

1. Allgemeines
2. TCP
3. UDP
4. Vergleich

- TCP und UDP setzen auf IP auf (TCP/IP)
- Es sind Transportprotokolle
- Auf Schicht 4 (Transportschicht) des OSI-Referenzmodells angesiedelt
- Teilen Daten in Pakete auf
- Beide verwenden sog. Ports als Endpunkte
- Das Betriebssystem oder die zuständigen Bibliotheken kümmern sich um die Umsetzung (Linux-Kernel, Winsock)

## 1. Allgemein

1.1. Generelles

1.2. Merkmale von TCP

## 2. Genaue Funktionsweise

2.1. Zuverlässige Datenübertragung

2.2. Drei-Wege-Handshake

2.3. Flusssteuerung

2.4. Aufbau eines TCP-Paketes

2.5. Beispiel einer Datenübertragung

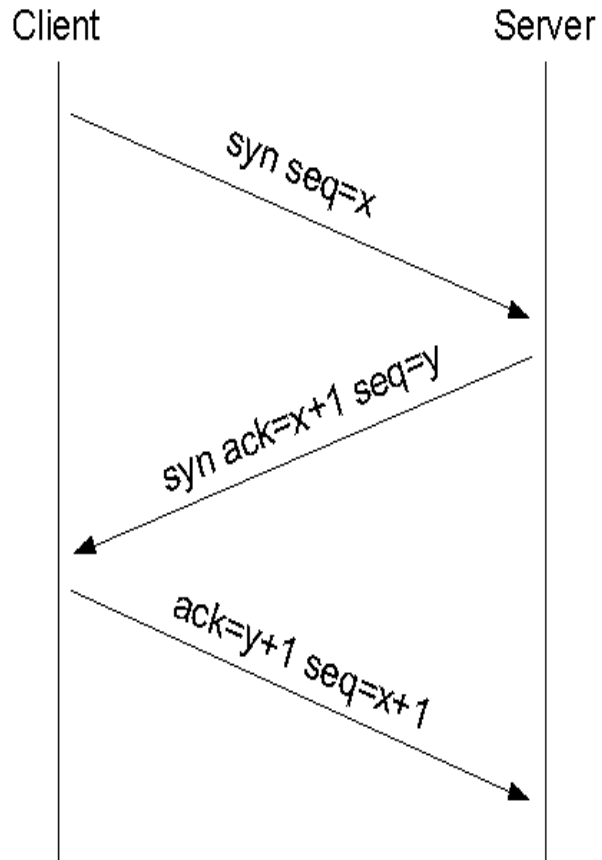
- TCP = **T**ransmission **C**ontrol **P**rotocol
- IP-Protokollnummer: 06
- Entwicklung 1973 begonnen
- Erste Standardisierung 1981 in RFC 793
- Wird für komplexe Protokolle verwendet (z.B. FTP, HTTP, SMTP)

- Verbindungsorientiertes Protokoll  
→ Datenpakete werden in der richtigen Reihenfolge weitergegeben
- Es wird garantiert, dass alle Datenpakete ankommen
- Pakete können gleichzeitig in beide Richtungen versendet werden (Vollduplex)
- Flusssteuerung

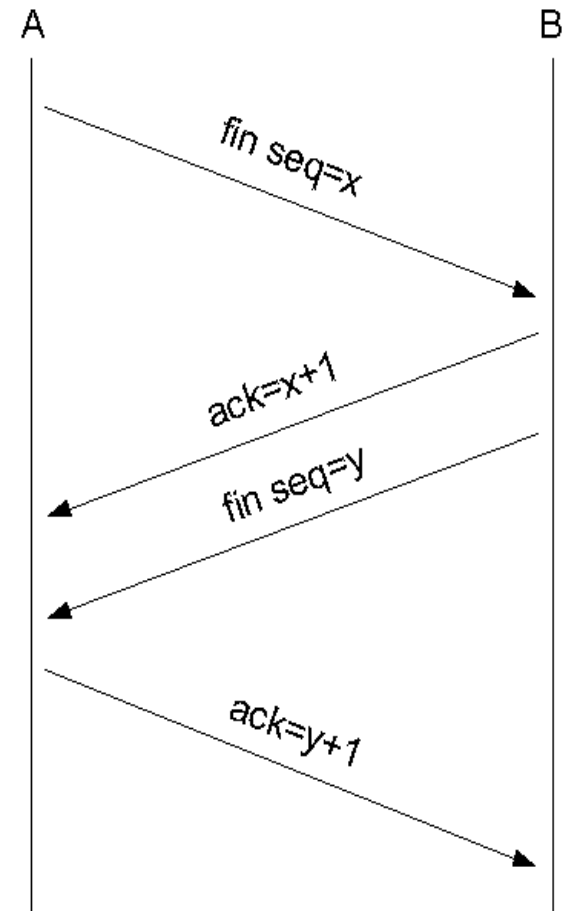
- TCP nummeriert die Datenpakete, so dass sie in der korrekten Reihenfolge zusammengesetzt werden können („Sequenznummer“)
- Der Empfänger muss jedes Paket bestätigen, indem er das nächste Paket anfordert (ACK)
- Erhält der Sender keine Bestätigung, sendet er das Paket erneut

- Zweck: Einleiten einer zuverlässigen Verbindung
- Grundlegender Ablauf:
  1. Client sendet Paket (SYN)
  2. Server bestätigt, dass er das Paket erhalten hat (SYN ACK)
  3. Client bestätigt, dass er das Bestätigungspaket des Servers erhalten hat
- Etwas Ähnliches wird auch beim Abbau der Verbindung verwendet

# Drei-Wege-Handshake (II)



Verbindungsaufbau

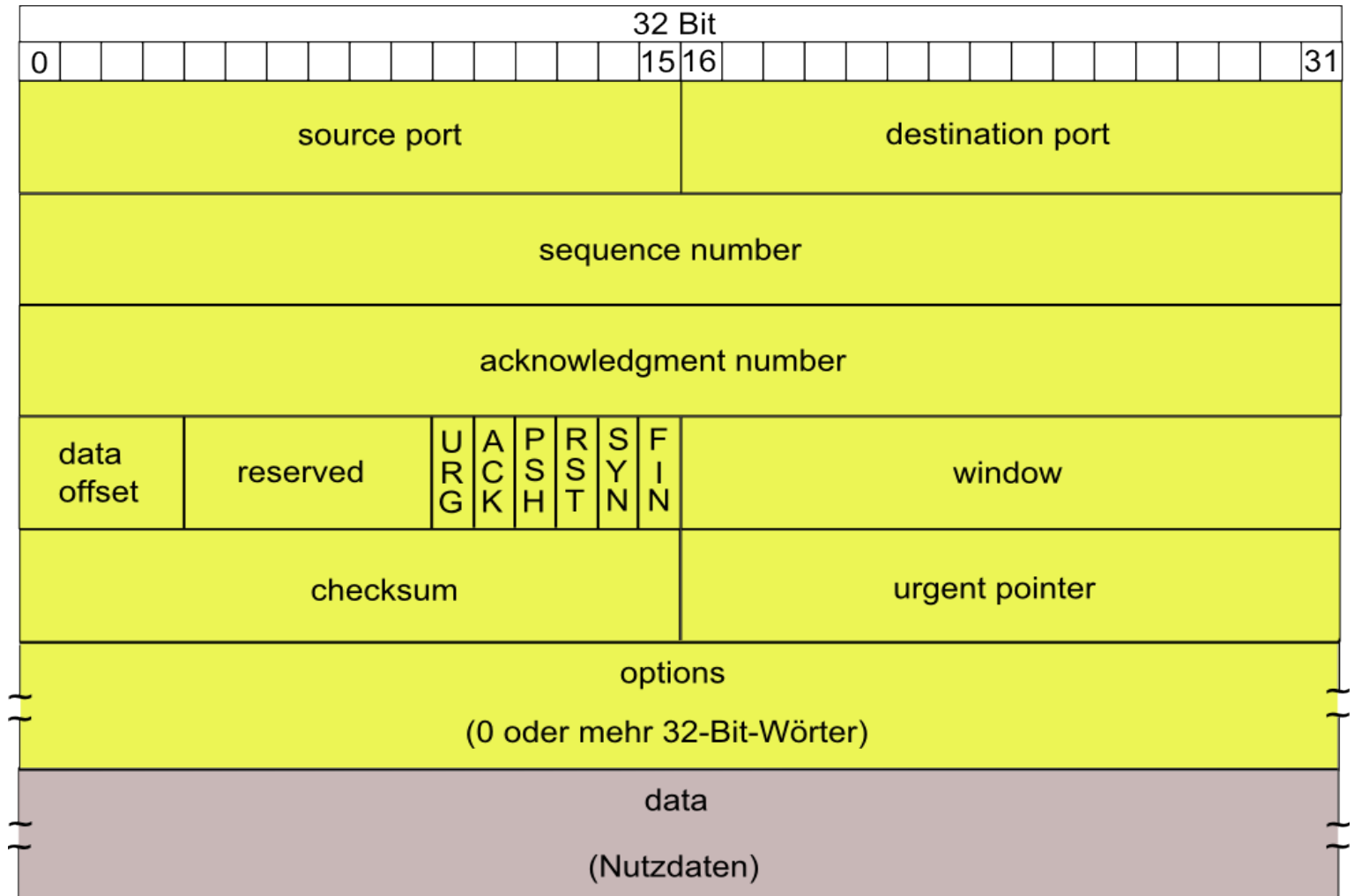


Verbindungsabbau

- Verhindert, dass ein langsamer Empfänger von einem schnellen Sender mehr Pakete bekommt, als er verarbeiten kann
- Szenario ohne Flusssteuerung: Pakete gehen verloren und müssen neu gesendet werden  
→ Zusätzliche Belastung
- Lösung: Empfänger teilt dem Sender mit, wie viele Pakete er noch empfangen kann

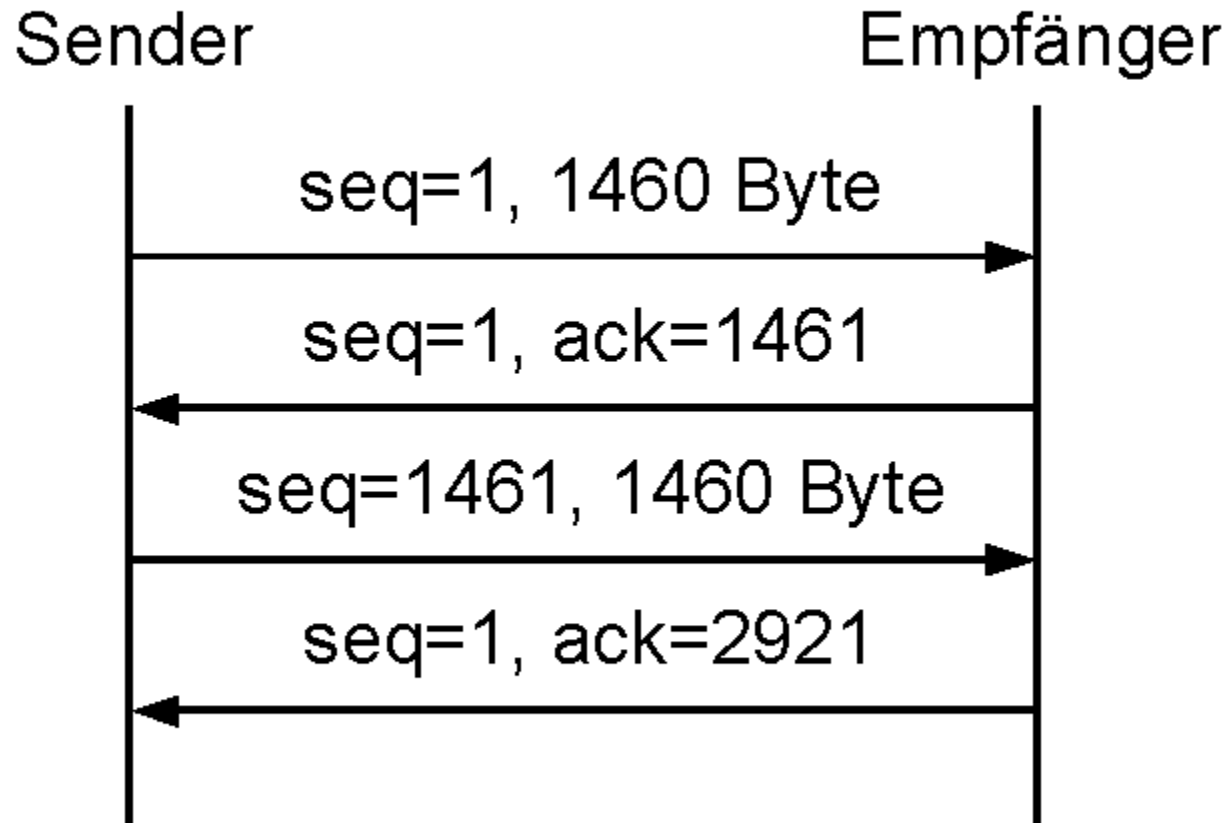
- Ein TCP-Paket („Segment“) besteht aus zwei Teilen:
  - **Header** (20 bis 60 Byte)
  - **Nutzdaten**
- Pakete müssen in die darunterliegenden Schichten passen

# Aufbau eines TCP-Paketes (II)



TCP-Header

# Beispiel einer Datenübertragung



## 1. Allgemein

1.1. Generelles

1.2. Merkmale von UDP

## 2. Genaue Funktionsweise

2.1. Aufbau eines UDP-Paketes

- UDP = **U**ser **D**atagram **P**rotocol
- IP-Protokollnummer: 17
- Entwicklung 1977 begonnen
- Standardisiert in RFC 768
- Wird zumeist für einfache Frage-Antwort-Protokolle verwendet (z.B. DNS, DHCP, NTP)

- Verbindungsloses Protokoll
- Es findet keine Kontrolle statt, ob die Pakete korrekt ankommen
- Datenpakete kommen nicht unbedingt in der richtigen Reihenfolge an
- Datenpakete werden direkt an die zuständige Anwendung weitergegeben

- Ein UDP-Paket („Datagramm“) besteht aus zwei Teilen:
  - **Header** (8 Byte, Teile sind optional)
  - **Nutzdaten**
- Pakete müssen in die darunterliegenden Schichten passen



# Vergleich von TCP und UDP

	<b>TCP</b>	<b>UDP</b>
Verbindungsorientiert	ja	nein
Zuverlässig	ja	nein
Duplex	ja	nein
Sortierung der Pakete	ja	nein
Flusssteuerung	ja	nein
Fehlererkennung	ja	optional
Fehlerbehebung	ja	nein
Drei-Wege-Handshake	ja	nein
Headergröße	20-60 Byte	8 Byte
Geschwindigkeit	langsam	schnell
Systembelastung	normal	gering



Ende

*Vielen Dank fürs  
Zuhören!*