Project Report: Wi-Fi Deauther Using ESP8266

About Me

I am Himangshu Rana, passionate cybersecurity enthusiast with a keen interest in ethical hacking and network security. I enjoy learning about how digital systems can be vulnerable and am always looking for new knowledge and skills to stay ahead of emerging cyber threats. This project on Wi-Fi Deauther using the ESP8266 is a part of my journey to explore and understand different cybersecurity tools. My goal is to help create safer networks while always following ethical practices in the ever-changing world of cybersecurity.

Guidance

This project was completed under the expert guidance of **Ms. Khushbu Thakur**, a seasoned professional with extensive experience in Java full-stack development, Python-based data analytics, and embedded systems. Ms. Thakur`s mentorship was instrumental in ensuring the technical accuracy and educational value of this project. Her notable achievements include:

- Over two years of experience as a Senior Java Faculty at Anudip Foundation, where she has mentored more than 400 students in full-stack project development.
- A strong background in teaching advanced topics such as Spring Boot, Hibernate, Angular, and Python analytics.
- Industry expertise in developing embedded systems and process control instruments.

I am grateful for her guidance and support throughout this project.

Abstract

This report explains how to build and use a Wi-Fi deauther with an ESP8266 microcontroller. The project demonstrates how deauthentication attacks can disrupt wireless connections, emphasising the security risks in Wi-Fi networks. The purpose is to raise awareness about these vulnerabilities while promoting ethical usage.

Introduction

Wi-Fi is one of the most popular forms of wireless communication today. It is used in homes, offices, and public spaces for various purposes, including internet access, smart devices, and industrial controls. Despite its convenience, Wi-Fi technology has vulnerabilities. One of the simplest ways to exploit a Wi-Fi network is by performing a deauthentication attack.

This project focuses on creating a Wi-Fi deauther using the ESP8266 module. A deauther sends deauthentication packets to disconnect devices from a Wi-Fi network. These packets force the targeted devices to repeatedly reconnect to the network, disrupting normal operation.

While the project showcases how to exploit a vulnerability, its main goal is to educate users on the risks and encourage better security practices.

Objectives

1. Build a functional Wi-Fi deauther using an ESP8266 module.
2. Demonstrate how deauthentication attacks work on Wi-Fi networks.
3. Educate users on the need for better network security.
4. Highlight the ethical implications of using such tools.

Components Required

To build a Wi-Fi deauther, the following components are necessary:

1. **ESP8266 Microcontroller**: A compact, low-cost Wi-Fi module.
2. **Power Source**: LiPo battery or USB power bank for portability.
3. **Micro USB Cable**: To program and power the ESP8266.
4. **Arduino IDE**: For writing and uploading the code.
5. **Laptop or Computer**: To program the ESP8266 and test the project.

Working Principle

Wi-Fi networks rely on the IEEE 802.11 protocol to manage communication between devices and access points. The protocol includes a deauthentication frame used to disconnect devices for legitimate reasons, such as when switching networks or troubleshooting. A Wi-Fi deauther exploits this by sending fake deauthentication frames, forcing devices to disconnect from the network.

The ESP8266 module is programmed to:

- Scan for available Wi-Fi networks.
- Identify connected devices.
- Send deauthentication packets to targeted devices.

This disrupts communication between devices and the Wi-Fi router, causing service interruptions.

Circuit Design

The circuit design for the Wi-Fi deauther is simple. The ESP8266 module connects directly to the power source, such as a USB cable or a battery. No additional components are required.

Connection Diagram:

1. **ESP8266 Microcontroller**: Connects via a micro USB cable.
2. **Power Source**: Provides energy to run the module.

Software Setup

Step 1: Install Arduino IDE

1. Download the Arduino IDE from the official website: https://www.arduino.cc.
2. Install the software and open it.

Step 2: Configure the ESP8266 Board

1. Go to File > Preferences in the Arduino IDE.
2. Add the following URL to the additional board manager section:
3. http://arduino.esp8266.com/stable/package_esp8266com_index.json
4. Open Tools > Board > Boards Manager.
5. Search for "ESP8266" and install the package.

Step 3: Download and Upload Code

1. Clone or download the Wi-Fi deauther project from GitHub (https://github.com/spacehuhn/esp8266_deauther).
2. Open the downloaded file in the Arduino IDE.

3. Select the appropriate board (e.g., NodeMCU 1.0 or ESP8266) from Tools > Board.
4. Choose the correct port under Tools > Port.
5. Click the "Upload" button to flash the code onto the ESP8266.

Step 4: Testing the Device

1. Power the ESP8266 using a USB cable or battery.
2. Connect to the deauther’ s Wi-Fi network (default SSID: "pwned").
3. Open a web browser and navigate to 192.168.4.1.
4. Use the web interface to select a Wi-Fi network and begin the deauthentication attack.

Features

- **Network Scanning**: Lists all available Wi-Fi networks and connected devices.
- **Selective Targeting**: Allows targeting specific devices or entire networks.
- **Portable Design**: Powered by a USB or battery for on-the-go use.
- **Web Interface**: Simple and intuitive control through a browser.

Applications

1. **Educational Demonstrations**: Showcases vulnerabilities in Wi-Fi networks for learning purposes.
2. **Security Testing**: Assists network administrators in testing their network’ s resilience.
3. **Awareness Campaigns**: Promotes the importance of network security.

Ethical Considerations

This project is for educational purposes only. Performing deauthentication attacks without proper authorisation is illegal and unethical. Misusing this tool can disrupt legitimate services, cause inconvenience, and potentially result in legal consequences.

Guidelines for Ethical Use:

- Obtain explicit permission before testing any network.
- Use the device in controlled environments only.
- Focus on improving network security rather than exploiting vulnerabilities.

Limitations

1. **Frequency Restriction**: Only works on 2.4 GHz Wi-Fi networks; does not affect 5 GHz networks.
2. **Range**: Limited by the ESP8266’ s transmission power and antenna quality.
3. **Detection**: Modern networks equipped with intrusion prevention systems can detect deauthentication attacks.
4. **Legal Constraints**: Using this device without permission is against the law in most countries.

Advantages

- **Cost-Effective**: Requires inexpensive components.
- **Easy to Build**: Simple setup process suitable for beginners.
- **Educational Value**: Demonstrates the vulnerabilities in Wi-Fi technology.

Mitigation Techniques

To protect against deauthentication attacks, consider the following measures:

1. **Use WPA3 Security**: Offers improved protection against spoofing attacks.
2. **Enable Intrusion Prevention Systems (IPS)**: Detects and blocks suspicious activities.
3. **Channel Hopping**: Automatically switches frequencies to avoid targeted attacks.
4. **Monitor Network Traffic**: Use tools to identify abnormal behaviour.

Conclusion

This project successfully demonstrates the functionality of a Wi-Fi deauther and highlights the vulnerabilities in wireless communication. While it provides valuable insights into network security, it also underscores the ethical responsibility of users. Proper use of this knowledge can help improve Wi-Fi security and reduce risks from malicious attacks.

Future Scope

1. **Support for 5 GHz Networks**: Expand functionality to newer Wi-Fi bands.
2. **Improved Detection Evasion**: Develop techniques to bypass modern security systems.
3. **Awareness Programs**: Use the device in workshops to educate users on network vulnerabilities.

References

1. Espressif Systems Documentation: https://espressif.com
2. GitHub Repository for ESP8266 Deauther: https://github.com/spacehuhn/esp8266_deauther
3. Arduino IDE Documentation: https://www.arduino.cc