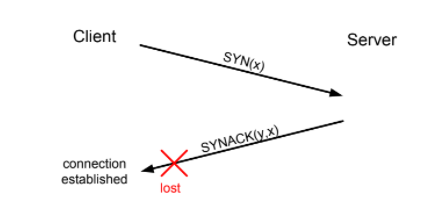
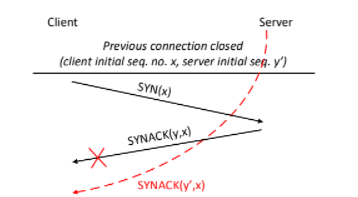
1. a.



If the SYNACK(y,x) does not reach the client then the client will send the SYN request the again to the server to connect again to the server.

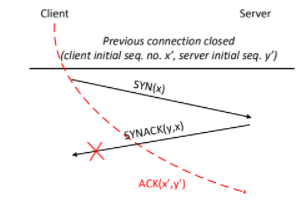
b.



yes, because it may happen to old SYN(x)message in the network and be received by the server at any time. In this scenario, three-way handshakes are used. If it is receiving SYN(x)message, the server sends the SYN ACK(x, y message to the client. on receiving SYN ACK (x, y) message from the server and client immediately reject that SYN ACK (x, y) message Connection closed duplicate SYN(x) SYN ACK(x, y) Receive data Rejected reject data

c.

SYN with the Sequence number A and sends a SYN segment with its own totally independent ISS number B in the Sequence number field. In addition, it sends an increment on the Sequence number of the last received segment (i.e., A+x where x is the number of octets that make up the data in this segment) in its Acknowledgment field. This Acknowledgment number informs the recipient that its data was received at the other end and it expects the next segment of data bytes to be sent, to start at sequence number A+x. This stage is aften called the SYN-ACK. It is here that the MSS is agreed. Host 1 receives this SYN-ACK segment and sends an ACK segment containing the next sequence number (B+y where y is the number of octets in this particular segment), this is called Forward Acknowledgement and is received by Host 2. The ACK segment is identified by the fact that the ACK field is set. Segments that are not acknowledged within a certain time span, are retransmitted.



d. The main reason of time-wait mechanism is to be used in TCP is to prevent delayed packets from one connection being accepted by a later connection. Concurrent connections are isolated by other mechanism

2. a. (i) An Acknowledgement or Negative Acknowledgement is a short message sent by the receiver to the sender to indicate whether it has correctly or incorrectly received a data packet, respectively. As stop-n-wait protocol is sent one packet and wait for the acknowledgement from receiver. This will be a bit slow in the total process, but this will be a noiseless channels protocols.

(ii) If we are expecting an acknowledgement for previously transmitted data, we check whether the received acknowledgment number is in the range between the present and the forward sequence number (in our example, this would be in the range between mySEQ + 1 and mySEQ + 3). If this is the case, our transmit data has been accepted, and we increment our sequence number by the number of bytes acknowledged. If the received sequence number does not correspond, we assume the remote has not yet received our transmit data, and we arrange for a re-transmission of our last data block sent.

b. The main use of sequence no. is for the receiver and whether sender is retransmitting previously transmitted packet ir a new packet. Consider a situation where the sender sends a packet 0 before packet 1 and the packets are received out of order to the receiver, such that packet 1 is received first. The receiver would have then sent an acknowledge for packet 1. The sender upon viewing this acknowledgement would move forward in module 2 arithmetic send a packet with sequence no. 0. However, at the same time, the receiver receives packet with sequence no 0. So, now the receiver has 2 packets with the same sequence no. with 0 Though, the packets are different, the receiver would not know that as they have the same sequence no., so it will disregard one packet considering it as a retransmission. In this case, using only one sequence member is not sufficient for identifying the packets to distinguish whether a packet is a retransmission or not.

c. Band width of the link =64 Mbps

propagation delay =8ms

size of data packets= 32 Kb =256x bits

size of acknowledgement =64 bytes= 64x8 bits=512 bits.

Total time= RTT +Dteams

RTT=8x8=16 ms

Dteams= L/R=(data+Ack)/bandwidth

=(256 x +512)/64 x sec

= 4.01 ms

Total time to transmit data= 16+4.01=20.01 ms

Useful time= L/bandwidth= 256/64=4ms

Efficiency or percentage of useful time= 400/20.01=19.99% ->20%

Throughput of the protocol= efficiency x Bandwidth

= 19.99 x8 x =20x 8x Mbps

3.Go-back-N vs Selective Repeat.

a.

(i) If a packet is malformed or out of order at the receiver's end, it will be silently discarded, and no acknowledgement will be sent.

(ii) If sender doesn’t receive an acknowledgement, then it discards all the following packets, and they have to be transmitted again.

b.

(i) In selective repeat, it doesn’t discard the unreceived packets. Rather, it sends a negative acknowledgement which allows the retransmission of the corrupted packet.

(ii) As only selected packets used to be retransmitted, sender doesn’t repeat the entire window. But only retransmits the missing packets.