Naval Innovathon 2025: Challenge Brief on a Unified IPsec Solution

Executive Summary

This document outlines the "SWAVLAMBAN 2025 HACKATHON CHALLENGE – 2," a component of the Naval Innovathon 2025. The central challenge is to design and develop a unified, cross-platform IPsec solution to secure data-in-transit across diverse enterprise and government networks. The core problem addressed is the operational complexity, deployment delays, and security misconfigurations arising from inconsistent IPsec management across operating systems like Windows, various Linux distributions (including BOSS OS), and MacOS.The objective is to create a single solution that can be deployed on any network node, automatically encrypting traffic based on centrally defined policies with minimal manual intervention after an initial administrator setup. Key technical requirements include support for all IPsec operational modes, flexible cryptographic controls, persistent and automated operation (including auto-start and auto-reconnection), and seamless interoperability. Evaluation will prioritize unified configuration, automatic encryption, robust cross-platform support, multi-tunnel capacity, and comprehensive monitoring. Submissions must include a fully functional solution, source code, technical documentation, and a five-minute demonstration video, which will be the basis for the first round of evaluation.

1. Challenge Overview and Problem Statement

The hackathon challenge addresses a critical security issue prevalent in modern enterprise and government networks: the lack of a standardized approach to IPsec configuration and management. These networks are characterized by a heterogeneous environment of devices running diverse operating systems, including Windows, Linux (Debian/Ubuntu, RHEL, BOSS, Fedora), and MacOS.While each operating system provides inherent IPsec support, the methods for configuration, management tools, and APIs vary significantly. This inconsistency creates substantial operational friction, leading to:

- **Deployment Delays:** Time-consuming processes to configure and validate IPsec on different platforms.
- **User Overheads:** Increased burden on users and administrators to manage multiple systems.
- **Operational Complexity:** Difficulty in maintaining a consistent security posture across the network.
- **Security Misconfigurations:** A heightened risk of errors that could compromise network security.The challenge's goal is to engineer a unified IPsec solution that overcomes these issues. The proposed solution should be deployable on every network node, regardless of the operating system. Once enabled by an administrator, it must automatically protect network traffic according to centrally defined security policies, ensuring persistent, automated, and interoperable secure communication between all devices.

2. Core Technical Objectives

The solution must meet a comprehensive set of technical requirements to be considered successful. These objectives focus on cross-platform uniformity, functional completeness, and robust automation.

2.1. Unified Cross-Platform Functionality

- **Platform Support:** The solution must be capable of automatically configuring and managing encrypted network traffic on Windows, Linux (Debian/Ubuntu, RHEL, BOSS, Fedora), and MacOS.
- **Uniform Operation:** It must operate uniformly across all devices in the network, ensuring consistent policy application.
- **Deployment Model:** The solution is intended for a one-time deployment by an administrator on each device.
- **Automation:** It must enable itself automatically at system start-up without requiring any manual user intervention.

2.2. Comprehensive IPsec Support

- **Operational Modes:** The solution must fully support all IPsec operational modes, allowing an administrator to select the most suitable option for the organization's security model. Supported modes include:
- ESP Tunnel mode
- ESP Transport mode
- AH Tunnel mode
- AH Transport mode
- Combined ESP + AH

2.3. Flexible Traffic Management

- **Selective Encryption:** The solution must support policies that allow an administrator to choose between encrypting all outgoing traffic or encrypting traffic for specific subnets or destinations.
- **Policy Exceptions:** It must be possible to create exceptions for certain traffic types, such as ICMP, to accommodate network configuration requirements.
- **Tunneling Models:** The solution must support both site-to-site and host-to-host encrypted tunnels and allow for seamless routing of encrypted traffic.

2.4. Performance and Cryptographic Control

- **Latency:** The implementation of packet encryption should not affect the latency of network traffic to the extent feasible.
- **Flexible Controls:** A network administrator must have the ability to configure and enforce the following parameters uniformly across all devices:
- **Encryption Algorithms:** AES-128, AES-256, or other supported options.
- **Integrity Algorithms:** SHA1, SHA-256, SHA-384, etc.
- **Key Exchange Parameters:** IKEv1/IKEv2, DH group selection, etc.

- **Authentication Methods:** Pre-Shared Key (PSK) or certificate-based.

2.5. Automation and Interoperability

- **Persistence:** Following the initial setup, the client must auto-start on system boot, automatically re-establish tunnels if disconnected, and securely persist configurations without requiring manual re-entry.
- **SA Management:** The solution must automate Security Association (SA) negotiation, renewal, and tear-down.
- **Firewall & Routing:** It should automatically manage necessary firewall and routing requirements, such as opening UDP ports 500/4500 and allowing ESP (protocol 50) and AH (protocol 51) traffic.
- **Monitoring:** The system must provide status checks, robust error handling, and detailed logs to give administrators full visibility into its operation.

3. Evaluation and Scoring Criteria

Submissions will be evaluated based on their effectiveness in meeting the core objectives. The key aspects for scoring are detailed below.| Category | Description || ------ | ------ || **Unified Configuration** | A centralized or consistent per-device configuration approach that applies the same policies and parameter structure across all supported operating systems. || **Automatic Encryption** | The ability for all outgoing network traffic from a device to be automatically encrypted using the selected IPsec mode and policies. || **Cross-Platform Support** | Demonstrated functionality across Windows, Linux, MacOS, and specifically BOSS OS, with uniform security parameters. || **Multi-Tunnel Capacity** | Support for establishing and maintaining simultaneous multiple encrypted tunnels to different peers, subnets, or workloads. || **Auto-Start and Recovery** | Automatic activation of the solution at system boot and the ability to automatically reconnect dropped tunnels without intervention. || **Monitoring and Logs** | The maintenance of detailed logs covering tunnel status, uptime, rekey events, and traffic summaries. || **Error Handling** | The capability to detect and report configuration errors, key mismatches, or peer unavailability. || **Network Interoperability** | The solution must allow for seamless operation of network-dependent devices such as printers and scanners post-implementation. |

4. Desirable Bonus Features

While not mandatory, solutions that incorporate the following features will be viewed favorably:

- **Tunnel Visualization:** A graphical interface representing active tunnels and their endpoints.
- **Real-Time Traffic Monitoring:** Dashboards or tools displaying statistics like bandwidth usage, latency, packet loss, and encryption performance.
- **Event Logs and Alerts:** A system for generating notifications related to tunnel downtime, security warnings, or configuration issues.
- **Configuration Dashboard:** An administrative interface for adding or removing tunnels, viewing SA details, and managing keys.

5. Final Deliverables and Submission Guidelines

Participants are required to submit a complete package of deliverables to be considered for evaluation. Partial submissions are permitted, provided that reasons for non-compliance are clearly indicated.

1. **Fully Functional IPsec Solution:** Complete, packaged, and ready-to-deploy solution for Windows, Linux, MacOS, and BOSS OS.
2. **Source Code:** The complete source code with a clear directory structure, comments, and coding standards, including build scripts for each supported OS and a README file.
3. **Technical Documentation (2-3 pages):** A document covering the technical architecture, code description, and an installation/user manual with dependencies. To be submitted as part of the Product Description Document.
4. **Test and Validation Document:** A report detailing the test environment, test scenarios, and results (including logs, screenshots, evidence of encrypted traffic, and performance/latency analysis comparing encrypted vs. clear traffic).
5. **Demonstration Video (5 minutes):** A video showcasing installation, configuration, client start-up, tunnel and SA verification, multi-OS encrypted traffic, a multi-tunnel scenario, auto-start functionality, and logs. **The first round of evaluation will be based entirely on this video.**
6. **Presentation (8-10 slides):** A short presentation summarizing the problem statement, solution approach, architecture, key features, demonstration summary, challenges faced, and potential future enhancements.All code and relevant documents (excluding the Technical Documentation) are to be submitted to a private repository on GitHub Classroom. Participants must submit their GitHub username and repository link via the provided portal: https://classroom.github.com/a/YvbnXGej.