Adding Acknowledgement Congestion Control to TCP

S. Floyd, A. Arcia, D. Ros, and J. Iyengar

draft-floyd-tcpm-ackcc-01.txt

TCPM

July 2007

What is this draft about?

- Adding an optional congestion control mechanism to TCP for pure ack traffic.
- Based on the ack congestion control (ackcc) in DCCP's CCID 2.
- Urgent? Nope.
- Useful? Probably.
  - E.g., TCP connections over asymmetric links.
  - Reducing congestion for aggregate traffic.
- Questions? Many.

How would TCP's ackcc work?

- Negotiation between sender and receiver:
  - (Ack-Congestion-Control-Permitted option).
- Start with an Ack Ratio of 2.
- The sender detects lost Ack packets:
  - And tells the receiver the new Ack Ratio.
- The sender uses Appropriate Byte Counting and rate-based pacing (in response to Acks acking more than two packets).

Related work:

- BPK97, Balakrishnan et al.:
  - Based on ECN, sender reporting ECN-marked ACK packets to receiver.
- TJW00, Ming-Chit et al.:
  - Receiver-based Ack congestion control.
- CCID-2, Floyd and Kohler:
  - The sender detects lost or marked ACKs,
  - computes the desired ACK ratio,
  - tells the receiver.

Possible Complications:

- Delayed acknowledgements.
- Duplicate acknowledgements.
- Two-way traffic.
- Reordering of Ack packets.
- Abrupt changes in the Ack path.
- …

Congestion on the reverse path:

- Does pure Ack traffic really contribute to congestion?
  - Yes, somewhat, if the queue is in units of packets.
  - Measurement studies of congested links?
- How might ackcc be useful to the connection?
  - ECN-capable ACK packets.
  - Possibly reducing the ACK drop rate even without ECN.
- How might ackcc be harmful to the connection?
  - Costs of a larger Ack Ratio.

Security Considerations:

- Cheating with ECN-capable ACK packets?
  - If the receiver cheats, the sender could detect it.
  - If the sender cheats, the receiver can't easily detect it.
    - Middleboxes probably could detect it.

Questions:

- A TCPM work item, for Experimental?
- Feedback?