Project Title: Enhancing WLAN Security Using Artificial Intelligence

I. Project Description

The project explores the integration of Artificial Intelligence (AI) to improve the security of Wireless Local Area Networks (WLANs). Wireless networks are susceptible to a wide range of attacks, such as eavesdropping, unauthorized access, denial of service (DoS), and man-in-the-middle (MITM) attacks. Traditional security mechanisms often rely on static configurations and pre-defined rules, which may fail against sophisticated and evolving threats. AI-powered systems offer the ability to dynamically detect, prevent, and respond to these threats by analyzing network traffic, user behavior, and attack patterns in real-time.

This project focuses on designing, implementing, and testing AI models for detecting and mitigating WLAN security threats. The solution will involve integrating AI with traditional security frameworks to create a more robust and adaptive WLAN security infrastructure.

II. Objectives

1. **Threat Detection:**
   Use AI models to identify abnormal patterns in WLAN traffic that indicate potential attacks, such as ARP spoofing, rogue access points, and DDoS attacks.
2. **User Behavior Analysis:**
   Employ machine learning algorithms to profile legitimate user behavior and identify anomalies caused by unauthorized access or compromised devices.
3. **Attack Mitigation:**
   Implement AI-driven responses to dynamically mitigate threats, such as isolating rogue devices or altering WLAN configurations in real-time.
4. **Evaluation and Testing:**
   Assess the effectiveness of the AI system against common WLAN attacks and compare its performance to traditional security mechanisms.

III. Scope of Work

1. **Model Development:**
   - Implement machine learning models (e.g., Random Forest, SVM, or Deep Learning models like LSTMs) to classify normal and malicious traffic.
   - Use unsupervised learning techniques (e.g., clustering or autoencoders) for anomaly detection.
   - Dataset: iot23-dataset, awid dataset, ….
2. **Integration with WLAN Security:**
   - Develop scripts or tools to interface the AI system with WLAN components (e.g., access points, firewalls, or RADIUS servers).

   - Automate responses, such as disabling rogue APs or blocking suspicious devices.
3. **Testing and Validation:**
   - Deploy the system in a simulated network environment with multiple VMs and real devices.
   - Evaluate detection accuracy, false positives, and response time.
4. **Reporting and Documentation:**
   - Document the project's architecture, implementation, results, and future recommendations.

IV. Proposed System Architecture

1. **Data Layer:**
   - Wireless traffic is captured using sniffers (e.g., Wireshark) and stored in a centralized database for analysis.
2. **AI Processing Layer:**
   - A machine learning engine processes the traffic data to identify threats.
   - Models for supervised learning (e.g., classification) and unsupervised learning (e.g., anomaly detection) are trained and deployed.
3. **Response Layer:**
   - Based on AI predictions, automated scripts update WLAN configurations or send alerts to administrators.
4. **User Interface:**
   - A dashboard visualizes detected threats, network performance, and security status.

V. Technologies and Tools

1. **WLAN Setup:**
   - Wireless Access Points (e.g., Cisco, Ubiquiti, or open-source alternatives like hostapd).
   - WPA2/WPA3 security protocols for testing.

2. **AI Development:**
   - **Programming Languages:** Python (with libraries like Scikit-learn, TensorFlow, or PyTorch).
   - **Tools:** Wireshark, Tcpdump, and NetfilterQueue for traffic analysis and interception.

3. **Simulation Tools:**
   - Kali Linux for attack simulation (e.g., WPA2 cracking, MITM).
   - VMware/VirtualBox for creating test environments with virtual machines.

4. **Data Visualization:**
   - Tools like Grafana or Matplotlib to present insights from the AI models.

VI. Real-World Test Scenario for the Developed WLAN Security Solution

You are tasked with testing the AI-based WLAN security system in a simulated corporate wireless network. The environment includes multiple users, devices, and access points, replicating real-world conditions. The system will be evaluated based on its ability to detect, mitigate, and respond to various WLAN security threats while ensuring minimal disruption to legitimate users.

Environment Setup

1. **Network Topology**:
   - A simulated corporate WLAN with:
     - Two Access Points (APs): One legitimate and one rogue (for testing).
     - Devices: At least four VMs representing legitimate clients (e.g., Ubuntu or Windows) and one attacker (Kali Linux).
   - Subnet: 192.168.1.0/24, with DHCP enabled on the legitimate AP.
2. **AI System Deployment**:
   - Install and configure the AI-driven WLAN security system on a centralized server or VM.
   - Connect the system to a database for logging and storing traffic data.
3. **Tools**:
   - **Wireshark**: For traffic monitoring.
   - **Scapy**: For generating custom traffic and simulating attacks.
   - **Kali Linux**: For executing attack scenarios like MITM, ARP spoofing, and WPA2 cracking.
   - **NetfilterQueue**: For managing real-time packet flow and implementing automated responses.

Attack Scenarios

1. **Rogue Access Point Detection**:
   - Launch a rogue AP using tools like airbase-ng from the Kali Linux VM.
   - Configure the rogue AP to use the same SSID as the legitimate AP but with a stronger signal.
   - Test the AI system's ability to detect the rogue AP by analyzing MAC address inconsistencies, signal strength variations, or unexpected SSIDs.
2. **ARP Spoofing (Man-in-the-Middle)**:
   - Use arpspoof or ettercap to poison the ARP cache of one or more clients, redirecting traffic through the attacker’s device.
   - Verify whether the AI system identifies the anomaly based on unusual MAC-IP mappings or traffic rerouting.
3. **Denial of Service (DoS)**:
   - Execute a DoS attack using mdk3 or aireplay-ng to flood the WLAN with deauthentication frames.
   - Assess the system's ability to detect and block malicious frames while maintaining legitimate connections.
4. **WPA2 Key Cracking**:
   - Capture WPA2 handshakes using airodump-ng.
   - Simulate brute-force attempts on the captured handshake.

- Test the AI system's ability to detect the excessive authentication attempts and take mitigating actions.

Testing Procedures

1. **Traffic Data Capture**:
   - Use Wireshark or Tcpdump to collect real-time traffic during each attack scenario.
   - Store captured data in a database for AI analysis.
2. **AI System Evaluation**:
   - Feed captured data into the AI system.
   - Monitor for:
     - Detection alerts for each attack.
     - Classification accuracy for normal vs. malicious traffic.
     - Response actions triggered (e.g., blocking the rogue AP, isolating spoofed clients).
3. **Response Validation**:
   - Verify that the AI system's mitigation measures (e.g., isolating devices, reconfiguring APs) are effective without disrupting legitimate users.
4. **Performance Metrics**:
   - Detection rate (true positives).
   - False positive rate (legitimate traffic flagged as malicious).
   - Response time for mitigation actions.
   - System resource usage (CPU, memory).

Expected Outcomes

1. **Detection Accuracy**:
   - The AI system identifies rogue APs, ARP spoofing, and DoS attacks with minimal false positives.
2. **Mitigation Effectiveness**:
   - Rogue APs are disabled, and spoofed clients are isolated promptly.
   - Legitimate users remain unaffected during mitigation.
3. **Performance Insights**:
   - The system demonstrates a balance between security and usability.
   - Metrics like detection accuracy and false positive rates align with acceptable thresholds.

VII. Deliverables

1. **Functional Prototype:**
   - An AI-driven WLAN security system capable of real-time threat detection and response.
2. **Documentation:**
   - User guide for deploying and managing the system.
3. **Presentation:**
   - A comprehensive demonstration of the system in a live or simulated environment.

VIII. Learning Outcomes

1. **Technical Skills:**
   - Understanding WLAN protocols and their vulnerabilities.
   - Hands-on experience with AI models for cybersecurity applications.
2. **Critical Thinking:**
   - Ability to analyze traffic patterns and infer security risks.
3. **Practical Application:**
   - Integration of AI with traditional network security tools to build adaptive defense mechanisms.