**Early Retransmit Threshold:** $\frac{1}{4}$ or $\frac{1}{8}$ #945

- Time-based loss detection recommends $\frac{1}{8}$ RTT threshold
- Early retransmit uses $\frac{1}{4}$ RTT because following Linux
- Older Google experiments indicate $\frac{1}{4}$ RTT is too slow

**Proposal:** Make them both $\frac{1}{8}$th until we have more data
Max Data Received before sending an ACK #1428

Reno is the documented congestion controller

Reno is primarily ACK-clocked

Sending ACKs less frequently increases the amount of time Reno is CWND limited, particularly during slow start

Proposal: Sender sends a transport param indicating retransmittable bytes received before sending an ACK
Explicit Max Ack Delay #981

- TCP has a proposal for explicitly communicating MAD
- Current text causes a spurious TLP the first time an ACK is delayed (by 25ms) if the RTT is ~10ms

Options:

1. Assume 25ms ack delay until we have ‘enough’ data
2. Add a transport param to communicate MAD
Max Ack Delay Permanent Increase #1438

Max Ack Delay is currently a max over the entire connection. Max Ack Delay informs the TLP and RTO timeouts.

If an ACK is lost, and the largest_acked does not increase, the next ACK may be sent with a very long ack delay.

Proposal: If the largest_acked does not change, set the ack_delay to 0 to indicate largest_acked is old.
Removing MinRTO #1017

MinRTO is currently 200ms

RTO already includes MaxAckDelay

TCP MAD proposal removes MinRTO and instead bases it on explicit max ack delay

Proposal: Remove MinRTO