



Transport Layer

1. What is the Transport Layer?

- The **Transport Layer** is **Layer 4** in the **OSI model** (and part of the TCP/IP model).
- Its main job: **provide reliable or fast end-to-end communication between applications running on different devices.**
- It sits **between the Application Layer and Network Layer.**

👉 In simple words:

It's like a **delivery service** → takes a package (data) from one app, ensures it gets delivered to the right app on the other device, and sometimes confirms safe delivery.

2. Key Responsibilities

1. Segmentation & Reassembly

- Breaks large messages into smaller segments.
- Reassembles them correctly at the receiver.

2. End-to-End Communication

- Ensures data moves between the correct apps on two hosts.
- Uses **port numbers** (e.g., HTTP → port 80, HTTPS → 443).

3. Connection Control

- Can establish, maintain, and terminate sessions.
- Two types: **Connection-oriented (TCP)** vs **Connectionless (UDP)**.

4. Reliability

- Provides error detection, retransmission, and acknowledgment (if using TCP).

5. Flow Control

- Prevents sender from overwhelming receiver.

6. Multiplexing/Demultiplexing

- Handles multiple applications using the same network simultaneously (e.g., YouTube + Zoom + WhatsApp).

3. Main Transport Layer Protocols

Protocol	Full Form	Type	Features	Used In
TCP	Transmission Control Protocol	Connection-oriented	Reliable, ordered, error-checked, acknowledgments	Web (HTTP/HTTPS), Email (SMTP/IMAP), File Transfer (FTP), gRPC
UDP	User Datagram Protocol	Connectionless	Faster, lightweight, no guarantees	Streaming (YouTube), Gaming, VoIP (calls), DNS
QUIC	Quick UDP Internet Connections	Built on UDP	Combines speed of UDP + reliability/security (TLS built-in)	HTTP/3, Google services, Zoom
SCTP	Stream Control Transmission Protocol	Hybrid	Multi-streaming, reliable, congestion control	Telecom signaling, some VoIP

4. TCP vs UDP (Quick Comparison)

Feature	TCP	UDP
---------	-----	-----

Connection	Connection-oriented (handshake)	Connectionless
Reliability	Reliable (acknowledgments, retransmission)	Unreliable (no delivery guarantee)
Ordering	Maintains order of packets	No ordering
Speed	Slower (overhead of reliability)	Faster (low overhead)
Use Cases	Web, Email, File transfer	Video streaming, Gaming, DNS, VoIP

5. Example Flows

♦ Web Page Load (TCP)

1. Browser sends HTTP request using **TCP**.
2. TCP ensures packets arrive in order.
3. If a packet is lost, TCP retransmits.
4. User gets full webpage without corruption.

♦ Video Call (UDP)

1. Voice/video packets sent via **UDP**.
 2. If one packet is lost → no retransmission (otherwise delay).
 3. Call continues smoothly, maybe with a tiny glitch.
-

6. Transport Layer in OSI vs TCP/IP

- **OSI Model** → Transport Layer is separate (Layer 4).

- **TCP/IP Model** → Also has Transport Layer (sits above Internet Layer).

✓ In short:

The **Transport Layer** ensures **end-to-end delivery of data between applications**, balancing **reliability (TCP)** and **speed (UDP/QUIC)** depending on use case.
