HyStart++: Modified Slow Start for TCP

draft-ietf-tcpm-hystartplusplus-00

TCPM, IETF 108
July 27, 2020

Praveen Balasubramanian, Yi Huang, Matt Olson
HyStart++ Recap

• Slow Start can overshoot ideal send rate & cause massive packet loss
  • Increased retransmissions
  • Time spent in recovery
  • Sometimes results in RTO (retransmission timeout)

• HyStart++
  • Simple modification to Slow Start
  • Only use Delay Increase algorithm from original HyStart
  • Compensate for premature slow start exit
  • Use max of Limited Slow Start (RFC3742) and Congestion Avoidance
  • Define tuning constants based on measurements and deployment experience
Algorithm Details

• On each ACK in slow start
  • Update the cwnd
  • If taking an RTT sample, measure current round’s MinRTT

• For each round (round approximates an RTT)
  • Remember last round’s minRTT
  • If cwnd >= (LOW_CWND * SMSS) and at least N_RTT_SAMPLE RTT samples taken
  • Check if currentRoundMinRTT is greater than lastRoundMinRTT + Threshold
  • If yes, set ssthresh = cwnd, exit slow start and enter Limited Slow Start (LSS)

• For each ACK in LSS
  • Update the cwnd as Max of RFC3742 and CA_cwnd()

• Exit HyStart++ on first congestion signal

• SHOULD use on first slow start and MAY use after idle
Changes since draft-03

• Incorporated review feedback by Mark Allman
• Changed Intended Status to Proposed Standard per WG feedback
• Key changes
  • New section on Tuning Constants
  • New section on Deployments and Performance Evaluations
• Tuning Constants
  • LOW_CWND = 16.
    • Low values can prevent measuring last round RTT, high values may cause overshoot
  • N_RTT_SAMPLE = 8
    • Low values lower accuracy, high values may add RTT measurement overhead
  • MIN_RTT_THRESH = 4 msec, MAX_RTT_THRESH = 16 msec
    • Small Min can cause early exit, higher Max can cause missed delay spikes
  • LSS_DIVISOR = 0.25
    • Not to exceed 0.5 otherwise too aggressive, lower values may lower perf
Status & Next Steps

• Implementations
  • Windows TCP
  • CloudFlare QUIC
  • Linux TCP – original HyStart (based on delay increase)

• Algorithm improvements
  • Look into usage of bandwidth or throughput estimate

• Measurements
  • More end to end measurements on workloads
  • Compare HyStart++ and BBR STARTUP phase

• Please review and provide feedback on draft
• Please share any performance data with the community