

1K1203 #6 2019-01-31

rdt 3.0 som vi konstruerade under tidigare föreläsning presterar inte så bra då det baseras på stop-and-wait.

- Utnyttjandegraden är i storleksordningen hundradelar av procent.
- Genom att skicka flera paket åt gången kan vi öka utnyttjandegraden.
- kräver fler sekvensnummer samt buffrar
- Två tekniker: Go-back-N & Selective Repeat.

### Go-back-N

- kumulativa ACKar
- Vid fel skickas alla efterföljande paket om.
- Fönsterstorlek styr hur många paket som kan skickas åt gången.

### TCP (RFC 793)

- förbindelsehantering
- tillförlitlighet
- flödeskontroll
- stopningskontroll
- förbindelseorienterat protokoll (handskakning)
- förbindelsen är full-duplex: data kan skickas i båda riktningar samtidigt
- En sändare till en mottagare
- tillförlitlig bytestream-service. TCP delar självt upp datan i lagom stora bitar (segment).
- Använder timer och ACK för att hantera återgårdning.
- ACK skickas vid mottaget paket, vid time-out återsänds paket.
- Adaptiv flödes- och stopningskontroll.
- Ett segment innehåller (20 byte) header, options samt data.
- Headern innehåller adress, port, sekvensnummer, ack-nummer, flaggor, fönsterstorlek m.m. (Bild i presentation)

Flaggor: ACK, RST, SYN, FIN

- 3-vägs handskakning
  1. Klienten skickar ett tomt segment med flaggan SYN och ett slumpmässigt sekvensnummer.
  2. Servern svarar med SYN, ACK och slumpmässigt sekvensnummer.
  3. Klienten skickar ACK och första datan.

Connection Teardown (kan initieras av klient eller server)

- båda sidor skickar ett FIN och ett ACK-meddelande.

MSS (Maximum Segment Size) anger största storleken på data-fältet.

- kan anges i options, annars används ett standardvärde.

Sändnings- och mottagningsbuffert används på respektive sida.

Flödeskontroll: styr hur fort data kan skickas

- mottagaren erbjuder initialt en fönsterstorlek
- storleken uppdateras vid varje ACK (dynamiskt)
- sändaren begränsas av användbara fönsterstorleken.

TCP räknar inte segment utan räknar istället antal byte.

Telnet - en bokstav skickas vid varje tangenttryckning. Vid varje ACK skickas samma bokstav tillbaka.

- smg paket med mycket overhead
- "tinygram" (41 bytes)

Nagle's algoritm

- Bara ett tinygram skickas åt gången, flera byte ackumuleras medan man väntar på att en ACK ska komma tillbaka.

"Silly window syndrome"

Fördröjda ACK-meddelanden: TCP väntar med att skicka en ACK-meddelanden för att kunna skicka färre.

I verkligheten skickas ACK efter vartannat segment.

Kapacitet = Bandwidth-Delay Product

Fönsterstorleken styrs av mottagaren. ACK kan skickas när som helst.

Stockning i nätet påverkar också.

Stockningskontroll: hanterar stockning i nätet.

- Congestion Window (cwnd): Anpassas efter fördröjningar.
- Slow start: Vid start är  $cwnd = MSS$ , men cwnd dubblas vid varje ACK. Efter tröskelvärde ökas cwnd tills stockning uppstår. Vid stockning minskas tröskelvärdet och cwnd sätts till MSS.
- Resulterar i: oscillerande flöde

- Fast Retransmit and Recovery
- förbättring med mindre "hopp"

(RTT)

Retransmission timeout-tiden beräknas dynamiskt efter fördröjningstiden

- fördubblas vid timeout.