Report

By: Hansen Liu z5092212

Answer:

The code is implemented to transfer file from sender to receiver by STP protocal. All the required features are implemented successfully: The file can be sended successfully and can handle the packet drop, duplicate packet, corruppted packet, reordered packet.

The code is seperated into 2 parts: Receiver and Sender.

The STP header:

Sequence Number
Ack Number
Ack Flag
SEQ Flag
FIN Flag
Check_sum
Length
Data

This header includes the sequcen/ack numbers. There are also 3 flags to indicates what kind of this segment is. The checkSum field is used to detect the corruption in the segment. Lastly, there is a length field and most importanly the Data field to store the actual data.

3. Discuss any design trade-offs considered and made. Describe possible improvements and extensions to your program and indicate how you could realise them.

Trade offs: all the flags in my program are in the integer format which makes the header very big: They can replaced by just a single bit.
(a) Run your protocol using pDrop = 0.1, MWS = 500 bytes, MSS = 100 bytes, seed = 100, gamma = 4, and pDuplicate, pCorrupt, pOrder, MaxOrder, pDelay, MaxDelay all set to 0. Transfer the file test0.pdf (available on the assignment webpage). The file should be received correctly at the Receiver. Show the sequence of STP packets that are observed at the Receiver. It is sufficient to just indicate the sequence numbers of the STP packets that have arrived. Run an additional experiment with pdrop = 0.3, transferring the same file (test0.pdf). In your report, discuss the resulting packet sequences of both experiments indicating where dropping occurred. Also, in the appendix section show the packet sequences for both the experiments.

Answer:
When the droprate increase from 0.1 to 0.3, the toal number of segment received increased from 35 to 50.The hence the previous one is about 1.42 times more efficient than the later. This difference is even bigger than the difference between 0.9 and 0.7(the segments that are not dropped). This is because that the receiver needs duplicate ack segment to ackno ledge that there is a missing segment.

(b) The timeout for STP is given by: TimeoutInterval = EstimatedRTT + gamma * DevRTT where gamma will be supplied to the program as an input argument, see Section 4.5. Set pdrop = 0.5, MWS = 500 bytes, MSS = 50 bytes, seed = 300, pdelay = 0.2, MaxDelay = 1000 and pDuplicate, pCorrupt, pOrder, MaxOrder all set to 0. Run three experiments with the following different gamma values: i. gamma = 2 ii. gamma = 4 iii. gamma = 9 6 and transfer the file test1.pdf using STP. Show a table that indicates how many STP packets were transmitted in total and how long the overall transfer took. Discuss the results.

Gamma value	2	4	6
Packets	11513	11445	11338
time	658.64	891.71	1092.75

According the result above, it can be concluded that when the drop rate is relatively high, the wating time can be huge if the gamma is set too big. This is because the gamma is directly influent the timeout, the program is spending a lot of time on waiting the packet that will not arrive or is very late. However, since the MaxDelay is also very high, the number of total packets will be smaller if the gamma is set large. This is because the program with larger gamma value are more likely to detect the packets that gets delayed.

(c) Use the following values and run STP to transfer test2.pdf. MWS=500bytes MSS=50 gamma=4 pDrop=0.1 pDuplicate=0.1 pCorrupt=0.1 pOrder=0.1 maxOrder=4 pDelay=0 maxDelay=0 seed=300 Has the file been successfully transferred? How long the overall transfer took? For this experiment, which of the factor (out of pDrop, pDuplicate, pCorrupt and pOrder) is the most critical contributing most in the overall transfer time? How have you determined this? Provide the screen shot for the initial transfer (connection establishment + first 20 entries) and the last 20 entries plus the summary statistics table for the sender_log.txt and receiver_log.txt files in appendix. Do not attach the complete log files due to their sizes.

The pDrop and pCorrupt are the two main factors to slow down the transfer time. I determined this by several comparison of values. (By control variate mathod)

Appendix:
first 20 lines:

SND	0.00	S	0	0	0
RCV	0.02	SA	0	0	1
SND	0.02	A	1	0	1
SND	0.05	D	1	50	1
SND	0.05	D	51	50	1
SND	0.05	D	101	50	1
RCV	0.05	A	1	0	51
RCV	0.05	A	1	0	101
SND	0.05	D	151	50	1
RCV	0.05	A	1	0	151
RCV	0.05	A	1	0	201
SND	0.05	D	201	50	1
RCV	0.05	A	1	0	251
SND	0.05	D	251	50	1
RCV	0.05	A	1	0	301
DROP	0.05	D	301	50	1
RCV/DA	0.05	A	1	0	301
SND	0.05	D	351	50	1
SND	0.06	D	401	50	1
SND	0.06	D	451	50	1

last 20 lines:

SND	675.68	D	1605451	50	1
RCV	675.68	A	1	0	1605051
SND	675.68	D	1605501	50	1
RCV	675.68	A	1	0	1605101
RCV	675.68	A	1	0	1605151
SND	675.68	D	1605551	35	1
RCV	675.68	A	1	0	1605201
RCV	675.68	A	1	0	1605251
RCV	675.68	A	1	0	1605301
RCV	675.68	A	1	0	1605351
RCV	675.68	A	1	0	1605401
RCV/DA	675.68	A	1	0	1605401
RCV/DA	675.68	A	1	0	1605401

RCV/DA 675.68 A 1 0 1605401
SND/RXT 675.68 D 1605401 50 1
RCV 675.68 A 1 0 1605586
SND 675.68 F 1605586 0 1
RCV 675.68 A 1 0 1605587
RCV 675.68 F 1 0 1605587
SND 675.68 A 1605587 0 2

==========

Size of the file (Bytes) 1605585
Segments transmitted (including drop & RXT) 40297
Number of Segments handled by PLDModule 32112
Number of Segments dropped 3199
Number of Segments Corrupted 3048
Number of Segments Re-ordered 1966
Number of Segments Duplicated 2778
Number of Segments Delayed 0
Number of Retransmissions due to TIMEOUT 2493
Number of FAST RETRANSMISSION 5694
Number of DUP ACKS received 21007

==========