



3. Blatt: Network Protocols and Architectures, WS 13/14

Aufgabe 1: (10 Punkte) VoIP (Internet-Telefonie) über TCP

Nimm an, dass eine VoIP Verbindung über TCP aufgebaut wurde. Was passiert im Fall von einem Paketverlust?

Aufgabe 2: (10 + 10 = 20 Punkte) RTT-Abschätzung von TCP

Es geht um die Abschätzung der RTT (Round-Trip-Time), wie sie bei TCP verwendet wird. Sei SampleRTT_1 die zuletzt gemessene RTT, sei weiterhin SampleRTT_2 die zuvor gemessene RTT und so weiter. Erinnerung: die Formel zur Abschätzung der RTT wie folgt lautet:

$$\text{EstimatedRTT}_{\text{neu}} = (1 - \alpha) \cdot \text{EstimatedRTT}_{\text{alt}} + \alpha \cdot \text{SampleRTT} \quad (1)$$

Gegeben n RTT-Messungen, kann die Formel auch wie folgt verallgemeinert werden:

$$\text{EstimatedRTT}_{\text{new}} = \alpha \cdot \sum_{i=1}^{n-1} (1 - \alpha)^{i-1} \cdot \text{SampleRTT}_i + (1 - \alpha)^{n-1} \cdot \text{SampleRTT}_n \quad (2)$$

- In der Formel (2) gehe n gegen unendlich. Erkläre, wieso diese Art den Durchschnitt zu bilden „exponentiell-gleitender Durchschnitt“ heißt und warum nicht ein einfacher Durchschnitt genommen wird!
- Begründe, warum TCP die Messung von SampleRTT bei wiederholt übertragenen („retransmitted“) Segmenten nicht berücksichtigt!

Aufgabe 3: (10 + 10 = 20 Punkte) TCP-Sequenznummernbereich

Betrachte die Übertragung einer enorm großen Datei der Größe L Bytes von Rechner A zu Rechner B. Nimm dabei an, dass die „Maximum Segment Size“ (MSS) 1434 Bytes groß ist.

- Nimm an, dass die Datei in einer einzelnen TCP-Verbindung übertragen wird. Wie groß darf die Datei maximal sein, so dass die TCP-Sequenznummern nicht überlaufen („wrap around“)? Beachte, dass das Sequenznummernfeld bei TCP vier Bytes groß ist.
- Gegeben sei die Größe L der Datei aus (a). Bestimme die Zeit, die benötigt wird, um die Datei zu übertragen. Nimm an, dass insgesamt 66 Byte an Transport-, Netzwerk- und Datalink-Headern zu jedem Datensegment hinzugefügt werden, bevor dieses über einen 100Mbit/s-Link versendet wird. Ignoriere Fluss- und Staukontrolle (flow and congestion control), so dass A die Segmente kontinuierlich hintereinander versenden kann.

Bitte wenden!

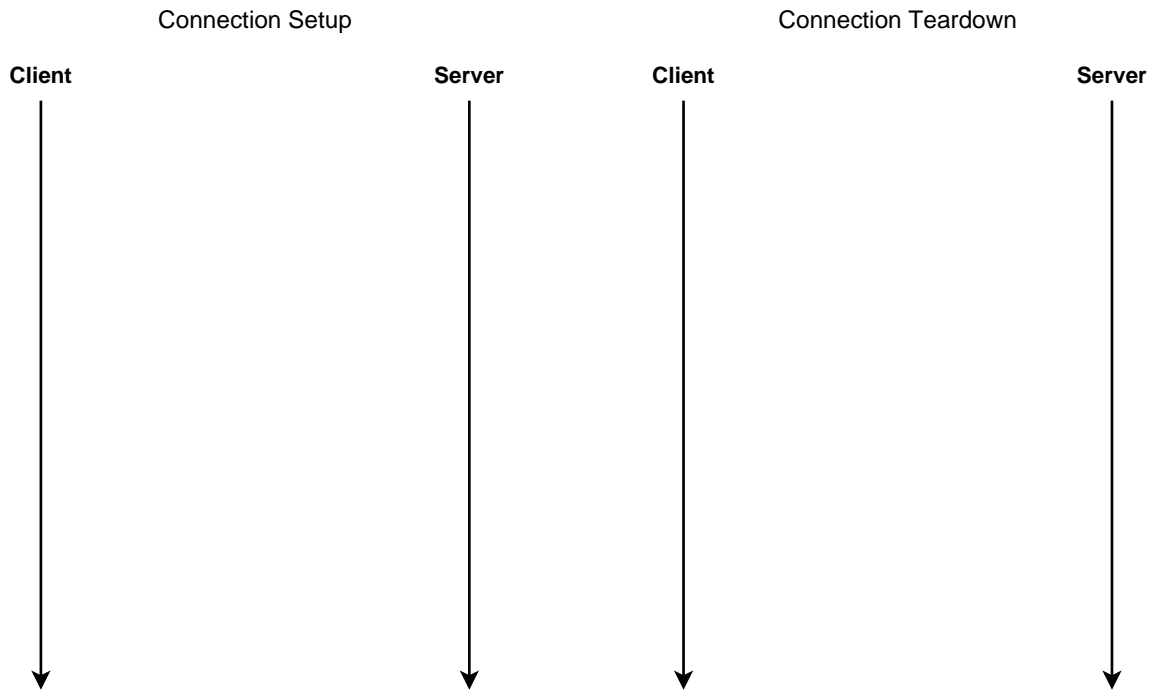
Aufgabe 4: (5 + 5 = 10 Punkte) *TCP-Sequenznummern*

Nimm an, dass Rechner A zwei TCP-Segmente direkt hintereinander an Rechner B über eine TCP-Verbindung schickt. Das erste Segment hat Sequenznummer 1346, das zweite 2016.

- Wieviele Daten (in Bytes) befinden sich im ersten Segment?
- Nimm an, dass das erste Segment verloren geht, das zweite Segment jedoch bei B ankommt. Welche Bestätigungsnummer schickt Rechner B daraufhin an Rechner A in der Bestätigung?

Aufgabe 5: (20 + 20 = 40 Punkte) *TCP-Verbindungsauf- und -abbau*

TCP gilt als Paradebeispiel für verbindungsorientierte Dienste. Im Folgenden wird das Verbindungsmanagement daher etwas genauer untersucht.



- Trage den Ablauf eines erfolgreichen Verbindungsaufbaus in ein Diagramm (siehe links oben) ein. Beschrifte die Pfeile mit den dabei relevanten Teilen der TCP-Segmentstruktur (Flags, Sequenznummer, Bestätigungsnummer). Die initialen (zufällig bestimmten) Sequenznummern von Client und Server sind 6400 (Client) und 15620 (Server).
- Trage den erfolgreichen Verbindungsabbau in ein weiteres Diagramm (siehe rechts oben) ein. Beschrifte wiederum die Pfeile mit den dabei relevanten Teilen der TCP-Segmentstruktur (Flags, Sequenznummer, Bestätigungsnummer). Gehe davon aus, dass seit dem Aufbau aus (a) folgende Daten übertragen worden sind: 490 Bytes vom Client zum Server und 12300 Bytes vom Server zum Client. Berücksichtige diese Werte zur Bestimmung der Sequenz- und Bestätigungsnummern.

Abgabe bis Mittwoch, den 13. November 2013 nur bis 09:55 h s. t.

- **Als PDF-Dateien (keine MS-Office- oder OpenOffice-Dateien):** Mittels ISIS hochladen (<https://www.isis.tu-berlin.de/2.0/course/view.php?id=349>)
- **In Papierform:** Postfach im Telefunkenhochhaus (Erdgeschoss, hinter dem Pfortner rechts)
- Gib auf deiner Lösung deinen Namen, deine Matrikelnummer **und** den Namen deines Tutors an.