Name	Aarush Maheshwari
UID	2023200076
Batch	A4

Experiment: FCAPS Network Management (Windows & Ubuntu)

Aim

To study and implement the FCAPS (Fault, Configuration, Accounting, Performance, and Security) network management model using Windows and Ubuntu operating systems.

Objectives

- To understand the FCAPS network management model
- To identify and analyze network faults
- To perform configuration management on networked systems
- To monitor accounting and performance parameters
- To apply basic security management techniques

Tools / Software Required

- Windows 10/11 OS
- Ubuntu Linux OS
- Command Prompt / PowerShell
- Nmap
- Wireshark
- SNMP utilities
- iPerf
- Windows Defender Firewall / UFW

1. Fault Management

Objective: To detect unreachable or faulty devices in the network.

Windows Steps

1. Open Command Prompt.
2. Execute: ping <IP_address>

```
C:\Users\Admin>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=1ms TTL=116
Reply from 8.8.8.8: bytes=32 time=1ms TTL=116
Reply from 8.8.8.8: bytes=32 time=1ms TTL=116
Reply from 8.8.8.8: bytes=32 time=1ms TTL=116

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 1ms, Average = 1ms

C:\Users\Admin>
```

Observation: The ping 8.8.8.8 command is used to test network connectivity to a remote host.The output shows all packets were successfully received with 0% packet loss and an average response time of 1 ms, indicating a stable and fast network connection.

3. Observe packet loss or timeout messages.

4. Execute: tracert <IP_address>

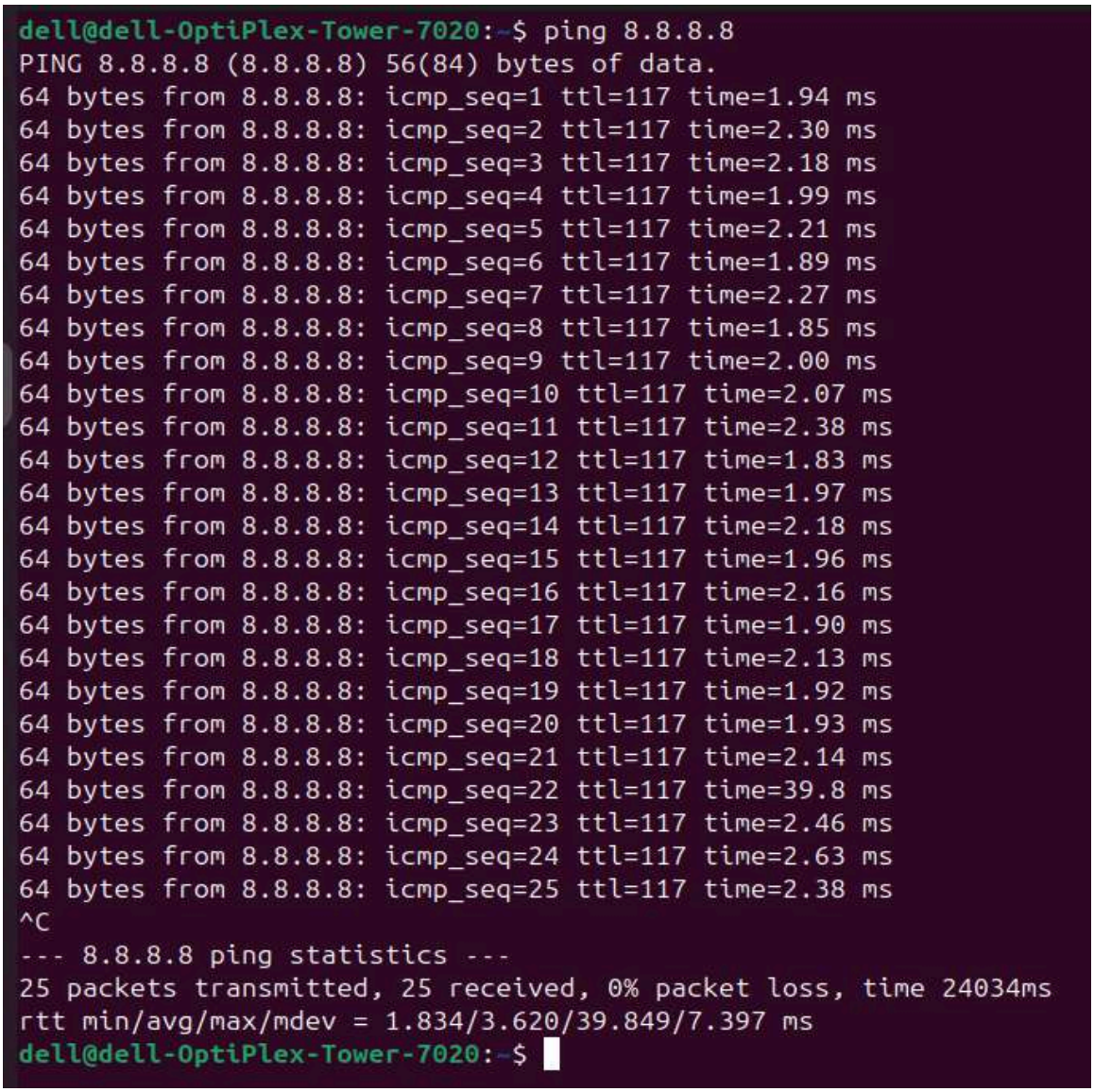
```
C:\Users\CCN>tracert 8.8.8.8

Tracing route to dns.google [8.8.8.8]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  10.10.53.1
  2    <1 ms    <1 ms    <1 ms  194.0.0.1
  3    11 ms     3 ms     1 ms  static-5.21.248.49-tataidc.co.in [49.
248.21.5]
  4     *        *        *     Request timed out.
  5     2 ms     *        *     10.129.21.54
  6     2 ms     2 ms     2 ms  72.14.210.20
  7    27 ms    22 ms    20 ms  216.239.57.17
  8     2 ms     2 ms     2 ms  142.251.69.105
  9     *        2 ms     2 ms  dns.google [8.8.8.8]

Trace complete.

C:\Users\CCN>show startup-config
'show' is not recognized as an internal or external command,
operable program or batch file.
```

Observation: The tracert 8.8.8.8 command displays the path taken by packets from the local system to the destination host.The output shows multiple intermediate routers with their response times, and a few timed-out hops, indicating normal routing behavior before successfully reaching the destination.
5. Identify where the connection fails.

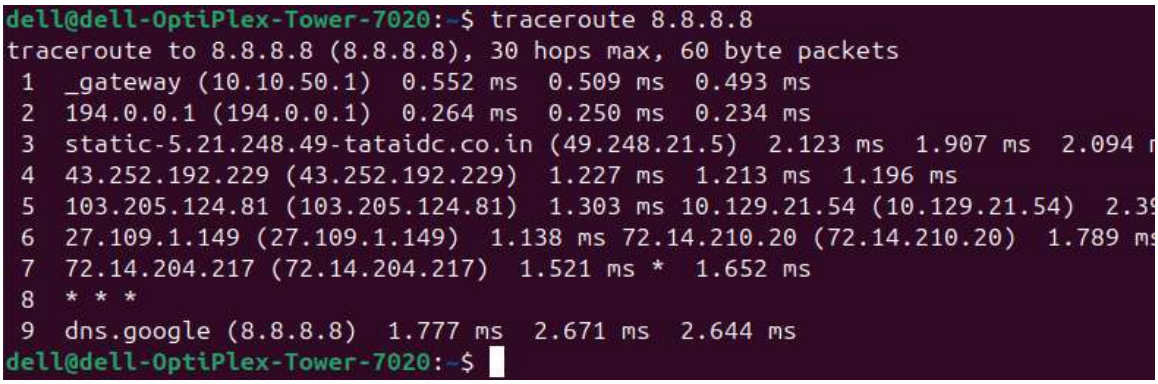
Connection fails at hop 4 where the request has been timed out.

Ubuntu Steps

1. Open Terminal.
2. Execute: ping <IP_address>
3. Stop ping using Ctrl+C.



Observation: The ping 8.8.8.8 command checks reachability of the destination host from the system.The output shows 25 packets transmitted and received with 0% packet loss and low average delay, indicating a stable and reliable network connection.
4. Execute: traceroute <IP_address>



Observation: The traceroute 8.8.8.8 command displays the sequence of network hops between the source and destination.The output lists intermediate gateways with their response times and successfully reaches the destination, showing normal packet routing across the network.

5. Identify unreachable hops.

Hop 8 is unreachable

2. Configuration Management

Objective: To view and manage network configuration details.

Windows Steps

1. Open Command Prompt.
2. Execute: ipconfig /all

```
C:\Users\Admin>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : DESKTOP-TGGJDPC
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) Ethernet Connection (17) I219-LM
   Physical Address. . . . . . . . . : E8-CF-83-38-45-50
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::57a4:d56b:7035:b6c3%5(Preferred)
   IPv4 Address. . . . . . . . . . . : 10.10.50.209(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.254.0
   Lease Obtained. . . . . . . . . . : 03 February 2026 09:38:25
   Lease Expires . . . . . . . . . . : 03 February 2026 10:08:25
   Default Gateway . . . . . . . . . : 10.10.50.1
   DHCP Server . . . . . . . . . . . : 10.10.50.1
   DHCPv6 IAID . . . . . . . . . . . : 115920771
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2F-AE-FF-D9-E8-CF-83-38-45-50
   DNS Servers . . . . . . . . . . . : 172.16.10.4
                                       172.16.10.5
                                       8.8.8.8
   NetBIOS over Tcpip. . . . . . . . : Enabled

C:\Users\Admin>
```

Observation: The ipconfig /all command displays detailed network configuration of the system.The output shows IP address, subnet mask, default gateway, DNS servers, and DHCP status, confirming correct network interface configuration.
3. Note IP address, gateway, and DNS details.

IP address : 10.10.50.209

Gateway : 10.10.50.1

DNS servers: 172.16.10.4 and 172.16.10.5
4. (Optional) Use SNMP tools to fetch system info.

Ubuntu Steps

1. Open Terminal.
2. Execute: ifconfig OR ip a

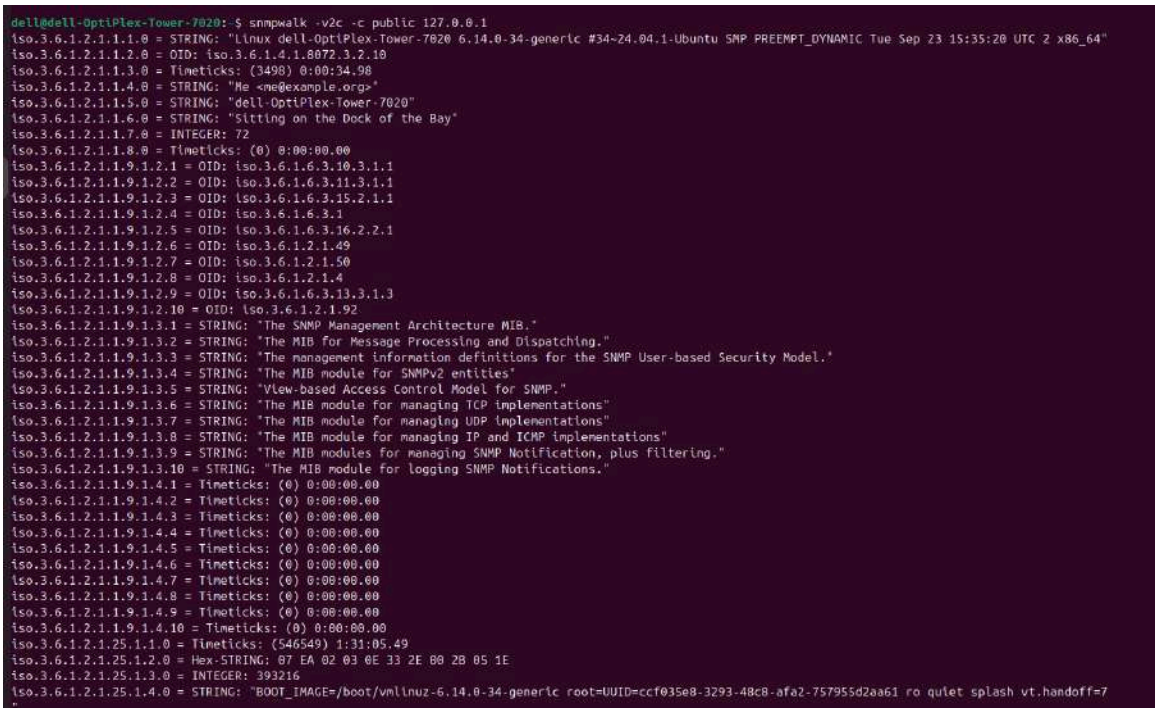
```
dell@dell-OptiPlex-Tower-7020:~$ ifconfig
enp0s31f6: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.10.50.209  netmask 255.255.254.0  broadcast 10.10.51.255
        inet6 fe80::567:c182:206c:7e72  prefixlen 64  scopeid 0x20<link>
        ether e8:cf:83:38:45:50  txqueuelen 1000  (Ethernet)
        RX packets 149536  bytes 135605544 (135.6 MB)
        RX errors 0  dropped 1861  overruns 0  frame 0
        TX packets 45838  bytes 22173329 (22.1 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  memory 0x70500000-70520000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 2524  bytes 265075 (265.0 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2524  bytes 265075 (265.0 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

dell@dell-OptiPlex-Tower-7020:~$
```

Observation: The ifconfig command provides information about network interfaces and their status.The output indicates the assigned IP address, MAC address, packet statistics, and confirms that the interface is active and running.

3. Execute: hostnamectl

```
dell@dell-OptiPlex-Tower-7020:~$ hostnamectl
 Static hostname: dell-OptiPlex-Tower-7020
       Icon name: computer-desktop
         Chassis: desktop 🖥️
      Machine ID: 2a59fb1414ad424d92ee1fe56fafb474
         Boot ID: 48cb61b73c3043eba46b23f04b83d133
Operating System: Ubuntu 24.04.2 LTS
          Kernel: Linux 6.14.0-34-generic
    Architecture: x86-64
 Hardware Vendor: Dell Inc.
  Hardware Model: OptiPlex Tower 7020
Firmware Version: 1.20.0
   Firmware Date: Thu 2025-09-04
    Firmware Age: 5month
dell@dell-OptiPlex-Tower-7020:~$
```

Observation: The hostnamectl command shows system identification and operating system details.The output displays the system hostname, OS version, kernel, architecture, and hardware information, confirming successful system configuration.
4. Use SNMP: snmpwalk -v2c -c public <IP_address> system



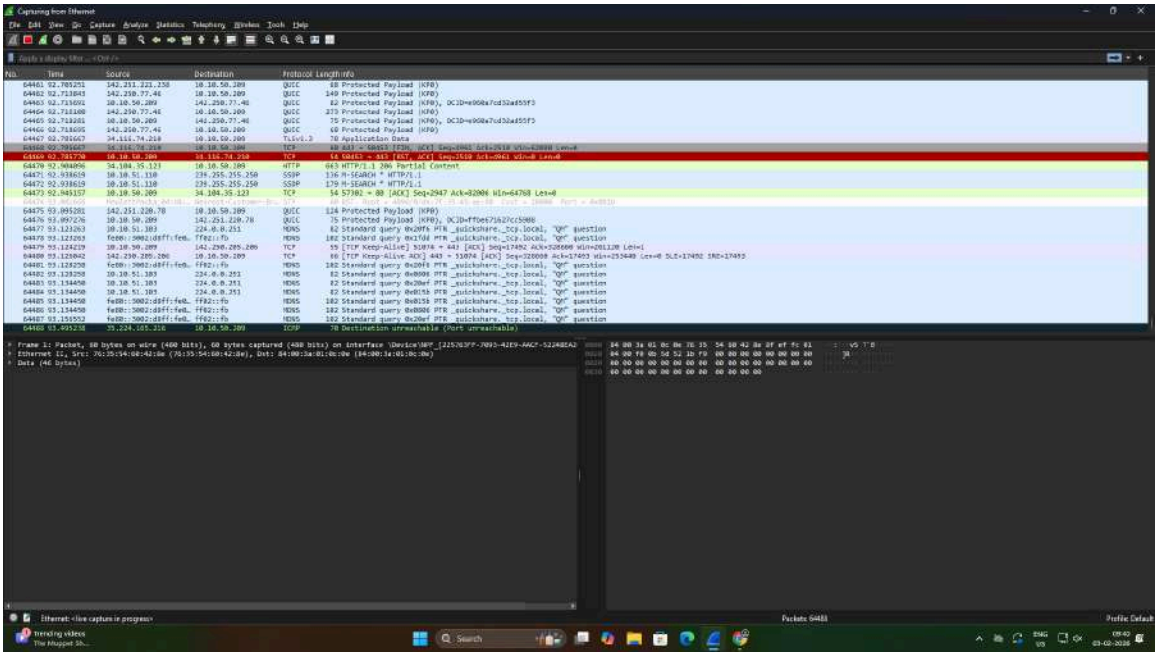
Observation: The snmpwalk -v2c -c public 127.0.0.1 command retrieves SNMP Management Information Base (MIB) data from the local system.
The output lists system details such as OS version, hostname, uptime, and SNMP modules, indicating that the SNMP service is active and responding correctly.

3. Accounting Management

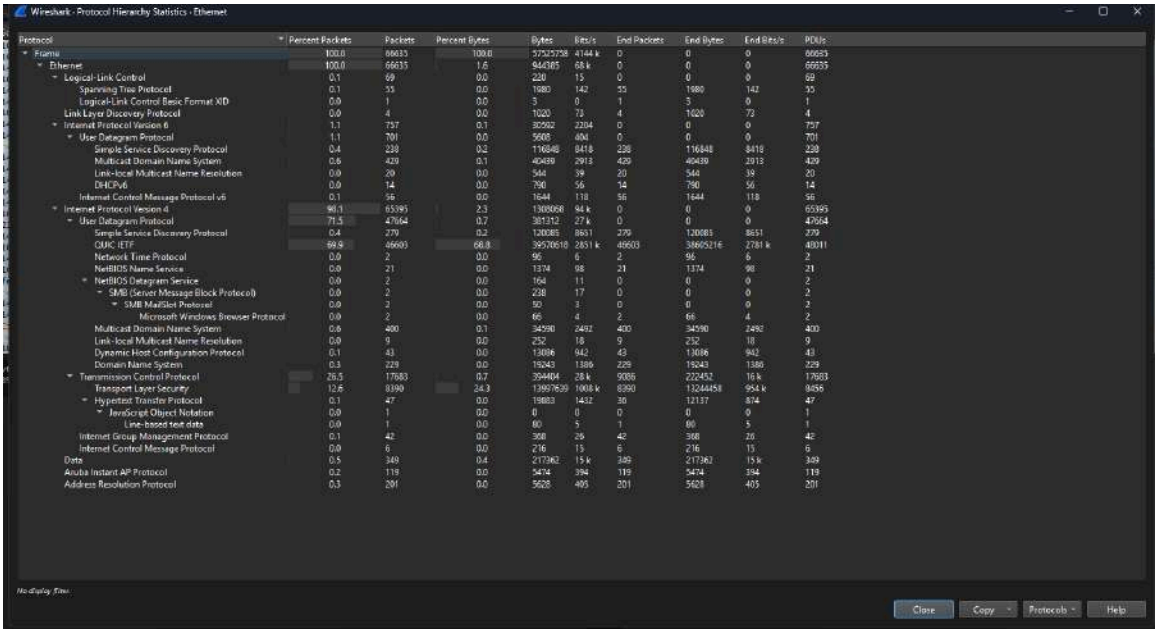
Objective: To monitor network usage.

Windows Steps

1. Launch Wireshark.
2. Select active network interface.
3. Start capture and generate network traffic.



Observation: The screenshot shows live network traffic captured using Wireshark. Different protocols such as TCP, UDP, DNS, and QUIC are visible along with source/destination IPs, indicating real-time packet flow and communication analysis.
4. Use Statistics → Protocol Hierarchy to analyze usage.



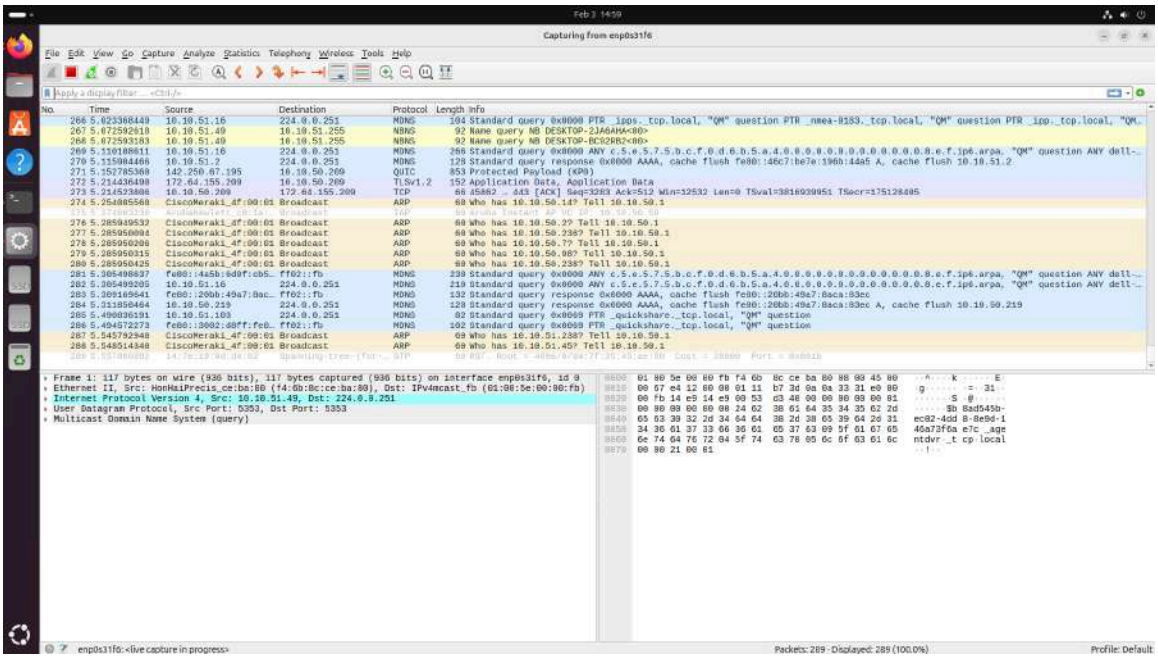
Observation: This summarizes captured traffic based on protocol distribution.
The output shows the percentage and packet count for each protocol, helping analyze network usage patterns and dominant protocols in the captured data.

Ubuntu Steps

1. Start Wireshark with sudo privileges.

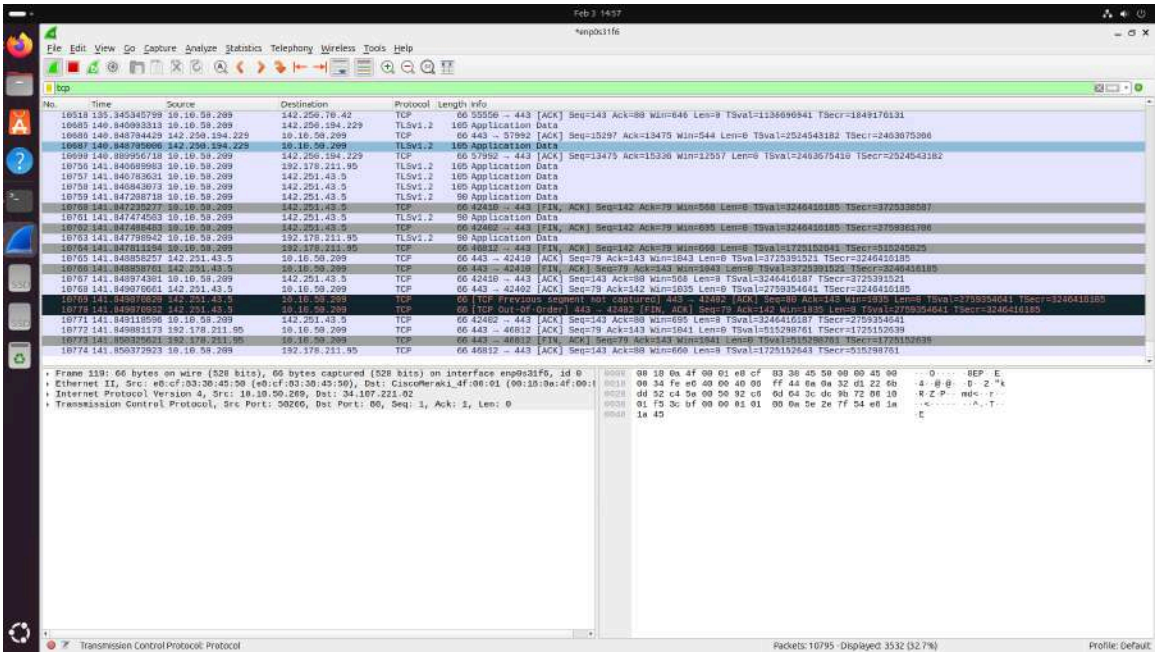
```
dell@dell-OptiPlex-Tower-7020:~$ sudo wireshark
[sudo] password for dell:
 ** (wireshark:4763) 14:59:35.949127 [GUI WARNING] -- QStandardPaths: XDG_RUNTIM
E_DIR not set, defaulting to '/tmp/runtime-root'
 ** (wireshark:4763) 14:59:40.133190 [Capture MESSAGE] -- Capture Start ...
 ** (wireshark:4763) 14:59:40.186481 [Capture MESSAGE] -- Capture started
 ** (wireshark:4763) 14:59:40.186540 [Capture MESSAGE] -- File: "/tmp/wireshark_
enp0s31f64507J3.pcapng"
 ** (wireshark:4763) 15:01:41.764104 [Capture MESSAGE] -- Capture Stop ...
 ** (wireshark:4763) 15:01:41.785440 [Capture MESSAGE] -- Capture stopped.
 ** (wireshark:4763) 15:01:41.785498 [Capture WARNING] ./ui/capture.c:722 -- cap
ture_input_closed():
```

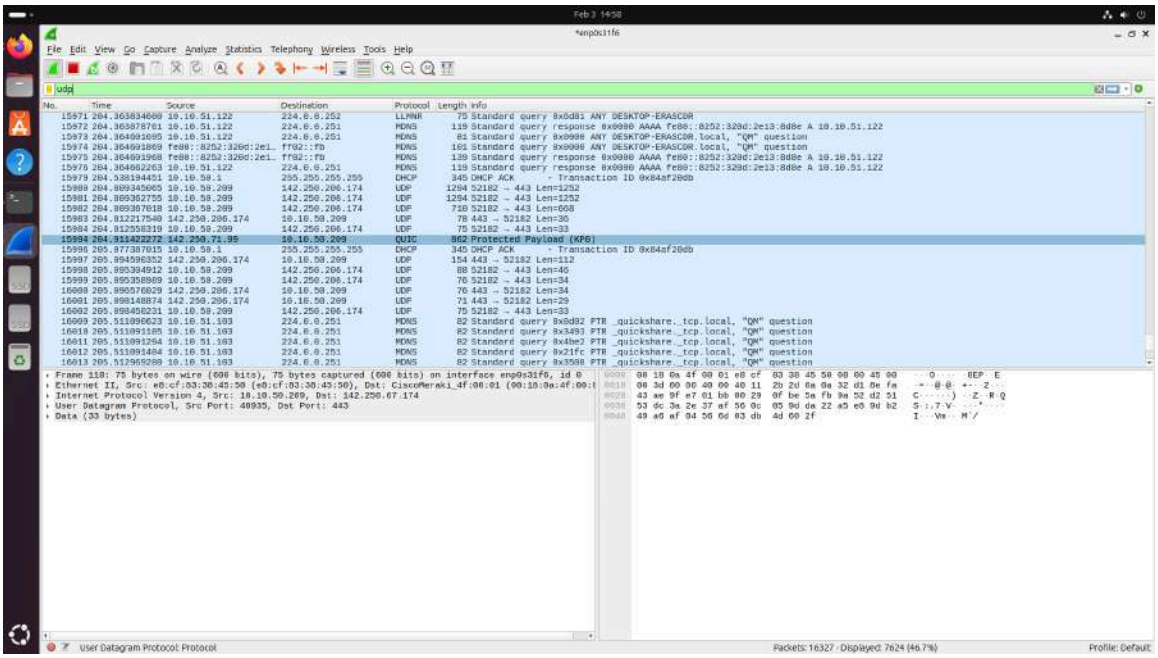
2. Capture packets on active interface.



Observation : The screenshot shows packet capture containing DNS queries, ARP requests, and responses. This indicates normal name resolution and address mapping activity occurring within the local network.

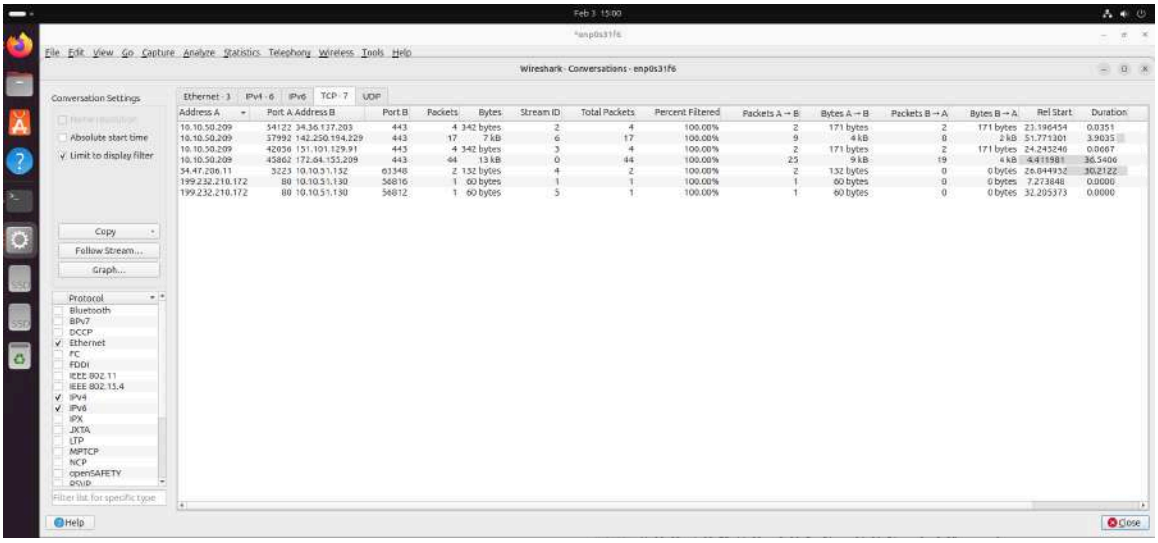
3. Apply filters such as tcp, udp.

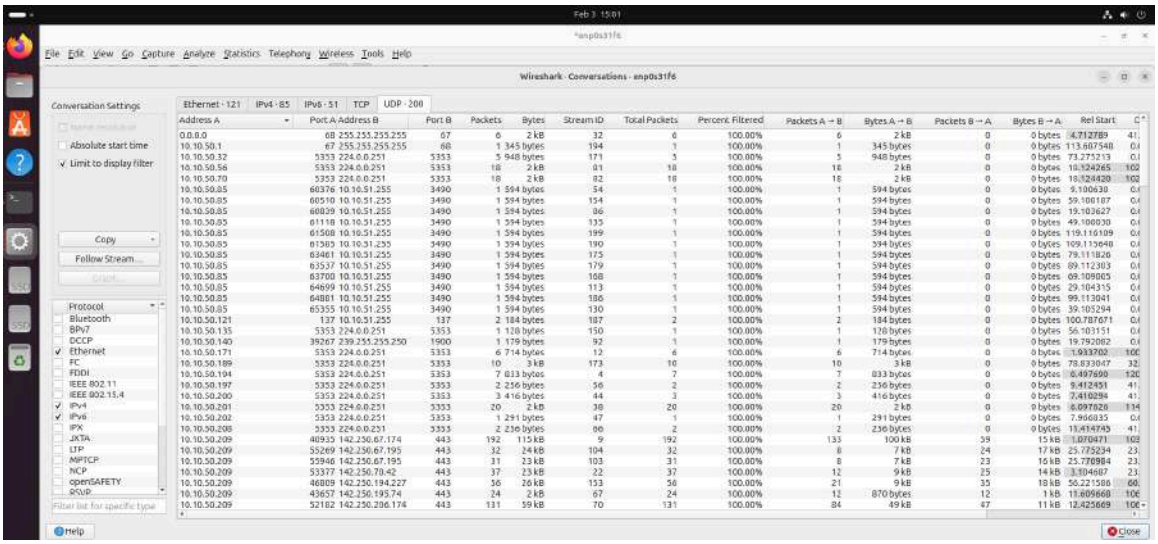




Observation : The captures highlight TCP and UDP packets with sequence numbers, acknowledgements, and flags. It demonstrates active TCP and UDP sessions and reliable data transfer between communicating hosts.

4. Analyze packet counts and bandwidth usage.





Observation : This displays active IPv4 conversations between source and destination IP addresses.It summarizes packet counts and data volume exchanged, helping identify major communication pairs.

4. Performance Management

Objective: To evaluate network performance.

Windows Steps

1. Install iPerf.
2. Run server: iperf -s
3. On another system, run: iperf -c <server_IP>
4. Note throughput values.

```
PS C:\Users\Admin\Downloads\iperf3.20_64\iperf3.20> .\iperf3.exe -s
-----------------------------------------------------------
Server listening on 5201 (test #1)
-----------------------------------------------------------
Accepted connection from 10.10.51.136, port 56358
[  5] local 10.10.51.139 port 5201 connected to 10.10.51.136 port 56359
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.01   sec  92.8 MBytes   774 Mbits/sec
[  5]   1.01-2.01   sec  94.6 MBytes   794 Mbits/sec
[  5]   2.01-3.00   sec  95.1 MBytes   799 Mbits/sec
[  5]   3.00-4.00   sec  94.5 MBytes   794 Mbits/sec
[  5]   4.00-5.00   sec  94.0 MBytes   790 Mbits/sec
[  5]   5.00-6.00   sec  94.9 MBytes   796 Mbits/sec
[  5]   6.00-7.01   sec  96.2 MBytes   797 Mbits/sec
[  5]   7.01-8.01   sec  95.5 MBytes   802 Mbits/sec
[  5]   8.01-9.01   sec  95.2 MBytes   801 Mbits/sec
[  5]   9.01-10.01  sec  94.8 MBytes   794 Mbits/sec
[  5]  10.01-10.01  sec   128 KBytes   700 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-10.01  sec   948 MBytes   794 Mbits/sec                  receiver
-----------------------------------------------------------
Server listening on 5201 (test #2)
-----------------------------------------------------------
```

Observation : The iperf3 -s command starts the system in server mode to measure network performance. The output shows successful client connections and high throughput values, indicating efficient data transfer over the network.

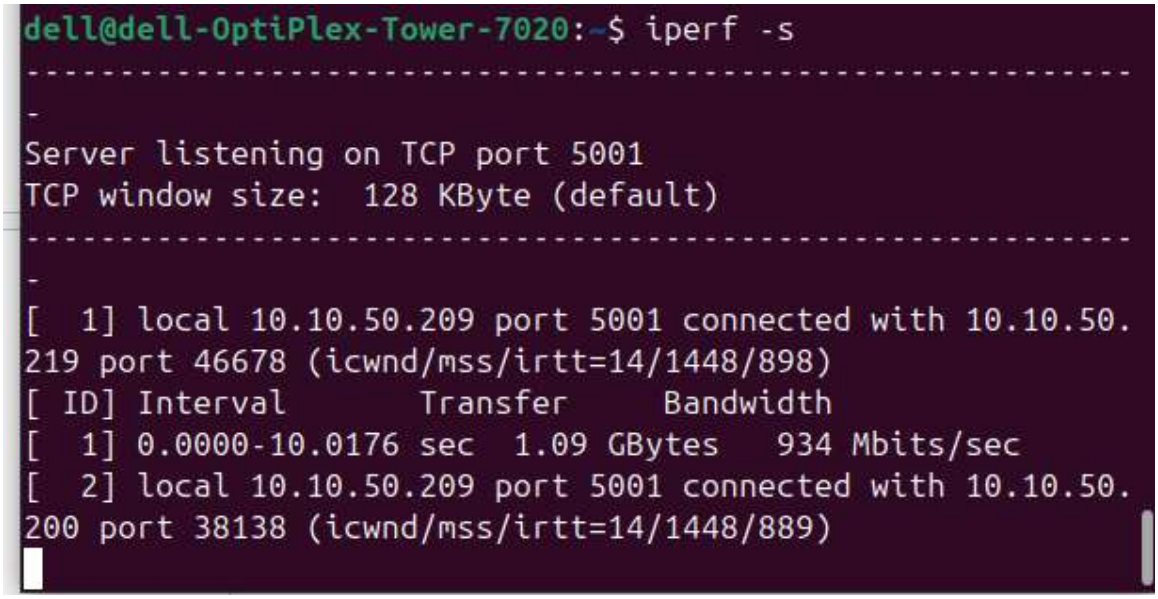
```
PS C:\Users\Admin\Downloads\iperf3.20_64\iperf3.20> .\iperf3.exe -c 10.10.51.136
Connecting to host 10.10.51.136, port 5201
[  5] local 10.10.51.139 port 49092 connected to 10.10.51.136 port 5201
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-1.01   sec  78.2 MBytes   648 Mbits/sec
[  5]   1.01-2.01   sec  77.0 MBytes   648 Mbits/sec
[  5]   2.01-3.01   sec  76.5 MBytes   643 Mbits/sec
[  5]   3.01-4.01   sec  77.2 MBytes   650 Mbits/sec
[  5]   4.01-5.00   sec  75.2 MBytes   634 Mbits/sec
[  5]   5.00-6.01   sec  77.9 MBytes   645 Mbits/sec
[  5]   6.01-7.01   sec  75.2 MBytes   633 Mbits/sec
[  5]   7.01-8.01   sec  76.1 MBytes   642 Mbits/sec
[  5]   8.01-9.01   sec  77.2 MBytes   646 Mbits/sec
[  5]   9.01-10.01  sec  72.8 MBytes   612 Mbits/sec
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate
[  5]   0.00-10.01  sec   764 MBytes   640 Mbits/sec                  sender
[  5]   0.00-10.01  sec   763 MBytes   640 Mbits/sec                  receiver

iperf Done.
PS C:\Users\Admin\Downloads\iperf3.20_64\iperf3.20>
```

Observation: The iperf3 -c <IP> command initiates a bandwidth test toward the server.

The results display per-interval transfer and bitrate, confirming stable and consistent network performance.

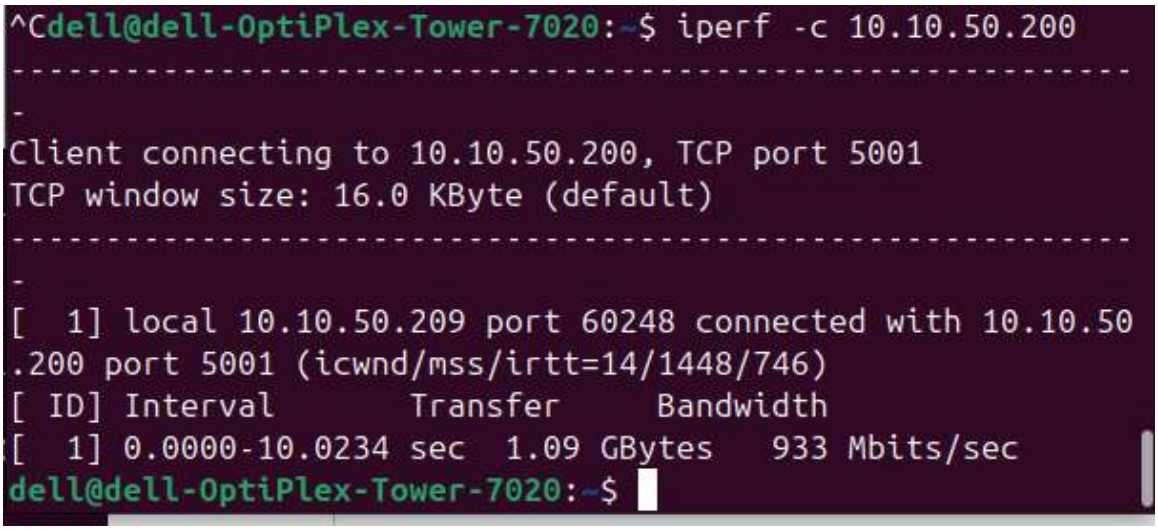
Ubuntu Steps

1. Install iPerf: sudo apt install iperf
2. Run server: iperf -s



Observation : The iperf -s command runs the system as a TCP performance measurement server. The output shows active connections with high bandwidth values, demonstrating good link capacity.

3. Run client: iperf -c <server_IP>



Observation : The iperf -c <IP> command connects to the remote server for throughput testing. The displayed bandwidth confirms successful data transmission and effective network performance.

4. Observe bandwidth and transfer rate.

Bandwidth : 1.09 GBytes

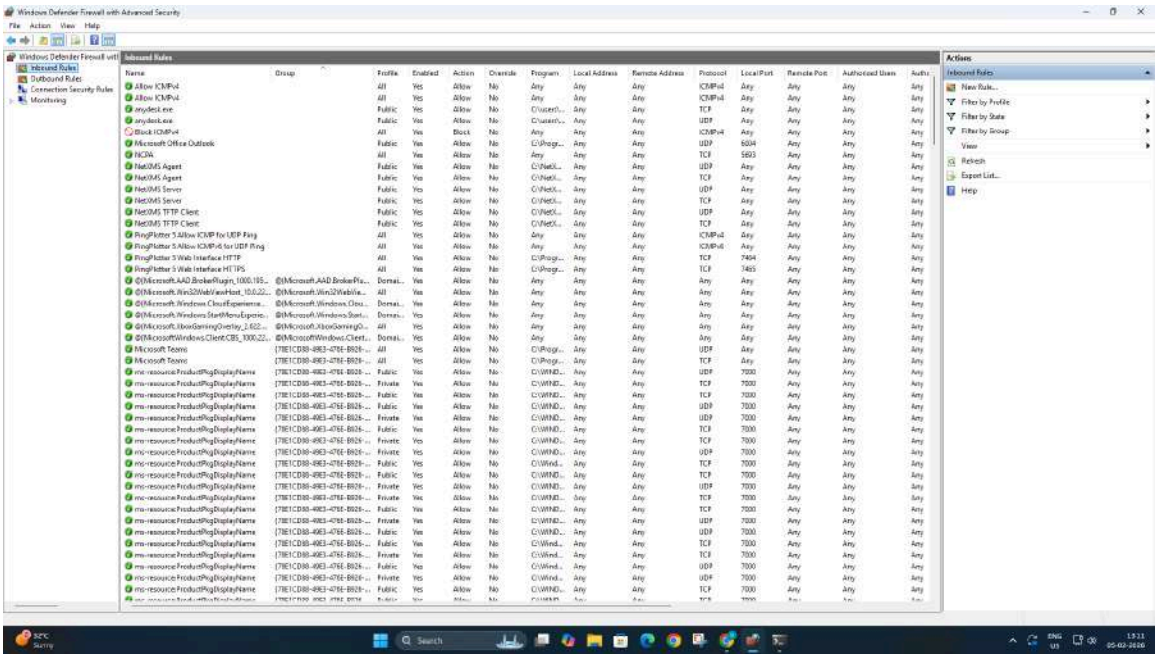
Transfer rate: 933 Mbits/sec

5. Security Management

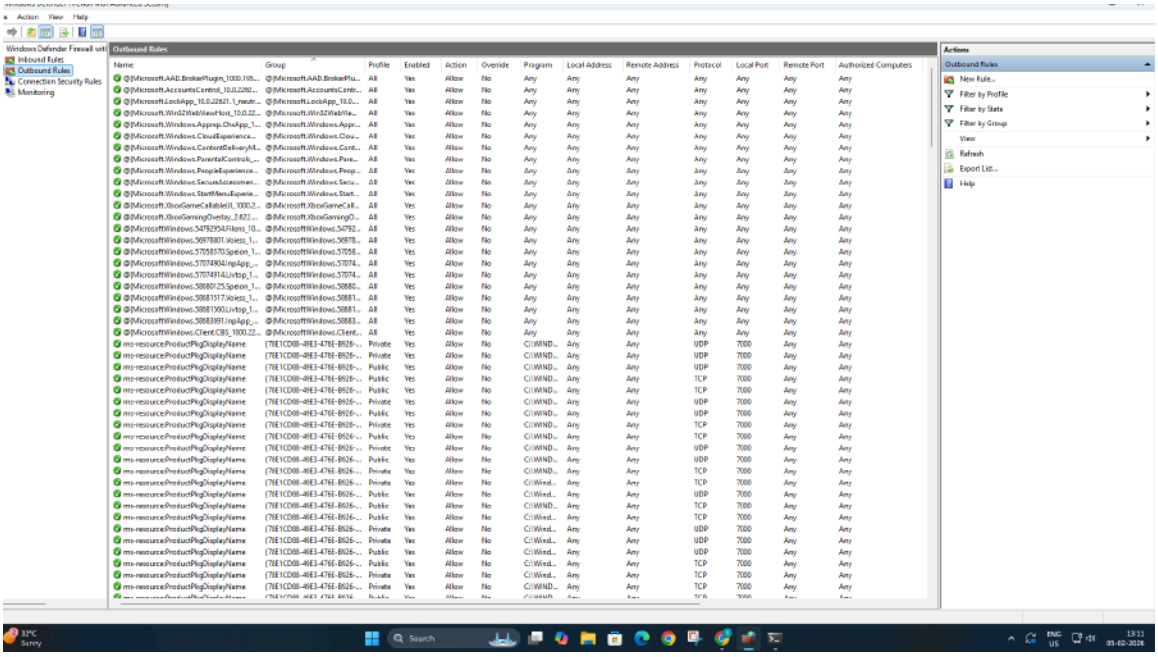
Objective: To secure the system and analyze vulnerabilities.

Windows Steps

1. Open Windows Defender Firewall.
2. Check inbound and outbound rules.



Observation : The screenshot displays configured inbound firewall rules in Windows Defender Firewall. It shows allowed and blocked rules based on protocol, port, and profile, indicating how incoming network traffic is controlled.



Observation : This shows outbound firewall rules that regulate traffic leaving the system. The rules define permissions for applications and services, helping enforce secure outbound communication.

3. Run port scan using Nmap.

```
PS C:\Users\Admin> nmap 10.10.51.161
Starting Nmap 7.98 ( https://nmap.org ) at 2026-02-05 13:10 +0530
Nmap scan report for 10.10.51.161
Host is up (0.0040s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
5357/tcp open  wsdapi
MAC Address: E8:CF:83:37:F8:11 (Dell)

Nmap done: 1 IP address (1 host up) scanned in 5.22 seconds
PS C:\Users\Admin>
```

Observation : The nmap <IP> command scans the target system for open TCP ports and services. The output identifies active services such as MSRPC, NetBIOS, and SMB, providing visibility into exposed network services.

4. Disable unused services if any.

Ubuntu Steps

1. Scan open ports: nmap <IP_address>

```
dell@dell-OptiPlex-Tower-7020:~$ nmap 10.10.50.209
Starting Nmap 7.94SVN ( https://nmap.org ) at 2026-02-03 14:10 IST
Nmap scan report for dell-OptiPlex-Tower-7020 (10.10.50.209)
Host is up (0.00015s latency).
All 1000 scanned ports on dell-OptiPlex-Tower-7020 (10.10.50.209) are in ignored states.
Not shown: 1000 closed tcp ports (conn-refused)

Nmap done: 1 IP address (1 host up) scanned in 0.10 seconds
```

Observation : The nmap 10.10.50.209 command scans the local system for open TCP ports.The output shows all scanned ports are closed or filtered, indicating no active services are exposed.

```
dell@dell-OptiPlex-Tower-7020:~$ nmap 8.8.8.8
Starting Nmap 7.94SVN ( https://nmap.org ) at 2026-02-03 14
:12 IST
Nmap scan report for dns.google (8.8.8.8)
Host is up (0.0026s latency).
Not shown: 993 filtered tcp ports (no-response), 2 filtered
 tcp ports (host-unreach)
PORT     STATE  SERVICE
53/tcp   open   domain
113/tcp  closed ident
443/tcp  open   https
2000/tcp open   cisco-sccp
5060/tcp open   sip
```

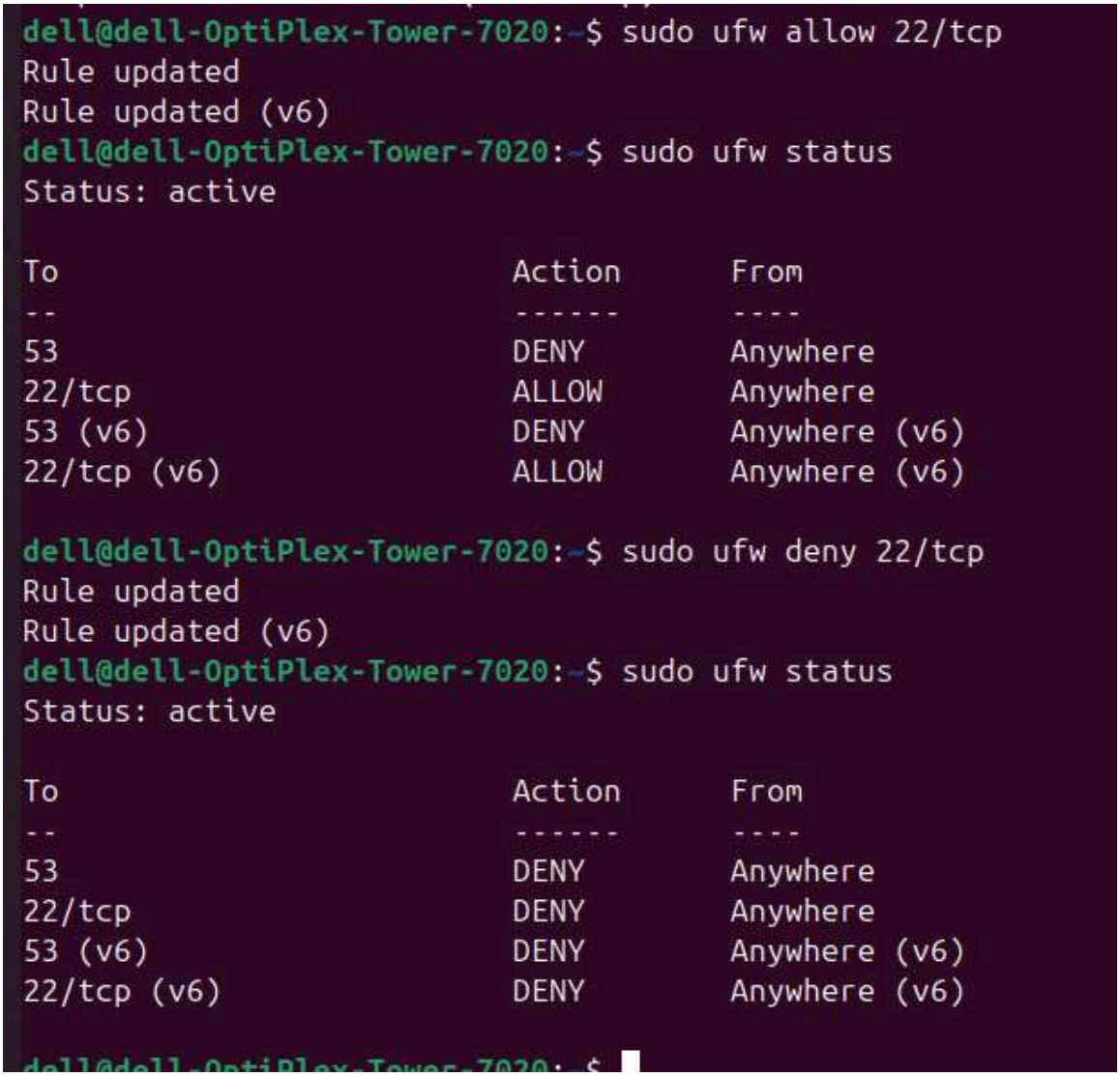
Observation : The nmap 8.8.8.8 command scans a public DNS server for open services.The results show specific open ports such as 53 and 443, indicating accessible DNS and HTTPS services.

2. Enable firewall: sudo ufw enable

```
dell@dell-OptiPlex-Tower-7020:~$ sudo ufw enable
Firewall is active and enabled on system startup
```

Observation : The sudo ufw enable command activates the Uncomplicated Firewall on the system. The output confirms that the firewall is enabled and will start automatically on system boot.

3. Block unused ports: sudo ufw deny <port>
4. Re-scan to verify security.



Observation : UFW rules are added to allow and then deny TCP port 22, followed by checking firewall status. The final status output confirms that SSH traffic is blocked, demonstrating effective firewall rule enforcement.

Conclusion : This lab successfully demonstrated the practical implementation of network management concepts based on the FCAPS model. Fault-related activities were observed through connectivity and reachability testing using commands like ping and traceroute. Configuration management was validated by examining system and network settings using ipconfig, ifconfig, hostnamectl, and firewall rule configuration.

Accounting and performance aspects were analyzed using tools such as iperf, snmpwalk, and Wireshark statistics to measure bandwidth usage and traffic distribution. Security management was demonstrated through port scanning with nmap and enforcing access control using Windows Defender Firewall and UFW. Overall, the experiment provided a comprehensive understanding of how FCAPS-based network management tools are applied in real-world systems.