



Network Scanning and Traffic Analysis Report

Prepared By: Rabya Hassan



MARCH 28, 2026

hassanrabya90@gmail.com

1. Introduction

In today's digital world, network security is a critical aspect of cybersecurity. Understanding how networks operate and how attackers analyze systems is essential for protecting sensitive data and systems.

This project focuses on two important techniques used in cybersecurity: network scanning and traffic analysis. Network scanning is used to discover devices, open ports, and services running on a system, while traffic analysis helps in understanding how data flows within a network.

In this project, Nmap was used for network scanning and Wireshark was used for analyzing network traffic.

2. Objectives

- To understand the concept of network scanning
- To identify open ports and services using Nmap
- To capture and analyze network traffic using Wireshark
- To identify potential security risks
- To suggest preventive measures for network security

3. Tools Used

3.1 Nmap

Nmap (Network Mapper) is a widely used tool for network discovery and security auditing. It helps in identifying open ports, running services, and system information.

3.2 Wireshark

Wireshark is a network protocol analyzer that captures and displays data packets traveling through a network in real time.

3.3 Kali Linux

Kali Linux was used as the operating system for performing all tasks, as it includes pre-installed cybersecurity tools.

4. Methodology

The project was divided into two main parts:

1. Network scanning using Nmap
2. Traffic analysis using Wireshark

All activities were performed in a controlled environment on the local system for educational purposes only.

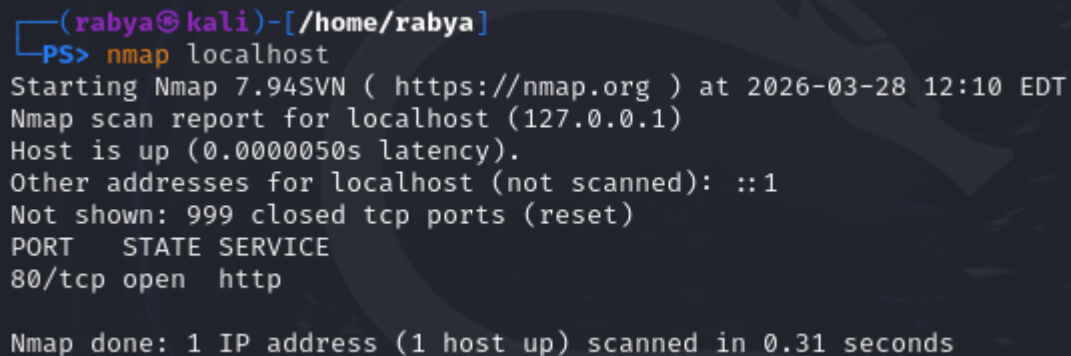
5. Network Scanning using Nmap

5.1 Basic Scan

A basic scan was performed using the following command:

```
nmap localhost
```

This scan provided information about open ports on the system.



```
(rabya@kali)-[/home/rabya]
PS> nmap localhost
Starting Nmap 7.94SVN ( https://nmap.org ) at 2026-03-28 12:10 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000050s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 0.31 seconds
```

5.2 Service Version Detection

A more detailed scan was performed using:

```
nmap -sV localhost
```

This command helped identify the services running on open ports along with their versions.

```
(rabya@kali)-[/home/rabya]
PS> nmap -sV localhost
Starting Nmap 7.94SVN ( https://nmap.org ) at 2026-03-28 12:11 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.0000060s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.62 ((Debian))

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.83 seconds
```

5.3 Advanced Scan

An advanced scan was conducted using:

```
nmap -A localhost
```

This scan provided detailed information such as:

- Operating system detection
- Service versions
- Additional network details

```
(rabya@kali)-[/home/rabya]
PS> nmap -A localhost
Starting Nmap 7.94SVN ( https://nmap.org ) at 2026-03-28 12:14 EDT
Nmap scan report for localhost (127.0.0.1)
Host is up (0.00012s latency).
Other addresses for localhost (not scanned): ::1
Not shown: 999 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.4.62 ((Debian))
|_http-server-header: Apache/2.4.62 (Debian)
|_http-title: Apache2 Debian Default Page: It works
Device type: general purpose
Running: Linux 2.6.X
OS CPE: cpe:/o:linux:linux_kernel:2.6.32
OS details: Linux 2.6.32
Network Distance: 0 hops
```

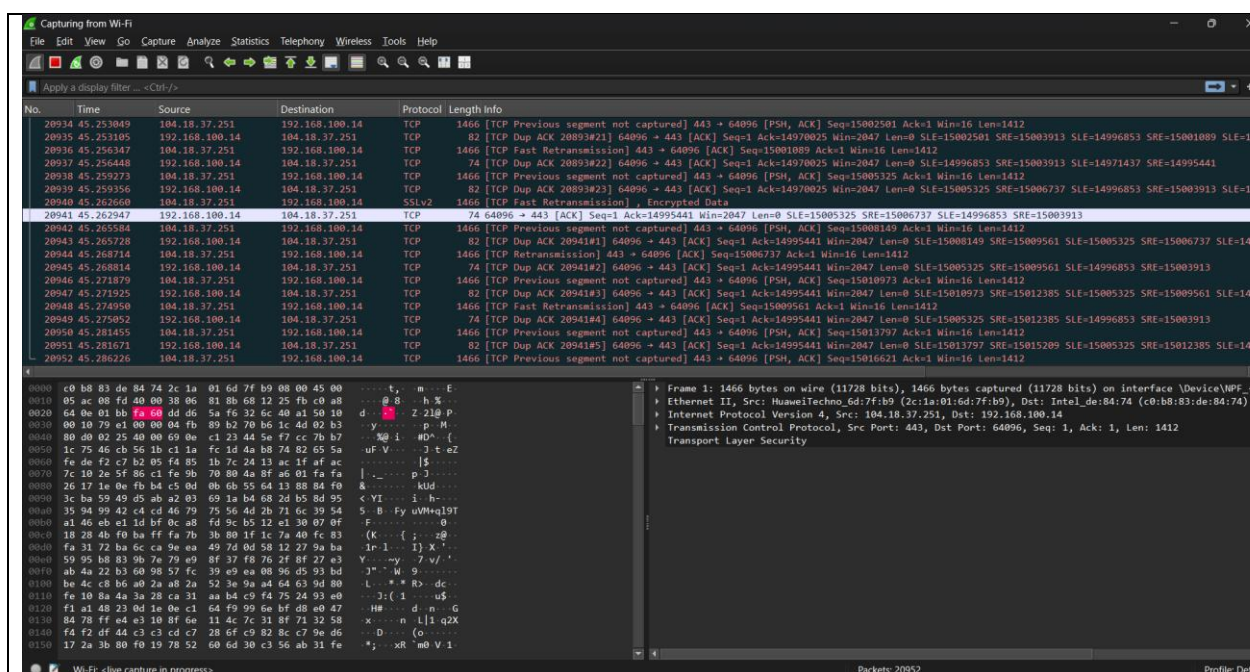
5.4 Observations

- Several ports were found open on the system
- Different services were running on these ports
- Advanced scanning provided more detailed system information

6. Traffic Analysis using Wireshark

6.1 Packet Capture

Wireshark was used to capture live network traffic. The network interface was selected and packet capturing was started.



6.2 Traffic Generation

Network activity was generated by browsing websites during the capture process.

NETWORK SCANNING AND TRAFFIC ANALYSIS REPORT

Wireshark packet capture showing TCP traffic. The packet list on the right shows a TCP Reset (RST) packet with Seq=41481737 and Len=1412.

6.3 Applying Filters

Different filters were applied to analyze specific types of traffic:

- **DNS Filter:**
Used to observe domain name system queries

Wireshark packet capture showing DNS traffic. The packet list on the right shows a DNS query for 'api.msn.com' and a response from 'api.msn.com'.

NETWORK SCANNING AND TRAFFIC ANALYSIS REPORT

- **HTTP Filter:**
Used to analyze web traffic

The screenshot displays a Wireshark capture of network traffic. The top pane shows a list of captured packets, with the selected packet (No. 51366) being an HTTP GET request. The middle pane shows the details of this packet, including the Ethernet II header, Internet Protocol Version 4 header, and the Hypertext Transfer Protocol (HTTP) section. The bottom pane shows the raw bytes of the packet in hexadecimal and ASCII format.

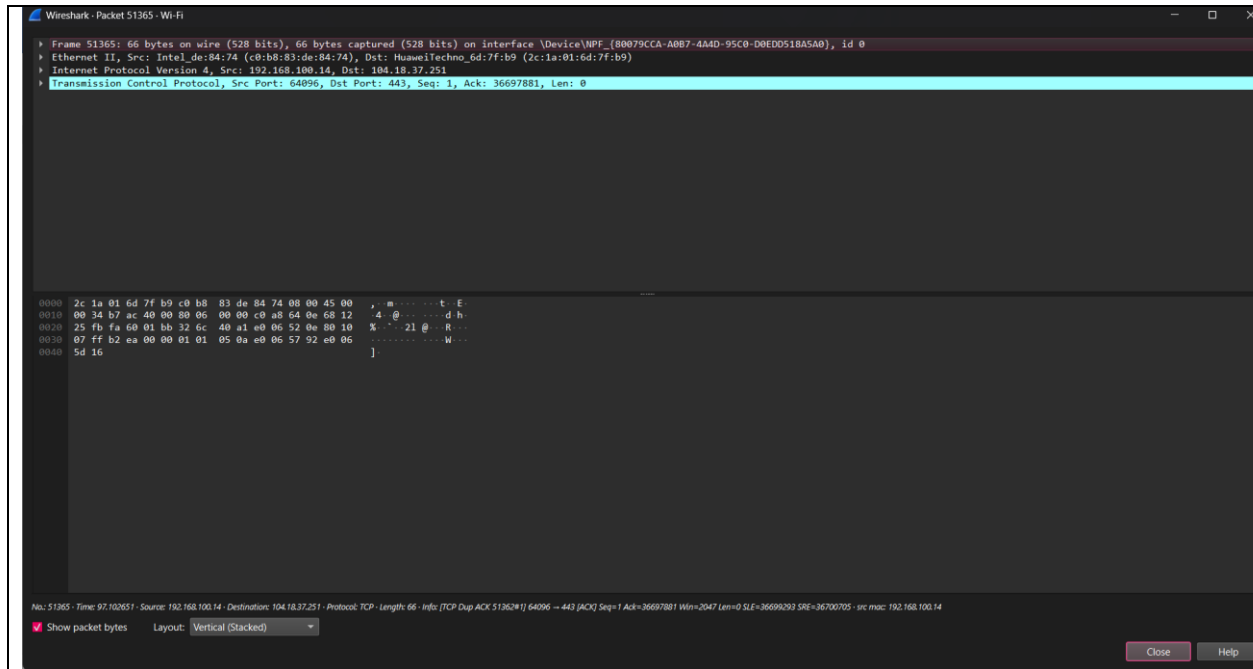
- **TCP Filter:**
Used to view transmission control protocol packets

The screenshot displays a Wireshark capture of network traffic. The top pane shows a list of captured packets, with the selected packet (No. 51365) being a TCP segment. The middle pane shows the details of this packet, including the Ethernet II header, Internet Protocol Version 4 header, and the Transmission Control Protocol (TCP) section. The bottom pane shows the raw bytes of the packet in hexadecimal and ASCII format.

6.4 Packet Analysis

Individual packets were selected to examine detailed information such as:

- Source and destination IP addresses
- Protocol type
- Packet size and data



6.5 Observations

- Network traffic consists of multiple protocols such as DNS, HTTP, and TCP
- Some data can be visible if not encrypted
- Packet-level analysis provides deep insight into communication

7. Findings

- Open ports act as entry points into a system
- Running services can expose vulnerabilities if not secured
- Network traffic can reveal sensitive information if not encrypted
- Attackers can use scanning and traffic analysis to gather information about targets

8. Security Risks

- Unauthorized access through open ports
- Exposure of sensitive data through unencrypted traffic
- Information leakage about system configuration
- Increased attack surface due to unnecessary services

9. Prevention Techniques

9.1 For Systems

- Close unused ports
- Disable unnecessary services
- Use firewalls to restrict access

9.2 For Networks

- Use secure protocols (HTTPS instead of HTTP)
- Encrypt sensitive data
- Monitor network traffic regularly

9.3 General Practices

- Keep systems updated
- Use strong authentication methods
- Conduct regular security audits

10. Conclusion

This project provided practical knowledge of network scanning and traffic analysis using Nmap and Wireshark. It demonstrated how systems can be analyzed and how data flows within a network.

The project also highlighted the importance of securing open ports and encrypting network traffic to prevent potential cyber attacks. Understanding these concepts is essential for building a strong foundation in cybersecurity.