

Informe Laboratorio 3

Sección x

Alumno x

e-mail: alumno.contacto@mail.udp.cl

Mayo de 2023

Índice

1. Descripción de actividades	2
2. Desarrollo (PASO 1)	3
2.1. identificar en qué se destaca la red del informante del resto	3
2.2. explica matemáticamente porqué se requieren más de 5000 paquetes para obtener la pass	4
2.3. obtiene la password con ataque por defecto de aircrack-ng	5
2.4. indica el tiempo que demoró en obtener la password	5
2.5. descifra el contenido capturado	6
2.6. describe como obtiene la url de donde descargar el archivo	6
3. Desarrollo (PASO 2)	8
3.1. indica script para modificar diccionario original	8
3.2. cantidad de passwords finales que contiene rockyou_mod.dic	10
4. Desarrollo (Paso 3)	12
4.1. obtiene contraseña con hashcat con potfile	12
4.2. identifica nomenclatura del output	12
4.3. obtiene contraseña con hashcat sin potfile	12
4.4. identifica nomenclatura del output	12
4.5. obtiene contraseña con aircrack-ng	12
4.6. identifica y modifica parámetros solicitados por pycrack	12
4.7. obtiene contraseña con pycrack	12

1. Descripción de actividades

Su informante quiere entregarle la contraseña de acceso a una red, pero desconfía de todo medio para entregársela (aún no llega al capítulo del curso en donde aprende a comunicar una password sin que nadie más la pueda interceptar). Por lo tanto, le entregará un archivo que contiene un desafío de autenticación, que al analizarlo, usted podrá obtener la contraseña que lo permite resolver. Como nadie puede ver a su informante (es informante y debe mantener el anonimato), él se comunicará con usted a través de la redes inalámbricas y de una forma que solo usted, como experto en informática y telecomunicaciones, logrará esclarecer.

1. Identifique cual es la red inalámbrica que está utilizando su informante para enviarle información. Obtenga la contraseña de esa red utilizando el ataque por defecto de aircrack-ng, indicando el tiempo requerido para esto. Descifre el contenido transmitido sobre ella y descargue de Internet el archivo que su informante le ha comunicado a través de los paquetes que usted ha descifrado.
2. Descargue el diccionario de RockyouLinks to an external site. (utilizado ampliamente en el mundo del pentesting). Haga un script que para cada string contenido en el diccionario, reemplace la primera letra por su letra en capital y agregue un cero al final de la password.
3. Todos los strings que comiencen con número toca eliminarlos del diccionario. Indique la cantidad de contraseñas que contiene el diccionario modificado debe llamarse rock-you_mod.dic A continuación un ejemplo de cómo se modifican las 10 primeras líneas del diccionario original.

2. Desarrollo (PASO 1)

2.1. identificar en qué se destaca la red del informante del resto

En esta actividad, es necesario encontrar la red inalámbrica del informante con el fin de tener acceso a la información que está intentando compartir. Para ello, se utiliza un adaptador WiFi que permite al computador acceder al modo monitor. En este modo, el computador es capaz de capturar paquetes WiFi de todo tipo, incluidos los que transmite el informante para comunicar la información. A continuación, se utiliza el comando `iwconfig` para encontrar el ID del adaptador WiFi, valor que será necesario para acceder al modo monitor y proceder con el resto de los pasos.

```
(base) informatica@informatica-11:~$ iwconfig
lo          no wireless extensions.

eno1       no wireless extensions.

wlx1027f5519918 unassociated  Nickname:"<WIFI@REALTEK>"
               Mode:Monitor  Frequency=2.452 GHz  Access Point: Not-Associated
               Sensitivity:0/0
               Retry:off   RTS thr:off   Fragment thr:off
               Power Management:off
               Link Quality=0/100  Signal level=0 dBm  Noise level=0 dBm
               Rx invalid nwid:0  Rx invalid crypt:0  Rx invalid frag:0
               Tx excessive retries:0  Invalid misc:0  Missed beacon:0

virbr0     no wireless extensions.

docker0    no wireless extensions.
```

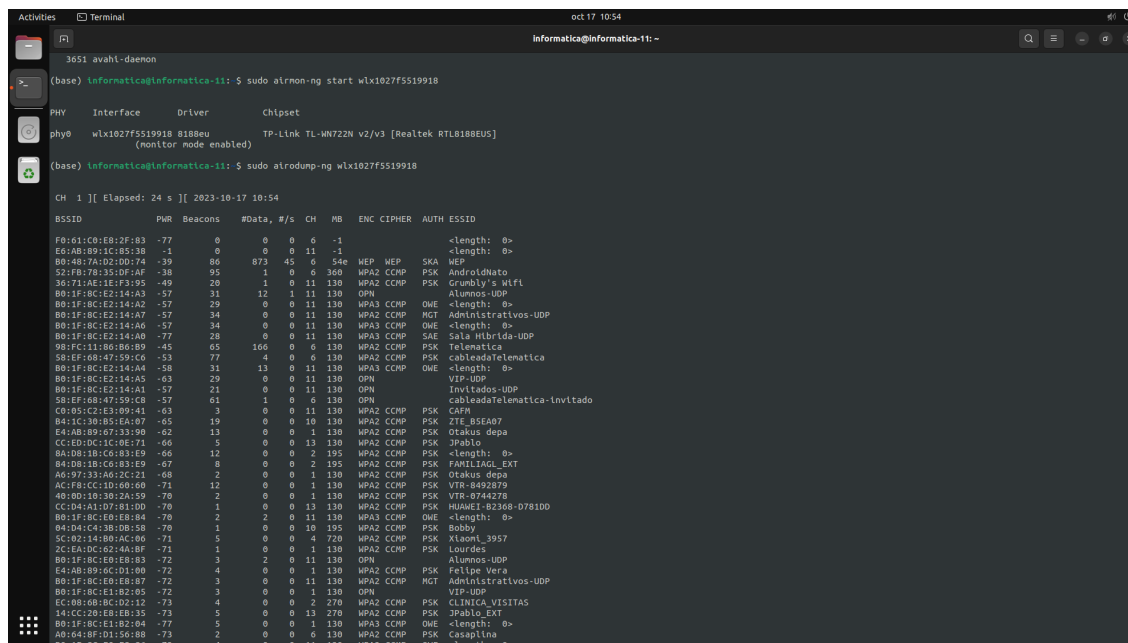
Figura 1: Comando `iwconfig` para encontrar el ID del adaptador WiFi

En la imagen, se puede apreciar un ID que comienza con los caracteres 'wlx', dicho valor corresponde al ID necesario para acceder al modo monitor en el dispositivo. A continuación se utiliza el comando `sudo airmon-ng start [ID]` con el fin de acceder al modo monitor mediante la herramienta `airmon-ng`. En la siguiente imagen, se puede apreciar el resultado obtenido al ejecutar la línea de comando anterior.

Se puede observar en la imagen que el comando creó una nueva interfaz en modo monitor con el ID del adaptador WiFi, lo cual permite al computador leer los paquetes WiFi que está transmitiendo desde su red inalámbrica. Sin embargo, no se sabe la dirección IPv6 correspondiente a dicha red, por lo que se debe utilizar el comando `sudo airodump-ng [ID]` para identificar cuál red inalámbrica está transmitiendo información de forma periódica. El comando en ejecución también se puede apreciar en la imagen anterior. Este comando muestra una larga lista de puntos de acceso detectados por el dispositivo de trabajo, junto a la lista de clientes conectados. Entre todas las conexiones detectadas, se puede observar que una de las redes posee cualidades únicas que la diferencian del resto, tales como en los apartados ENC, CIPHER y ESSID de la lista de puntos de acceso, donde los apartados comparten el nombre WEP. WEP es un algoritmo de seguridad utilizado para asegurar la confidencialidad de la información que se transmite. En la misma fila, también se observa que la red

2.2 explica matemáticamente porqué se requieren más de 5000 paquetes para obtener la pass

2 DESARROLLO (PASO 1)



```
3651 avahd-daemon
(base) Informatica@Informatica-11: ~$ sudo airmon-ng start wlx1027f5519918
PHY Interface Driver Chipset
phy0 wlx1027f5519918 8188eu TP-Link TL-WN722N v2/v3 [Realtek RTL8188EUS]
(nonitor mode enabled)
(base) Informatica@Informatica-11: ~$ sudo airodump-ng wlx1027f5519918
CH 1 [( Elapsed: 24 s )] [ 2023-10-17 10:54
BSSID PWR Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
F0:61:C0:E8:2F:83 -77 0 0 0 6 -1 <length: 0>
E6:AB:89:1C:85:38 -1 0 0 0 11 -1 <length: 0>
B0:48:7A:D2:DD:74 -39 86 873 45 0 54e WEP WEP SKA WEP
32:28:70:25:0F:A0 -30 95 1 0 0 360 WPA2 CCMP PSK Androldato
36:71:AE:1E:F3:95 -49 20 1 0 11 130 WPA2 CCMP PSK Grumbly's Wlfi
B0:1F:8C:E2:14:A3 -57 31 12 1 11 130 WPA2 CCMP PSK Alumnos-UDP
B0:1F:8C:E2:14:A2 -57 29 0 0 11 130 WPA2 CCMP PSK <length: 0>
B0:1F:8C:E2:14:A7 -57 34 0 0 11 130 WPA2 CCMP PSK Administrativos-UDP
B0:1F:8C:E2:14:A0 -57 34 0 0 11 130 WPA2 CCMP PSK <length: 0>
B0:1F:8C:E2:14:A0 -77 28 0 0 11 130 WPA2 CCMP PSK Sala Híbrida-UDP
98:FC:11:80:80:89 -45 65 166 0 0 130 WPA2 CCMP PSK Telematica
58:EF:68:47:59:C6 -53 77 4 0 0 130 WPA2 CCMP PSK cabledataTelematica
B0:1F:8C:E2:14:A4 -58 31 13 0 11 130 WPA2 CCMP PSK <length: 0>
B0:1F:8C:E2:14:A5 -63 29 0 0 11 130 WPA2 CCMP PSK VIP-UDP
B0:1F:8C:E2:14:A1 -57 21 0 0 11 130 WPA2 CCMP PSK Invitados-UDP
58:EF:68:47:59:C6 -57 61 1 0 0 130 WPA2 CCMP PSK cabledataTelematica-Invitado
C0:85:C2:E3:89:41 -63 3 0 0 11 130 WPA2 CCMP PSK CAFM
B4:1C:30:B5:EA:07 -65 19 0 0 10 130 WPA2 CCMP PSK ZTE_B5EA07
E4:AB:89:07:33:50 -62 13 0 0 1 130 WPA2 CCMP PSK Otakus depa
CC:ED:DC:1C:0E:71 -66 5 0 0 13 130 WPA2 CCMP PSK JPablo
BA:08:1B:C6:83:E9 -66 12 0 0 2 195 WPA2 CCMP PSK <length: 0>
84:0B:18:C6:83:E9 -67 8 0 0 2 195 WPA2 CCMP PSK FAHILLAD_EXT
A6:97:33:A6:2C:21 -68 2 0 0 1 130 WPA2 CCMP PSK Otakus depa
AC:F8:CC:1D:60:60 -71 12 0 0 1 130 WPA2 CCMP PSK VTR-8492879
A0:0D:10:10:2A:59 -70 1 0 0 1 130 WPA2 CCMP PSK VTR-8744278
CC:D4:A1:D7:B1:00 -70 1 0 0 13 130 WPA2 CCMP PSK HUAMEI-B2368-D78100
B0:1F:8C:E0:E8:84 -70 2 2 0 11 130 WPA2 CCMP PSK <length: 0>
04:04:C4:10:00:10 -70 1 0 0 10 195 WPA2 CCMP PSK Bobby
5C:D2:14:80:AC:06 -71 5 0 0 4 720 WPA2 CCMP PSK Xlaoni_3957
2C:EA:DC:D2:AA:BF -71 1 0 0 1 130 WPA2 CCMP PSK Lourdes
B0:1F:8C:E0:E8:83 -72 3 2 0 11 130 WPA2 CCMP PSK Alumnos-UDP
E4:AB:89:0C:D1:00 -72 4 0 0 1 130 WPA2 CCMP PSK Felipe Vera
B0:1F:8C:E0:E8:87 -72 3 0 0 11 130 WPA2 CCMP PSK Administrativos-UDP
B0:1F:8C:E1:82:05 -72 3 0 0 1 130 WPA2 CCMP PSK VIP-UDP
EC:88:88:8C:D2:12 -73 4 0 0 2 270 WPA2 CCMP PSK CLINICA_VISITAS
14:CC:20:E8:EB:35 -73 5 0 0 13 270 WPA2 CCMP PSK JPablo_EXT
B0:1F:8C:E1:82:04 -77 5 0 0 1 130 WPA2 CCMP PSK <length: 0>
A0:04:8F:D1:56:88 -73 2 0 0 0 130 WPA2 CCMP PSK Casaplina
B0:1F:8C:E0:E8:86 -73 4 0 0 11 130 WPA2 CCMP PSK <length: 0>
```

Figura 2: Comando airmon-ng para acceder al modo monitor y comando airodump-ng para mostrar la lista de puntos de acceso

inalámbrica posee una autenticación AUTH de tipo SKA, o Shared Key Authentication, un proceso de verificación en el que un computador gana acceso a una conexión inalámbrica que usa WEP, permitiendo a aquellos dispositivos con un modem inalámbrico tener acceso completo a cualquier tipo de conexión WEP. Con esta información, es posible identificar que la red inalámbrica con BSSID B0:48:7A:D2:DD:74 corresponde a la dirección del informante, por lo que se utiliza este valor para proceder con la captura de tráfico.

2.2. explica matemáticamente porqué se requieren más de 5000 paquetes para obtener la pass

Luego de haber identificado la red inalámbrica del informante, es posible ahora interceptar el tráfico de su red. Dicha acción se puede lograr utilizando el comando "sudo airodump-ng -c X -bssid B0:48:7A:D2:DD:74 -w captura [ID]". Con -c, se indica el número del canal donde se encuentra la red inalámbrica transmitiendo, -w es el parámetro utilizado para nombrar el archivo en el que se guardarán los datos interceptados, y el parámetro -bssid indica la dirección MAC de la red inalámbrica. A continuación se presenta la ejecución de la línea de comando para interceptar el tráfico de la red:

En la imagen se puede observar que se capturaron más de 80000 paquetes WiFi para asegurar que la captura de tráfico pueda encontrar la password. Dicho número se debe a que cada paquete posee un IV, un bloque de bits esencial para permitir el cifrado de flujo. Sin embargo IVs pueden reutilizarse, lo que significa que la cantidad de IVs distintos es menor a la cantidad de paquetes capturados.

2.3 obtiene la password con ataque por defecto de aircrack-ng DESARROLLO (PASO 1)

```
quitting...
(base) informatica@informatica-11:~$ sudo airodump-ng -c 6 --bssid B0:48:7A:D2:DD:74 -w captura wlx1027f5519918
11:00:38 Created capture file "captura-01.cap".

CH 6 ][ Elapsed: 4 mins ][ 2023-10-17 11:04

BSSID          PWR RXQ Beacons  #Data, #/s CH  MB  ENC CIPHER AUTH ESSID
BSSID          STATION    PWR  Rate  Lost  Frames  Notes  ProbesQuitting...

B0:48:7A:D2:DD:74 8A:0B:DA:32:CC:82 -38  54e- 6e    0    408
B0:48:7A:D2:DD:74 B8:27:EB:35:AB:17 -42  54e-54e  800  82822
(base) informatica@informatica-11:~$
```

Figura 3: Comando para interceptar el tráfico de la red inalámbrica

2.3. obtiene la password con ataque por defecto de aircrack-ng

Una vez capturada una cantidad suficiente de paquetes en el archivo CAP, se debe utilizar dicho archivo para determinar cuál es la contraseña que se utilizó para cifrar el tráfico emitido por la red inalámbrica del informante. Para ello, se utiliza el comando "sudo aircrack-ng -b [ID] captura-0X.cap", donde el parámetro -b corresponde al BSSID de la red inalámbrica interceptada y captura-0X.cap corresponde al archivo que contiene los paquetes, junto a la clave potencialmente utilizada para cifrar el tráfico.

```
(base) informatica@informatica-11:~$ sudo aircrack-ng -b B0:48:7A:D2:DD:74 captura-04.cap
Reading packets, please wait...
Opening captura-04.cap
Read 152124 packets.

1 potential targets
KEY FOUND! [ 12:34:56:78:90 ]
Attack wDecrypted correctly: 100%00 captured ivs.
Got 69689 out of 65000 IVsStarting PTW attack with 69689 ivs.

(base) informatica@informatica-11:~$
```

Figura 4: Comando para obtener la clave utilizada para cifrar el tráfico de paquetes

En la imagen anterior, se puede notar que el comando, luego de leer todos los paquetes almacenados en el archivo, logró encontrar el paquete con la llave de encriptación. Se puede observar que la llave de encriptación corresponde a "12:34:56:78:90".

2.4. indica el tiempo que demoró en obtener la password

2.5. descifra el contenido capturado

Luego de obtener la clave en el comando anterior, se utiliza dicho valor para descryptar la captura obtenida. Dicha acción se puede completar con el comando "sudo airdecap-ng -w 12:34:56:78:90 captura-0X.cap", donde -w es el parámetro donde se inserta la clave para descryptar el archivo y captura-0X.cap corresponde al archivo al cuál se va a aplicar la clave de descifrado.

```
(base) informatica@informatica-11:~$ sudo airdecap-ng -w 12:34:56:78:90 captura-04.cap
Total number of stations seen          9
Total number of packets read          152124
Total number of WEP data packets       69760
Total number of WPA data packets       0
Number of plaintext data packets       0
Number of decrypted WEP packets        69760
Number of corrupted WEP packets        0
Number of decrypted WPA packets        0
Number of bad TKIP (WPA) packets       0
Number of bad CCMP (WPA) packets       0
(base) informatica@informatica-11:~$
```

Figura 5: Comando para descifrar la captura de tráfico

La imagen muestra el procedimiento realizado por el comando para descifrar la captura de red obtenida, donde se detectaron más de 69000 paquetes de datos WEP, y se logró descifrar cada uno de estos paquetes sin encontrar ninguna falla en el proceso.

2.6. describe como obtiene la url de donde descargar el archivo

Una vez descifrados los paquetes de la captura, es posible observar la información que transmitió el informante por medio de su red inalámbrica. Para ello, se accede a la captura mediante la herramienta Wireshark, y se puede observar el siguiente set de paquetes.

2.6 describe como obtiene la url de donde descargar el archivo DESARROLLO (PASO 1)

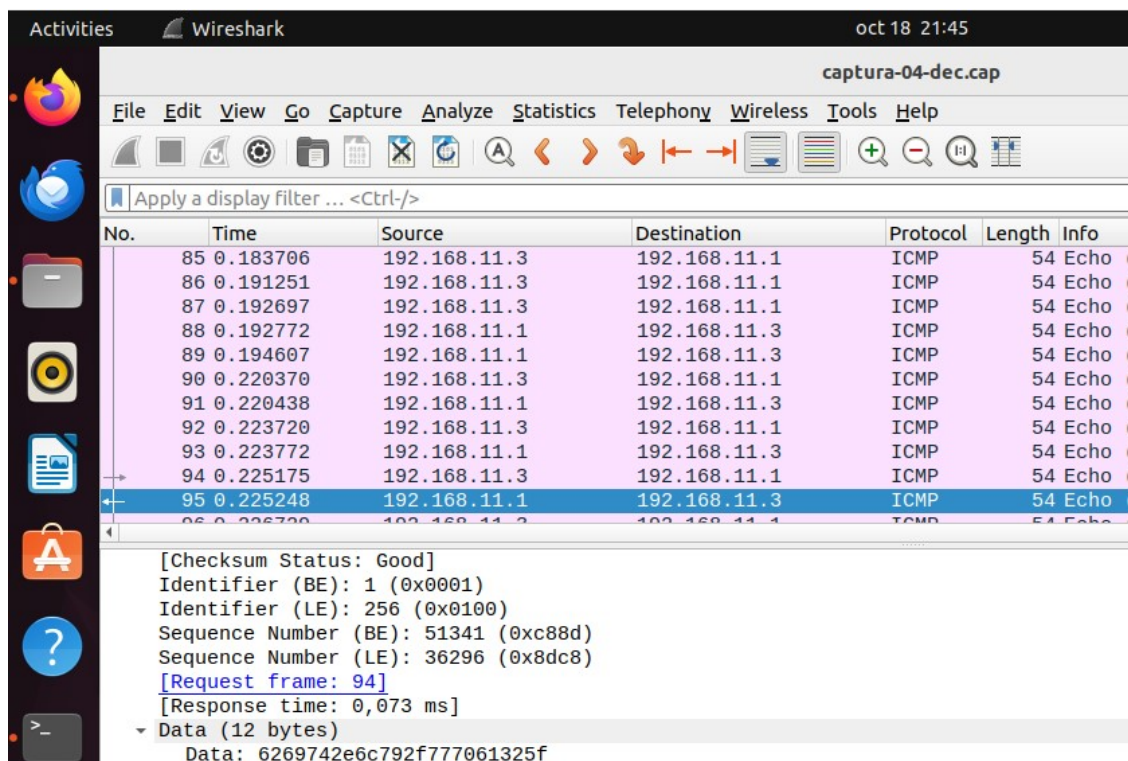


Figura 6: Lista de paquetes descifrados

En la imagen se observa que todos los paquetes corresponden al protocolo ICMP que se intercambian entre las direcciones IP 192.168.11.1 y 192.168.11.3 de manera constante. Sin embargo, la información más importante de estos paquetes se encuentra en la sección Data de cada uno de ellos, esto se puede ver con mejor claridad en la siguiente imagen:

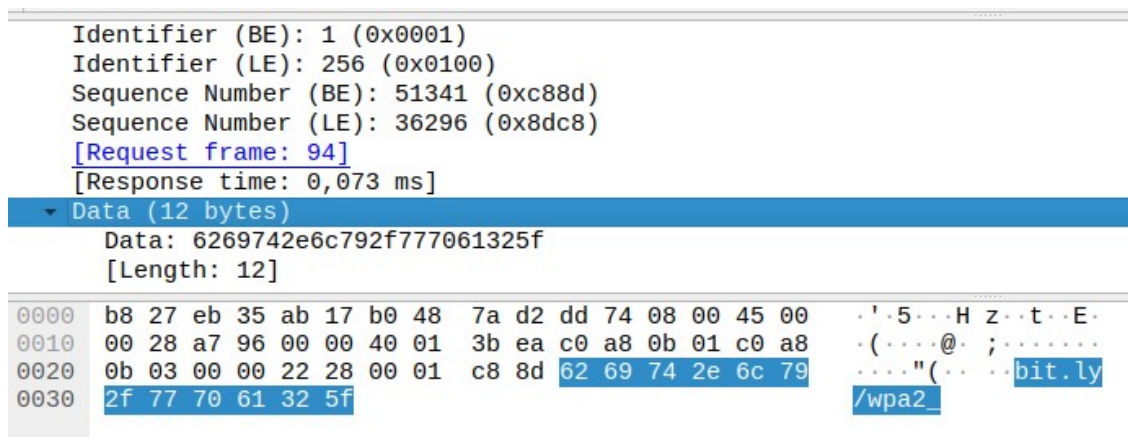


Figura 7: Sección Data de los paquetes

Observando la sección Data del paquete, es posible notar que aquellos bytes en hexadecimal corresponden a un enlace, siendo este "bit.ly/wpa2_". Si se intenta acceder a este enlace por internet, se redirige a una página web que contiene una captura de red en Cloudshark llamada "handshake.pcap". La captura se puede ver con mejor claridad en la siguiente imagen.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	ee:de:67:8c:df:8b	Tp-LinkT_d2:dc:18	802.11	123	Association Request, SN=2292, FN=0, Flags=....., SSID=VTR-164
2	0.000002	ee:de:67:8c:df:8b	ee:de:67:8c:df:8b (ee:de:67:8c:df:8b) (RA)	802.11	10	Acknowledgement, Flags=.....
3	0.002401	Tp-LinkT_d2:dc:18	ee:de:67:8c:df:8b	802.11	102	Association Response, SN=1184, FN=0, Flags=.....
4	0.002402	Tp-LinkT_d2:dc:18	ee:de:67:8c:df:8b (b0:48:7a:d2:dc:18) (RA)	802.11	10	Acknowledgement, Flags=.....
5	0.007381	Tp-LinkT_d2:dc:18	ee:de:67:8c:df:8b	EAPOL	133	Key (Message 1 of 4)
6	0.009336	Tp-LinkT_d2:dc:18	ee:de:67:8c:df:8b (b0:48:7a:d2:dc:18) (RA)	802.11	10	Acknowledgement, Flags=.....
7	0.017080	ee:de:67:8c:df:8b	Tp-LinkT_d2:dc:18	EAPOL	155	Key (Message 2 of 4)
8	0.017082	ee:de:67:8c:df:8b	ee:de:67:8c:df:8b (ee:de:67:8c:df:8b) (RA)	802.11	10	Acknowledgement, Flags=.....
9	0.017087	ee:de:67:8c:df:8b	ee:de:67:8c:df:8b (ee:de:67:8c:df:8b) (RA)	802.11	10	Clear-to-send, Flags=.....
10	0.050774	Tp-LinkT_d2:dc:18	ee:de:67:8c:df:8b	EAPOL	189	Key (Message 3 of 4)
11	0.050776	Tp-LinkT_d2:dc:18	ee:de:67:8c:df:8b (b0:48:7a:d2:dc:18) (RA)	802.11	10	Acknowledgement, Flags=.....

Frame 1: 123 bytes on wire (984 bits), 123 bytes captured (984 bits)
 IEEE 802.11 Association Request, Flags:
 IEEE 802.11 Wireless Management

```

0000 00 00 3a 01 b0 48 7a d2 dc 18 ee de 67 8c df 8b  ...Hz....8...
0010 b0 48 7a d2 dc 18 40 8f 31 04 01 00 00 0b 56 54  .Hz...8.1....VT
0020 52 2d 31 36 34 35 32 31 33 01 08 82 84 8b 96 0c  R-1645213.....
0030 12 18 24 30 14 01 00 00 0f ac 04 01 00 00 0f ac  ..$0.....
0040 04 01 00 00 0f ac 02 00 00 32 04 30 48 60 6c 3b  ....2.0H'l;
0050 10 51 51 53 54 73 74 75 76 77 78 7c 7d 7e 7f 80  .QqStstuvwx|}~..
0060 82 7f 05 04 00 00 00 01 dd 07 00 50 f2 02 00 01  ....P....
0070 00 dd 08 8c fd f0 01 01 02 01 00  ....

```

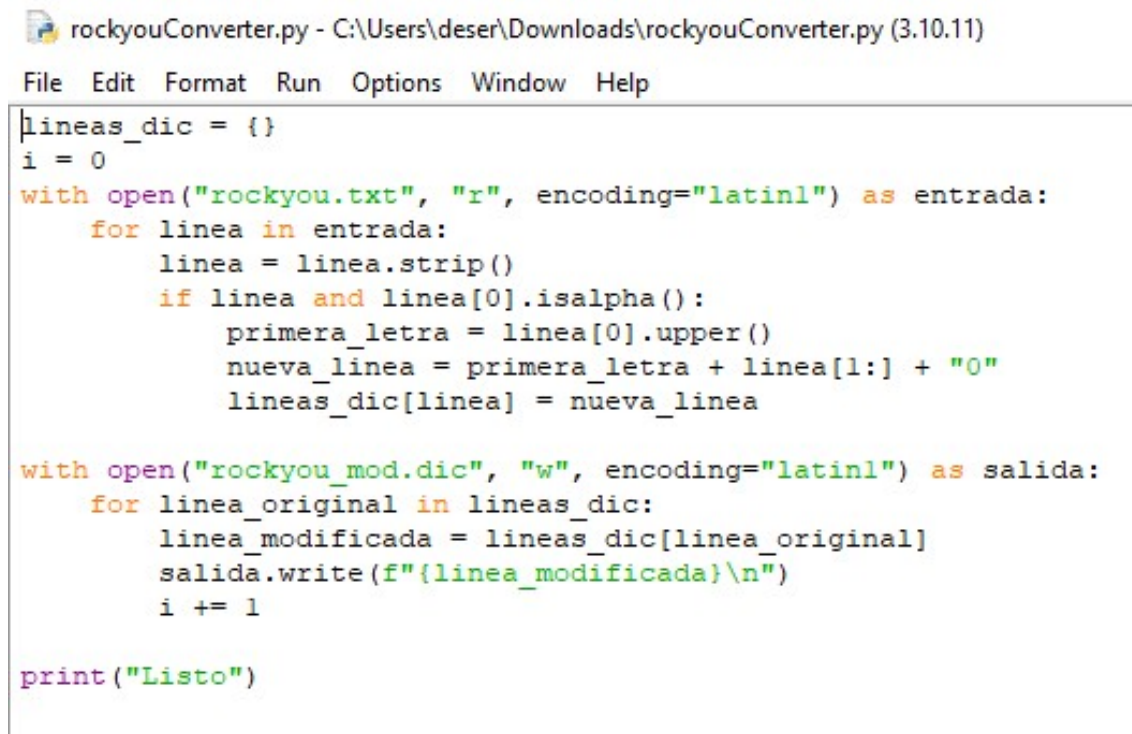
Figura 8: Captura en Cloudshark accedida mediante la captura anterior

Este archivo es esencial para la sección de ataques de fuerza bruta de este laboratorio, por lo que se debe descargar.

3. Desarrollo (PASO 2)

3.1. indica script para modificar diccionario original

Para esta sección, hay que crear un código en Python que permita interactuar con el diccionario rockyou.txt y realizar las modificaciones necesarias, tales como descartar las contraseñas que comiencen por un número, o la modificación de contraseñas que comiencen en una letra, de tal forma que su caracter inicial sea mayúscula y que su dígito final sea un 0. Para ello, se utiliza el comando open para abrir y leer el archivo.txt línea por línea. En esta sección se detecta si la línea en la que se encuentra comienza con una letra, y si es así, se procede a cambiar la primera letra del string por su versión en mayúscula, además de adicionar el caracter 0 al final del string, para luego almacenarlo dentro de un diccionario. En la segunda parte del código se realiza la escritura en un nuevo archivo DIC llamado "rockyou_mod", el cual almacena las nuevas contraseñas que salieron de la primera parte del código.

The image shows a screenshot of a Python script editor window titled "rockyouConverter.py - C:\Users\deser\Downloads\rockyouConverter.py (3.10.11)". The menu bar includes "File", "Edit", "Format", "Run", "Options", "Window", and "Help". The script code is as follows:

```
lineas_dic = {}  
i = 0  
with open("rockyou.txt", "r", encoding="latin1") as entrada:  
    for linea in entrada:  
        linea = linea.strip()  
        if linea and linea[0].isalpha():  
            primera_letra = linea[0].upper()  
            nueva_linea = primera_letra + linea[1:] + "0"  
            lineas_dic[linea] = nueva_linea  
  
with open("rockyou_mod.dic", "w", encoding="latin1") as salida:  
    for linea_original in lineas_dic:  
        linea_modificada = lineas_dic[linea_original]  
        salida.write(f"{linea_modificada}\n")  
    i += 1  
  
print("Listo")
```

Figura 9: Captura en Cloudshark accedida mediante la captura anterior

Luego de ejecutar el código, es posible encontrar un nuevo archivo en la carpeta donde se ubica el script, el cual se llama "rockyou_mod.dic", confirmando que la ejecución funciona sin problemas.

3.2. cantidad de passwords finales que contiene rockyou_mod.dic

Si se observan los primeros 10 datos del diccionario original junto al diccionario con modificaciones, es posible notar que el código realizó los cambios propuestos. Una de las principales diferencias que se pueden notar es el hecho de que en el diccionario original, existen 5 contraseñas compuestas exclusivamente de números, las cuales se encuentran completamente ausentes en el diccionario modificado. Por otro lado, aquellas contraseñas del diccionario anterior que si aparecen en el nuevo diccionario ahora comienzan en letra mayúscula y terminan en un 0.

```
waluigi@waluigi-VirtualBox:~/Downloads$ head -10 rockyou.txt
123456
12345
123456789
password
iloveyou
princess
1234567
rockyou
12345678
abc123
waluigi@waluigi-VirtualBox:~/Downloads$
```

Figura 10: Diccionario rockyou original

```
waluigi@waluigi-VirtualBox:~/Downloads$ head -10 rockyou_mod.dic
Password0
Iloveyou0
Princess0
Rockyou0
Abc1230
Nicole0
Daniel0
Babygirl0
Monkey0
Lovely0
waluigi@waluigi-VirtualBox:~/Downloads$
```

Figura 11: Diccionario rockyou modificado

Si se observa el nuevo archivo "rockyou_mod.dic" dentro de un editor de texto, es posible observar que el diccionario modificado termina con una cantidad de 10962619 contraseñas distintas, reduciendo su tamaño en 20MB en comparación al archivo rockyou original.

3.2 cantidad de passwords finales que contiene rockyou_mod3licDESARROLLO (PASO 2)

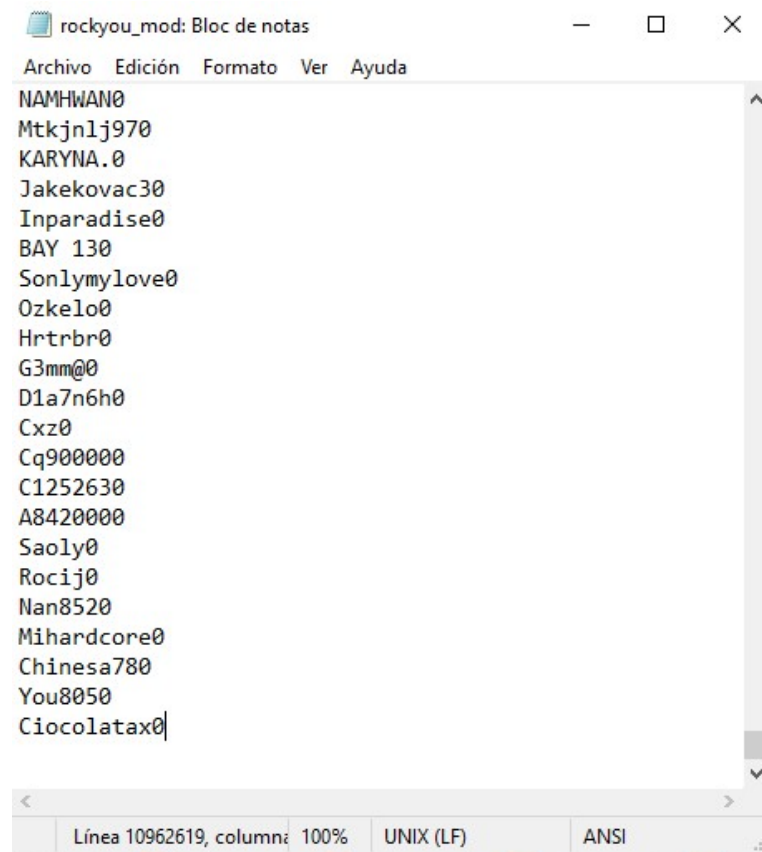


Figura 12: Numero de passwords

4. Desarrollo (Paso 3)

- 4.1. obtiene contraseña con hashcat con potfile**
- 4.2. identifica nomenclatura del output**
- 4.3. obtiene contraseña con hashcat sin potfile**
- 4.4. identifica nomenclatura del output**
- 4.5. obtiene contraseña con aircrack-ng**
- 4.6. identifica y modifica parámetros solicitados por pycrack**
- 4.7. obtiene contraseña con pycrack**

Conclusiones y comentarios

En este laboratorio, se tuvo que trabajar en la identificación e interceptación de tráfico de una red inalámbrica, el descifrado de los paquetes interceptados, la modificación de las contraseñas de un diccionario y la realización de varios ataques de fuerza bruta mediante las herramientas hashcat, aircrack-ng y pycrack con el objetivo de encontrar una contraseña que un informante trata de compartir por medio de su red inalámbrica.

Las actividades de laboratorio evidenciaron que si bien el uso de la estrategia del informante para encriptar su tráfico de red para compartirla con un oyente puede ser útil para evitar que una persona externa pueda tener un acceso simple a la información que trata compartirse, basta que una persona tenga un adaptador WiFi para ser tan capaz como el receptor de interceptar la información y utilizar las herramientas de esta actividad para descifrar y obtener la contraseña por ataques de fuerza bruta.