**A TCP sender is just about to send a segment of size 100 bytes with sequence number 1234 and ack number 436 in the TCP header. What is the highest sequence number up to (and including) which this sender has received all bytes from the receiver?**

A. 1233

B. 436

C. 435

D. 1334

E. 536

Answer:

C. 435

Reason:

The sender is just about to send sequence number is 1234, the ack number is 436, that means ack number 435 has already been received

**Q2 A TCP sender is just about to send a segment of size 100 bytes with sequence number 1234 and ack number 436 in the TCP header. Is it possible that the receiver has received byte number 1335?**

1. Yes

2. No

Answer:

2. No

Reason:  
The sender is just about to send sequence number 1234, with a segment size of 100 bytes. Therefore, it is impossible for the receiver to have already received byte number 1335.

**Q3 A TCP sender maintains a SmoothedRTT of 100ms. Suppose the next SampleRTT is 108ms. Which of the following is true of the sender?**

1. Will increase SmoothedRTT but leave the timeout unchanged

2. Will increase timeout

3. Whether it increases SmoothedRTT depends on the deviation.

4. Whether it increases the timeout depends on the deviation

5. Will chomp on fries left over from the rdt question earlier

Answer:

4. Whether it increases the timeout depends on deviation.

Reason:

The timeout would depend on the SmoothedRTT and RTTVAR, if the deviaition is significant enough then the RTO will increase.

**Q4 A TCP sender maintains a SmoothedRTT of 100ms and DevRTT of 8ms. Suppose the next SampleRTT is 108ms. What is the new value of the timeout in milliseconds? (Numerical question)**

Answer:

133ms

Reason:

EstimatedRTT = 0.875 × 100 + 0.125 × 108 = 101ms

DevRTT = 0.75 × 8 + 0.25 × ∣108 − 100∣ = 8ms

Timeout = EstimatedRTT + 4 × DevRTT = 101 + 32 = 133ms

**Q5 Which is the purpose of the receive window field in a TCP header?**

A. Reliability

B. In-order delivery

C. Flow control

D. Congestion control

E. Pipelining

Answer:

C

Reason:

Ensures that the sender doesn't overwhelm the receiver with too much data at once, managing the flow of data between them.

**Q6 Roughly how much time does it take for both the TCP sender and receiver to establish connection state since the connect() call?**

A. RTT

B. 1.5RTT

C. 2RTT

D. 3RTT

Answer:

1.5RTT

Reason:

It takes 1 RTT to reach the receiver and half of an RTT to return.

**Q7 TCP uses cumulative ACKs like Go-back-N, but does not retransmit the entire window of outstanding packets upon a timeout. What mechanism lets TCP get away with this?**

A. Per-byte sequence and ack numbers

B. Triple duplicate ACKs

C. Receive window-based flow control

D. Using a better timeout estimation method

E. Ketchup (for the fries)

Answer:

B.

Reason:

When the sender receives three duplicate ACKs (Probably meaning that a specific packet was lost but the rest of the packets were received), it assumes that the missing packet was lost and retransmits it immediately, rather than waiting for a timeout. This avoids retransmitting the entire window.

**Q8 v A sender that underestimates the round-trip time of a connection may unnecessarily induce a TCP timeout v T/F**

Answer:

True

Reason:

The timeout might be set too short, because of this the sender might think that they are lost and trigger a timeout.

**Q9 v Which of the following services use TCP? § DHCP § SMTP § HTTP § TFTP § FTP**

Answer:

SMTP, HTTP and FTP

Reason:

DHCP and TFTP both use UDP not TCP

**Ben Bitdiddle’s home network connection can upload at 125,000 bytes per second. His router has 100,000 byte first in first out buffer for packets awaiting transmission.**

**If the buffer is completely full, how long will it take for the buffer to clear?**

A. 0.4 seconds

B. 0.6 seconds

C. 0.8 seconds

D. 1 second

E. 1.25 seconds

Answer:

C

Reason:

100,000 is 80% of 125,000, this bottleneck means the buffer will clear in 80% of a second in this case. 80% of a second = 0.8 seconds.

**Q11 Ben Bitdiddle’s home network connection can upload at 125,000 bytes/second. His router has a 100,000 byte first in first out buffer for packets awaiting transmission.**

**At time 0, Ben’s client starts sending 1,000 byte packets at 150 packet/s. When will the first packet be dropped by the router?**

A. 2 seconds

B. 3 seconds

C. 4 seconds

D. Buffer will never discard a packet in this case

Answer:

C

Reason:

Sending 150,000 bytes a second.

150,000 – 125,000 = 25,000 bytes a second processed.

100,000 / 25,000 = 4 second.

**Q12 Alyssa P.Hacker and Ben Bitdiddle communicate over a link with capacity of 100 pkts / sec. The latecy (RTT) on this link is 100 ms.**

**If a sliding window protocol with acknowledgement packets is used, and there is a FIXED window size of 4 packets, what is the maximum rate of traffic on the link?**

A. 20 pkts / s

B. 40 pkts / s

C. 80 pkts / s

D. 100 pkts / s

Answer:

B

Reason:

100ms = 0.1 seconds

100 packets x 0.1 = 10 packets a second.

Fixed windows seize x packets per second = 40 packets per second