Implementation of a Network Security Mechanism

1. Assignment Overview

This group assignment aims to provide hands-on experience with Free and Open-Source Software network security mechanisms. Students will work in groups of two to deploy, configure, and evaluate a real security control (e.g., firewall, IDS, IPS, or network monitoring tool) and assess its effectiveness against representative network attacks.
A core component of the assignment is the use of an associated visualization or analysis tool to interpret security events and demonstrate detection or prevention capabilities.

2. Learning Objectives

By completing this assignment, students will be able to:

- Deploy and configure a real-world network security mechanism using free open-source tools
- Understand the operational role of firewalls, IDS, IPS, and network monitoring systems
- Simulate or generate network attacks and suspicious traffic in a controlled environment
- Analyze and interpret security alerts and logs using visualization tools
- Critically evaluate the strengths, limitations, and accuracy of security mechanisms
- Produce a professional technical report documenting setup, testing, and findings

3. Group Structure

- Group size: 2 students per group
- Both students are expected to contribute equally to design, implementation, testing, analysis, and reporting

4. Scope and Tool Selection

4.1 Security Mechanism (Choose One Category)

Each group must select one primary free open-source network security mechanism, such as:

- Firewall (packet filtering or stateful inspection)
- Intrusion Detection System (IDS)
- Intrusion Prevention System (IPS)
- Network traffic monitoring or anomaly detection tool

The selected mechanism must operate primarily at the network or transport layer (application-only security tools are not sufficient).

4.2 Visualization / Analysis Tool

Each group must integrate a visualization or analysis interface associated with the chosen security mechanism. This may include dashboards, alert viewers, log correlation tools, or traffic visualization interfaces.

The visualization component must be actively used to:

- Display alerts, detections, or blocked traffic
- Support analysis of attack behavior and system response

5. Experimental Environment and Deployment

Students may choose any suitable deployment environment, provided that it is clearly documented and justified. Acceptable environments include:

- Virtual machines (e.g., VirtualBox, VMware)
- Docker or other container-based platforms
- Local lab or testbed networks

The environment must include:

- At least one protected system or service
- At least one traffic or attack source
- A clearly defined network topology

5.1 Recommended Dockerized Deployment (Resource-Constrained Environments)

Students are strongly encouraged to use a containerized (Docker-based) deployment, particularly when physical or virtual machine resources are limited.

A Dockerized setup allows the security mechanism, traffic generator (attacker), protected service, and visualization or logging components to be deployed as multiple isolated containers interconnected through one or more Docker networks. This approach offers several advantages:

- Reduced hardware and memory requirements compared to full virtual machines
- Clear separation of roles (e.g., attacker, sensor, dashboard, target)
- Easier setup, teardown, and reconfiguration of experiments
- Improved reproducibility and portability across different student systems

Typical Dockerized components may include:

- One container running the security mechanism (e.g., firewall, IDS/IPS)
- One or more containers generating benign and malicious traffic
- One container hosting the visualization or analysis tool

Students adopting this approach must:

- Clearly document the container architecture and network topology
- Explain the role of each container
- Provide deployment artifacts (e.g., Docker Compose files) or equivalent configuration documentation

Use of Docker is recommended but not mandatory. Non-containerized deployments are acceptable provided the setup is well documented and justified.

6. Attack and Testing Requirements

Each group must test the selected security mechanism against at least three distinct attack or threat scenarios, such as:

- Network scanning and service enumeration
- Denial-of-Service–style traffic
- Suspicious or malformed packets
- Unauthorized access attempts
- Policy violations or abnormal traffic patterns

For each scenario, students must:

- Describe the attack objective
- Explain how the attack traffic was generated
- Demonstrate how the security mechanism detects, logs, or blocks the activity
- Provide evidence using logs and visualization dashboards

7. Deliverables

7.1 Written Technical Report

A brief yet structured report including:

1. Introduction and objectives
2. Selected security mechanism and justification
3. System architecture and deployment setup
4. Configuration details
5. Attack scenarios and testing methodology
6. Visualization and alert analysis
7. Results and observations
8. Limitations and evasion considerations
9. Conclusion and lessons learned

Relevant screenshots of dashboards, alerts, logs, and configurations must be included.

7.2 Presentation / Demonstration

- Short presentation summarizing the implementation and findings
- Optional live or recorded demonstration of detections or alerts