Byte and Packet Congestion Notification

draft-briscoe-tsvwg-byte-pkt-mark-01.txt

Bob Briscoe, BT & UCL

IETF-70 tsvwg Dec 2007

updated individual draft

- Byte and Packet Congestion Notification
  - **updated draft:** draft-briscoe-tsvwg-byte-pkt-mark-01.txt
  - **intended status:** informational
  - **immediate intent:** move to WG item

reminder (exec summary)

- in any AQM
  propose SHOULD NOT give smaller packets preferential treatment
- adjust for byte-size when transport reads NOT when network writes

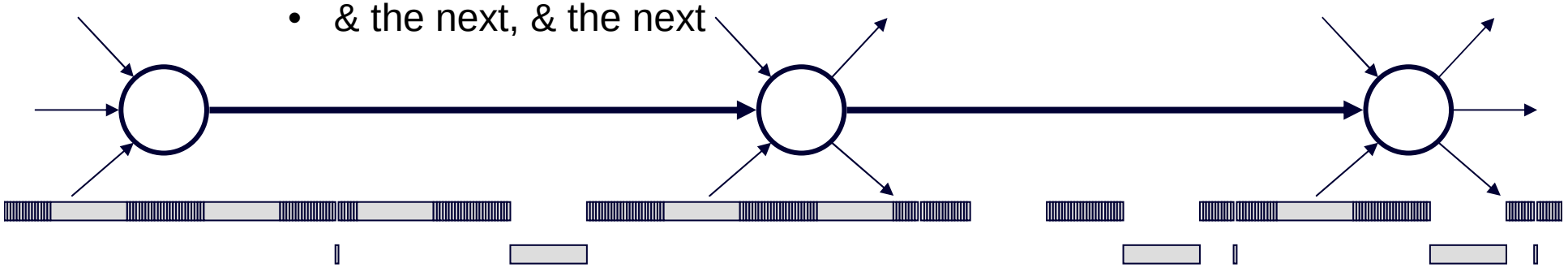
Terminology: RED's 'byte mode queue measurement' (often called just 'byte mode') is OK, only 'byte mode packet drop' deprecated

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

favouring small packets, main change:

DoS vulnerability

- small packet attacks push out larger packets
  - leaving most small packets to attack the next queue
  - & the next, & the next



- DoS vulnerability similar to that of drop tail queues
- AQM was partly about not locking-out large packets*
  - shouldn't add lock-out back again in the AQM algorithm

* not stated and not a motivation according to at least one author (Floyd)

other changes

- emphasised equal applicability to any AQM and to drop or ECN
  - e.g. PCN, RED (with drop or ECN)
- restructured
  - pulled main recommendations together into the conclusions
  - moved a couple of lumps of text to appendices
- fixed for (Floyd's) original motivations for RED's byte-mode drop
  - protecting SYNs & pure ACKs more than equalising small segment TCPs
- added more examples of preferable transport approaches
  - tcpm-ecnsyn & tcpm-ackcc added to TFRC-SP etc
- updated survey data (but no change since IETF-69 slides)
- clarification & update throughout
- full diff at <www.cs.ucl.ac.uk/staff/B.Briscoe/pubs.html#byte-pkt-mark>

thoughts for next draft

- long off-list discussions haven't resolved differences, but could
- main points in favour of size-dependent drop:
  - control packets tend to be small (e.g. SYNs, pure ACKs)
    - so less drop of small packets gives performance win
  - already have mix of size-dependent (drop-tail) and size-independent drop
    - so doesn't reduce complexity by only having size-independent
  - apps have other (OS) incentives not to use small packets
- main points in favour of size-independent drop
  - not all small pkts are control, so favouring all smallness creates unintended consequences
  - the more size-independent AQM, the less transport uncertainty over queue behaviours
  - mustn't provide incentives for new transports to use small data pkts
- possible ways forward
  - focus only on PCN?
  - but still mileage in reaching consensus on RED too

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 don’t do byte-mode drop anyway
  - retrospective tidy-up to RFC series
- not reached consensus

- discuss

- WG item pls

Byte and Packet Congestion Notification

draft-briscoe-tsvwg-byte-pkt-mark-01.txt

Q&A

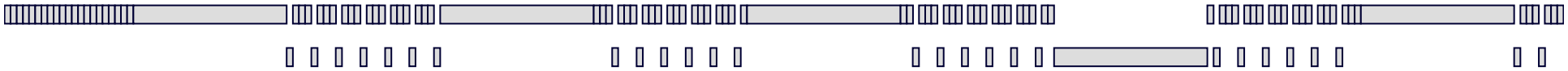
example: comparing each RED mode

simple packet streams (no congestion response)

RED packet-mode packet drop

- same drop probability for any packet
- universally deployed
- propose: SHOULD

	1500B pkts	60B pkts
input	1Mbps	1Mbps
drop prob.	25%	25%
output	750kbps	750kbps

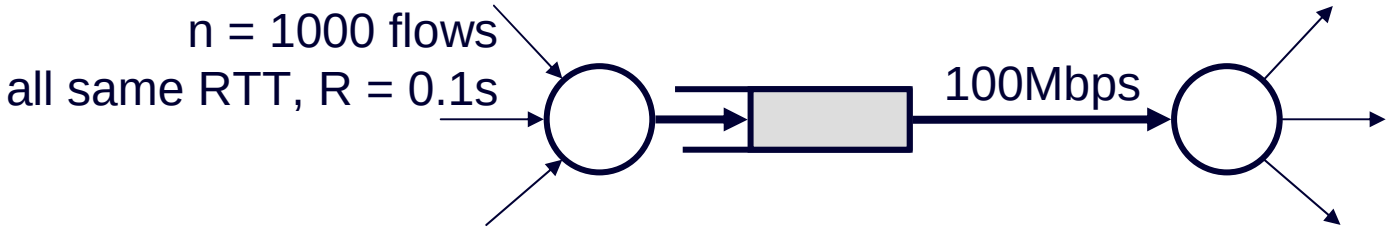


RED byte-mode packet drop

- lower drop probability for smaller packets
- ‘RED’ RFC2309 (sort of) recommends
- propose: SHOULD NOT

	1500B pkts	60B pkts
input	1Mbps	1Mbps
drop prob.	25%	1%
output	750kbps	990kbps

proof



- proof strategy
  - fix the amount of congestion in flight, then consider how much notification needed
- Imagine aggregate overload of 103Mbps in flight (for 1 RTT)
- arrives at queue running at its desired operating point
  ⇒ 3Mbps for 0.1s = 300kb = 37.5kB to discard (or mark)
- If all the traffic is in packets of size s [B], e packets need to be lost

s	e
1500B	25
60B	625
9000B	4

- to lose this excess
- if queue reduces drop rate for smaller packets
  - if all packets are small, queue will have to be longer

  to notify the same congestion queue shouldn't have to change its length
- so transports need to respond more strongly to larger missing packets

∴ TCP's insensitivity to drop size is an artefact, not a principle to be copied