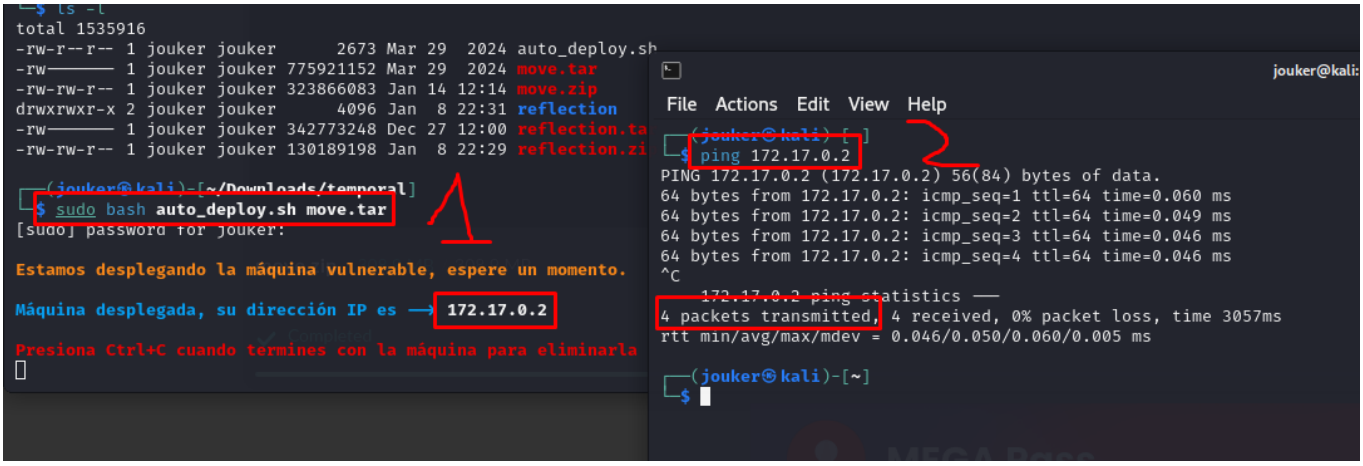
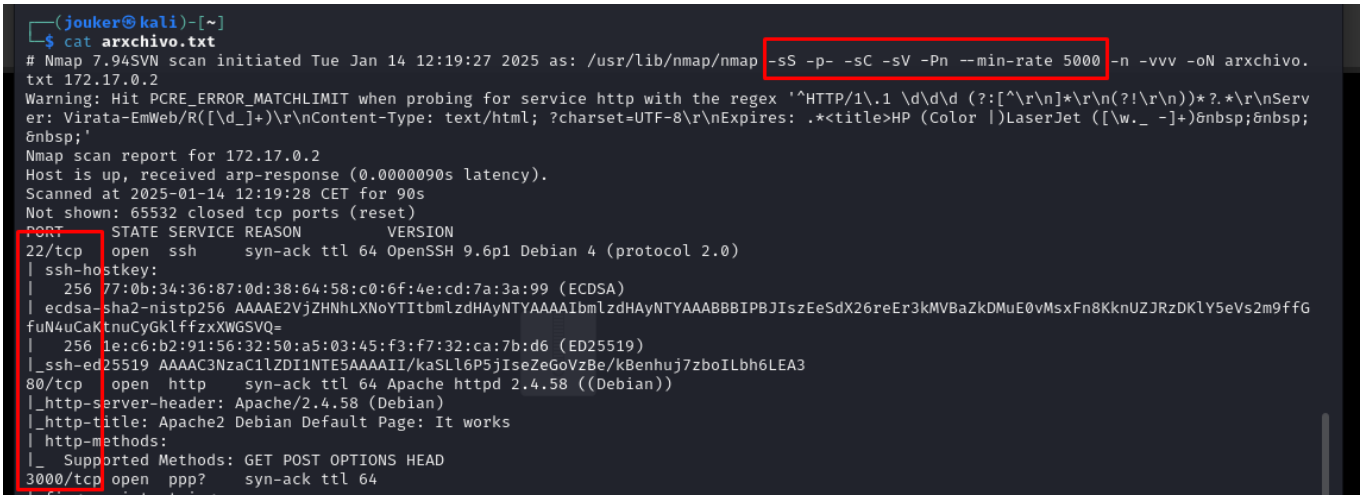
Desplegamiento inicial de máquina y ping de reconocimiento, marcados respectivamente con 1 y 2. Podemos comprobar que si que tenemos conexión a la máquina donde nos queremos conectar.

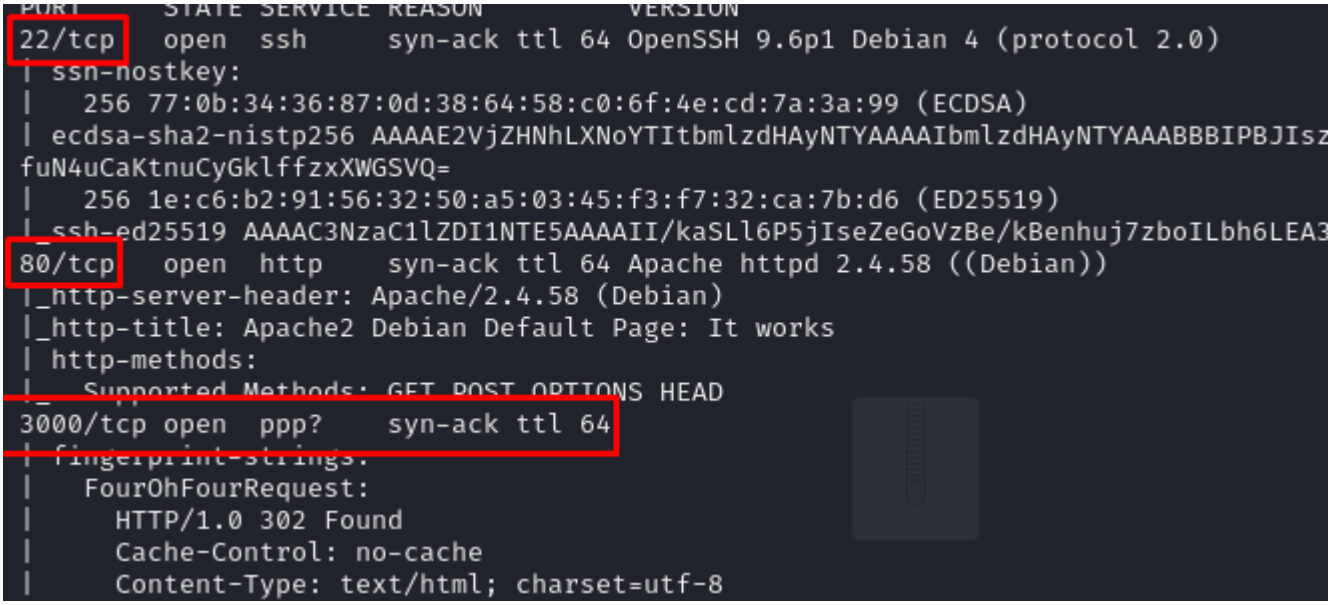


Realizamos el escáner de puertos con NMAP para ver realmente que puertos tiene abiertos esta máquina move.

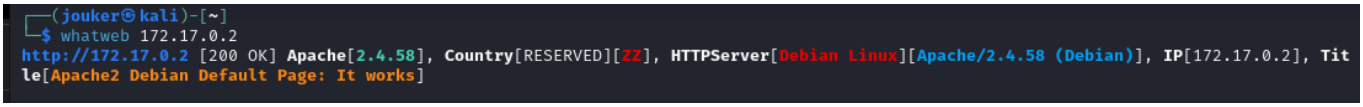


Conseguimos realizar el NMAP i vemos 3 puertos abiertos, los habituales 80 y 22 pero hay uno que es el 3000, que no identifica aún del todo NMAP. De momento vamos a centrarnos en descubrir que

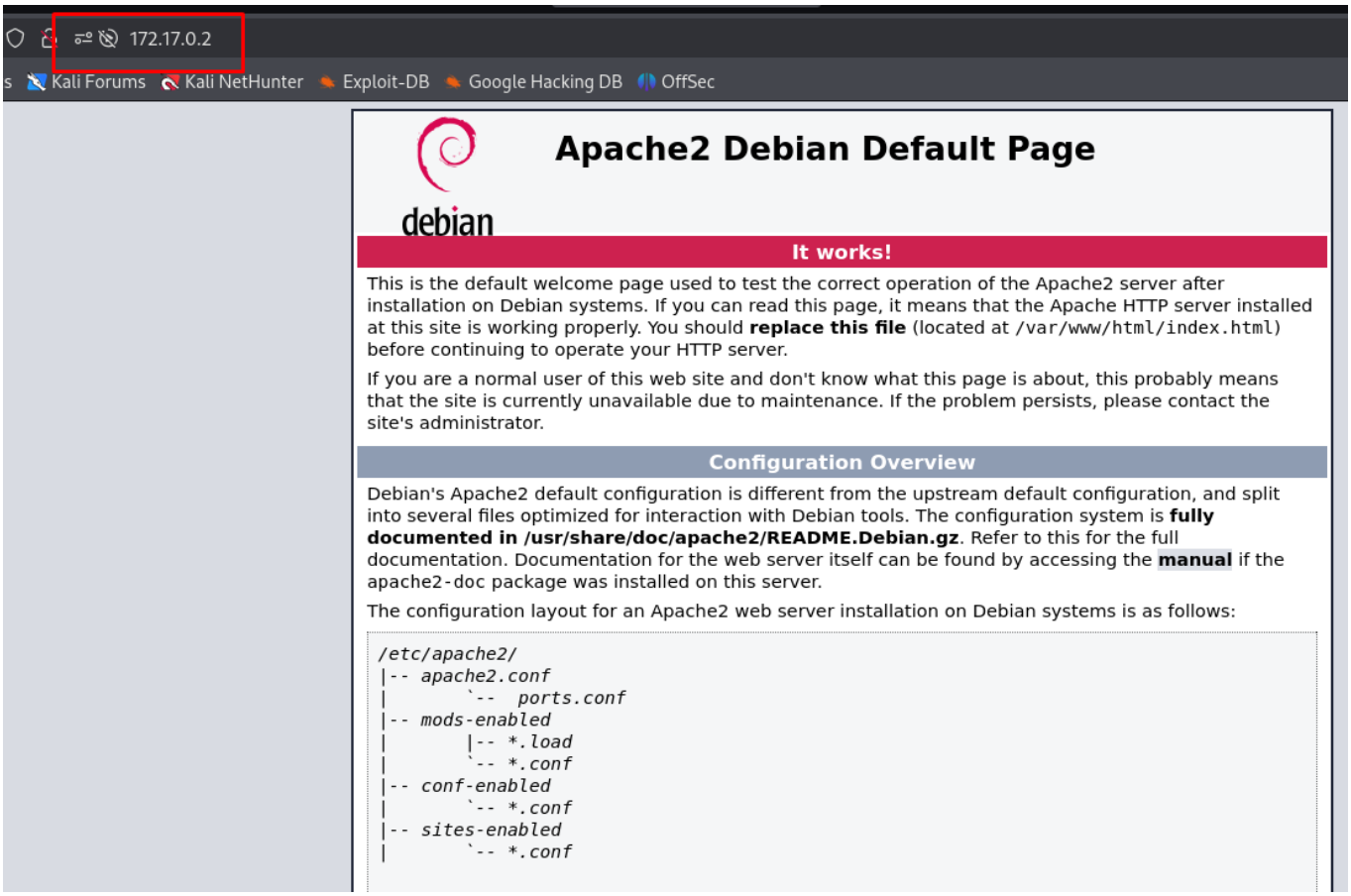
hay en el puerto 80



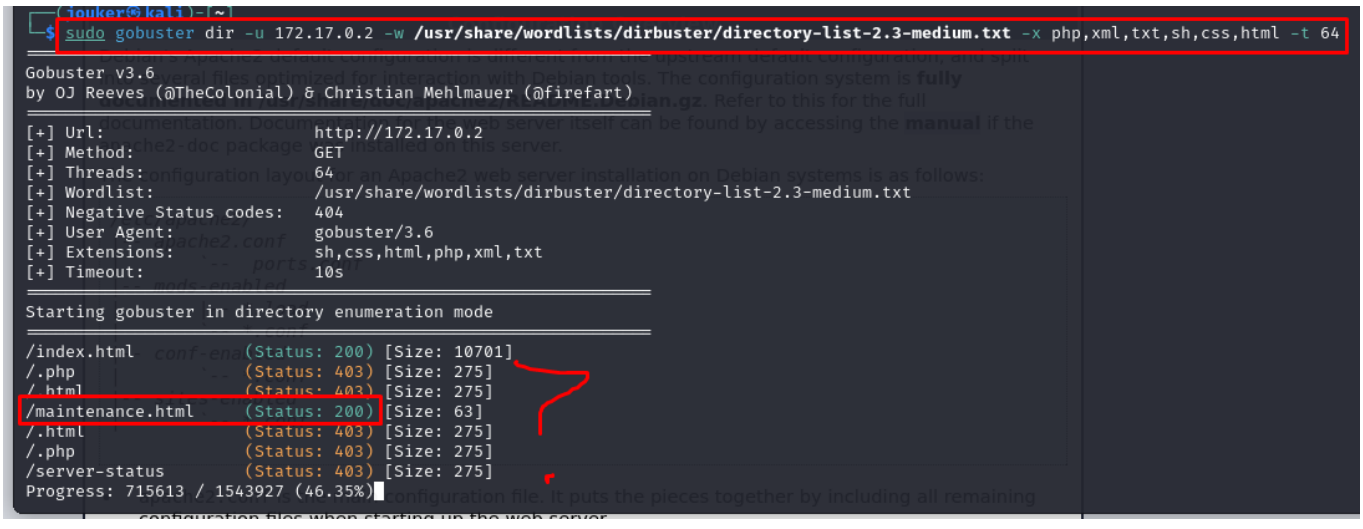
Al intentar observar que tiene el puerto 80 con whatweb, vemos que realmente no hay nada fuera de lugar, no parece que haya nada más aparte de la página por defecto de apache.



Siguiendo con la exploración directamente desde la WEB podemos ver como realmente no hay nada, solo la página por defecto

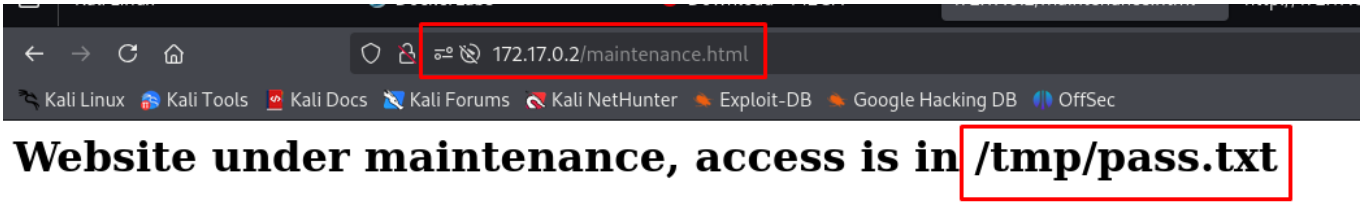


Al aplicar fuzzing web con gobuster podemos listar un par de directorios que al parecer son interesantes, vamos a ver que hay dentro.

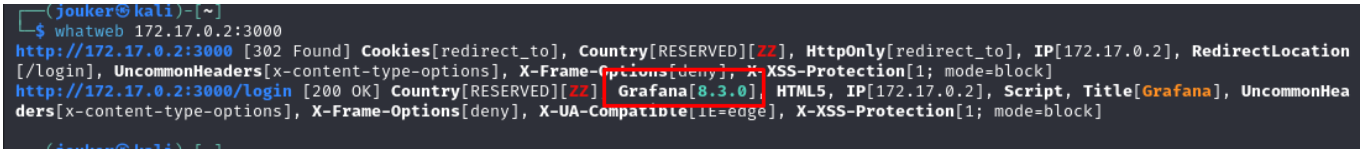


Mensaje de la página web, nos dice una ruta absoluta a algo que parecen credenciales, no he conseguido listar más información en el puerto 80 así que vamos a descubrir si con la información que

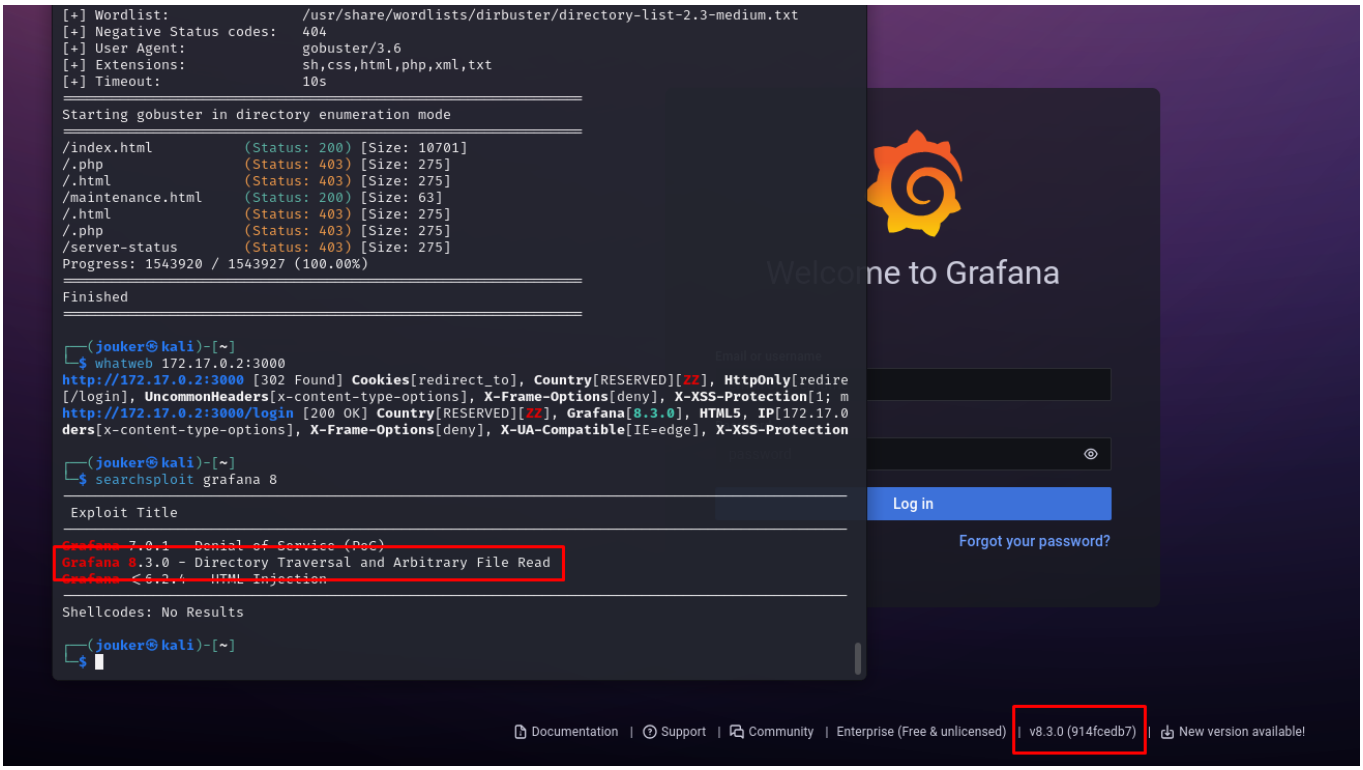
encontremos en el puerto 3000 esta ruta nos sirve



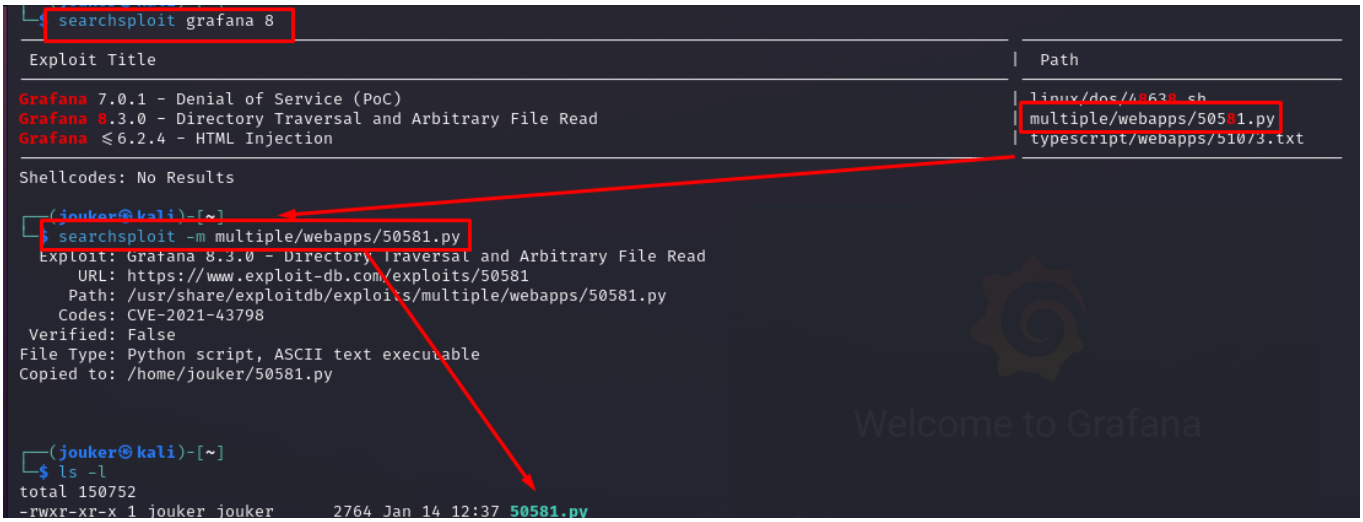
De nuevo a traves de whatweb podemos ver que lo vulnerable en este caso parece ser un login de grafana



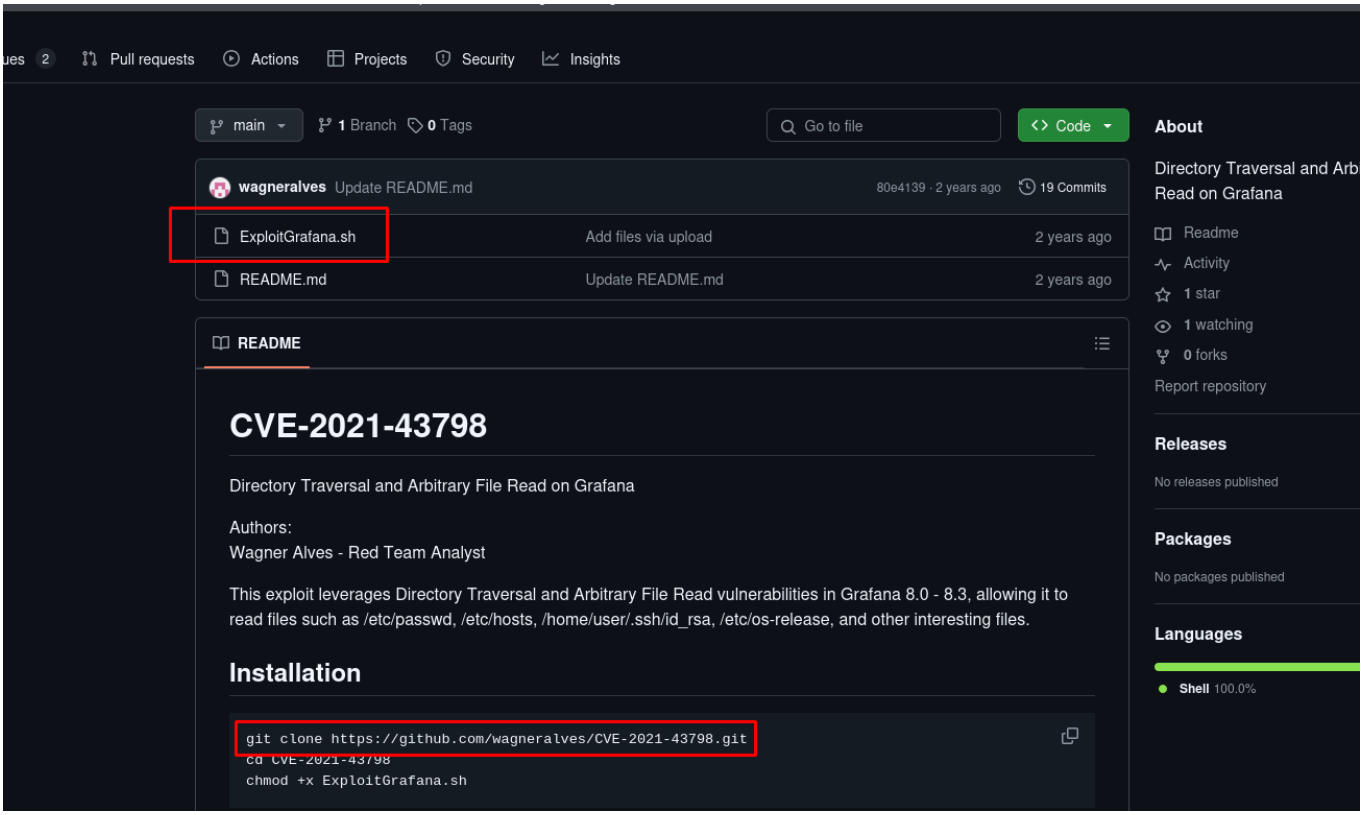
Tampoco me he ido muy lejos, haciendo la comanda searchsploit veo que hay especificamente un exploit para Grafana 8.3.0 que es la versión que ahora mismo tengo que atacar, trata sobre directory travesal y arbitrary file read, vamos a descargar ese documento para ver si este nos sirve para vulnerar la página



Usamos searchsploit -m para descargar el archivo que queríamos solicitar.

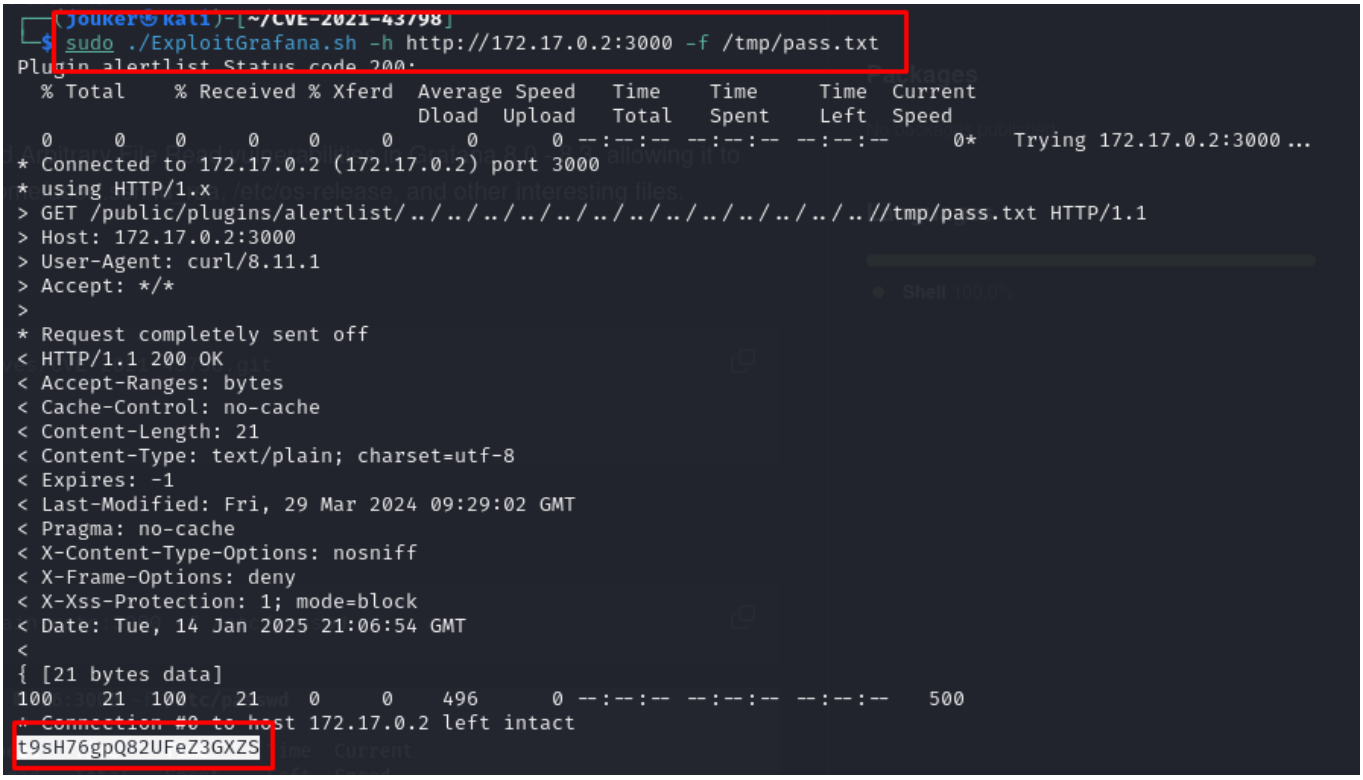


Despues de hacer uso del script que me he descargado, no me ha funcionado correctamente,por lo que vamos a buscar el mismo payload pero en internet y encontramos el password con el script encontrado a traves de github.



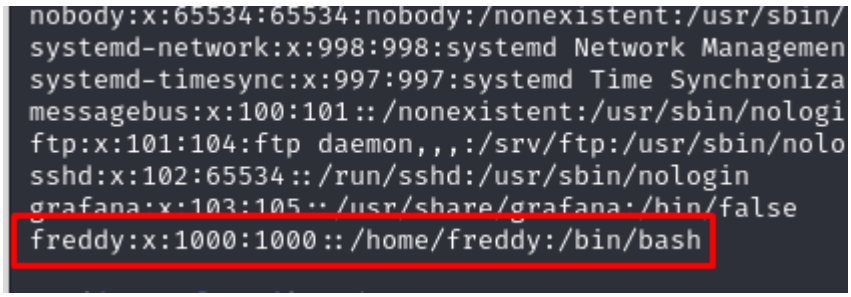
El exploit nos indica que hay que usar -h y -f para especificar lugar y el archivo que queremos encontrar, al parecer tenemos una

contraseña



He intentado buscar con fuerza bruta el usuario, pero no tiene sentido ya que simplemente al tener acceso con esta vulnerabilidad podemos listar el archivo /etc/passwd para ver los usuarios.

Este es el archivo haciendo la misma comanda que antes pero en el -f en vez de poner /tmp/pass.txt, he puesto /etc/passwd y esto nos muestra que el usuario freddy es vulnerable



Para los curiosos, el ataque de hydra seguramente funcionase ya que el nombre freddy se encuentra en la lista de usuarios del wordlist que he hecho con anterioridad

```
┌──(jouker㉿kali)-[~/CVE-2021-43798]
└─$ cat /usr/share/wordlists/seclists/Usernames/xato-net-10-million-usernames.txt | grep "freddy"
freddy
freddy1
freddy69
freddyk
freddy12
freddyg
freddyy
freddyb
freddyz
freddyp
freddym
```

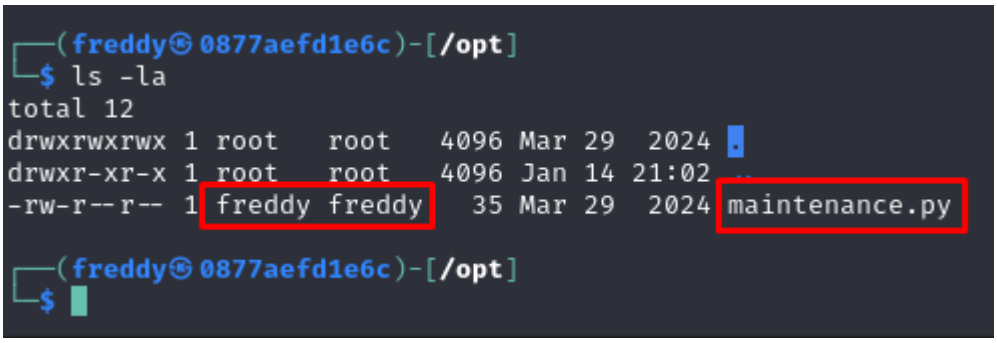
Nos conectamos por ssh y vemos que al poner las credenciales obtenidas ANTES somos directamente freddy.

```
Host key verification failed.

┌──(jouker㉿kali)-[~]
└─$ ssh freddy@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:vI77ttzFmsp8NiCsxBpeZipRCZ9MdfkeMJojz7qMiTw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
freddy@172.17.0.2's password:
Linux 0877aefd1e6c 6.11.2-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.11.2-1kali1 (2024-10-15) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
┌─(Message from Kali developers)
│
│ This is a minimal installation of Kali Linux, you likely
│ want to install supplementary tools. Learn how:
│ ⇒ https://www.kali.org/docs/troubleshooting/common-minimum-setup/
│
└─(Run: "touch ~/.hushlogin" to hide this message)
┌──(freddy㉿0877aefd1e6c)-[~]
└─$
```

Dentro del usuario freddy vemos con sudo -l que puede ejecutar codigo python a un archivo maintenance.py

```
└─(Run: "touch ~/.hushlogin" to hide this message)
┌──(freddy㉿0877aefd1e6c)-[~]
└─$ sudo -l
Matching Defaults entries for freddy on 0877aefd1e6c:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User freddy may run the following commands on 0877aefd1e6c:
    (ALL) NOPASSWD: /usr/bin/python3 /opt/maintenance.py
```

De dicho archivo freddy es propietario, por lo que le podemos asignar los permisos que queramos y editar el contenido de dentro para invocar una shell como superusuario.



Os adjunto un cat de lo que he metido dentro del maintenance.py para que se vea como invocar una shell, es tan simple como eso, ya que al ejecutar la tarea como sudo la bin/bash en vez de ser normal es de super usuario.