**✅ TCP/IP (Transmission Control Protocol / Internet Protocol)**

**Key Features**

* **Connection-oriented**: A handshake (3-way) is established before data transfer.
* **Reliable**: Guarantees delivery, order of packets, and error-checking.
* **Flow Control**: Resends lost packets.
* **Heavyweight**: More overhead due to reliability mechanisms.
* **Slower than UDP** because of connection setup and ACK.

**Why is it used?**

* For **critical data** (e.g., firmware updates, database logs, web servers).
* IoT examples:
  + Sending data to **cloud services** (AWS, Azure, Thingspeak) via **HTTP/MQTT over TCP**.
  + Remote monitoring dashboards.

**Structure**

* IP layer → Routes packets.
* TCP layer → Handles reliability (ACK, retransmission).
* Data is **stream-based** (continuous flow).

**✅ UDP (User Datagram Protocol)**

**Key Features**

* **Connectionless**: No handshake before data transfer.
* **Unreliable**: No guarantee of delivery or order.
* **Lightweight**: Very low overhead.
* **Faster than TCP** because no connection setup or ACK.

**Why is it used?**

* For **real-time applications** where speed > reliability.
* IoT examples:
  + **Sensor broadcasting** (temperature, humidity).
  + **Video/Audio streaming**, gaming.
  + Sending quick telemetry data to a local controller.

**Structure**

* Each packet is independent (datagram).
* No retransmission if lost.

**✅ TCP vs UDP Quick Comparison Table**

| **Feature** | **TCP** | **UDP** |
| --- | --- | --- |
| **Connection** | Connection-oriented | Connectionless |
| **Reliability** | Guaranteed (ACK, retries) | No guarantee |
| **Speed** | Slower | Faster |
| **Overhead** | High | Low |
| **Use Case** | Web servers, cloud uploads | Real-time data, sensors |

**Why TCP for IoT?**

* **Cloud integration** needs guaranteed delivery (MQTT, HTTP use TCP).

**Why UDP for IoT?**

* **Fast local communication** between microcontrollers or real-time monitoring.

✅   
They often ask:  
**Q: Why is UDP used in IoT if it's unreliable?**  
👉 Answer: Because in IoT telemetry (e.g., temperature updates every second), **speed matters more than guaranteed delivery**—if one reading is lost, the next will come soon.

**Q: Which protocol does MQTT use?**  
👉 MQTT is built on **TCP** for reliability.

**✅ 1. What is TCP/IP?**

**Answer:**  
TCP/IP is a suite of protocols used for communication over the internet.

* **TCP (Transmission Control Protocol)**: Ensures reliable, ordered, and error-checked data delivery.
* **IP (Internet Protocol)**: Handles addressing and routing of packets across networks.

**✅ 2. What is UDP?**

**Answer:**  
UDP (User Datagram Protocol) is a lightweight, connectionless transport protocol that sends data without guaranteeing delivery or order. It's faster than TCP but less reliable.

**✅ 3. Key differences between TCP and UDP?**

| **Feature** | **TCP** | **UDP** |
| --- | --- | --- |
| Reliability | Reliable (ACK, retransmit) | Unreliable |
| Connection | Connection-oriented | Connectionless |
| Speed | Slower | Faster |
| Overhead | High | Low |

**✅ 4. Which is better for IoT sensor data: TCP or UDP?**

**Answer:**  
UDP is better for **real-time telemetry** (temperature, humidity) because it’s faster and small losses don’t matter. TCP is used for **critical data** like firmware updates.

**✅ 5. Why is TCP called connection-oriented?**

**Answer:**  
Because it uses a **three-way handshake** (SYN → SYN-ACK → ACK) to establish a connection before data transfer.

**✅ 6. Why is UDP called connectionless?**

**Answer:**  
UDP does not establish a session before sending data; it sends datagrams directly without handshakes.

**✅ 7. Explain the TCP three-way handshake.**

**Answer:**

1. Client → SYN → Server
2. Server → SYN-ACK → Client
3. Client → ACK → Server  
   This establishes a reliable connection.

**✅ 8. Does UDP guarantee delivery of packets?**

**Answer:**  
No, UDP does **not** guarantee delivery, order, or error correction.

**✅ 9. Why is UDP faster than TCP?**

**Answer:**  
Because UDP avoids handshakes, acknowledgments, and retransmissions, reducing latency and overhead.

**✅ 10. Can UDP be used for video streaming? Why?**

**Answer:**  
Yes, because in video streaming, **speed matters more than perfect reliability**—a lost frame can be ignored.

**✅ 11. Which protocol does MQTT use and why?**

**Answer:**  
MQTT uses **TCP** because it requires guaranteed message delivery for IoT applications.

**✅ 12. Which protocol is used for DNS: TCP or UDP?**

**Answer:**  
UDP is used for most DNS queries because it’s faster. TCP is used for large transfers like DNS zone transfers.

**✅ 13. Which layer does TCP and UDP operate on?**

**Answer:**  
They operate at the **Transport Layer** of the **OSI model** (Layer 4).

**✅ 14. Why do IoT systems often use UDP for sensor data?**

**Answer:**  
Because sensor updates are frequent, and losing a few packets does not significantly impact functionality.

**✅ 15. Can you implement TCP on Arduino?**

**Answer:**  
Yes, using Ethernet or Wi-Fi modules with TCP/IP stacks (e.g., ESP8266, W5500 Ethernet Shield).

**✅ 16. Explain ports in TCP and UDP.**

**Answer:**  
Ports identify **specific applications** or processes on a device. TCP and UDP both use port numbers (0–65535).

**✅ 17. What is congestion control and does UDP have it?**

**Answer:**  
TCP has congestion control to avoid network overload. UDP does **not** have it; it keeps sending data regardless of congestion.

**✅ 18. Why does TCP require more resources than UDP?**

**Answer:**  
TCP maintains state information (connections, ACKs, windows), while UDP is stateless.

**✅ 19. Name some IoT protocols built on TCP and UDP.**

**Answer:**

* **TCP-based:** HTTP, MQTT, FTP
* **UDP-based:** CoAP, TFTP, mDNS

**✅ 20. If UDP is unreliable, how can you ensure reliability?**

**Answer:**  
By adding **application-level checks**, like:

* Adding **ACK messages** manually.
* Using **checksums** for integrity.
* Retransmitting missing data at application level.