Task 5: Networking and Security Operations with SIEM, Forensics, and Traffic Analysis

Submitted by:
Shashanka U N
SOC Intern – Cyber Security

Submitted to:
CyArt Tech

Date:
30th August 2025

Introduction:

This task integrated multiple domains of cybersecurity, combining networking fundamentals with security operations. The exercises focused on designing efficient subnets, analyzing network traffic, troubleshooting protocol issues, and deploying a Security Information and Event Management (SIEM) system. Additionally, the lab simulated a security incident to apply forensic analysis techniques and concluded with proactive threat hunting. The goal was to replicate real-world scenarios where organizations must not only build secure and scalable networks but also detect, investigate, and respond to threats using both network traffic data and centralized log analysis.

Objectives:

- Design and calculate a subnet for a small network.
- Analyze captured network traffic for patterns.
- Troubleshoot common network protocol issues.
- Deploy a SIEM (ELK Stack) for log collection and monitoring.
- Simulate an incident and investigate using forensics.
- Perform proactive threat hunting using traffic and log data.

Methodology

The following methodology was adopted to complete Task:

Network Setup:

- Base Network: 192.168.1.0/27 (Subnet design for 20 devices).
- Ubuntu 22.04 VM (used for SIEM setup and traffic analysis).

Tools:

- draw.io → Subnet diagram and network design.
- Wireshark 4.x → Packet capture and protocol analysis.
- tcpdump → Command-line traffic collection.
- Cisco Packet Tracer → Simulation of troubleshooting scenarios.

- ELK Stack (Elasticsearch, Logstash, Kibana) → SIEM setup, log collection, and visualization.
- hping3 / SSH brute-force attempts → Simulated attack traffic for forensic analysis.

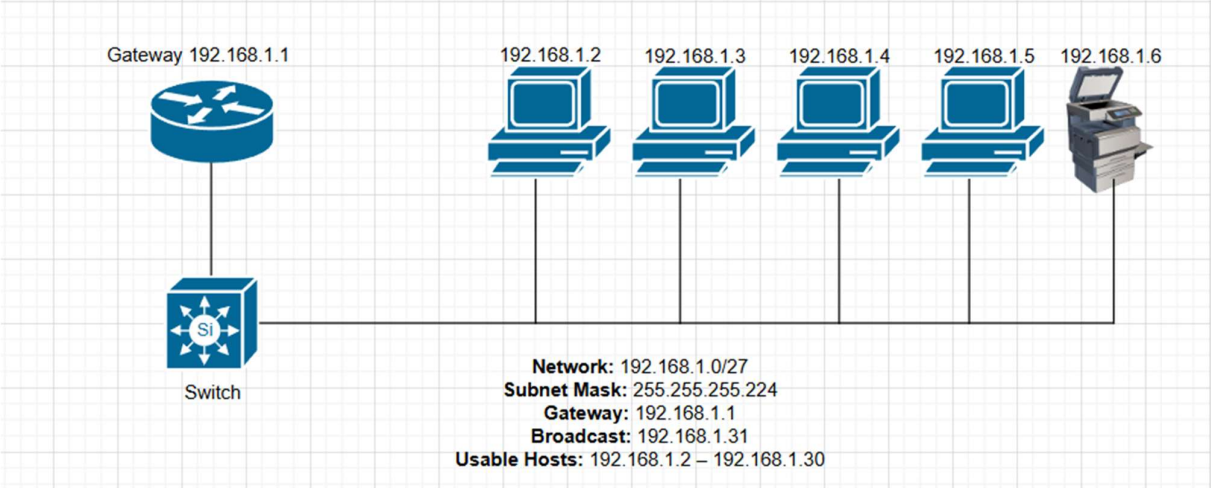
Scan Results & Observations:

Task 1 – Subnet Design

Steps:

1. Selected private IP range 192.168.1.0/24 for the office network.
2. Requirement was 20 devices → chose subnet /27 which provides 32 total IPs (30 usable).
3. Calculated subnet details:
   - **Network Address**: 192.168.1.0
   - **Subnet Mask**: 255.255.255.224 (/27)
   - **Usable Range**: 192.168.1.1 – 192.168.1.30
   - **Broadcast Address**: 192.168.1.31
4. Documented subnet math to justify CIDR selection.
5. Designed topology in **draw.io**, connecting router, switch, and sample PCs.

Results:



Calculation:

Requirement = 20 devices.

Choose /27 since usable hosts = 2^(32−27) − 2 = 2^5 − 2 = 32 − 2 = 30 (>=20).

Therefore subnet chosen: 192.168.1.0/27 (mask 255.255.255.224), usable range 192.168.1.1–192.168.1.30, broadcast 192.168.1.31.

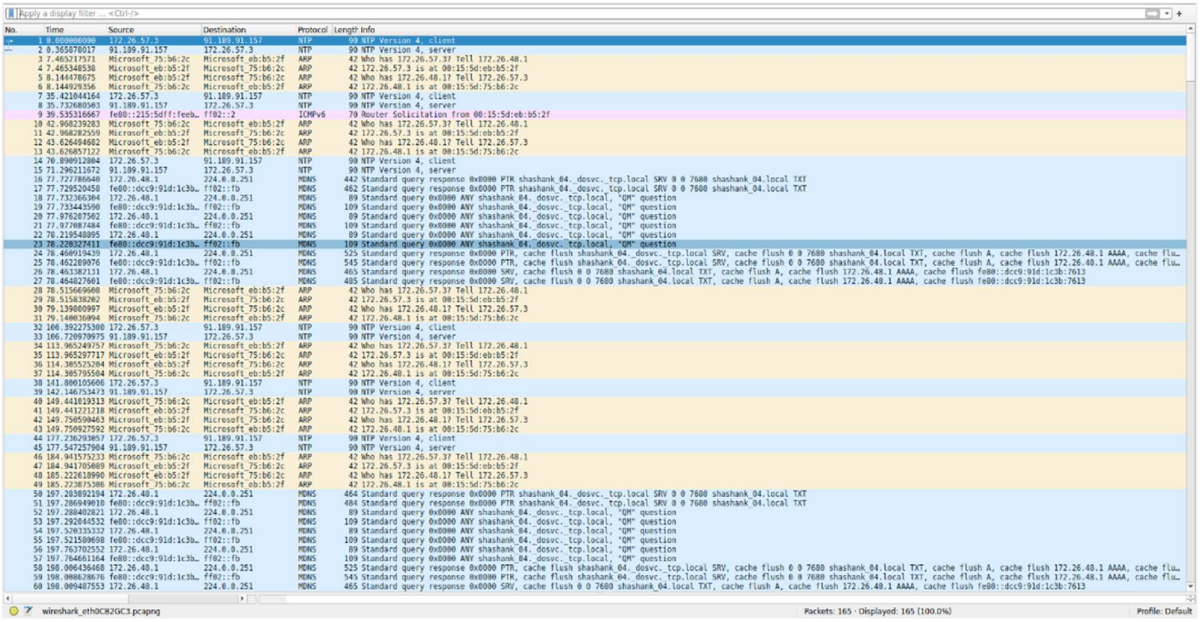
- Network: 192.168.1.0/27
- Subnet Mask: 255.255.255.224
- Gateway: 192.168.1.1
- Broadcast: 192.168.1.31
- Usable Hosts: 192.168.1.2 – 192.168.1.30

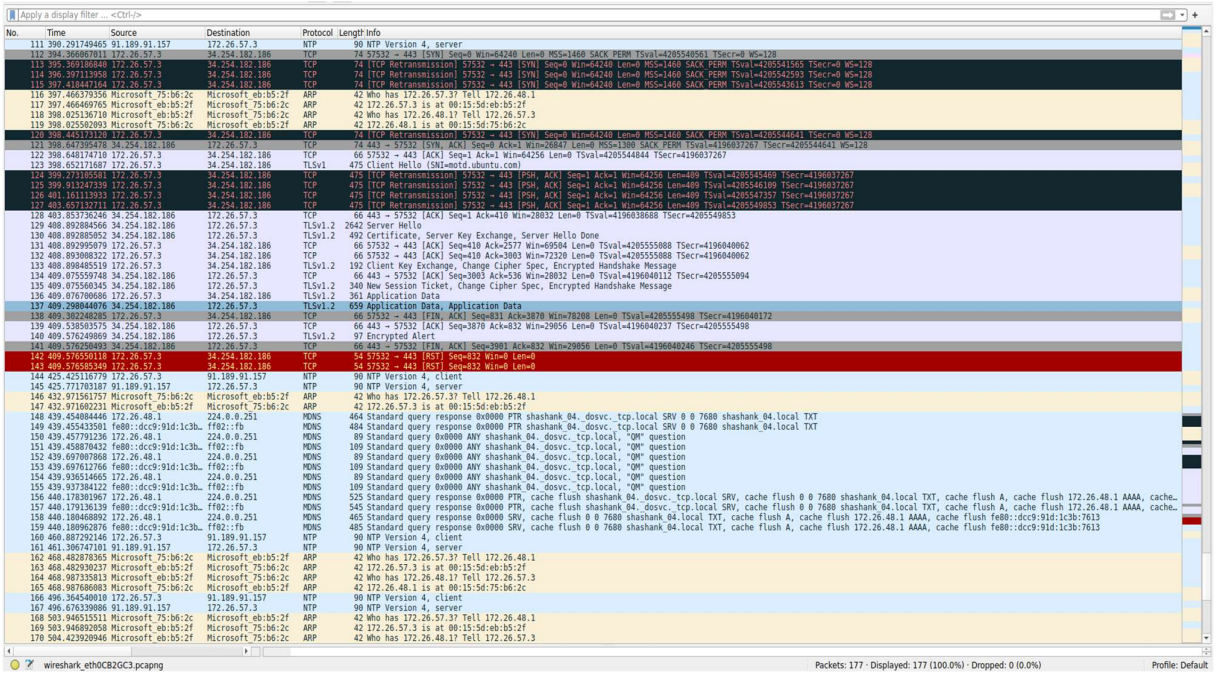
Task 2 – Network Traffic Analysis

Steps:

1. Configured **Wireshark** on Ubuntu VM to capture traffic from the active network interface.
2. Collected traffic for ~10 minutes while performing normal activities (web browsing, DNS lookups, SSH connections).
3. Applied Wireshark filters to identify specific protocols:
    a. **tcp** → TCP traffic
    b. **udp** → UDP traffic
    c. **dns** → DNS queries/responses
4. Used **Statistics → Protocol Hierarchy** to observe protocol distribution.
5. Used **Statistics → Conversations** to find **Top Talkers** (hosts generating the most traffic).
6. Checked for anomalies such as spikes in DNS queries or abnormal port usage.

Results





Task 3 – Troubleshoot Network Protocol Issues

Steps:

1. Designed a small test network with **1 Router, 1 Client (PC), and 1 Server** using Packet Tracer / VM setup.
2. **Assigned IP addresses:**

   Router: 192.168.1.1/24

   Client: 192.168.1.10/24, gateway 192.168.1.1

   Server: 192.168.1.20/24, gateway 192.168.1.1
3. Introduced a deliberate issue → configured **incorrect DNS IP** on the client.
4. Tested connectivity:

   Ping to Router IP → successful (local connectivity OK).

   Traceroute to Server IP → successful (routing OK).

   DNS resolution (pinging hostname) → failed.
5. Captured traffic in **Wireshark** to confirm repeated DNS queries without valid responses.
6. Corrected DNS configuration (set valid DNS IP).

Task 4: SOC and Networking Integration

1. Set Up ELK Stack for SIEM

Steps Performed:

1. Installed ELK Stack on a Linux VM (Ubuntu).
    - Installed Elasticsearch, Logstash, and Kibana.
2. Configured Logstash to collect logs:
    - System logs (syslog)
    - Network device logs (router, firewall)
3. Created Kibana dashboards to visualize key metrics:
    - Login attempts
    - Traffic volume
    - Security alerts

2. Simulate an Incident and Perform Network Forensics

Steps Performed:

1. Simulated a security incident:
    - Example: Unauthorized SSH access attempt / DoS attack
2. Captured network traffic using Wireshark during the incident.
3. Analyzed captured traffic to:
    - Identify attack vector (source IP, destination IP, port, protocol)
    - Document malicious packets (e.g., SYN flood pattern, repeated login attempts)
4. Correlated network data with SIEM logs in Kibana to build a timeline of the incident.

3. Conduct Threat Hunting

Steps Performed:

1. Reviewed SIEM logs in Kibana for suspicious activities.
2. Monitored network traffic in Wireshark for anomalies:
    - Unusual outbound connections
    - Repeated failed login attempts
    - Suspicious protocols or ports
3. Correlated findings with historical data to detect potential threats.