TCP ACK Rate Request (TARR) option

draft-ietf-tcpm-ack-rate-request-03

Carles Gomez

Universitat Politècnica de Catalunya

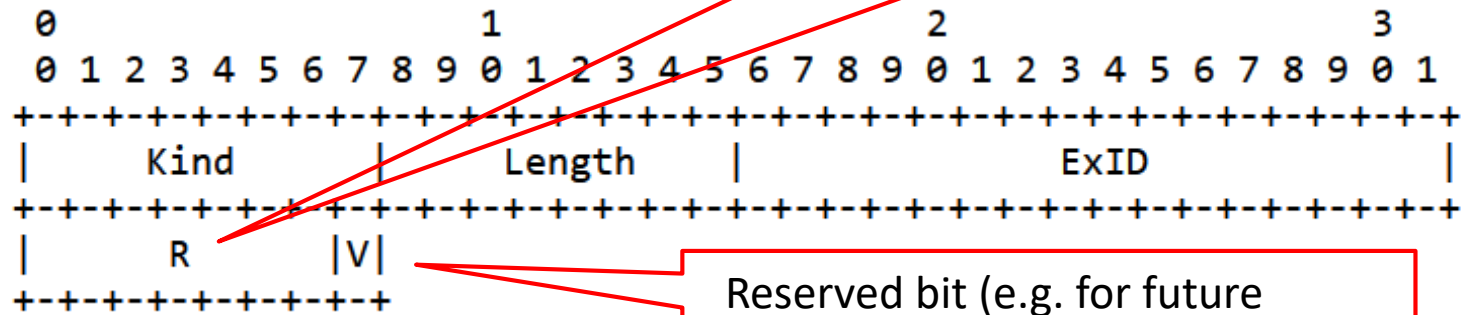**Jon Crowcroft**

University of Cambridge

IETF 118 Prague, TCPM WG, November 2023

Intro: motivation

- Delayed ACKs
  - Intended to reduce protocol overhead
  - But may also contribute to suboptimal performance

- "Large" cwnd scenarios (i.e. cwnd >> MSS):
  - Saving more than 1 of every 2 ACKs may improve performance
- "Small" cwnd scenarios (i.e. cwnd up to ~1 MSS):
  - Delayed ACKs may incur delay, limit cwnd growth...

Intro: main TARR option format

R carries binary encoding of ACK rate
- Maximum value of R: 127

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      Kind     |      Length   |            ExID               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|      R        |V|
+-+-+-+-+-+-+-+-+
```

Reserved bit (e.g. for future encodings/values of R, if needed)

- **"R" is the ACK rate requested by the sender**
  - R = 0: request an immediate ACK (but keep steady state R)

3

Status

- ## WG adoption
  - ## draft-ietf-tcpm-ack-rate-request-00
    - Same content as draft-gomez-tcpm-ack-rate-request-06
  - ## February 2023

- ## Version -03
  - Aims to address comments from IETF 117
  - Incorporates feedback from draft-ietf-quick-ack-frequency authors

Updates (I/V)

- Section 3.1: "Sender behavior"
  - Clarifications (in red):
    - A TCP sender MUST NOT communicate a value of R corresponding to an amount of data bytes to be acknowledged at once by the receiver greater than the last known rwin size or greater than the current cwnd size

  - Also, added:
    - Requesting an immediate ACK from the receiver can help reduce the time it takes to detect and/or recover from packet loss

Updates (II/V)

- Section 3.2: "Receiver behavior" (I/II)
  - Clarifications/Additions (in red):
    - Following the behavior specified in RFC 5681, in order to aid the sender in segment loss detection and repair, a TARR-option-capable receiver SHOULD send a duplicate ACK immediately when an out-of-order segment arrives [RFC5681], regardless of the ACK rate requested by the sender.

  - Also, added:
    - A TARR-option-capable receiver SHOULD send an immediate ACK when the incoming segment fills in all or part of a gap in the sequence space [RFC5681], regardless of the ACK rate requested by the sender

Updates (III/V)

- Section 3.2: "Receiver behavior" (II/II)

  - Also, added:

    – In any case, as specified in RFC 9293, the delay for an ACK MUST be less than 0.5 seconds

Updates (IV/V)

- Section 5.3: "Lower frequency of RTT samples"

  - Updated (in red):
    - In order to limit this issue, when there are segments in flight, a sender needs to trigger a sufficient number of ACKs per round trip.

    - Thanks to Ian Swett!

Updates (V/V)

- Section 6: "Changing the ACK rate during the lifetime of a TCP connection"
  - The sender may notice that the ACKs it receives cover more segments than the ACK rate requested
  - Two reasons:
    - ACK decimation is occurring en route
      » The sender may reduce the ACK frequency to reduce receiver workload and network load up to the ACK decimation point
    - The receiver uses Large Receive Offload (LRO)
      » The sender may want to increase the ACK frequency to compensate for the impact of LRO

Thanks!
Questions? Comments?

Carles Gomez

Universitat Politècnica de Catalunya

Jon Crowcroft

University of Cambridge

IETF 118 Prague, TCPM WG, November 2023