

Pacing in Transport Protocols

draft-irtf-iccrp-pacing-02

Michael Welzl, Wesley Eddy, Vidhi Goel, Michael Tüxen

ICCRG
IETF 126

What's new? draft-irtf-iccrg-pacing-01

- A discussion of Appropriate Byte Counting (ABC)
 - RFC 5681 recommends limiting increase by 1 SMSS per ACK in Slow Start
 - With pacing, this may be unnecessary
 - HyStart++ (RFC 9406) specifies that no such limit should be applied ("L=infinity")
- More FreeBSD text
 - Clarify that socket options set the *maximum* rate
 - Update the packetdrill example to use a lower rate, such that the effect becomes more pronounced
 - Describe what we see in the packetdrill example: that the pauses after micro-bursts of 4 packets are the expected delays
 - Clarify that this is only valid for the RACK stack

What's new? draft-irtf-iccrg-pacing-02

- More details on Apple OSes
 - API allows to enable pacing and set a maximum rate
 - Target rate calculated on every cwnd update as $\text{cwnd} / \text{SRTT}$ in bytes / s
 - Doubled in slow start
 - Application cap, if set, is then applied
- Burst budget of roughly 244 μs of data, with a minimum of one MSS.
 - Packets are assigned transmit timestamps using a leaky-bucket scheme
 - Same timestamp for packets within the current burst budget
 - Budget exhausted: advance timestamp by $\text{budget} / \text{rate}$
 - Packet(s) carry timestamp into the per-interface AQM, which holds the packet in its per-flow queue until the wall clock reaches that timestamp before transmitting it

Next:

WGLC?

Please give us your feedback!