Low Latency Low Loss Scalable throughput (L4S) and RACK
– an opportunity to remove HoL blocking from links
Recent ACKnowledgements (RACK): Background

• Loss is when sender deems absence has been long enough
  • Classic TCP: 3 DupACKs
  • TCP RACK: a fraction ($\varepsilon$) of the RTT (termed the reordering window)

• Tradeoff – larger $\varepsilon$:
  • minimizes spurious retransmissions (before ACKs of reordered packets arrives)
  • but takes longer $(1+\varepsilon)\times$RTT to repair genuine losses

• So, RACK adapts the reordering window:
  • starts small (which rapidly repairs losses in short flows)
  • then adapts to measured reordering degree (rapid loss repair less critical for performance of elephants)

• See draft-ietf-tcpm-rack-04
L4S Recap

- Motivation
  - Extremely low queuing delay for all Internet traffic, including link saturating
  - already 1-2 orders better than state of the art
  - 500 μs vs 5-15 ms (fq-CoDel or PIE)

- Architecture

\[ r \propto \frac{1}{p} \]
\[ r \propto \frac{1}{\sqrt{p}} \]

\( r \): packet rate
\( p \): drop/mark probability
5th Requirement for L4S senders

- L4S 'TCP Prague' Requirements (for all transports protocols, not just TCP) draft-ietf-tsvwg-ecn-l4s-id-05#section-4.3
- to use ECT(1), a scalable congestion control:
  - MUST NOT detect loss in units of packets
  - rather, by counting in units of time
- Then link technologies that support L4S can remove head-of-line blocking delay
  - see Appendix A.1.7

like the TCP 3DupACK rule
like TCP RACK
Why the “MUST NOT”?  
• “to use ECT(1), a scalable congestion control MUST NOT detect loss in units of packets”
Benefits of universal RACK to links (1/2)

- as well as e2e (layer-4) benefits, RACK offers potential for link (layer-2) performance improvements
- as flow rates scale up
  - with 3 DupACK rule
    - reordering tolerance time scales down
    - for multi-channel (bonded) links, skew tolerance time scales down
  - with rule relative to RTT
    - tolerance time remains constant
      (given min practical e2e RTT remains fairly constant)
Benefits of universal RACK to links (2/2)

• for lossy links (e.g. radio)
  - with 3 DupACK rule
    • link rcvr buffers packets behind each gap while link re-xmts
    • head-of-line blocking
    • recall that packets on a link will be from different flows and different streams within flows
  - with rule relative to RTT
    • link rcvr can forward packets out of order
    • no reordering buffer
    • in parallel, link rexmt will typically fill gap within min RACK reordering window

• e.g. RTT=24ms
  - 12 pkts / RTT
  - 6ms
  - 96 pkts / RTT

- retransmit buffer
- resequencing buffer
- RACK adaptation range
- RACK min, e.g. RTT/8
- fwd'ing
- 750μs

3 DupACK rule
For discussion

• MUST NOT use packet counting at all (for L4S congestion controls)
  • is stricter than RACK
  • RACK starts with 3 DUP-ACK, then evolves to measured reordering window

• Starting with, say, RTT/8 would be an alternative
  • But at the start of a flow, SRTT is not (always) a good estimate
  • For TFO, might be completely wrong
  • But is it any more wrong than 3 DupACK?

• Discuss