AMITY SCHOOL OF ENGINEERING & TECHNOLOGY

DATA COMMUNICATION & COMPUTER NETWORK LAB

LABWORK – 03

(Experiment: TCP Three-Way Handshake)

COURSE NAME: DCCN Lab

COURSE CODE: CSE2306

DEPARTMENT: Computer Science & Engineering

FACULTY NAME: Dr. Sarang Maruti Patil

SUBMITTED BY

STUDENT NAME:

ENROLLMENT NUMBER:

CLASS:

SEMESTER:

DATE OF SUBMISSION:

Aim:

TCP Three-Way Handshake

Objective:

To capture and analyze the TCP Three-Way Handshake process.

- To understand the roles of SYN, SYN-ACK, and ACK packets in establishing a TCP connection.

- To learn how to calculate the Round-Trip Time (RTT) from capture data.

Tools Required:

A computer with Wireshark installed.

- Internet access (LAN or Wi-Fi).
- A web browser (Google Chrome, Firefox, etc.).

Theory:

TCP (Transmission Control Protocol) is connection-oriented and uses a Three-Way Handshake to establish a reliable connection between a client and a server.

Steps:

1. SYN → Client sends a synchronization request to start communication.

2. SYN-ACK → Server responds, acknowledging the request and synchronizing back.

3. ACK → Client acknowledges the server's response, and the connection is established.

4. Procedure

Step 1: Start Wireshark

1. Open Wireshark.

2. Select your active network interface and start capturing.

Step 2: Generate TCP Traffic

1. Open a web browser.

2. Visit http://example.com (HTTP).

Step 3: Stop the Capture

1. Click the red square Stop button in Wireshark.

Step 4: Filter TCP Packets

1. In Display Filter bar, type: tcp

2. Press Enter.

Step 5: Identify the Three-Way Handshake

1. Find first SYN packet (Client → Server).

2. Find SYN, ACK packet (Server → Client).

3. Find ACK packet (Client → Server).

Step 6: Analyze the Details

1. Check Sequence and Acknowledgment Numbers.

2. Verify ACK numbers.

Step 7: Calculate RTT

1. Right-click SYN packet → Set Time Reference.

2. Check time difference for SYN-ACK packet.

Expected Observations

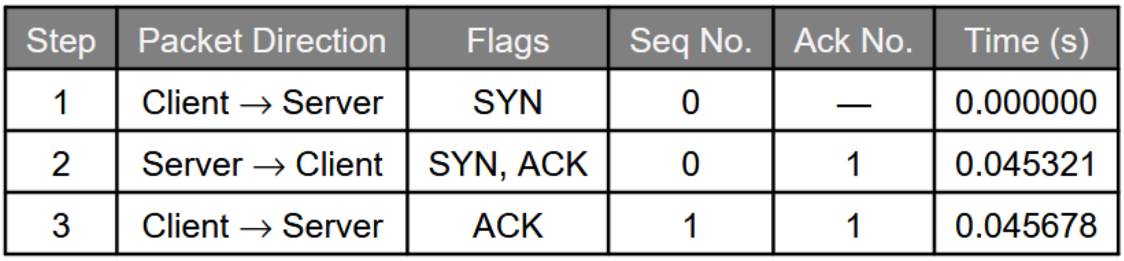
- Packet 1 (SYN) → Client initiates connection.

- Packet 2 (SYN, ACK) → Server acknowledges and sends own SYN.

- Packet 3 (ACK) → Client confirms connection ready.

- RTT value for handshake.

6. Example Output Table



Step	Packet Direction	Flags	Seq No.	Ack No.	Time (s)
1	Client → Server	SYN	0	—	0.000000
2	Server → Client	SYN, ACK	0	1	0.045321
3	Client → Server	ACK	1	1	0.045678

Questions

1. Why does TCP require a three-step handshake instead of two steps?

2. What is the RTT for the handshake in your capture?

3. How are sequence and acknowledgment numbers related?

4. What could cause a delay in the SYN-ACK response?

8. Precautions

- Ensure you visit an HTTP site, not HTTPS.

- Do not run this experiment on restricted networks without permission.

Step Packet Direction Flags Seq No. Ack No. Time (s)

1 Client → Server SYN 0 — 0.000000

2 Server → Client SYN, ACK 0 1 0.045321

3 Client → Server ACK 1 1