Congestion Control

- TCP implements congestion control at the sender
  - This control is intended to reduce congestion in the network.
- The sender has two parameters for congestion control:
  - Congestion Window ($cwnd$; Initial value is MSS bytes)
  - Threshold Value ($ssthresh$; Initial value is 65536 bytes)
- The window size at the sender is set as follows:
  Allowed Window = MIN (advertised window, congestion window)
  advertised window: flow-control window at the receiver
  MSS: Maximum Segment Size (set with option field in TCP header)

Example of Slow Start

- The congestion window size grows very rapidly
  - For every ACK, we increase $cwnd$ by 1 irrespective of the number of segments
  - ACKed
  - With byte-counting, the $cwnd$ would increase by the bytes acknowledged in an ACK, which means in the example $cwnd$ would become 4MSS when the ACK for segments 2+3 is received
- TCP slows down the increase of $cwnd$ when $cwnd > ssthresh$

Slow Start

- Whenever starting traffic on a new connection, or whenever increasing traffic after congestion was experienced:
  - Set $cwnd = \text{MSS}$ bytes ($cwnd$ is stored in bytes)
  - Each time an ACK is received, the congestion window is increased by 1 segment (= MSS bytes).
  - If an ACK acknowledges two segments, $cwnd$ is still increased by only 1 segment (without byte counting)
  - Even if ACK acknowledges a segment that is smaller than MSS bytes long, $cwnd$ is increased by MSS bytes.
  - If $cwnd$ is 3 but there is still one outstanding ACK, the sender can only send two segments
- Does Slow Start increment slowly? Not really. In fact, the increase of $cwnd$ is exponential

Normal operation of Slow Start / Congestion Avoidance

If $cwnd <= ssthresh$ then
  /* Slow Start Phase */
  Each time an ACK is received:
  $cwnd = cwnd + \text{segsize}$

else /* $cwnd > ssthresh$ */
  /* Congestion Avoidance Phase */
  Each time an ACK is received:
  $cwnd = cwnd + \text{segsize} \times \text{segsize} \div cwnd + \text{segsize} / 8$
endif

$\text{segsize} = \text{MSS}$
Slow Start/Congestion Avoidance Example

- Assume that \( ssthresh = 8 \)

### Computation of cwnd on previous slide

- Upto and including ack 2561, this TCP connection is in slow start, and cwnd is increased by 1 MSS bytes each time an ACK is received.
- Note that when cwnd = ssthresh, slow start is still applied. Hence when ack 2561 is received, cwnd = 2560 + 512 = 3072.
- When the last ack shown on the previous slide is received, the TCP connection is in congestion avoidance since cwnd is > ssthresh. Therefore, \( cwnd = cwnd + MSS \times MSS / cwnd + MSS / 8 = 3072 + 512 \times 512 / 3072 + 512 / 8 = 3222 \)

Example of slow start and congestion avoidance

- Assume MSS=512 bytes; advertised window = 5120 bytes

<table>
<thead>
<tr>
<th>Event</th>
<th>cwnd</th>
<th>ssthresh</th>
</tr>
</thead>
<tbody>
<tr>
<td>cwnd=512; ssthresh=2560</td>
<td>512</td>
<td>2560</td>
</tr>
<tr>
<td>PSH 1:513 (512) ack 10</td>
<td>cwnd=1024</td>
<td></td>
</tr>
<tr>
<td>ack 513</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cwnd=2560; ssthresh=2560</td>
<td>2560</td>
<td>2560</td>
</tr>
<tr>
<td>PSH 1537:2049 (512) ack 10</td>
<td>cwnd=3072</td>
<td></td>
</tr>
<tr>
<td>cwnd=3072; ssthresh=2560</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enter congestion avoidance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>cwnd=3222; ssthresh=2560</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

When congestion occurs: Congestion Avoidance Algorithm

- When congestion occurs (indicated by timeout),
  - ssthresh is set to half the current window size (the minimum of the advertised window (AW) and cwnd): \( ssthresh = \min(cwnd, AW) / 2 \) but at least 2 segments
  - cwnd is changed according to:
    \[ cwnd = 1 \ \text{segsize} = 1 \ \text{MSS bytes} \text{ in case of timeout only} \]
- When new data is acknowledged, cwnd is increased according to whether it is in slow start or CA