2020/10/23

Assignment 3

CS-335

ST No - 2004152588

Q:2

Performing 1's complement of the 8-bit bytes

We have (i) 01010011 (ii) 01100110 (iii) 01110100

→ adding (i) and (ii)

$$\begin{array}{r} 01010011 \\ +\ 01100110 \\ \hline 10111001 \end{array}$$

∴ (i) + (ii) = 10111001 — (iv)

→ adding (iv) + (iii)

$$\begin{array}{r} 01110100 \\ +\ 10111001 \\ \hline 100101101 \end{array}$$

→ adding carry

$$\begin{array}{r} 00101101 \\ +\ 1 \\ \hline 00101110 \end{array}$$

∴ (i) + (ii) + (iii) = 00101110 — (v)

→ reversing each bit for 1's complement from (v)

∴ 1's compliment = 11010001

→ The reason why we cannot use sum for UDP is because, checksum of UDP can be derived by 1's compliment and sum won't give the checksum of UDP.

→ As 1's compliment gives checksum, it is used to check errors. Also 1's compliment has bits which are off that gives error.

→ No 1-bit error will be missed, Only if all the bits are on, ~~then not~~. there won't be any errors. They will be caught using 1's compliment.

→ Possible that 2-bit errors went undetected. That is only when sum and checksum sees different bits as same.

Q: 3

Given that,

Window Size = 4 = (N)

Sequence number range = 1024

a) Consider

reciever got K packets (expected) and recieves with acknowledgement of K-1 packets. Window range for Sender will be [K, K-1+N] when there is no loss

If sender fails to recieve all the acknowledgements then window range will be [K-N, K-1].

∴ possible set of sequence number will lie in [K-N, K].

b) Considering recieuer is waiting for packets and recieves N-1 earlier packets before that then the range for ACK values will be [K-N, K-1]. (ACK - Acknowledgement).

This is because the ACK comes back when they fail to reach the sender. For a same reason when sender sends K-N packets and also recieved K-N-1 packets ACK, it won't send Ack which are before K-N-1.

∴ possible values of Ack recieved back to sender will lie in ~~[K-1, K-N-1]~~. [K-N-1, K-1].

Q:5

given that,

SampleRTT 106ms, 120ms, 140ms, 90ms, 115ms

EstimatedRTT $\alpha = 0.125$

EstimatedRTT before first = 100ms

DevRTT $\beta = 0.25$

DevRTT after each sample = (?)

TCP timeoutInterval = (?)

For 106ms,

$$\text{EstimatedRTT}_{106} = (1-\alpha)\,\text{EstimatedRTT} + \alpha \times \text{SampleRTT}$$

$$= (1-0.125) \times 100 + 0.125 \times 106$$

$$= \underline{100.75}$$

$$\text{DevRTT}_{106} = (1-\beta)\,\text{DevRTT} + \beta \times |\text{SampleRTT} - \text{Estimated}|$$

$$= (1-0.25) \times 5 + 0.25 \times |106 - 100.75|$$

$$= \underline{5.06}$$

$$\text{TimeoutInterval}_{106} = \text{EstimatedRTT} + 4 * \text{DevRTT}$$

$$= 100.75 + 4 \times 5.06$$

$$= \underline{120.99}$$

Similarly,

for	120	140	90	115
EstimatedRTT	103.15	107.76	10.5.54	106.72
DevRTT	8	14.07	14.43	12.89
TimeoutInterval	135.15	164	163.26	158.28

Q:6

given that,

Delay was introduced by the TCP slow-start phase

Rate of transmission between Client and Web is R

Maximum Segment Size (MSS) is S

Round Trip Time (RTT) is constant.

a. $4S/R > S/R + RTT > 2S/R \quad = \frac{4S}{R} > \frac{S}{R} + RTT > \frac{2S}{R}$

Total RTT for both sender and reciever is 2RTT

$2(S/R + RTT) + 12\,S/R + 2RTT = 4RTT + 14\,S/R$

$\therefore 4RTT + \frac{14S}{R}$ time is required to retrieve (a.) object.

b.

$3(S/R + RTT) + 8S/R + 2RTT = 5RTT + 11S/R$

C.

$1(S/R + RTT) + 14S/R + 2RTT = 3RTT + 15S/R$

Q:4

Given that,

Connection is TCP,

Total bytes recieved by Host B = 126

~~Number of~~ 2 segments were sent with 80 bytes and 46 bytes data size.

Sequence number for ~~the~~ first segment is 127

Port number 302.

a.

Sequence number before sending second segment will be,

$127 + 80 = 207$, source port will be 302 and destination port will be 80.

b. ACK of first arriving segment will have numbe 207, with source port 80 and destination port 302

c. ACK of First if second arrives first will have number 127

d. Following is a timing diagram showing all segments and ACKs. (Assuming no Packet losses).