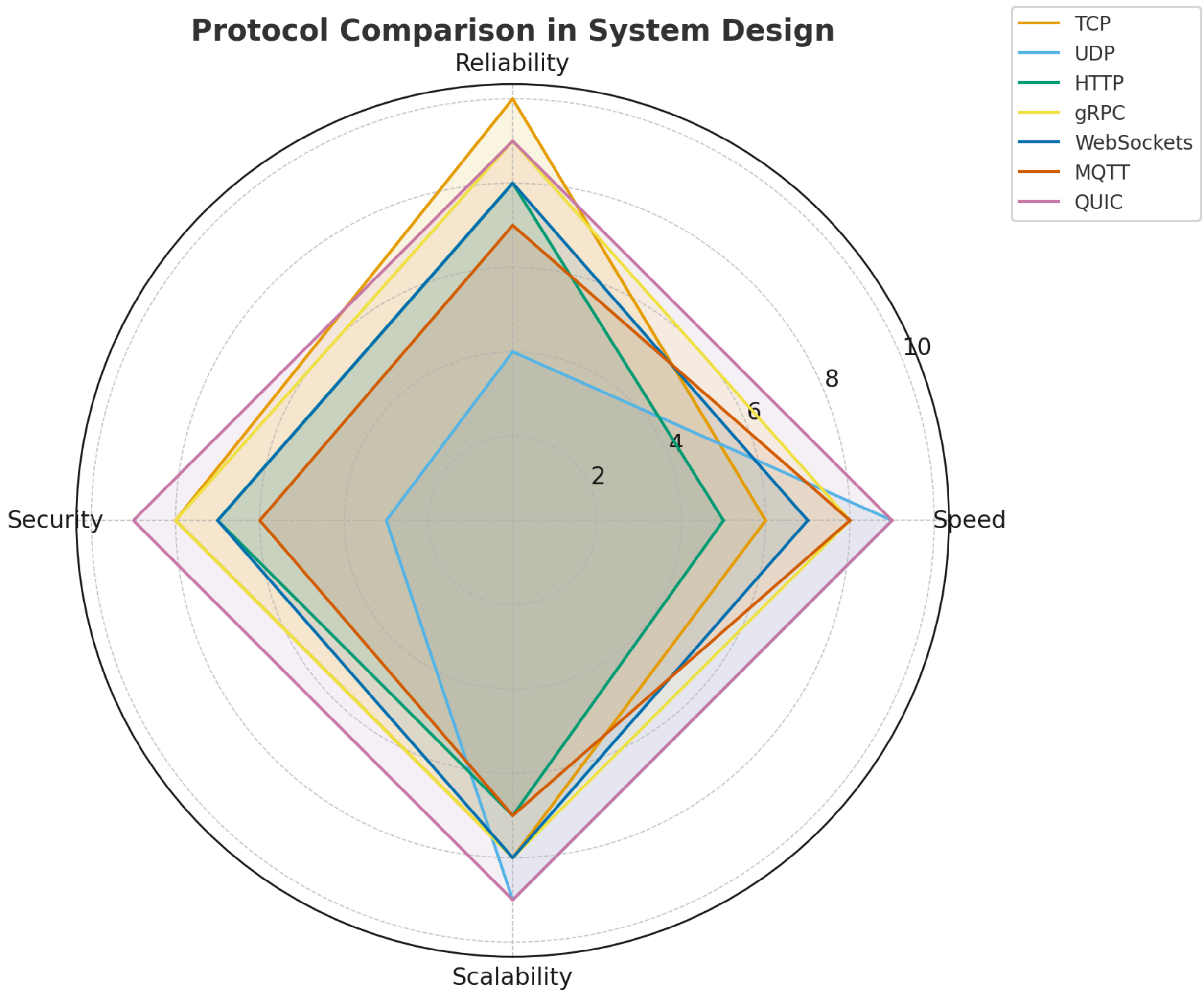

📊 Protocol Comparison

This diagram compared **seven major protocols**. Here’s the detailed breakdown with full forms:

🔹 1. TCP – Transmission Control Protocol

- **Reliability = 10** → Guarantees delivery, retransmission, ordering.

- **Security = 8** → Often paired with TLS (e.g., HTTPS, SSH).
- **Speed = 6** → Slower due to 3-way handshake + ACK overhead.
- **Scalability = 8** → Works well, but many open connections consume resources.
  ✅ Best for: APIs, databases, file transfers.

🔹 2. UDP – User Datagram Protocol

- **Speed = 9** → Very fast (no handshake).
- **Reliability = 4** → No guarantee of delivery/order.
- **Security = 3** → Weak by default, needs extra encryption like DTLS.
- **Scalability = 9** → Excellent for real-time broadcast systems.
  ✅ Best for: Video streaming, gaming, VoIP (Voice over IP).

🔹 3. HTTP – HyperText Transfer Protocol

- **Reliability = 8** → Built on TCP, so reliable.
- **Security = 7** → Secure with **HTTPS (HTTP Secure)**.
- **Speed = 5** → Verbose (text-based, lots of headers).
- **Scalability = 7** → Works well but less efficient for high-volume services.
  ✅ Best for: REST APIs, web applications.

🔹 4. gRPC – Google Remote Procedure Call

- **Reliability = 9** → Built on HTTP/2 (which uses TCP).
- **Security = 8** → Supports TLS encryption.
- **Speed = 8** → Uses binary encoding (**Protobuf – Protocol Buffers**).
- **Scalability = 8** → Efficient for microservices.
  ✅ Best for: Service-to-service communication in microservices.

🔹 5. WebSockets

- **Reliability = 8** → Built on TCP.
- **Security = 7** → Secure with **WSS (WebSocket Secure)**.
- **Speed = 7** → Persistent connection, faster than HTTP polling.
- **Scalability = 8** → Good for real-time, but server needs to manage many open connections.
  ✅ Best for: Chat apps, stock tickers, collaborative tools, dashboards.

🔹 6. MQTT – Message Queuing Telemetry Transport

- **Speed = 8** → Lightweight, minimal overhead.

- **Reliability = 7** → Has QoS (Quality of Service) levels to control delivery guarantees.
- **Security = 6** → Can use TLS but often less secured in practice for IoT devices.
- **Scalability = 7** → Designed for IoT, not massive-scale web systems.
  ✅ Best for: IoT devices, sensors, mobile push notifications.

🔹 7. QUIC – Quick UDP Internet Connections

- **Speed = 9** → Fast connection setup, no head-of-line blocking.
- **Reliability = 9** → Ensures delivery like TCP, but with UDP's efficiency.
- **Security = 9** → Has **TLS 1.3** built-in by default.
- **Scalability = 9** → Optimized for modern distributed systems.
  ✅ Best for: HTTP/3, mobile networks, CDNs (Content Delivery Networks).

📌 Key Takeaways with Full Forms

- **TCP (Transmission Control Protocol)** = Reliable but slower.
- **UDP (User Datagram Protocol)** = Very fast, but unreliable.
- **HTTP (HyperText Transfer Protocol)** = Universal for web apps, but heavy.
- **gRPC (Google Remote Procedure Call)** = Fast & efficient for microservices.

- **WebSockets** = Best for real-time two-way communication.
- **MQTT (Message Queuing Telemetry Transport)** = Lightweight protocol for IoT.
- **QUIC (Quick UDP Internet Connections)** = Next-gen protocol (basis for HTTP/3).