HyStart++: Modified Slow Start for TCP

TCPM, IETF 106

Praveen Balasubramanian, Yi Huang, Matt Olson

Microsoft

HyStart Recap

- Slow Start can overshoot ideal send rate and cause massive packet loss

- HyStart: Exit slow start early based on Delay Increase algorithm
  - Inter-Packet Arrival algorithm does not perform well due to ACK compression

- Delay Increase algorithm works well but has false positives
  - Latency fluctuations on wireless links
  - Transient queue buildup

HyStart "Delay Increase" algorithm

- Keep track of minimum observed RTT in each round in slow start

- For rounds where cwnd is at or higher than MIN_SSTHRESH and N_RTT_SAMPLE RTT samples have been obtained

```
Eta = clamp(MIN_ETA, lastRoundMinRTT / 8, MAX_ETA)
if (currentRoundMinRTT >= (lastRoundMinRTT + Eta))
    ssthresh = cwnd
    exit slow start
```

- MIN_SSTHRESH = 16, MIN_ETA = 4 msec, MAX_ETA = 16 msec, N_RTT_SAMPLE = 8

HyStart++

- HyStart “Delay Increase” for only the initial slow start
- Compensate for premature slow start exit
  - Congestion Avoidance algorithm can take time to ramp up
- Use Limited Slow Start (RFC3742) until next congestion signal

- For each arriving ACK in LSS, where N is the number of previously unacknowledged bytes acknowledged in the arriving ACK:

  K = cwnd / (LSS_DIVISOR * ssthresh)

  cwnd = max(cwnd + N / K, CA_cwnd())
- LSS_DIVISOR = 0.25

Test framework & Metrics

- A/B test framework using emulated WAN environment
- Test parameters
    - Latency
    - bottleneck buffer size
    - Bandwidth
    - artificial random loss
- Metrics
    - Goodput
    - Retransmitted bytes %
    - Fast retransmits
    - Timeouts
    - Loss recovery success rate

Results

- 100 Mbps bandwidth, BDP size bottleneck buffer
- For large RTT flows (100 msec)
  - Up to 39% improvement in average and P90 goodput for short flows
  - Up to 14% improvement in average and P90 goodput for long flows
- No noticeable improvement for small RTT flows (50 msec, 25 msec)
- Across all tests
  - Number of bytes retransmitted reduced by 50%
  - Number of RTOs reduced by 36%
  - Loss recovery success rate improves 43.48% -> 52%
- Awaiting results from production A/B test
  - Preliminary numbers show 20% reduction in retransmissions – not scoped

Status & Next Steps

- HyStart++ is deployed on by default for all connections
  - Windows 10 May 2019 Update onwards
  - Windows Server 2019 1903 version onwards
- Draft Status
  - draft-balasubramanian-tcpm-hystartplusplus-01 posted
- More A/B tests, please suggest interesting test cases
- Future: compare HyStart++, BBR STARTUP phase, and Paced Chirping

- Adopt document in tcpm