PS1: AI-Driven Dynamic Cybersecurity Shield for IoT Networks

Objective

Participants will design a **real-time cybersecurity shield for IoT devices**, which dynamically detects, mitigates, and adapts to cyber threats such as **malware injections, botnet attacks, and side-channel exploits**.
This challenge requires teams to:

- Simulate an **IoT ecosystem** with multiple devices.
- Implement a **real-time network intrusion detection system (NIDS)** with AI-based anomaly detection.
- **Emulate cyberattacks** and design **self-healing mechanisms** to defend against them.
- Deploy a **secure communication protocol** that balances **latency vs. security** for IoT networks.

All implementations will be simulated in a **software-defined IoT environment**, ensuring feasibility for an online hackathon.

Problem Breakdown (Four Subparts)

Hardware Part (Simulation-Based)

1. IoT Network Simulation & Attack Emulation

🔹 *Problem Statement:*
IoT networks are vulnerable to various cyber threats due to **limited processing power** and **inconsistent security updates**. Your task is to **simulate an IoT environment** and generate **realistic cyberattack scenarios**.

🔹 *Tasks:*

1. **IoT Device Emulation:**
    - Use tools like **IoTIFY, Node-RED, Cooja (for Contiki OS), or NS-3** to create a virtual **network of IoT devices** (e.g., smart cameras, sensors, smart locks).

    - Assign different constraints to each device, such as **low RAM, limited CPU power, or intermittent connectivity**.
2. **Cyberattack Simulation:**
    - Emulate **Denial-of-Service (DoS) attacks, ARP Spoofing, DNS Poisoning, and Man-in-the-Middle (MITM) attacks** using Kali Linux, Scapy, or custom scripts.
    - Implement **side-channel attacks** that extract sensitive data based on power consumption patterns.
3. **Data Logging & Capture:**
    - Capture **network packets and system logs** to analyze intrusion patterns.
    - Use Wireshark or custom log parsers to extract critical attack signatures.

---

2. Secure & Adaptive Edge Processing for IoT Security

🔹 *Problem Statement:*
Traditional **cloud-based security solutions** for IoT networks introduce **high latency and bottlenecks**. Your task is to **implement an edge-based intrusion detection system** (IDS) that detects and responds to attacks locally.

🔹 *Tasks:*

1. **Edge-Based IDS Deployment:**
    - Set up an **AI-powered anomaly detection system** on an emulated IoT gateway (using a Raspberry Pi VM, Docker containers, or a virtualized Linux edge node).
    - Implement **lightweight attack detection models** using TinyML or ONNX models for IoT scalability.
2. **Real-Time Threat Mitigation:**
    - Develop an **automated response mechanism** to isolate compromised devices.
    - Implement a **quarantine system** that cuts off infected devices from the network without affecting normal operations.
3. **Secure Communication Protocol:**
    - Design a **low-latency, encrypted protocol** for communication between IoT devices.
    - Compare performance trade-offs between **AES, ECC, and Post-Quantum Cryptography (PQC)** for real-time security.

Software Part (AI-Based Processing & Detection)

3. AI-Powered Intrusion Detection & Threat Prediction

🔹 *Problem Statement:*
Cyberattacks often exploit **previously unseen vulnerabilities**, making traditional rule-based security solutions ineffective. Your task is to **build an AI-driven intrusion detection system (IDS)** that can dynamically learn and predict emerging threats.

🔹 *Tasks:*

1. **Dataset Creation & Augmentation:**
   - Use datasets like **CICIDS2017, TON_IoT, or generate synthetic intrusion data** using attack simulations.
   - Apply **data augmentation techniques** to simulate real-world network conditions (packet loss, delay, jitter).
2. **AI-Based Threat Detection Model:**
   - Train a model using **LSTMs, Transformers, or Graph Neural Networks (GNNs)** for real-time intrusion detection.
   - Implement an **online learning approach**, allowing the IDS to **adapt to new attacks over time**.
3. **Feature Engineering & Explainability:**
   - Extract network features like **packet entropy, request frequency, and unusual device behavior**.
   - Implement **SHAP (SHapley Additive exPlanations) or LIME** to explain why an attack was detected.
4. **Performance Evaluation:**
   - Compare **false positive vs. false negative rates** for different attack types.
   - Optimize the model to **run efficiently on edge devices** with minimal power consumption.

---

4. AI-Powered Self-Healing Security Framework

🔹 *Problem Statement:*
An effective security system should **not just detect attacks** but also **autonomously recover and strengthen itself** over time. Your task is to **design a self-healing security mechanism** that automatically **reconfigures** the network after an intrusion.

🔹 *Tasks:*

1. **Automated Incident Response System:**
    - Develop an **AI-driven risk scoring system** that classifies threats based on impact severity.
    - Implement an **automated response mechanism** (e.g., rate limiting, dynamic firewall updates, device lockdowns).
2. **Attack-Adaptive Security Policies:**
    - Implement **Reinforcement Learning-based policy updates** that fine-tune firewall rules and access controls based on attack history.
    - Use Bayesian Optimization or Genetic Algorithms to **automatically adjust network parameters** for enhanced security.
3. **Recovery & Self-Healing Mechanism:**
    - Design a **fault-tolerant framework** where compromised devices can be securely restored.
    - Implement **secure firmware rollback or containerized OS recovery mechanisms**.
4. **Scenario-Based Testing:**
    - Simulate various attack scenarios (e.g., IoT botnet infection, ransomware propagation).
    - Measure **system resilience, recovery time, and adaptability metrics**.

Deliverables

✅ **Simulation Demonstration:** A video or live demo showcasing a **real-time cyberattack and defense response** in the simulated IoT network.
✅ **Technical Report:** A **detailed methodology** covering network setup, AI models, intrusion detection results, and security policies.
✅ **Code Repository:** A GitHub or cloud-based repository containing **attack emulation scripts, AI models, and secure communication protocols**.

Expected Outcome

By the end of the hackathon, participants should have developed a **fully functional, AI-driven cybersecurity shield** that:

- **Detects and predicts** cyber threats in real-time.

- **Responds autonomously** to neutralize threats without human intervention.
- **Optimizes network security** for IoT devices while balancing latency vs. protection.
