SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

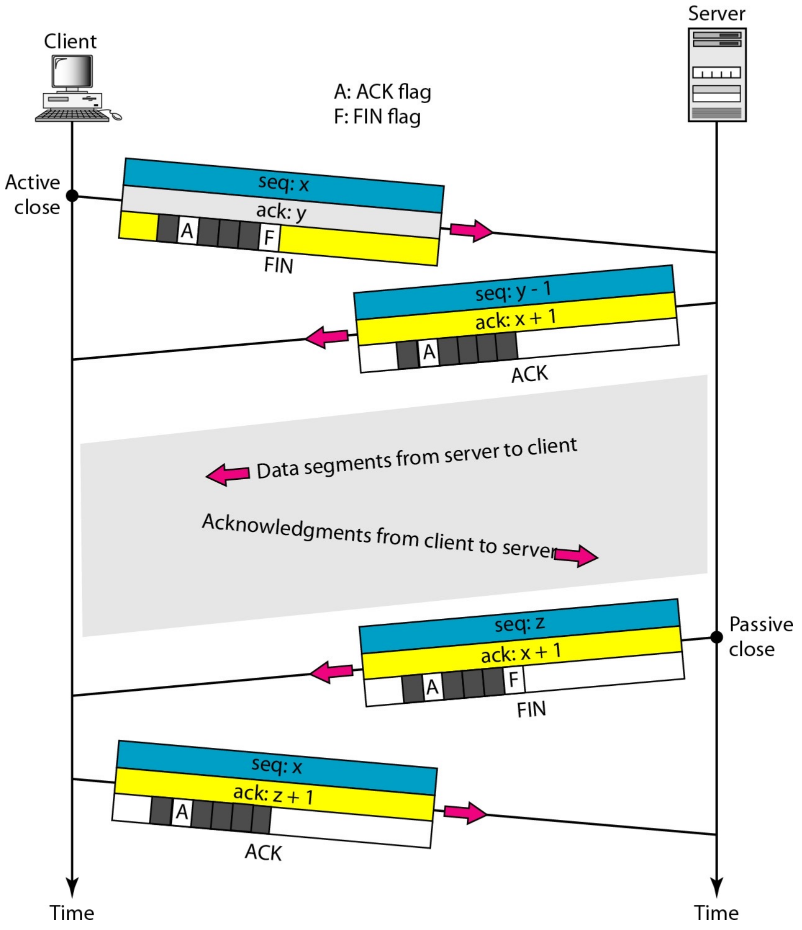
Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING

19CSB302- COMPUTER NETWORKS

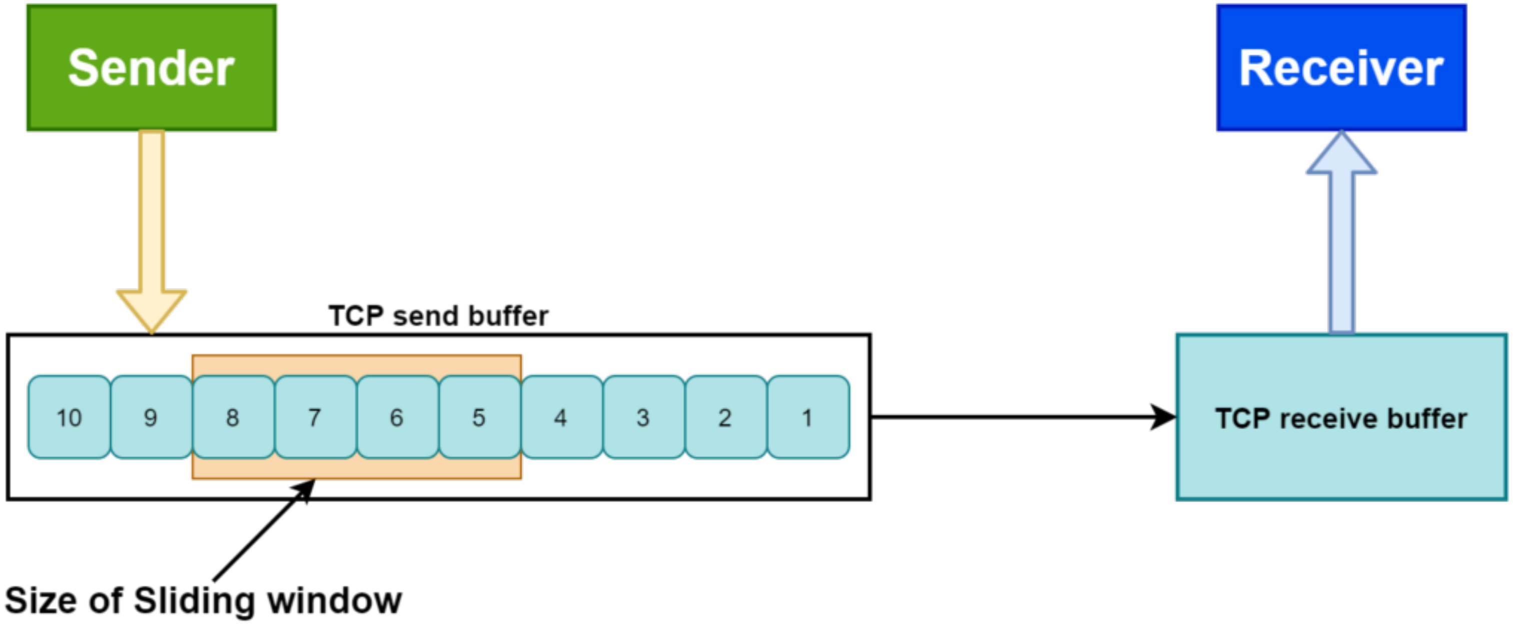
UNIT-4 TRANSPORT LAYER

Half close



- Client half-closes the connection by sending a FIN segment.
- Server sends an ACK segment.
- Data transfer from client to the server *stops*.
- After sending all data, server sends FIN segment to client, which is acknowledged by the client.

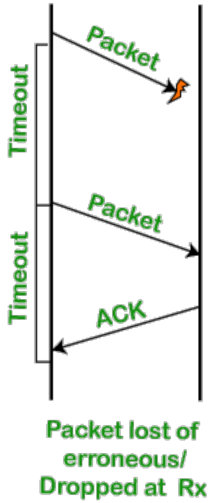
TCP FLOW CONTROL



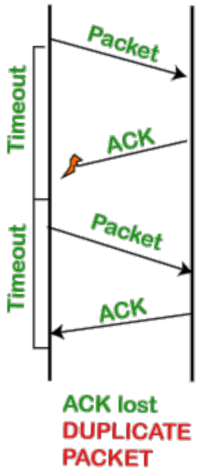
A sliding window is used to make transmission more efficient as well as to control the flow of data so that the destination does not become overwhelmed with data.TCP sliding windows are byte oriented

- The TCP stores the data that needs to be sent in the send buffer and the data to be received in the receive buffer.
- Flow control makes sure that no more packets are sent by the sender once the receiver’s buffer is full as the messages will be dropped and the receiver won’t be able to handle them.
- In order to control the amount of data sent by the TCP, the receiver will create a buffer which is also known as Receive Window.

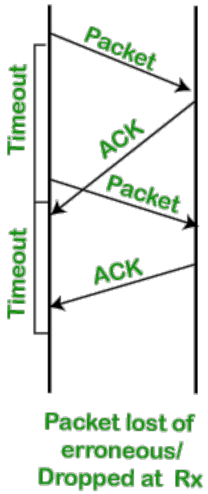
- The TCP retransmission means resending the packets over the network that have been either lost or damaged.



Data packet is lost



Acknowledgement is lost



Early timeout

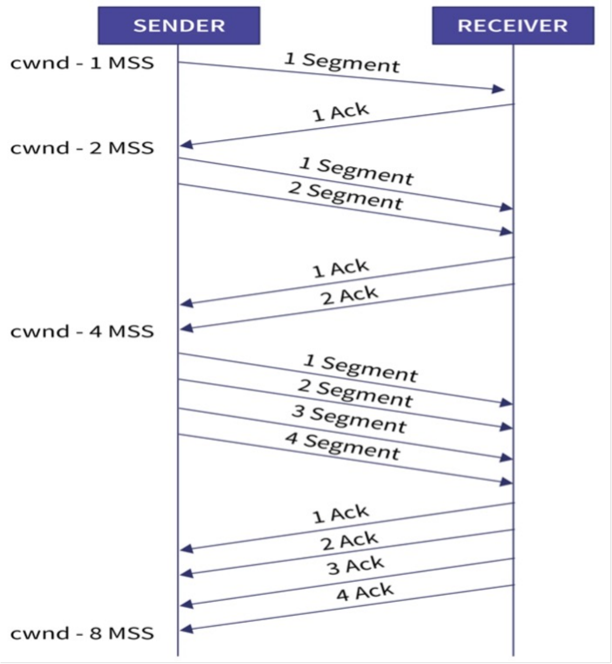
TCP Congestion Control

Congestion occurs if load (number of packets sent) is greater than capacity of the network (number of packets a network can handle).

- When too many packets are contending for the same link
- When load exceeds capacity, queues become full and the routers discard some packets and throughput declines sharply.
- ○ The queue overflows ○ Packets get dropped ○ Network is congested

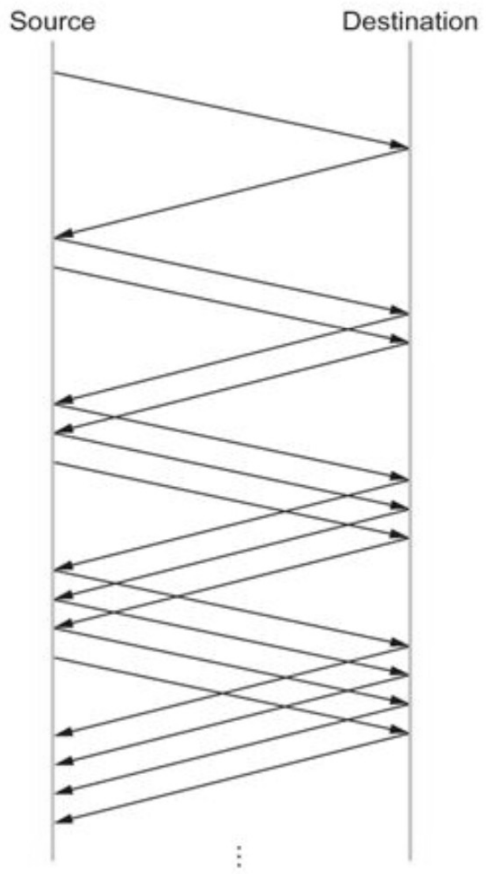
Slow Start Phase

- The sender sets congestion window size = maximum segment size (1 MSS) at the initial stage. The size of the congestion window increases **exponentially** in this phase.
- Slow start is repeated until CongestionWindow reaches **CongestionThreshold** and thereafter 1 packet per RTT.

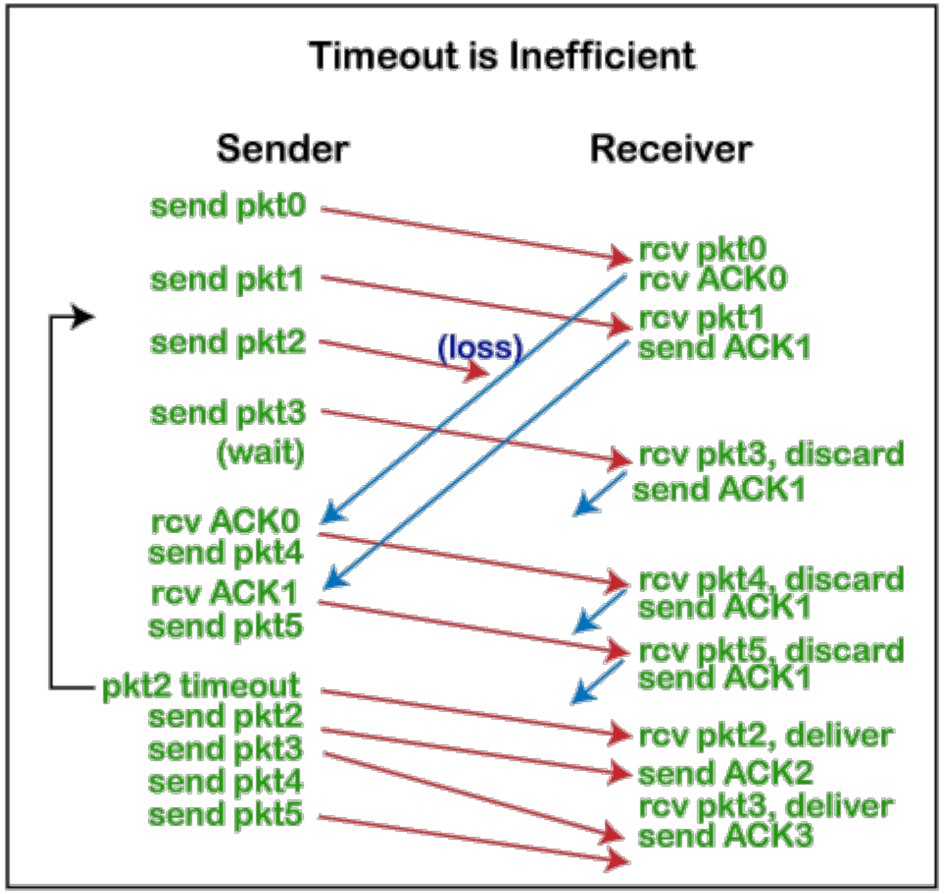


tive Increase / Multiplicative Decrease (AIMD)

- TCP source ***initializes*** CongestionWindow based on congestion level in the network.
- Source ***increases*** CongestionWindow when level of congestion goes down and ***decreases*** the same when level of congestion goes up.
- TCP interprets ***timeouts*** as a sign of congestion and reduces the rate of transmission.
- On timeout, source reduces its CongestionWindow by **half,** i.e., *multiplicative decrease*. For example, if CongestionWindow = 16 packets, after timeout it is 8.
- When ACK arrives CongestionWindow is *incremented* marginally, i.e., ***additive increase***.



Retransmit And Fast Recovery



- When a packet arrives **out of order**, receiving TCP resends the same acknowledgment (***duplicate ACK***) it sent last time.
- When ***three duplicate*** ACK arrives at the sender, it infers that corresponding packet may be lost due to congestion and retransmits that packet. This is called ***fast retransmit*** before regular timeout.