EXPLOTACIÓN DE VULNERABILIDADES SQL INJECTION EN FRISTILEAKS CON 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

EXPLOTACIÓN DE VULNERABILIDADES SQL INJECTION EN FRISTILEAKS CON 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

Este informe documenta el proceso de vulnerar la seguridad de un sitio web de "Fristi" mediante pruebas de SQL Injection. El objetivo fue identificar vulnerabilidades en la página que permitiera acceder a su base de datos, extraer información confidencial y agregar nuevos usuarios de forma no autorizada. Todo se realizó en un entorno controlado con máquinas virtuales, demostrando así la importancia de proteger las aplicaciones web contra ataques.

2. Objetivos

2.1. Objetivo General

Realizar un laboratorio práctico aplicando SQL Injection para ingresar a la base de datos de *Fristi* y generar credenciales de acceso.

2.2. Objetivos Específicos

- Analizar la estructura de la página web "Fristi" para entender cómo funciona y dónde están sus debilidades.
- Aplicar SQL Injection al sistema y agregar dos usuarios nuevos a la base de datos.
- Acceder a la página web con las credenciales de los usuarios creados y verificar que funcionan correctamente.

3. Justificación

Justifico este trabajo como una forma práctica de aprender sobre seguridad en la red. Al explotar vulnerabilidades en "Fristi", demostramos que ningún sitio es 100% seguro si no se protege adecuadamente. Además, creamos conciencia sobre cómo un ataque de SQL Injection puede afectar la privacidad de los usuarios y la reputación de una empresa.

4. Planteamiento de la Actividad

El problema central radica en que muchas páginas web no están correctamente protegidas frente a ataques de inyección SQL. Esta falla permite que un atacante ingrese comandos no autorizados en los formularios y acceda a la base de datos. En el caso de la página *Fristi*, se evidenció que los formularios podían ser explotados para descubrir información interna y hasta crear usuarios falsos.

La actividad planteada consistió en:

- Revisar el contenido y estructura de la página *Fristi*.
- Usar inyección SQL para ingresar a la base de datos.
- Crear dos usuarios y acceder al sitio con esas credenciales.

.

5. Desarrollo del tema

5.1.Vulnerabilidad con FristiLeaks

5.1.1 Modificación de la dirección MAC para conectividad de FristiLeaks

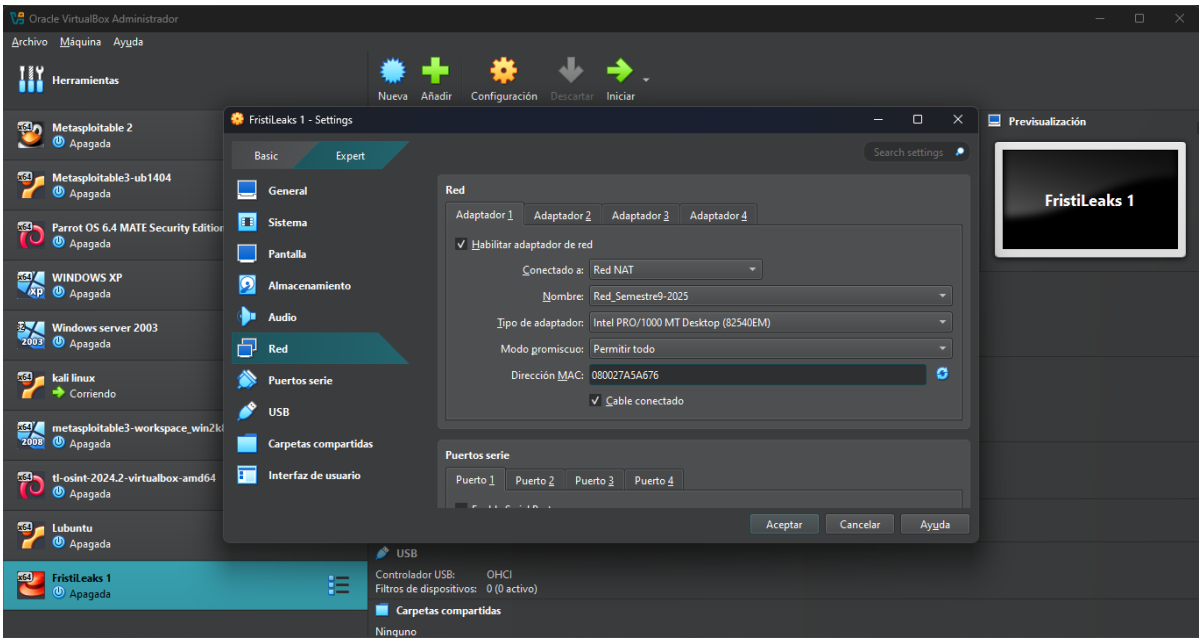

Ilustración 1. Configuración de Red de la Máquina FristiLeaks

Esta imagen muestra la configuración de la máquina virtual FristiLeaks en VirtualBox. Se ha cambiado la dirección MAC del adaptador de red a 080027A5A676 para asegurar que la máquina pueda conectarse correctamente a la red NAT.

5.1.2 Inicio del Sistema vulnerable FristiLeaks

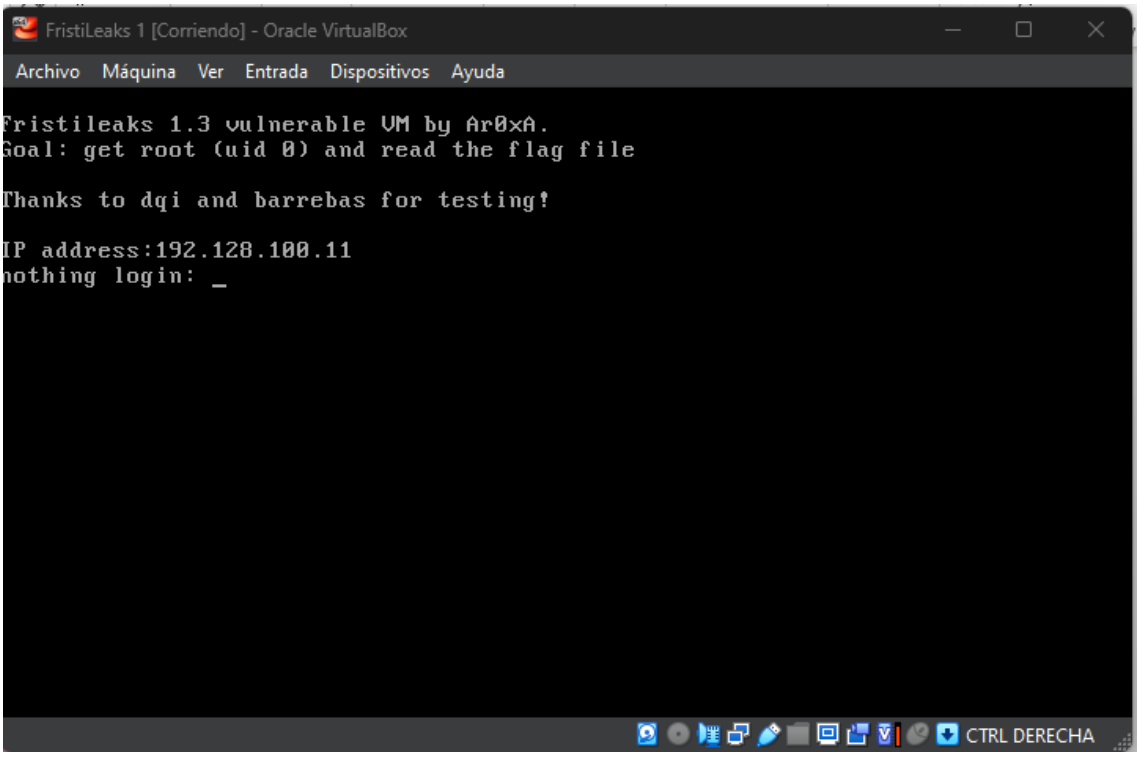

Ilustración 2. Inicio de la Máquina Virtual FristiLeaks

Esta imagen muestra la máquina virtual "FristiLeaks 1.3" en ejecución. En la pantalla se ve el prompt de inicio de sesión con el mensaje:

- IP address: 192.128.108.11 (Dirección IP asignada a la máquina).
- nothing login**:** (Esperando entrada de usuario).

5.1.3 Interfaz visual inicio de Fristi

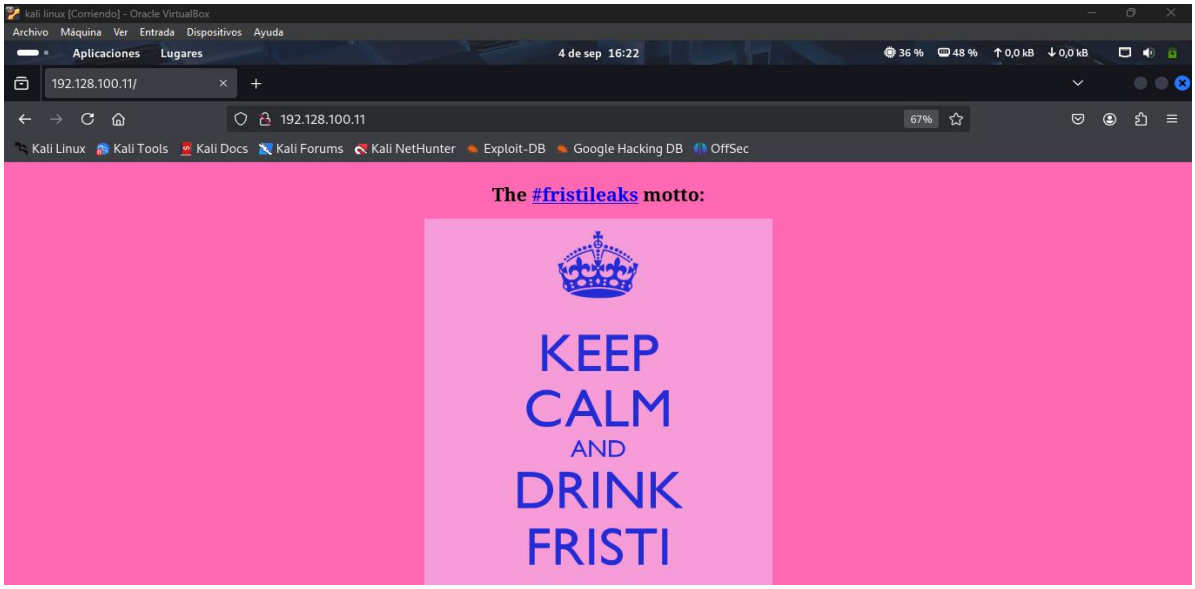

Ilustración 3. Página Principal de FristiLeaks

Esta imagen muestra la página web principal de FristiLeaks, que contiene el lema oficial: KEEP CALM AND DRINK FRISTI. Este es el sitio que se va a analizar y explotar mediante técnicas de SQL Injection.

5.1.4 Exploración del código fuente en la página FristiLeaks

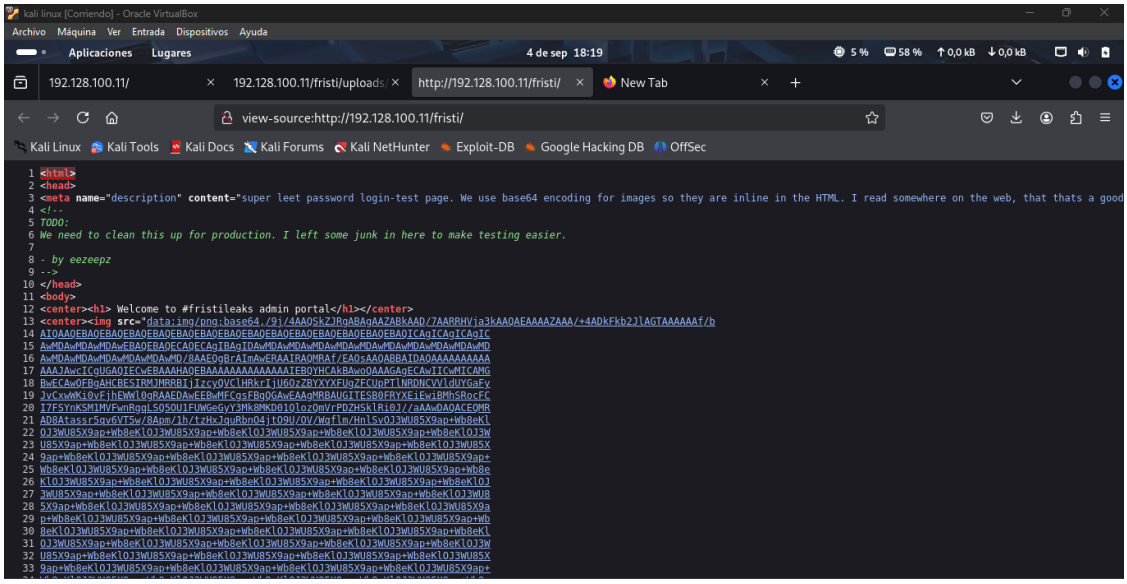

Ilustración 4. Código fuente de la página FristiLeaks

En esta imagen se observa el código fuente de la página principal de *FristiLeaks*. En él aparecen comentarios del desarrollador que indican que se utilizaron imágenes codificadas en base64 y también aparece el nombre de usuario del desarrollador.

5.1.5 Formulario de inicio de sesión en FristiLeaks

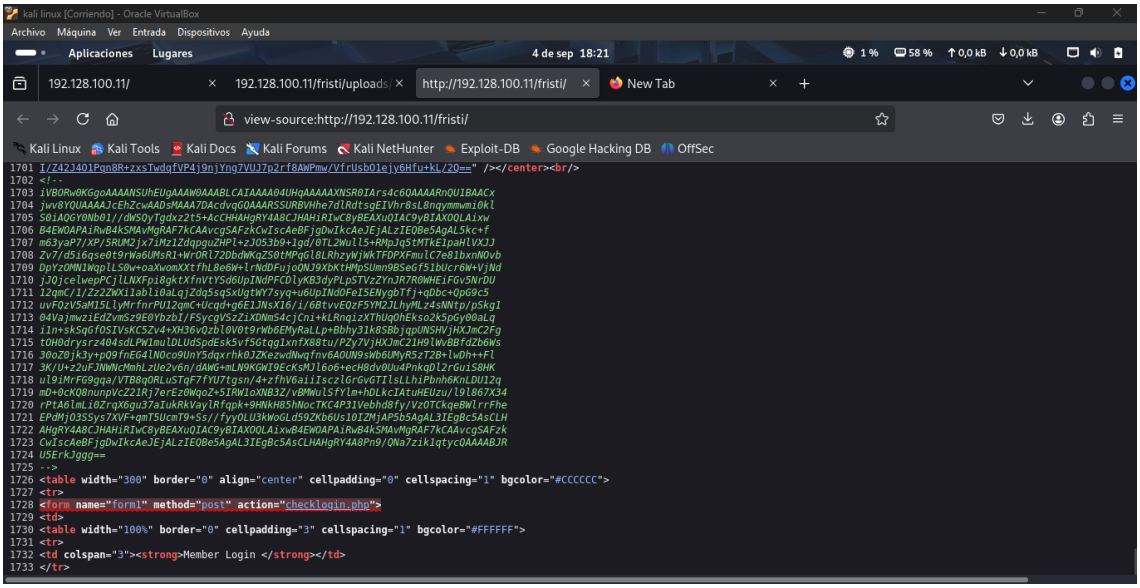

Ilustración 5. Sección del código fuente con el formulario de Login

La imagen muestra el fragmento del código fuente de *FristiLeaks* donde se encuentra definido el formulario de inicio de sesión. Se observa que el formulario envía la información por método POST al archivo checklogin.php, lo que indica que allí se validan las credenciales de acceso. Este punto es clave para realizar la inyección SQL

5.1.6 Exploración de directorios en FristiLeaks

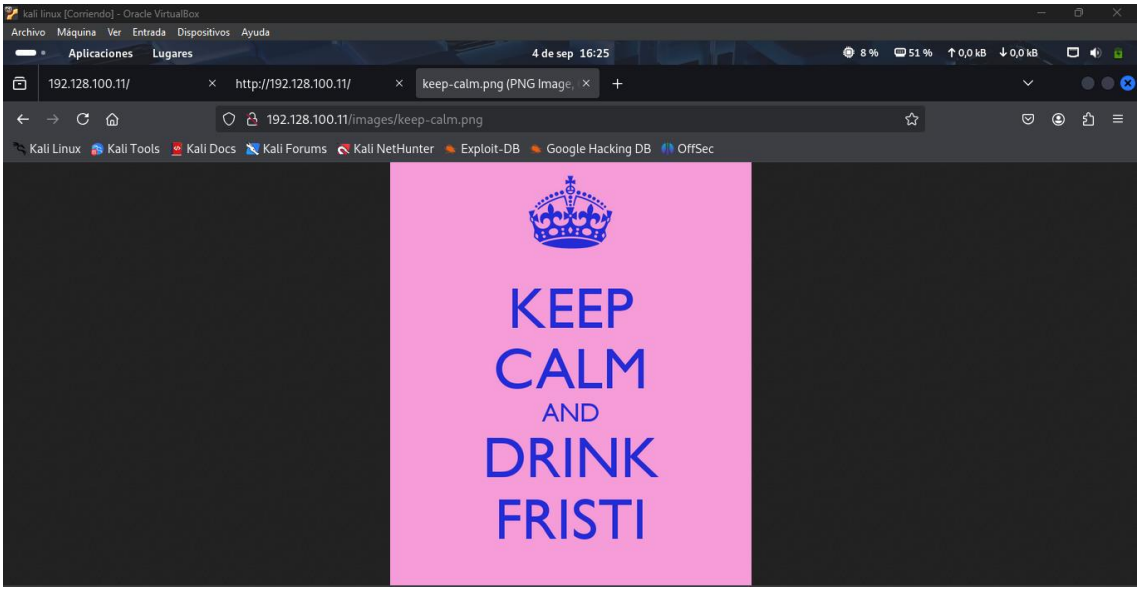

Ilustración 6. Carpeta de imágenes en el sitio FristiLeaks

En la imagen se observa cómo se accede al directorio /images del servidor web de *FristiLeaks*. Allí se encuentra alojado el archivo keep-calm.png, que corresponde al logotipo principal de la página.

5.1.7 Escaneo de directorios con Dirb

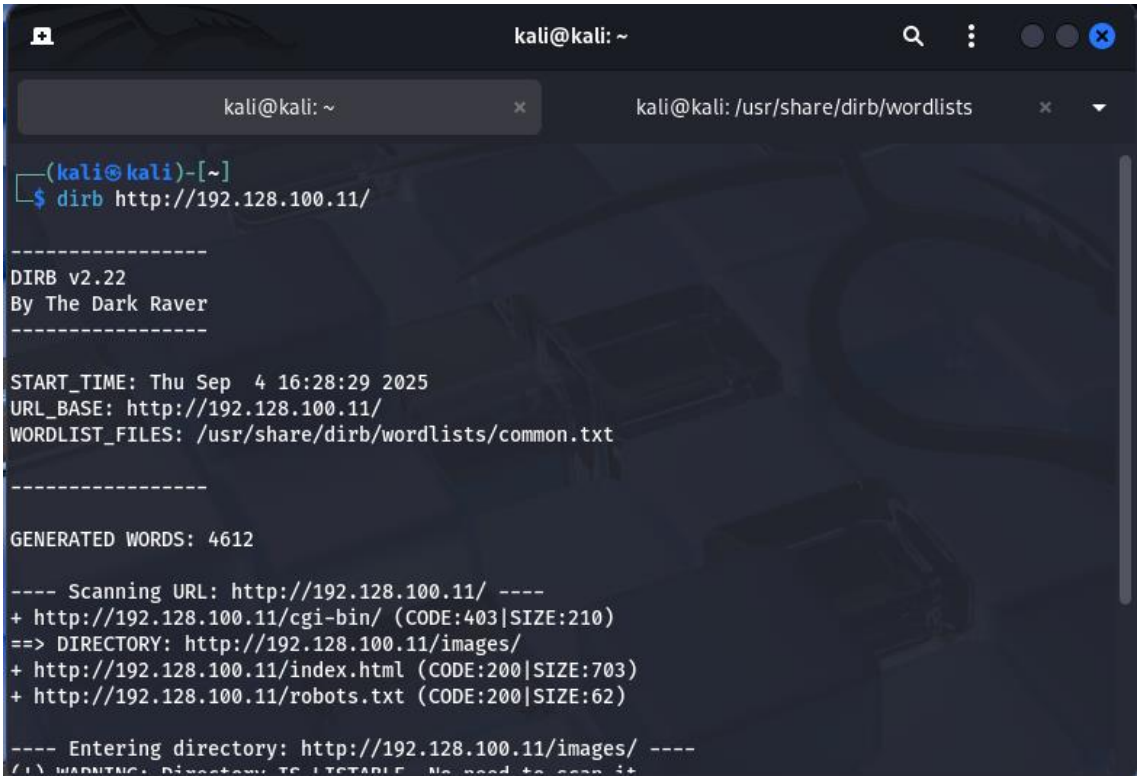

Ilustración 7. Resultados del escaneo de FristiLeaks usando Dirb

En la imagen se observa la ejecución de Dirb contra la dirección http://192.128.100.11/. El escaneo mostró directorios y archivos interesantes como /images/, index.html y robots.txt.

5.1.8 Exploración en el directorio wordlists

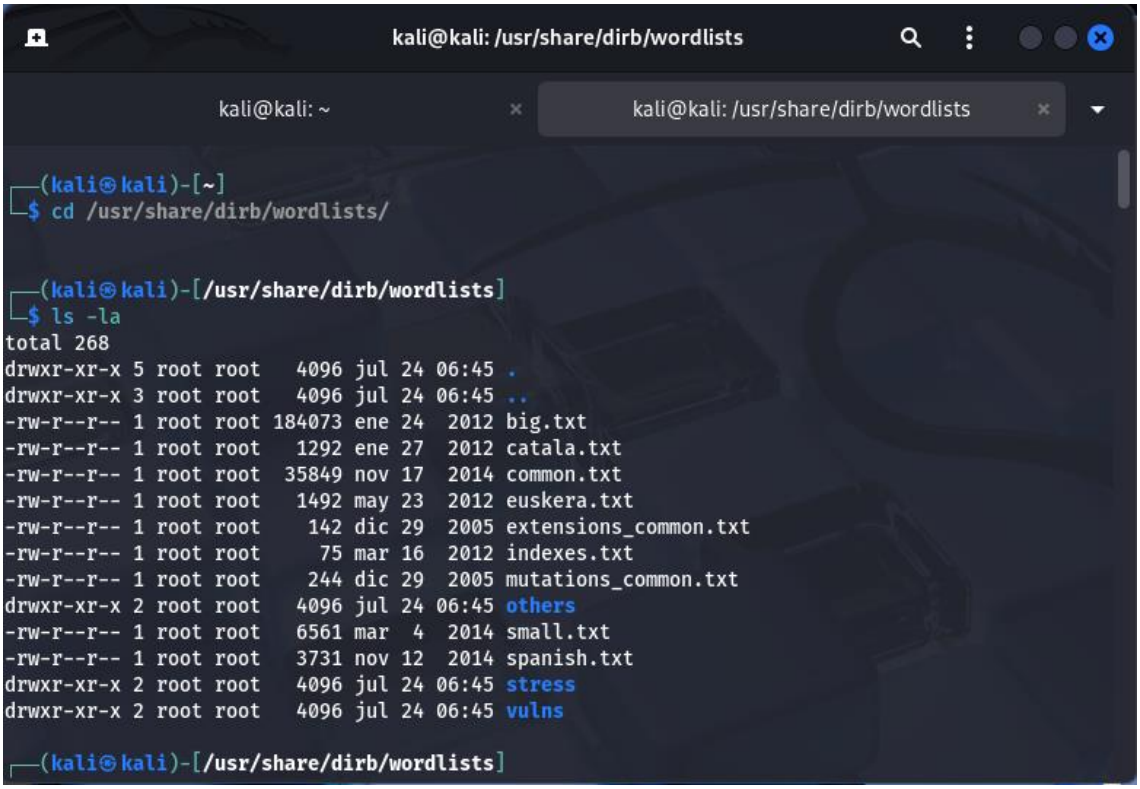

Ilustración 8. Archivos de wordlists utilizados por Dirb

En esta imagen se muestra el contenido de la carpeta /usr/share/dirb/wordlists/. Aquí se encuentran los diferentes archivos de *wordlists* que utiliza la herramienta Dirb para realizar búsquedas de directorios y archivos ocultos en sitios web.

5.1.9 Escaneo alternativo con Dirb

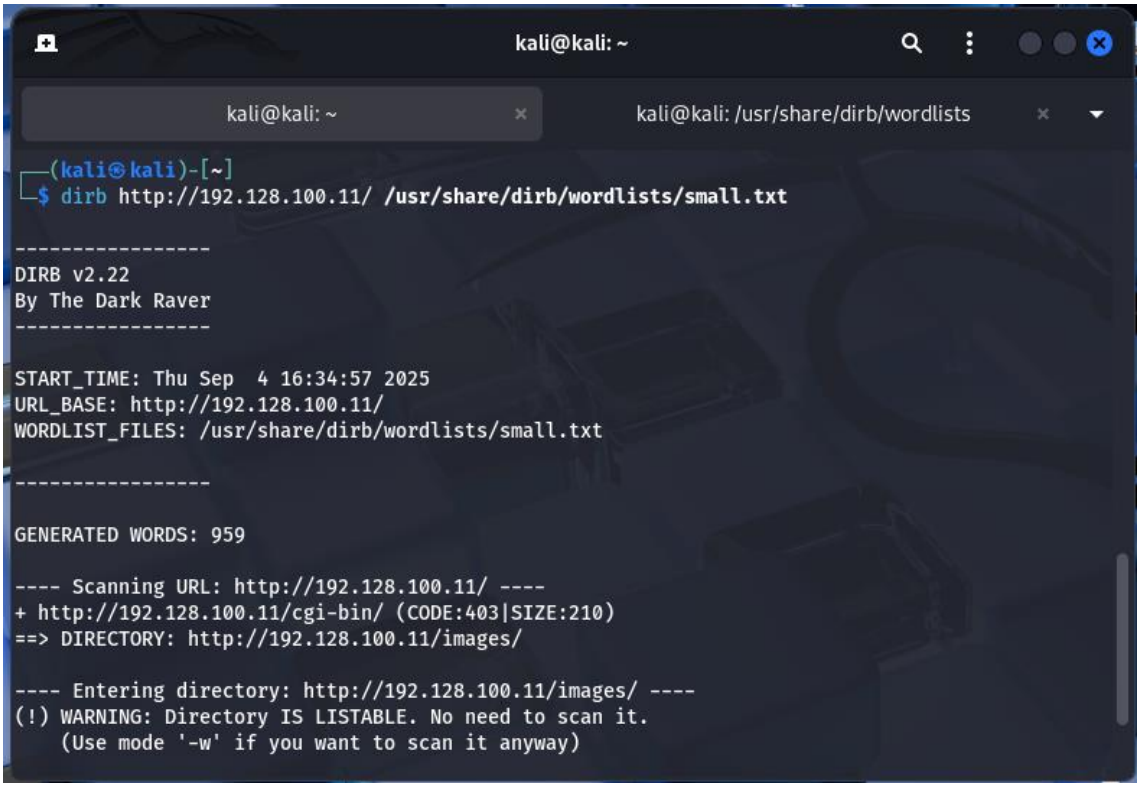

Ilustración 9. Búsqueda de directorios con la wordlist small.txt

En esta imagen se observa la ejecución de Dirb contra la dirección http://192.128.100.11/ utilizando el archivo de lista small.txt. Este escaneo generó menos combinaciones de palabras en comparación con common.txt, pero igualmente permitió identificar directorios relevantes como /images/ y /cgi-bin/.

5.1.10 Restricciones de acceso en FristiLeaks

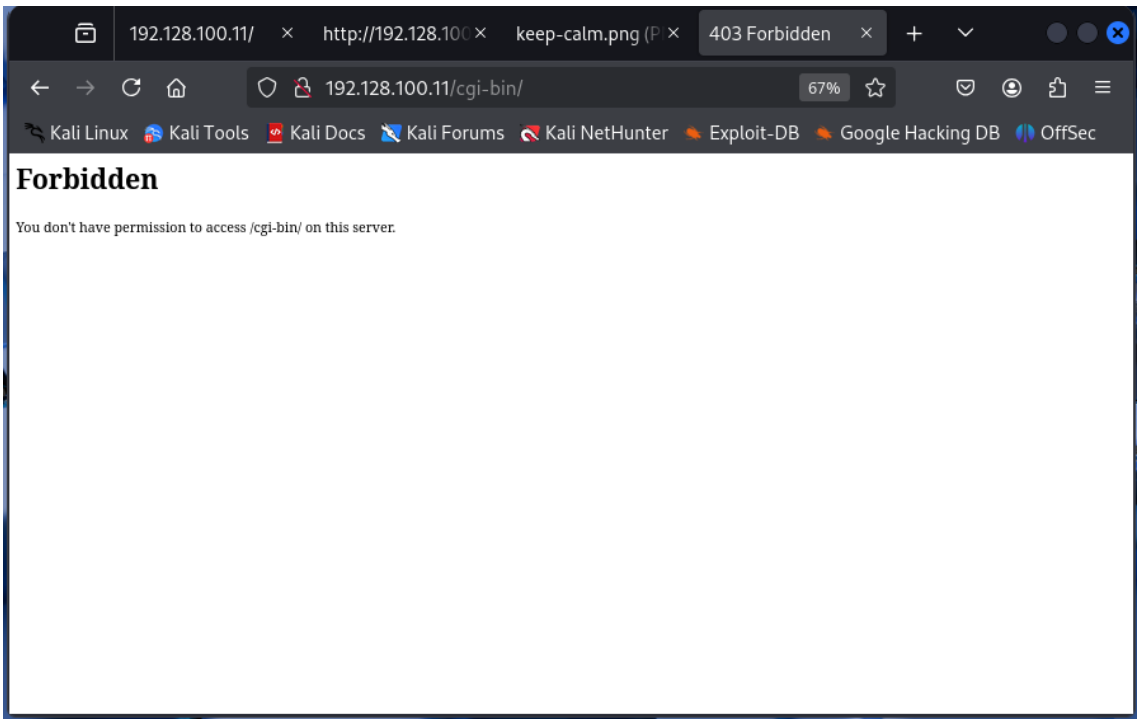

Ilustración 10. Mensaje de error 403 al acceder al directorio /cgi-bin/

En la imagen se observa que, al intentar acceder al directorio /cgi-bin/ del servidor http://192.128.100.11/, el navegador devuelve un error 403 Forbidden. Esto indica que el directorio existe, pero el servidor no permite visualizar su contenido por motivos de permisos.

5.1.11 Descubrimiento de archivos y recursos del servidor web

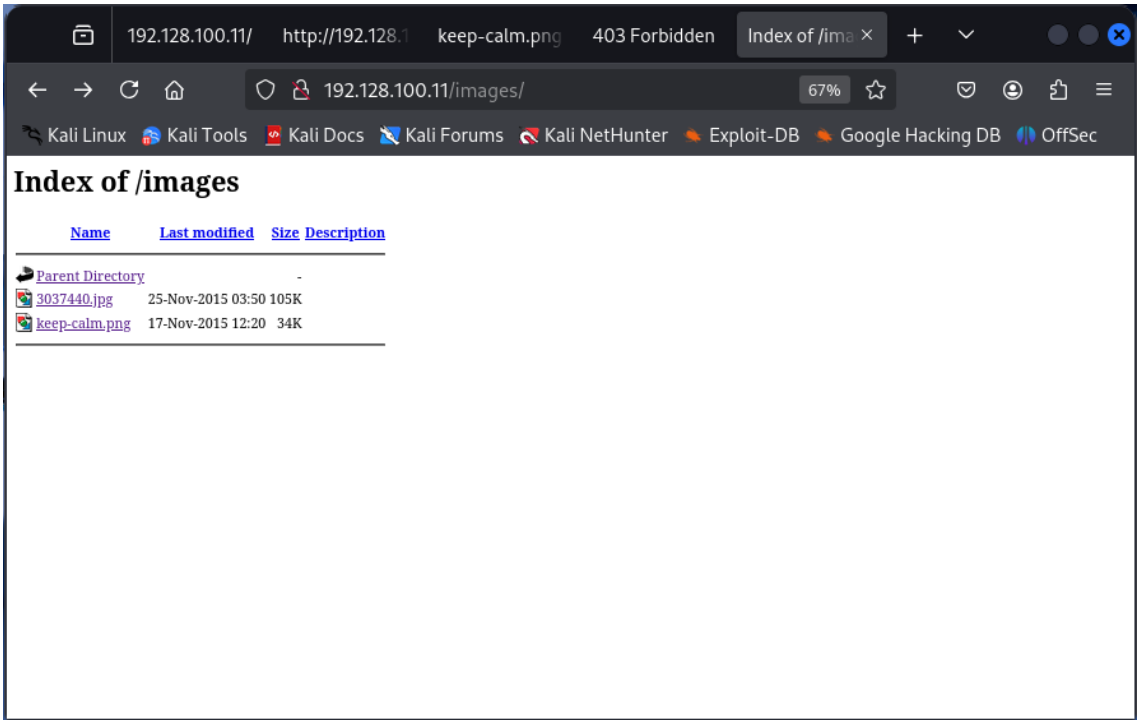

Ilustración 11. Directorio de Imágenes del Sitio FristiLeaks

Esta imagen muestra el listado del directorio /images en el servidor web de FristiLeaks.

Se revela que hay dos archivos almacenados:

- **3037440.jpg** (105 KB, modificado en noviembre de 2015)
- **keep-calm.png** (34 KB, modificado en noviembre de 2015)

5.1.12 Exploración de URLs y descubrimiento de contenido oculto

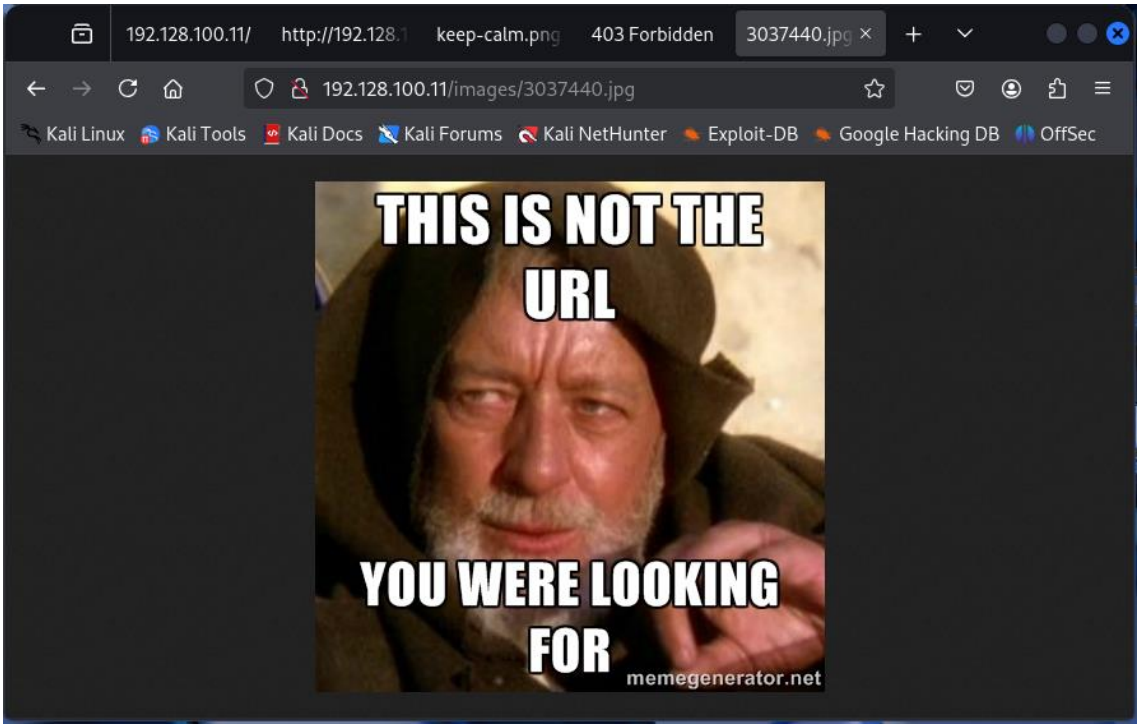

Ilustración 12. Acceso a Imagen Oculta

Esta imagen muestra un mensaje humorístico al intentar acceder a una imagen o recurso específico en el sitio. Aunque parece un error, a veces estas páginas pueden tener pistas ocultas.

5.1.13 Identificación de directorios restringidos mediante robots.txt

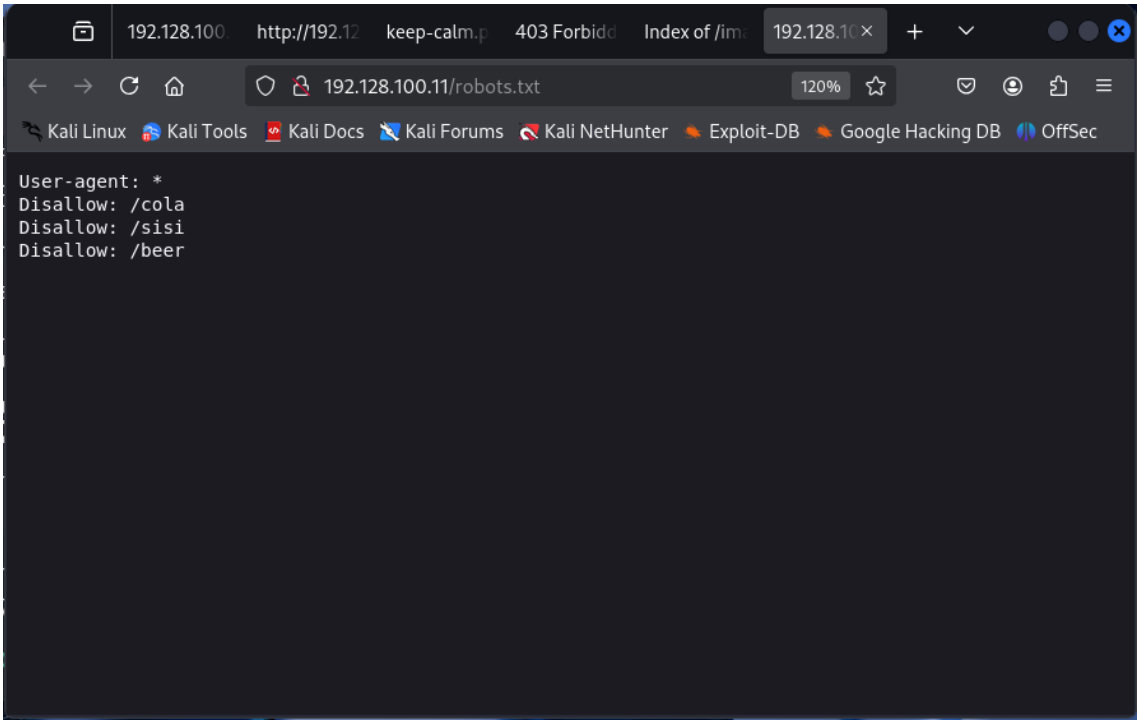

Ilustración 13. Archivo robots.txt Descubierto en FristiLeaks

Esta imagen muestra el contenido del archivo robots.txt del sitio web FristiLeaks. Este archivo se usa para indicar a los motores de búsqueda. En este caso, se revelan tres rutas restringidas:

- /cola
- /sisi
- /beer

5.1.14 Conversión de datos ocultos para descubrir credenciales

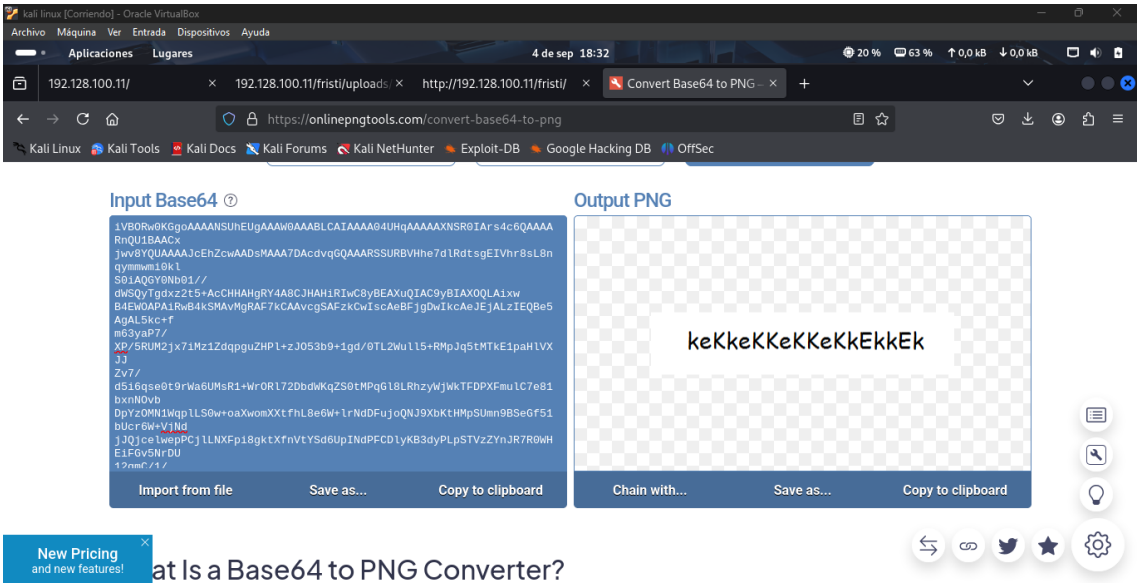

Ilustración 14. Decodificación de Contraseña en Base64 desde el Código Fuente

Esta imagen muestra el proceso de decodificación de una cadena en Base64 que fue encontrada en el código fuente de la página de FristiLeaks. El texto codificado seconvirtió a formato PNG usando una herramienta en línea.

Al decodificarla, se muestra la contraseña necesaria para acceder a la página.

5.1.15 Preparación de una shell maliciosa para obtener acceso al servidor

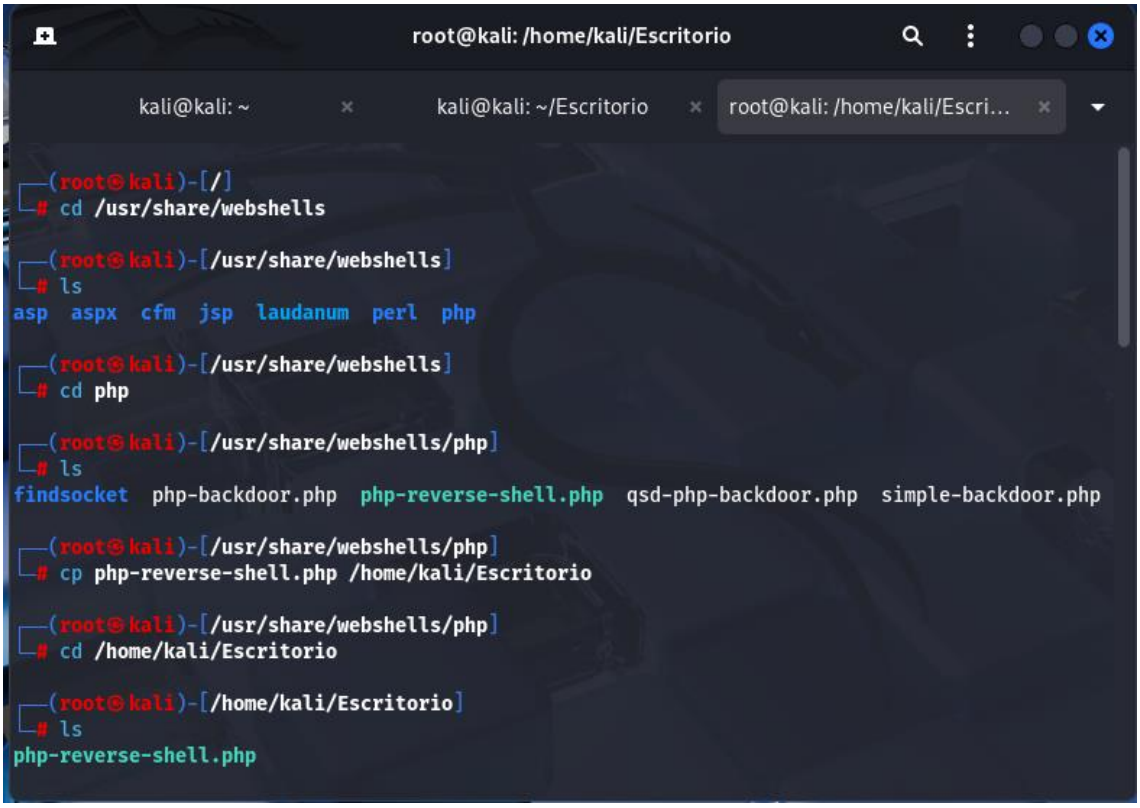

Ilustración 15. Copia de un Shell de PHP para Conexión

En esta imagen se muestra cómo se copia un archivo php-reverse-shell.php desde el directorio /usr/share/webshells/php/ al Escritorio. Este archivo es un script malicioso que, al subirse a un servidor web vulnerable, permite establecer una conexión desde el servidor hacia la máquina del atacante (Kali Linux).

5.1.16 Renombramiento y configuración del archivo malicioso

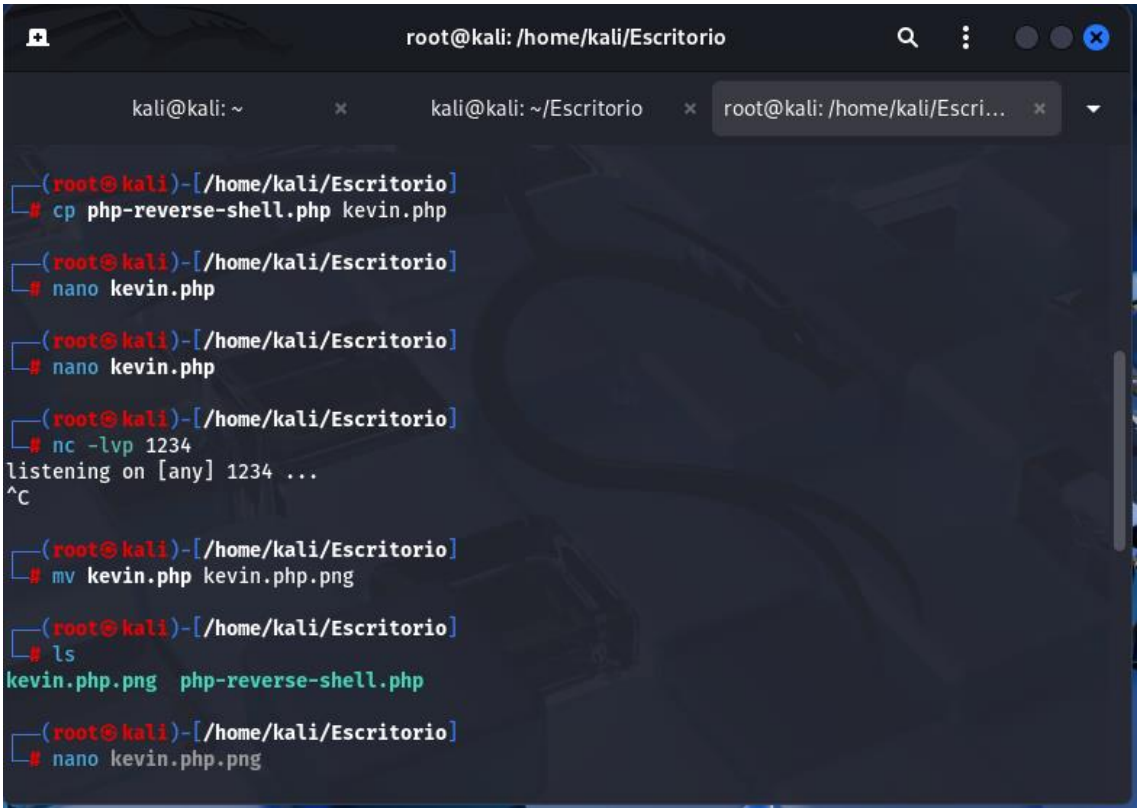

Ilustración 16. Modificación de la Shell PHP

En esta imagen se realizan varias acciones clave para preparar la shell maliciosa:

1. Se copia el archivo php-reverse-shell.php a kevin.php para trabajar sobre él.
2. Se usa el editor nano para modificar kevin.php para configurar la IP (la IP de Kali).
3. Iniciar con nc -lvp 1234 para esperar la conexión.
4. Se renombra el archivo a kevin.php.png para intentar la subida de archivos que solo permiten imágenes.

5.1.17 Personalización de la shell maliciosa con la IP

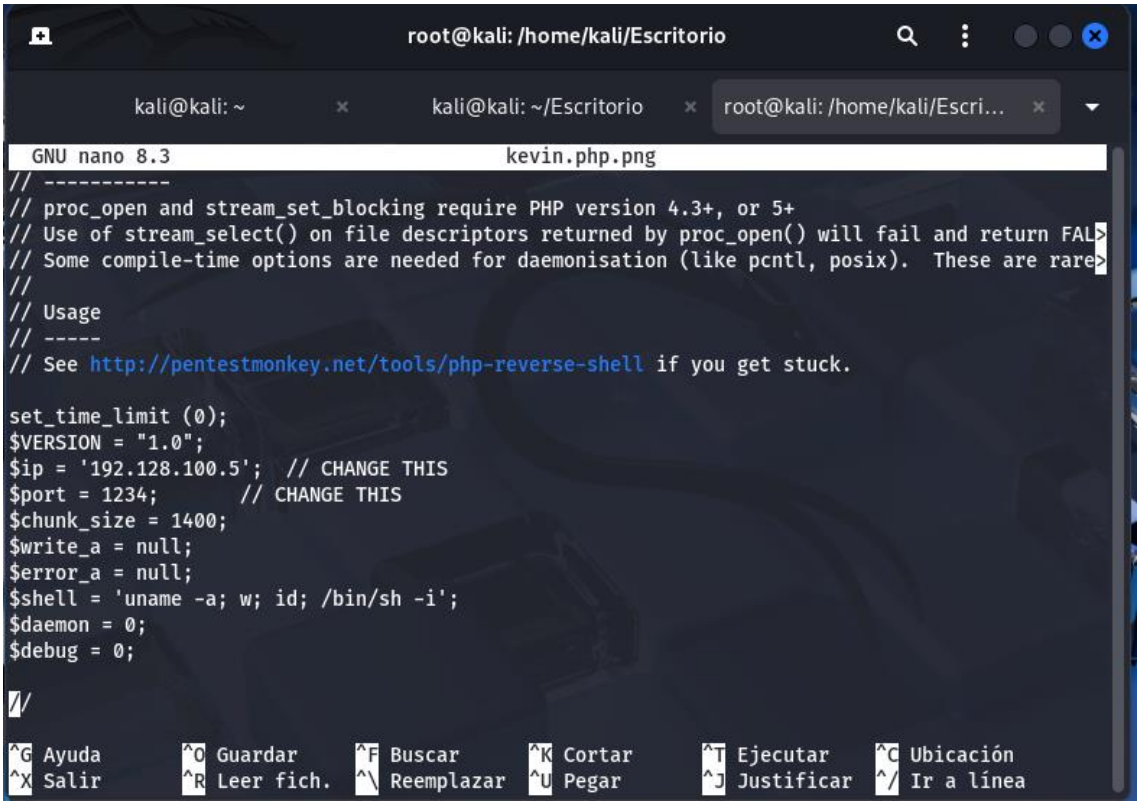

Ilustración 17. Configuración de la Shell PHP

Esta imagen muestra el contenido del archivo kevin.php.png editado con Nano. Se han modificado los parámetros críticos para la conexión:

- **IP de Kali:** 192.128.100. (La dirección de la máquina atacante).

5.1.18 Evidencia de la creación exitosa del archivo kevin.php.png

Ilustración 18. Archivo Malicioso Camuflado en el Escritorio de Kali

Esta imagen muestra el escritorio de Kali Linux con el archivo kevin.php.png . Este archivo es php-reverse-shell.php que fue renombrado y configurado.

5.1.19 Acceso con credenciales obtenidas mediante la decodificación

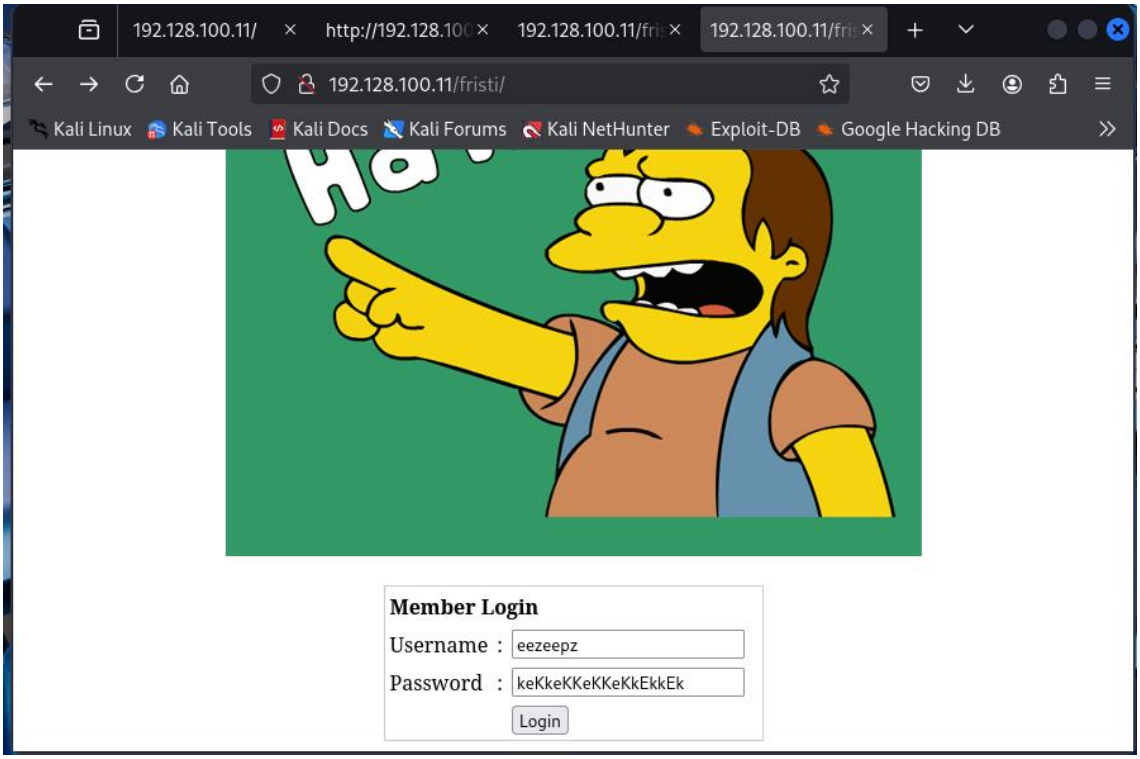

Ilustración 19. Inicio de Sesión en el Panel de Administración de FristiLeaks

Esta imagen muestra el panel de inicio de sesión de FristiLeaks donde se han ingresado las credenciales:

- **Usuario:** eezeepz
- **Contraseña:** keKkeKkeKkeKkeKkeKkeK

Estas credenciales fueron obtenidas al decodificar la cadena Base64 encontrada en el código fuente de la página.

5.1.20 Confirmación de inicio de sesión

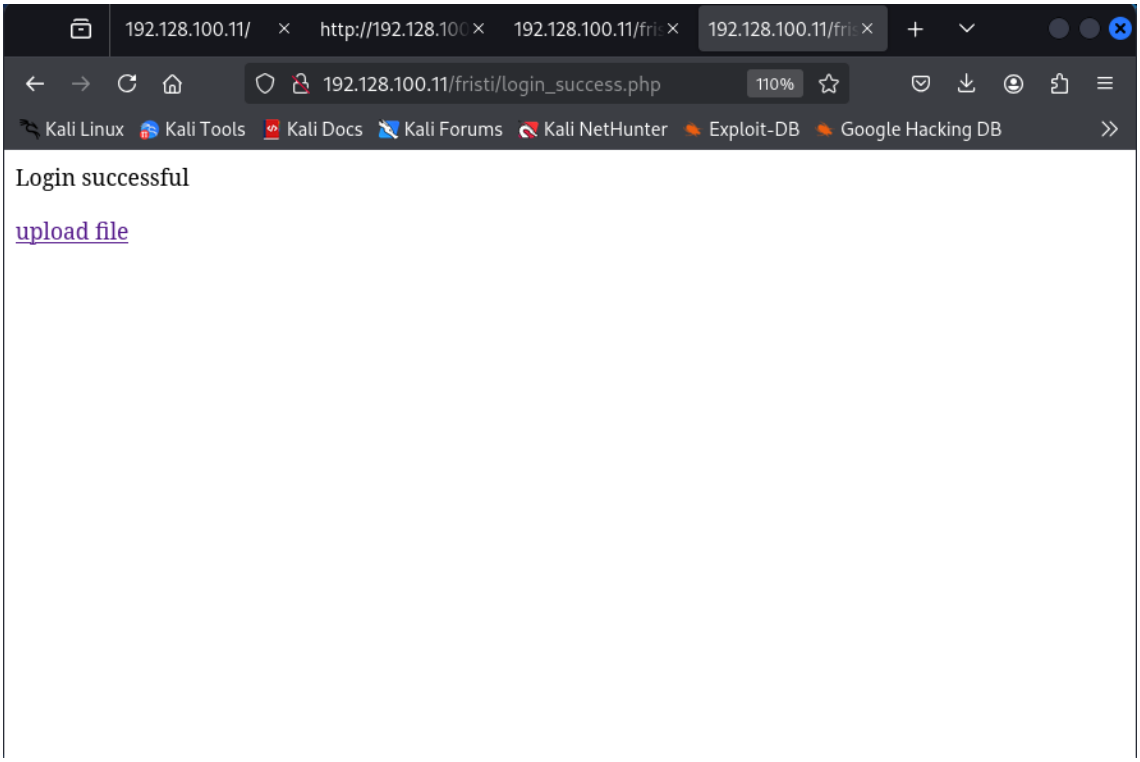

Ilustración 20. Acceso Exitoso al Panel de Administración de FristiLeaks

Esta imagen muestra un mensaje Inicio de sesión exitoso, seguido de la opción upload file (subir archivo). Esto confirma que las credenciales obtenidas (eezeepz y la contraseña decodificada) son válidas y permiten el acceso al panel de administración del sitio.

5.1.21 Interfaz para subir archivos al servidor de FristiLeaks

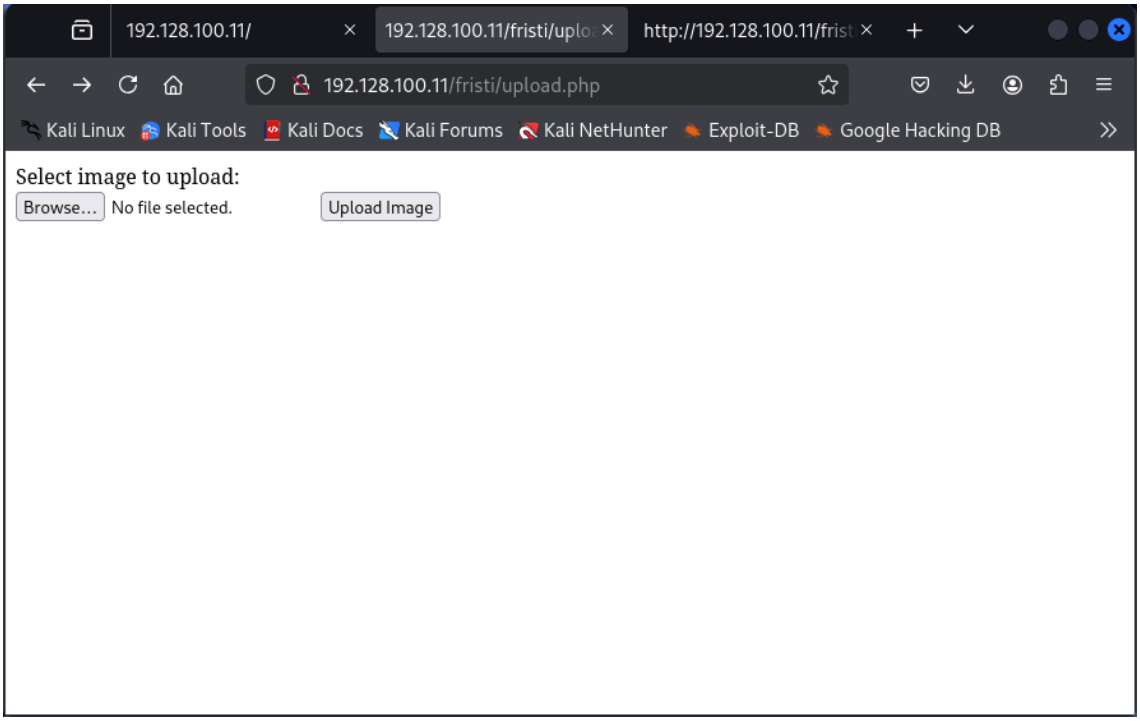

Ilustración 21. Carga de Imágenes en el Panel de Administración

Esta imagen muestra el formulario de carga de archivos en el panel de administración de FristiLeaks. La interfaz incluye:

- Un botón "Browse..." para seleccionar un archivo desde el equipo.
- Un botón "Upload Image" para enviar el archivo al servidor.

5.1.22 Eligiendo el archivo "kevin.php.png" para subir a FristiLeaks

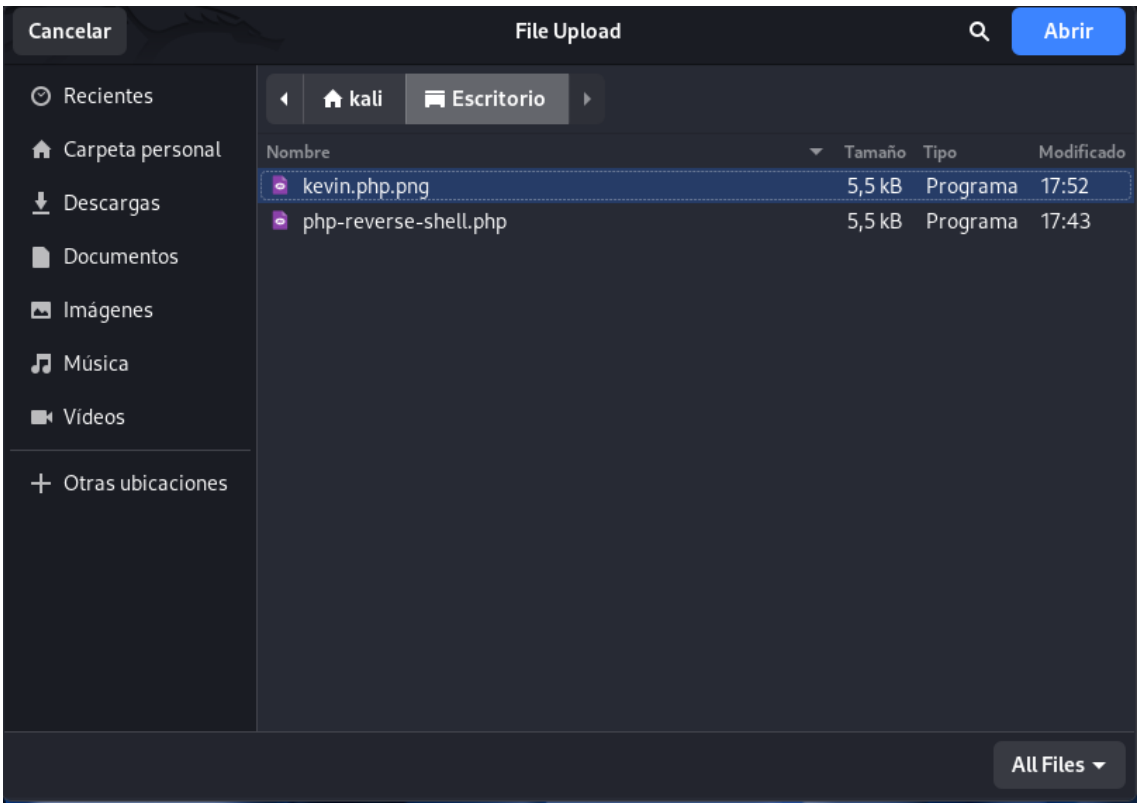

Ilustración 22. Selección del Archivo Malicioso para Cargar al Servidor

Esta imagen muestra la ventana de diálogo para seleccionar un archivo desde el sistema. Se está navegando hasta la carpeta Descargas donde se encuentra el archivo kevin.php.png.

5.1.23 Confirmación de la selección del archivo en el formulario de carga

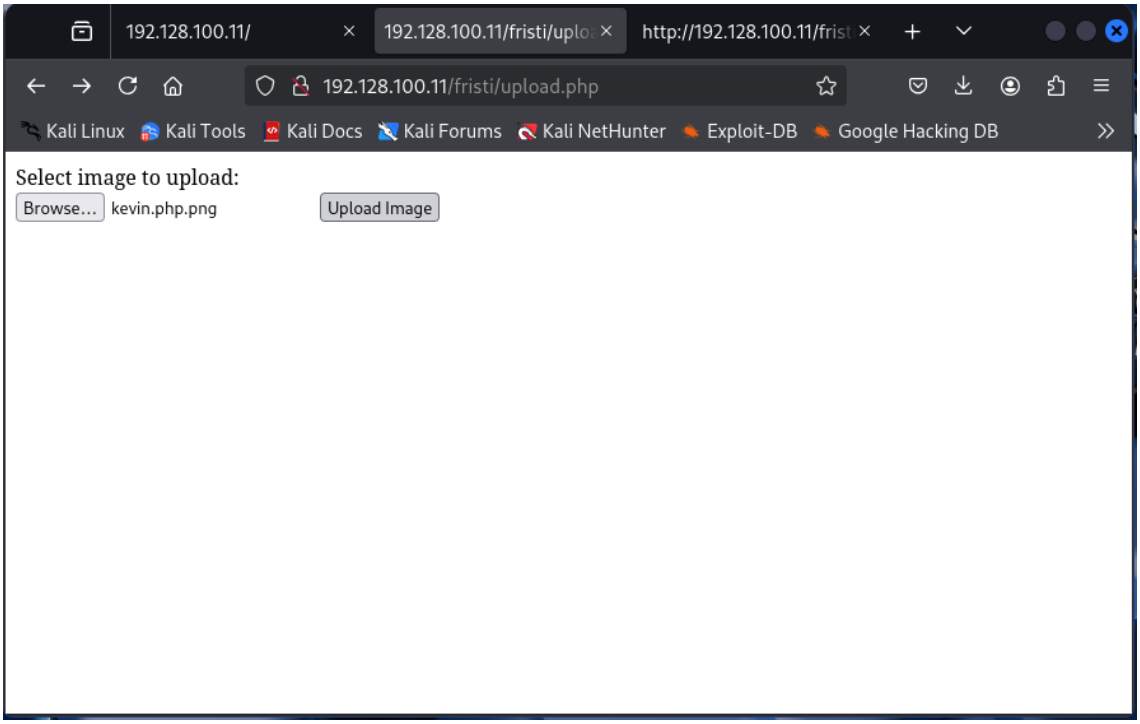

Ilustración 23. Archivo Malicioso Seleccionado para Cargar

Esta imagen muestra el formulario de carga de archivos después de seleccionar el archivo kevin.php.png. El campo que antes decía "No file selected" ahora muestra el nombre del archivo malicioso (kevin.php.png). Esto indica que el archivo está listo para ser subido al servidor de FristiLeaks al hacer clic en el botón Upload Image.

5.1.24 Confirmación de la carga del archivo al servidor de FristiLeaks

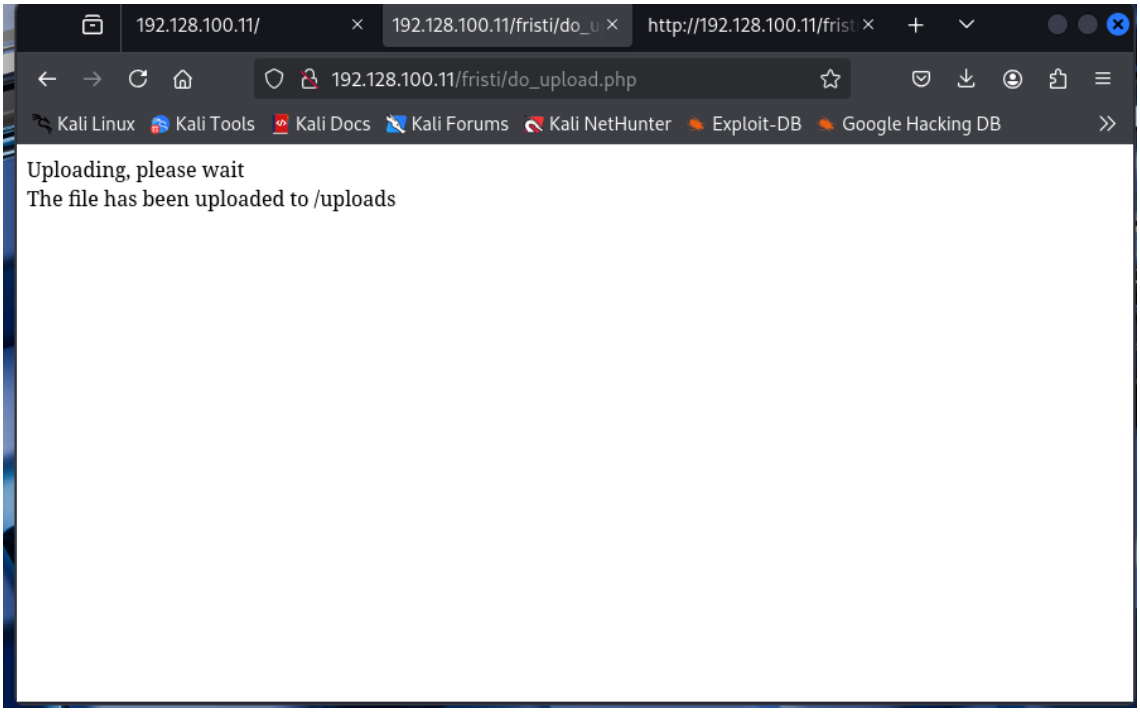

Ilustración 24. Archivo Malicioso Subido Exitosamente al Servidor

Esta imagen muestra el mensaje de éxito después de subir el archivo. Esto confirma que el archivo kevin.php.png se ha subido correctamente al servidor en la ruta /uploads. Ahora, si el servidor es vulnerable y ejecuta archivos PHP aunque tengan extensión de imagen.

5.1.25 Identificación de la ruta donde se almacenó el archivo malicioso

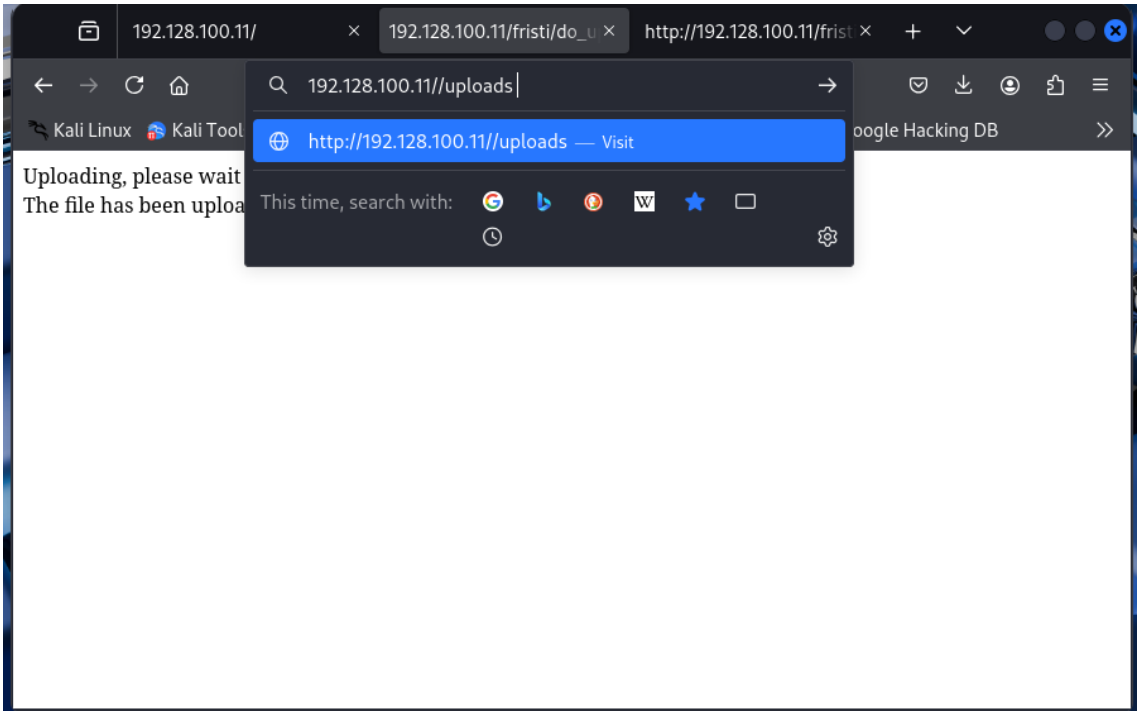

Ilustración 25. Confirmación de la Ubicación del Archivo Subido

Esta imagen muestra que el archivo se ha subido exitosamente al servidor y está disponible en la ruta:

http://192.128.100.11/uploads/

Esta URL es crucial porque al acceder a ella (por ejemplo, a http://192.128.100.11/uploads/kevin.php.png), el servidor podría ejecutar el código PHP oculto en el archivo como imagen, lo que activaría la conexión hacia Kali Linux.

5.1.26 Kali Linux en espera de la activación de la shell maliciosa

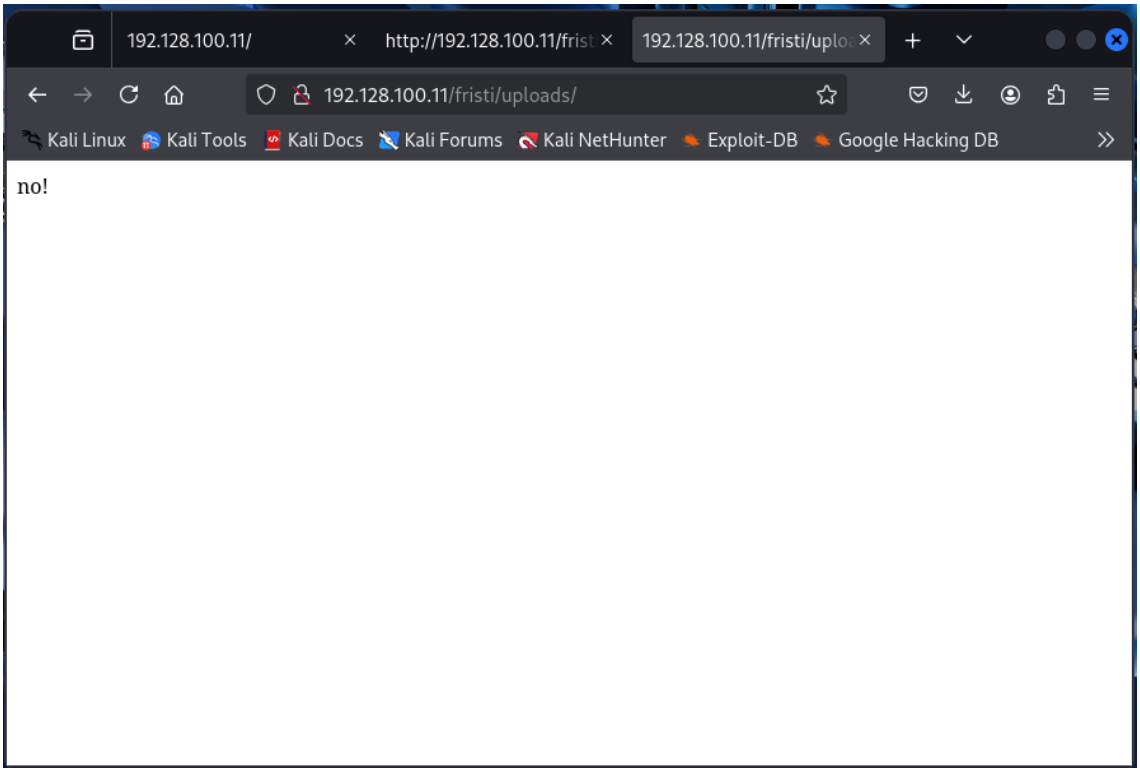

Ilustración 26. Esperando la Conexión Reversa desde el Servidor

Esta imagen muestra la interfaz de Kali Linux después de haber subido el archivo malicioso. En este punto, se debe iniciar nc -lvp 1234 para esperar la conexión desde el servidor de FristiLeaks. Cuando alguien (o el mismo atacante) acceda a la URL del archivo subido http://192.128.100.11/uploads/kevin.php.png, el código PHP se ejecutará

5.1.27 Shell obtenida en el servidor de FristiLeaks

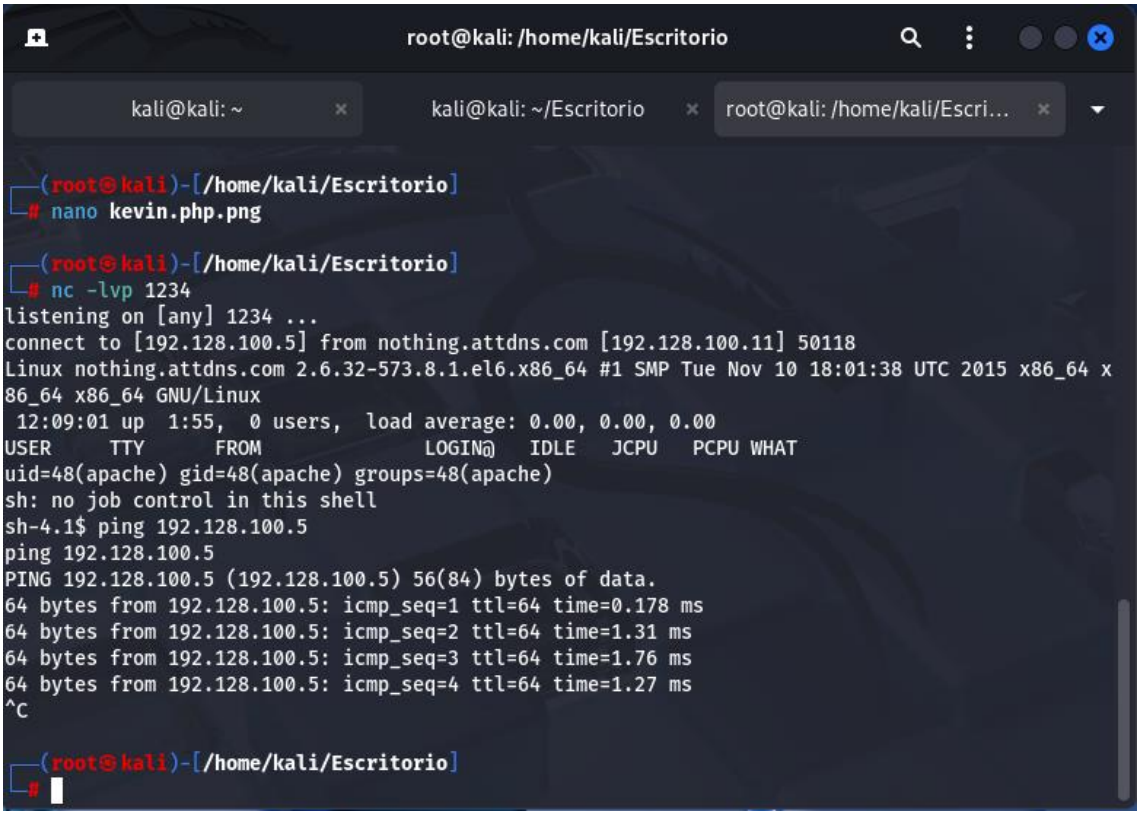

Ilustración 27. Conexión Exitosa y Acceso al Servidor

5.1.28 Ejecución del archivo malicioso y conexión exitosa

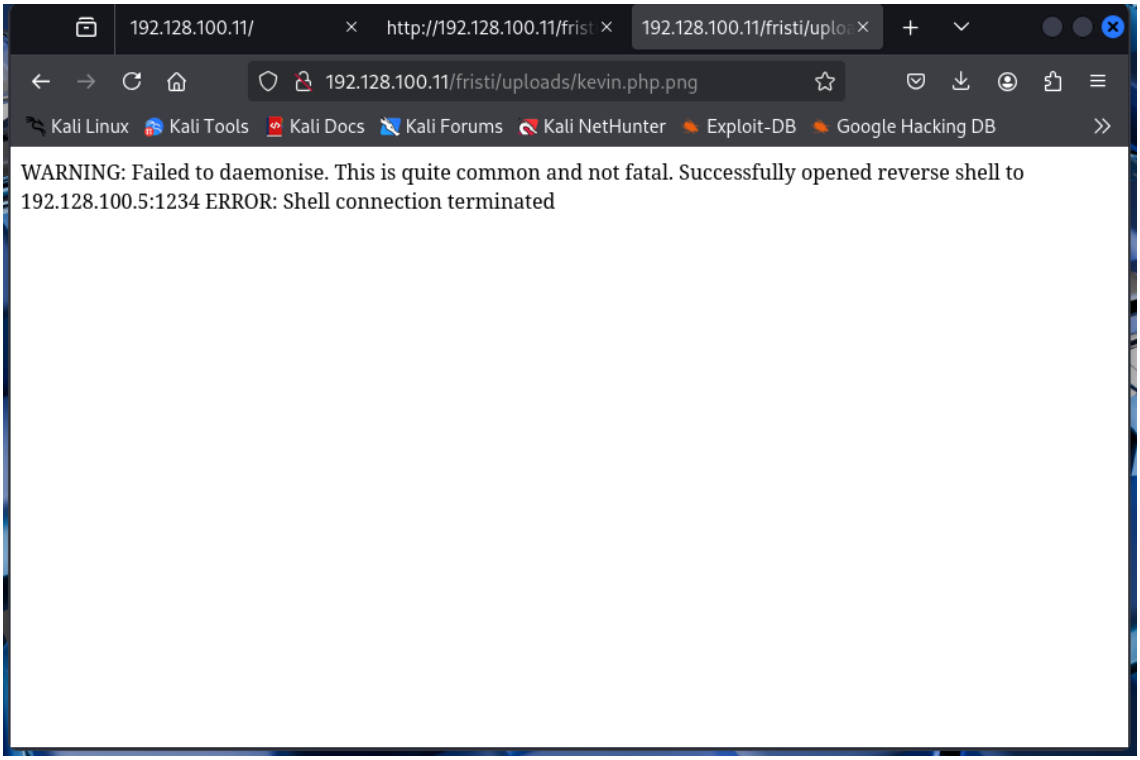

Ilustración 28. Activación de la Shell desde FristiLeaks

Esta imagen muestra el resultado de acceder a la URL del archivo malicioso subido:

http://192.128.100.11/fristi/uploads/kevin.php.png

- El servidor ejecutó el código PHP y estableció una conexión hacia Kali Linux en la IP 192.128.100.5 y puerto 1234.
- "Shell connection terminated" dice que la conexión se cerró después de un tiempo, pero confirma que la explotación fue exitosa.

5.2.Ingresar nuevo usuarios

5.2.1 Listado de archivos y directorios en el servidor de FristiLeaks

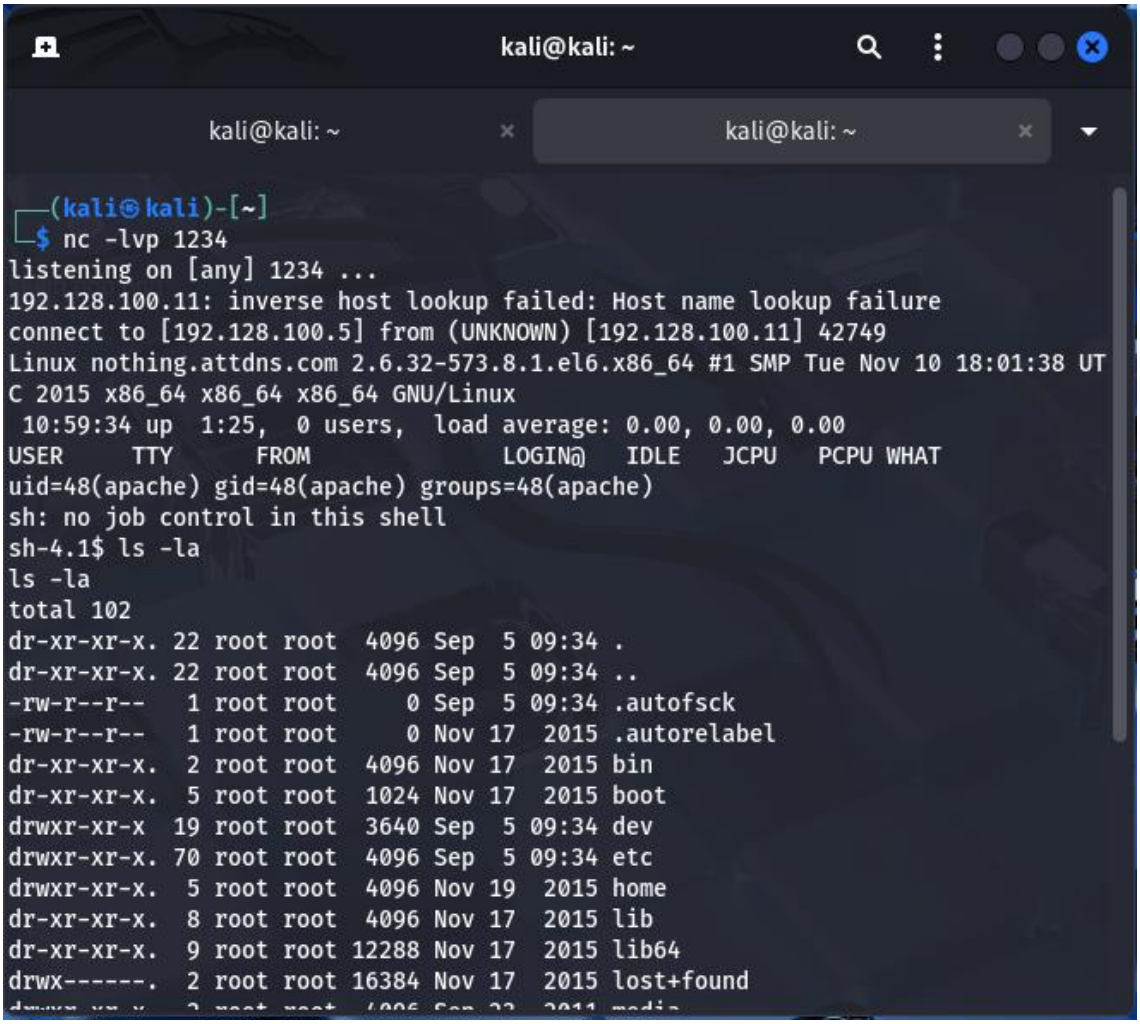

Ilustración 29. Exploración del Sistema del Servidor vulnerado

Esta imagen muestra los resultados del comando ls -la ejecutado en la shell del servidor que va ser atacado. La estructura de directorios del sistema:

- **Directorio raíz:** Se observan carpetas críticas como /bin, /boot, /etc, /home, /lib, /var, entre otras.
- La mayoría de los archivos son propiedad de root, lo que indica que hay que darle privilegios.

5.2.2 Exploración de directorios para encontrar datos sensibles

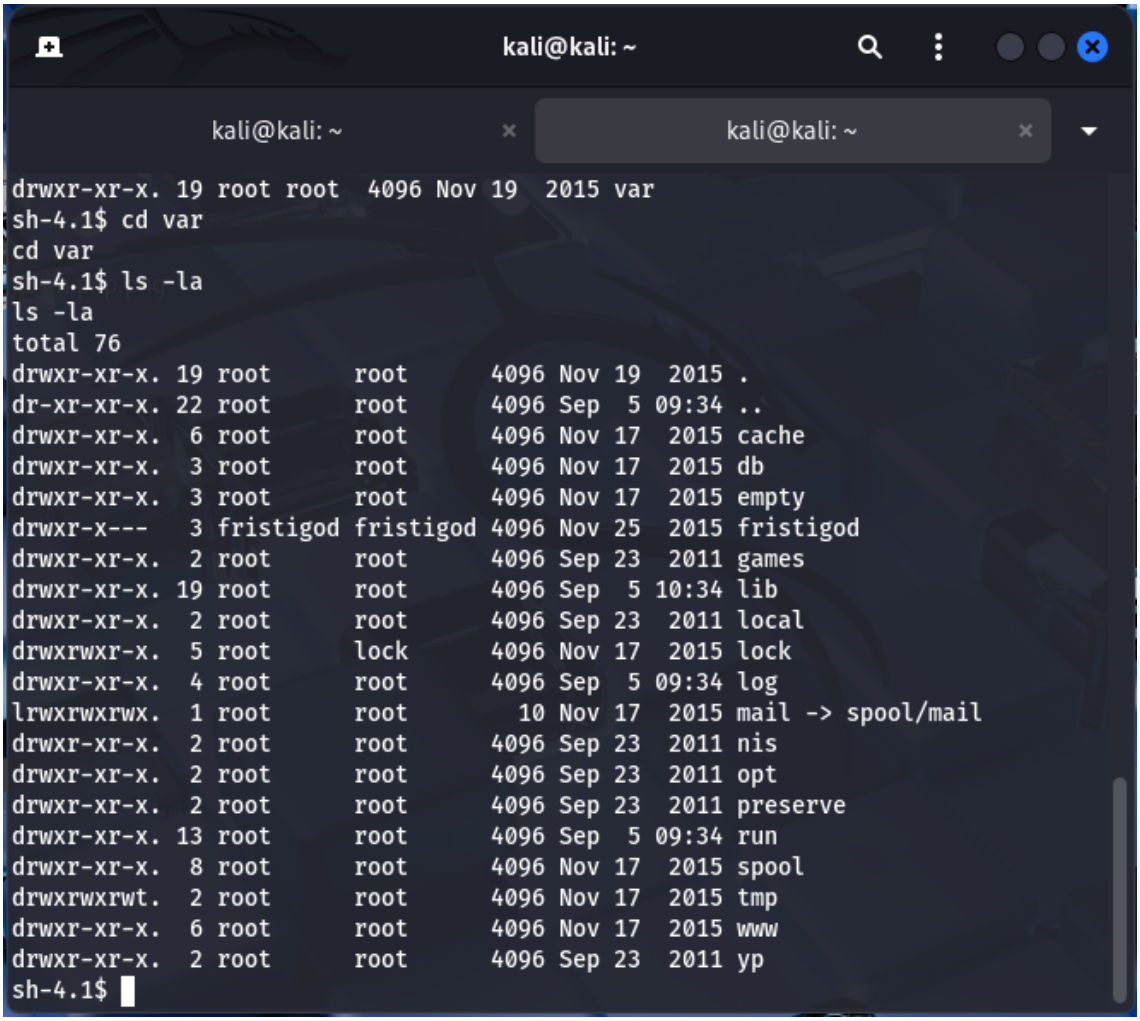

Ilustración 30. Acceso al Directorio /var del Servidor

Esta imagen muestra que se ha navegado al directorio /var del servidor FristiLeaks, un lugar clave donde se almacenan:

- Archivos de logs (registros del sistema).
- Datos de aplicaciones web (como el sitio de FristiLeaks).
- Bases de datos o archivos de configuración.

5.2.3 Descubrimiento de la estructura del sitio web de FristiLeaks

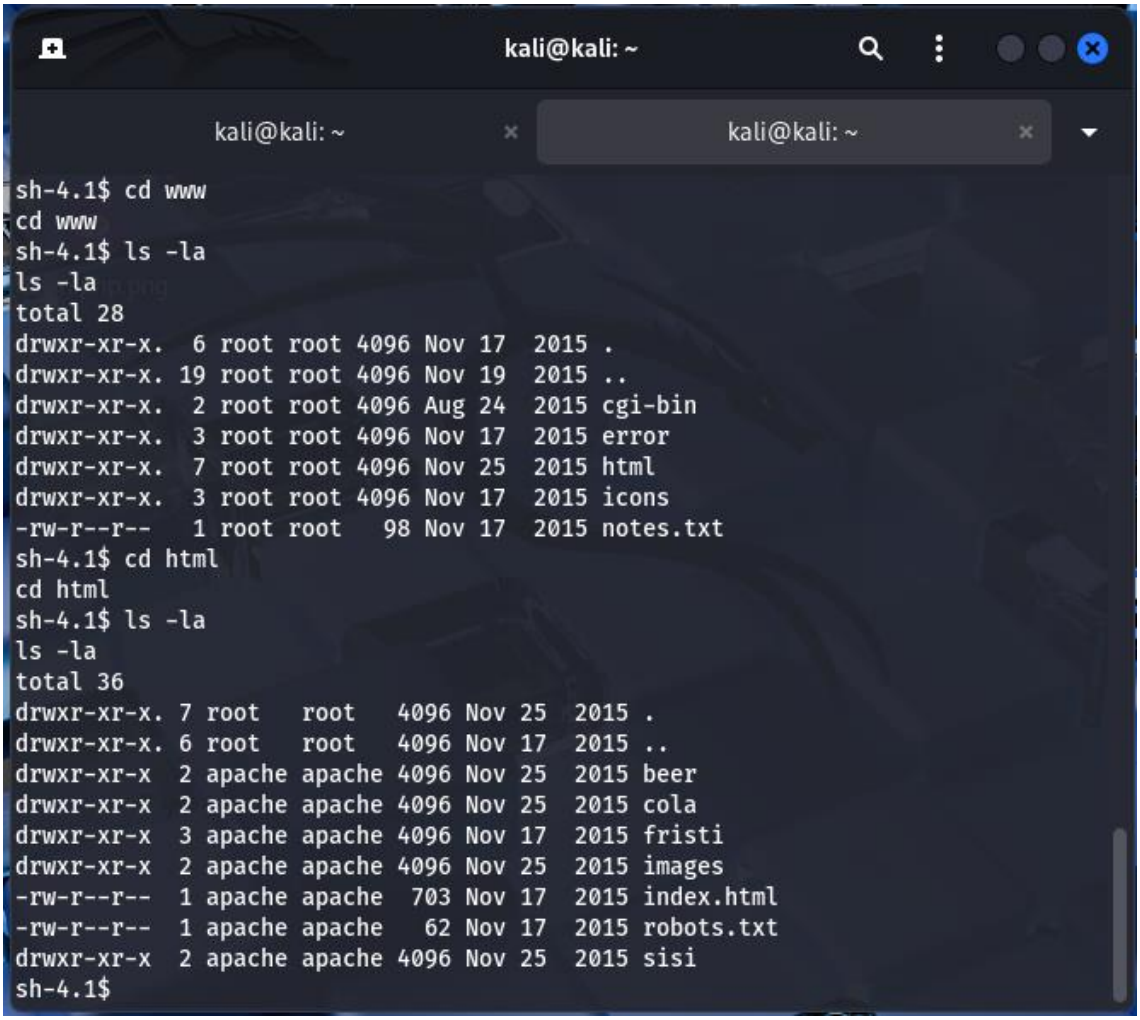

Ilustración 31. Exploración de los Archivos del Sitio Web en /var/www/html

Esta imagen muestra la exploración del directorio /var/www/html, donde se aloja el sitio web de FristiLeaks. Se listan los siguientes elementos clave:

- Carpetas restringidas**:** beer, cola, sisi (mencionadas previamente en el archivo robots.txt).
- Carpeta principal**:** fristi (contiene probablemente los archivos centrales del sitio).
- Archivos clave**:** index.html (página principal) y robots.txt.

5.2.4 Identificación de scripts PHP

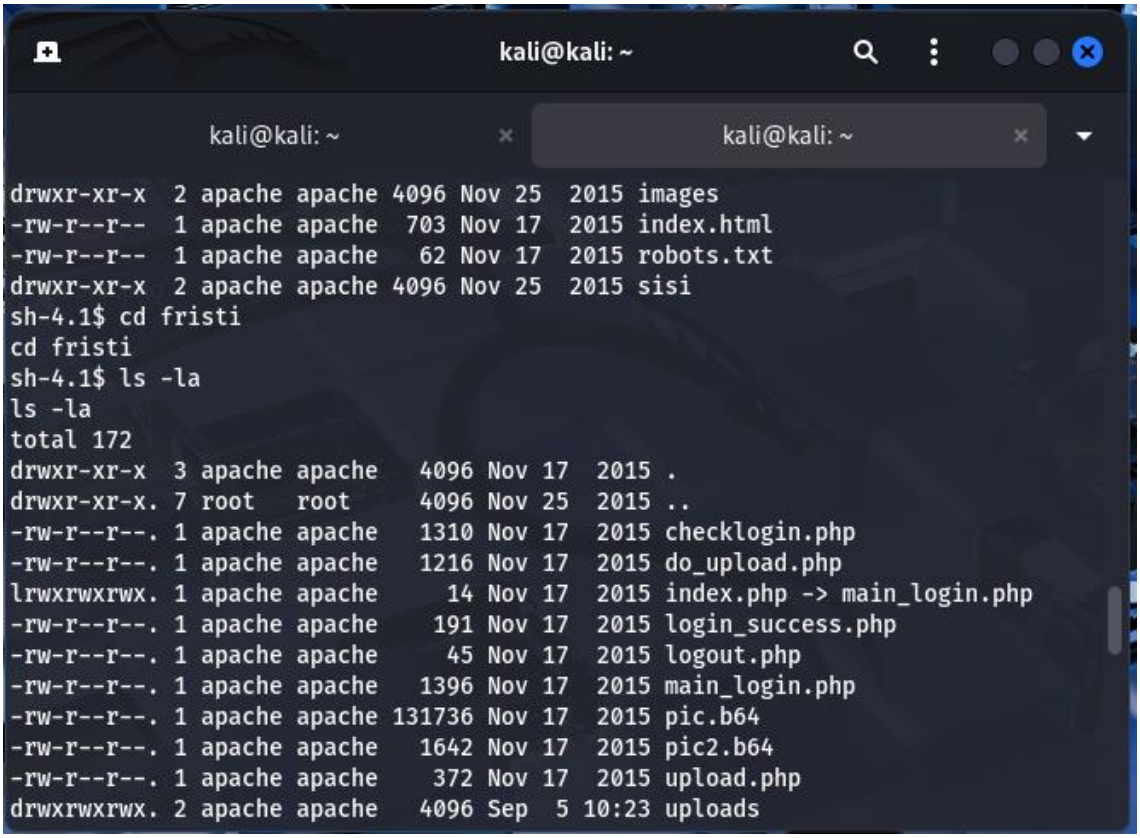

Ilustración 32. Descubrimiento de Archivos en el Directorio /fristi

Esta imagen revela los archivos internos del sitio de FristiLeaks en /var/www/html/fristi:

- checklogin.php, main_login.php, login_success.php (manejan autenticación).

5.2.5 Muestra de las credenciales de la base de datos MySQL

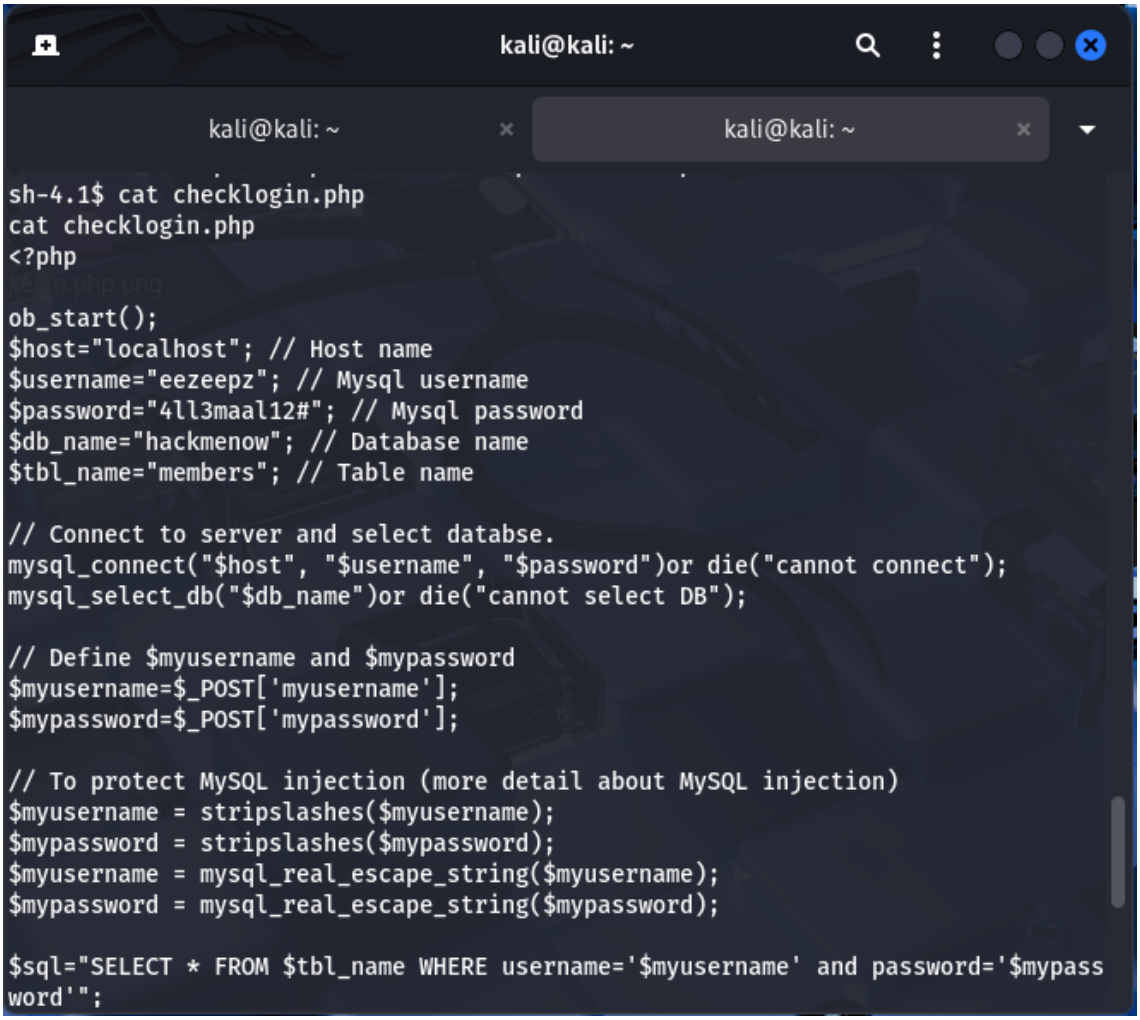

Ilustración 33. Credenciales de la Base de Descubiertas en checklogin.php

Esta imagen muestra el contenido del archivo checklogin.php, que contiene las credenciales de acceso a la base de datos del sitio:

- **Host:** localhost
- **Usuario de MySQL:** eezeepz
- **Contraseña de MySQL:** 4113maa112#
- **Nombre de la base de datos:** hackmenow
- **Tabla de usuarios:** members

5.2.6 Acceso a la base de datos usando las credenciales encontradas

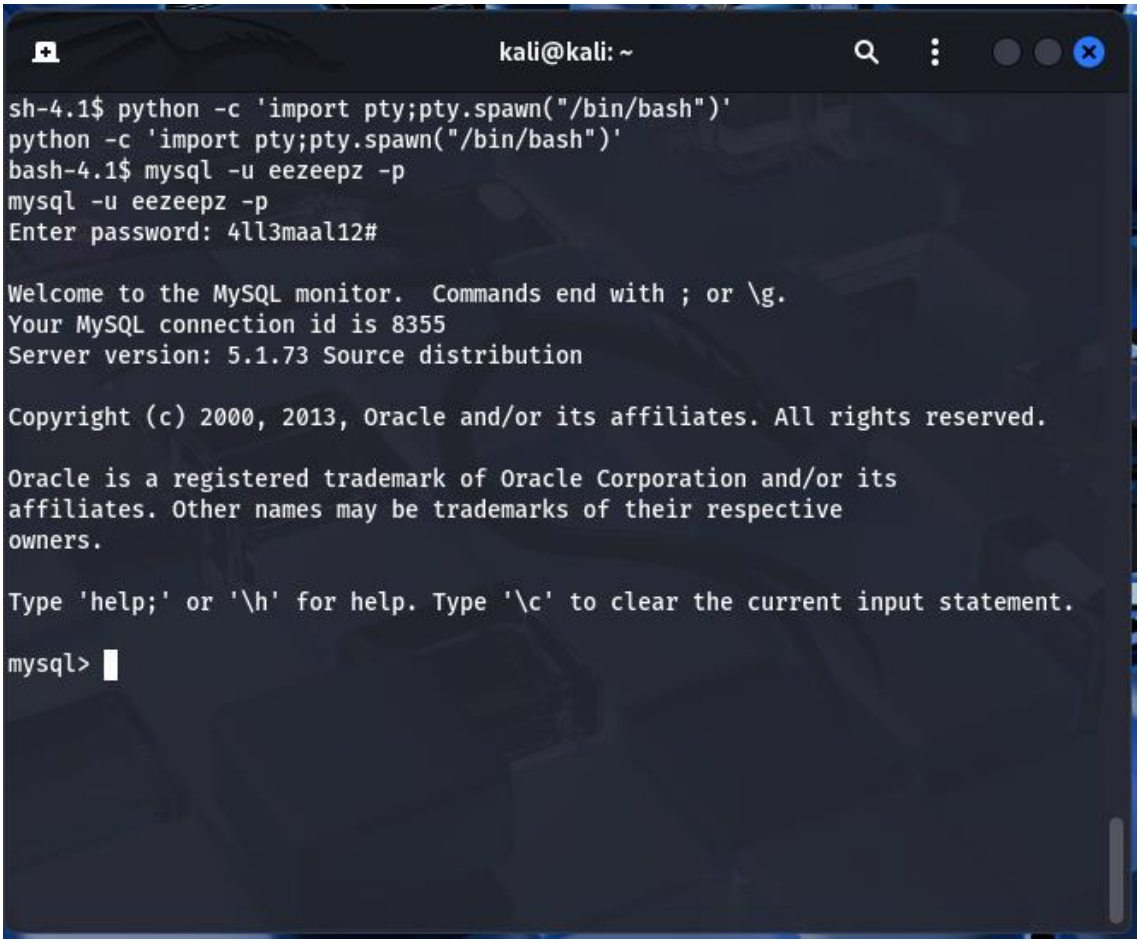

Ilustración 34. Conexión Exitosa a la Base de Datos MySQL

Esta imagen muestra cómo se estableció una conexión interactiva con la base de datos MySQL del servidor FristiLeaks:

1. Se usó python -c 'import pty;pty.spawn("/bin/bash")' para obtener un bash funcional.
2. Se ejecutó mysql -u eezeepz -p y se ingresó la contraseña 4ll3maa112# (ligera variación de la encontrada en el script, pero esencialmente la misma).

5.2.7 Identificación de las tablas y estructura de la base de datos

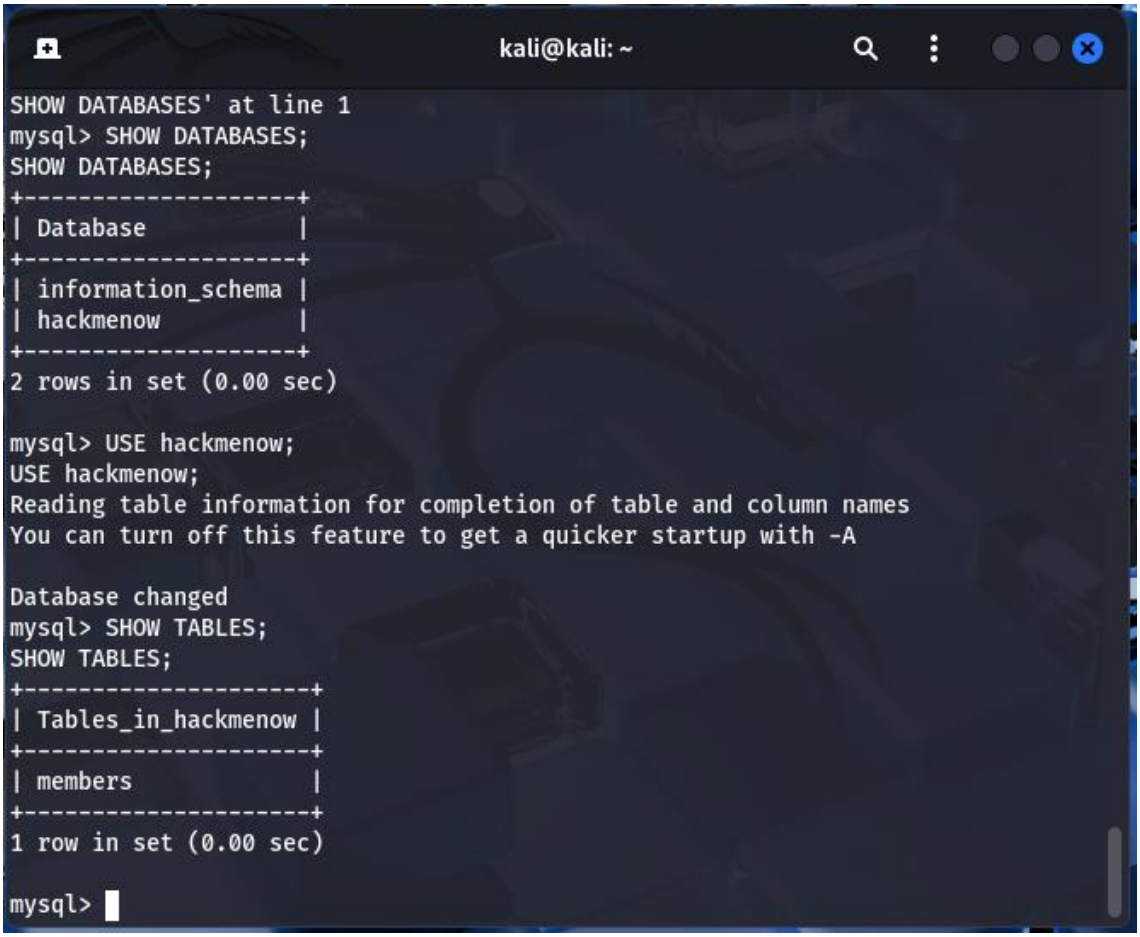

Ilustración 35. Exploración de la Base de Datos "hackmenow"

Esta imagen muestra los comandos SQL utilizados para explorar la base de datos:

1. SHOW DATABASES; revela que existen dos bases: information_schema (estándar de MySQL) y hackmenow (la base de datos del sitio).
2. USE hackmenow; para trabajar directamente con la base de datos objetivo.
3. SHOW TABLES; confirma que solo existe una tabla llamada members, que muy probablemente almacena los usuarios y contraseñas del sistema.

5.2.8 Credenciales almacenadas en la base de datos

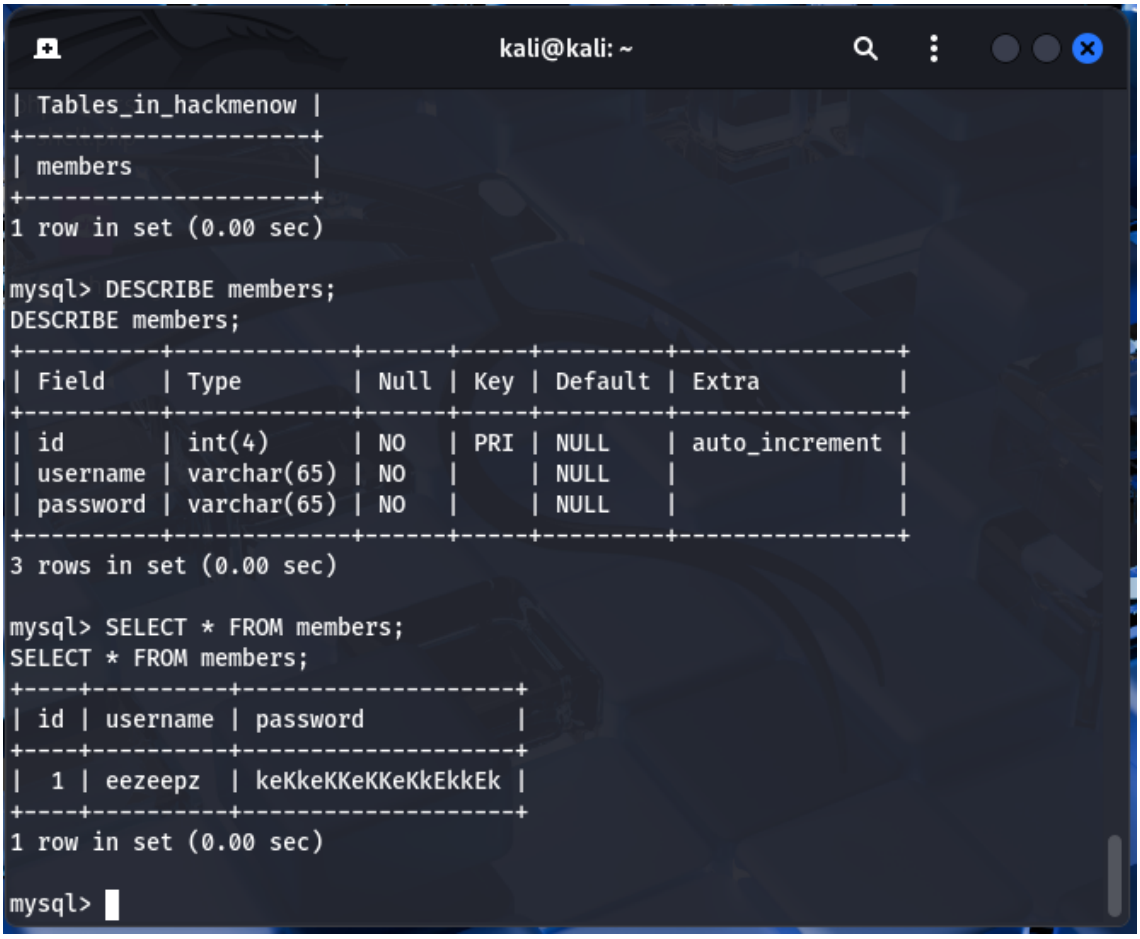

Ilustración 36. Contenido de la Tabla "members" con Credenciales

Esta imagen muestra el resultado de la consulta SELECT * FROM members;, que muestra el contenido completo de la tabla "members" en la base de datos "hackmenow". Se observa:

- **Único usuario registrado:**
  - **Username:** eezeepz
  - **Password:** Una contraseña larga compuesta por repeticiones de "kek" (ej: kekkekkekkek...).

5.2.9 Campos y tipos de datos de la tabla members

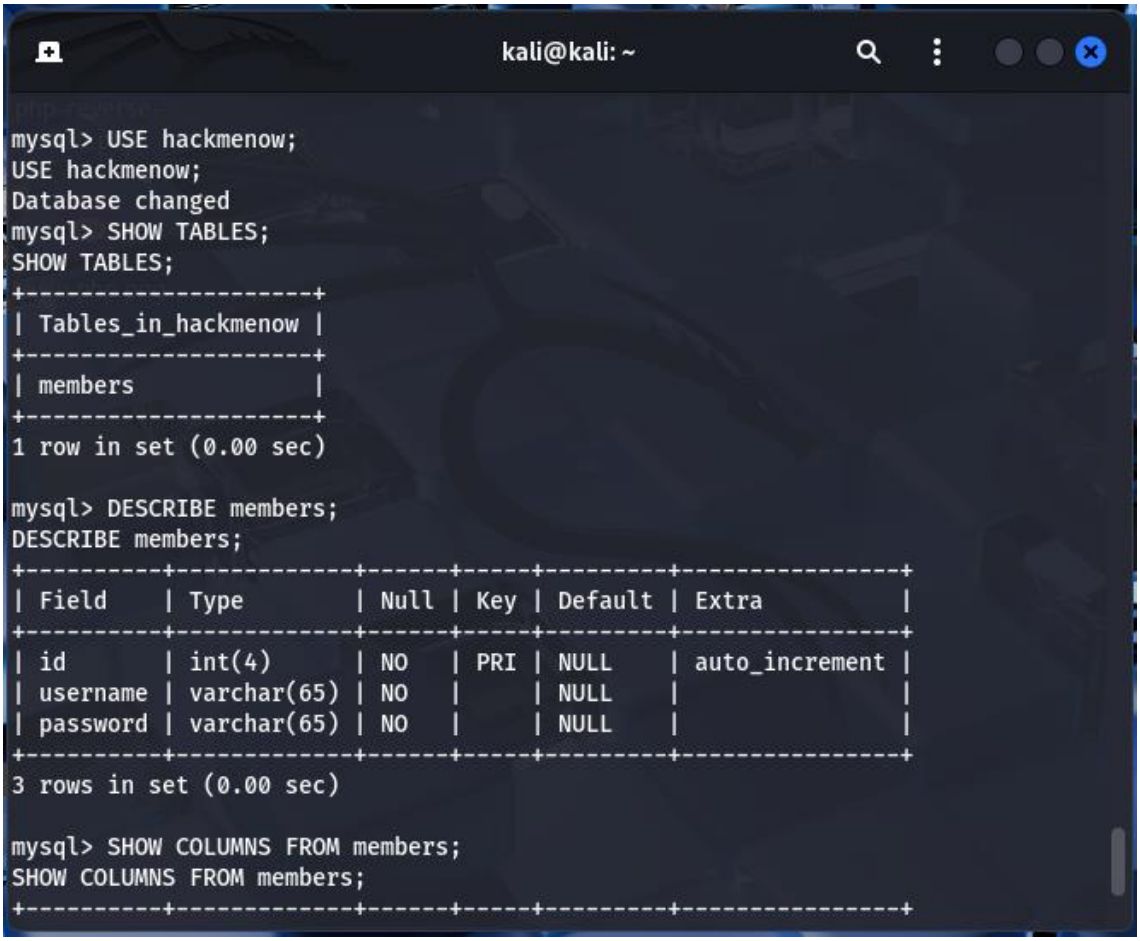

Ilustración 37. Estructura de la Tabla "members" en la Base de Datos

Esta imagen muestra los comandos utilizados para examinar la estructura de la tabla "members":

1. USE hackmenow; (para asegurar que esta en la base correcta).
2. SHOW TABLES; confirma que solo existe la tabla "members".
3. DESCRIBE members; revela la estructura de la tabla:
    - **id:** Entero (int) de 4 dígitos, clave primaria y auto-incrementable.
    - **username:** Campo de texto (varchar) de hasta 65 caracteres.
    - **password:** Campo de texto (varchar) de hasta 65 caracteres.

5.2.10 Creación de usuarios para acceso al sistema

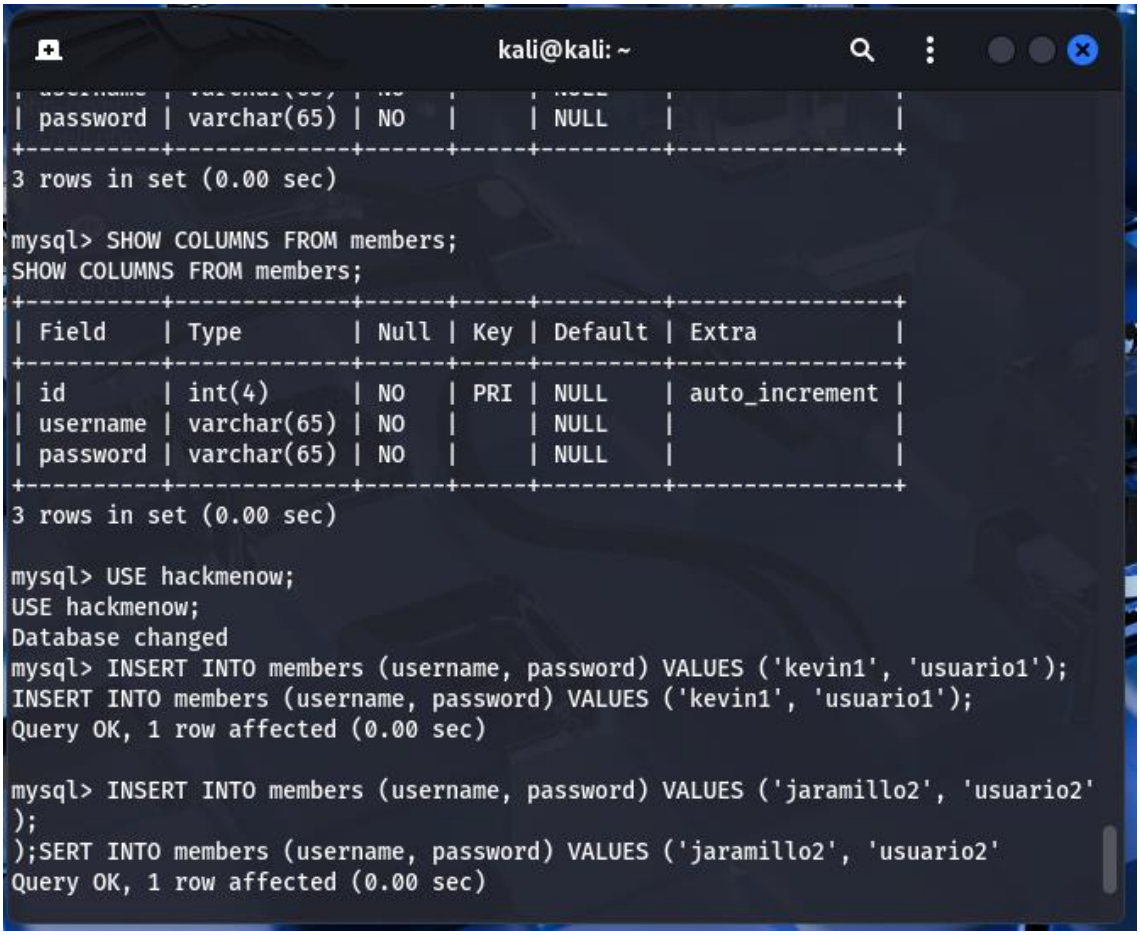

Ilustración 38. Inserción de Nuevos Usuarios en la Base de Datos

Esta imagen muestra el uso del comando INSERT INTO para agregar dos nuevos usuarios a la tabla "members" de la base de datos:

1. **Usuario 1:**
    - **Username:** kevin1
    - **Password:** usuario1
2. **Usuario 2:**
    - **Username:** jaramillo2
    - **Password:** usuario2

5.2.11 Confirmación de la inserción exitosa de los usuarios

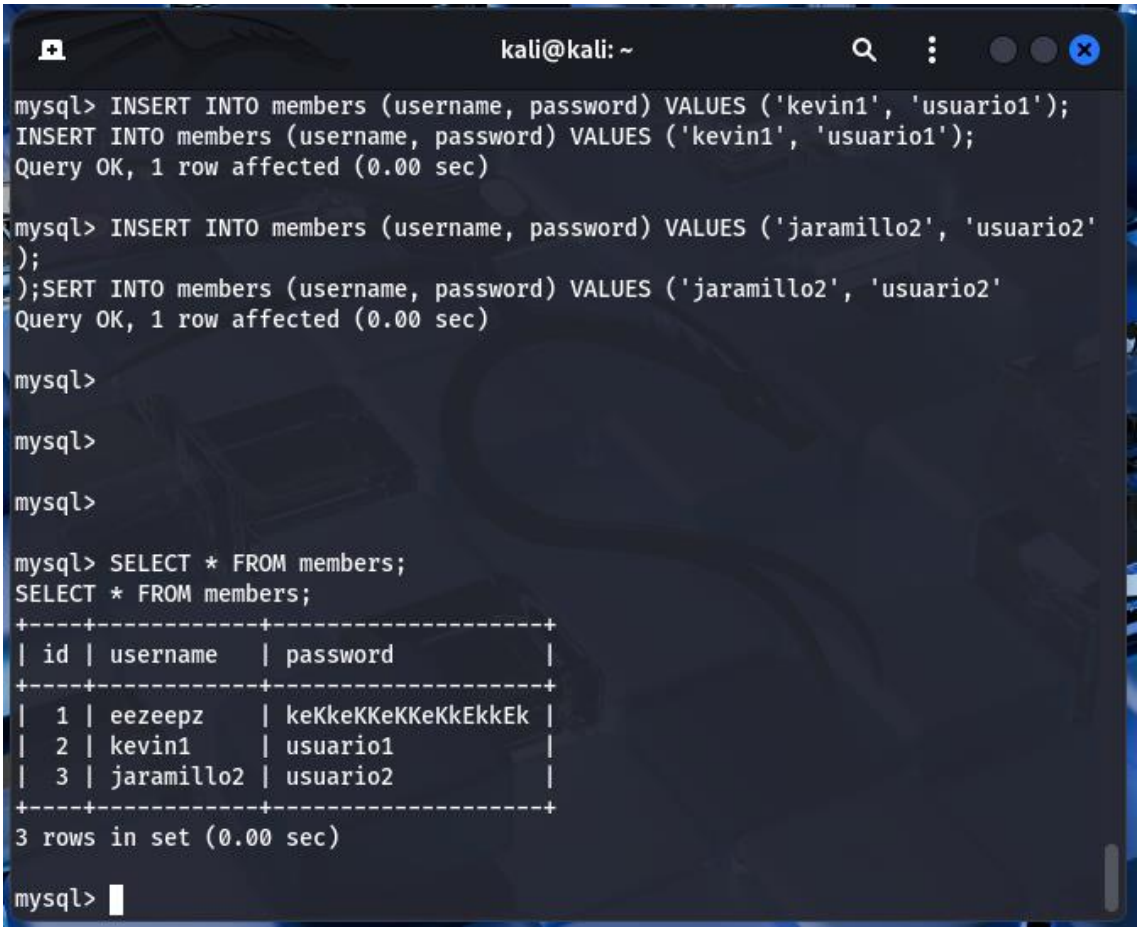

Ilustración 39. Verificación de los Nuevos Usuarios en la Base de Datos

Esta imagen muestra el resultado de la consulta SELECT * FROM members; después de insertar los nuevos usuarios. Se confirma que la tabla ahora contiene tres registros:

1. **Usuario original:**
    - **Username:** eezepz (posible error tipográfico de "eezeepz")
    - **Password:** Una contraseña larga compuesta por repeticiones de "kek"
2. **Nuevo usuario 1:**
    - **Username:** kevin1
    - **Password:** usuario1
3. **Nuevo usuario 2:**

- **Username:** jarami1lo2 (nombre con posible error tipográfico)
- **Password:** usuario2

5.2.12 Acceso utilizando las credenciales creadas en la base de datos

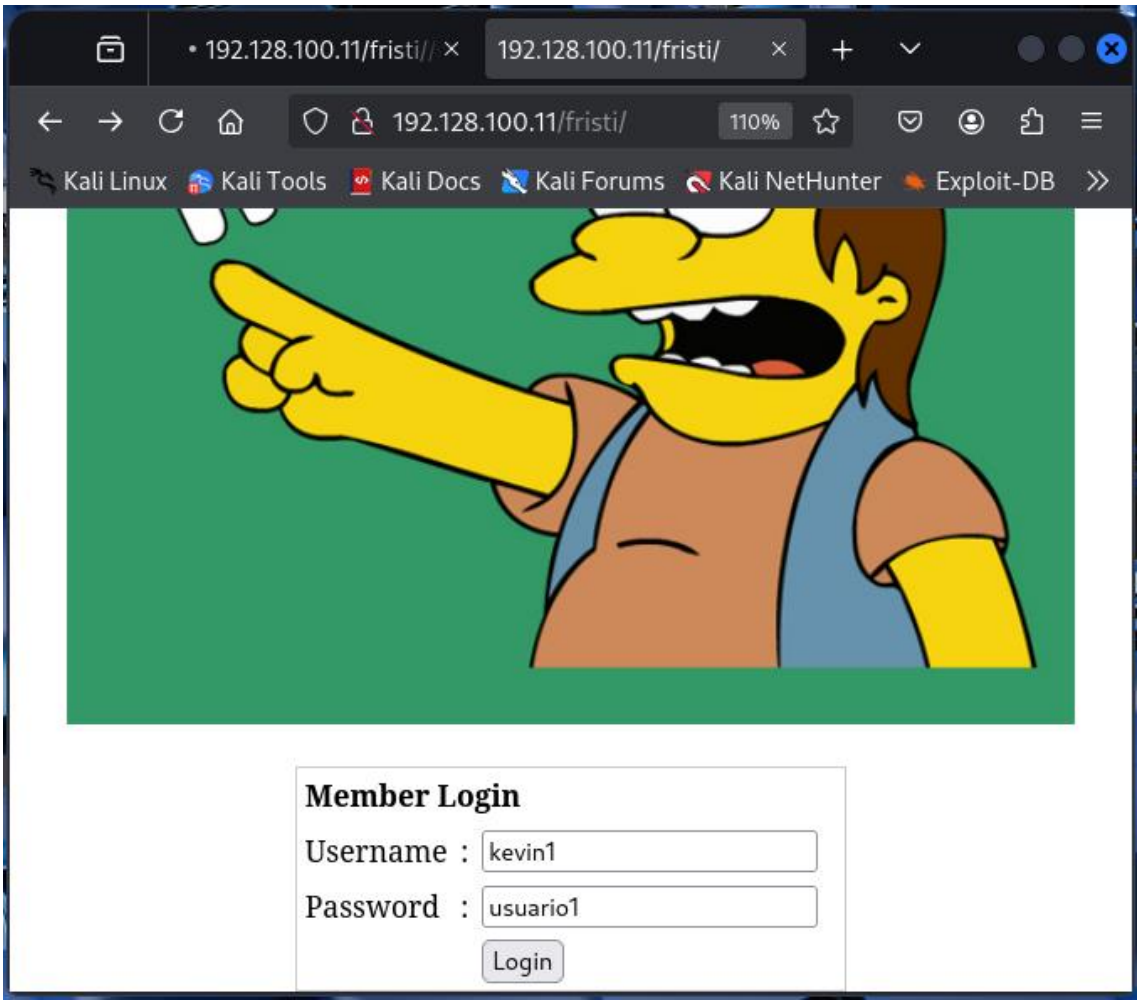

Ilustración 40. Inicio de Sesión con el Nuevo Usuario "kevin1"

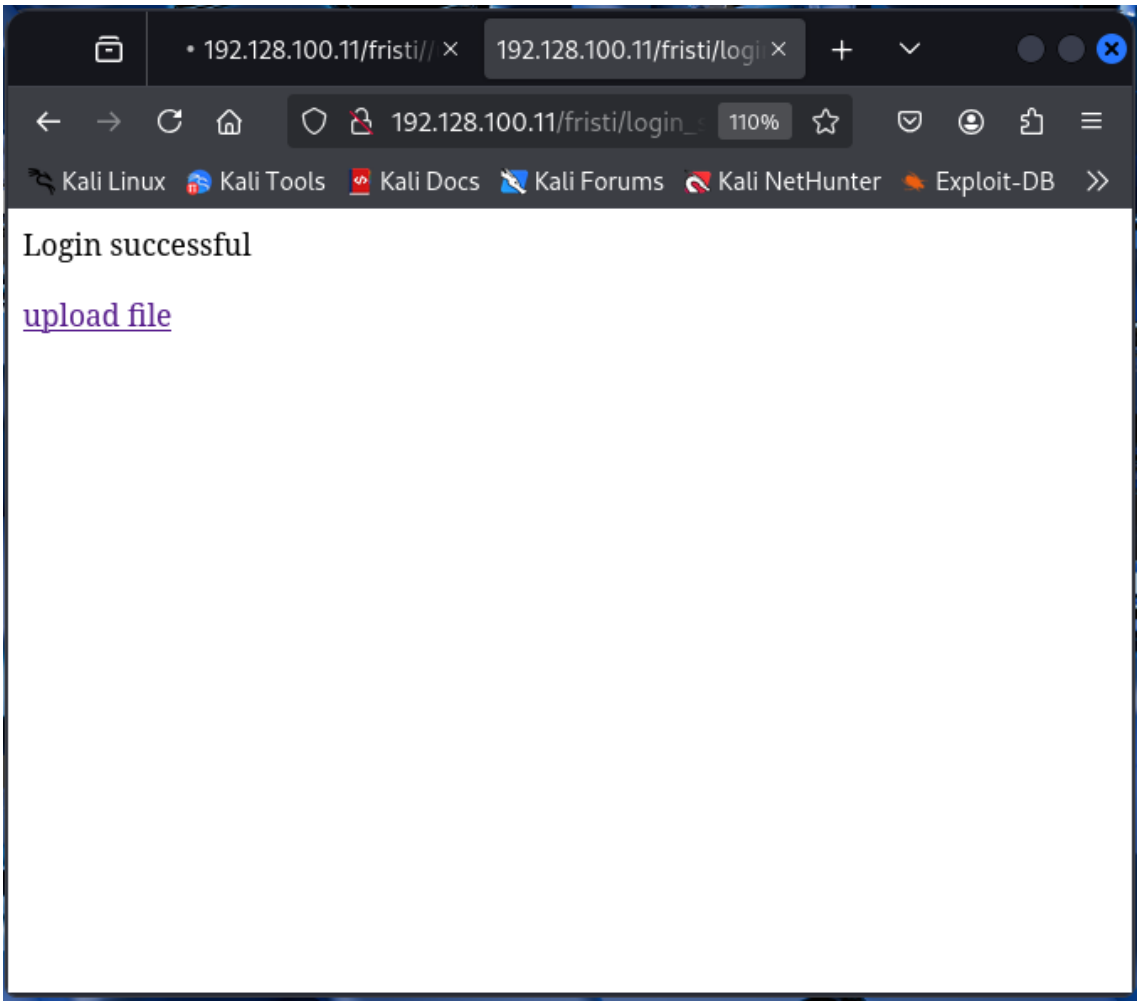

Ilustración 41. Inicio de Sesión Exitoso con el Usuario "kevin1"

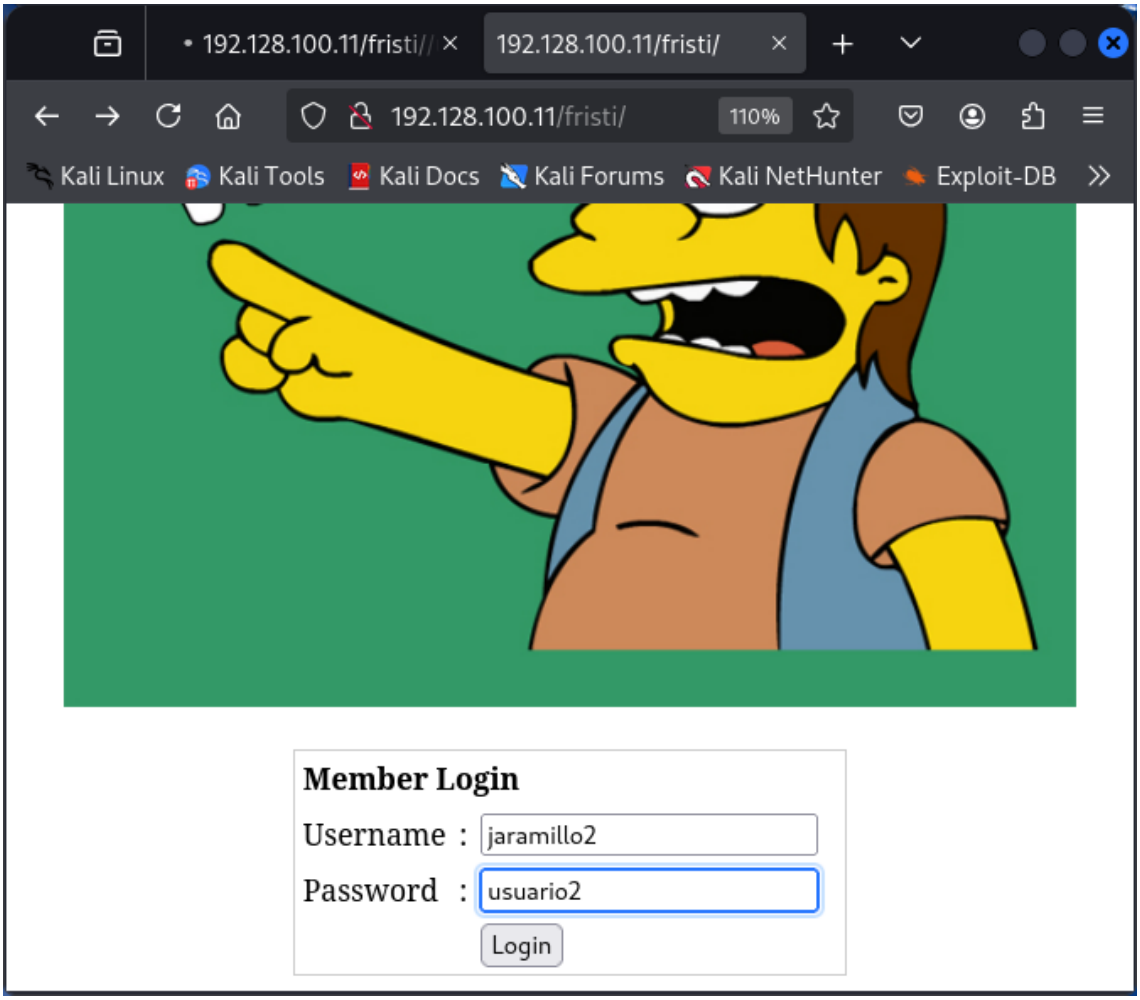

Ilustración 42. Inicio de Sesión con el Nuevo Usuario "jaramillo1"

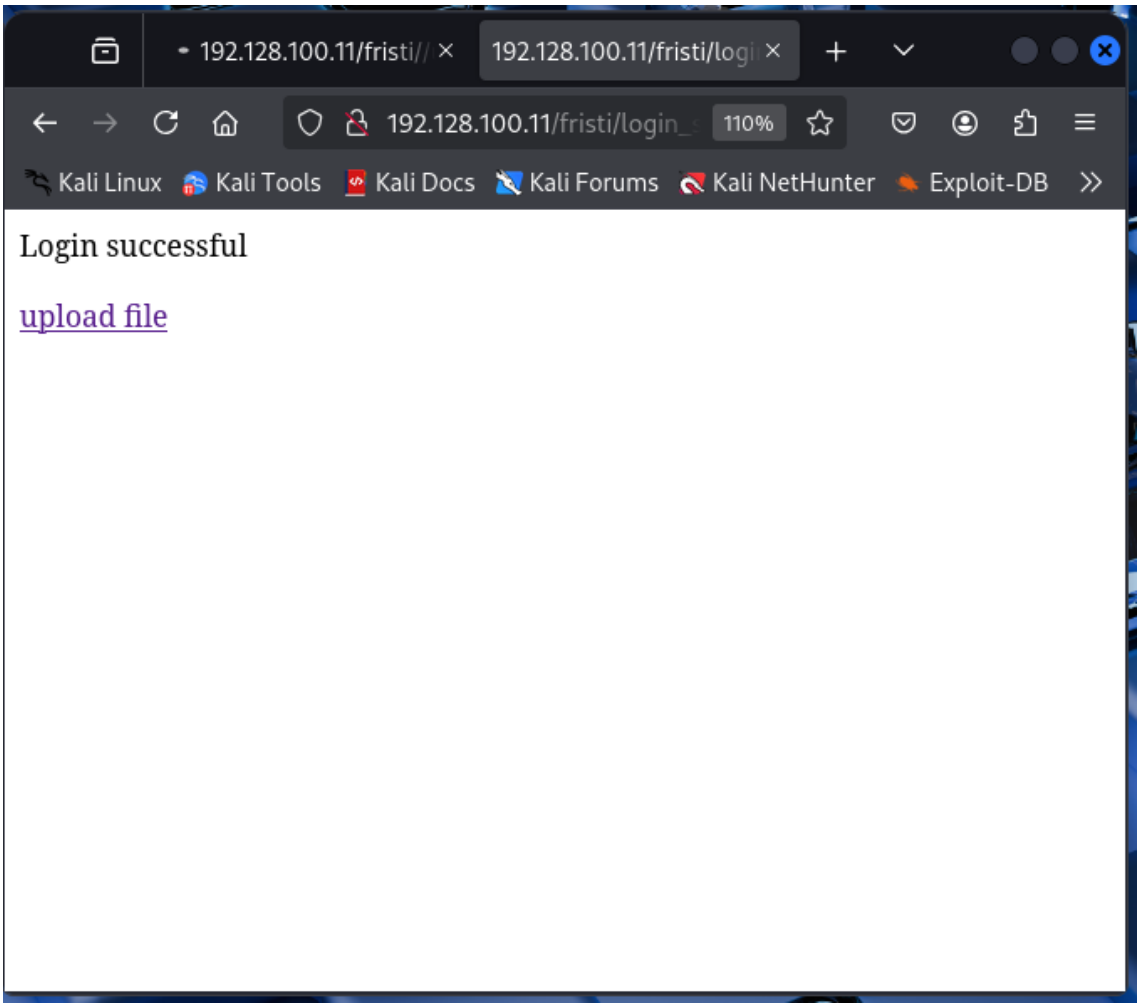

Ilustración 43. Inicio de Sesión Exitoso con el Usuario "jaramillo1"

5.3.Cambio de mac en Lubuntu

5.3.1 Dirección MAC anterior

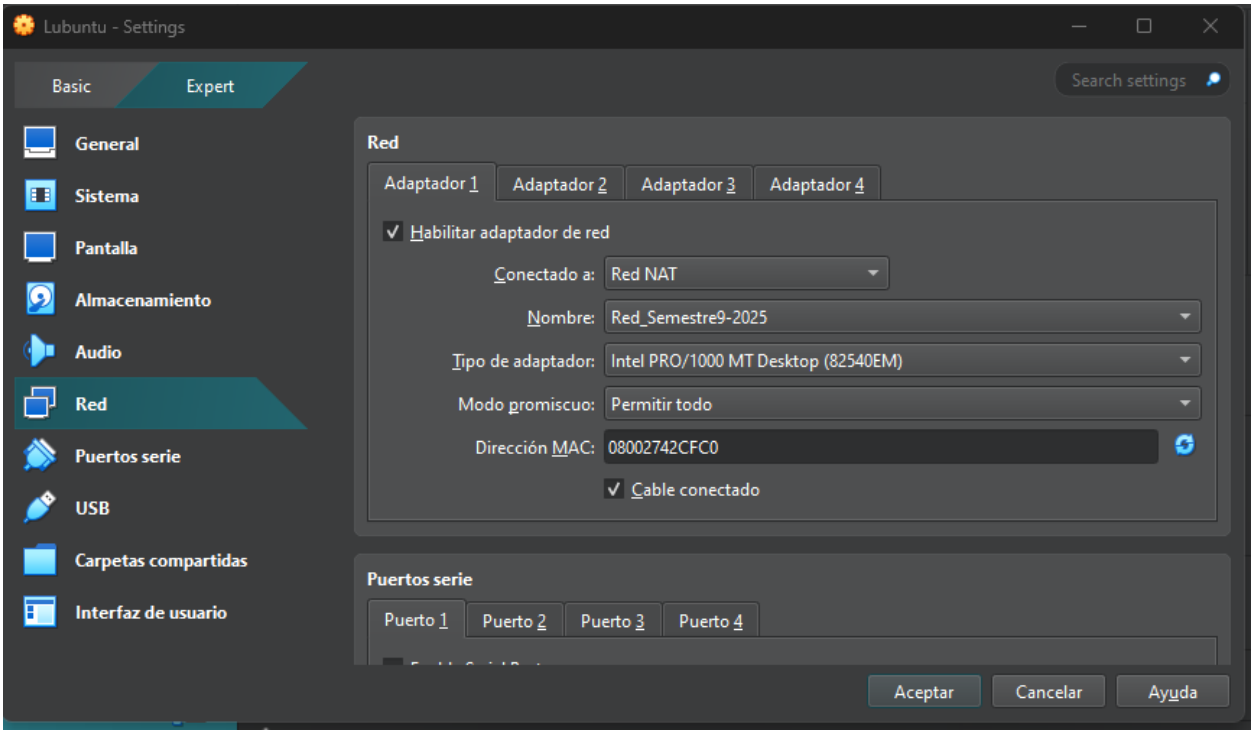

Ilustración 44. Configuración Original de Red de la Máquina Lubuntu

Esta imagen muestra la configuración de red original de la máquina Lubuntu en, donde la dirección MAC asignada era 08002742CFC0.

5.3.2 Nueva dirección MAC asignada

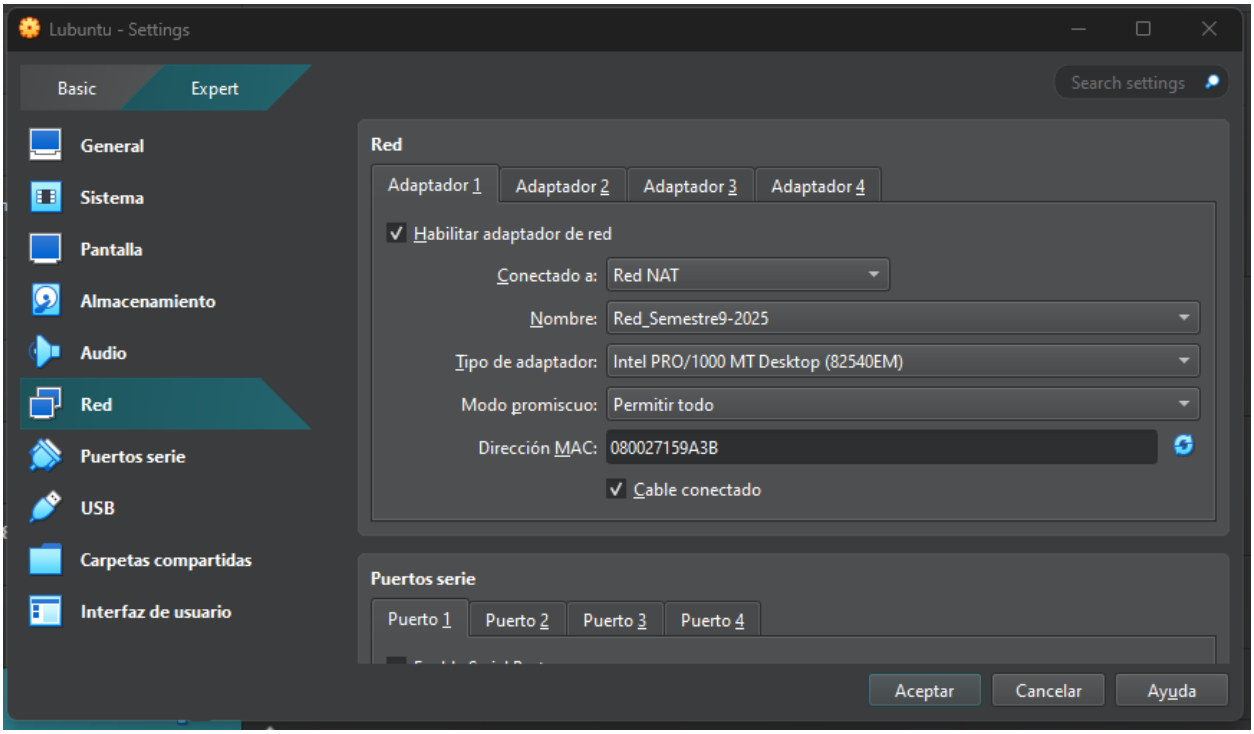

Ilustración 45. Configuración de Red Modificada

Esta imagen muestra la configuración actualizada de la máquina Lubuntu, donde la dirección MAC ha sido cambiada a 080027159A3B.

5.3.3 Comprobación de la comunicación con Kali

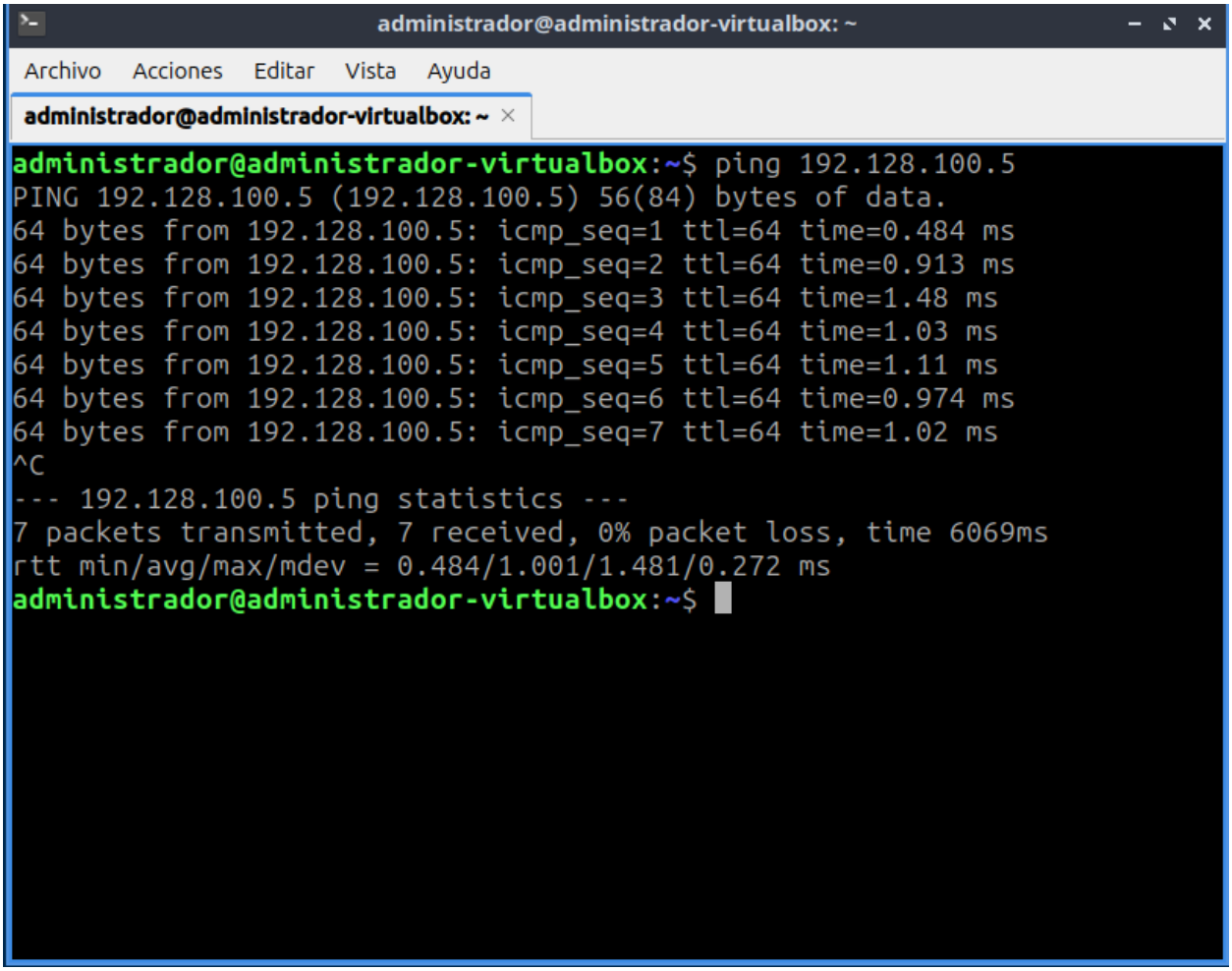

Ilustración 46. Prueba de Conectividad con Kali Linux desde Lubuntu

Esta imagen muestra el resultado del ping 192.128.100.5 ejecutado en la máquina Lubuntu (con la dirección MAC modificada). A pesar del cambio de MAC, Lubuntu aún puede comunicarse con Kali Linux (IP: 192.128.100.5)

5.3.4 Actualización de paquetes

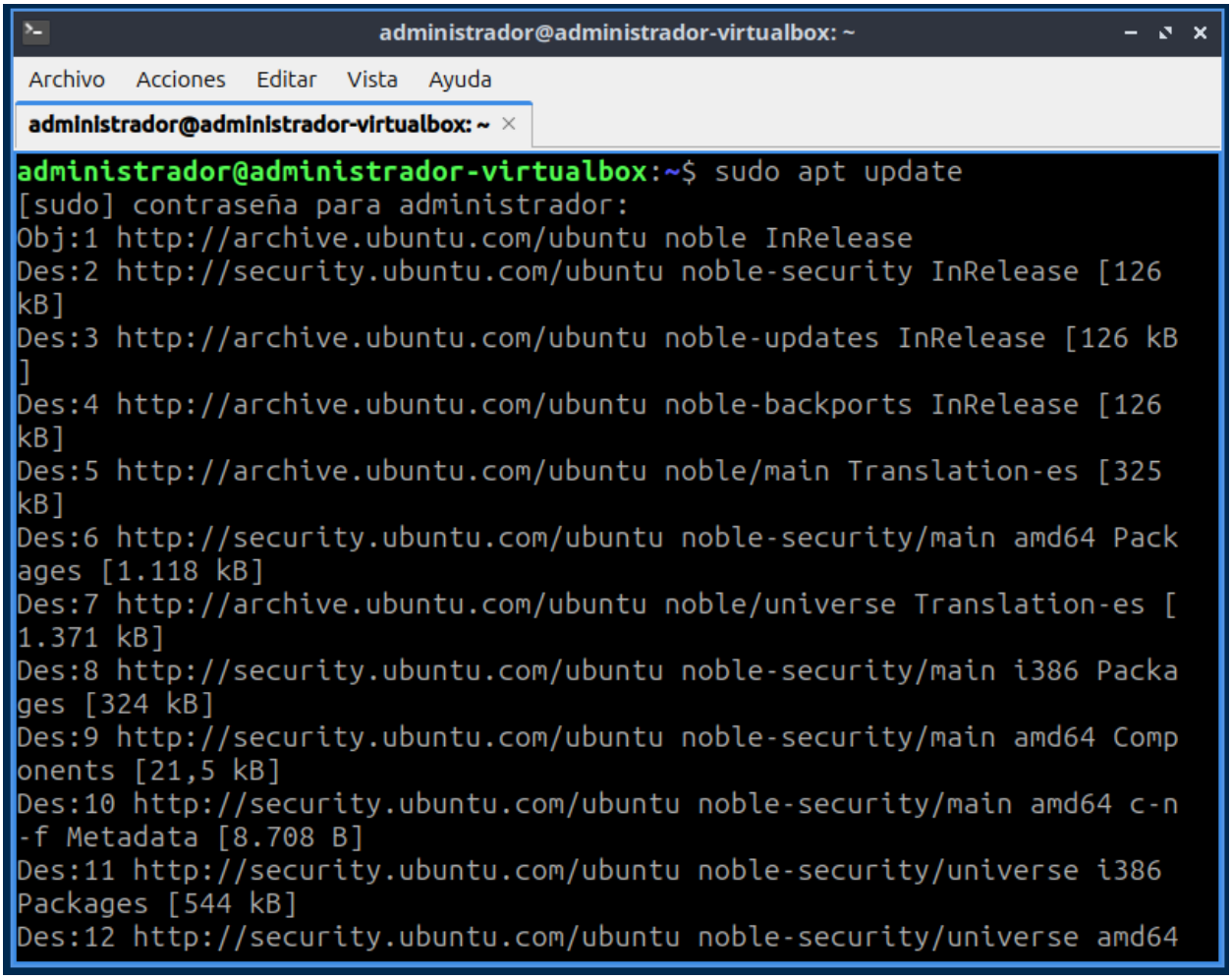

Ilustración 47. Actualización del Sistema Lubuntu

Esta imagen muestra la ejecución del comando sudo apt update en la máquina Lubuntu para la actualización de paquetes y repositorios

5.3.5 Bloqueo de tráfico entrante y saliente

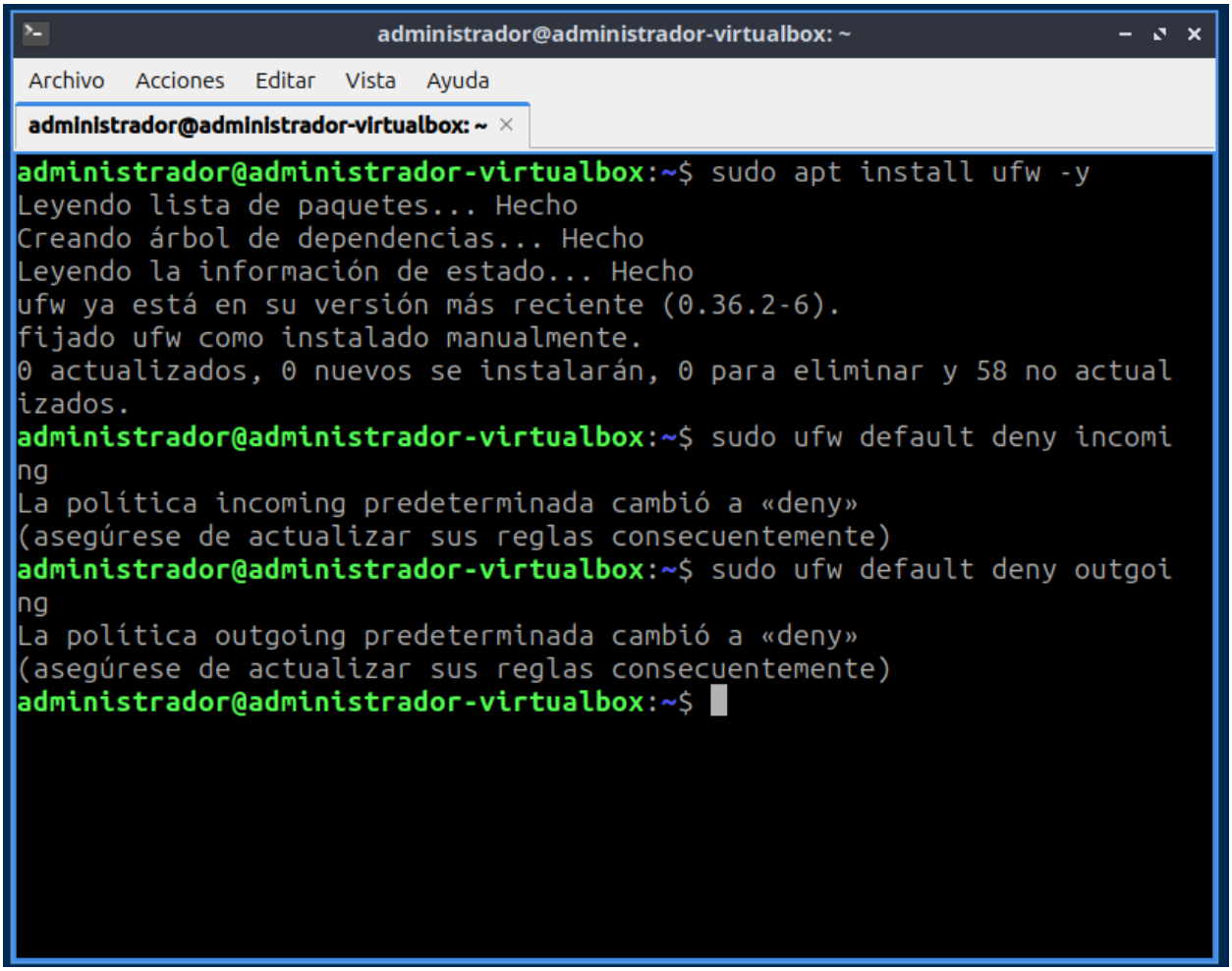

Ilustración 48. Configuración del Firewall UFW para Aislar Lubuntu

Esta imagen muestra la configuración del firewall en Lubuntu para lograr el aislamiento:

1. sudo ufw default deny incoming → Bloquea todo el tráfico entrante.
2. sudo ufw default deny outgoing → Bloquea todo el tráfico saliente.

Estas políticas restringen completamente la comunicación de la máquina Lubuntu, impidiendo tanto conexiones entrantes como salientes.

5.3.6 Edición de reglas para denegar ping

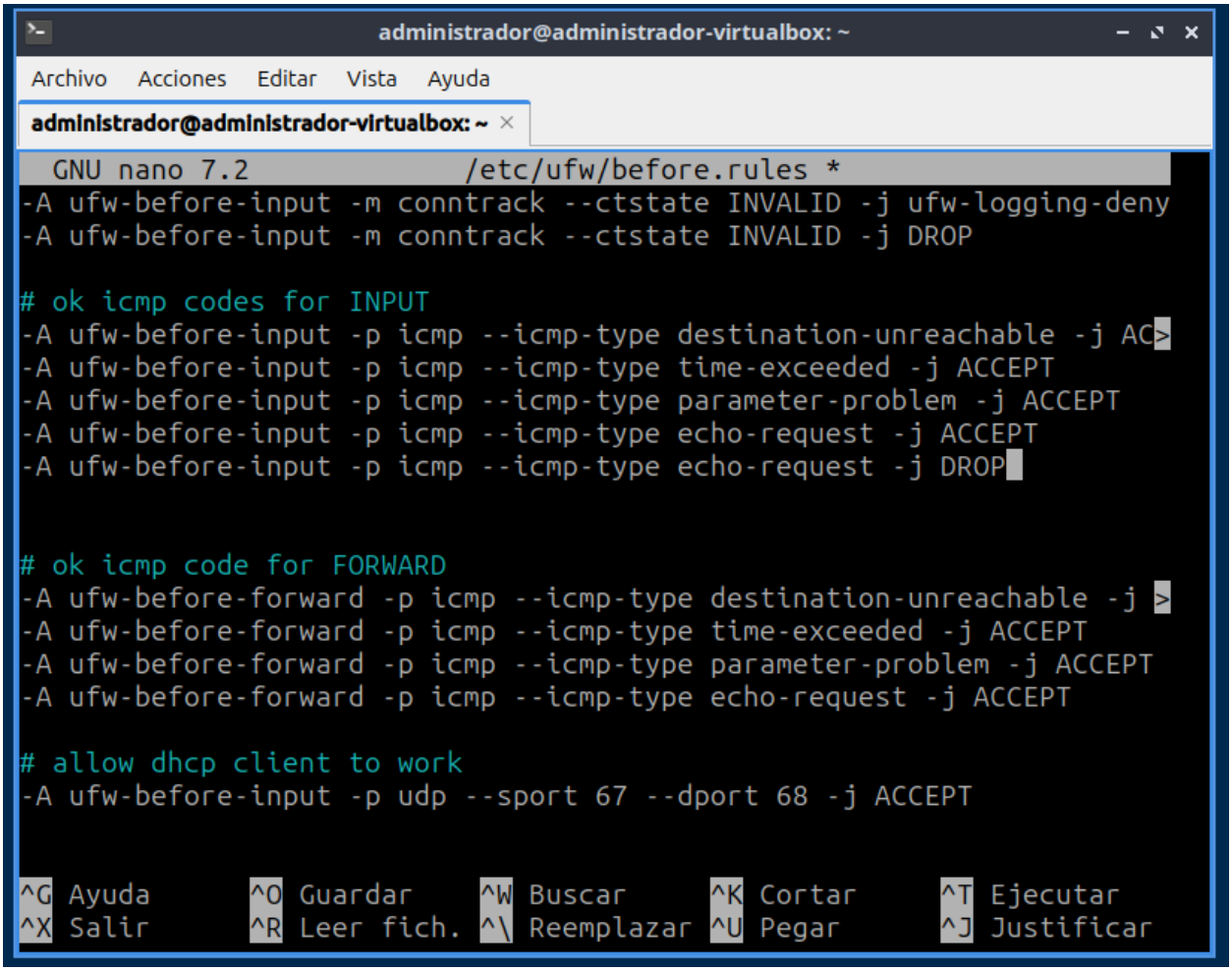

Ilustración 49. Modificación de Reglas del Firewall

Esta imagen muestra la edición del archivo de configuración /etc/ufw/before.rules utilizando el editor Nano. Se ha añadido una regla personalizada para bloquear específicamente las solicitudes ICMP (ping):

- -A ufw-before-input -p icmp --icmp-type echo-request -j DROP

  Esta regla asegura que las solicitudes de ping sean denegadas.

5.3.7 Habilitación y confirmación del firewall

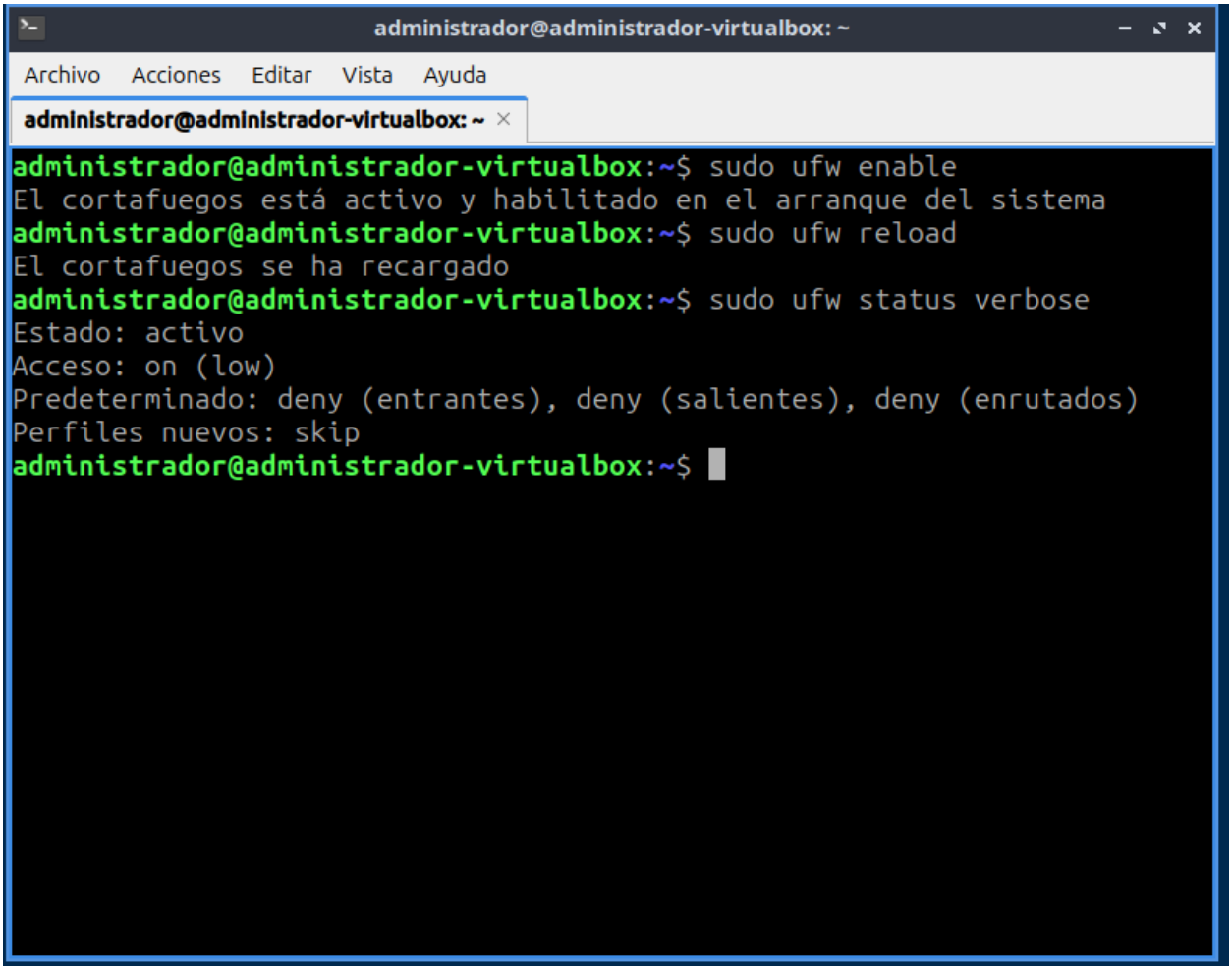

Ilustración 50. Activación y Verificación del Firewall

Esta imagen muestra la activación y verificación del firewall UFW en Lubuntu:

1. sudo ufw enable → Habilita UFW y lo activa en el arranque del sistema.
2. sudo ufw reload → Aplica los cambios realizados en las reglas (incluyendo la denegación de ICMP).
3. sudo ufw status verbose → Confirma que el firewall está **activo** con políticas predeterminadas.

5.3.8 Comprobación del bloqueo de comunicaciones mediante ping

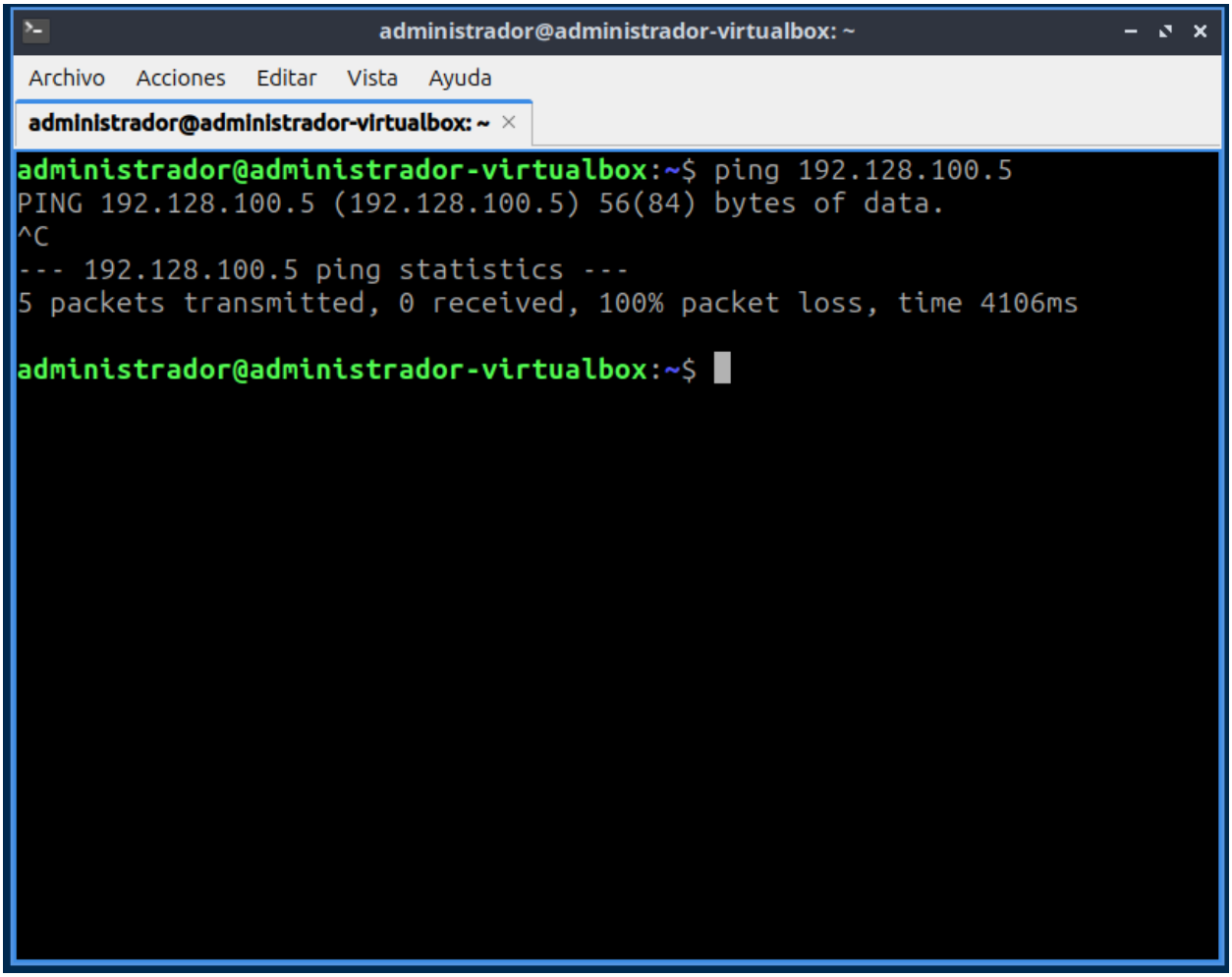

Ilustración 51. Prueba de Aislamiento con Firewall

Esta imagen muestra el resultado del ping 192.128.100.5 (dirección IP de Kali Linux) después de activar el firewall en Lubuntu.

6. Problemas encontrados

Durante la elaboración del laboratorio no tuve inconvenientes, ya que todos los pasos los ejecuté de manera ordenada. La clave fue seguir la secuencia de comandos y verificar las configuraciones

Algunos comandos iniciales no funcionaron como se esperaba, lo que requirió investigación adicional para encontrar alternativas válidas.

7. Recomendaciones

- Verificar sintaxis de comandos
- Anotar los comandos exitosos para replicarlos en futuros escenarios
- Probar comandos paso a paso
- leer los mensajes de error con atención

8. Conclusión

Concluyo diciendo que con este ejercicio se pudo comprobar que una página web sin la protección adecuada puede ser fácilmente atacada con inyección SQL. Se logró ingresar a la base de datos de *Fristi*, crear nuevos usuarios y acceder con ellos al sistema. La experiencia sirvió para aprender de manera práctica y sencilla cómo se explotan estas fallas y para valorar la importancia de reforzar la seguridad en aplicaciones web.

9. Bibliográfica

Rafael Sandoval Morales (2025). Explotación de Vulnerabilidades SQL Injection en FristiLeaks con Kali Linux