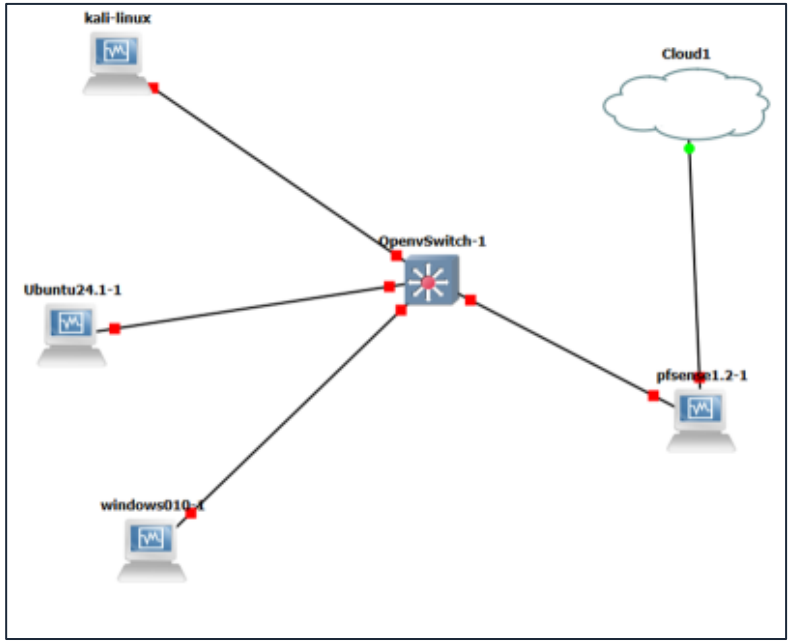
Project Overview

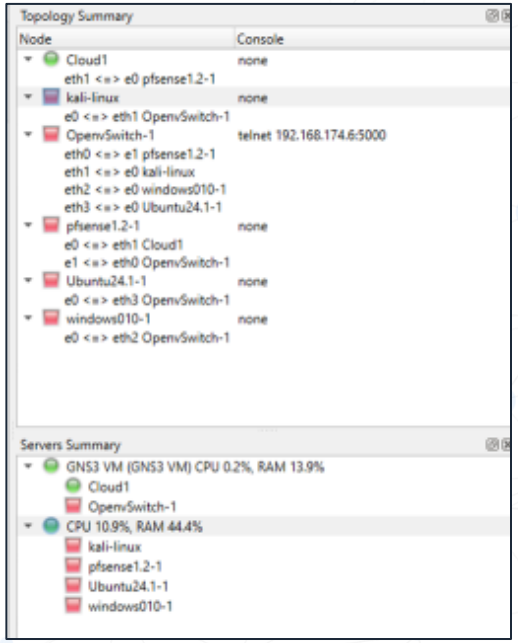
- Designed a secure, multi-segment network using pfSense with **LAN**, **DMZ**, and **WAN** separation.
- Configured **stateful firewall rules**, Hybrid NAT, and strict segmentation to control traffic between networks.
- Deployed **Suricata IDS/IPS** in inline mode on LAN & DMZ to monitor and block malicious activity.
- Implemented **OpenVPN remote-access** solution using certificate-based authentication and a dedicated tunnel network.
- Validated VPN functionality by accessing DMZ services while ensuring LAN access remained restricted.
- Performed **attack simulations** (SYN flood, ICMP flood, Slowloris, Nmap scanning, Hydra brute-force) to evaluate defensive controls.
- Verified detection & alerting via Suricata and confirmed enforcement of firewall rules and segmentation policies.

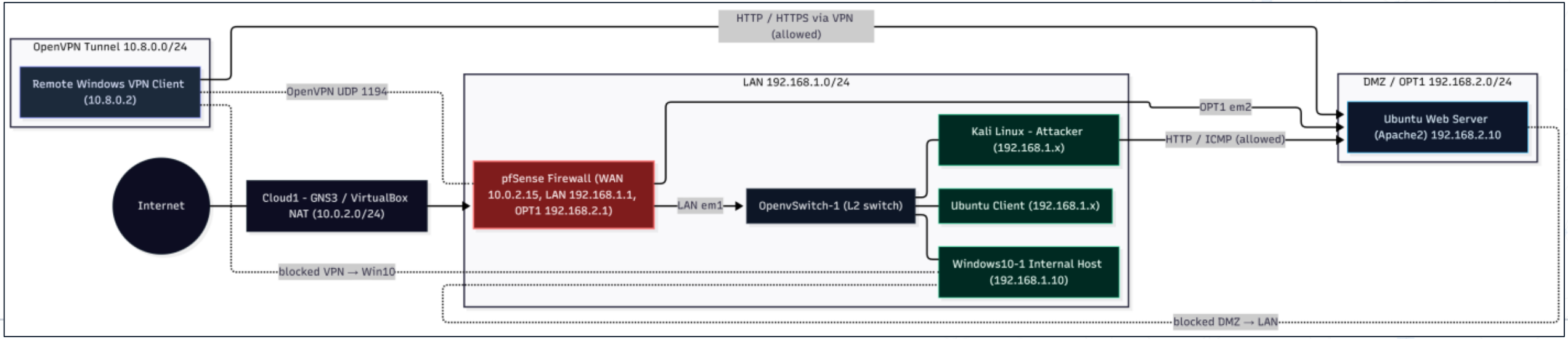
Technology Stack

Deployed Network Topology

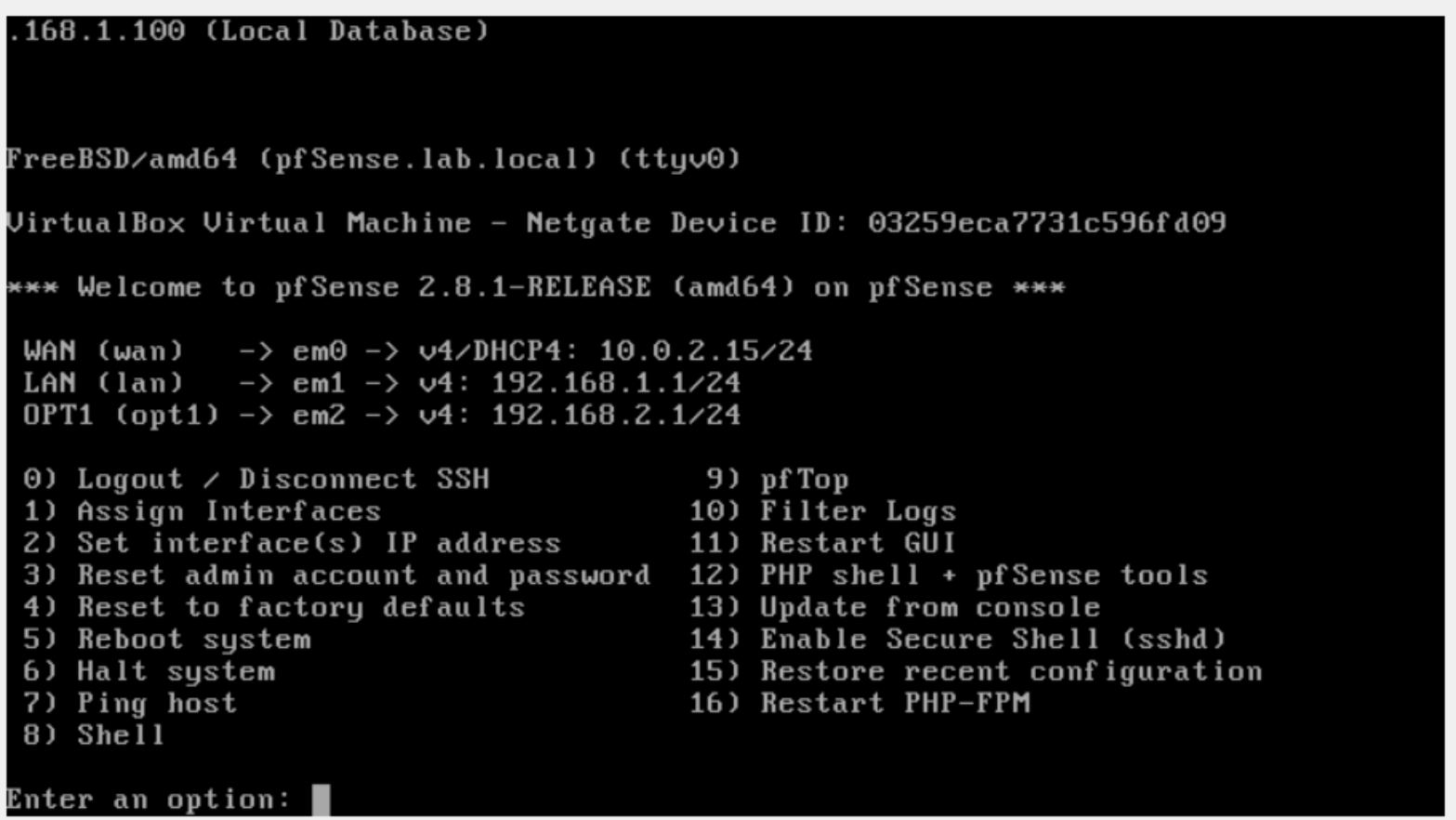
- Built a multi-segment virtual network in GNS3 consisting of **Kali**, **Ubuntu (DMZ server)**, **Windows 10**, and **pfSense**.
- All internal hosts connect through **Open vSwitch**, forming the 192.168.1.0/24 LAN.
- **pfSense** serves as the security gateway with **WAN ↔ Cloud** connectivity and **LAN/OPT1 (DMZ)** segmentation.
- The Ubuntu server is positioned in the DMZ, while Kali and Windows reside in the LAN for internal and attacker-role testing.
- This topology mirrors a real enterprise layout, supporting segmentation between LAN, DMZ, and external networks.







Firewall Configuration



- pfSense automatically detected and initialized three interfaces: **WAN (em0)**, **LAN (em1)**, and **OPT1 (em2)** during boot.
- The WAN interface was assigned a DHCP address (**10.0.2.15/24**) by the VirtualBox NAT adapter, giving the firewall external connectivity.
- We configured the **LAN** interface as **192.168.1.1/24** to serve as the default gateway for all internal hosts.
- We also configured **OPT1** as **192.168.2.1/24**, designating it as the DMZ segment in our network.
- The console view confirms successful interface assignment and establishes the base routing required for further firewall, NAT, and segmentation setup.

Firewall Rules

WAN Rules

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✕	0/0 B	*	RFC 1918 networks	*	*	*	*	*		Block private networks	
✕	0/0 B	*	Reserved Not assigned by IANA	*	*	*	*	*		Block bogon networks	
✓	0/0 B	IPv4 UDP	*	*	WAN address	1194 (OpenVPN)	*	none		Allow OpenVPN	
✓	0/0 B	IPv4 UDP	*	*	WAN address	1194 (OpenVPN)	*	none		OpenVPN OpenVPN-Server-Cert wizard	

LAN Rules

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓	3/1.30 MiB	*	*	*	LAN Address	443 80	*	*		Anti-Lockout Rule	
✓	36/22.88 MiB	IPv4 *	LAN subnets	*	*	*	*	none		Default allow LAN to any rule	
✓	0/0 B	IPv6 *	LAN subnets	*	*	*	*	none		Default allow LAN IPv6 to any rule	

OPT1 Rules

	States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
✓	0/47 KiB	IPv4 TCP	OPT1 subnets	*	This Firewall (self)	53 (DNS)	*	none		Allow DMZ DNS	
✓	0/5 KiB	IPv4 ICMP echoreq	OPT1 subnets	*	This Firewall (self)	*	*	none		Allow DMZ ping to firewall	
✓	0/1.70 MiB	IPv4 *	OPT1 subnets	*	*	*	*	none		Allow DMZ Outbound interent	
✕	0/840 B	IPv4 *	OPT1 subnets	*	LAN subnets	*	*	none		block DMZ access to LAN	
✓	0/0 B	IPv4 TCP	LAN subnets	*	192.168.2.10	80 (HTTP)	*	none		Allow LAN to DMZ Web Server HTTP	

Outbound NAT Rules

Mode

Automatic outbound NAT rule generation. (IPsec passthrough included)	Hybrid Outbound NAT rule generation. (Automatic Outbound NAT + rules below)	Manual Outbound NAT rule generation. (AON - Advanced Outbound NAT)	Disable Outbound NAT rule generation. (No Outbound NAT rules)
○	◉	○	○

Save

Mappings

	Interface	Source	Source Port	Destination	Destination Port	NAT Address	NAT Port	Static Port	Description	Actions
✓	WAN	192.168.2.0/24	*	*	*	WAN address	*		DMZ outbound NAT	

Add | Add | Delete | Toggle | Save

Automatic Rules

	Interface	Source	Source Port	Destination	Destination Port	NAT Address	NAT Port	Static Port	Description
✓	WAN	127.0.0.0/8 ::1/128 192.168.1.0/24 192.168.2.0/24 10.8.0.0/24	*	*	500	WAN address	*	✓	Auto created rule for ISAKMP
✓	WAN	127.0.0.0/8 ::1/128 192.168.1.0/24 192.168.2.0/24 10.8.0.0/24	*	*	*	WAN address	*		Auto created rule

WAN Rules:

- Blocks all unsolicited inbound traffic;
- RFC1918 & Bogon filtering enabled;
- OpenVPN 1194 allowed.

LAN Rules:

- LAN fully allowed outbound;
- Anti-Lockout on 443;
- LAN → DMZ webserver (192.168.2.10:80) permitted.

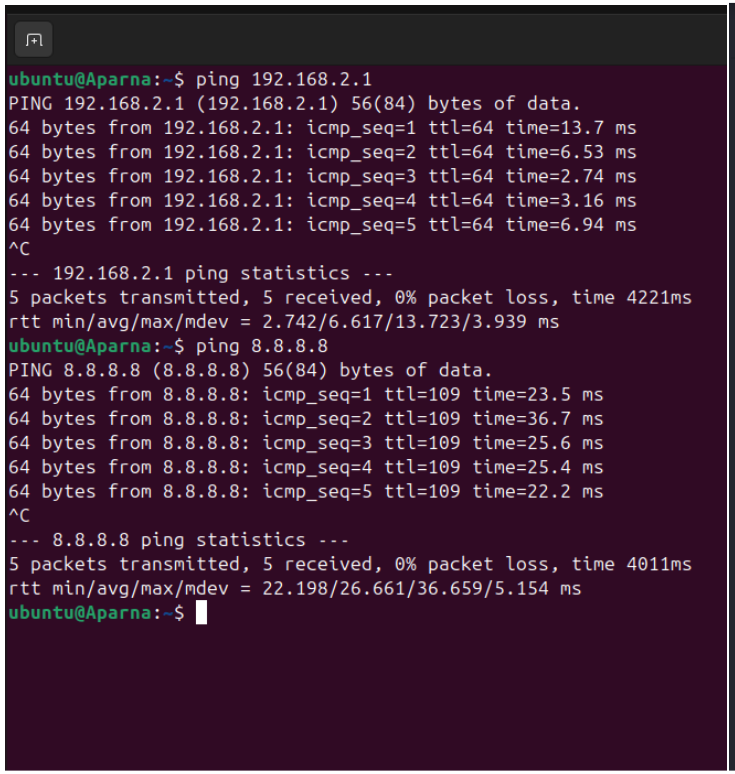
DMZ (OPT1) Rules:

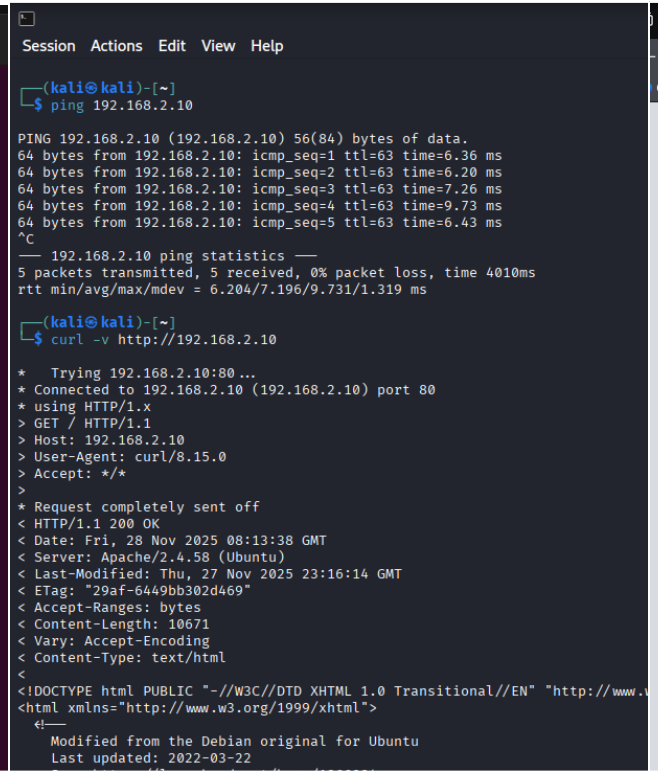
- DMZ allowed DNS & ICMP;
- outbound internet via NAT;
- DMZ → LAN traffic explicitly blocked.

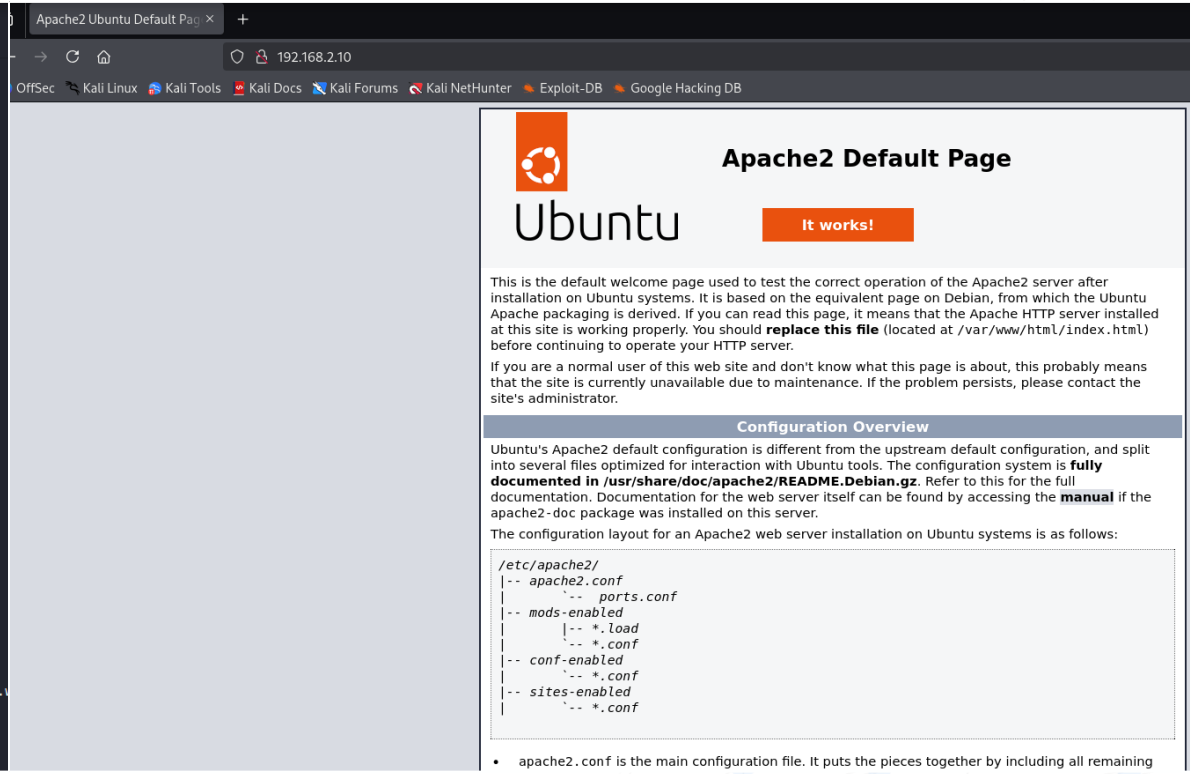
Outbound NAT:

- Hybrid NAT;
- DMZ subnet 192.168.2.0/24 translated to WAN;
- auto-rules support VPN/ISAKMP.

Setup Verification







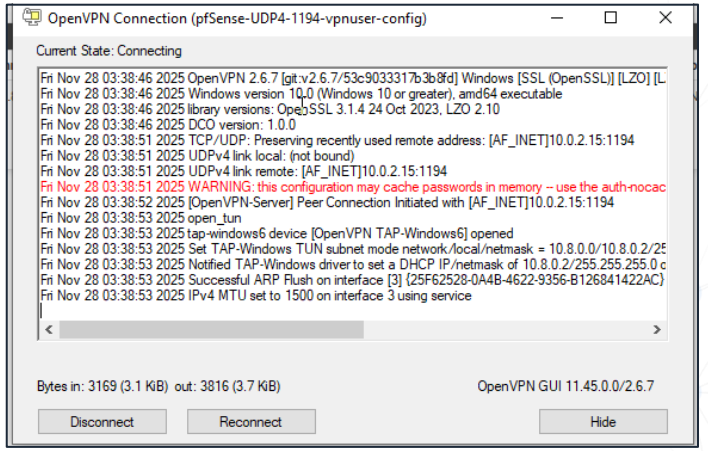
Key Checks Completed:

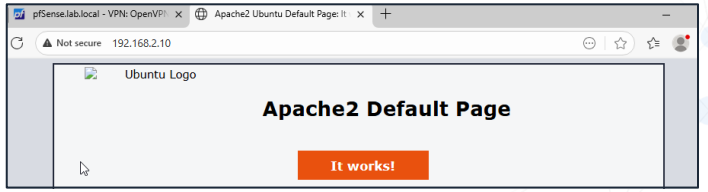
- **DMZ Gateway Reachability:**
  Successfully pinged 192.168.2.1, confirming the Ubuntu webserver is correctly connected to the DMZ network.
- **Internet Connectivity from DMZ:**
  Ping to 8.8.8.8 verified outbound NAT and internet access.
- **Kali → DMZ Host Access:**
  Kali machine pinged 192.168.2.10 and reached the Apache server over HTTP.

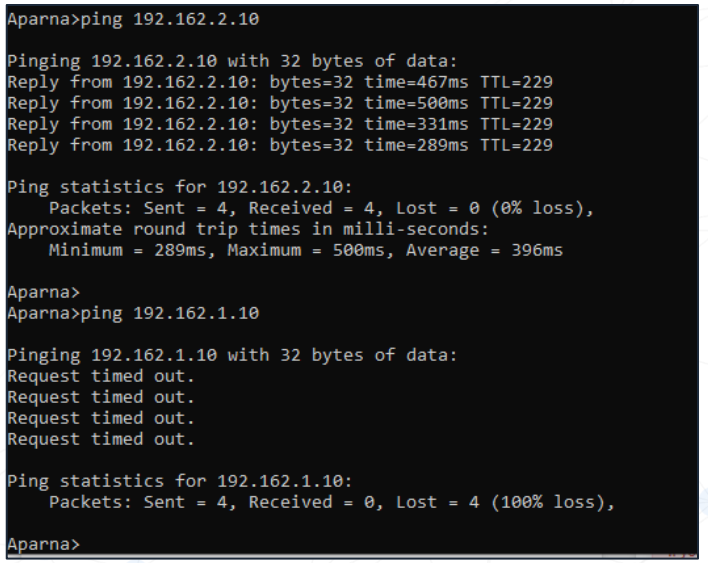
- **Webserver Functionality:**
  HTTP request to http://192.168.2.10 returned the Apache2 default page, confirming service availability.
- **Firewall Verification (Implicit):**
  Traffic flow between LAN, DMZ, and WAN confirmed that pfSense rules and NAT configurations are functioning as intended.

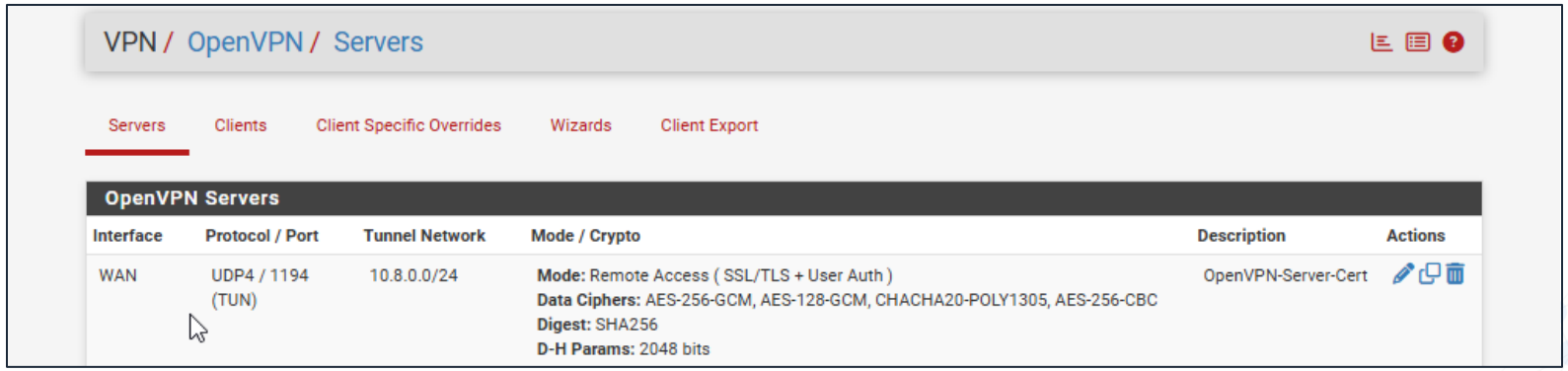
OpenVPN Configuration & Verification

- **OpenVPN server configured** on pfSense (UDP/1194, tunnel network: **10.8.0.0/24**).
- **VPNuser profile exported** via Client Export Utility → confirms certificate + config provisioning.
- **Windows client successfully connected**, OpenVPN tunnel established, TAP adapter assigned **10.8.0.2**.
- **Remote access to DMZ confirmed:**
  - VPN user **pinged 192.168.2.10** (DMZ webserver) successfully.
  - Apache webpage loaded via **VPN tunnel**, proving routing + firewall allowance.
- **LAN access blocked by design:**
  - Ping / curl to **192.168.1.10** failed → confirms segmentation & least-privilege policy.
- **Firewall rules working as intended:**
  - VPN → DMZ **allowed**
  - VPN → LAN **denied**
  - NAT & routing correctly forwarding VPN traffic.





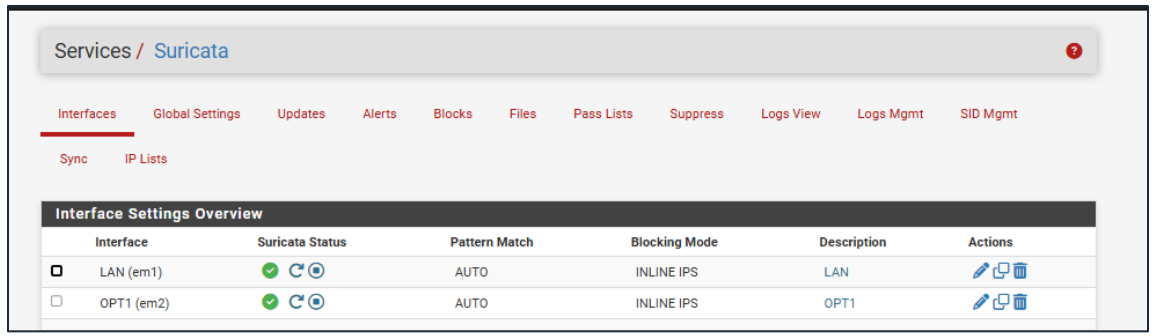


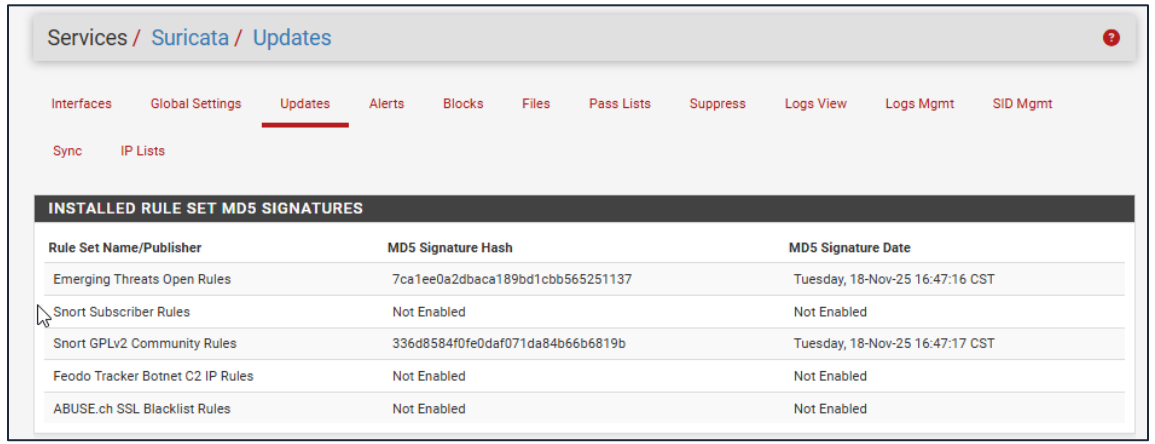


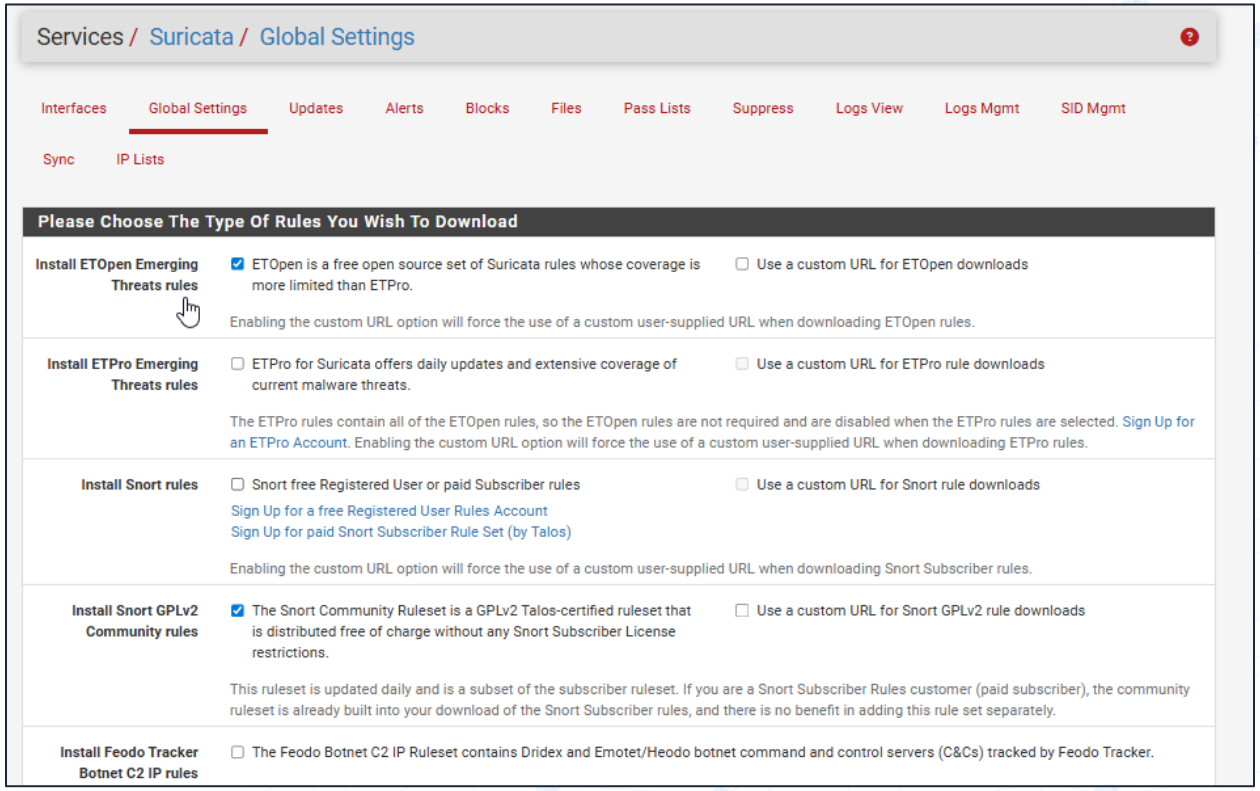
Configuring Suricata for Intrusion Detection & Prevention

Suricata IPS Deployment & Rule Activation

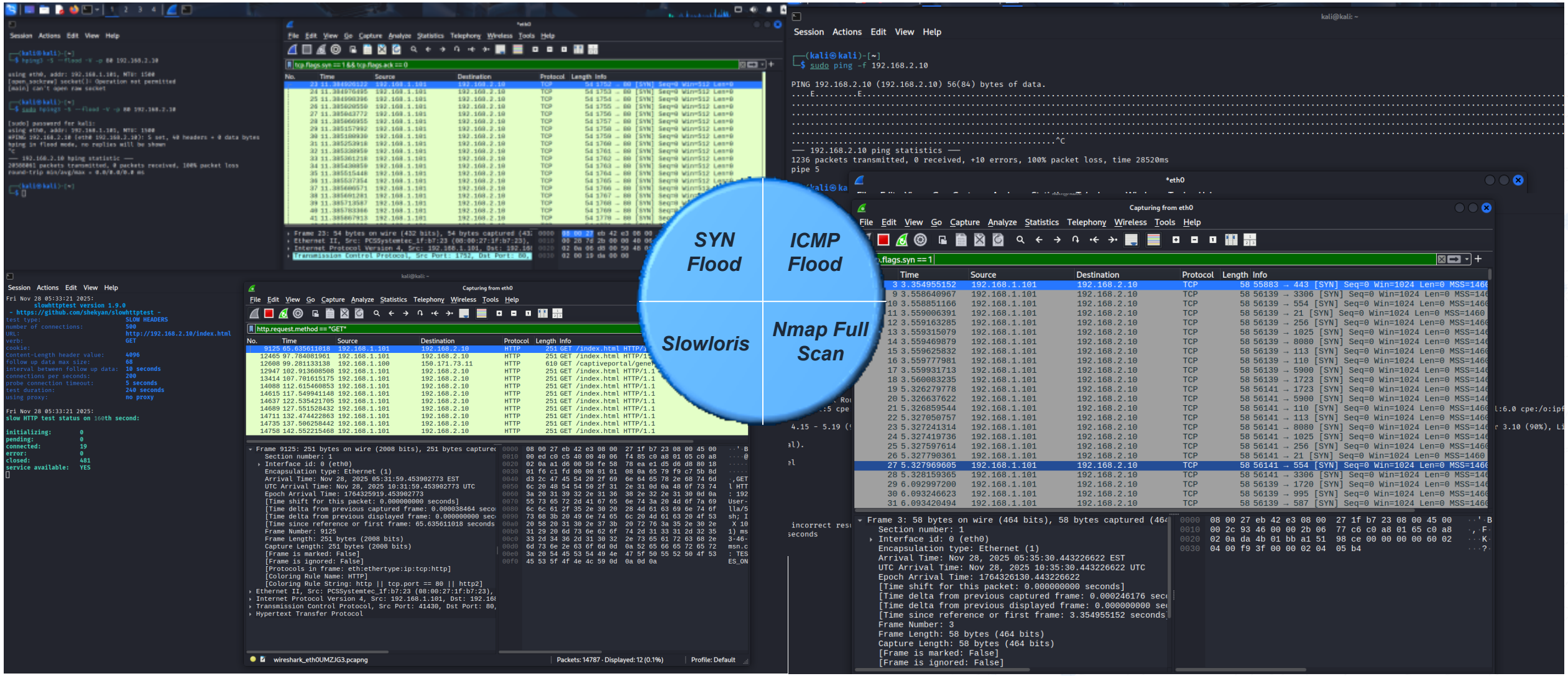
- Suricata enabled in **INLINE IPS mode on LAN and DMZ (OPT1)**
- **Loaded ETOpen Emerging Threats** rule set
- Rules updated successfully and IPS is actively blocking malicious traffic
- System ready to inspect SYN, ICMP, HTTP, Slowloris, Nmap, Hydra traffic







Attack Simulations on the Network



Suricata Detection results

- Suricata successfully detected **multiple attack types** generated from Kali.
- Alerts triggered for:
  - **SYN traffic anomalies**
  - **ICMP flood patterns**
  - **HTTP protocol irregularities (Slowloris indicators)**
- Each alert includes **source, destination, ports, protocol**, and **GID:SID** for rule mapping.
- IPS (Inline mode) inspected traffic in real time and flagged packets violating protocol behavior.
- Results confirm Suricata is actively monitoring and enforcing security at both **LAN** and **DMZ** boundaries.

Alert Log View Filter

Last 250 Alert Entries. (Most recent entries are listed first)

Note: Alerts triggered by DROP rules that resulted in dropped (blocked) packets are shown with highlighted rows below.

Date	Action	Pri	Proto	Class	Src	SPort	Dst	DPort	GID:SID	Description
11/28/2025 04:16:59	⚠	3	TCP	Generic Protocol Command Decode	192.168.2.10	80	192.168.1.101	34054	1:2260002	SURICATA Applayer Detect protocol only one direction
11/28/2025 04:16:57	⚠	3	ICMP	Generic Protocol Command Decode	192.168.2.10	0	192.168.1.101	9	1:2200025	SURICATA ICMPv4 unknown code
11/28/2025 04:16:57	⚠	3	ICMP	Generic Protocol Command Decode	192.168.1.101	8	192.168.2.10	9	1:2200025	SURICATA ICMPv4 unknown code
11/28/2025 04:16:55	⚠	3	ICMP	Generic Protocol Command Decode	192.168.2.10	0	192.168.1.101	9	1:2200025	SURICATA ICMPv4 unknown code
11/28/2025 04:16:55	⚠	3	ICMP	Generic Protocol Command Decode	192.168.1.101	8	192.168.2.10	9	1:2200025	SURICATA ICMPv4 unknown code
11/28/2025 04:10:44	⚠	3	TCP	Generic Protocol Command Decode	20.25.227.174	443	192.168.1.100	49843	1:2210038	SURICATA STREAM FIN out of window
11/28/2025 04:10:32	⚠	3	TCP	Generic Protocol Command Decode	20.25.227.174	443	192.168.1.100	49843	1:2210038	SURICATA STREAM FIN out of window

Key Findings

Network Segmentation

VPNs isolated from DMZ, LAN resources protected.

OpenVPN Validated

Secure remote access and correct IP assignment.

Firewall Rules

DMZ reachable, LAN blocked, protected ports monitored.

IPS Active

Malicious traffic inspected, protected hosts secured.

Weaklink Captures

Attack behavior detected (VPN Scan, ICMP floods, HTTP requests).

No IPS Bypasses

Malicious traffic blocked, alerts confirmed in Suricata.

Hydra Mitigation

SSH brute-force vulnerabilities blocked, credential attacks prevented.

Rule Updates

Current threat intelligence successfully loaded.

Overall Assessment: The environment demonstrates strong isolation, effective threat detection, and reliable VPN security. Firewall and IPS configurations worked cohesively to enforce policy and detect adversarial behavior.

Conclusion

- The security architecture worked as intended, enforcing clear separation between LAN, DMZ, and VPN networks.
- OpenVPN provided secure, encrypted access while maintaining strict least-privilege controls.
- Suricata effectively detected and alerted on multiple attack types, proving IPS visibility and responsiveness.
- Firewall rules successfully blocked unauthorized access attempts and allowed only permitted flows.
- The overall setup demonstrated a **defense-in-depth** approach with layered protection and validated the network's ability to withstand common attack techniques.

Challenges Faced

- Creating and isolating LAN, DMZ, and VPN networks was difficult, especially aligning routes, gateways, and access rules without breaking connectivity.
- VMs on Ubuntu and Windows repeatedly lost internet access due to NAT conflicts and pfSense interface instability, requiring multiple fixes and restarts.
- Running Suricata alongside heavy scans (Nmap), floods (ICMP/SYN), and Slowloris attacks caused significant CPU spikes on pfSense, making the firewall slow or unresponsive.
- pfSense occasionally froze, failed to apply rules correctly, or stopped routing properly, forcing full system reboots to restore functionality.

Thank You!