Offensive and Defensive Wi-Fi/Bluetooth PEN Tool

Project Report

A report submitted to the

Department of Electrical and Information Engineering

Faculty of Engineering

University of Ruhuna

On 2nd of July 2025

In partial completion of the module

EE6304 - Embedded Systems Design

By Group 10:

Dewmith M.K.J.	EG/2021/4474
Sandaruwan A.M.A.H.	EG/2021/4781
Sewinda L.L.D.	EG/2021/4807

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our supervisor, Mr. D. S. De Silva and also Mr. R. Saahith Ahamed and Mr. Gevin Harindu for the guidance and support provided throughout the Embedded Systems lectures to improve ourselves. We also thank the peers of the Department of Electrical and Information Engineering for their assistance provided while testing the product in the laboratories.

ABSTRACT

This project presents the development of a portable Wi-Fi and Bluetooth Penetration Testing (PEN) Tool using the ESP32 microcontroller. The PEN tool enables both offensive and defensive wireless network analysis, supporting functions such as Wi-Fi scanning, Bluetooth device discovery, packet sniffing, rogue access point simulation and SSID flooding. A TFT touchscreen provides a user-friendly interface for interaction. Captured network traffic can be stored for offline analysis as well. This project aims to provide a low-cost, flexible alternative to commercial tools while helping students and professionals understand wireless security risks.

CONTENTS

1 INTRODUCTION

1.1 Problem Statement

In today's mutual connected world, wireless networks such as Wi-Fi and Bluetooth are integral parts of many equipment ranging from smartphones to IoT systems. However, these networks are susceptible to various cybersecurity hazards, including de-authentication attacks, evil access points and unauthorized device tracking. Ethical penetration testing is necessary to identify and reduce these weaknesses; however, many existing equipment for such testing are either complex and expensive or sometimes limited in their ability to assess networks both in offensive and defensive approaches. In addition, consumer-grade hardware like ESP32 often suffers from signal noise and limited range, which reduces effectiveness in diverse environments.

This project designs a compact and portable device based on ESP32 microcontroller, an **Offensive and Defensive Wi-Fi/Bluetooth PEN Tool**. The device will increase the strength of the signal and reduce noise interference, as well as adapted to customize wireless scanning and testing capabilities, equipped with a TFT touch display. The device will provide an interactive interface for ethical penetration testing and sniffing packets, enabling users to more effectively assess and secure the wireless network.

1.2 Objective

To design and implement a secure, portable Wi-Fi/Bluetooth penetration testing tool for noisy and high traffic networks by enhancing signal strength. ESP32 bases firmware to conduct offensive and defensive wireless security assessments with improved signal reliability and user-friendly operation.

1.3 Importance

This tool supports education and ethical hacking practices in controlled environments, helping future engineers understand wireless network vulnerabilities.

2 PROJECT SCOPE

This project focuses on a compact, user-friendly pentesting tool for Wi-Fi and Bluetooth networks using a microcontroller. Customized code enhances the device signal range, clarity and efficiency. A user involving the TFT touch screen will improve the interface. This tool is designed for ethical penetration testing applications, operating in the network to maximize security.

The device supports the following functionalities.

- **Wi-Fi Network Scanning and Analysis**: Finding details such as SSID, channel, RSSI and encryption type for network reconnaissance, access points (APs) and client devices.
- **Bluetooth Device Discovery**: Identifies nearby Bluetooth devices, including names, addresses and signal power, to evaluate Bluetooth security configuration.
- **Rough AP Simulation**: Transmits fake SSID to simulate rough access points to imitate network answers to client and unauthorized APS, this.
- **Packet Sniffing and Logging**: Capture Wi-Fi packets, frames in PCAP format, and save to an SD card for offline vulnerability analysis.
- **TFT Touch Screen based User Interface**: Display results and interact with the user, showing system prompts and menu options.
- **Graphical User Interface**: Enables users to select tasks, scroll through results and start actions through user-friendly inspection.
- **Offline Monitoring**: Stores captured packets in PCAP format for offline analysis (e.g., with Wireshark).

Although this device is designed to be secure and practical, there are several constraints as well.

- **Control Over the Selected Authorized Network for PEN Testing**: The device operates, with the devices having MAC addresses for penetration testing through Wi-Fi/Bluetooth-based remote control, enhancing security but with remote configuration.
- **Range**: Depending on the antenna quality (especially the inbuilt antenna), strength of the measuring signals and noise will vary with the network environment (typically 50-100 meters with internal antenna).
- **De-authentication Attacks**: Due to the restrictions of ESP32-IDF, device cannot send de-auth frames which in result, limits certain Wi-Fi attack simulations.
- **No Cloud Integration**: The system avoids cloud connectivity to prevent remote attacks; however, this approach restricts features like real-time data sharing.

3 SPECIFICATIONS

3.1 Network Analysis through Real Time Operations

Wi-Fi Network Analysis:

- Access Point Scan: Detects SSIDs, channels, RSSI and encryption types (e.g., WEP, WPA2).
- Station Scanning: Identifies connected client devices and helps to detect rough devices.
- Channel Analyzer: Displays Wi-Fi channel activity to identify congestion or anomalies in the network.
- Packet Sniffing: Captures PCAP Wi-Fi packages for offline analysis and reveals non-encrypted traffic or weaknesses in the network protocol.

Discovery of Bluetooth Devices:

- Classic Bluetooth Scanning: Lists nearby devices with names, addresses and signal forces.
- BLE Scanning: Detects low-energy Bluetooth devices which are common in IoT, to evaluate pairing or firmware vulnerabilities.

Rough Access Point Simulation:

- Broadcasting Fake AP: Creates fake SSIDs to test customer self-connection behaviors or network monitoring systems.
- Use Case: Helps to identify devices that are vulnerable to connecting with unauthorized or rogue access points.

SSID Flooding:

- Flood Attack Simulation: Transmits several false SSIDs to stress-test network scanners or client devices.
- Usage: Evaluates the effectiveness of the Intrusion Detection System (IDS) against the flooding of APs.

Offensive Attacks for PEN Testing:

- Port Control: Uses evil port attacks for ethical operations.
- Data Portability: Stores logs on the Zero Flipper SD card.

3.2 Offline Operation

The device will function entirely offline to monitor and analyze PCAP files on SD card.

- PCAP Storage: Saves captured packets to SD card for detailed analysis with tools like Wireshark.
- Scan Logs: Records scan results for reporting and documentation.

4 SYSTEM DESIGN

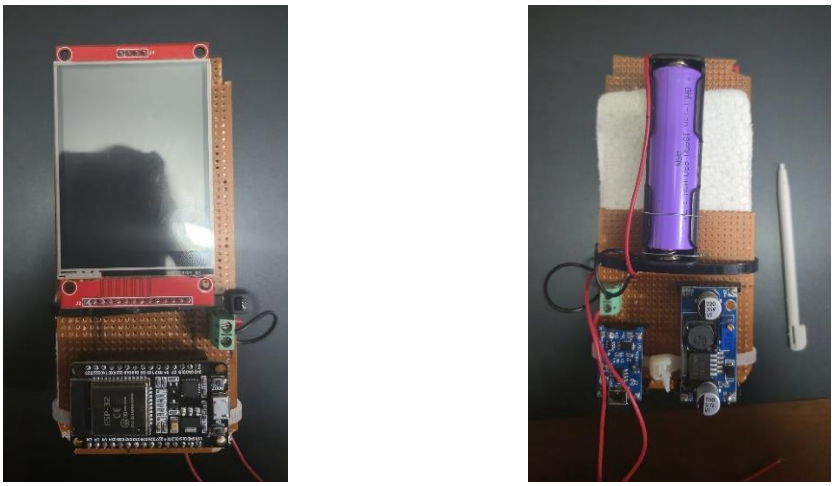

Figure 4.1: System Build

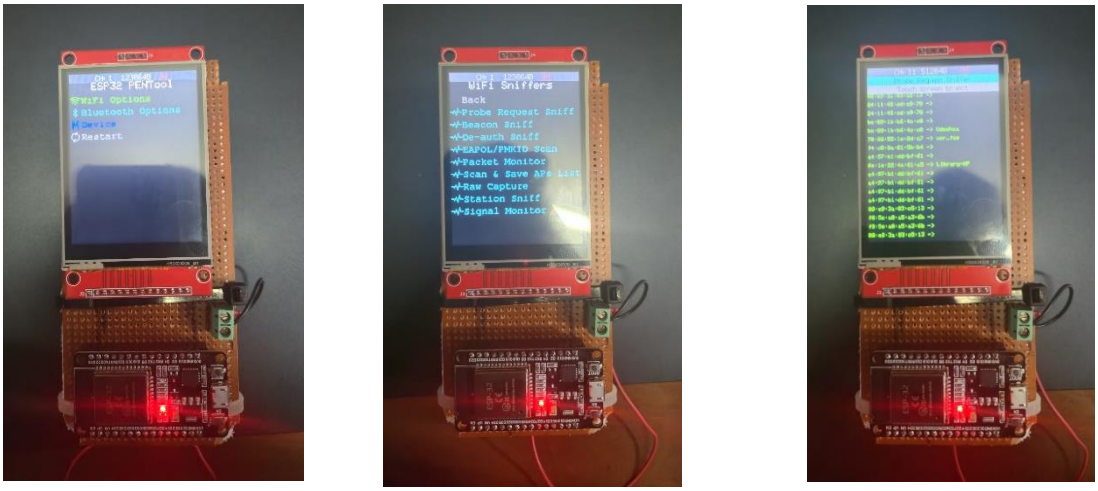

Figure 4.2: Display Implementation

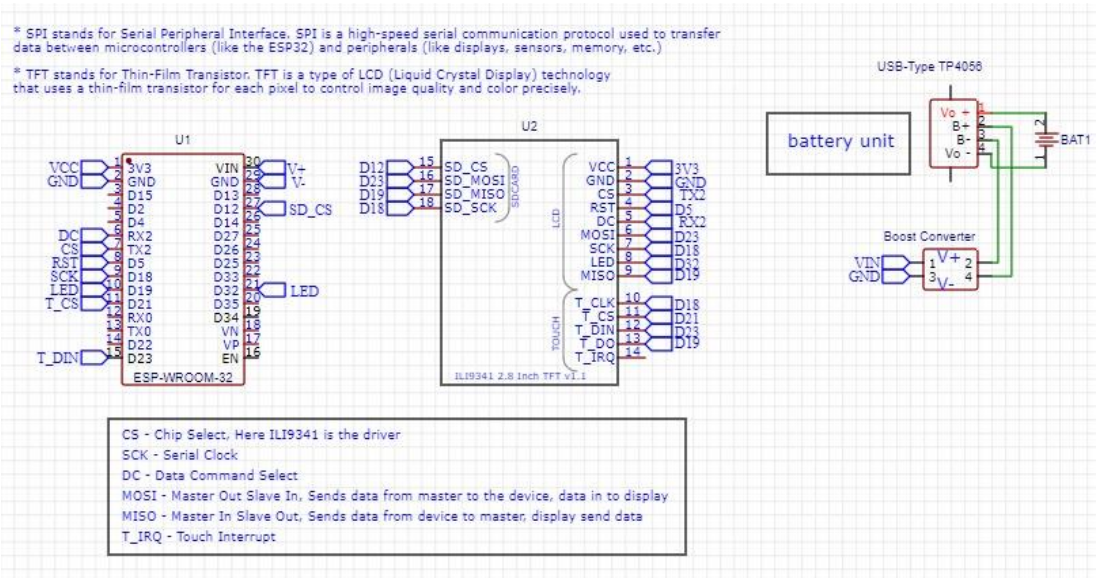

Figure 4.3: Easy EDA Design of the Circuit

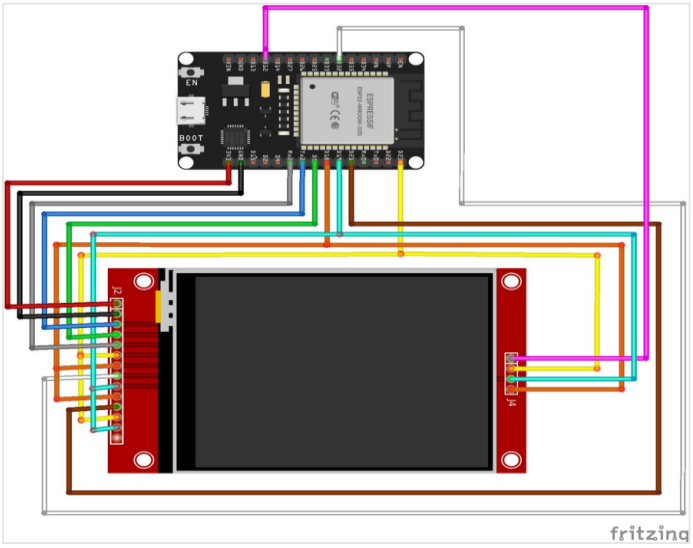

Figure 4.4: Design of the Circuit Fritzing Software

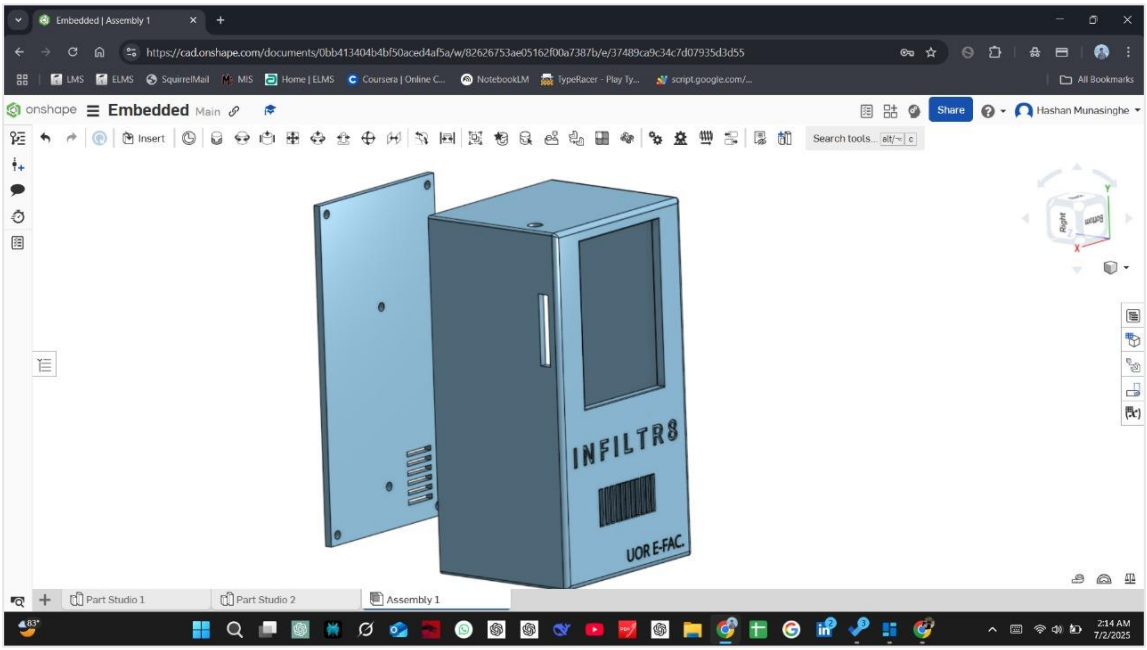

Figure 4.5: Enclosure Design Using Onshape

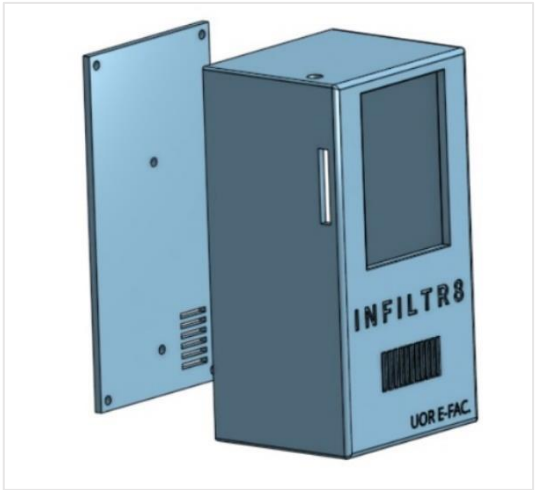

Figure 4.6: Enclosure of the PEN Tool

5 RESULTS AND TESTING

- All modules functioned as expected
- Successfully scanned APs and Bluetooth devices
- GUI worked smoothly with the touchscreen

5.1 Features Implemented

Here, are the features we have implemented in the device.

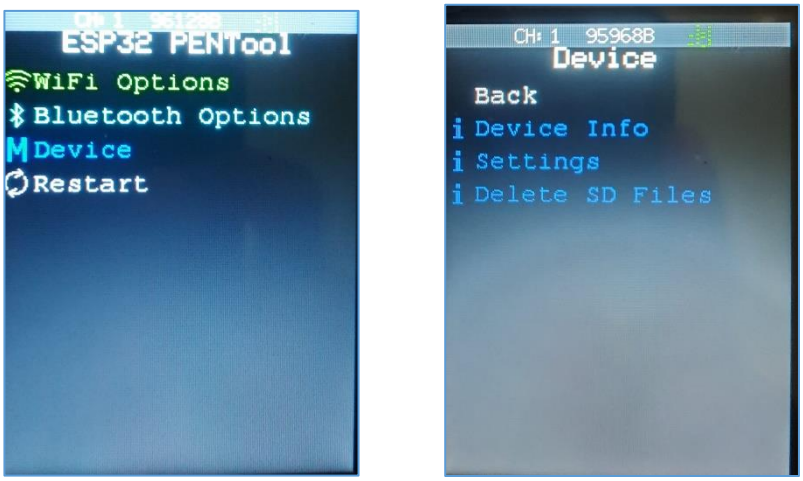

Figure 5.1: Home Menu and Device Settings

Wi-fi Options

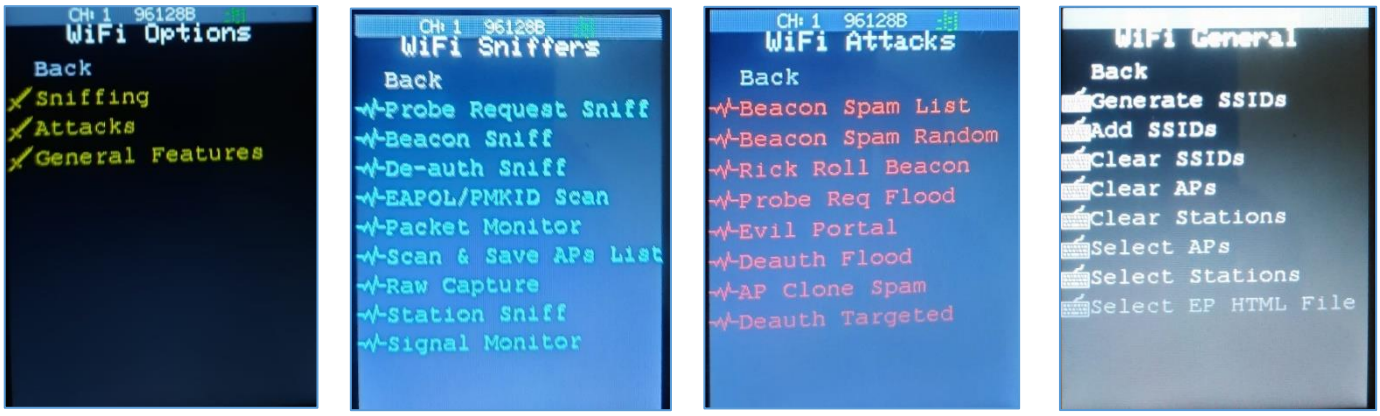

Figure 5.2: Wi-Fi Options Implemented

Wi-fi Sniffing (Silently Listening)

- Probe Request Sniffer

  Check for probe request, when a wi-fi device is basically pining other devices looking for nearby SSID. It will show up down as a list of mac address. It is going to show any nearby sell phones (looking for AP) If the cell phone is idle, it requests hundreds of probe request for an hour. Determine which wi-fi devices are in any given area.

- Beacon Sniffer

  Sniffing out Wi-Fi AP nearby only.

- Scan and Save APs List

  Save the APs nearby to analyze.

Wireshark Analysis on Beacon Sniffing:

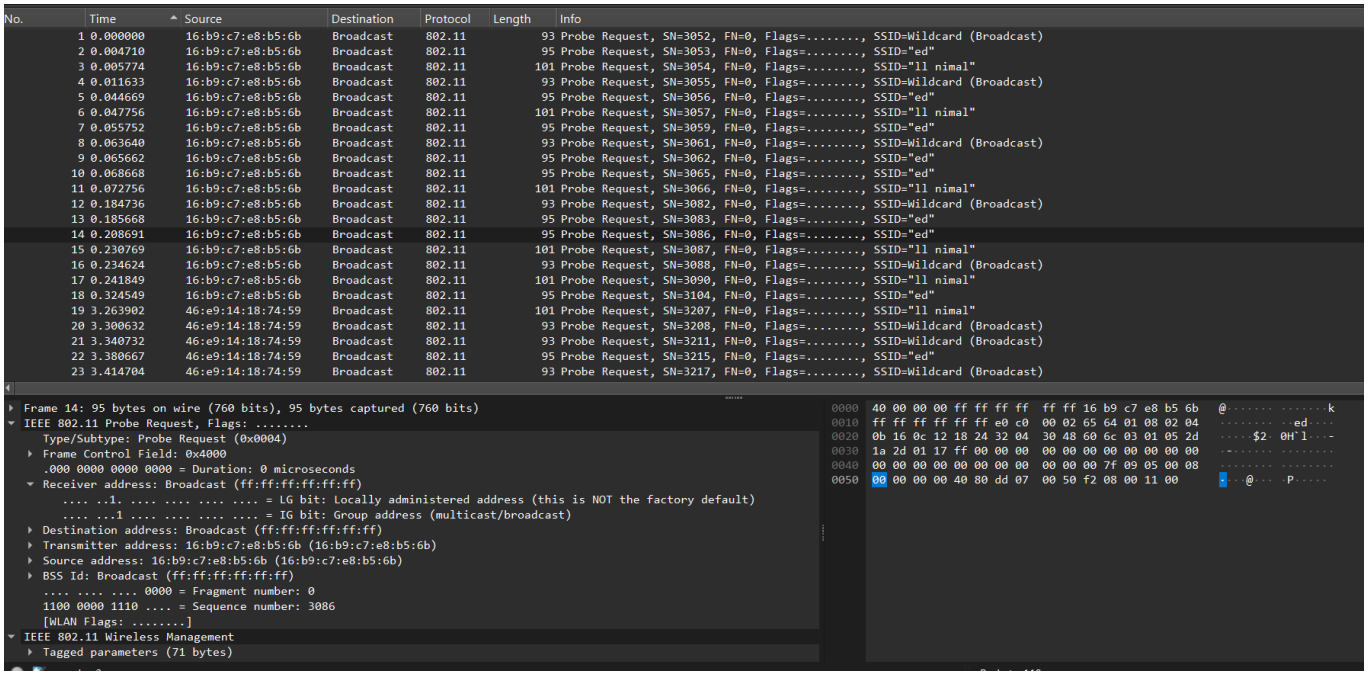

Figure 5.3: Beacon Sniffing Analysis on Wireshark

- De-auth Sniffer

  Searching for the DE authentication packets. If someone is going to attack, we can observe the kick off device (AP) from this (e.g., router)

Demonstration of De-auth Sniffing:

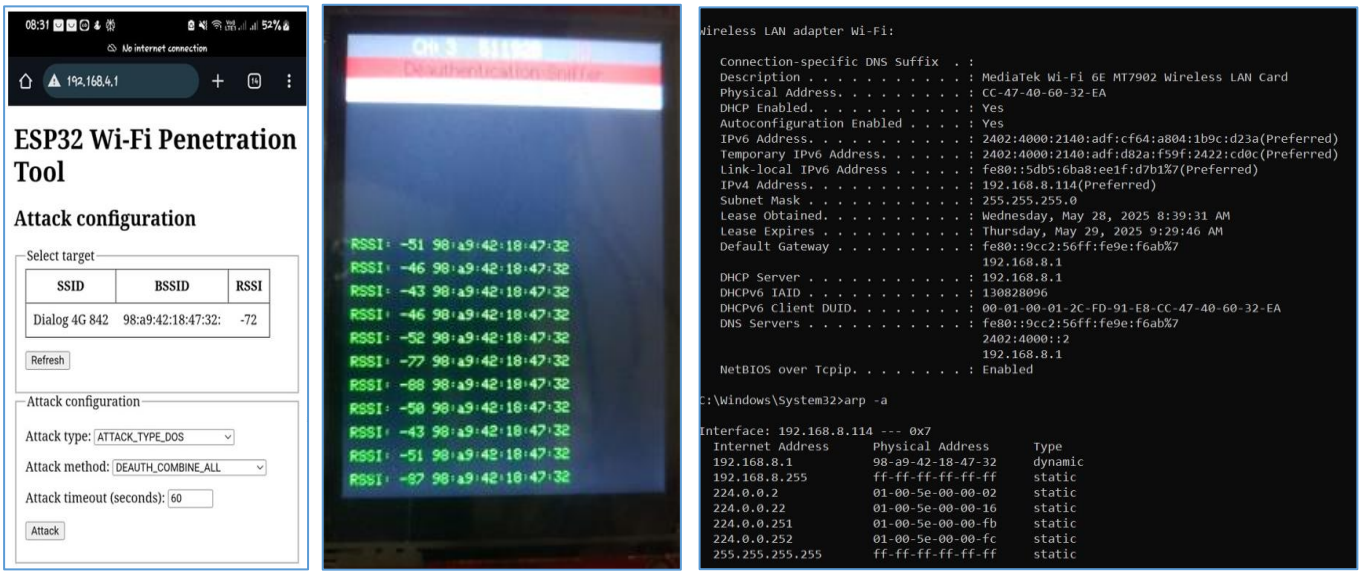

Figure 5.4: Demonstration of De-auth Sniffing

I. Offensive Mode

The first screenshot demonstrates a DE authentication Attack being sent to the target access point: Dialog 4G 842. The attack parameters were configured through the PEN Tool's touchscreen interface.

II. Defensive Monitoring

The second screenshot shows real-time monitoring of the targeted AP's MAC address and RSSI using the PEN Tool. This operation is performed ethically within an authorized testing area.

III. Validation of Attack

The third image confirms that devices connected to the targeted AP were successfully identified using the ARP table on a terminal. During the DE-auth attack, these connected clients were disconnected (kicked off) as expected.

Attacker can get the snip handshake when the device tries to reconnect and attacker can do MIM attacks during that time. (logins, credentials, bank details, etc.)

- EAPOL/PMKID Scan

  Search for specific packet sent over wi-fi

  Extensive Authentication Protocol over Lan (Four-way handshake packets device uses to authenticate credentials with the access point) Devices uses to authenticate credentials with an AP we used those earlier again to crack through hashcat.

  PMKID- Pairwise Master Key Identifier (Similar to the handshake packets but only one of them and it doesn't require the device to be de-authenticated and re-authenticated. This way hacker can get the only one file and use that to grab the hash over the Wi-fi network. (Hash- simply a string of data that's converted to another type. Typically, Encryptions). Then we can save them into the SD card.

- Raw Capture

  Showing around me MACs with RSSI resulting how strong the receiving signal strength.

- Station Sniffer

  First select a specific AP (Wi-fi general → Select AP) through saving the AP in the Save AP option. It sniffs out any device that is attach to that chosen AP. (Popping a list)

  Offensive attack: We can give a targeted de-auth attack directly. More over evil twin attack through sending de-auths.

- Signal Monitor

  Need to connect to a directly to a network and it shows the signal strength for that network.

<u>Wi-Fi Attacks</u>

- Beacon Spam List

  Spam any Aps and SSIDs that are in in our saved list. Add an AP using the (General→Add AP)

  *Our own AP is in the Network

- Beacon Spam Random

  Create many numbers of APs and send it to the Network. Really annoying thing.

- Rick Roll Beacon

  Sends out the Rick Astley's classic song as SSID phrases.

- Probe Request Flood

  Send Probe request to AP over and over in a rapid rate. Can jam the access point. Devices in the network can't be connected to that AP.

- De-auth Flood

  Send the de-authentication packets to the all AP in the network. Just like the demonstration in the above.

- AP Clone Spam

  Send out cloned SSIDs for any ither access point n the list.

- Deuth Targeted

  First, Select Stations (AP and others devises that are on attached to our network). When running this by selecting one of station (MAC) it will attack on the network and try to de-authenticate just one device. De-off one device.

Wi-fi General

Normal Functionalities are there. (Generate or Clear SSIDs, Select APs, Stations)

Bluetooth Options

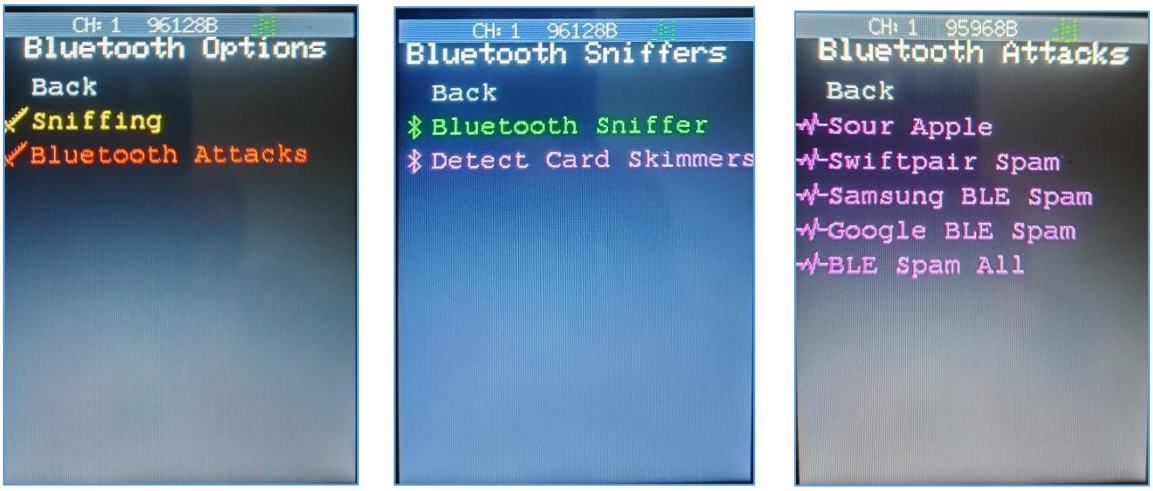

Figure 5.5: Bluetooth Options Implemented

Sniffing

- Bluetooth Sniffers

  Sniffing out the Bluetooth signal devices.

- Detect card Skimmers

  Detect the credit card skimmers that people are installing on credit card or ATM readers. (Supermarkets)

Bluetooth Attacks

- Sour Apple

  Spoofs BLE advertisements that Apple devices. Broadcasts fake BLE advertisements imitating peripherals

- Swift Pair Spam

  Continuously sends BLE advertisements mimicking devices ready to pair. Target: Windows 10 and 11 systems. Windows shows popups like “Tap to connect” over and over. This is designed to be annoying or distracting rather than fully exploitative.

- Samsung BLE Spam

  Samsung Android devices are targeted. Spoofs BLE devices that Samsung phones will attempt to auto-recognize or interact with. Impact is the Samsung phones may auto-launch Bluetooth dialogs or notifications. This can be sued to Testing BLE advertisement parsing behavior in Samsung's One UI/Android BLE stack.

- BLE Spam

  Any BLE-capable device. This broadcast randomized or custom BLE advertisements repeatedly.

  Crowded the BLE space by sends multiple fake BLE packets that advertise fake devices.

- BLE Spam All

  Universal — All BLE-capable devices. A combo mode that runs all the above attacks together. To simulates BLE DoS or stress test scenario.

5.2 Implementation Steps

1. Solder touchscreen to ESP32 using SPI
2. Connect SD card module to SPI pins
3. Flash firmware via USB
4. Test basic boot-up, touchscreen calibration
5. Validate scanning and sniffing
6. Capture packets and test analysis on Wireshark
7. Test AP spoof and SSID flooding features
8. Evaluate power and signal stability

6 CONCLUSION

We successfully built a Wi-Fi/Bluetooth PEN Tool using ESP32 microcontroller. The system supports multiple functions for wireless security testing and education. The use of a touchscreen and offline storage enhanced the usability. This tool works well for education and field analysis. Although the ESP32 is cost-effective, it has limits in attack simulation. This PEN tool enhances understanding of wireless protocol vulnerabilities and enables ethical testing in a controlled environment.