
WEEK 1 – NETWORKING FOUNDATIONS

DAY 5 – TCP vs UDP

---

1. TCP vs UDP – Core Comparison

Networking transport protocols manage **end-to-end delivery**. TCP and UDP are the **two main transport-layer protocols**.

Feature	**TCP**	**UDP**
Type	Connection-oriented	Connectionless
Reliability	Reliable	Unreliable
Error Detection	Yes (ACKs & retransmission)	Minimal (checksum only)
Ordering	Guaranteed	Not guaranteed
Flow Control	Yes	No
Congestion Control	Yes	No
Use Cases	Web, Email, File Transfer	Streaming, DNS, VoIP
Speed	Slower	Faster

Key idea:

- **TCP** = “reliable delivery with checks”
- **UDP** = “fast delivery, no checks”

---

2. The 3-Way Handshake (TCP Connection Establishment)

TCP is **connection-oriented**, meaning a session must be established before sending data.

Steps:

1. **SYN (Synchronize)**

- Client sends SYN to server to start connection
- Includes **initial sequence number**

2. **SYN-ACK (Synchronize-Acknowledge)**

- Server replies with SYN + ACK
- Acknowledges client’s sequence number
- Provides its own sequence number

3. **ACK (Acknowledge)**

- Client acknowledges server's sequence number
- Connection is established

Result: Both sides now know **initial sequence numbers** → reliable delivery possible.

Interview sentence:

"TCP establishes a reliable connection using a 3-way handshake: SYN → SYN-ACK → ACK."

---

3. Flow Control

Problem: Sender may overwhelm receiver

TCP Solution: Sliding window mechanism

- Receiver tells sender **how many bytes it can handle**
- Sender adjusts sending speed accordingly

Analogy:

- Water pipe → regulate flow to prevent overflow

Interview line:

"TCP flow control ensures the sender does not overwhelm the receiver using the sliding window mechanism."

---

4. Congestion Control

Problem: Network may become congested (too many packets → dropped)

TCP Mechanisms:

1. **Slow Start** – Start small, increase gradually
2. **Congestion Avoidance** – Avoid overwhelming the network
3. **Fast Retransmit & Fast Recovery** – Quickly recover from lost packets

Interview line:

"TCP congestion control prevents network overload using slow start and congestion avoidance algorithms."

---

5. Ports and Sockets

Port

- Identifies a **specific application** on a device
- Range: 0 – 65535

- Categories:
    - 0 – 1023 → Well-known ports (HTTP=80, HTTPS=443, SSH=22)
    - 1024 – 49151 → Registered
    - 49152 – 65535 → Dynamic/private

Socket

- Combination of:
    - IP address
    - Port number
    - Protocol (TCP/UDP)
- Example: 192.168.1.10:80/TCP
- Uniquely identifies a communication endpoint

Interview line:

"A socket is a combination of IP, port, and protocol, uniquely identifying a communication endpoint."

---

6. UDP – Key Points

- No handshake
- No acknowledgment
- No retransmission
- Suitable for real-time applications

Examples:

- Video calls, VoIP, live streaming, DNS queries

Analogy:

"UDP is like mailing a postcard: fast but no guarantee it arrives."

---

7. Security: SYN Flood Attack

Problem: TCP handshake can be exploited

How SYN Flood Works

1. Attacker sends **many SYN requests** to server
2. Server allocates resources for each connection
3. Attacker never completes handshake (no final ACK)
4. Server memory exhausted → legitimate requests fail

Result: Denial of Service (DoS)

Mitigation

- SYN cookies → don't allocate resources until handshake completes
- Firewall rate limiting
- Intrusion Detection Systems

Interview line:

"SYN flood is a DoS attack exploiting TCP's 3-way handshake by sending numerous SYN requests without completing them."

---

8. Real-Life Example (TCP vs UDP)

- TCP → Sending an email (must be delivered correctly)
- UDP → Streaming a live match (speed more important than perfection)