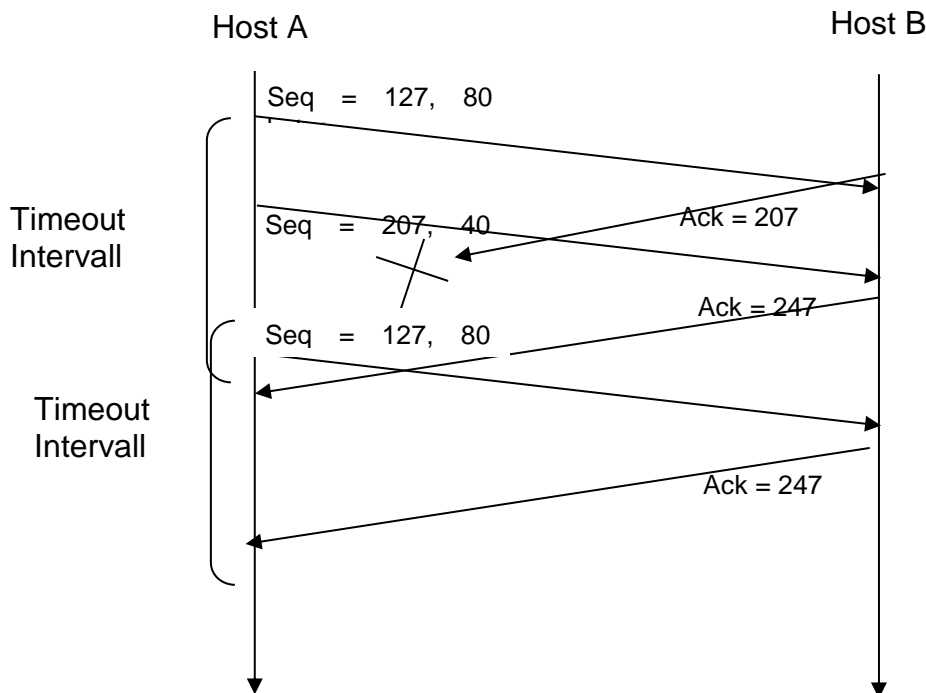


Lösung 11: TCP

Aufgabe 1: TCP – Unidirektionale Kommunikation

- Seq = 207, Src = 302, Dst = 80
- Src = 80, Dst = 302, Ack = 207 (nächstes erwartetes Byte)
- Ack 127 (kumulative ACKs; als nächstes wird Byte Nummer 127 erwartet).
- Siehe Skizze!

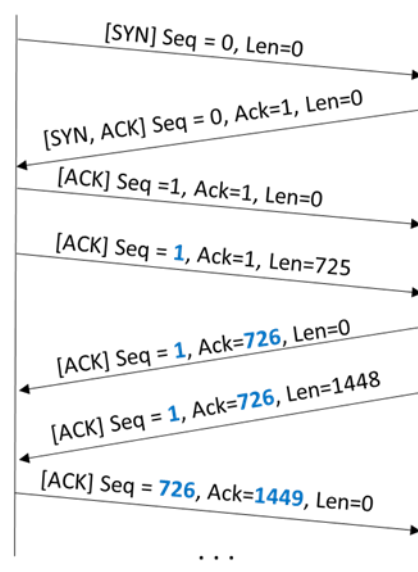
Hinweis: TCP ist angelehnt an das Go-Back-N Prinzip. Im Gegensatz zum reinen Go-Back-N Ansatz wird bei einem Timeout aber nur das älteste noch unbestätigte Paket erneut gesendet.



Aufgabe 2: TCP – Bidirektionale Kommunikation

Hinweise:

- Beim Handshake wird für jede Richtung eine Sequenznummer verbraucht, auch wenn keine Nutzdaten fließen. Beispiel: Die 3. Nachricht trägt die Sequenznummer 1.
- Sequenznummern in einer Richtung „entsprechen“ den ACK-Nummern in der Gegenrichtung. Beispiel: ACK=726 in der vorletzten Nachricht und Seq = 726 in der letzten Nachricht.
- Die Sequenznummer einer Nachricht ergibt sich aus der Sequenznummer der Vorgängernachricht + Länge der Vorgängernachricht. Beispiel: Die Seq=726 der letzten Nachricht ergeben sich aus der Seq=1 der 4. Nachricht + deren Länge 725 (Len = 725).



Aufgabe 3: TCP in Wireshark

a)

	TCP/HTTP Client	TCP/HTTP Server
IP Adresse	192.168.1.102	128.119.245.12
Portnummer	1161	80

b) Siehe Paket #1 im Trace: Seq = 0 (müsste nicht zwingend 0 sein). SYN Flag ist auf 1 gesetzt

c) Siehe Paket #2 im Trace: Seq = 0, Ack = 1, Syn/Ack Flag gesetzt.
Interessant: Der TCP Server bestätigt mit ACK=1 anstatt wie erwartet mit ACK=0. Der Grund warum man ACK=0 erwarten würde: Im SYN-Paket werden noch keine Nutzdaten versendet. Für den Verbindungsaufbau gilt jedoch eine Ausnahme.

d) Wireshark markiert das Paket #199 als HTTP Post Paket. Zur Übertragung des HTTP Posts wurden 122 TCP Pakete benötigt → „122 Reassembled Segments“

e) Man erkennt im Wireshark, dass das 1. Segment des HTTP Posts im TCP Paket #4 steht (unter „Reassembled Segments“). Im unteren „Byte“ Feld von Wireshark erkennt man, dass in Paket #4 tatsächlich das „HTTP Post“ steht.

f) Am besten fertig man eine Skizze an!

	Paketnummer in Wireshark	Sequenznummer	Paketnummer des dazugehörigen ACKs
1. Segment	4	1	6
2. Segment	5	566	9
3. Segment	7	2026	12
4. Segment	8	3486	14

g) In der SYNACK Nachricht ist das *Receive Window* 5840 Bytes groß. Es wird dann schnell vergrößert bis auf 62780 Bytes, aber nie verkleinert.
Schön sieht man das in Wireshark, wenn man die Entwicklung des Empfangsfensters plottet: *Statistiken, TCP Stream Graph, Window Skalierung*.

h) Jede Segmentnummer kommt nur einmal vor (mit Ausnahme des Verbindungsabbaus). Die Verbindung war also stabil ohne Retransmissions.