PENETRATION TEST SAMPLE REPORT: WINDOWS 10 SYSTEM

Presented By: Thien Anh

Date: July 10, 2025

Introduction & Objectives

What is a Penetration Test?

- A simulated cyberattack against a system to identify vulnerabilities.
- Proactive security measure to strengthen defenses.

Objectives for Windows 10 System:

- Identify unpatched vulnerabilities and misconfigurations.
- Assess the effectiveness of existing security controls.
- Attempt to gain unauthorized access and escalate privileges.
- Evaluate potential for data exfiltration.
- Provide actionable recommendations for remediation.

Methodology: Our Approach

1. Scope Definition

Agreed-upon target(s) and acceptable testing methods (e.g., Specific Windows 10 workstation, internal network access).

2. Information Gathering

Passive: OSINT (Open-Source Intelligence); Active: Network scanning (Nmap), service and user enumeration.

3. Vulnerability Analysis

Automated scanning (e.g., Nessus) for CVEs, manual analysis of configurations and services.

4. Exploitation

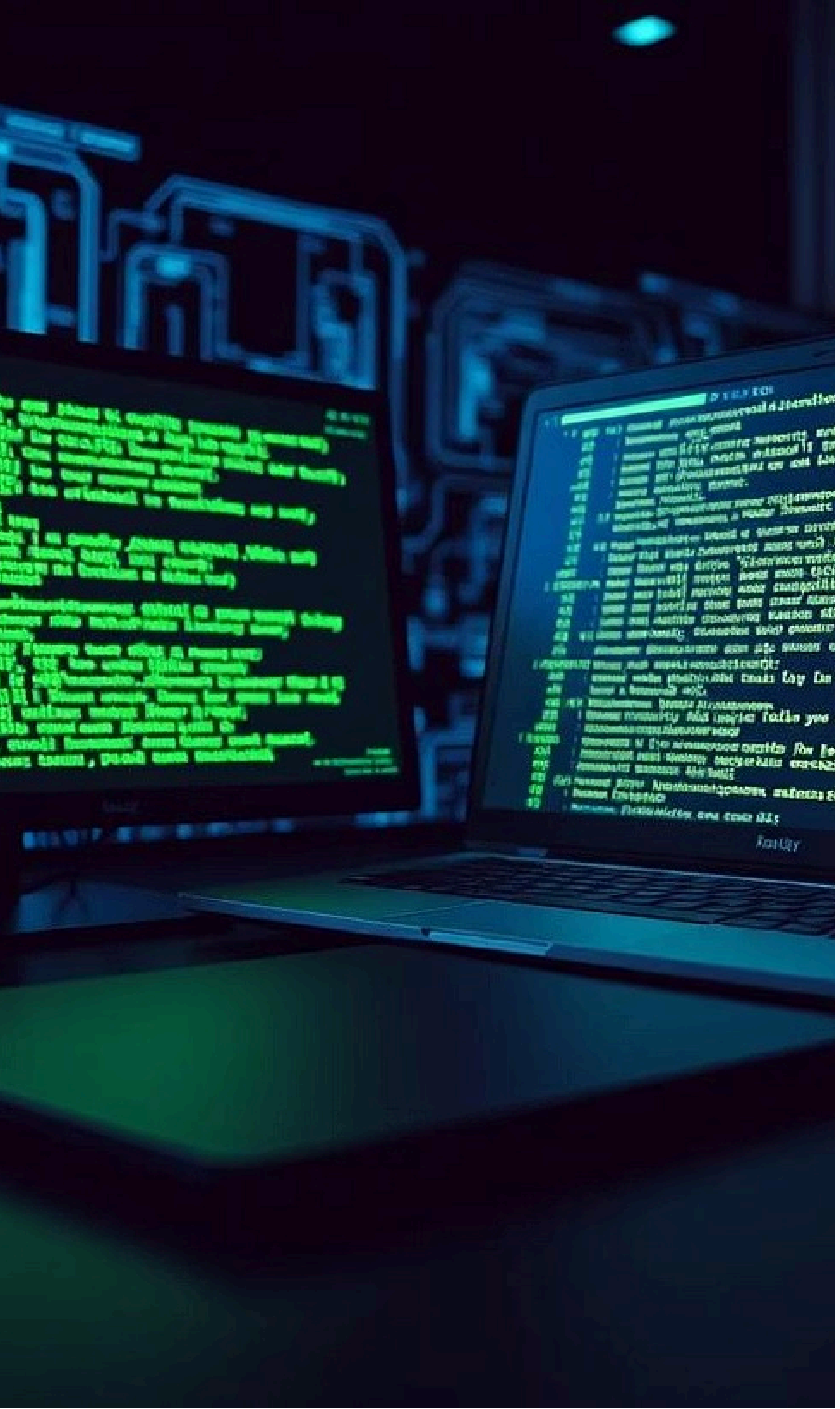
Leveraging identified vulnerabilities to gain access or escalate privileges, focusing on non-disruptive techniques.

5. Post-Exploitation

Maintaining persistence, lateral movement, data exfiltration simulation to understand breach impact.

6. Reporting

Documenting comprehensive findings, assessed risks, and actionable recommendations.



Tools & Techniques Utilized

Our penetration test employed a diverse suite of industry-standard tools to thoroughly assess the Windows 10 system's security posture.

Network & Vulnerability

- **Nmap:** Port scanning, service version detection, OS fingerprinting.
- **Responder:** LLMNR/NBT-NS poisoning for credential harvesting.
- **Nessus/OpenVAS:** Automated vulnerability identification and reporting.

Exploitation & Post-Exploitation

- **Metasploit Framework:** Extensive exploit modules, payloads, and post-exploitation capabilities.
- **CrackMapExec:** Post-exploitation tool for Active Directory environments.
- **Mimikatz:** Extracting plaintext passwords and hashes from memory.
- **BloodHound:** Mapping Active Directory relationships for privilege escalation.

Key Findings & Examples

Unpatched Critical OS Vulnerability

Windows 10 system vulnerable to **CVE-202X-XXXXX (Remote Code Execution in SMBv3)** due to outdated patches. A remote attacker could execute arbitrary code with elevated privileges. Successful remote shell obtained.

Weak Local Administrator Passwords

Several local administrator accounts used easily guessable or common passwords. This allows for local privilege escalation and potential lateral movement.

Insecure Service Permissions

A critical service configured with an unquoted path, enabling an attacker to inject malicious executables and escalate privileges from a low-privileged user to SYSTEM.

Outdated Software/Applications

Third-party applications (e.g., web browser, PDF reader) had known vulnerabilities, posing a risk for client-side exploitation and initial access.

Risk Assessment & Impact: HIGH

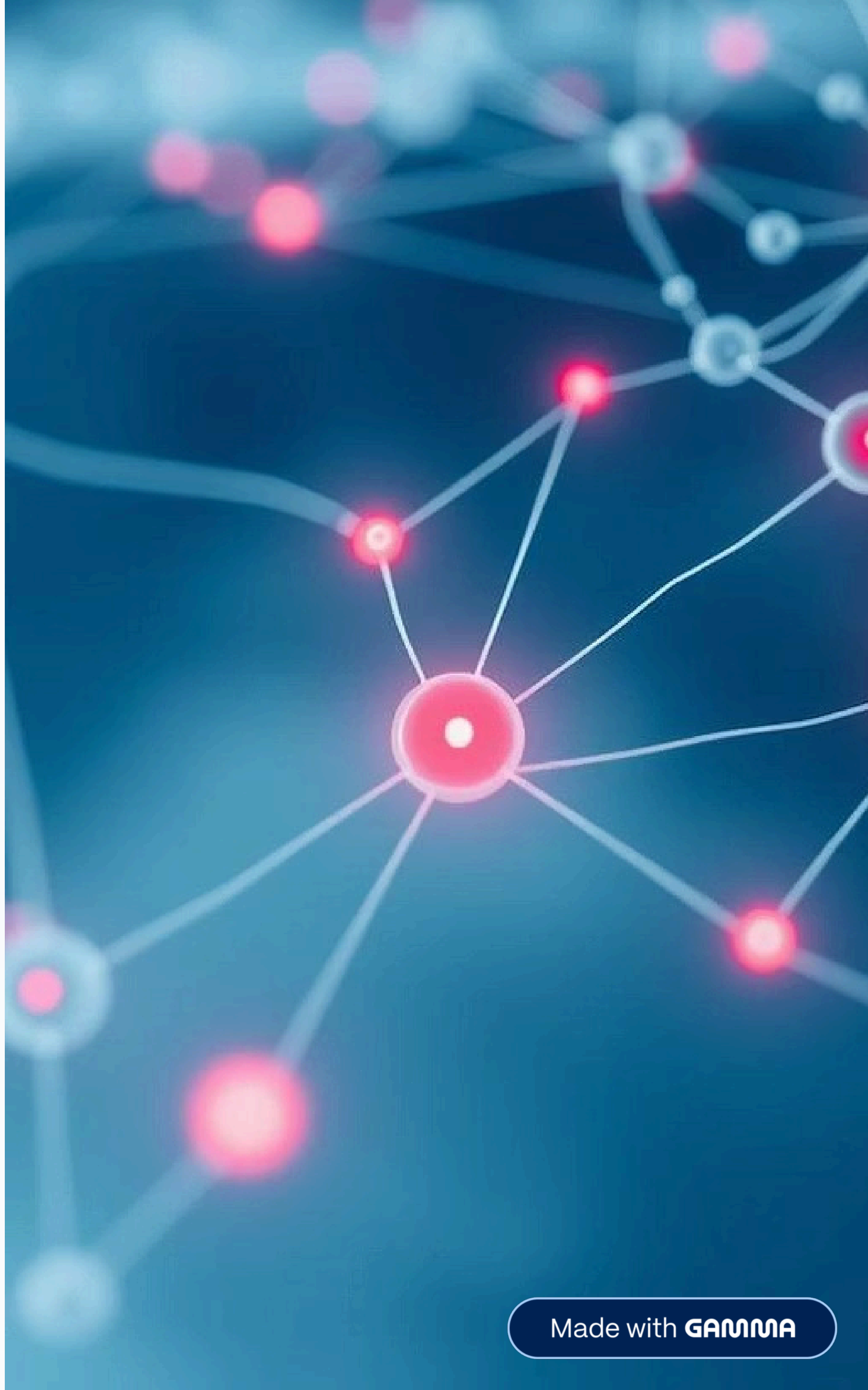
The identified vulnerabilities present a significant risk, exposing the system to severe compromise and far-reaching business implications.

Potential Impact of Exploitation

- **Confidentiality:** Unauthorized access to sensitive data (credentials, system files).
- **Integrity:** Tampering with system configurations, data modification, malware deployment.
- **Availability:** Potential for denial of service and critical system disruption.

Business Implications

- **Reputational Damage:** Erosion of trust among clients and partners.
- **Regulatory Non-Compliance:** Potential fines and legal repercussions.
- **Financial Losses:** Due to data breaches, operational downtime, and recovery efforts.



Recommendations & Next Steps

Prioritized actions to strengthen your Windows 10 system against identified vulnerabilities.

Immediate Patching

Apply all critical and high-priority Windows security updates. Establish a robust patch management process.

Strong Password Policies & MFA

Enforce complex password requirements for all accounts, especially administrators. Implement multi-factor authentication (MFA).

Principle of Least Privilege

Restrict user and service account privileges to the absolute minimum necessary for functionality.

Application Hardening

Update and securely configure third-party applications. Remove all unnecessary software.

Endpoint Detection and Response (EDR)

Deploy and configure EDR solutions for continuous monitoring and advanced threat detection.

Regular Vulnerability Scanning

Implement a schedule for routine vulnerability assessments to proactively identify new risks.

Security Awareness Training

Educate users on identifying and reporting suspicious activities, such as phishing attempts.

Questions & Discussion