

Informe Laboratorio 3

Sección 1

Valentina Díaz Gonzalez
e-mail: valentina.diaz_go@mail.udp.cl

Mayo de 2023

Índice

1. Descripción de actividades	2
2. Desarrollo (PASO 1)	2
2.1. identificar en qué se destaca la red del informante del resto	2
2.2. explica matemáticamente porqué se requieren más de 5000 paquetes para obtener la pass	3
2.3. obtiene la password con ataque por defecto de aircrack-ng	4
2.4. indica el tiempo que demoró en obtener la password	4
2.5. descifra el contenido capturado	5
2.6. describe como obtiene la url de donde descargar el archivo	6
3. Desarrollo (PASO 2)	6
3.1. indica script para modificar diccionario original	6
3.2. cantidad de passwords finales que contiene rockyou_mod.dic	7
4. Desarrollo (Paso 3)	7
4.1. obtiene contraseña con hashcat con potfile	7
4.2. identifica nomenclatura del output	8
4.3. obtiene contraseña con hashcat sin potfile	9
4.4. identifica nomenclatura del output	10
4.5. obtiene contraseña con aircrack-ng	10
4.6. identifica y modifica parámetros solicitados por pycrack	10
4.7. obtiene contraseña con pycrack	13

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)

El primer paso para realizar el laboratorio es obtener el id de la tarjeta de red para poder hacer uso de ella en modo monitor, para ello se ocupa el comando *iwconfig*. Como consecuencia se obtiene que el id de la tarjeta de red es *wlx1027f5518665*. Con este id, se procede a cambiar el modo de la tarjeta de red a Monitor, el comando a utilizar es el siguiente: *sudo airmon-ng start wxl1027f5518665*.

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

Para detectar redes se utiliza el siguiente comando *sudo airodump-ng wxl1027f5518665*, se obtiene la siguiente captura:

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

BSSID	PWR	Beacons	#Data	#/s	CH	MB	ENC	CIPHER	AUTH	ESSID
E6:AB:89:1C:85:38	-1	0	0	0	11	-1				<length: 0>
78:29:ED:AF:C6:D7	-1	0	10	0	11	-1	WPA			<length: 0>
B0:1F:8C:E2:14:A4	-62	2	0	0	11	130	WPA3	CCMP	OWE	<length: 0>
B0:1F:8C:E2:14:A2	-59	4	0	0	11	130	WPA3	CCMP	OWE	<length: 0>
B0:1F:8C:E2:14:A0	-61	4	0	0	11	130	WPA3	CCMP	SAE	Sala Hibrida-UDP ^
CCC:ED:DC:1C:0E:71	-67	3	0	0	8	130	WPA2	CCMP	PSK	JPablo
CC:ED:DC:1C:0E:71	-67	3	0	0	8	130	WPA2	CCMP	PSK	JPablo
Quitting...1:B2:01	-80	3	0	0	1	130	OPN			Invitados-UDP
84:D8:1B:C6:83:E9	-66	12	0	0	2	195	WPA2	CCMP	PSK	FAMILIAGL_EXT
B0:1F:8C:E2:14:A1	-80	4	0	0	1	130	OPN			Invitados-UDP
B0:48:7A:D2:DD:74	-34	13	15	7	3	54e	WEP	WEP		WEP
58:EF:68:47:59:C8	-63	6	0	0	6	130	OPN			cableadaTelematic
58:EF:68:47:59:C6	-63	4	1	0	6	130	WPA2	CCMP	PSK	cableadaTelematic
98:FC:11:86:B6:B9	-44	7	8	0	6	130	WPA2	CCMP	PSK	Telematica
20:AA:4B:31:A2:D4	-75	4	0	0	1	130	WPA2	CCMP	PSK	OF-CCFI
18:35:D1:48:E8:39	-77	2	0	0	1	130	WPA2	CCMP	PSK	VTR-5376275
C0:05:C2:E3:09:41	-73	5	0	0	1	130	WPA2	CCMP	PSK	CAFM
8A:D8:1B:C6:83:E9	-67	9	0	0	2	195	WPA2	CCMP	PSK	<length: 0>
B0:1F:8C:E2:14:A5	-62	3	0	0	11	130	OPN			VIP-UDP
B0:1F:8C:E2:14:A1	-64	4	0	0	11	130	OPN			Invitados-UDP

Figura 1: Redes obtenidas con Airodump

La red del informante se destaca en ocupar el protocolo de seguridad para redes Wi-Fi WEP, el cual ya no se utiliza puesto que se considera muy inseguro por las vulnerabilidades que presenta.

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

Para poder obtener la llave de cifrado se utiliza el método "Birthday Attack", que consiste en encontrar la llave buscando una colisión entre alguno de los paquetes capturados. La explicación matemática se debe a la paradoja de los cumpleaños, este indica que si se tienen 23 personas en una misma habitación existe un 50 % de probabilidad de que dos personas tengan la misma fecha de cumpleaños, si se tienen 50 personas esta probabilidad es mayor al 97 %.

La fórmula de utilizada para calcular la probabilidad es:

$$Probabilidad = 1 - \prod_{k=1}^{y-1} \left(1 - \frac{k}{n}\right) \quad (1)$$

Donde n representa el número de combinaciones posibles, que en el contexto de cumpleaños es 365. Por lo tanto, si se tienen 50 personas el resultado es el siguiente:

$$Probabilidad = 1 - \prod_{k=1}^{50-1} \left(1 - \frac{k}{365}\right) = 97 \% \quad (2)$$

Ahora bien, si lo adoptamos a nuestro escenario la cantidad de elementos pasa a ser 5000 y las combinaciones posibles es 2^{24} . Por lo tanto, el resultado es:

$$Probabilidad = 1 - \prod_{k=1}^{5000-1} \left(1 - \frac{k}{2^{24}}\right) = 99,7 \% \quad (3)$$

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

Por lo tanto, con más de 5000 se obtiene una probabilidad de encontrar una colisión cercana al 100 %

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

Para poder obtener la contraseña se deben capturar paquetes antes. Para ello, se utilizan dos datos del paso anterior, el BSSID de la señal y el canal en que está operando. El BSSID es *B0:48:7A:D2:DD:72* y el canal es el 3. Ahora, se utiliza el comando *sudo airodump-ng -c 3 -bssid B0:48:7A:D2:DD:72 -w captura wlx1027f5518665*.

```
(base) Informatica@Informatica-10:~$ sudo airodump-ng -c 3 --bssid B0:48:7A:D2:DD:74 -w captura wlx1027f5518665
09:07:06 Created capture file "captura-07.cap".

CH 3 ][ Elapsed: 15 mins ][ 2023-10-20 09:22

BSSID            PWR RXQ Beacons   #Data, #/s CH  MB  ENC CIPHER AUTH ESSID
B0:48:7A:D2:DD:74 -41  82      7449   622345 1021  3   54e WEP  WEP    WEP
BSSID            STATION          PWR   Rate    Lost    Frames  Notes  Probes
B0:48:7A:D2:DD:74 E0:0A:F6:3C:E0:91 -42   54e-54e 1620   636532
```

Figura 2: Captura de paquetes con Airodump

Mientras se realizaba la captura de paquetes, se utilizó el siguiente comando *time aircrack-ng -b B0:48:7A:D2:DD:72 captura-07.cap*

```
(base) Informatica@Informatica-10:~$ time aircrack-ng -b B0:48:7A:D2:DD:74 captura-07.cap
Reading packets, please wait...
Opening captura-07.cap
Read 1493667 packets.

1 potential targets          Got 670703 out of 670000 IVsStarting PTW attack with 670703 iv
s.
KEY FOUND! [ 12:34:56:78:90 ]
Decrypted correctly: 100%

real    0m2,007s
user    0m1,945s
sys     0m0,044s
(base) Informatica@Informatica-10:~$
```

Figura 3: Llave encontrada

La llave encontrada es [12:34:56:78:90].

2.4. indica el tiempo que demoró en obtener la password

En el comando del paso anterior se antepuso la palabra *time*, con ella se obtuvo el tiempo que demoró paso anterior en encontrar la contraseña. El tiempo fue de 2,007 segundos.

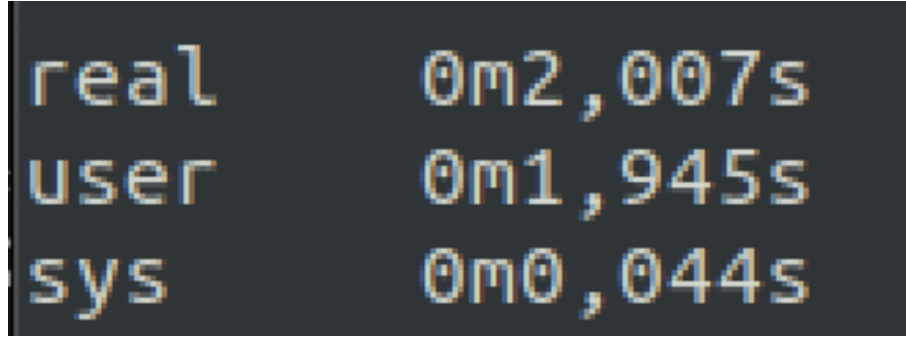


Figura 4: Tiempo

2.5. descifra el contenido capturado

El archivo encriptado se ve de la siguiente manera:

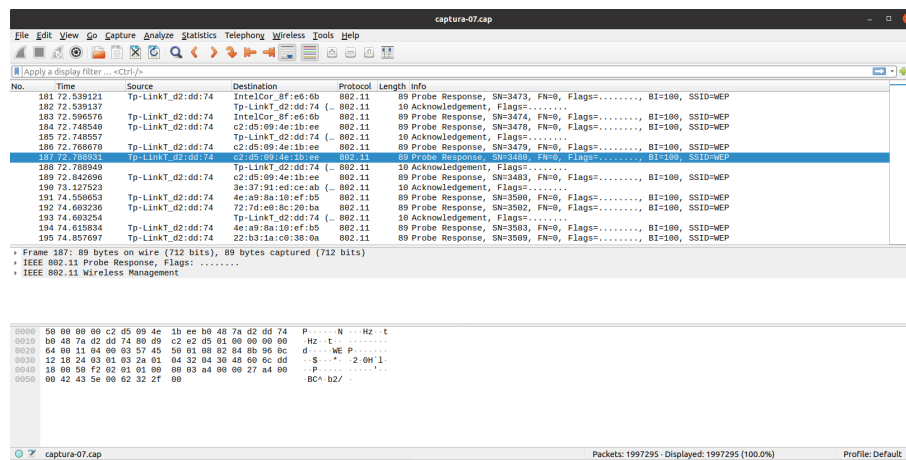


Figura 5: Captura descifrada

Para poder descifrar el contenido de la captura se ocupa el siguiente comando *airdecap-ng -w 12:34:56:78:90 captura-07.cap*

2.6 describe como obtiene la url de donde descargar el archivo DESARROLLO (PASO 2)

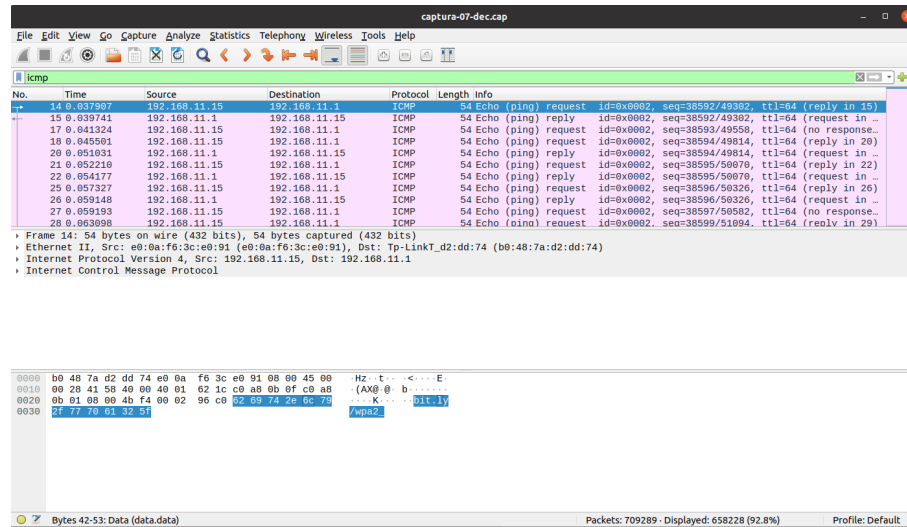


Figura 6: Captura descifrada con la llave obtenida

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

A partir de la captura descifrada, se ve que en los paquetes ICMP en el campo Data se obtiene el siguiente url `bit.ly/wpa2_`.

```
b0 48 7a d2 dd 74 e0 0a f6 3c e0 91 08 00 45 00  .Hz..t..<....E.
00 28 41 58 40 00 40 01 62 1c c0 a8 0b 0f c0 a8  .(AX@.@.b.....
0b 01 08 00 4b f4 00 02 96 c0 62 69 74 2e 6c 79  ....K...:bit.ly
2f 77 70 61 32 5f                                /wpa2_
```

Figura 7: URL de donde descargar archivo

3. Desarrollo (PASO 2)

3.1. indica script para modificar diccionario original

El script realizado procesa un archivo de contraseñas ('rockyou.txt'). Para cada contraseña en el archivo se realiza lo siguiente:

- Si la contraseña comienza con una letra, la convierte en mayúscula y agrega un "0" al final.
- Se guarda la contraseña modificada en un nuevo archivo ('rockyou_mod.txt').

Para finalizar, muestra el número de contraseñas modificadas.

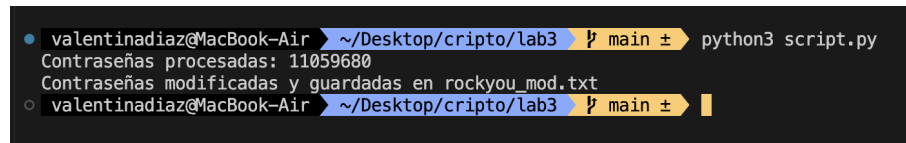
```
1 archivo_entrada = 'rockyou.txt'
2 archivo_salida = 'rockyou_mod.txt'
3
```

3.2 cantidad de passwords finales que contiene rockyou_mod.dicDESARROLLO (PASO 3)

```
4 cont = 0
5 with open(archivo_entrada, 'r', encoding='utf-8',
6 errors='ignore' ) as entrada, open(archivo_salida, 'w')
7 as salida:
8     for linea in entrada:
9         linea = linea.strip()
10        if len(linea)>0:
11            if linea[0].isdigit()==False:
12                cont += 1
13                linea_modificada = linea[0].upper() + linea[1:]
14                + "0"
15                salida.write(linea_modificada + '\n')
16
17
18 print ("Contrase as procesadas:", cont)
19 print("Contrase as modificadas y guardadas en", archivo_salida)
```

3.2. cantidad de passwords finales que contiene rockyou_mod.dic

La cantidad de contraseñas que tiene el archivo son 11059680.



```
• valentinadiaz@MacBook-Air ~/Desktop/cripto/lab3 1 main ± python3 script.py
Contraseñas procesadas: 11059680
Contraseñas modificadas y guardadas en rockyou_mod.txt
○ valentinadiaz@MacBook-Air ~/Desktop/cripto/lab3 1 main ±
```

Figura 8: Cantidad de contraseñas modificadas

4. Desarrollo (Paso 3)

4.1. obtiene contraseña con hashcat con potfile

Primero se debe convertir archivo descargado de .pcap a .hc22000, esto se hace a través de <https://hashcat.net/cap2hashcat/>.

Luego, se debe instalar Hashcat. Una vez instalado, se utiliza el siguiente comando `./hashcat/hashcat -m 22000 hash.hc22000 rockyou_mod.dic`, se obtiene lo siguiente:

```

Host memory required for this attack: 195 MB

Dictionary cache built:
* Filename...: rockyou_mod.dic
* Passwords...: 11059680
* Bytes.....: 119974148
* Keyspace...: 11059673
* Runtime...: 1 sec

1813acb976741b446d43369fb96dbf90:b0487ad2dc18:eede678cdf8b:VTR-1645213:Security0

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 22000 (WPA-PBKDF2-PMKID+EAPOL)
Hash.Target.....: hash.hc22000
Time.Started....: Sat Oct 21 01:52:22 2023 (1 sec)
Time.Estimated...: Sat Oct 21 01:52:23 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (rockyou_mod.dic)
Guess.Queue.....: 1/1 (100.00%)
Speed.#2.....: 51422 H/s (7.43ms) @ Accel:256 Loops:1024 Thr:32 Vec:1
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 99853/11059673 (0.90%)
Rejected.....: 34317/99853 (34.37%)
Restore.Point....: 0/11059673 (0.00%)
Restore.Sub.#2...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#2....: Password0 -> Danyelle10
Hardware.Mon.#2..: Util: 99%

Started: Sat Oct 21 01:51:56 2023
Stopped: Sat Oct 21 01:52:24 2023

```

Figura 9: Contraseña obtenida a partir de Hashcat

La contraseña para la red *VTR-1645213* es *Security0*.

4.2. identifica nomenclatura del output

El output es el siguiente:

1813acb976741b446d43369fb96dbf90:b0487ad2dc18:eede678cdf8b:VTR-1645213:Security0

Este output sigue el siguiente formato:

Hash:Salt:PMKID:SSID

Por lo tanto:

- Hash: 1813acb976741b446d43369fb96dbf90
- Salt: b0487ad2dc18
- PMKID: eede678cdf8b
- Nombre red:SSID : VTR-1645213:Security0

Por otro lado, se obtienen los siguientes parámetros sobre el proceso:

- Hash Mode: Indica el modo de hash utilizado en el ataque que es 22000.
- Hash Target: Corresponde al archivo de hash que se está intentando descifrar.
- Time Started: Es el tiempo de inicio en que se comenzó el proceso.
- Kernel Feature: Información sobre el kernel.
- Guess Base: Señala que se está utilizando un diccionario de contraseñas para realizar los intentos de descifrado.
- Speed: Velocidad a la que se están probando contraseñas.
- Recovered: Señala la cantidad de contraseñas que se recuperan con éxito.
- Progress: Señala la progresión del proceso de descifrado.
- Candidate Engine :Señala el modo en que se generan las contraseñas candidatas.
- Candidates: Muestra el rango de contraseñas que se están probando en ese momento.
- Hardware Monitoring: Información sobre las condiciones del hardware mientras se realiza el proceso.

4.3. obtiene contraseña con hashcat sin potfile

Se utilizó el siguiente comando `./hashcat/hashcat -m 22000 hash.hc22000 rockyou_mod.dic -potfile-disable -force`

```
1813acb976741b446d43369fb96dbf90:b0487ad2dc18:eede678cdf8b:VTR-1645213:Security0
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 22000 (WPA-PBKDF2-PMKID+EAPOL)
Hash.Target.....: hash.hc22000
Time.Started.....: Sat Oct 21 02:23:03 2023, (3 secs)
Time.Estimated...: Sat Oct 21 02:23:06 2023, (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (rockyou_mod.dic)
Guess.Queue.....: 1/1 (100.00%)
Speed.#2.....: 52557 H/s (7.37ms) @ Accel:256 Loops:512 Thr:64 Vec:1
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 193940/11059673 (1.75%)
Rejected.....: 62868/193940 (32.42%)
Restore.Point....: 0/11059673 (0.00%)
Restore.Sub.#2...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#2....: Password0 -> Kali4nia0
Hardware.Mon.#2..: Util: 99%

Started: Sat Oct 21 02:22:57 2023
Stopped: Sat Oct 21 02:23:07 2023
```

Figura 10: Contraseña encontrada

4.4. identifica nomenclatura del output

Los resultados son iguales al paso anterior, con la diferencia de que hay un nuevo candidato.

Cuando se repite el proceso, Hashcat indicará que las contraseñas encontradas están registradas en el archivo potfile de un ataque anterior. No obstante, al desactivar el potfile, el ataque se llevará a cabo de manera normal todas las veces que se repita.

4.5. obtiene contraseña con aircrack-ng

Para esta sección se utilizó el comando `aircrack-ng handshake.pcap -w rockyou mod.dic`

```
Aircrack-ng 1.7

[00:00:00] 2977/9296333 keys tested (6171.76 k/s)

Time left: 25 minutes, 5 seconds                                0.03%

KEY FOUND! [ Security0 ]

Master Key      : 55 E1 E0 F0 8E D7 53 80 F6 27 C6 DC 48 20 74 54
                  B7 54 98 37 71 FF C8 03 1D 89 C5 19 8D 6F AC 76

Transient Key   : 3C 1B 89 A6 31 30 BA 04 B6 59 D9 7E 65 BD D2 07
                  9E C6 8D 2A D6 EF 7F 9E A1 95 1C BC CC 62 A6 5D
                  CC 07 B2 E3 9D 12 99 A7 66 D4 53 00 00 00 00 00
                  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

EAPOL HMAC      : 18 13 AC B9 76 74 1B 44 6D 43 36 9F B9 6D BF 90

valentinadiaz@MacBook-Air ~/Desktop/cripto/lab3 main ±
```

Figura 11: Contraseña encontrada

4.6. identifica y modifica parámetros solicitados por pycrack

Dentro del archivo `pywd.py` se deben modificar todos los parámetros que se muestran a continuación:

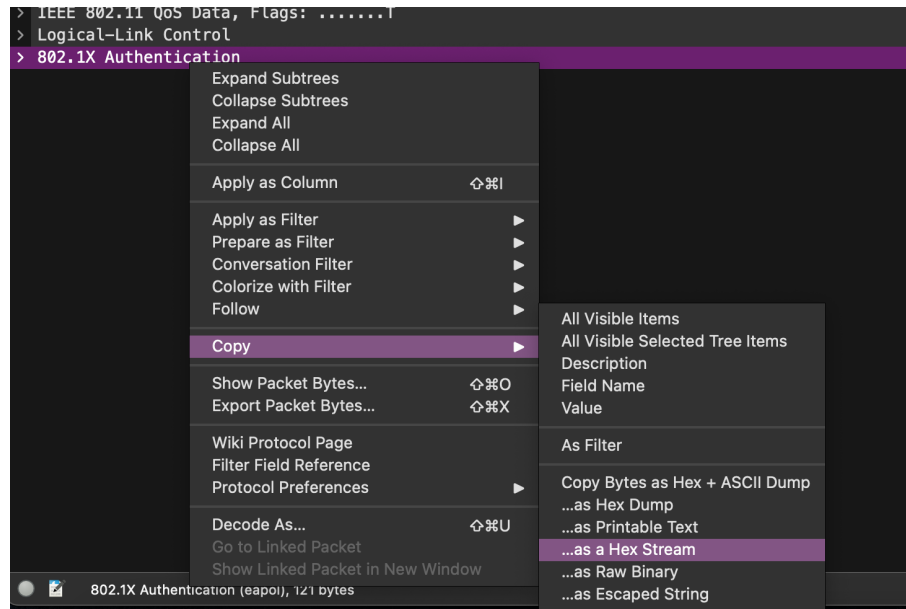


Figura 17: Valor data

4.7. obtiene contraseña con pycrack

Una vez que se reemplazan los valores en el script y se corre el programa, se obtiene lo siguiente:

```

x valentinadiaz@MacBook-Air ~/Desktop/cripto/lab3 main ± python3 pywd.py
pmk: EBB5D703F8834A08D61A67A982FA009E08F747DD65D82C240169E604218B3ACF

ptk: 63E412CE67759BD5CEBD0F5B5A487CA155ADD51D771293E31C05BF05A3A98BCFE64

desired mic: D5355382B8A9B806DCAF99CDAF564EB6
actual mic: C2EE0E125962261C897A05E33B579F5C
MISMATCH

desired mic: 1E228672D2DEE930714F688C5746028D
actual mic: 6D60808DE292A32BAE1D381B3D295B2F
MISMATCH

desired mic: 9DC81CA6C4C729648DE7F00B436335C8
actual mic: D5F07A0FBC8F376541D46591FDA74470
MISMATCH

!!!Password Found!!!
Desired MIC1: 1813acb976741b446d43369fb96dbf90
Computed MIC1: 1813acb976741b446d43369fb96dbf90

Desired MIC2: a349d01089960aa9f94b5857b0ea10c6
Computed MIC2: a349d01089960aa9f94b5857b0ea10c6

Desired MIC2: 5cf0d63af458f13a83daa686df1f4067
Computed MIC2: 5cf0d63af458f13a83daa686df1f4067
Password: Security0
valentinadiaz@MacBook-Air ~/Desktop/cripto/lab3 main ±

```

Figura 18: Contraseña obtenida

Password: Security0.

Conclusiones y comentarios

Durante esta enriquecedora experiencia, se utilizaron múltiples herramientas especializadas, las cuales son Hashcat, Aircrack-ng y Pycrack, todas ellas centradas en un mismo objetivo: comprender y abordar los ataques dirigidos a una red de manera efectiva.

Se profundizó en el uso de estas herramientas para descifrar hashes, lo que implicó explorar a fondo sus funcionalidades y procesos subyacentes. Además, se expandió el conocimiento en relación a los ataques de fuerza bruta, donde se emplearon diccionarios como valiosas herramientas para encontrar contraseñas.

Los objetivos de la experiencia se lograron con éxito en su totalidad, y los resultados obtenidos se mantuvieron coherentes en cada una de las actividades realizadas. Este enfoque integral permitió un mayor entendimiento de las amenazas y vulnerabilidades de la seguridad de redes, así como las estrategias necesarias para protegerlas de manera más efectiva.