Rechnerkommunikation

Felix Leitl

19. September 2024

Inhaltsverzeichnis

Transportschicht

UDP vs. TCP

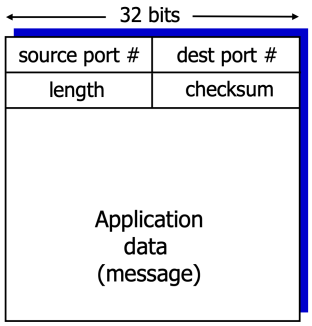
UDP

- verbindungslos, keine Kontrollmechanismen, bewahrt Reihenfolge nicht
- Schnittstelle für einfache Paketvermittlung mittels IP, Verantwortung für Kontrollmechanismen bei Anwendung

TCP

- verbindungsorientiert, Fehler-, Fluss, Überlastkontrolle, keine Gütegarantie
- bietet Abstraktion eines Bytestroms

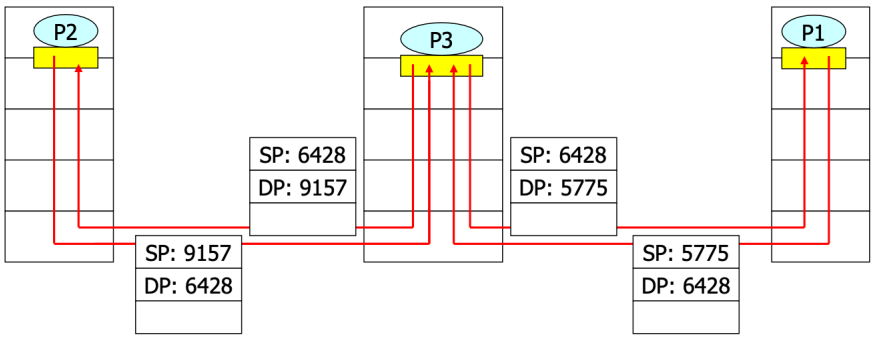
UDP



- source port: 16 Bit
- dest port: 16 Bit
- length: 16 Bit (gestates Segment)
- checksum: 16 Bit (16 $\times 0$ $\Rightarrow$ungenutzt)

Multiplexen und Demultiplexen

- Multiplexen: Zusammenführen der Segemente verschiedener Anwendungsprozesse auf Quellhost
- Demultiplexen: Ausliefern der Segmente an verschiedene Prozesse des Zielhosts



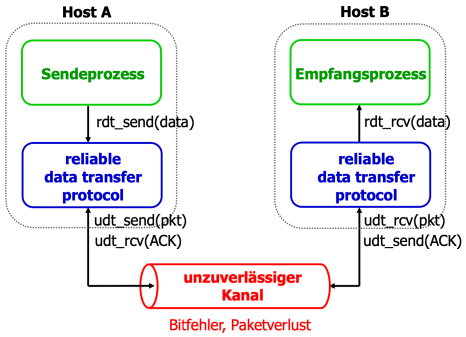
Prüfsumme

- Segment wird als Folge von Dualzahlen der Länge 16 Bit aufgefasst
- diese werden in Einerkomplementarithmetik addiert
- Das Ergebnis wird invertiert und zur Prüfsumme
- Empfänger berechnet auch Prüfsumme und addiert diese auf die übergebene $\rightarrow 1111111111111111_2$ $\Rightarrow$kein Fehler liegt vor
- Nur einzelne Fehler werden erkannt

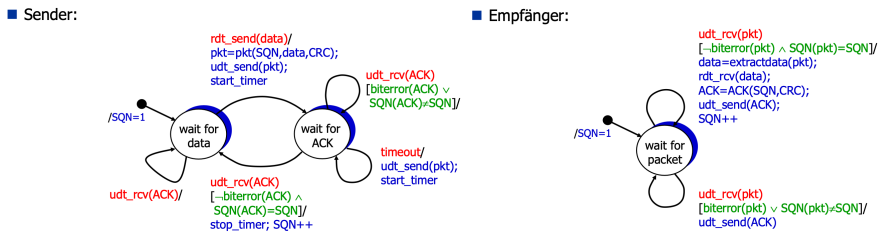
Pseudo-header

- Pseudo-header enthält Quell- und Ziel-IP-Adresse, Protokollnummer (17 UDP) und Segmentlänge
- UDP des Senders schreibt zuerst 0 ins Checksum-Feld und berechnet dann die Prüfsumme über Segment und Pseudo-header
- Vorteil: Es werden fehlgeleitete Pakete erkannt
- Nachteil: Verletzung des Schichtenprinzips

Fehlerkontrolle



Stop-and-Wait

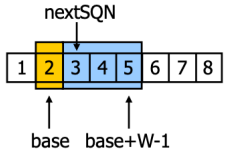


Bei Stop-and-Wait reichen die Sqeuenznummern 0 und 1 $\rightarrow$Alternating -Bit-Protokoll

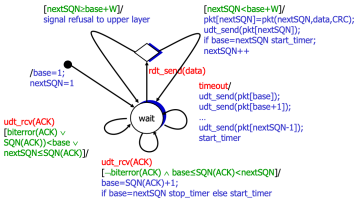
Go-Back-N

Um die Ineffizienz von Stop-and-Wait zu vermeiden, senden Schiebefensterprotokolle mehrere Pakete, bevor die Bestätigung zurückkommt

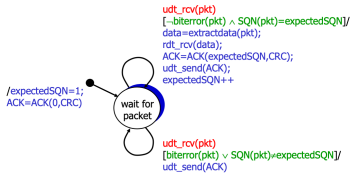
- base: SQN des ältesten unbestätigten Paktes
- nextSQN: SQN des nächsten zu verschickenden Pakets
- W: Fenstergröße, Anzahl der Pakete, die der Sender vor Erhalt eines ACKs senden darf



Go-Back-N: Sender

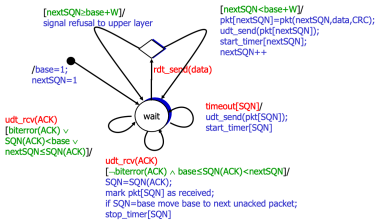


Go-Back-N: Empfänger

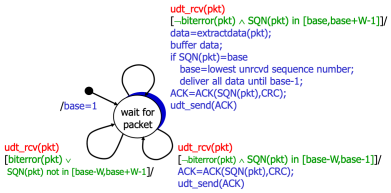


Selective Repeat

Selective Repeat: Sender



Selective Repeat: Empfänger



Sequenznummernraum

Hinreichende Bedingungen: (m Werte, W Fenstergröße)

- Empfängerfenstergröße = 1: W < m
- Sendefenstergröße = Empfangsfenstergröße = W > 1 : W < (m+1)/2

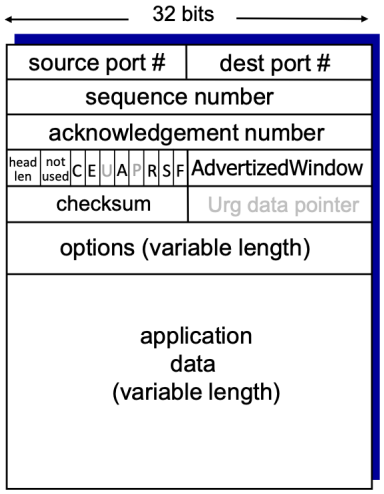
TCP

- Punkt-zu-Punkt
- reihenfolgebewahrender Bytestrom
- fensterbasierte Fehlerkontrolle
- vollduplex: 2 entgegengesetzte Datenströme

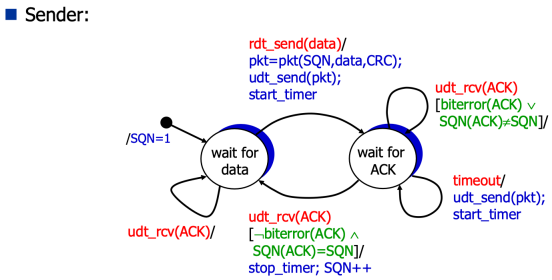
- verbindungsorientiert
- Flusskontrolle
- Überlastkontrolle

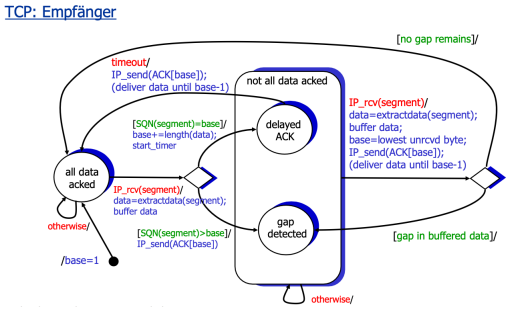
- seq: Nummer des ersten Bytes des Segments im Bytestrom
- ack: Nummer des nächsten erwarteten Bytes im Bytestrom
- Flags:
  - CWR (Congestion Window Reduced)
  - ECE (ECN-Echo)
  - ACK (ACK gültig)
  - RST (reset connection)
  - SYN (synchronisiere Verbindung)
  - FIN (beende Verbindung)
- AdvertizedWindow: Fenstergröße für Flusssteuerung

- checksum: wie bei UDP



Fehlerkontrolle





Verbindungsaufbau

3-Wege-Handshake:

1. SYN-Segment: Client sendet Segment mit SYN-Flag=1, zufälliger initialer Client-SQN (client_isn), ohne Daten
2. Server sendet Segment mit SYN-Flag=Ack-Fag=1, zufälliger initialer Srever-SQN (server_isn), ACK=client_isn+1, ohne Daten; er legt Puffer und Variablen an

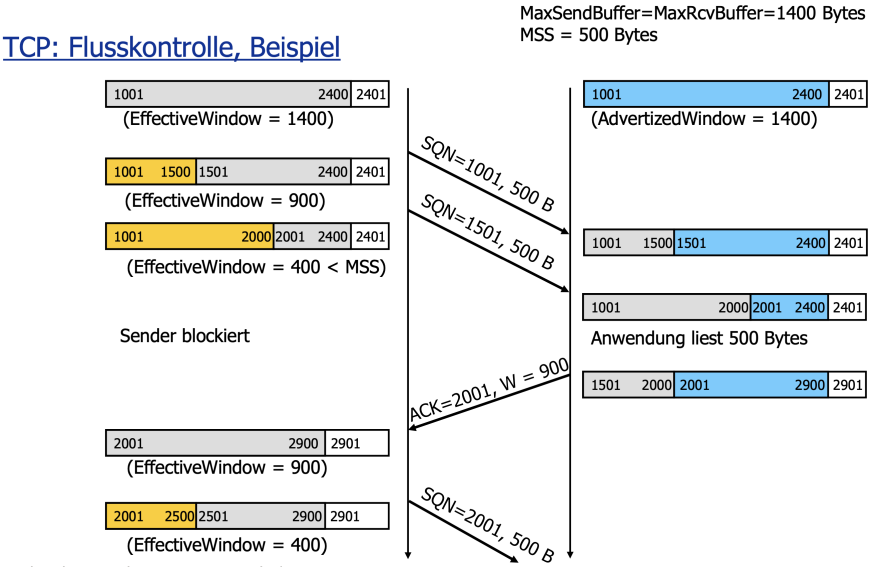
3. ACK-Segment: Client sendet Segment mit ACK-Flag=1; SQN=client_isn+1, ACK=server_isn+1 und ggf. Daten; er elgt Puffer uns Variablen an

Segmente mit SYN-Flag=1 oder FIN-Flag=1 dürfen keine Daten enthalten

Verbindungsabbau

- jede Seite kann Verbindungsabbau durch Segment mit FIN-Flag=1 veranlassen
- die andere Seite bestätigt mit ACK-Flag=1
- beide Seiten müssen ihre Hälfte der Verbindung schließen
- hat eine Seite geschlossen, sendet sie keine Daten mehr, nimmt aber noch welche an
- Time Wait: die Seite, die den Verbindungsabbruch veranlasst, wartet zum Schluss noch 2 Segmentlebensdauern, um alte Segmente zu empfangen

Flusskontrolle



Überlastkontrolle

Slow Start:

1. CongestionWindow = MSS setzten und bis 3 doppelte ACKs zurückkommen verdoppelt
2. CongestionWindow wird anschließend halbiert und wächst linear, bis 3 doppelte ACKs zurückkommen (AIMD [Additive Increase, Multiplicative Decrease])
3. wiederhole AIMD
4. konservative Reaktion nach Timeout: dann wird Slow Start bis zur hälfte des aktuellen CongestionWindows und danach AIMD durchgeführt

Netzwerkschicht

Sicherungsschicht

Physikalische Schicht