

IK1203

Networks and Communication

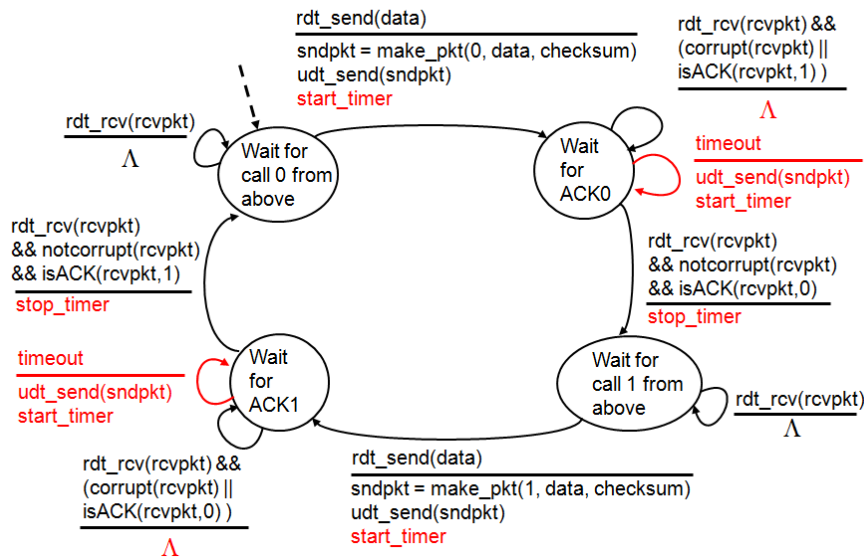
Recitation 2 – Transport layer

Solutions

1.
 - a) Because the retransmission timer can then expire and cause an unnecessary retransmission.
 - b) The receiver will immediately send an ACK (no more delay). This is what causes the typical “ACK every other segment” pattern in case of bulk transfers over TCP.
2. Flow control is a way for the receiver to regulate the sender’s transmission pace. The purpose is to prevent the receiver from becoming overwhelmed with more data than it can process. Congestion control deals with the situation when packets are dropped by routers along the path between sender and receiver. The purpose of TCP congestion control is to regulate the sender’s transmission pace based on the current network conditions.
3.
 - a) A window of data can be sent every 20th ms. It gives a maximum throughput of:
 $(65535 * 8) / (20 * 10^{-3}) \text{ bps} = 26\,214\,000 \text{ bps} (\approx 26 \text{ Mbps}).$
 - b) Channel utilization = $26\,214\,000 / 1\,000\,000\,000 \approx 2,6\%$
4. 35 ms. TCP begins in slow start by first sending 2 kB. After one RTT TCP will send 4kB; after two RTT TCP will send 8 kB; after 3 RTT TCP will send the last 1 kB of data ($2+4+8+1 = 15$). The last kB will reach the server application 5 ms later. So, in total: $3 * RTT + RTT/2 = 35 \text{ ms}$.
5.
 - a) Slow start: [1, 6] and [23, 26]
 - b) Congestion avoidance: [6, 16] and [17, 22].
 - c) In this case, the packet loss is detected through three duplicate ACKs. This can be concluded since TCP goes to fast recovery and reduces the congestion window to half (roughly) the current size. If the packet loss had been detected through a timeout, TCP would go to slow start and reduced the congestion window to 1 MSS.
 - d) After the 22nd transmission round, segment loss is detected due to timeout, and hence the congestion window size is set to 1.
 - e) The threshold is initially 32, since it is at this window size that slow start stops and congestion avoidance begins.
 - f) The threshold is set to half the value of the congestion window when packet loss is detected. When loss is detected during transmission round 16, the congestion windows size is 42. Hence the threshold is 21 during the 18th transmission round.
 - g) The threshold is set to half the value of the congestion window when packet loss is detected. When loss is detected during transmission round 22, the congestion windows size is 29. Hence the threshold is 14 (taking lower floor of 14.5) during the 24th transmission round.
 - h) During the 1st transmission round, packet 1 is sent; packet 2-3 are sent in the 2nd transmission round; packets 4-7 are sent in the 3rd transmission round; packets 8-15 are sent in the 4th transmission round; packets 16-31 are sent in the 5th transmission round; packets 32-63 are sent in

the 6th transmission round; packets 64 – 96 are sent in the 7th transmission round. Thus packet 70 is sent in the 7th transmission round.

- i) The threshold will be set to half the current value of the congestion window (8) when the loss occurred and congestion window will be set to the new threshold value + 3 MSS. Thus the new values of the threshold and window will be 4 and 7 respectively.
 - j) Threshold is 21, and congestion window size is 1.
 - k) Round 17, 1 packet; round 18, 2 packets; round 19, 4 packets; round 20, 8 packets; round 21, 16 packets; round 22, 21 packets. So, the total number is 52.
6. The finite state machine is illustrated below. The receiver's FSM is not changed when the timer-based retransmission mechanism is introduced.



7. a)

1. The computer on Newton Technologies, “atom.nt.com,” sends a DNS query to the local DNS server.
2. Since the local DNS server has an empty cache, the local DNS server sends the query to a root DNS server.
3. The root DNS server knows the IP address(es) of the the Top-Level Domain (TLD) servers for the top-level domain ”com” and responds with the IP addresses of those servers to the local DNS server.
4. The local DNS server sends the query to one of the TLD servers for the “com” top-level domain.

5. The “.com” TLD DNS server has the IP address to the authoritative DNS server for the domain name ”fws.com”. The TLD DNS server responds with this IP address to the local DNS server.
 6. The local DNS server sends the query to the authoritative DNS server for the domain name ”fws.com”. The authoritative DNS server is responsible for all names in this domain, so (most likely) it can answer the question.
 7. The authoritative DNS server responds to the local DNS server with the IP address(es) of the host in question.
 8. The local DNS server now has the complete answer to the query, and sends this answer back to “atom.nt.com”.
- b) DNS query (1) is processed in a recursive way by the local DNS server, while queries (2), (4) are (6) handled iteratively.
- c) Yes, there is nothing in DNS that would prevent this.
8. “Dig” is a very useful program for doing DNS queries from the command line. The dig commands for the (valid) queries are included in the table below.

Query description	Query type	Dig command
IP version 4 address for “www.kth.se”	A	dig a www.kth.se
IP version 6 address for “www.kth.se”	AAAA	dig aaaa www.kth.se
TCP port number for HTTP server at “www.kth.se”	–	
Incoming mail server for “kth.se”	MX	dig mx kth.se
Outgoing mail server for “kth.se”	–	
Authoritative name server for “kth.se”	NS	dig ns kth.se
Web server for “kth.se”	–	
Main (canonical) name for alias “www.kth.se”	CNAME	dig cname www.kth.se
Host name with address “130.237.28.40”	PTR	dig -x 130.237.28.40 <i>or</i> dig ptr 40.28.237.130.in-addr.arpa

Problems from course book (Kurose and Ross, 7th ed), Solutions

P45.

- a) The loss rate, L , is the ratio of the number of packets lost over the number of packets sent. In a cycle, 1 packet is lost. The number of packets sent in a cycle is

$$\begin{aligned}
 \frac{W}{2} + \left(\frac{W}{2} + 1\right) + \dots + W &= \sum_{n=0}^{W/2} \left(\frac{W}{2} + n\right) \\
 &= \left(\frac{W}{2} + 1\right) \frac{W}{2} + \sum_{n=0}^{W/2} n \\
 &= \left(\frac{W}{2} + 1\right) \frac{W}{2} + \frac{\frac{W}{2}(\frac{W}{2} + 1)}{2} \\
 &= \frac{W^2}{4} + \frac{W}{2} + \frac{W^2}{8} + \frac{W}{4} \\
 &= \frac{3}{8}W^2 + \frac{3}{4}W
 \end{aligned}$$

Thus the loss rate is

$$L = \frac{1}{\frac{3}{8}W^2 + \frac{3}{4}W}$$

- b) For W large, $\frac{3}{8}W^2 \gg \frac{3}{4}W$. Thus $L \approx \frac{3}{8}W^2$ or $W \approx \sqrt{\frac{8}{3L}}$. From the text, we therefore have average throughput

$$\begin{aligned}
 &= \frac{3}{4} \sqrt{\frac{8}{3L}} \times \frac{MSS}{RTT} \\
 &\approx \frac{1.22 \times MSS}{RTT \times \sqrt{L}}
 \end{aligned}$$

P46.

- a) Let W denote the max window size measured in segments. Then, $W \cdot MSS / RTT = 10$ Mbps, as packets will be dropped if the maximum sending rate exceeds link capacity. Thus, we have $W \cdot 1500 \cdot 8 / 0.15 = 10 \cdot 10^6$, then W is about 125 segments.
- b) As congestion window size varies from $W/2$ to W , then the average window size is $0.75W = 94$ (ceiling of 93.75) segments. Average throughput is $94 \cdot 1500 \cdot 8 / 0.15 = 7.52$ Mbps.
- c) $94/2 \cdot 0.15 = 7.05$ seconds, as the number of RTTs (that this TCP connections needs in order to increase its window size from $W/2$ to W) is given by $W/2$. Recall the window size increases by one in each RTT.