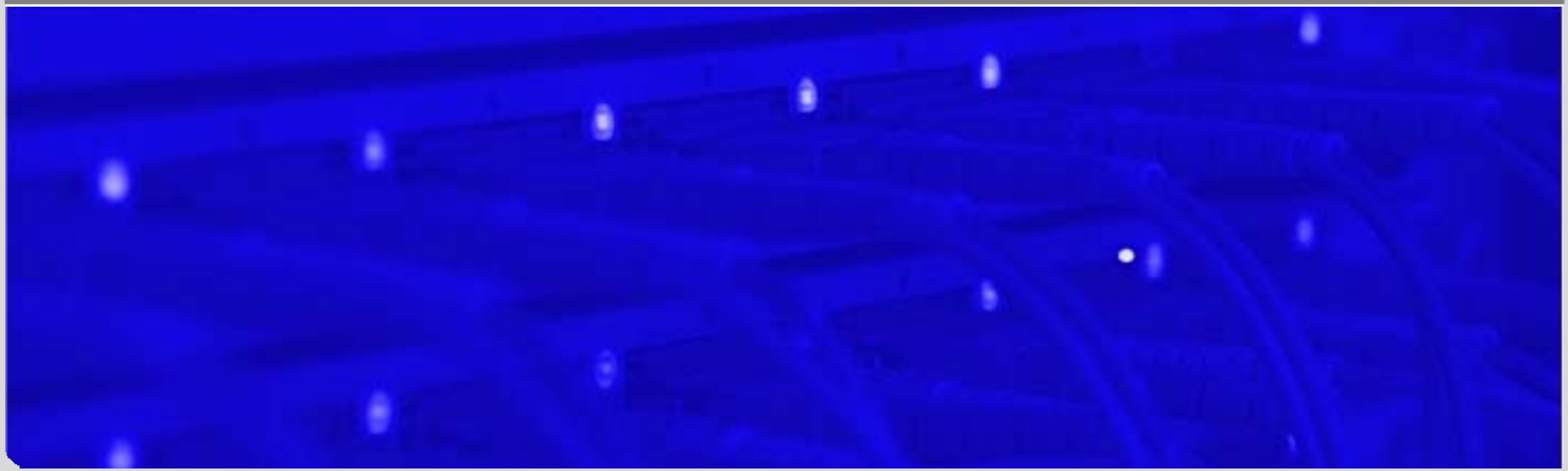
Karlsruher Institut für Technologie

TELEMATICS

Next Generation Internet

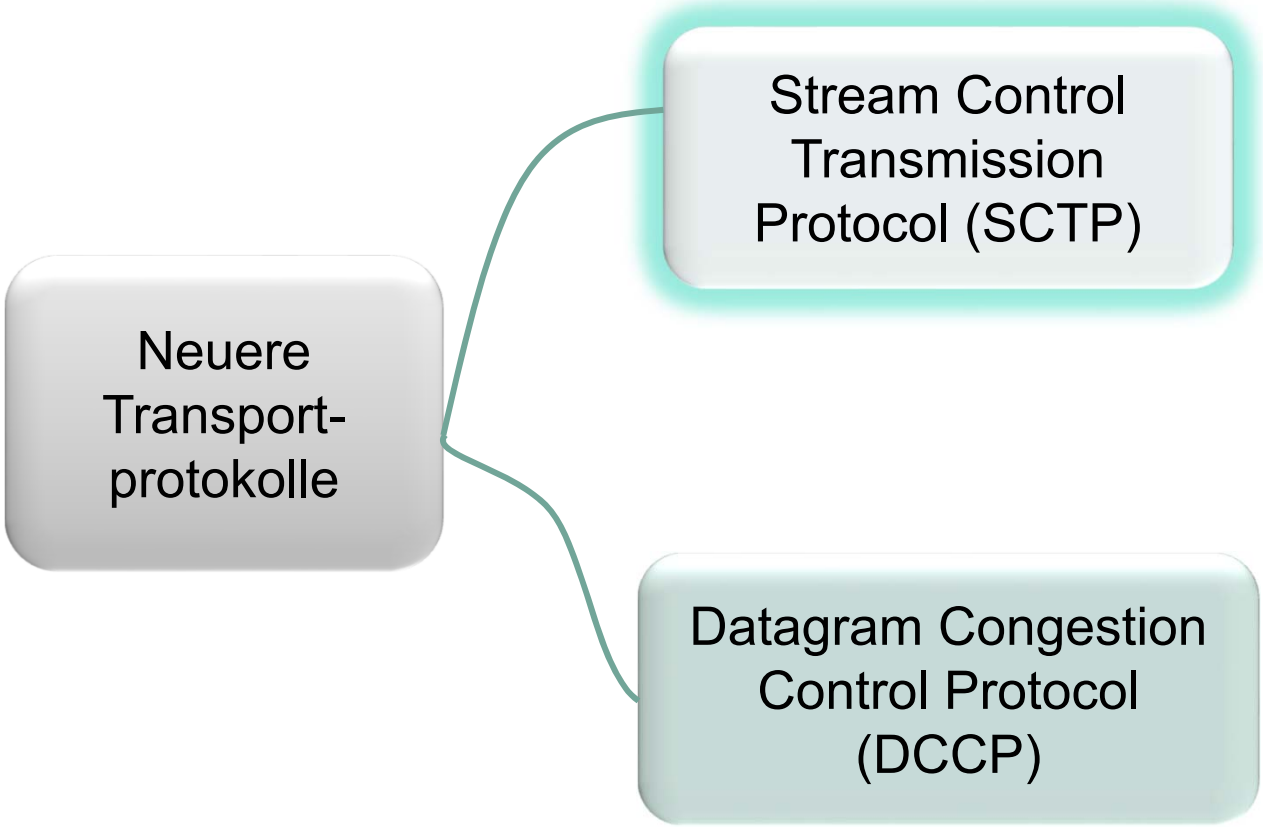
8. Neuere Transportprotokolle

INSTITUT FÜR TELEMATIK



Überblick Kapitel 8

Überblick


Neuere Transport-protokolle
Stream Control Transmission Protocol (SCTP)
Datagram Congestion Control Protocol (DCCP)

SCTP Motivation

- Stream Control Transmission Protocol (SCTP)
- Motivation


[RFC4960]

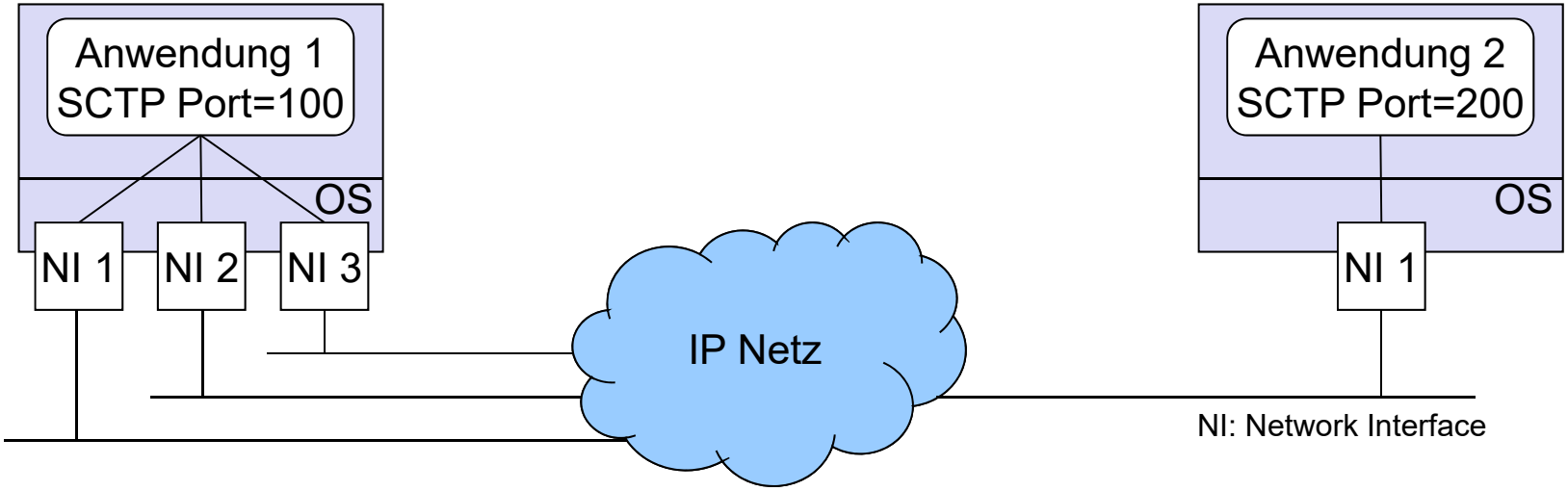
  - Ursprüngliches Ziel: Signalisierung im Telefonnetz (SS7) über IP transportieren
  - Anforderungen an Signalisierung
    - geringe Verzögerung
    - hohe Ausfallsicherheit
  - TCP erfüllt Anforderungen nur unzureichend
    - strikte Reihenfolgeerhaltung nur für Teil der Nachrichten notwendig (Head-of-Line Blocking erhöht Verzögerung)
    - Bytestromorientierung evtl. ungünstig für Nachrichten
    - fehlende Unterstützung für (Host-)Multihoming
    - Anfälligkeit gegen Denial-of-Service (SYN Floods)

SCTP Merkmale

- Eigenschaften
  - Nachrichtenorientiert
  - Erlaubt teilweise Umordnung der Daten
  - Unterstützung für Mehrfachanbindung/Multi-homing
  - Erweiterbarkeit (einfache Integration neuer Chunk-Typen)
- Protokollmechanismen
  - 4-Wege-Verbindungsaufbau zur Verhinderung von Denial-of-Service-Angriffen
  - Segmentieren, Reassemblieren, Bundling
  - Staukontrolle
  - kreditbasierte Flusskontrolle
  - Selektive Quittungen

SCTP Adressierungs-Konzepte

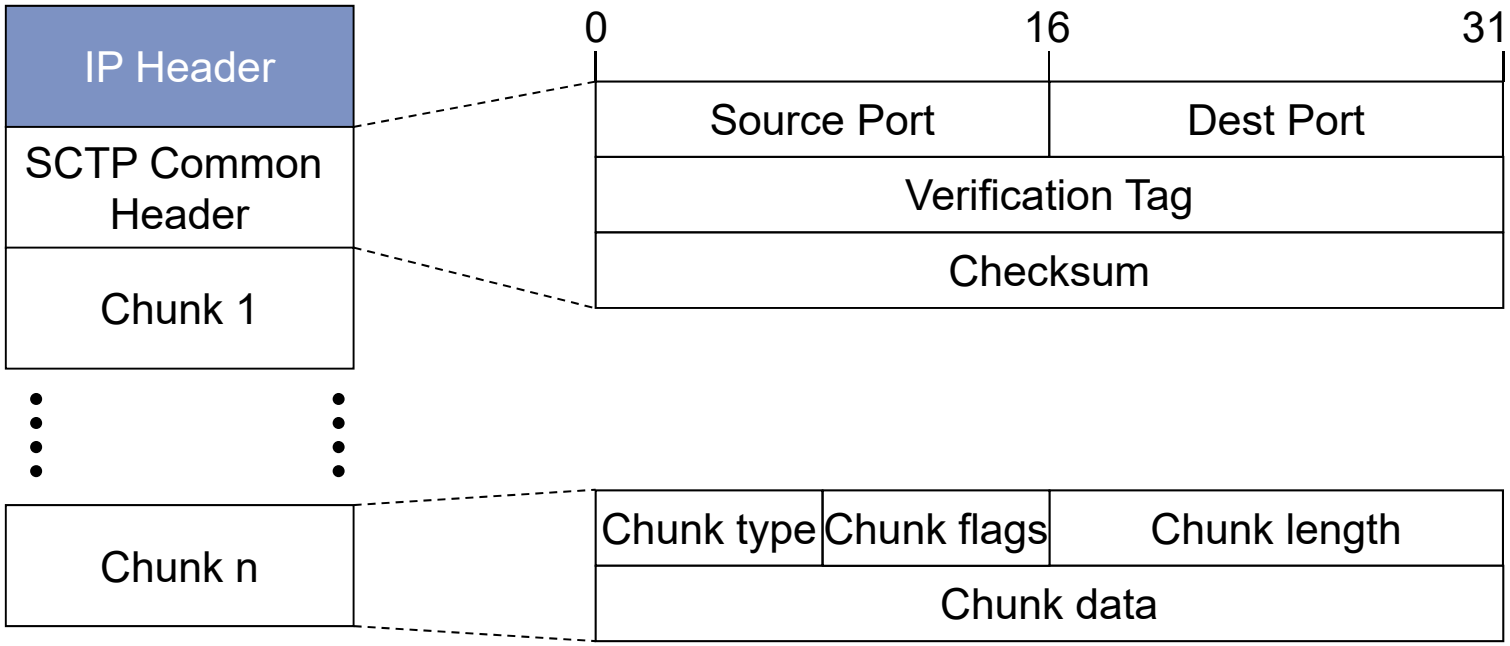
- SCTP Transportadresse= IP-Unicast-Adresse + SCTP Port
- SCTP Endpoint
  - logischer Kommunikationsendpunkt [141.3.72.42:100]
  - Multi-homed SCTP-Endpoints z.B. [141.3.72.42,129.13.10.60:100]
- SCTP Association
  - Kommunikationsbeziehung zwischen zwei Endpunkten, z.B. {[141.3.72.42,129.13.10.60,161.10.8.56:100]:[128.33.5.6:200]}
  - Ergebnis einer erfolgreichen Aufbauprozedur zum Austausch von Zustanddaten zur Etablierung eines Verbindungskontextes



SCTP Streams

- Innerhalb SCTP-Association: Unterscheidung von bis zu 65536 (unidirektionalen) Streams
- Innerhalb eines jeden Streams: Auslieferung der Nachrichten zuverlässig und reihenfolgetreu
- Head-of-Line-blocking innerhalb eines Streams betrifft andere Streams nicht → Innerhalb einer Association also nur teilweise reihenfolgetreu

SCTP Paketformate

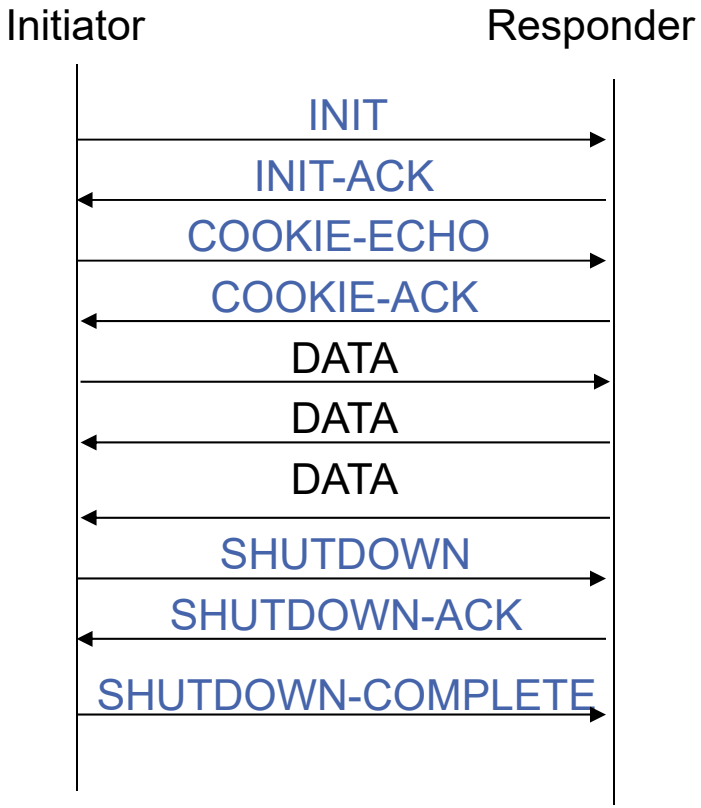


Chunk Typen:

- Daten: DATA,
- Kontrolle: INIT, INIT-ACK, SACK, HEARTBEAT, HEARTBEAT-ACK, ABORT, SHUTDOWN, ERROR, COOKIE-ECHO, COOKIE-ACK, ECNE (ECN-ECHO), CWR (Congestion Window Reduced), SHUTDOWN-COMPLETE
- Control-Chunks stehen immer vor Daten-Chunks

SCTP-Verbindungsaufbau

- Weg-Zeit-Diagramm

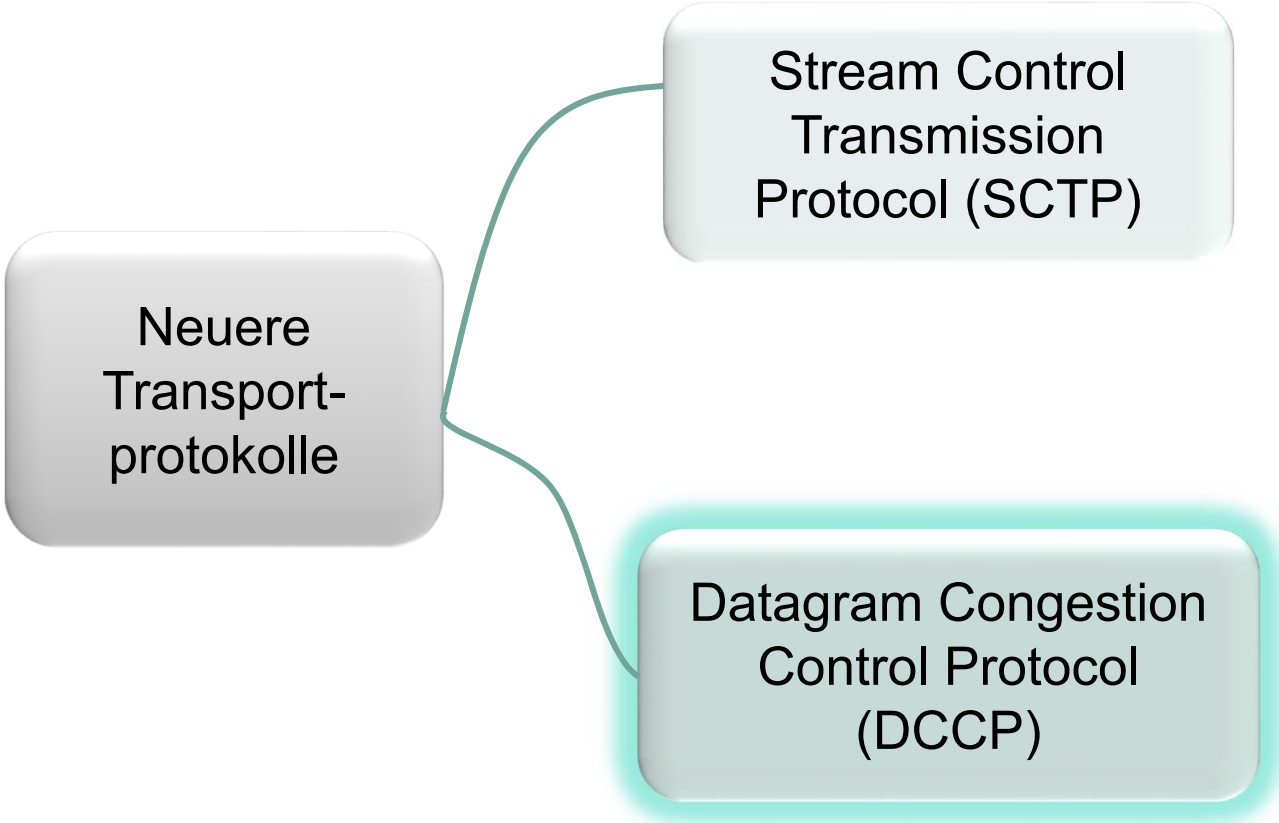


- Init enthält mind. Initiation Tag, Receive Window Credit, Outbound Streams, Max Inbound Streams, Initial TSN, optional: NI-Adressen
- Init-ACK enthält zusätzlich State-Cookie
- Cookie-Echo schickt State-Cookie zurück
- Cookie-ACK bestätigt Empfang des Cookie-Echo Chunks
- Cookie-Echo und Cookie-ACK können zusätzlich Daten-Chunks enthalten

SCTP Ausblick

- NAT Traversal erschwert SCTP-Einsatz
  - UDP-Kapselung als Ausweg?
- PR-SCTP: Partial Reliable Extension  [RFC3758]
  - es kann pro Nachricht festgelegt werden, wie zuverlässig sie übertragen werden soll
  - neuer Chunk-Typ FORWARD TSN, zeigt an, dass Empfänger seinen Quittungsnummer
  - ermöglicht Mischen von zuverlässiger und unzuverlässiger Übertragung innerhalb einer Assoziation
- Authenticated Chunk (AUTH)  [RFC4895]
- Address Change (ADD-IP)  [RFC5061]

Überblick


Neuere Transport-protokolle
Stream Control Transmission Protocol (SCTP)
Datagram Congestion Control Protocol (DCCP)

DCCP Motivation (1)

- Datagram Congestion Control Protocol (DCCP)


[RFC4340]

- Motivation  [RFC4336]
  - Zunahme langlebigerer Streaming-Anwendungen ist zu erwarten
  - Beispielsweise Video-on-Demand, Voice over IP, Online-Spiele, etc.
  - Diese benutzen üblicherweise UDP wegen Anforderung an niedrige Verzögerung
    - Rechtzeitigkeit wichtiger als Zuverlässigkeit
    - TCP-Pakete weisen oft lange Verzögerung aufgrund Übertragungswiederholung und Erhaltung der Reihenfolgetreue auf
  - UDP verfügt aber über keine Staukontrollmechanismen → Staukollaps droht

DCCP Motivation (2)

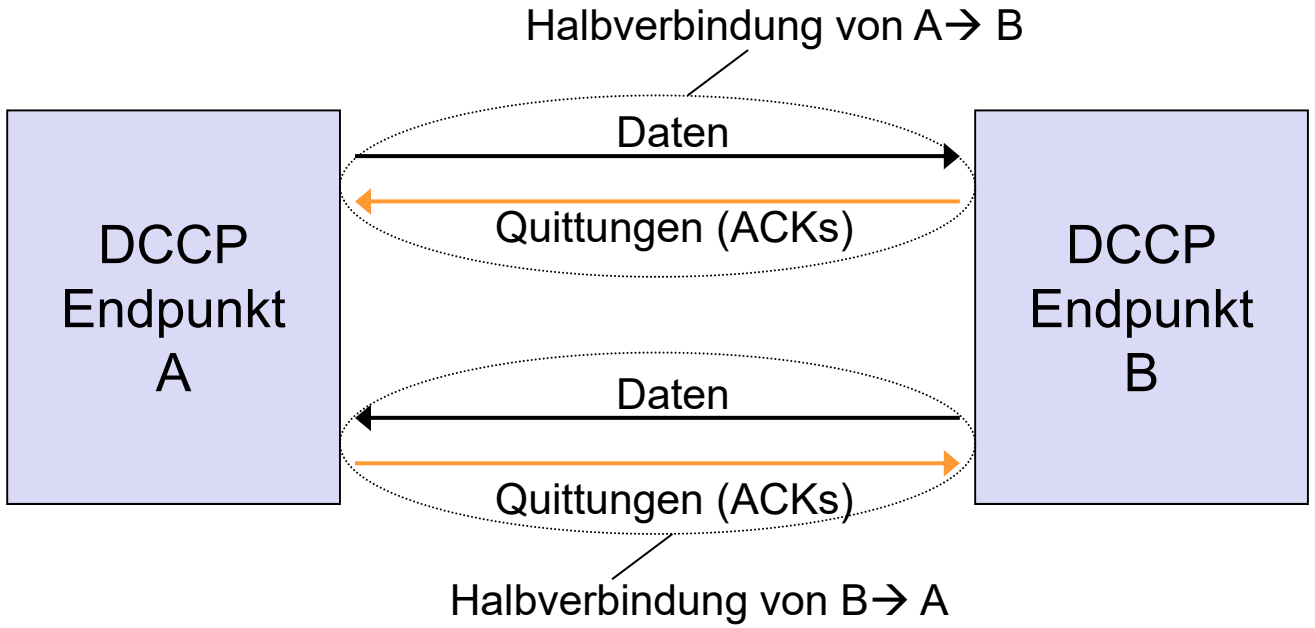
- Staukontrolle in Anwendungen korrekt zu implementieren ist nicht einfach
- Standardisierte Möglichkeit für Rückmeldung und Aushandlung der ECN-Nutzung fehlt
- Möglichkeit zur Auswahl TCP-freundlicher Staukontrollmechanismen
  - Fensterbasierte Flusskontrolle nicht immer optimal für alle Anwendungen

DCCP Merkmale

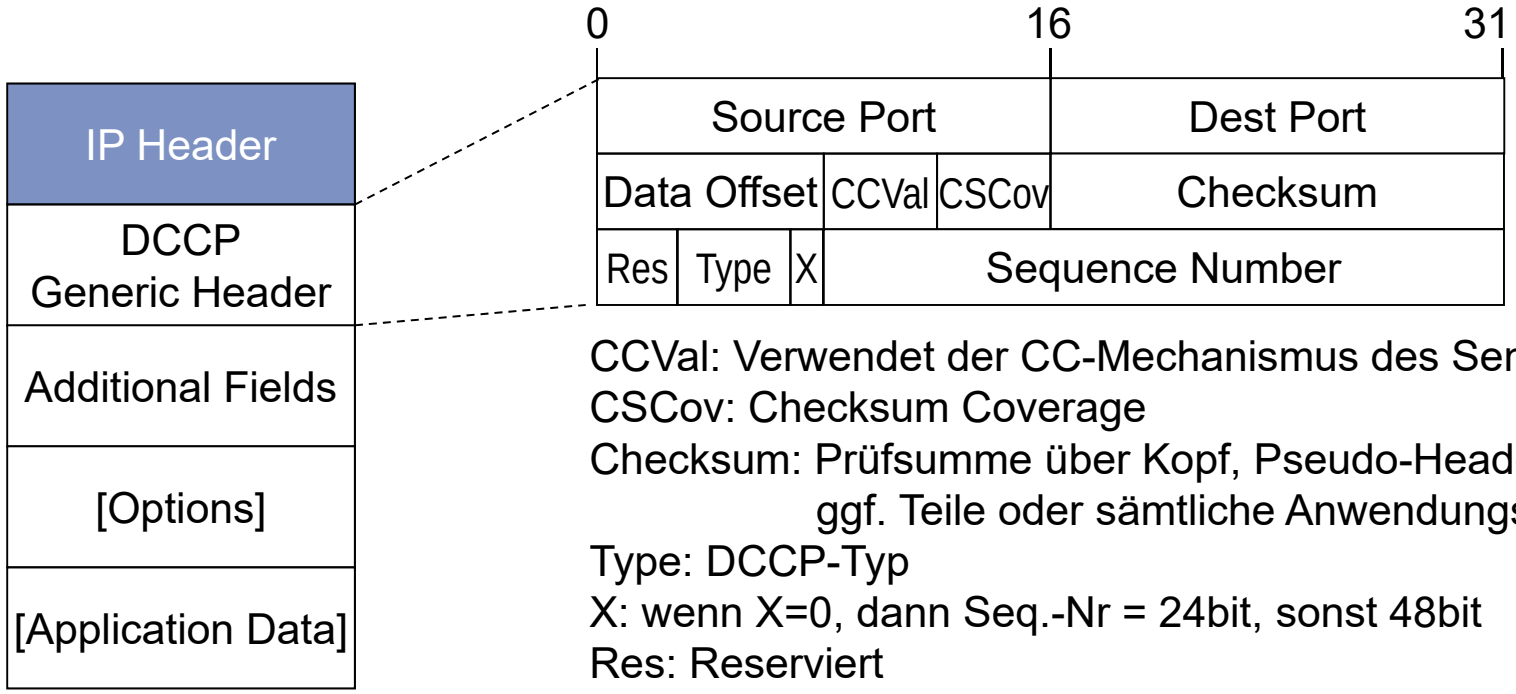
- Gesicherter Verbindungsaufbau mit 3-Wege-Handshake
  - erlaubt Aushandlung von Protokollmerkmalen
  - Init-Cookie-Option zur Verhinderung von DoS-Angriffen
- Unzuverlässiger Dienst aber mit Staukontrolle
  - Staukontrollmechanismus wählbar
    - TCP-like Congestion Control  [RFC4341]
    - TCP-friendly Rate Control  [RFC4348]
  - Nutzung des ECN-Verfahrens
  - Keine Flusskontrolle (Wieso?)
- Niedriger Zusatzaufwand (DCCP-Kopf: 12 Bytes statt 8 Bytes UDP)
- Sequenznummern zählen Pakete, nicht Bytes
- Unterscheidung der Ursachen für Paketverlust

DCCP-Verbindungen

- DCCP ist verbindungsorientiert!
- Ein Endpunkt initiiert die Verbindung aktiv (=Client), Server ist passiv
- DCCP-Verbindungen sind bidirektional, bestehen logisch aus zwei Halbverbindungen (praktisch Überlappung durch DCCP-DataAcks)



DCCP Paketformat

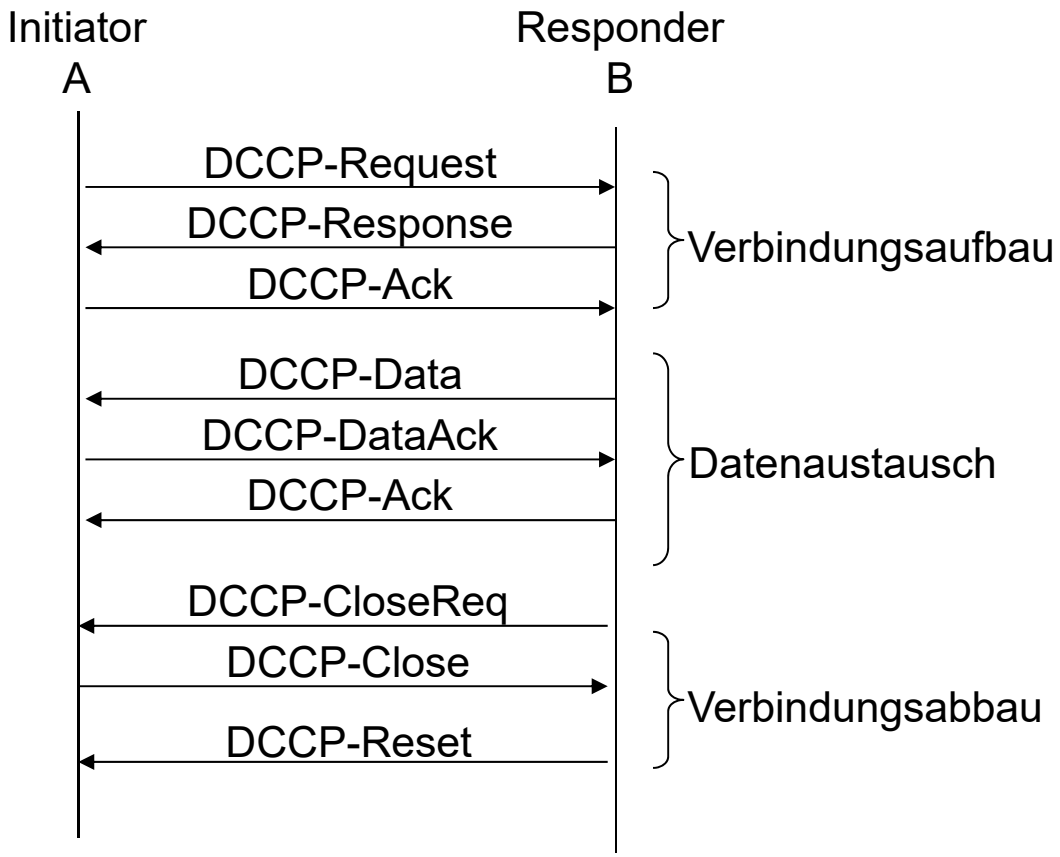


CCVal: Verwendet der CC-Mechanismus des Senders
CSCov: Checksum Coverage
Checksum: Prüfsumme über Kopf, Pseudo-Header und ggf. Teile oder sämtliche Anwendungsdaten
Type: DCCP-Typ
X: wenn X=0, dann Seq.-Nr = 24bit, sonst 48bit
Res: Reserviert

- 10 verschiedene Pakettypen:
  - DCCP-Request
  - DCCP-Response
  - DCCP-Data
  - DCCP-Ack
  - DCCP-DataAck
  - DCCP-CloseReq
  - DCCP-Close
  - DCCP-Reset
  - DCCP-Sync
    DCCP-SyncAck:
    Resynchronisation der Sequenznummern nach größeren Bündelverlusten

DCCP Verbindungsverwaltung

- Weg-Zeit-Diagramm



- Sofortige Quittierung (sobald Optionen verarbeitet)
- Kein Empfangsfenster (warum nicht?)
- Kein gleichzeitiger Verbindungsaufbau
- Keine halbgeschlossenen Zustände wie bei TCP

Aufgaben

1. Weshalb ist TCP für den Transport von Signalisierungsnachrichten nicht so gut geeignet?
2. Was ist eine SCTP-Assoziation?
3. Was zeichnet einen SCTP Stream aus?
4. Wie verhindert SCTP Denial-of-Service-Angriffe?
5. Für welche Anwendungen wird DCCP benötigt?
6. Weshalb gibt es bei DCCP keine Flusskontrolle?
7. Wie hoch ist der Zusatzaufwand im Vergleich zu UDP?
8. Welche Staukontrollalgorithmen werden durch DCCP unterstützt?

Literatur

[StXi02] R. Stewart, Q. Xie: Stream Control Transmission Protocol (SCTP), Addison-Wesley, 2002

[RFC 3758] R. Stewart, M. Ramalho, Q. Xie, M. Tuexen und P. Conrad. Stream Control Transmission Protocol (SCTP) Partial Reliability Extension. RFC 3758 (Proposed Standard), Mai 2004. URL: http://www.ietf.org/rfc/rfc3758.txt.

[RFC 4336] S. Floyd, M. Handley und E. Kohler. Problem Statement for the Datagram Congestion Control Protocol (DCCP). RFC 4336 (Informational), März 2006. URL: http://www.ietf.org/rfc/rfc4336.txt

[RFC 4340] E. Kohler, M. Handley und S. Floyd. Datagram Congestion Control Protocol (DCCP). RFC 4340 (Proposed Standard), März 2006. URL: http://www.ietf.org/rfc/rfc4340.txt

[RFC 4895] M. Tuexen, R. Stewart, P. Lei und E. Rescorla. Authenticated Chunks for the Stream Control Transmission Protocol (SCTP). RFC 4895 (Proposed Standard), August 2007. URL: http://www.ietf.org/rfc/rfc4895.txt.

Literatur

[RFC 4960] R. Stewart (Editor). Stream Control Transmission Protocol. RFC 4960 (Proposed Standard), September 2007. URL: http://www.ietf.org/rfc/rfc4960.txt

[RFC 5061] R. Stewart, Q. Xie, M. Tuexen, S. Maruyama und M. Kozuka. Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration. RFC 5061 (Proposed Standard), September 2007. URL: http://www.ietf.org/rfc/rfc5061.txt.