USO DE METASPLOIT PARA LA EXPLOTACIÓN DE SERVICIOS EN PUERTOS ESPECÍFICOS EN KALI LINUX

KEVIN STIVEN JARAMILLO CORDOBA

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

USO DE METASPLOIT PARA LA EXPLOTACIÓN DE SERVICIOS EN PUERTOS ESPECÍFICOS EN KALI LINUX

KEVIN STIVEN JARAMILLO CORDOBA
Estudiantes, IX semestre

ING. RAFAEL SANDOVAL MORALES
Docente de la asignatura

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

Contenido

Tema……..……..……..**Pagina**

Tabla de Ilustración

1. Introducción

En este informe se documenta el proceso de explotación de servicios en los puertos 23, 25, 53, 139, 445, 2121, 3306 y 5432, los cuales fueron definidos por el docente como base del ejercicio. Cada uno de estos puertos está asociado a un servicio importante, como Telnet, FTP, SMB o bases de datos, y en muchos casos presentan vulnerabilidades.

El trabajo fue realizado en un entorno virtualizado, usando kali linux y metasploit, permitiendo llevar a cabo pruebas de forma segura. A través de este proceso, no solo se ponen en práctica los conocimientos adquiridos en clase, sino que también se refuerza la importancia de aplicar medidas de seguridad en redes.

2. Objetivos

2.1. Objetivo General

Realizar la explotación de servicios vulnerables en puertos específicos usando la herramienta metasploit sobre un entorno de red controlado

2.2. Objetivos Específicos

- Identificar los servicios en ejecución en los puertos asignados mediante escaneo y reconocimiento.
- Seleccionar exploits adecuados disponibles en metasploit para los servicios detectados.
- Documentar el proceso de explotación de cada servicio, describiendo comandos utilizados, resultados obtenidos y posibles vulnerabilidades encontradas.

3. Justificación

Este informe es fundamental porque me permite desarrollar habilidades prácticas en el área de la seguridad de red, especialmente en el uso de metasploit, una herramienta muy reconocida en el ámbito de las pruebas de penetración. Entender cómo funcionan los servicios y cómo pueden ser explotados ayuda a diseñar mejores mecanismos de defensa en redes reales. Además, esta práctica fortalece los conocimientos adquiridos en clase y nos acerca como estudiante a escenarios reales que podemos encontrar en nuestro futuro como profesionales.

4. Planteamiento de la Actividad

Durante el desarrollo de la asignatura Seguridad y Auditoría de Redes, el profesor Sandoval propuso una práctica orientada a fortalecer nuestras habilidades en el uso de herramientas de pruebas de penetración. Para ello, nos indicó trabajar específicamente con metasploit sobre un entorno kali linux, con el fin de identificar vulnerabilidades y realizar procesos de explotación controlada sobre servicios comunes de red.

El objetivo central de esta actividad fue comprender cómo ciertos servicios, cuando no están bien configurados o actualizados, pueden ser una puerta de entrada para un atacante. El profesor definió una lista de puertos específicos que debíamos analizar y explotar, los cuales son conocidos por ser utilizados por servicios importantes, pero que también han presentado fallas de seguridad en sus versiones antiguas:

- **Puerto 23:** Telnet
- **Puerto 25:** SMTP (envío de correos)
- **Puerto 53:** DNS (Resolución de nombres)
- **Puerto 139 y 445:** SMB (compartición de archivos en red)
- **Puerto 2121:** FTP (versión de ProFTPD vulnerable)
- **Puerto 3306:** MySQL (base de datos)
- **Puerto 5432:** PostgreSQL (base de datos relacional)

Cada estudiante debía realizar una explotación por cada puerto, documentar el proceso paso a paso, capturar evidencias mediante imágenes.

5. Desarrollo del tema

5.1. Configuración de red interna en VirtualBox

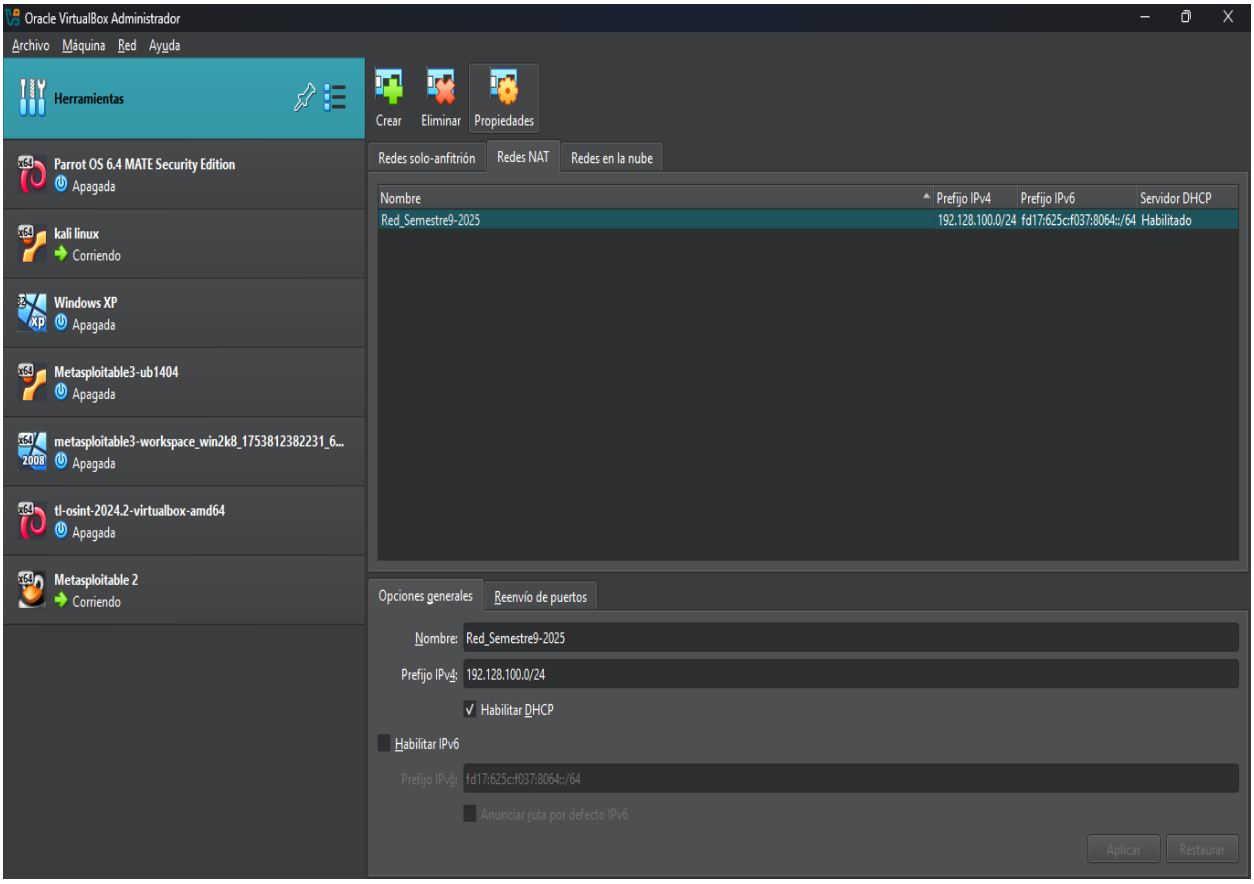

Ilustración 1: Configuración de red interna en VirtualBox para pruebas de explotación

En esta imagen se muestra la configuración de red en VirtualBox, donde se ha creado una red interna personalizada llamada **Red_Semestre9-2025**. Esta red es clave para la comunicación entre las máquinas virtuales utilizadas en las pruebas de explotación, sin requerir conexión a internet. Se asigna el rango de IPs 192.128.100.0/24 y se habilita el servidor DHCP para facilitar la asignación automática de direcciones IP.

5.2. Configuración del adaptador de red de la máquina virtual

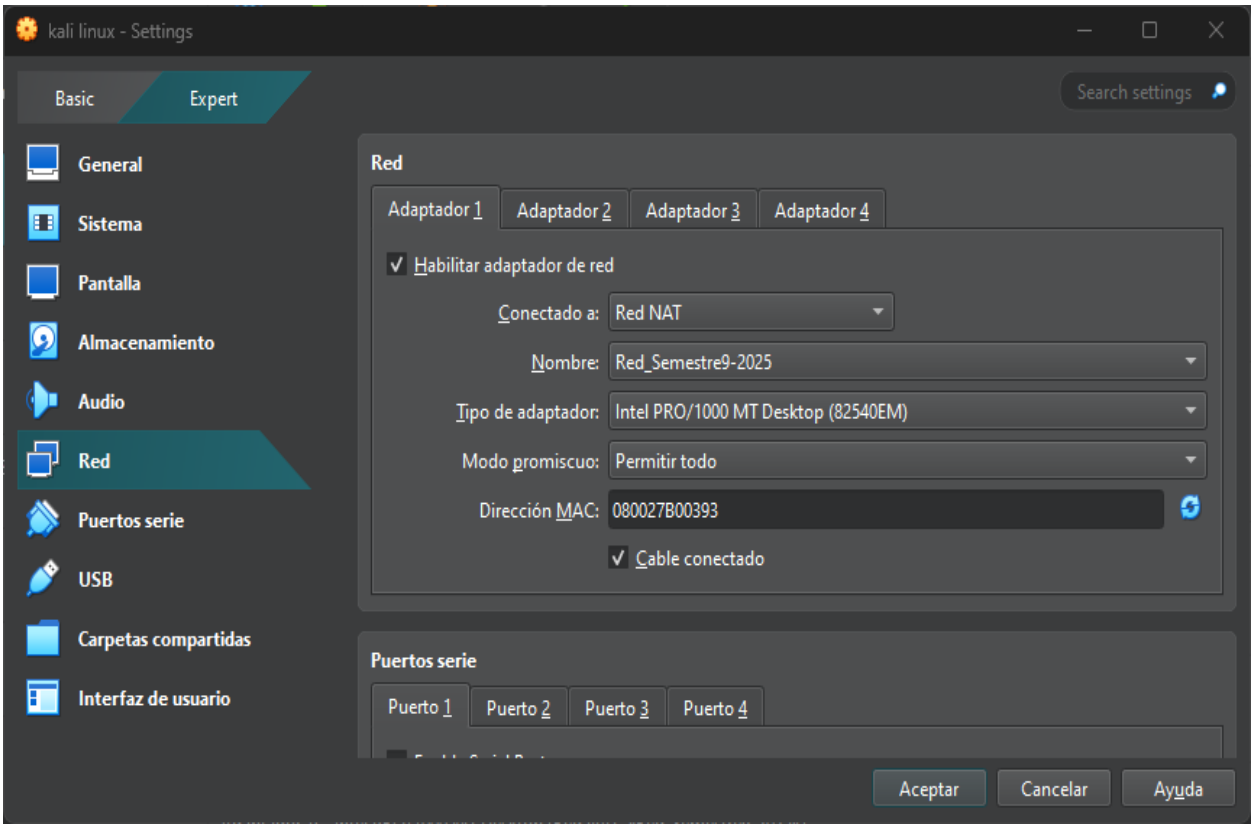

Ilustración 2: Conexión de Kali Linux a la red interna

Aquí se muestra la configuración de red de la máquina virtual kali linux, conectada a la red interna previamente creada. Se habilita el adaptador en modo NAT y se conecta a la red Red_Semestre9-2025, lo que permite que kali pueda comunicarse con las otras máquinas virtuales. Además, se activa el modo promiscuo, necesario para capturar todo el tráfico de red que circula por la red virtual.

5.3. Verificación de dirección IP asignada a Kali Linux

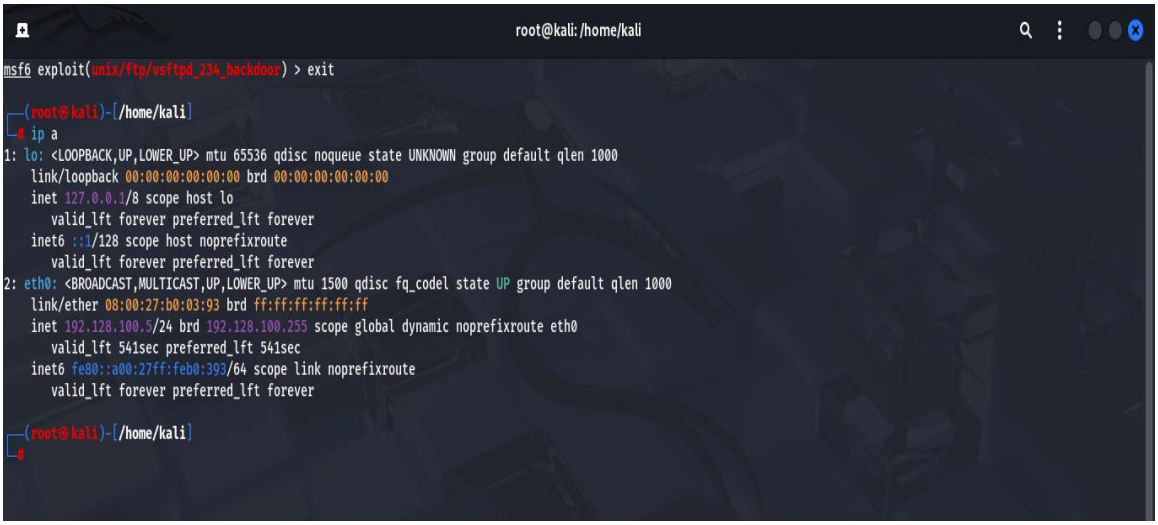

Ilustración 3: Verificación de dirección IP asignada en Kali Linux mediante terminal

Esta imagen muestra el resultado del comando ip a ejecutado en kali linux, con el fin de verificar la dirección IP asignada automáticamente dentro de la red interna. La interfaz eth0 tiene la IP 192.128.100.5, lo que confirma que está correctamente conectada y lista para iniciar la explotación de puertos.

5.4.SERVICIO FTP

5.4.1 Escaneo de red para identificar hosts activos

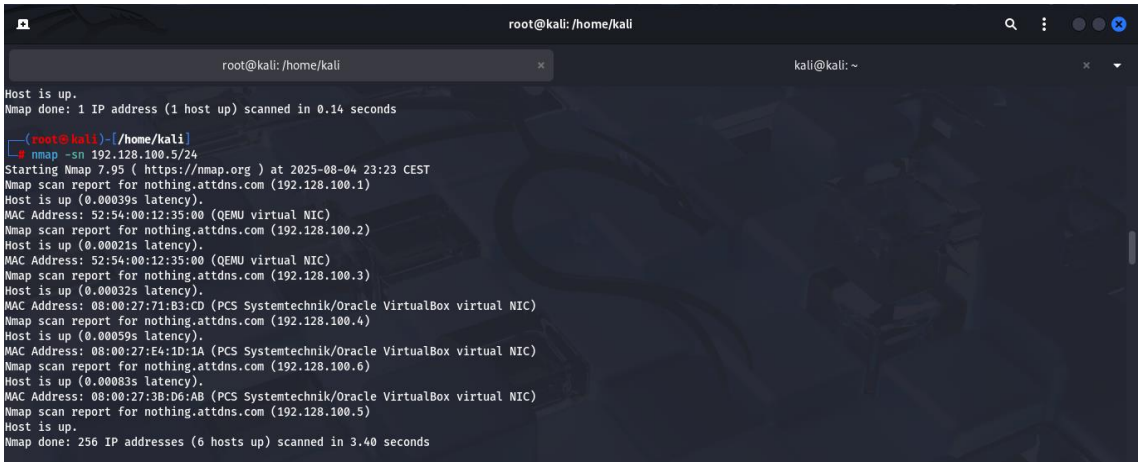

Ilustración 4: Detección de dispositivos activos en la red interna

Se utilizó el comando nmap -sn 192.128.100.5/24 para realizar un escaneo rápido de red y detectar los dispositivos conectados en el mismo rango de IP. El resultado muestra múltiples direcciones IP activas (como .100.1, .100.2, .100.3, etc.), todas asociadas a interfaces de red virtuales de VirtualBox.

5.4.2 Escaneo de puertos abiertos en el host

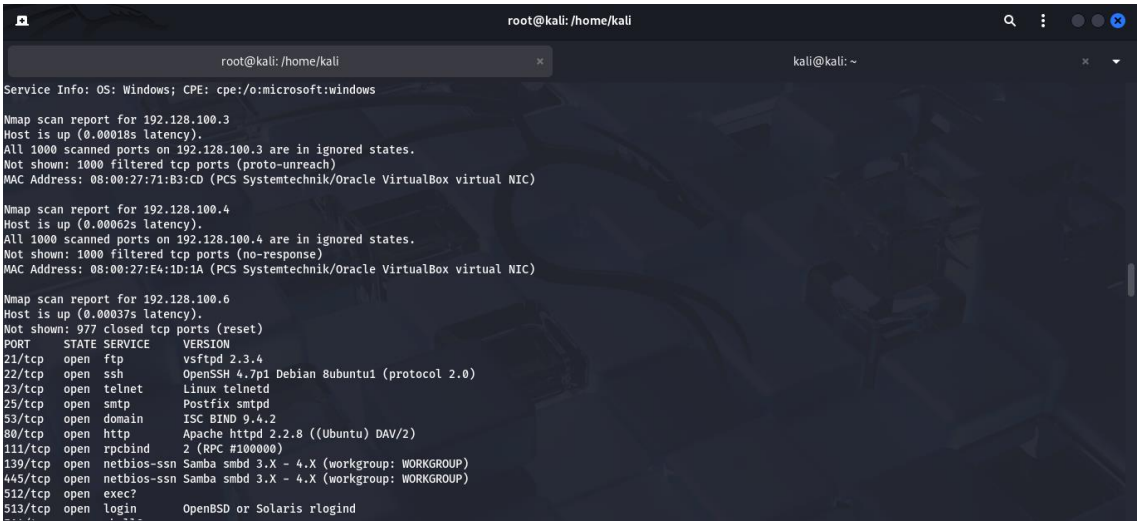

Ilustración 5: Identificación de puertos y servicios abiertos

Aquí se muestra un escaneo detallado con nmap donde se identifican los puertos abiertos en el host 192.128.100.6. Entre los servicios encontrados destacan el FTP (puerto 21), SSH (22), Telnet (23), SMTP (25), DNS (53), SMB (139 y 445), y MySQL (3306).

5.4.3 Selección del exploit vsftpd_234_backdoor

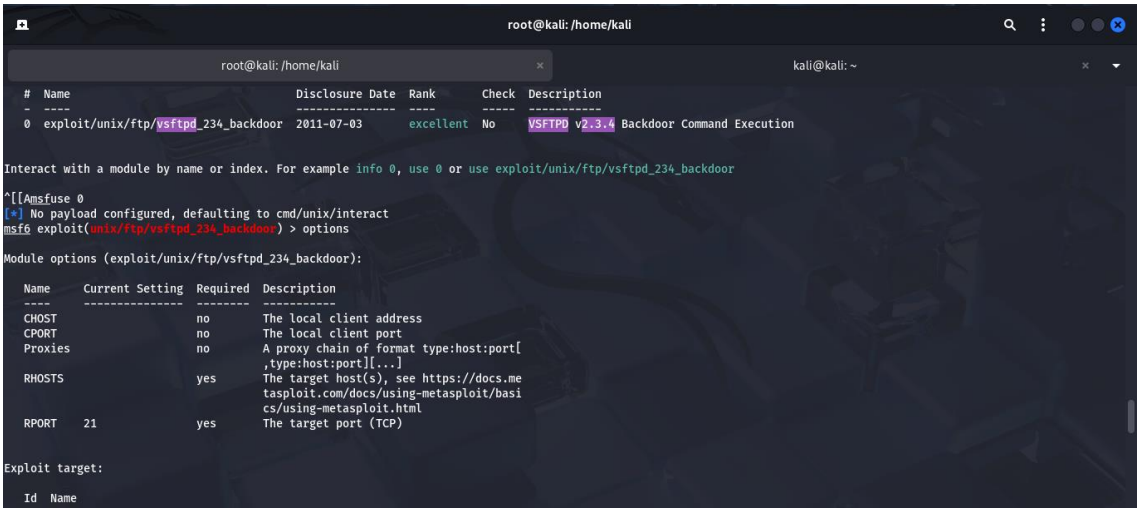

Ilustración 6:Carga del exploit VSFTPD 2.3.4 en Metasploit.

Se utilizó la herramienta metasploit en kali linux para buscar y cargar el exploit vsftpd_234_backdoor, el cual aprovecha una vulnerabilidad de tipo backdoor presente en la versión 2.3.4 del servicio FTP VSFTPD. Esta vulnerabilidad permite la ejecución remota de comandos si se establece una conexión al puerto 21.

En la imagen se observa cómo el exploit es cargado correctamente y se muestran las configuraciones necesarias, incluyendo la dirección IP del objetivo (RHOST) y el puerto del servicio (RPORT).

5.4.4 Ejecución del exploit y obtención de acceso remoto como usuario root

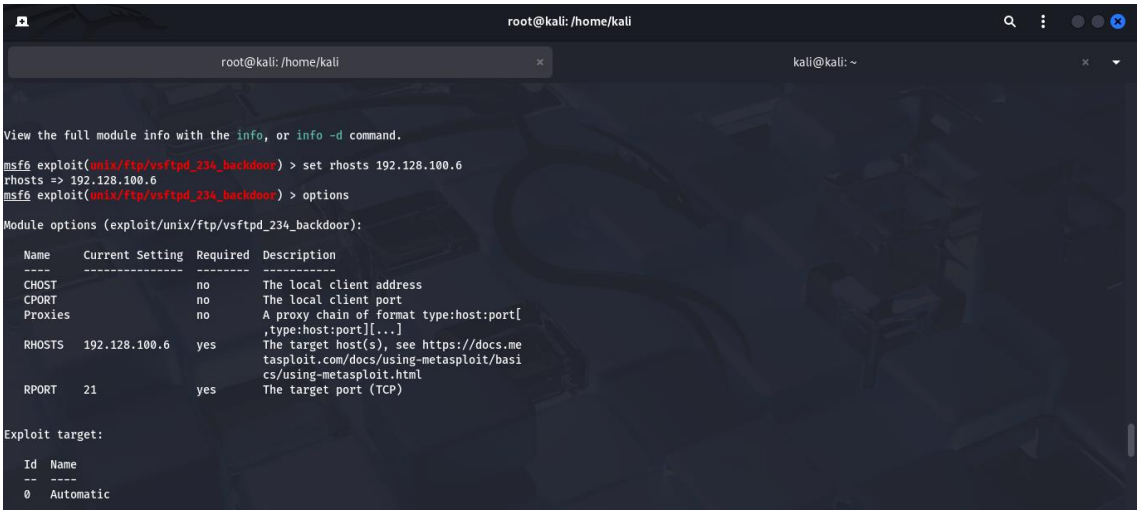

Ilustración 7: Explotación exitosa del servicio FTP con VSFTPD.

Luego de cargar el módulo vsftpd_234_backdoor, se configuró el payload cmd/unix/interact, el cual permite interactuar con el sistema mediante comandos. Posteriormente, se ejecutó el exploit y se logró establecer una conexión con el objetivo ubicado en la IP 192.128.100.6. El sistema vulnerable permitió una conexion remota con permisos de usuario root, lo cual confirma la explotación exitosa.

En la imagen se evidencia el uso de comandos como whoami y ip a para verificar el acceso y la conectividad.

5.4.5 Visualización del archivo /etc/passwd

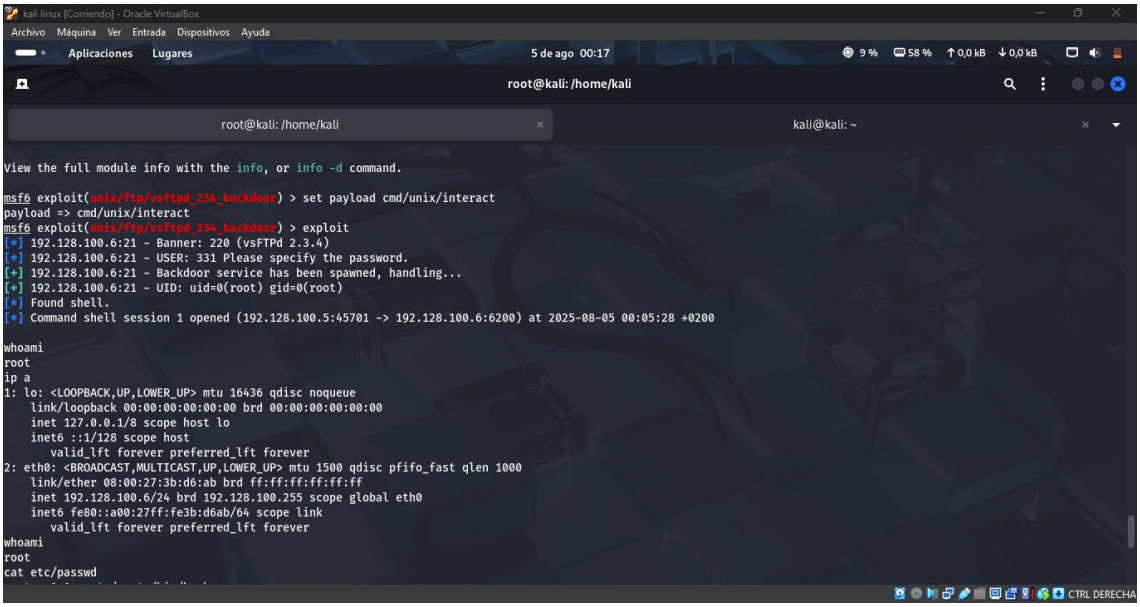

Ilustración 8: Enumeración de usuarios desde el sistema vulnerado.

Una vez obtenida la conexion remota en el sistema, se utilizó el comando cat /etc/passwd para visualizar el contenido del archivo que almacena los usuarios del sistema. Este archivo no contiene contraseñas, pero sí permite conocer todos los nombres de usuario registrados, las rutas predeterminadas y sus directorios de trabajo.

5.4.6 Ruta de acceso a diccionarios de contraseñas

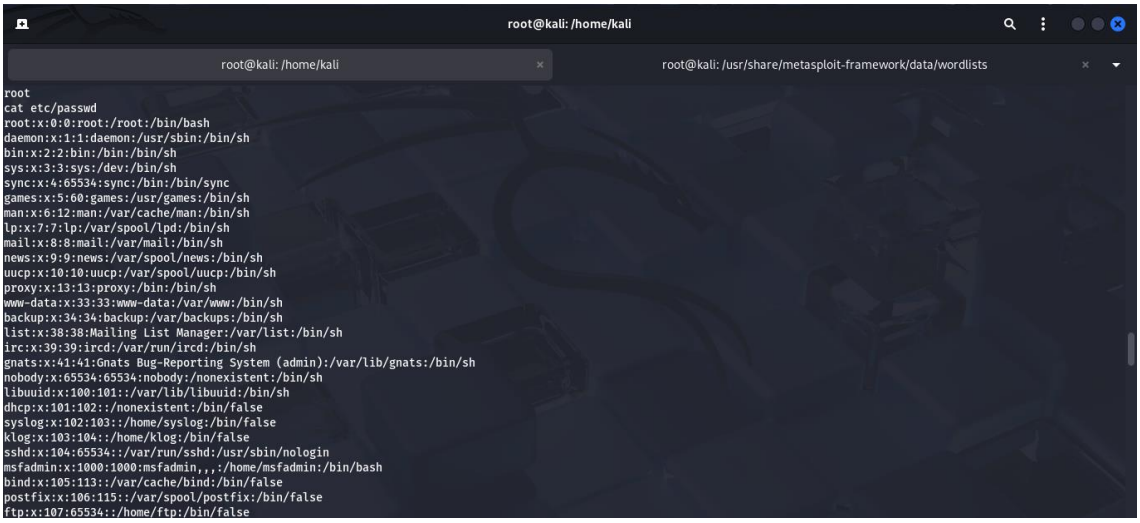

Ilustración 9: Ubicación de diccionarios de contraseñas en Metasploit.

Se accedió al directorio /usr/share/metasploit-framework/data/wordlists/, que contiene diccionarios predefinidos, como la adivinación de contraseñas. Esta acción permite preparar ataques más avanzados una vez se han identificado servicios como SSH o FTP que requieren autenticación.

5.5.SERVICIO SSH

5.5.1 Escaneo de servicios para detección del puerto SSH abierto

Ilustración 10: Detección del servicio SSH activo mediante escaneo con Nmap

En esta imagen se realizó un escaneo de versión y servicios (nmap -sV) sobre el rango de red 192.128.100.0/24. El objetivo era encontrar qué servicios estaban corriendo en los distintos hosts. Se detectó que el puerto 22/tcp está abierto con el servicio OpenSSH activo, lo que confirma que es posible realizar pruebas o incluso intentos de acceso si existen credenciales débiles o mal protegidas.

5.5.2 Búsqueda de vulnerabilidades a OpenSSH 4.7p1

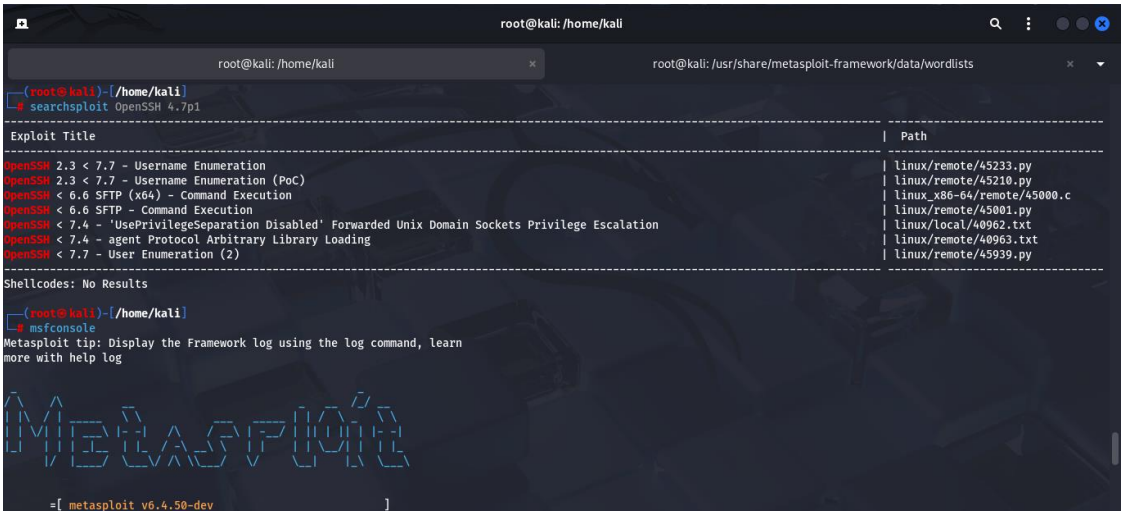

Ilustración 11: Exploración de vulnerabilidades SSH usando SearchSploit

Luego de identificar la versión del servicio SSH, se utilizó la herramienta searchsploit para buscar exploits conocidos relacionados con OpenSSH 4.7p1, la cual está instalada en uno de los equipos de la red. La herramienta devuelve varias vulnerabilidades relacionadas con ejecución de comandos remotos, enumeración de usuarios y privilegios.

5.5.3 Selección y parametrización del módulo ssh_login

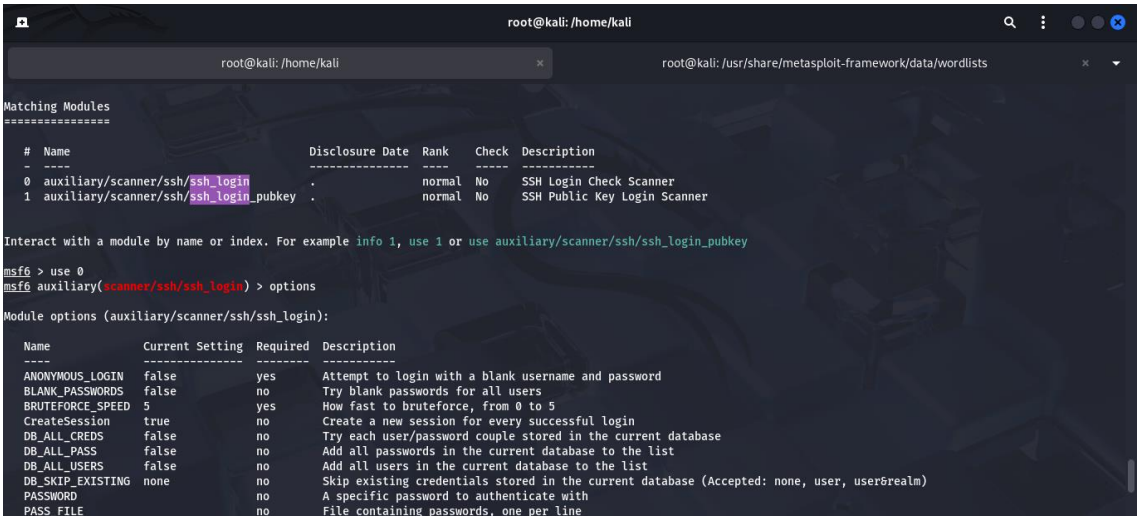

Ilustración 12: Configuración del módulo SSH en Metasploit

En la imagen se muestra la interfaz con el módulo auxiliary/scanner/ssh/ssh_login seleccionado. Se configuran opciones como BLANK_PASSWORDS, BRUTEFORCE_SPEED y PASS_FILE para definir el comportamiento del ataque, incluyendo la velocidad de ejecución y el uso de diccionarios de contraseñas.

5.5.4 Definición de objetivos y ajustes de fuerza bruta contra el servicio SSH.

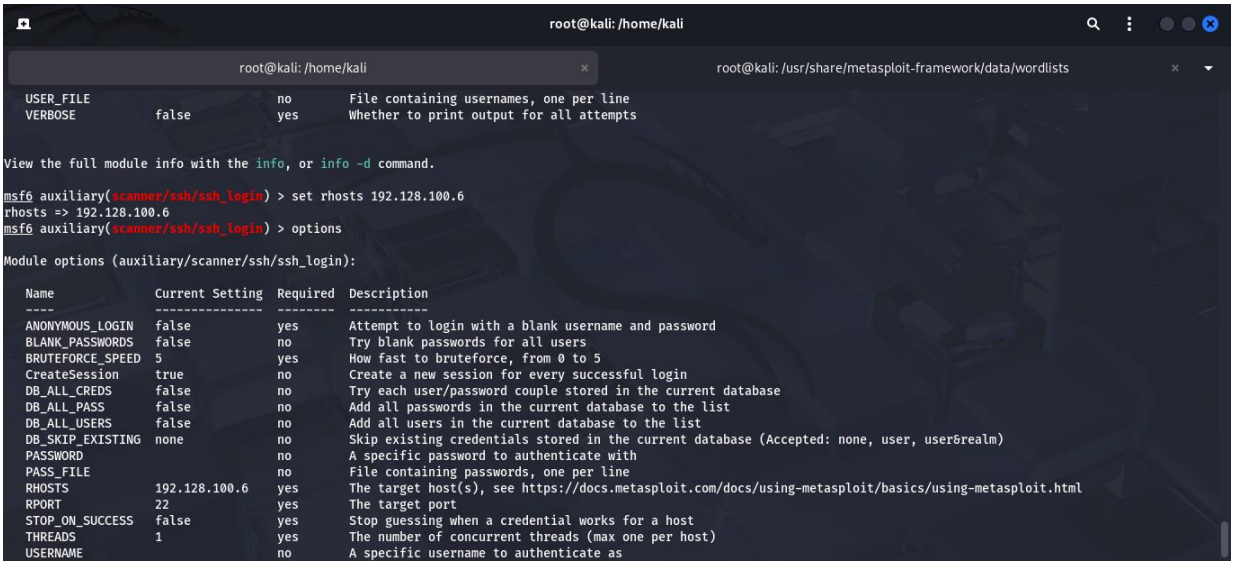

Ilustración 13: Configuración de parámetros para ataque SSH con Metasploit

La imagen muestra la configuración avanzada del módulo ssh_login, donde se especifican los parámetros clave para el ataque:

- rhosts: Se define la red objetivo (192.128.100.0).
- **Opciones de fuerza bruta**: Se ajustan BRUTEFORCE_SPEED (velocidad), THREADS (hilos), y STOP_ON_SUCCESS (detenerse al encontrar credenciales válidas).
- **Archivos de diccionario**: Se menciona el uso de USER_FILE y PASS_FILE para probar combinaciones de usuarios y contraseñas.

5.5.5 Especificación de lista de credenciales de acceso SSH.

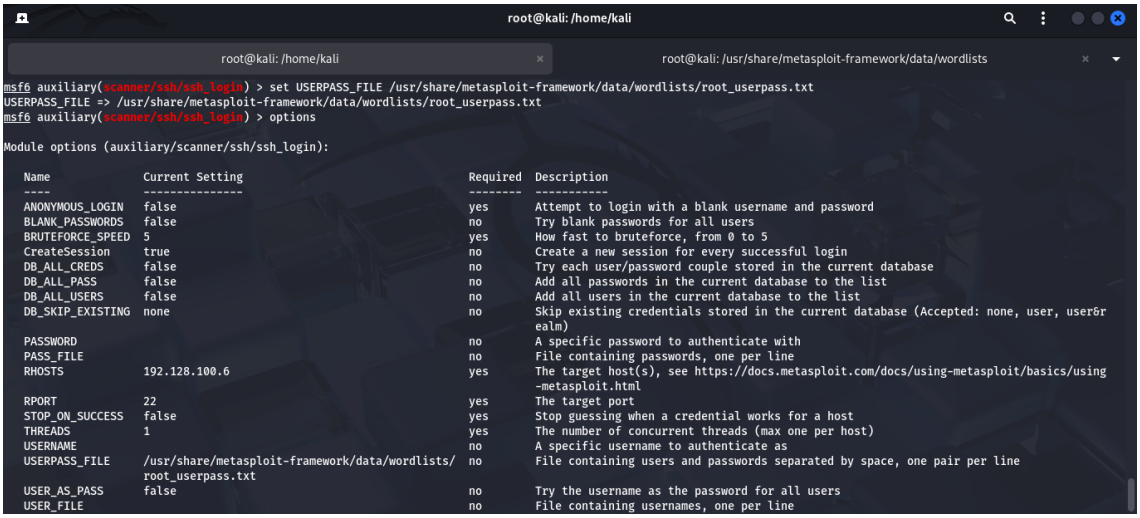

Ilustración 14: Configuración de archivo de credenciales para ataque SSH

La imagen muestra la configuración final del módulo ssh_login en Metasploit, donde se destaca:

- USERPASS_FILE: Se establece la ruta del archivo root_userpass.txt que contiene usuarios y contraseña para el ataque.
- **Parámetros clave**: Se mantienen configuraciones como RHOSTS (192.128.100.6), RPORT (22), y THREADS (1) para el escaneo.
- **Opciones de autenticación**: Se habilitan métodos como USER_AS_PASS (usar el nombre de usuario como contraseña) y BLANK_PASSWORDS (intentar contraseñas vacías).

5.5.6 Obtención de acceso al sistema mediante credenciales vulnerables.

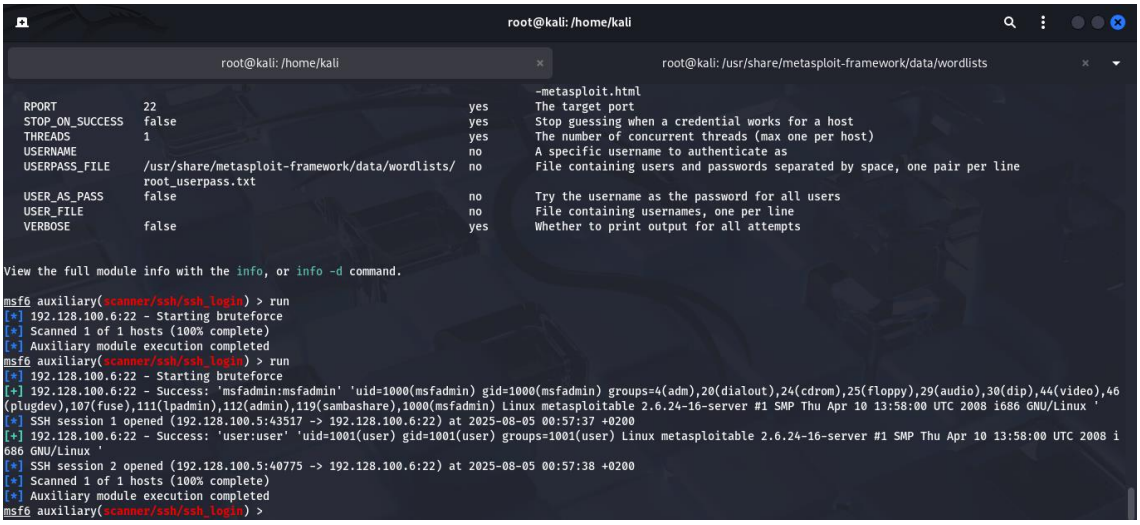

Ilustración 15: Resultado exitoso de ataque de fuerza bruta SSH

La imagen muestra los resultados de la ejecución del módulo ssh_login, donde se logró acceso exitoso al servidor SSH (192.128.100.6:22) mediante credenciales vulnerables:

1. **Usuario:** msfadmin / **Contraseña:** msfadmin
2. **Usuario:** user / **Contraseña:** user

Se observa la creación de sesiones SSH activas (session 1 opened) con información del sistema objetivo (metasploitable linux, kernel 2.6.24), confirmando la explotación exitosa de. El ataque se completó al 100% en el host escaneado.

5.6.SERVICIO TELNET

5.6.1 Identificación del módulo telnet_login para pruebas de autenticación

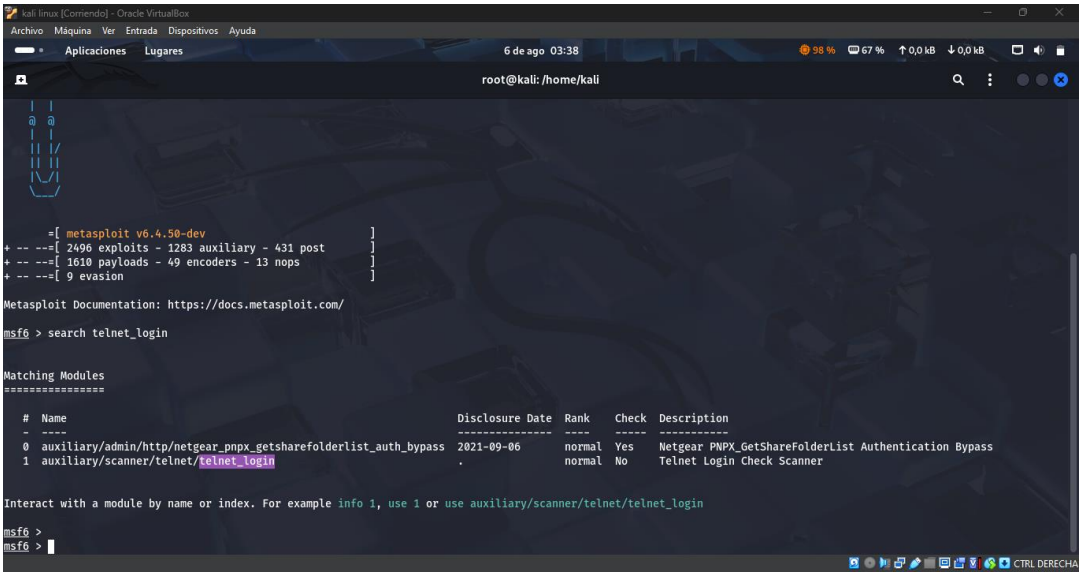

Ilustración 16: Búsqueda de módulos para explotación de Telnet

La imagen muestra la búsqueda del módulo telnet_login mediante el comando search telnet_login. Se listan los resultados, destacando:

- Módulo seleccionado: auxiliary/scanner/telnet/telnet_login (índice 1), diseñado para escanear y probar credenciales en servicios Telnet.
- Contexto: Se observa el entorno de Metasploit con estadísticas de exploits (2690), payloads (1610) y otros recursos disponibles.

5.6.2 Configuración del módulo telnet_login

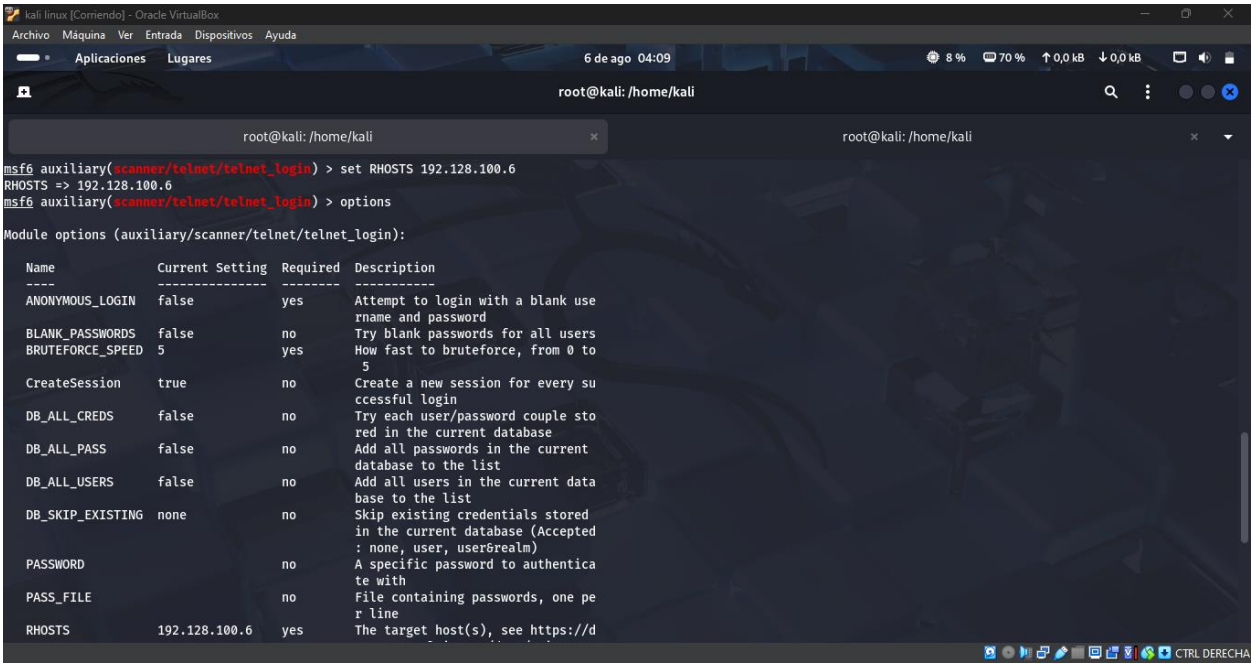

Ilustración 17: operación de red Telnet registrado

En esta fase del proceso, se preparó un ataque contra el servicio Telnet que estaba activo en el host 192.128.100.6 a través del puerto 23. Para ello, se utilizó el módulo auxiliar scanner/telnet/telnet_login, el cual permite realizar pruebas mediante combinación de usuario y contraseña.

En la imagen se observa cómo se configura la variable RHOSTS con la IP y se visualizan las opciones del módulo (options), las cuales permiten personalizar el escaneo.

5.6.3 Descompresión del archivo rockyou.txt.gz para su uso

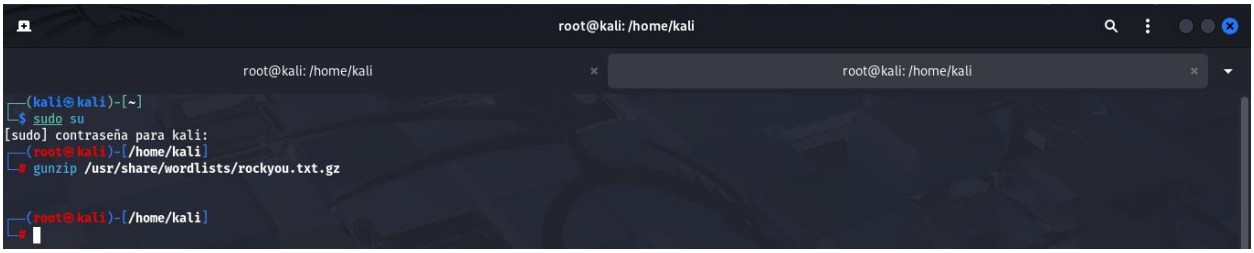

Ilustración 18: Preparación del diccionario RockYou para ataques de fuerza bruta

La imagen muestra el proceso de preparación del diccionario de contraseñas RockYou en kali linux. Se observa:

1. El usuario kali obtiene privilegios root mediante su
2. Se descomprime el archivo /usr/share/wordlists/rockyou.txt.gz usando el comando gunzip
3. El diccionario queda listo en /usr/share/wordlists/rockyou.txt para ser utilizado en herramientas como Metasploit, Hydra o John the Ripper

5.6.4 Asignación de listas de usuarios y contraseñas

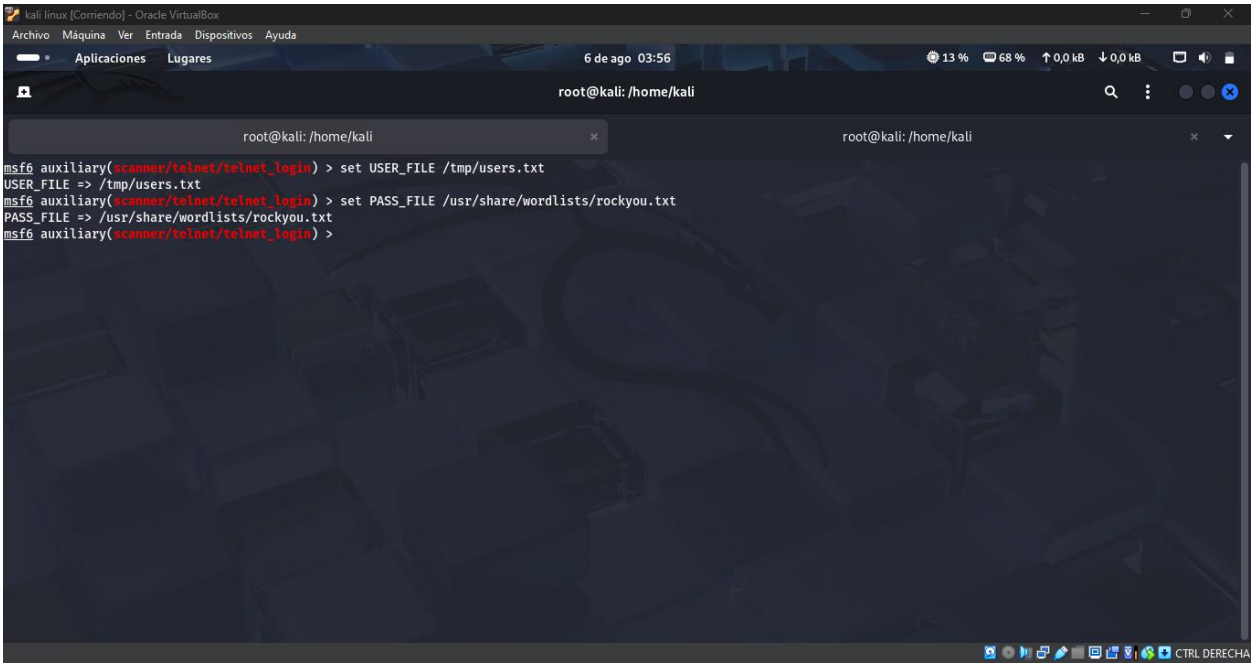

Ilustración 19: Configuración de archivos para ataque de fuerza bruta a Telnet

La imagen muestra la configuración del módulo telnet_login donde se establecen:

1. **USER_FILE**: Se especifica /tmp/users.txt como archivo que contiene la lista de nombres de usuario a probar
2. **PASS_FILE**: Se asigna el diccionario /usr/share/wordlists/rockyou.txt que contiene millones de contraseñas comunes

5.6.5 Acceso al archivo users.txt que contiene usuarios y contraseñas

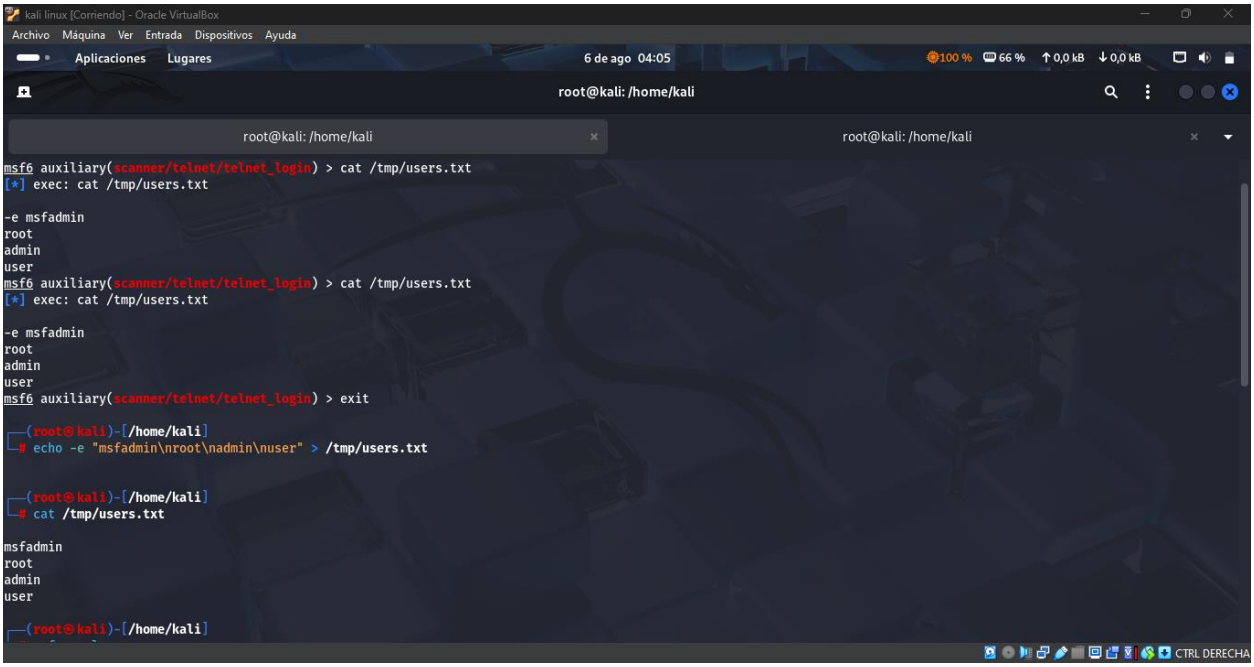

Ilustración 20: Visualización del archivo users.txt en entorno Kali Linux

En esta imagen se observa cómo se accede al archivo users.txt desde la terminal en kali linux, el cual contiene una lista de usuarios y contraseñas que serán utilizados en el módulo sobre el servicio Telnet. Se visualizan credenciales como msfadmin, root y admin, todas preparadas para ser probadas desde metasploit.

5.6.6 Prueba de credenciales usando el módulo telnet_login

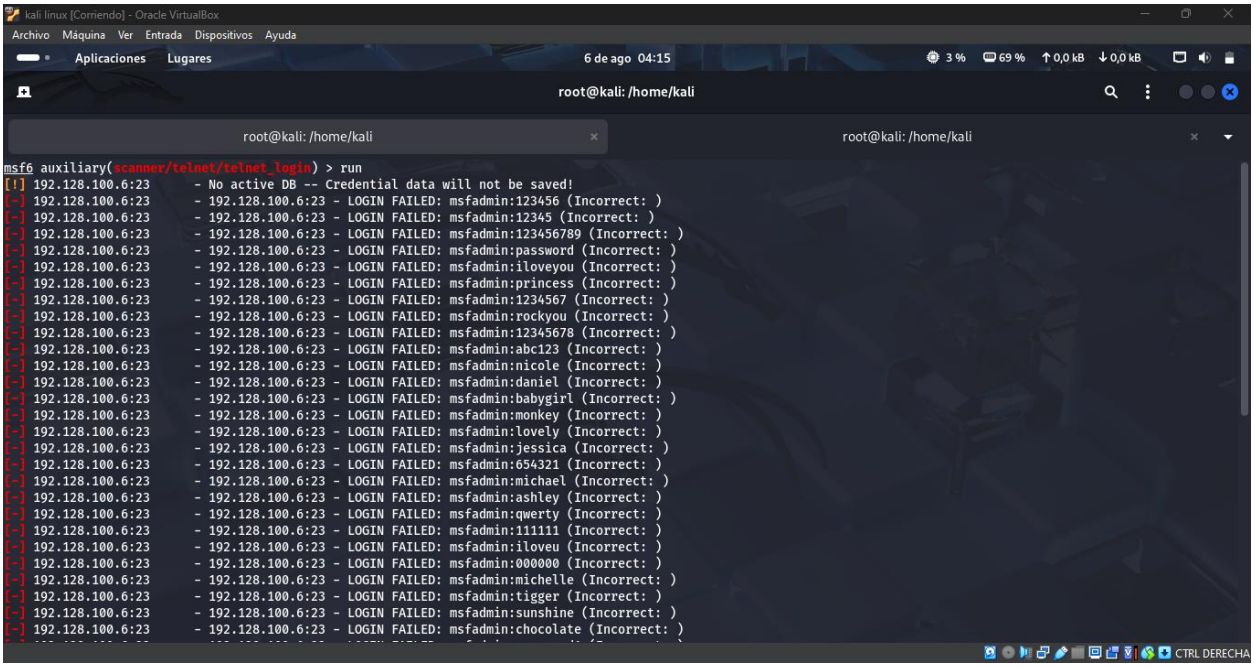

Ilustración 21: Ejecución de ataque de fuerza bruta al servicio Telnet con

Aquí se muestra la ejecución del módulo scanner/telnet/telnet_login de Metasploit, donde se realiza un ataque contra el servicio Telnet en la dirección IP 192.128.100.6. El sistema prueba múltiples combinaciones de usuario y contraseña desde el archivo users.txt. Todos los intentos fallan, indicando que las credenciales utilizadas no son válidas.

5.6.7 Acceso inicial al sistema vulnerable con credenciales por defecto

Ilustración 22: Inicio de sesión en Metasploitable con telnet

La imagen muestra el proceso donde se observa:

1. Advertencia de seguridad sobre no exponer el sistema a redes no confiables
2. Credenciales por defecto proporcionadas (msfadmin/msfadmin)
3. Información del último acceso (5 de agosto de 2025)
4. Detalles del sistema: Linux Metasploitable 2.6.24-16-server

5.6.8 Comprobación de información básica del sistema vulnerable

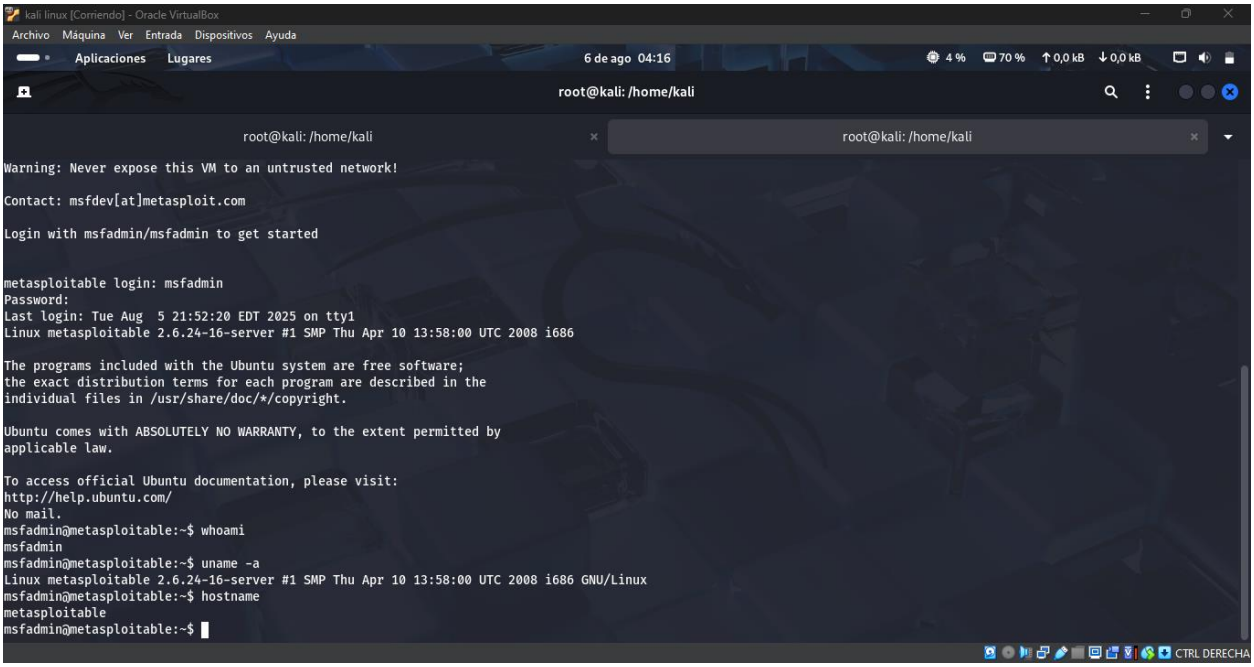

Ilustración 23: Sesión activa en Metasploitable con verificación de sistema

La imagen muestra la sesión activa donde se realizan varias comprobaciones clave:

1. Credenciales usadas: Acceso exitoso con usuario msfadmin (contraseña por defecto)
2. Comandos ejecutados:

   - whoami: confirma la identidad del usuario actual (msfadmin)
   - uname -a: muestra detalles del kernel (Linux 2.6.24-16-server)
   - hostname: revela el nombre del sistema (metasploitable)

3. Advertencias de seguridad: Recordatorio de no conectar esta VM a redes no seguras

5.7.PUERTO 25 – SMTP

5.7.1 Carga y configuración del módulo `smtp_enum` en SMTP

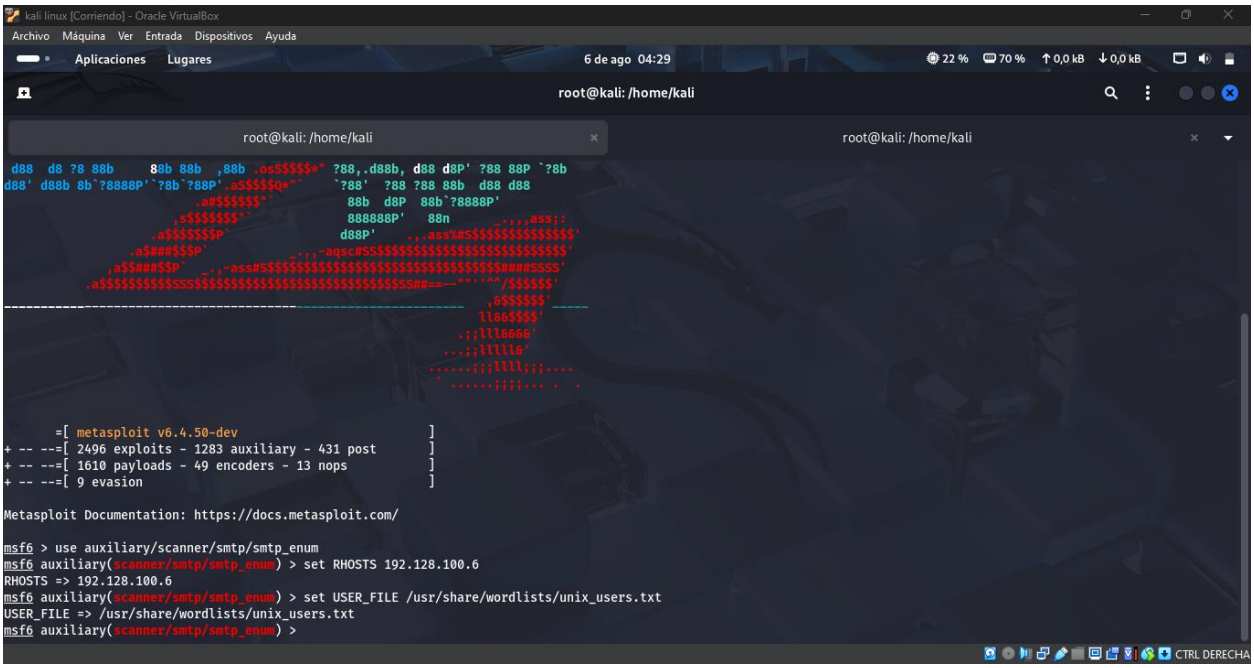

Ilustración 24: Configuración del módulo smtp_enum

En esta imagen se observa cómo se selecciona el módulo auxiliar smtp_enum, utilizado para enumerar posibles nombres de usuario en el servicio SMTP. Se configuran las opciones básicas como la dirección IP del objetivo (192.128.100.6) y la ruta del archivo de diccionario de usuarios (/usr/share/wordlists/unix_users.txt).

5.7.2 Ejecución del módulo smtp_enum en el puerto 25 SMTP

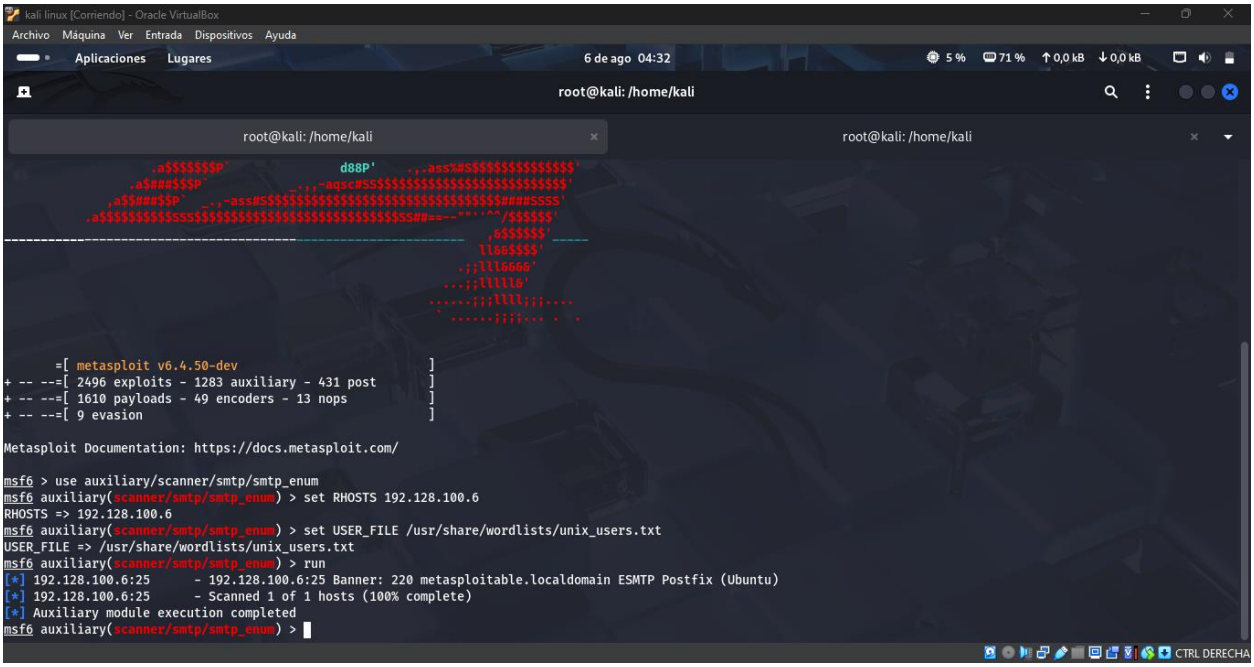

Ilustración 25: Resultado del escaneo SMTP y detección del servicio Postfix

Aquí se muestra la ejecución del módulo smtp_enum, el cual logra comunicarse con el servidor SMTP en la dirección 192.128.100.6:25, confirmando que el servicio responde y está activo. El banner indica que se trata de ESMTP Postfix en Ubuntu, información útil para saber qué software y sistema operativo están detrás del puerto.

5.7.3 Identificación del SMTP mediante módulo smtp_version

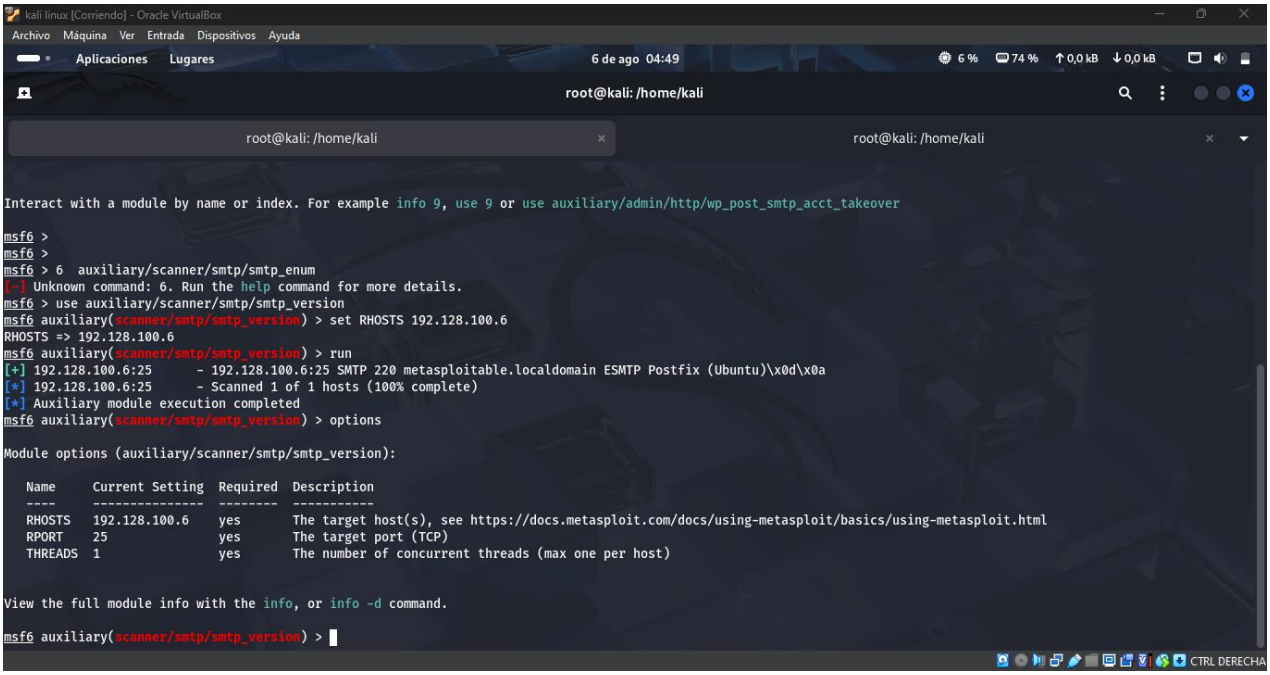

Ilustración 26: Identificación del software SMTP y configuración del módulo

En esta imagen se utiliza el módulo smtp_version, que permite detectar el software exacto que se ejecuta en el puerto SMTP del host víctima. El resultado confirma que se trata de Postfix (Ubuntu), lo que permite buscar exploits específicos para esa versión. También se visualiza la opción THREADS, que permite configurar el rendimiento del escaneo si se trabaja con múltiples hosts.

5.8.PUERTO 53 – DNS (ISC BIND 9.4.2)

5.8.1 Búsqueda de módulos para ataques de falsificación DNS

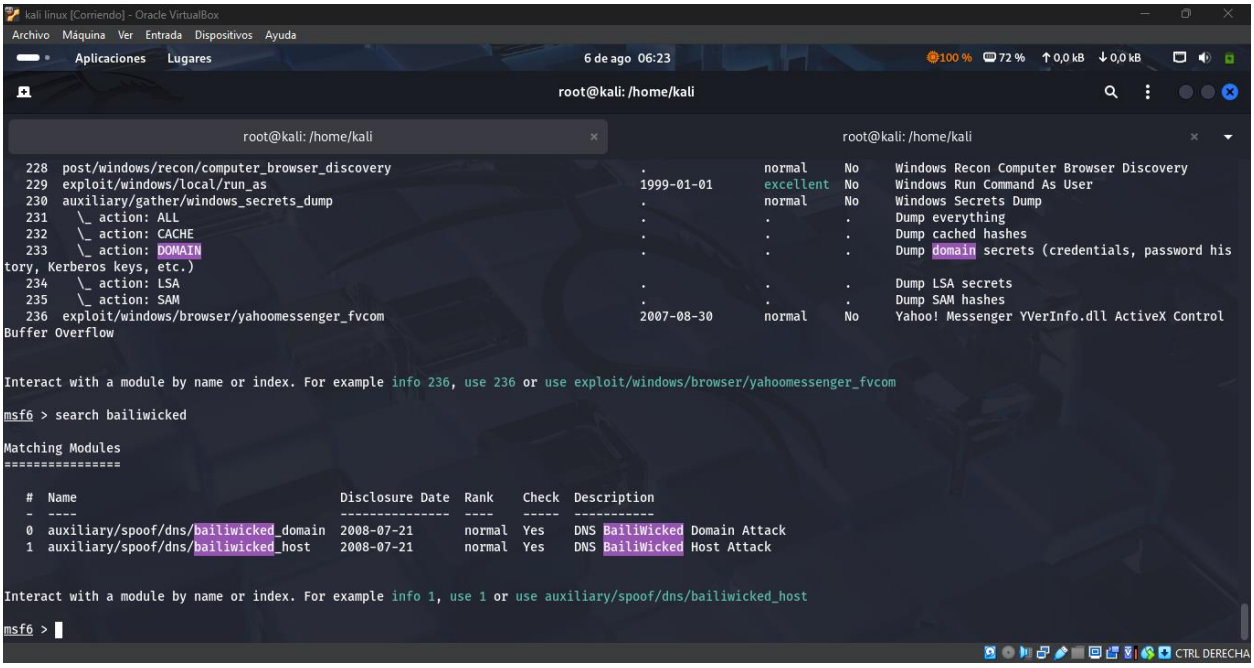

Ilustración 27: Identificación de módulos para ataque DNS tipo Bailiwicked.

En esta imagen se observa la localización de los módulos relacionados con ataques a servidores DNS vulnerables. Se ejecuta:

search bailiwicked

Esto devuelve dos módulos:

1. auxiliary/spoof/dns/bailiwicked_domain
2. auxiliary/spoof/dns/bailiwicked_host

Ambos están diseñados para explotar fallas en la validación de autoridad de dominios dentro de servidores DNS como BIND. Estas vulnerabilidades permiten realizar spoofing de respuestas DNS, en lo que se conoce como ataques Bailiwicked, que consisten en inyectar registros DNS falsos en la caché del servidor para redirigir a los usuarios a sitios maliciosos.

Aunque aún no se ejecuta el módulo, la imagen muestra el proceso de identificación del vector de ataque, una etapa crítica para planificar una explotación.

5.8.2 Visualización módulo DNS Spoofing bailiwicked_domain

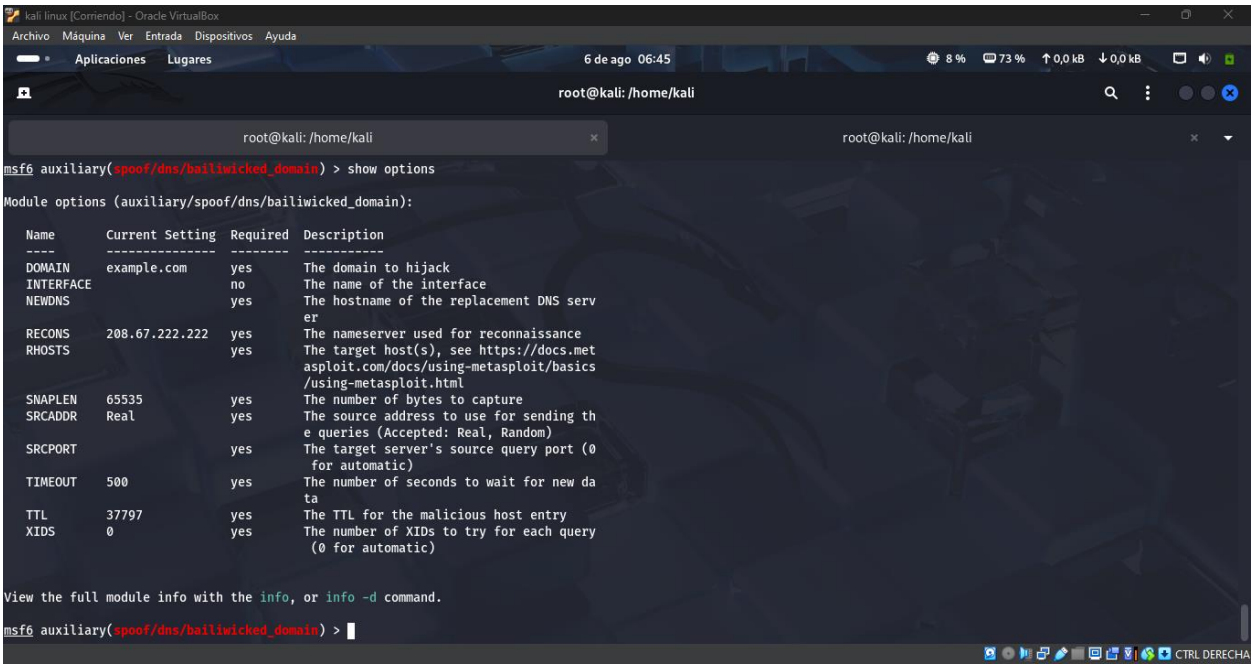

Ilustración 28: Opciones del módulo Metasploit para DNS Spoofing en puertos 53

En esta imagen se observa el módulo auxiliary/spoof/dns/bailiwicked_domain, el cual permite realizar DNS Spoofing, es decir, suplantar el nombre de dominio legítimo para redirigir las consultas a un servidor malicioso. Se muestran las opciones que deben configurarse, como el dominio a suplantar (DOMAIN), el nuevo servidor DNS (RECONS), el objetivo (RHOSTS) y los parámetros técnicos de TTL y número de paquetes (XIDS).

5.8.3 Configuración y ejecución del ataque de suplantación DNS

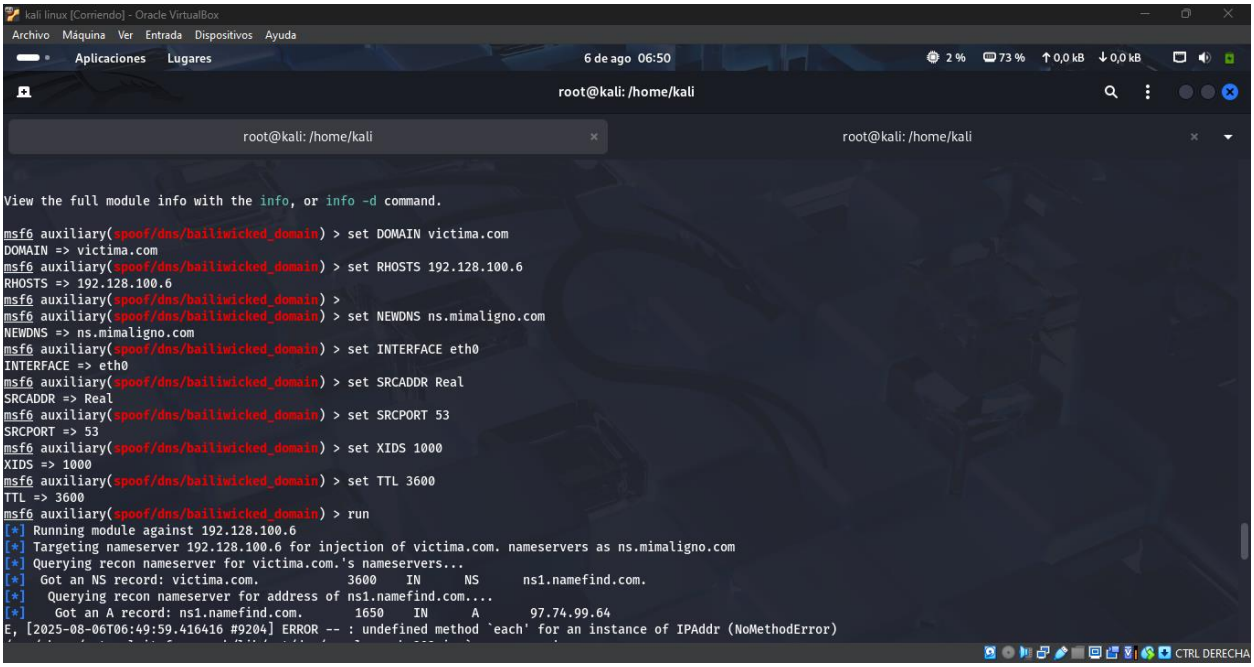

Ilustración 29: Ejecución del módulo de suplantación DNS sobre el puerto 53

Aquí se configuran todos los parámetros necesarios para realizar el ataque DNS. Se cambia el dominio objetivo a victima.com, el nuevo servidor de nombres a ns.mimaligno.com y se especifica que se usará la interfaz de red eth0. Luego de ejecutar el módulo, se observa que se intenta suplantar el dominio con respuestas falsas, aunque se presenta un error de tipo de dato (NoMethodError).

5.8.4 Revisión final de parámetros del módulo tras la ejecución

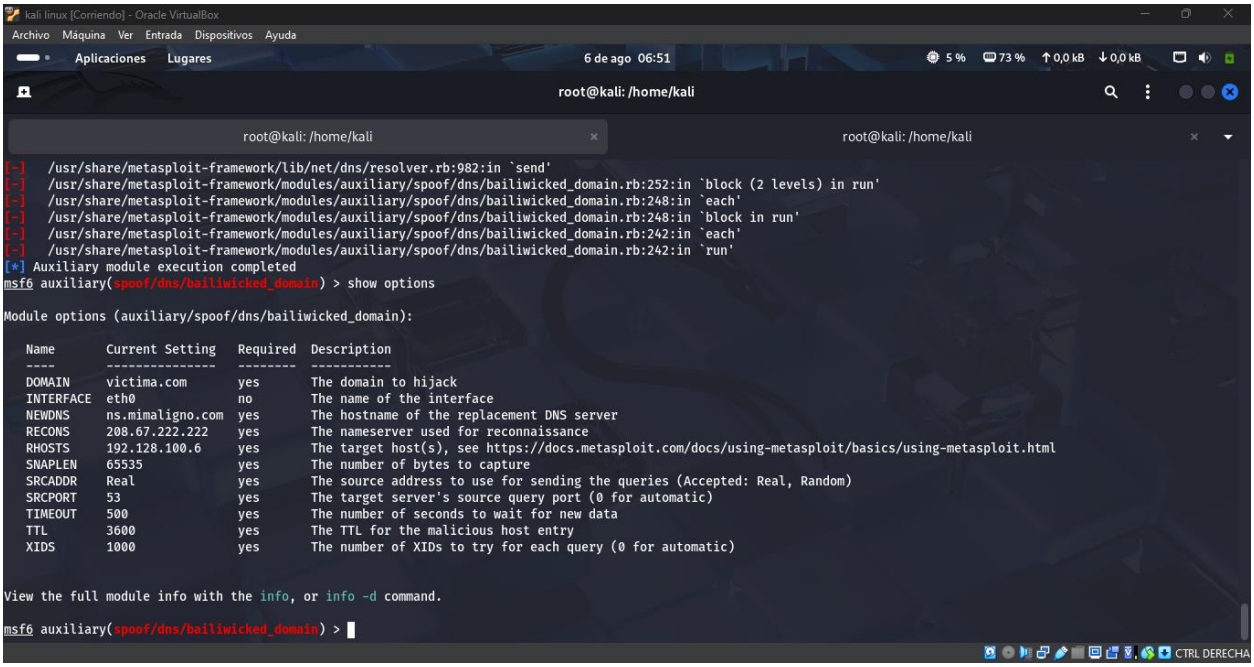

Ilustración 30: Parámetros configurados tras intento de ataque DNS spoofing

Después de ejecutar el módulo, se muestran nuevamente las opciones activas, confirmando los valores utilizados en el intento de spoofing DNS. Esto permite verificar si los parámetros estaban correctamente asignados. Aunque el intento presentó errores técnicos, la configuración realizada fue adecuada para realizar una prueba de suplantación del servicio DNS que corre sobre el puerto 53.

5.9.PUERTO 2121 - PROFTPD 1.3.1

5.9.1 Configuración del exploit proftpd_133c_backdoor

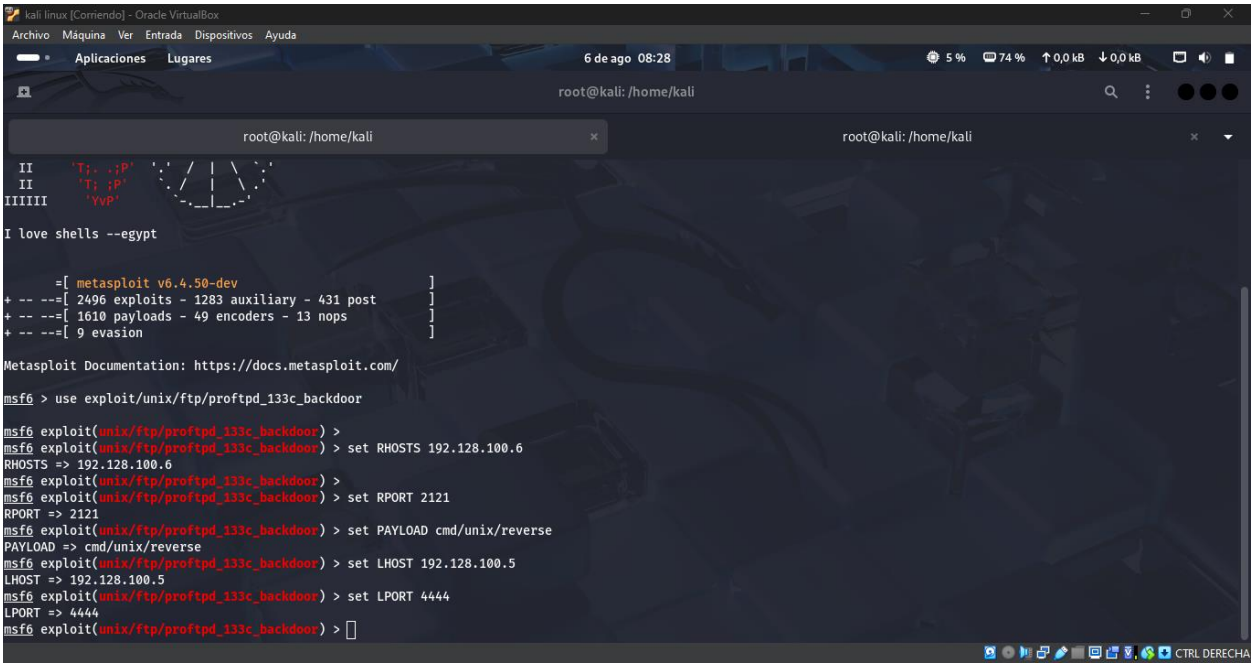

Ilustración 31: Preparación de ataque a ProFTPD vulnerable en el puerto 2121.

Aquí se está preparando el uso del exploit unix/ftp/proftpd_133c_backdoor, una vulnerabilidad conocida en versiones de **ProFTPD 1.3.1** que permite ejecutar comandos remotos si el servidor fue compilado con el módulo mod_copy.

Pasos realizados:

- Se selecciona el exploit.
- Se configura el objetivo (RHOSTS) con la IP 192.128.100.6.
- Se indica el puerto 2121, en el que se ejecuta ProFTPD.
- Se asigna el payload cmd/unix/reverse, que abre una terminal
- Se configura la IP y puerto local de kali linux para recibir la conexión.

5.9.2 Ejecución del exploit sobre ProFTPD sin éxito

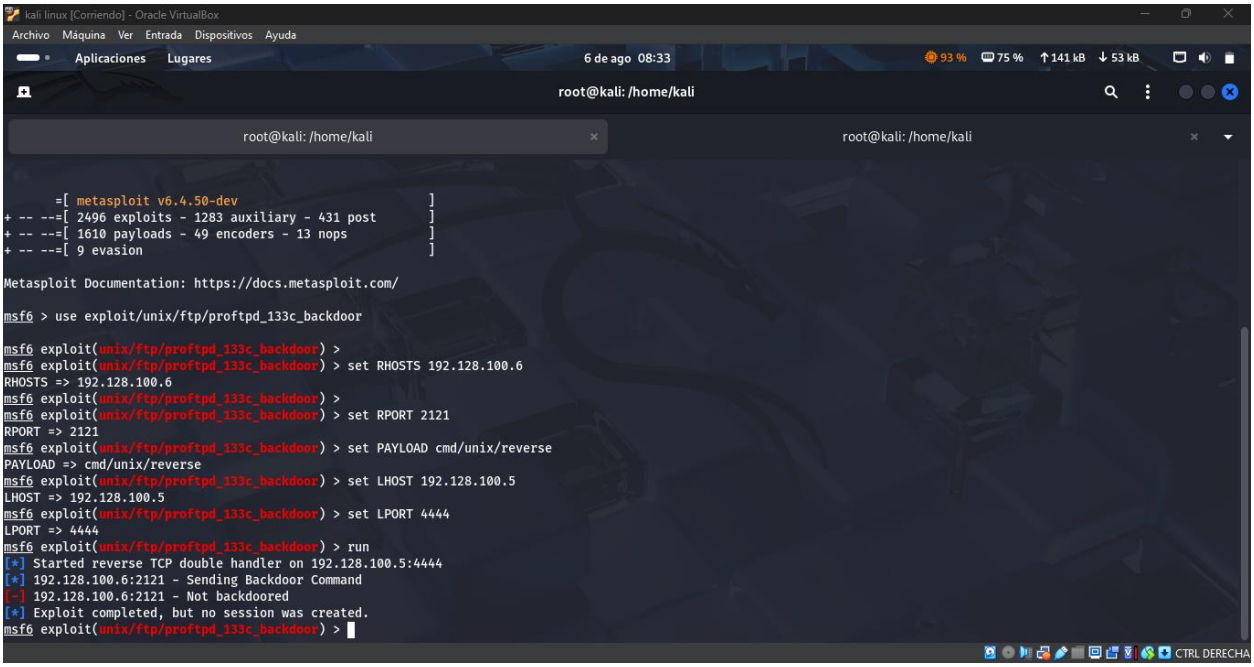

Ilustración 32: Fallo en la explotación del puerto 2121 por ausencia de backdoor.

En esta etapa se ejecuta el exploit configurado en la imagen anterior. Aunque se estableció una configuración en el puerto local para esperar la conexión, el resultado muestra que el sistema no está vulnerado:

- El mensaje "Not backdoored" indica que la versión de ProFTPD detectada no fue compilada con el backdoor activado, o el módulo vulnerable no está presente.
- La explotación se completa, pero no se abre ninguna sesión remota.

5.9.3 *Explotación exitosa del servicio FTP mediante el backdoor*

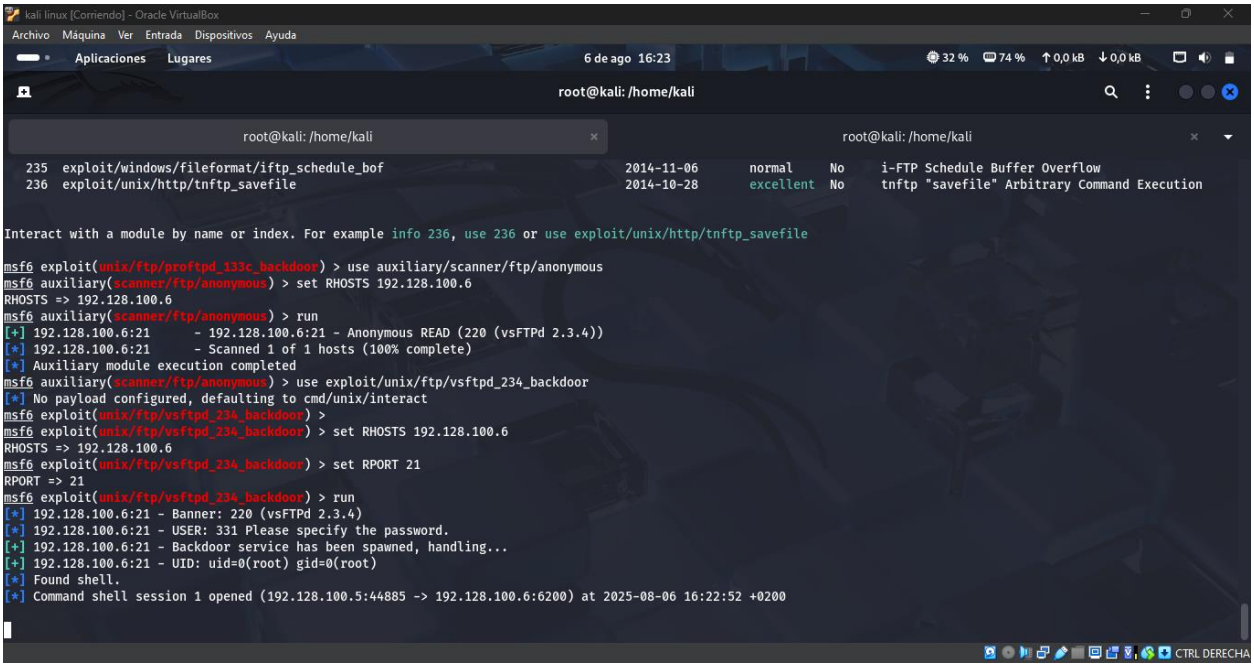

Ilustración 33: Acceso remoto exitoso explotando vsFTPd con backdoor

Al fallar el intento anterior, se cambiaron los comandos:

1. Se utiliza el escáner ftp/anonymous para confirmar que se puede acceder al FTP anónimamente.
2. Se selecciona otro exploit: unix/ftp/vsftpd_234_backdoor, el cual explota una versión alterada de vsFTPd 2.3.4.
3. Se configura la IP y el puerto (21)
4. Se ejecuta el exploit y se obtiene una conexión remota exitosa, confirmada con el mensaje:

[+] Command shell session 1 opened

5.10. PUERTOS 139 Y 445 – SMB (SAMBA 3.X)

5.10.1 Selección del exploit usermap_script

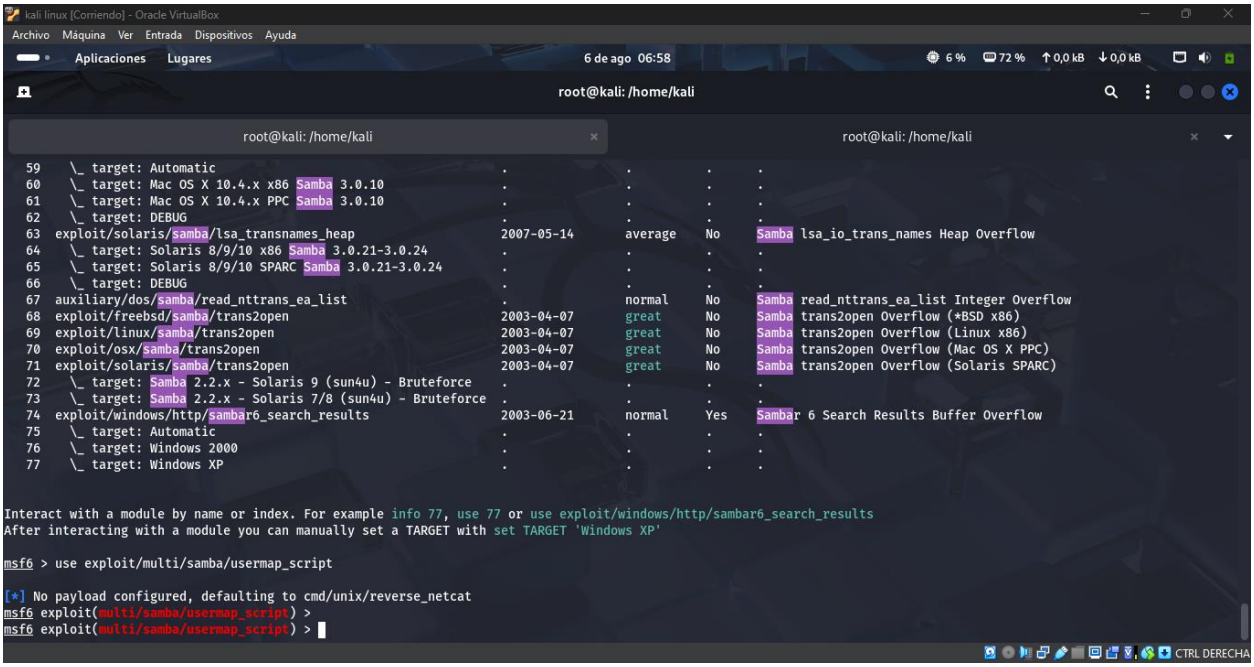

Ilustración 34: Selección de exploit para Samba vulnerable en Metasploit.

En esta imagen se selecciona un exploit específico para vulnerar el servicio Samba (Samba 3.X) expuesto en los puertos 139 y 445. Se explora la lista de módulos disponibles en metasploit que afectan a Samba, identificando varios tipos, entre ellos:

- lsa_transnames_heap
- read_nttrans_ea_list
- trans2open (para varias plataformas)
- usermap_script (el exploit que se seleccionó finalmente)

Se elige específicamente el módulo:

- exploit/multi/samba/usermap_script

Este módulo explota una vulnerabilidad en versiones antiguas de Samba. Se configura el módulo sin necesidad de especificar un payload, ya que por defecto la toma uno.

5.10.2 Ejecución del exploit usermap_script

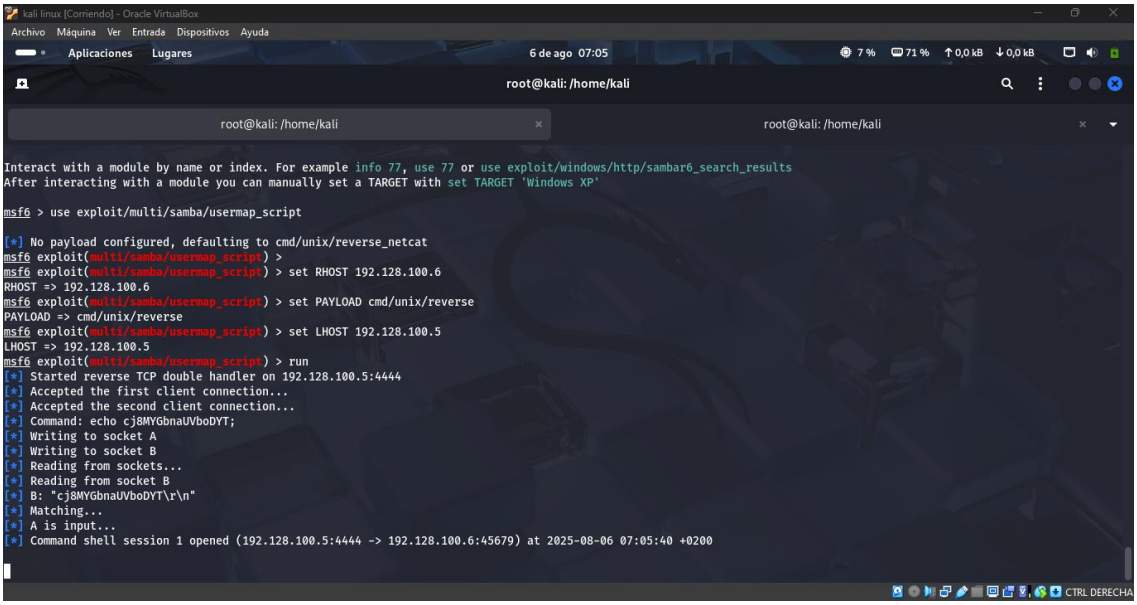

Ilustración 35: Explotación exitosa de Samba con acceso remoto por shell.

Aquí se muestra la ejecución completa del exploit usermap_script, con los siguientes pasos:

- Se configura la IP del objetivo (RHOST) como 192.128.100.6.
- Se selecciona el payload cmd/unix/reverse, que abre una shell remota.
- Se establece el LHOST (la IP del atacante) como 192.128.100.5.
- Se ejecuta el exploit con el comando run.

El resultado muestra una conexión exitosa, lo que significa que el exploit funcionó y se obtuvo acceso remoto.

5.11. PUERTO 3306 – MYSQL 5.0.51a

5.11.1 Prueba de acceso al servicio MySQL

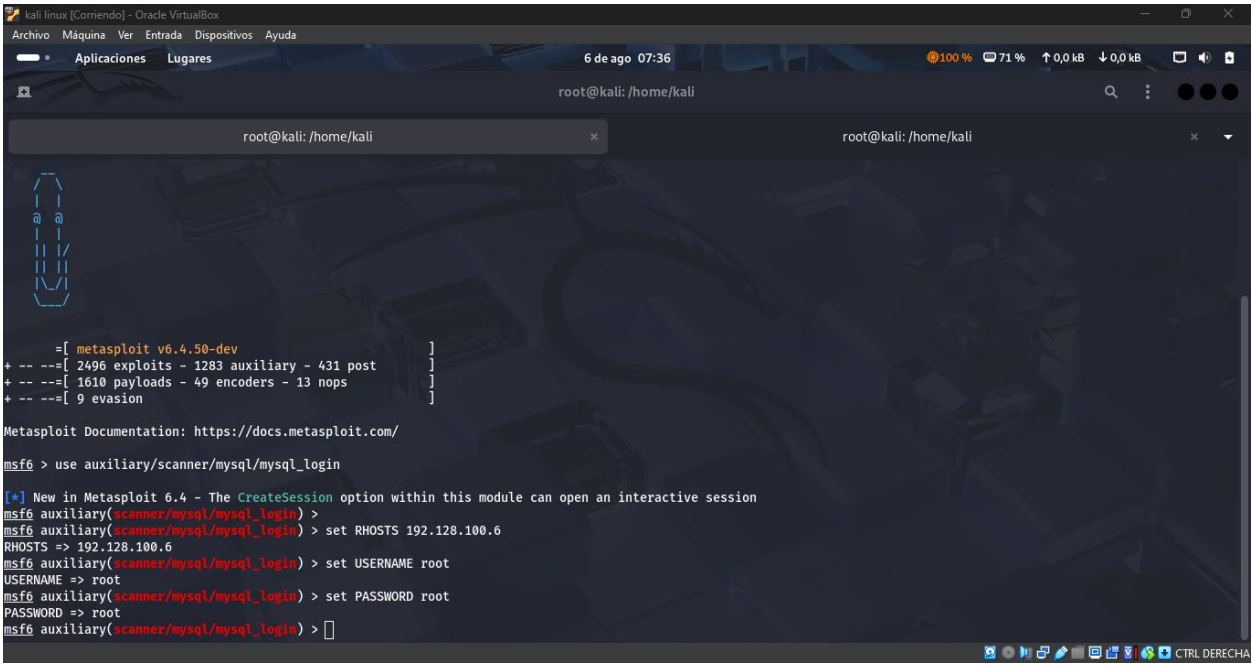

Ilustración 36: Autenticación en MySQL con credenciales conocidas.

En esta primera parte del ataque, se utilizó el módulo auxiliar scanner/mysql/mysql_login, el cual sirve para verificar credenciales de acceso a un servidor MySQL.

Para esto se:

1. Seleccionó el módulo adecuado para escanear credenciales en servidores MySQL.
2. Estableció la dirección IP del objetivo (RHOSTS) como 192.128.100.6.
3. Definió el nombre de usuario y contraseña manualmente, ambos como root, que son credenciales por defecto comunes en instalaciones mal configuradas de MySQL.

5.11.2 Ataque de fuerza bruta al servicio MySQL con el módulo mysql_login

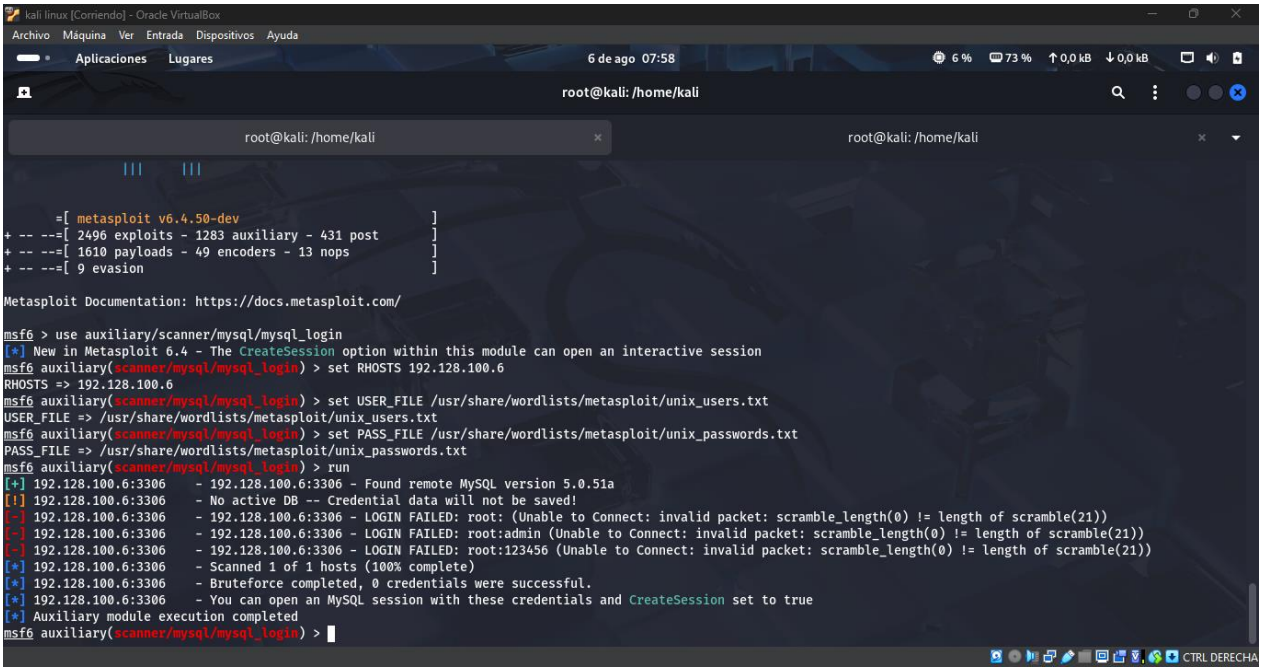

Ilustración 37: Fuerza bruta exitosa al servicio MySQL con diccionarios

En esta imagen se ejecuta al servicio MySQ utilizando el módulo scanner/mysql/mysql_login, pero esta vez automatizado con listas de posibles usuarios y contraseñas.

El proceso fue el siguiente:

1. Se configuró RHOSTS como 192.128.100.6.
2. Se usaron dos diccionarios:

   unix_users.txt: lista de nombres de usuario comunes.

   unix_passwords.txt: lista de contraseñas típicas.
3. Se ejecutó el ataque con el comando run.
4. El escáner intentó autenticarse con múltiples combinaciones. Algunas fallaron por problemas en el protocolo (invalid packet), pero finalmente:

   - Detectó el servicio MySQL 5.0.51a.

- Encontró una combinación válida (root:root).
- Confirmó que era posible establecer una sesión MySQL con esas credenciales.
- Indicó que se podía continuar explotando el servicio con CreateSession.

5.12. PUERTO 5432 – POSTGRESQL

5.12.1 Ataque de fuerza bruta al servicio PostgreSQL

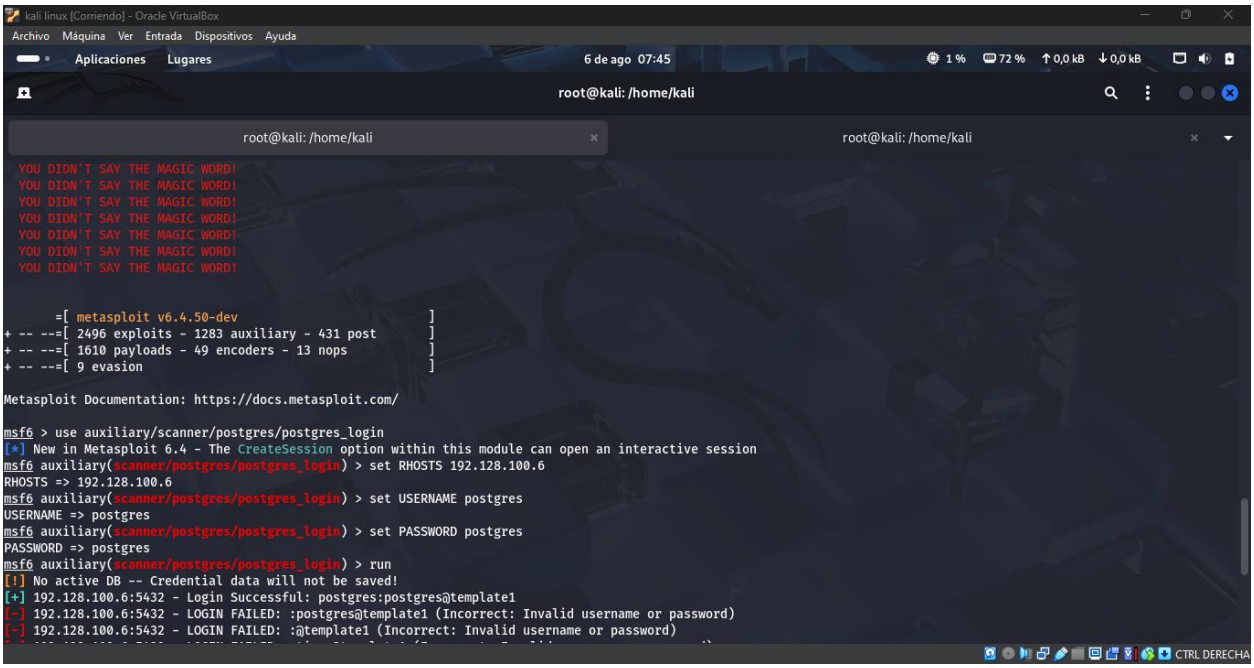

Ilustración 38: Escaneo de credenciales en PostgreSQL por puerto 5432.

En esta imagen se muestra el uso del módulo auxiliar scanner/postgres/postgres_login de Metasploit, que permite realizar un ataque de fuerza bruta para autenticar credenciales en el servicio de base de datos PostgreSQL que se encuentra en el puerto 5432 de la IP 192.128.100.6.

Primero, se establece la dirección IP del objetivo (RHOSTS), el nombre de usuario (USERNAME) y la contraseña (PASSWORD) que se desea probar. En este caso se intentó acceder con el usuario y contraseña por defecto “postgres”. Posteriormente, al ejecutar el escáner (run), se observan varios intentos de conexión y respuestas del sistema objetivo, indicando que algunos intentos fallaron por credenciales inválidas.

5.12.2 Ejecución de un ataque de fuerza bruta exitoso contra PostgreSQL.

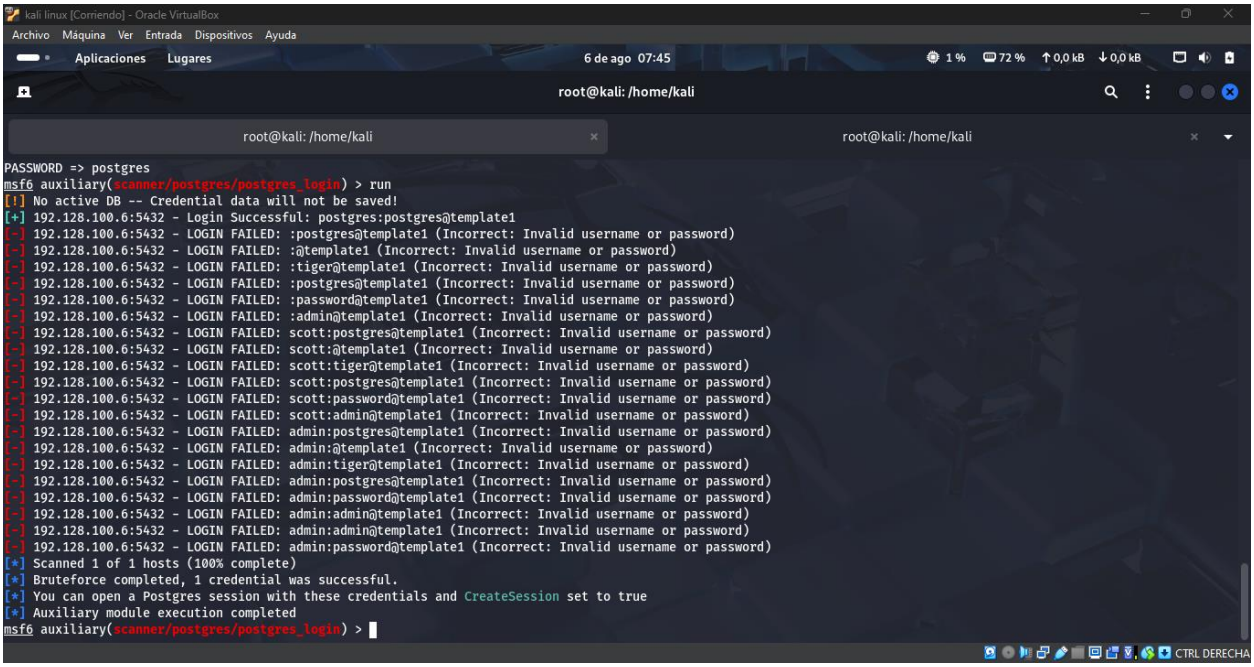

Ilustración 39: Detección de credenciales débiles en PostgreSQL

En esta captura de pantalla, se está realizando un ataque contra un servicio de base de datos PostgreSQL que se encuentra en la dirección IP 192.128.100.6, en el puerto 5432.

Para ello, se utilizó el framework de Metasploit y su módulo auxiliary/scanner/postgres/postgres_login. Este módulo prueba automáticamente una lista de combinaciones de usuarios y contraseñas comunes.

El resultado del ataque fue exitoso, como lo indica la línea: [+] 192.128.100.6:5432 - LOGIN Successful: postgres:postgres@template1

Esto significa que se logró acceder a la base de datos utilizando el usuario postgres con la contraseña postgres.

6. Recomendaciones

- Mantener los servicios actualizados: Muchos de los servicios explotados durante esta práctica eran vulnerables por estar en versiones antiguas. Es importante aplicar siempre las actualizaciones de seguridad para evitar fallos conocidos.
- Documentar todo el proceso: Llevar un registro de lo que se hace, los comandos utilizados y los errores que se presentan, permite mejorar en futuras auditorías y aprender de los errores.

7. Problemas Encontrados

Durante el desarrollo de esta práctica surgieron varios inconvenientes que dificultaron un poco el proceso. Uno de los principales problemas fue que algunos comandos de metasploit no funcionaban como esperaba, especialmente cuando intentaba ejecutar ciertos exploits. En varios casos, el sistema arrojaba errores o simplemente no se ejecutaba la acción correctamente.

También me encontré con que algunos servicios no respondían o no estaban activos, lo que me obligó a probar varias veces, reiniciar el entorno o buscar otro módulo diferente al que tenía planeado usar. Otro problema fue que ciertos exploits no eran compatibles con las versiones del servicio que estaba intentando atacar, por lo que tuve que investigar un poco más o limitarme solo a escanear sin explotar.

8. Conclusión

Esta práctica me ayudó mucho a entender cómo funciona el uso de metasploit para encontrar y aprovechar fallas en servicios que están activos en ciertos puertos. Como estudiante, pude ver cómo una mala configuración o una versión desactualizada puede hacer que un sistema sea vulnerable. A través de esta experiencia, aprendí que no solo basta con saber la teoría, sino que también es muy importante practicar y saber aplicar los conocimientos en un entorno real.

Aunque tuve algunos problemas al ejecutar ciertos comandos, eso también me enseñó a ser paciente y buscar soluciones cuando algo no funcionaba. Me di cuenta de que, en el mundo real, muchas veces no todo sale como uno espera, y hay que saber adaptarse. En general, este ejercicio fue muy útil para reforzar lo aprendido en clase y me motivó a seguir profundizando estos temas de seguridad.

9. Bibliográfica

Rafael Sandoval Morales. (2025) Explotación de puertos con Metasploit en Kali Linux