Update on LISA: A Linked Slow-Start Algorithm for MPTCP

draft-barik-mptcp-lisa-01

Runa Barik, Simone Ferlin, Michael Welzl
University of Oslo

97th IETF Meeting
Seoul
15 November 2016
Context

• This is a simple algorithm that avoids a multiple-IW burst mixed with ongoing SS
  – Reduces loss, improves Flow Completion Time (FCT)
  – In emulation and in real-life testbed (NorNet)

• Published at ICC

• Presented to MPTCP at IETF 94
  – Feedback: show results with different BDP

• Presented to MPTCP at IETF 96
  – Very little time, doesn’t fit charter
• Decision: discuss it in ICCRG

• If MPTCP charter updated to allow congestion control changes, bring it back there

• Else publish it via ICCRG as Experimental IRTF doc
  – Need reviews!
What’s the problem?

• No coupling in SS
  – Does not seem necessary: subflow 1 doubles, subflow 2 doubles $\Rightarrow$ aggregate doubles
  – However, IW of new subflows starting makes this different
  – New subflows typically start at the same time, while SS is already ongoing

• E.g. 8 subflows = SS + IW70

2 subflows, shared bottleneck:
2.5Mbps, 70ms
transfer size: 300KB
Short overview

• LISA tries to be no more aggressive than one TCP during SS

• Basic approach in LISA:
  – From subflows in SS, select subflow with maximum sending rate (max_subflow)
  – From max_subflow’s cwnd, take between 3 and 10 packets as “packet credit” to give new_subflow as IW
  – Max_subflow does not increase cwnd for (packets_inflight – cwnd) ACKs
  – If no max_subflow, IW of new_subflow = 10
Previously shown results

• **Retransmissions & completion time**, depending on:
  
  – *File size*: problem does not occur with very small files (need to send multiple IWs), and becomes less relevant for very large files
  
  – *Queue length*: whether queue exceeded heavily (many losses & retransmissions → long FCT) or not is a matter of luck: depends on how (# subflows*IW) aligns with (BDP+queue). By reducing the burst, LISA sometimes helps
  
  – *Number of subflows*: more subflows make the problems worse and the effect of LISA better
  
  – *RTT, Bandwidth*: varying BDP has similar effect to varying queue length
Example result: non-shared BN

This is a “lucky-BDP” case...
Non-shared bottleneck
5Mbps, 40ms, 2 subflows

At the end of slow start (after loss recovery)
Retransmissions until then
Shared Bottleneck
5Mbps, 40ms, 2 subflows

At the end of slow start (after loss recovery)

Retransmissions until then
Next steps

• LISA paper, draft, presentations, Linux kernel patch available from:
  http://heim.ifi.uio.no/~runabk/lisa/

• Please read & comment!