Educational Walkthrough: Wi-Fi Hacking and WPA Cracking with Python and Aircrack-ng Tools

By: Computer Doctor

Introduction:

This educational walkthrough is designed to help you understand the methods used to capture and crack WPA handshakes in Wi-Fi networks. The purpose is to educate users on how attackers may try to compromise wireless networks, thereby empowering you to secure your own network against these types of attacks(**Deauthentication Attack**).

Disclaimer:

This walkthrough is intended for educational purposes **only**. Unauthorized access to networks you do not own or have permission to test is illegal and punishable by law (The Cybercrime Act, 2015 of The United Republic of Tanzania). This guide is strictly for educational purposes to aid in securing networks, not for unauthorized use.

About the Author:

Computer Doctor is a cybersecurity expert and ethical hacker dedicated to helping individuals and organizations understand digital vulnerabilities and adopt effective security measures. With a strong background in ethical hacking, penetration testing, network security, and secure software development, Computer Doctor is passionate about fostering a safer digital environment.

For questions or further guidance on using this walkthrough, please feel free to contact Computer Doctor at **help@computerdoctortz.co.tz** Visit **www.computerdoctortz.co.tz** to explore additional IT services offered.

Step 1: Understanding Wi-Fi Security and Setting Up a Practice Environment

Before diving into Wi-Fi security testing, it's crucial to understand the basics of Wi-Fi security and encryption methods and then set up a **safe practice environment** where you can explore security techniques legally and responsibly.

Wi-Fi networks are typically secured using encryption protocols, with the most common being **WEP**, **WPA**, and **WPA2/WPA3**. Each protocol protects your network from unauthorized access, but they vary in strength and reliability:

- **WEP (Wired Equivalent Privacy)**: An older security protocol, now outdated and highly vulnerable to attacks. Avoid using WEP for any real network security needs.
- **WPA (Wi-Fi Protected Access)** and **WPA2**: These protocols improve security by introducing more robust encryption, with WPA2 being the standard for most modern networks. WPA2 uses **AES encryption**, which is currently secure, although it can still be vulnerable to attacks if weak passwords are used.

- **WPA3**: The latest standard, WPA3 provides even stronger encryption and additional protection for public Wi-Fi networks. It's currently the most secure option and is recommended if your router and devices support it.

When a device (like a smartphone or laptop) connects to a Wi-Fi network, it goes through a process known as a **handshake**. The handshake establishes a secure connection between the device and the router. Capturing and analyzing this handshake can provide insights into how secure (or insecure) a network may be.

Setting Up Your Own Practice Wi-Fi Network

Since experimenting on networks without permission is illegal, the best way to practice Wi-Fi security techniques is to set up your own Wi-Fi network. Here's how you can do it:

1. **Use a Test Router or Hotspot**: If you have a router, set it up as a dedicated test network. Alternatively, you can use a mobile device to create a Wi-Fi hotspot for testing.
2. **Configure WPA2 Security**: Set the router or hotspot to use **WPA2 encryption** with a password. This will allow you to capture WPA2 handshakes and experiment with password cracking techniques without violating any laws.
3. **Choose a Simple Password**: Set a simple password for the test network. This makes it easier to understand the process of password cracking and the impact of password strength on security.
4. **Enable Monitor Mode on Your Wireless Adapter**: You'll need a wireless adapter that supports **monitor mode**, which allows you to capture packets on the network. Some laptops or external adapters can enter monitor mode, which is essential for security testing.
5. **Verify Permissions and Scope**: Ensure that this network is isolated and that you have complete permission and control over it. Avoid using this setup in areas where other networks may interfere or where unintended devices might connect.

Step 2: Enabling Monitor Mode on Your Wireless Adapter

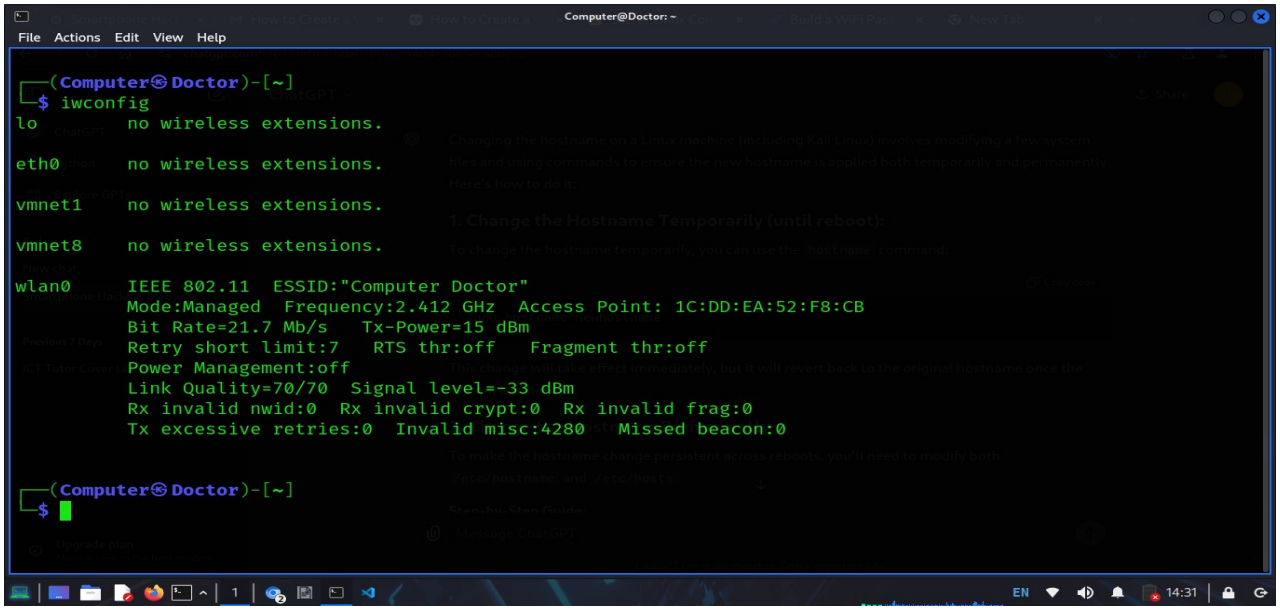
To capture packets on a Wi-Fi network and analyze network traffic, your wireless adapter must be set to monitor mode. Monitor mode allows your adapter to listen to all network traffic on a channel without needing to connect to the network. This is essential for capturing WPA handshakes.

The steps below assume you are using a Linux-based system (such as Kali Linux), which is ideal for network security testing due to its pre-installed tools.

1. **Identify Your Wireless Adapter Interface**

   First, open a terminal window and use the following command to list your network interfaces and identify your wireless adapter:

   $iwconfig , $ip a or $ifconfig



You should see a list of interfaces. Look for the wireless adapter (often named wlan0, for my computer).

2. **Disable NetworkManager**

   $sudo systemctl stop NetworkManager

   You can restart/re-enable latter after the pracitce using command

   $sudo systemctl start NetworkManager

3. **Enable Monitor Mode**

   using tool **Airmon-ng** (part of the Aircrack-ng suite), you can use it to set up monitor mode more easily:

   $sudo airmon-ng start wlan0

   (be sure to chek the name of your interface wireless card)

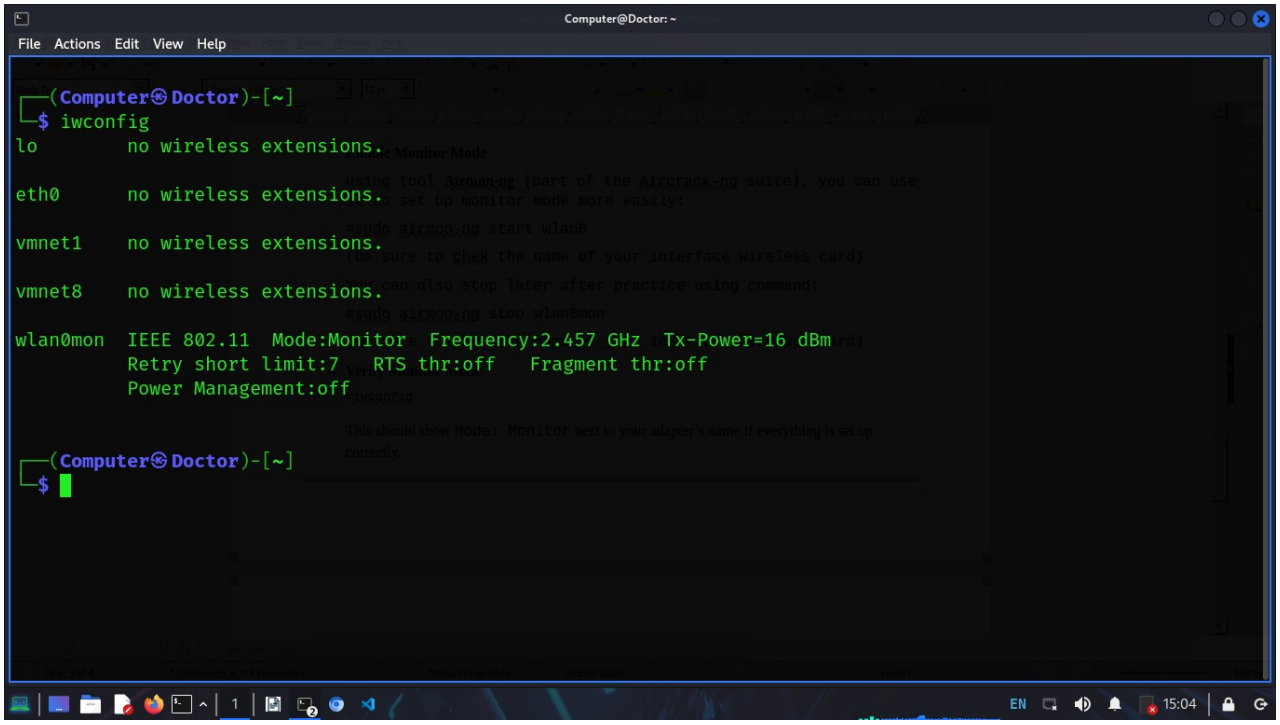
   You can also stop later after practice using command:

   $sudo airmon-ng stop wlan0mon

   (be sure to chek the name of your interface wireless card)

4. **Verify Monitor Mode**

   $iwconfig



   This should show Mode: Monitor.

Step 3: Using Airodump-ng to Identify BSSID and Channel

Before running the Python script to capture and crack WPA handshakes, you first need to gather some essential information about the target Wi-Fi network. Specifically, you'll need the **BSSID** (MAC address of the router or Access Point) and the **channel** on which the network operates. This information is necessary for the script to focus on the correct network when capturing the handshake.

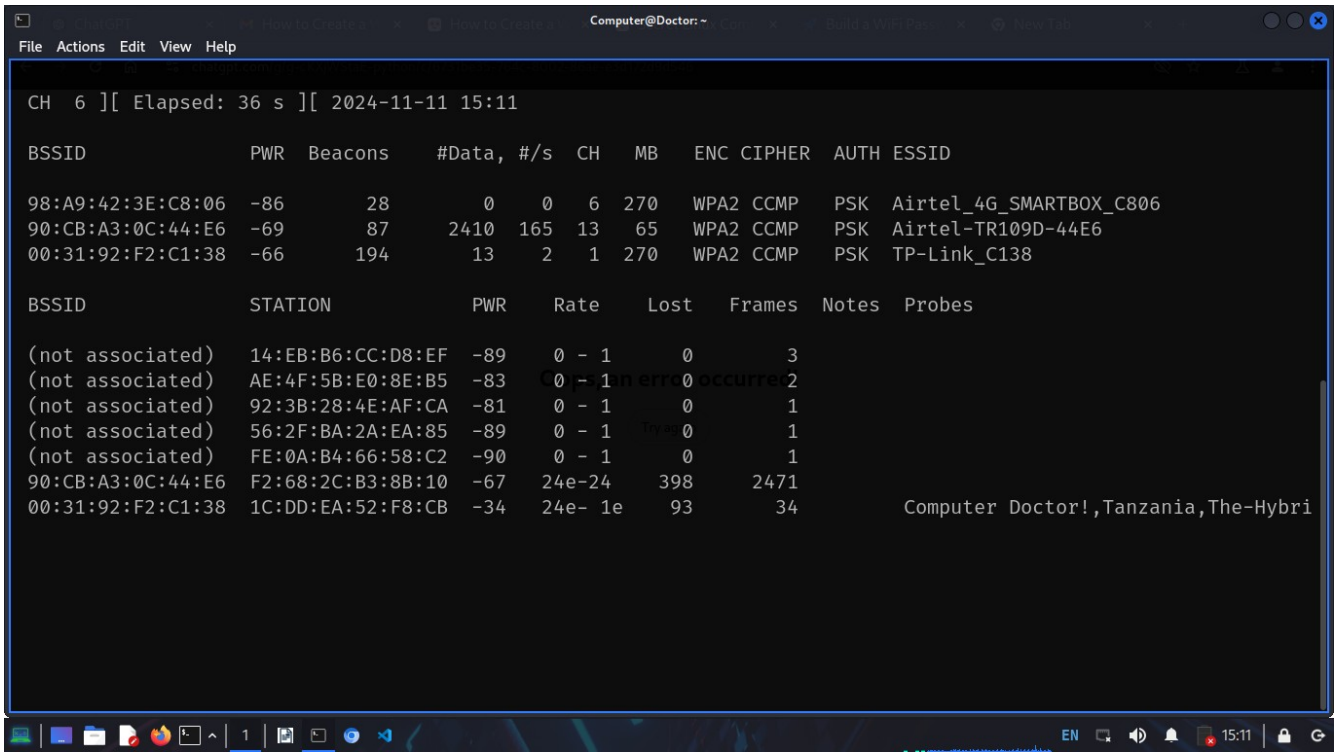
Airodump-ng is a tool within the Aircrack-ng suite that allows you to monitor all Wi-Fi networks in range, providing key details like the BSSID, channel, and signal strength of each network.

Use command:

$sudo airodump-ng wlan0mon

(Be sure to replace with your actual wireless interface adapter name, for mine was wlan0mon and already set to Monitor mode)

The outpt displayed in screenshot bellow shows all hosts devices and stations: for this practice I used **00:31:92:F2:C1:38 -66 194 13 2 1 270 WPA2 CCMP PSK TP-Link_C138**.



From the captured image above, we can see the following from the TP-Link_C138 that is the router used to test/demonstrate this task of WI-fi password cracking.

BSSID: The MAC address of the target router (**`00:31:92:F2:C1:38`**).

Channel: The channel number of the target network (**`1`**).

NB:

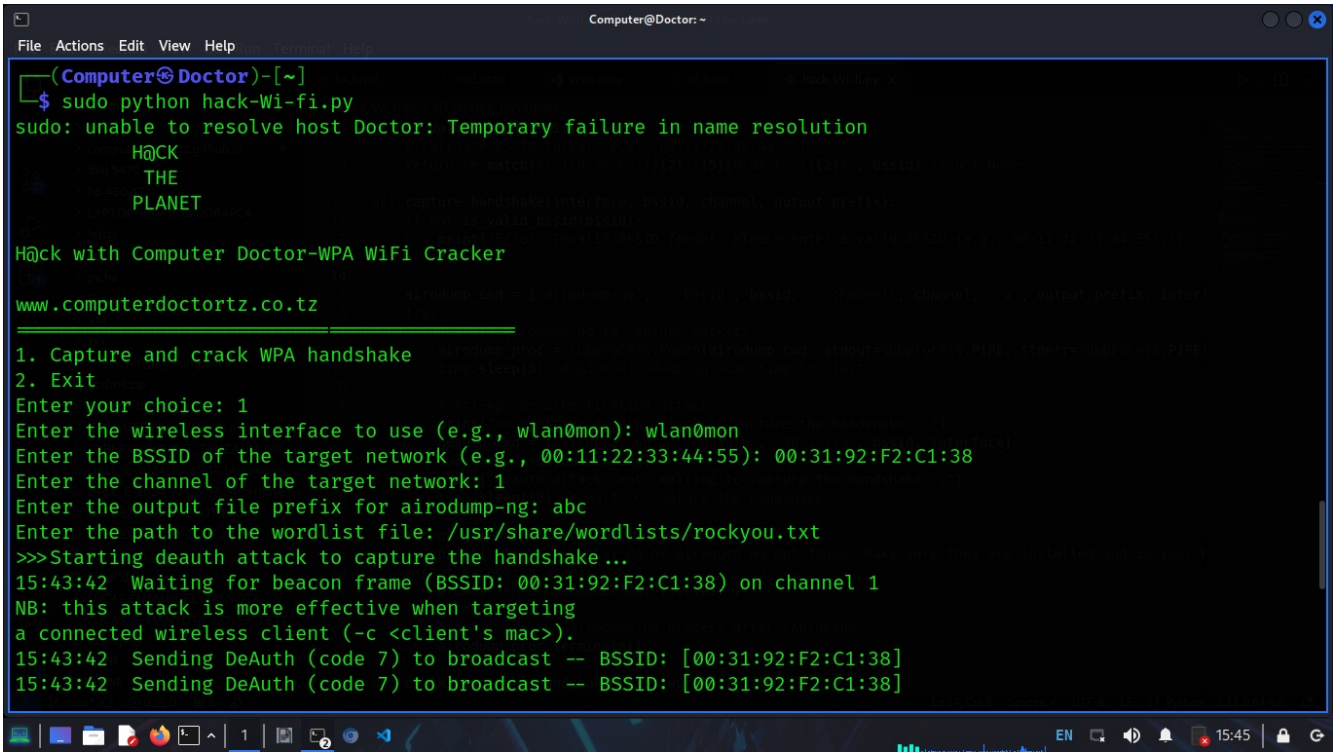
You need also to know the directory path of your wordlists if you created for you or if your using the default. Example in this walkthrough the default wordlists of Rockyou.txt was used which it's path is **/usr/share/wordlists/rockyou.txt**

Step 4: Using the Python Script to Capture and Crack the WPA Handshake

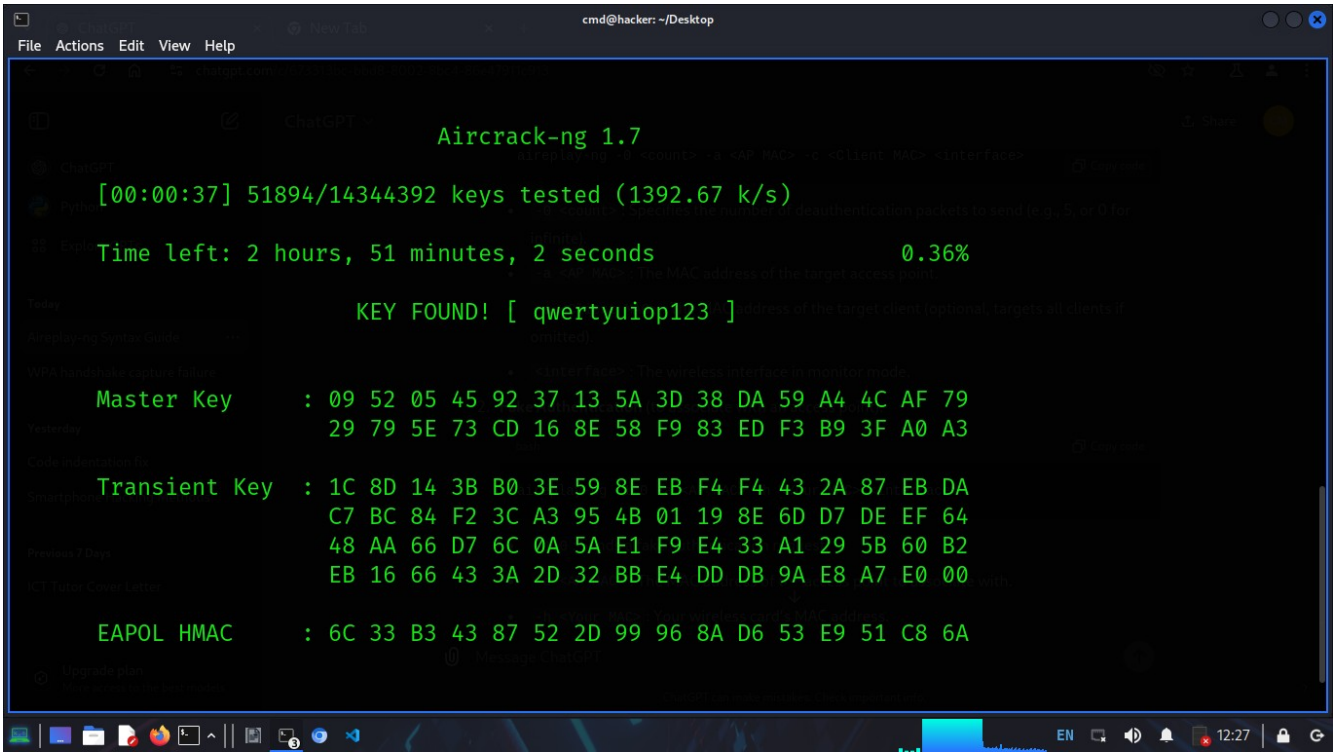
Now, you have the **BSSID** and **channel** of your target network from Step 3, you're ready to use the Python script provided. This script automates the process of capturing a WPA handshake and attempting to crack it, saving time and simplifying the procedure by combining multiple commands and tools into one script. I created the script using python programming language inter-grate with the aircrack-ng tool suite to make all task in a single terminal then simplify the task and crack a WiFi password in a menu like buying an internet bundle for your smartphone. (Laugh….)

To run the script move to the directory where the script is, or you can specify direct when executing the command. Since the script I wrote using **Python** then to run the script you will start with python. How ever the aircrack-ng tool need root prevellege to run therefore you will need to include **sudo** command;

$sudo python hack-Wi-fi.py



After run the python script enter all input correctly as appear to your WiFi you set up for practice then wait the script to make all task after the input of correct requirements. When the password found you will see the output like this bellow:



Wow! Now you can see the key found, the password for **TP-Link_C138** was **qwertyuiop123**