Byte and Packet Congestion Notification

draft-briscoe-tsvwg-byte-pkt-mark-02.txt

Bob Briscoe, BT & UCL

IETF-71 tsvwg Mar 2008

updated individual draft

- Byte and Packet Congestion Notification
  - **updated draft:** draft-briscoe-tsvwg-byte-pkt-mark-02.txt
  - **intended status:** informational
  - **immediate intent:** move to WG item

reminder (exec summary)

- question: in any AQM (e.g. RED drop, RED ECN, PCN) should we allow for packet-size when network writes or when transport reads a loss or mark?
- propose AQM SHOULD NOT give smaller packets preferential treatment
- adjust for byte-size when transport reads NOT when network writes

Terminology: RED's 'byte mode queue measurement' (often called just 'byte mode') is OK, only 'byte mode packet drop' deprecated

NOTE: don't turn off RED completely: drop-tail is as bad or worse

why decide now?

between transport & network

- near-impossible to design transports to meet guidelines [RFC5033]
  - if we can’t agree whether transport or network should handle packet size
- DCCP CCID standardisation
  - hard to assess TFRC small packet variant experiment [RFC4828]
- PCN marking algorithm standardisation
  - imminent (chartered) but depends on this decision
- part of answering ICCRG question
  - what’s necessary & sufficient forwarding hardware for future cc?
  - ICCRG open issues draft intends to incorporate this I-D by ref
- given no-one seems to have implemented network layer bias
  - advise against it before we’re stuck with an incompatible deployment fork
- what little advice there is in the RFC series (on RED) is unclear:
  - it seems to give perverse incentives to create small packets
  - it seems to encourage a dangerous DoS vulnerability
- encouraging larger PMTUs by not favouring smaller ones
  - may start to solve other scaling problems

widespread updates & restructuring

following long discussion at IETF-70 with Sally Floyd

deltas summarised in draft

full diff at <www.cs.ucl.ac.uk/staff/B.Briscoe/pubs.html#byte-pkt-mark>

- explained why I-D advice doesn’t deprecate ‘buffer carving’
- distinguished separate arguments against:
  - normalising TCP’s bit-rate with packet-size in queues
  - favouring control packets by queues favouring small packets
- added test whether a congestion ctrl scales with pkt size
- gave up trying to coin a word for both drop & ECN
- generalised to all congestible forwarding, not just IP
  - ie any queue, but also non-queue examples (wireless)

non-issue
‘buffer carving’: fixed size packet buffers

- some memory carved into pools of different fixed size pkt buffers
  - Q. can favour small packets, so are we deprecating what already exists?
  - A. no
- this I-D distinguishes two issues
  1. whether to measure congestion in packets or bytes
  2. whether dropping or marking a specific packet depends on its size

1. measuring congestion of fixed size packet buffers
   - should be, and is, in packets – relative to max no of buffers for size of pkt
   - borrowing of large buffers by small packets simply means smaller packets see a max no of buffers that includes the larger buffers
   - smaller packets see less drop because they actually do cause less congestion
2. dropping or marking a specific packet
   - doesn’t depend on its own size in any of these architectures (complies with I-D)

BTW, artificially favouring small pkts (e.g. RED byte-mode drop)
designed to advantage small packets far more than the outcome of buffer carving

expedients have unintended consequences

tempting to reduce drop for small packets

- drops less control packets, which tend to be small
  - SYNs, ACKs, DNS, SIP, HTTP GET etc
- but small != control
  - favouring smallness will encourage smallness, not ‘controlness’
    - malice: small packet DoS
    - innocent experimentation: “Hey, smaller packets go faster”
      OS tweaks, application evolution

principles, not expedients

- I-D sets principle and now gives numerous examples of
  - good transport practices making control packets robust to drop
  - most now in progress through IETF transport area

conclusion

- unequivocal UPDATE to RFC2309 (‘RED manifesto’)
  - adjust for byte-size when transport reads NOT when network writes
  - previously gave both options with ‘more research needed’
- all known implementations follow this advice anyway
  - retrospective tidy-up to RFC series
- still some consensus to reach
  - but should be as WG item now
  - if WG item, I’ll spend time compressing the incremental additions

Byte and Packet Congestion Notification

draft-briscoe-tsvwg-byte-pkt-mark-02.txt

Q&A