IT321-Ethical Hacking

Lab9: Wireless Network Hacking

Student Name: Adwait Purao UCID:2021300101

Objective: To perform a wireless security assessment (audit) and penetration testing

Outcomes:

[1] Describe wireless network architecture and terminology
[2] Identify wireless network types and forms of authentication
[3] Explain the various types of attacks on wireless networks.
[4] Install and configure the wireless network
[5] Identify wireless hacking methods and tools
[6] Demonstrate the methodology and steps for testing wireless networks

System Requirements:

[1] VirtualBox
[2] Kali Linux
[3] Wireless NIC compatible for monitor mode (USB Dongle)
[4] Wireless Client (Linux/Windows)
[5] Wireless Router or Access Point (A)- Optional
[6] Raspberry Pi Board with Bootable Kali Linux- Optional
[7] Wireless Tools- ,nexmon, Kismet, Netstubmber, airodump-ng, airplay-ng, airmon-ng, aircrack-ng,hostapd,wifite,wifi-radar

Introduction:

Wireless penetration testing comprises six main steps: reconnaissance, identifying wireless networks, vulnerability research, exploitation, reporting, and remediation.

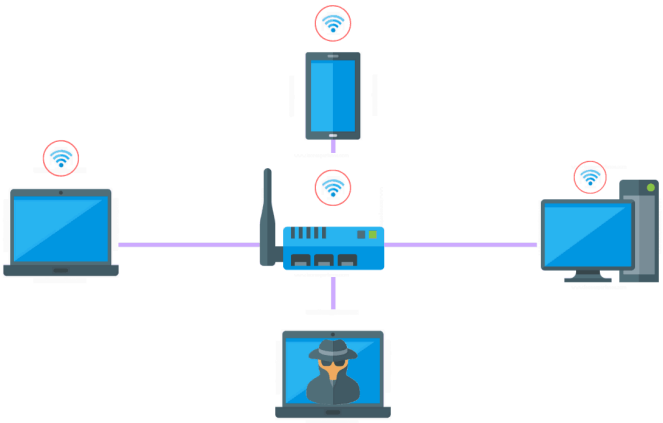


Figure-1: Wireless Network Setup [Courtesy/Source: Purplesec and Google Images]

Wireless Network Penetration Methodology:

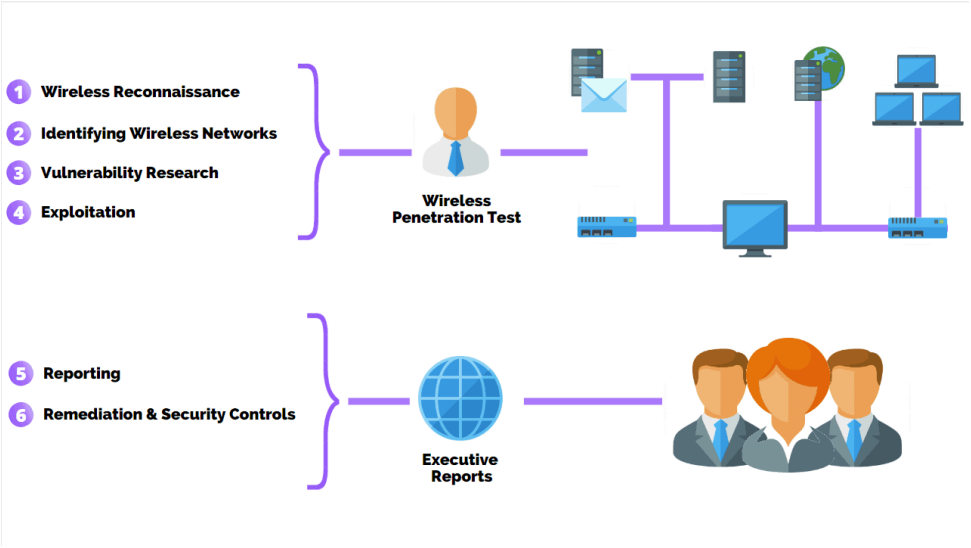


Figure-2: Wireless Network Penetration Methodology [Courtesy/Source: Purplesec and Google Images]

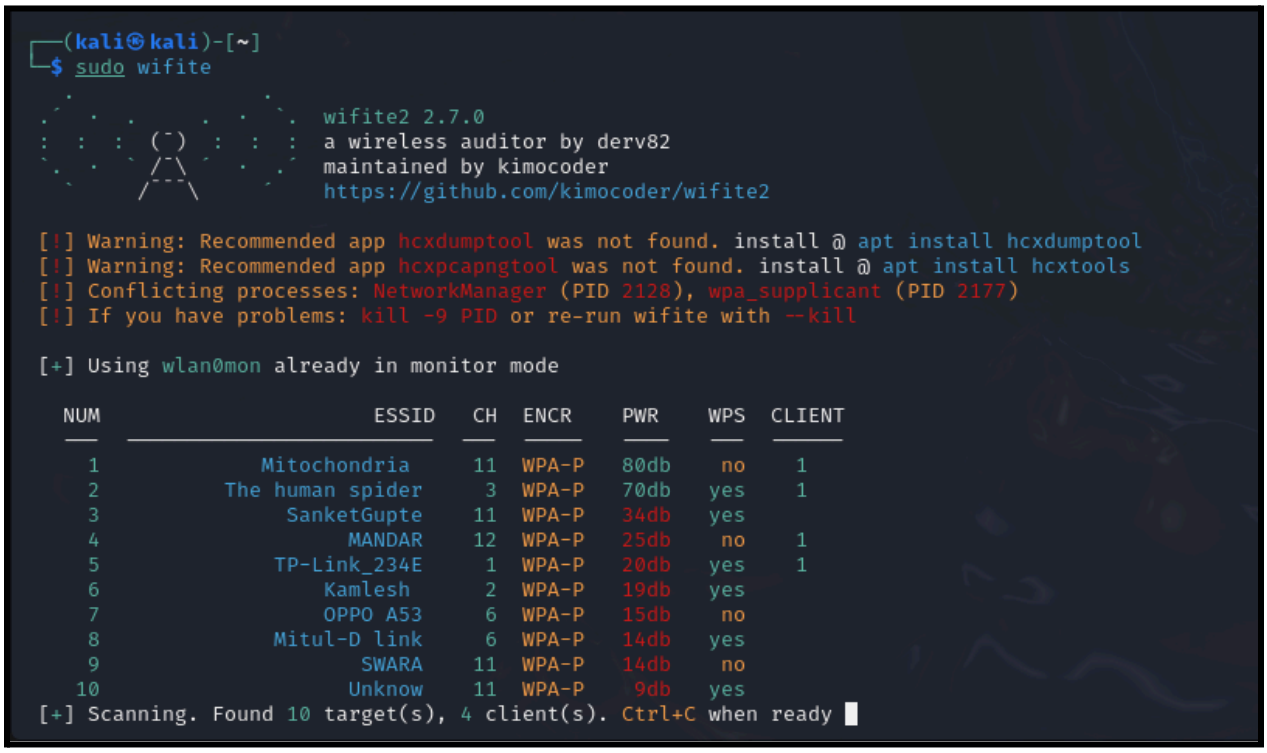
About wireless Tools:

[1] **Nexmon-** It is our C-based firmware patching framework for Broadcom/Cypress WiFi chips that enables you to write your own firmware patches, for example, to enable monitor mode with radiotap headers and frame injection.
[2] **hostapd** - user space IEEE 802.11 AP and IEEE 802.1X/WPA/WPA2/EAP Authenticator
[2] **airmon-ng** - bash script designed to turn wireless cards into monitor mode.
[3] **airodump-ng** - a wireless packet capture tool for aircrack-ng
[4] **aircrack-ng - a 802.11 WEP / WPA-PSK key cracker**
[5] **airbase-ng** - multi-purpose tool aimed at attacking clients as opposed to the Access Point (AP) itself
[6] **aireplay-ng** - inject packets into a wireless network to generate traffic
[7] **Kismet**- Wireless Sniffing and monitoring
[8] **Wifite**- Python script to automate wireless auditing using aircrack-ng tool

[9] **Wifi-radar**- graphical utility for managing Wi-Fi profiles
[10] **NetSurveyor** is a diagnostic tool that falls under the category of WiFi Scanners or 802.11 Network Discovery Tools.

Execution:
Tool used : Wifite

1) Network Assessment



I started wifite with administrative privileges using sudo.The tool thus started checking if my wireless network adapter supports monitor mode. If it does then only i can perform the attack.

Wifite then Scans network for wifi and their details. This includes their ESSID, Channels, Encryption protocol like WPA, WEP, etc and Strength of signal. An attack can be performed by stopping this scan. Wifite will then ask to select a target.

2) The attack

After selecting a target, wifite will execute all possible attacks (known to it).
In this case, only one attack, the WPA handshake attack was performed. And the password was found in the wordlist-probable.txt password list file. Other attacks include PixieDust, Null key, PMKID, etc.

WPA Handshake attack

1) Wifite captures all packets sent to the victim from any host
2) Wifite compares this packets with the format of WPA handshake packets
3) WPA handshake packets are only sent when a device connects to the wifi.
4) If a client is already connected to the wifi prior to the attack, Wifite deauths the client so that an handshake happens
5) This captured WPA handshake is stored in a temporary file.
6) Wifite checks each password from wordlist to the WPA handshake hash.
7) When a match occurs, the attack is complete

If a user has set a strong password that is not in the wordlist, this attack is useless.

Conclusion:

In wireless attacks, success depends on exploiting vulnerabilities in security protocols or weaknesses in network configurations. It is easy to capture wireless packets as long as the attacker is in vicinity. While tools like Wifite automate the process, their effectiveness is limited by the strength of passwords and the security measures implemented by the target network, rendering them ineffective against robust defenses.

References:

[1] https://github.com/derv82/wifite2
[2] http://nutsaboutnets.com/archives/netsurveyor-wifi-scanner/
[3] https://purplesec.us/perform-wireless-penetration-test/