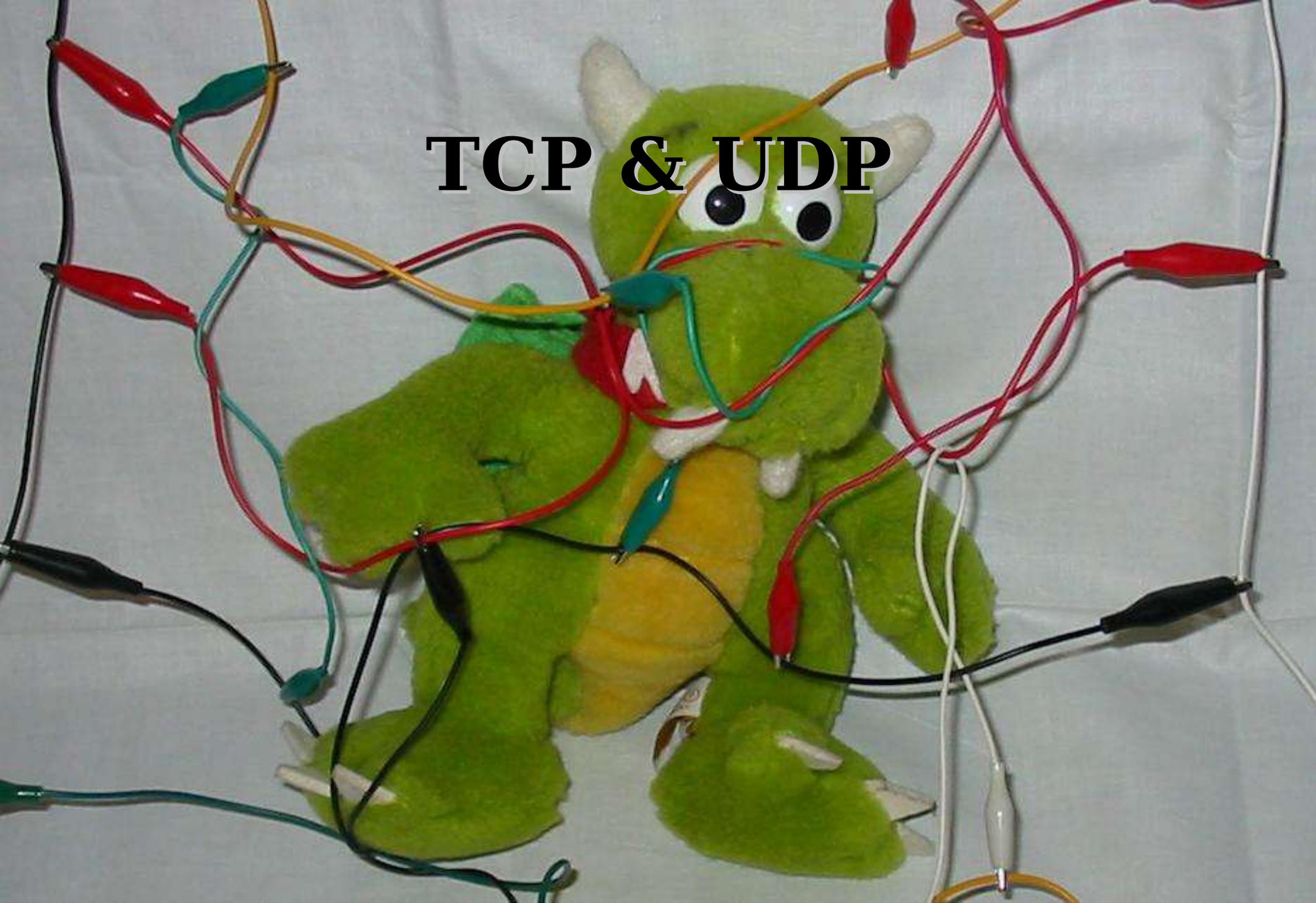
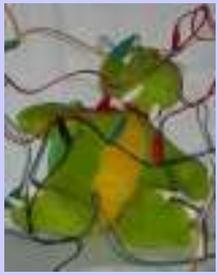


TCP & UDP

A green stuffed dinosaur toy with a yellow belly is the central focus. It is surrounded by a complex network of colorful wires (red, yellow, blue, black, white) and alligator clips (red, black, blue, white) connected to various points on the toy and extending outwards. The background is a plain, light-colored surface.



UDP

- Einfaches Paket-Sende-Protokoll
- Identifiziert Sender-/Empfänger-Programm via 16bit Portnummer
- Bietet keine Zuverlässigkeit



UDP Paketaufbau

Ethernet Header	IP Header	UDP Header	UDP Data...	Ethernet Trailer
-----------------	-----------	------------	-------------	------------------

16bit source port	16bit destination port
16bit UDP length	16bit UDP checksum
data...	



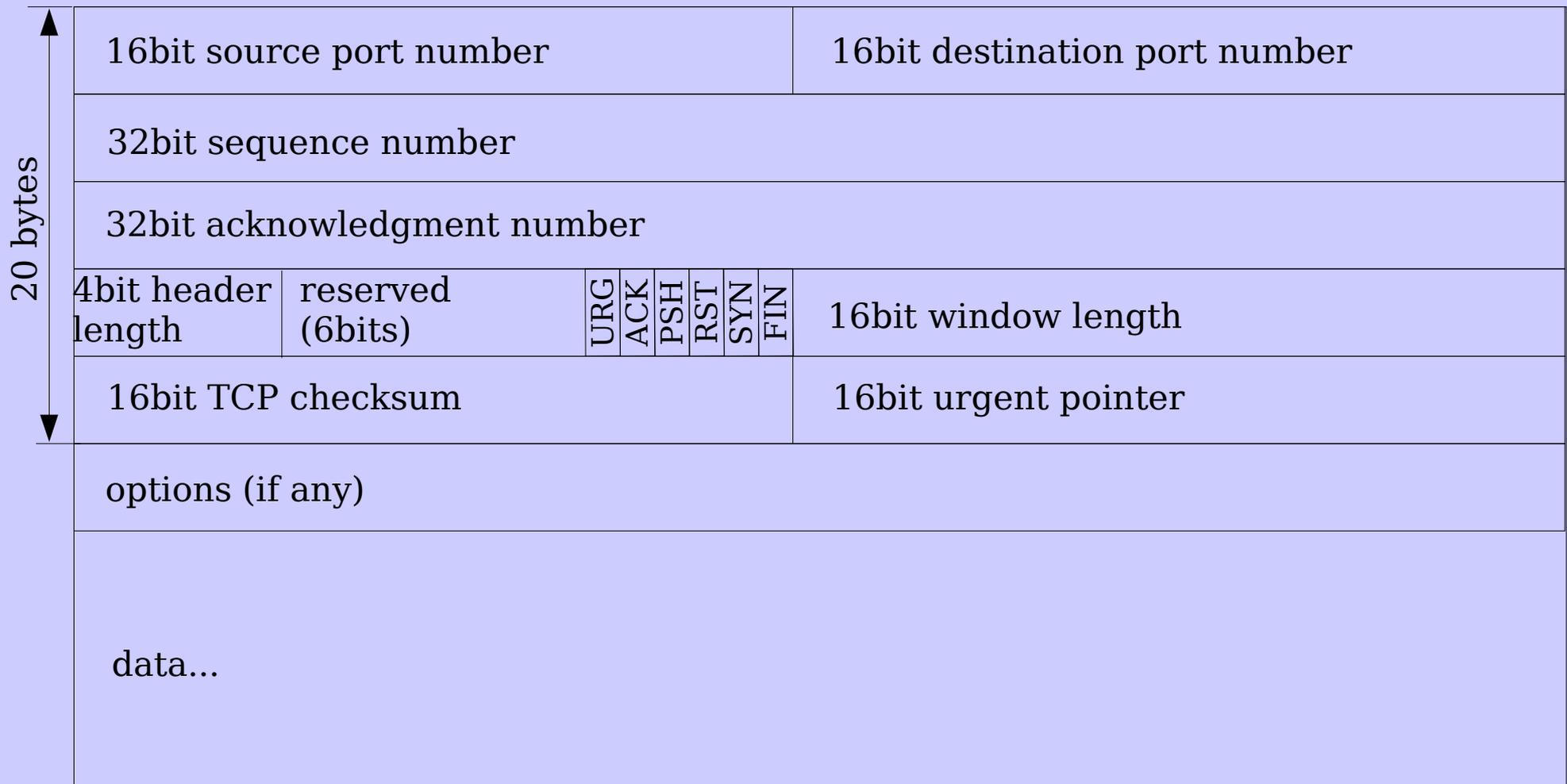
TCP

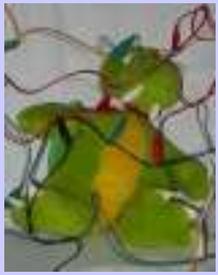
- Datenstrom-Protokoll
- 16bit Ports (wie UDP)
- Simuliert Verbindung über IP
- Automatische transparente Absicherung:
 - verlorene Daten werden wiederholt
 - kaputte Daten werden wiederholt
- Verbindung identifiziert durch:
2x IP und 2x Port



TCP Paketaufbau

Ethernet Header	IP Header	TCP Header	TCP Data...	Ethernet Trailer
-----------------	-----------	------------	-------------	------------------





TCP Header

- Sequence #: Offset des ersten Bytes im Paket relativ zum ersten Byte der Verbindung
 - moderne Implementationen beginnen bei einer zufälligen Zahl, statt bei 0
- Acknowledgment #: letztes erfolgreich empfangenes Byte
- header length: Länge des Headers in 32bit Worten



TCP Header

- reserved: immer Null
- window size: verbleibender Platz im Empfangspuffer
- urgent pointer: Zeiger auf ein einzelnes Byte out-of-band Daten (siehe URG flag)

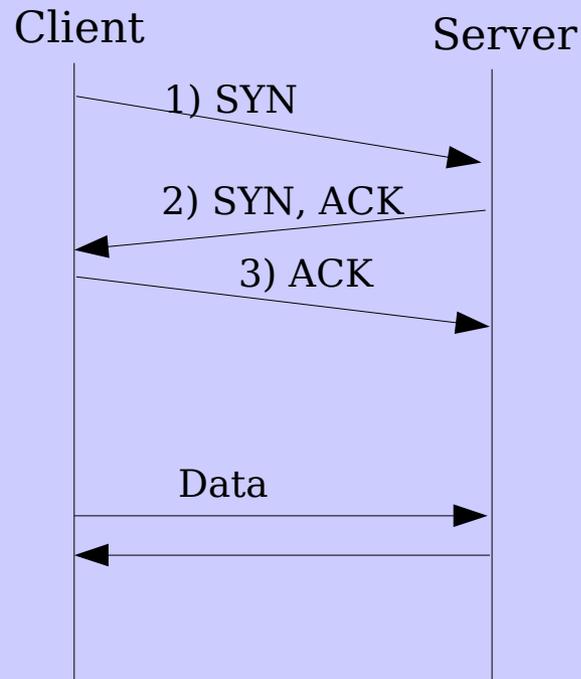


TCP Header

- Flags:
 - URG: out-of-band Daten werden gesendet (max. 1 Byte pro Paket)
 - ACK: Bestätigung empfangener Daten
 - PSH: “push” - Daten so schnell wie möglich an Applikation ausliefern
 - RST: “reset” - Verbindung besteht nicht mehr oder kann nicht aufgebaut werden
 - SYN: Verbindungsaufbau
 - FIN: Verbindungsabbau



3-way handshake: SYN



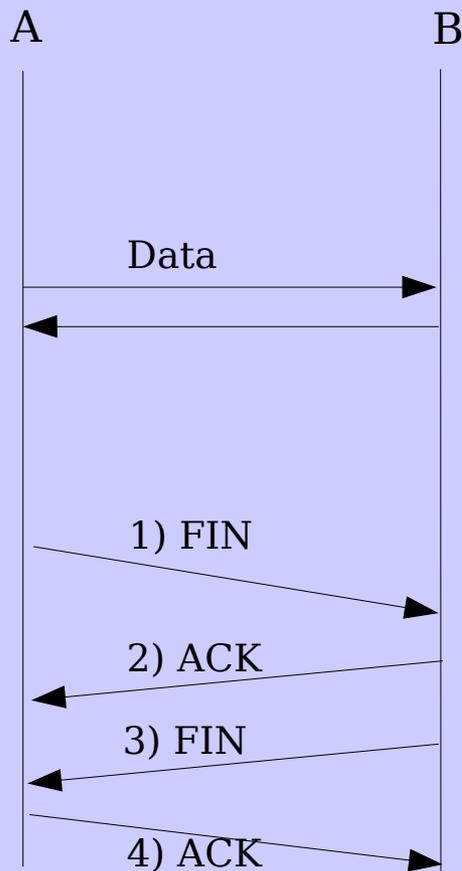
1) Client bittet Server um Verbindung

2) Server bestätigt Verbindung

3) Client bestätigt Verbindung



Verbindungsabbau: FIN



1) A beginnt Verbindungsabbau

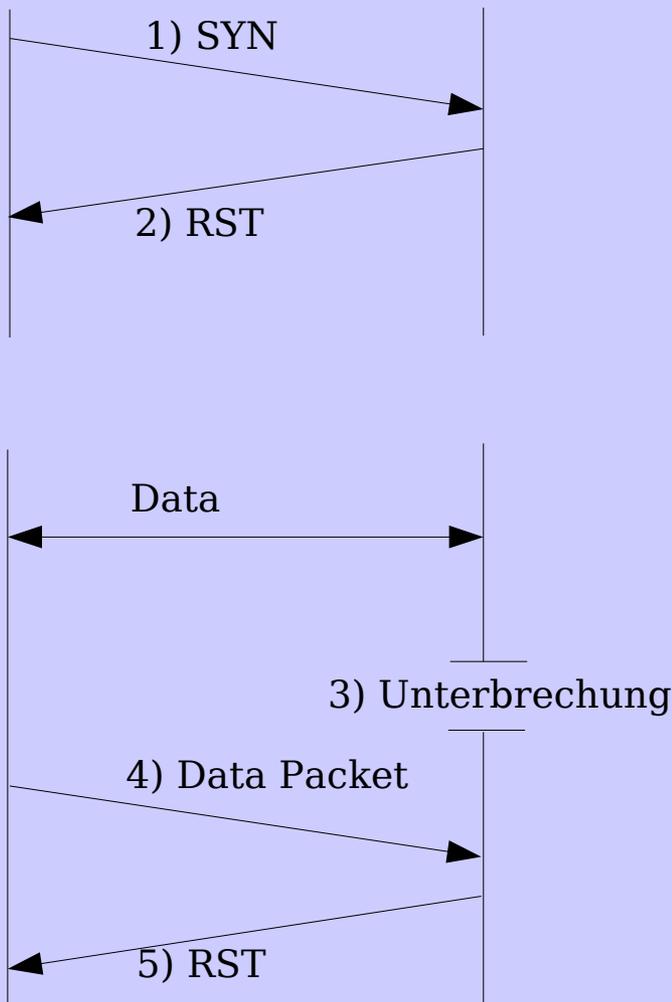
2) B bestätigt
Verbindungsabbau
(A kann nicht mehr senden,
aber noch empfangen)

3) B beginnt Verbindungsabbau

4) A bestätigt
Verbindungsabbau



Reset Paket



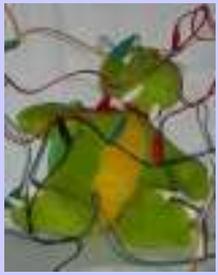
1) Client bittet um Verbindung zu Server

2) Server lehnt ab

3) Reboot, Timeout, etc.pp.

4) normales Datenpaket für obsoletere Verbindung

5) RST signalisiert Verbindungsabbau



Datenstrom

- TCP ordnet Pakete automatisch nach Offset
- Paket wird erst an Applikation abgegeben, wenn alle vorherigen Bytes da sind
- Wenn Bestätigung 30s ausbleibt wird Paket automatisch wiederholt (bis zu 3mal)



Fragen?

?

