Informatik Bachelor, LV „Rechnernetze“, SS 2025

Dozent: Prof. Dr. Michael Massoth **Datum: 18.06.2025**	**Übung-12**
Vorname: David **Nachname: Schirrmeister**	**TCP-Fallstudie:** **TCP-TimeOut**

Aufgabe 1: (6 P = 3+3 P) TCP-Fallstudie: Klassischer TimeOut

TCP-Sender erhält einen Timeout

Annahmen:
- Kein Fast Retransmit / Fast Recovery
- Keine Duplicate ACKs
- Ein TCP-Segment bleibt **unbestätigt**, und der **Retransmission Timer** läuft ab → **Timeout wird ausgelöst**

Frage: Was passiert alles? (Alle Variable und Windows)

==========

1. Retransmission
TCP-Segment wird neu gesendet, da das ACK ausbleibt und RTO abläuft

Betroffene Variablen: Retransmission Counter (implizit hochgezählt)

---

2. Congestion Window (cwnd)
Wird auf Initialwert zurückgesetzt
Betroffene Variablen:
- cwnd := 1 MSS

---

3. ssthresh (Slow Start Threshold)
- ssthresh := max(cwnd_alt/2 ; 2 MSS)
- Gibt an, **wann TCP künftig von Slow Start zu Congestion Avoidance wechselt**

Betroffene Variablen:
- ssthresh wird neu gesetzt → max(cwnd_alt/2 ; 2 MSS)

---

4. Retransmission Timer
Wird oft verdoppelt, wenn kein ACK kommt
Betroffene Variablen: RTO := RTO*2 **,** RTT möglicherweise nicht aktualisiert

---

5. swnd Sendefenster

Das Sendefenster wird neu berechnet: swnd = min(cwnd, rwnd)

Betroffene Variablen:
swnd := min(1 MSS, rwnd)

---

6. Andere Variablen (nicht direkt verändert, aber betroffen):

Variable	Auswirkung ?
LastByteAcked	Bleibt unverändert (kein neues ACK)
LastByteSent	Zeigt weiterhin auf gesendetes, nicht bestätigtes Segment
LastByteWritten	Bleibt gleich (schon geschrieben, nicht gesendet)
rwnd	Bleibt gleich (Empfänger-gesteuert)

Zusammenfassung der Zustandsänderungen

Komponente	Vor dem Timeout	Nach dem Timeout
cwnd	z. B. 16 MSS	1 MSS
ssthresh	z. B. 16 MSS	8 MSS
swnd	z. B. min(16 MSS, rwnd)	Min (1 MSS, rwnd)
RTO	Standardwert	Ggf. verdoppelt
LastByteAcked	X	X
LastByteSent	X+1	X+1
Verhalten danach	**Congestion Avoidance**	**Slow Start beginnt von vorn**