H T
 W I
 G N
 Hochschule Konstanz Technik, Wirtschaft und Gestaltung (HTWG)
 Fakultät Informatik
 Rechner- und Kommunikationsnetze
 Prof. Dr. Dirk Staehle

Vorlesung Rechnernetze

Theorieübung zu Page Load Times

Prof. Dr. Dirk Staehle

Die Abgabe erfolgt durch Hochladen der Lösung in Moodle und exemplarisches Vorrechnen in der Laborübung.

Bearbeitung in Zweier-Teams

Team-Mitglied 1:

Team-Mitglied 2:

In dieser Aufgabe wird eine Web-Seite betrachtet, deren Aufbau in Tabelle 1 beschrieben ist. Hinweise zur Notation und Parameter der TCP-Verbindungen finden Sie in Tabelle 2.

- 1) Im Browser und auf dem Web-Server läuft die http-Version "Persistent http ohne Pipelining". Skizzieren Sie in Abbildung 1 den Verlauf der Seitenübertragung beginnend mit dem ersten Request bis zur vollständigen Übertragung von Bild 5. Tragen Sie pro RTT die Anzahl Segmente im Socket (Spalte Tx), die Sendefenstergröße (Spalte cwnd) sowie die Anzahl und den Inhalt der vom Server bzw. Client übertragenen Segmente ein. Gehen Sie davon aus, dass alle Pakete, die gleichzeitig gesendet werden auch gleichzeitig und in der richtigen Reihenfolge ankommen.
- 2) Im Browser und auf dem Web-Server läuft die http-Version "<u>Persistent http mit Pipelining</u>".Skizzieren Sie in Abbildung 2 den Verlauf der Seitenübertragung beginnend mit dem ersten Request bis zur vollständigen Übertragung von Bild 5.
- 3) Im Browser und auf dem Web-Server läuft die http-Version "Persistent http ohne Pipelining". Skizzieren Sie in Abbildung 3 die Übertragung von Bild 15 und Bild 16, wenn Segment 56 von Bild 15 als erstes und einziges Segment bei der Übertragung der Web-Seite verloren geht. Wie groß ist jeweils das Sendefenster des Web-Servers nach Empfang des letzten ACKs für ein Segment von Bild 15 bzw. von Bild 16?

Verwenden Sie folgende Notation, wobei Bytes die Anzahl der übertragenen Bytes pro Request bzw. Response ist.

- Requests: MOReq[Bytes], IOnReq[Bytes]
- Segmente der Objekte:
 - o MO:Sx[Start-Bytes]-Sy[End-Bytes]
 o IOn:Sx[Start-Bytes]-Sy[End-Bytes]
 - Segment- und Byteangaben können bei vollständige Objekten bzw.
 Segmenten entfallen
- ACKs: n x ACK (die Anzahl genügt)

Tabelle 1: Aufbau der Web-Seite

Objekt	Re	equest	Response (Objekt)		
	Bytes	Segmente	Bytes	Segmente	
HTML Code	1200	2	2000	3	
Bild 1	1600	2	3200	4	
Bild 2	3200	4	800	1	
Bild 3	4800	6	8000	10	
Bild 4	800	1	30000	38	
Bild 5	1200	2	11600	15	
Bild 6-14	800	1	8000	10	
Bild 15	800	1	52000	65	
Bild 16	800	1	11200	14	
Bild 17 - 27	800	1	3200	4	

Tabelle 2: Übersicht weiterer Parameter

MSS	800 Bytes
IW	2 MSS

Tx	cwnd	Client (Browser)	7	Server	cwnd	Tx
1200	2	MOReq	—————————————————————————————————————	2xACK, MO:S1-S2	2	2000
0	4	2xACK	\longrightarrow	MO:S3[400]	4	400
1600	4	IO1Req, 1xACK		2xACK, IO1:S1-S4	5	3200
3200	6	IO2Req, 4xACK		4xACK, I02:S1	9	800
4800	10	IO3Req, 1xACK		6xACK, IO3:S1-S10	10	8000
800	16	IO4Req, 10xACK		1xACK, IO4:S1-S20	20	30000
0	17	20xACK		IO4:S21-S38	40	14000
1200	17	IO5Req, 18xACK		2xACK, IO5:S1-S15	58	11200
	19	15xACK		ZAACK, 103.31 313	73	11200
	一					
$\vdash \vdash \vdash$	$\vdash \vdash \vdash$					
			$\rfloor \longrightarrow$			

Abbildung 1: Skizze für Persistent http ohne Pipelining

Matrikelnummer:

Tx	cwnd	Client (Browser)	1	Server	cwnd	Tx
1200	2	MoReq	\rightarrow	2xACK,MO:S1-S2	2	2000
0	4	2xACK		MO:S3[400]	4	400
29200	4	IO1Req,IO2Req[1600], 1xACK		4xACK, IO1:S1-S4	5	3200
26000	8	IO2Req[1600],IO3,4xACK		8xACK, IO2,IO3[6400]	9	8800
19600	16	IO4Req-IO19Req[400], 9xACK		16xACK, IO3,IO4[12800]	18	184800
0	32	IO19Req-IO27Req, 18xACK	$\qquad \qquad \longrightarrow$	9xACK,IO4,IO5	36	19920
0	41	36xACK	$\qquad \qquad \longrightarrow$, ,	72	170400
					$\vdash \vdash \vdash$	
					\vdash	
						

Name: Matrikelnummer:

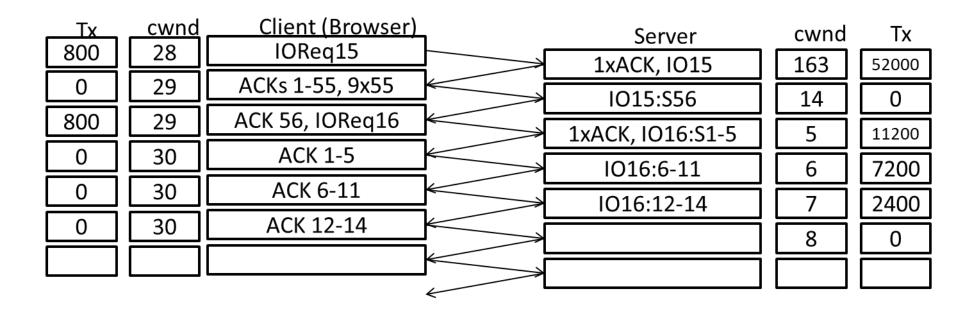


Abbildung 3: Skizze für Persistent http ohne Pipelining mit Paketverlust

Name: Matrikelnummer: