L4S Status Update

Bob Briscoe, Independent <ietf@bobbriscoe.net>
Koen De Schepper, NOKIA Bell Labs <koen.de_schepper@nokia.com>
Olivier Tilmans, NOKIA Bell Labs <olivier.tilmans@nokia-bell-labs.com>
Asad Sajjad Ahmed, Independent <me@asadsa.com>
Joakim Misund, Uni Oslo <joakim.misund@gmail.com>

tsvwg, IETF-110, Mar 2021
Focus

- Prague CC implementation
  - Faster getting up to speed (GUTS)
- Parallel activities
  - Prague requirements survey + text updates
  - Ops guidance on Classic ECN AQM issue

CC: Congestion Control
ECN: Explicit Congestion Notification
AQM: Active Queue Management
New Prague CC Draft

- New full write-up
  - draft-briscoe-iccrg-prague-congestion-control-00
  - only published yesterday
  - answers FAQs

- Transport-protocol- and OS-agnostic
  - plus copious transport- & implementation-specific notes
  - majority: TCP Prague over Linux

- There is not 'One True Prague CC'
  - L4S Prague Requirements [draft-ietf-tsvwg-ecn-l4s-id] are for that
  - This draft describes a reference design with implementation notes
Prague CC: Reality + Promise

- **Maturity Status (paraphrasing from the abstract):**
  - Implementation does not satisfy all the Prague req's (yet)
  - IETF might relax certain req's as an outcome of the process of trying
  - In two cases, research code replaced by placeholders until fully evaluated
    - 1) faster flow start
    - 2) faster additive increase

- **§2: Proven code, defaults**

- **§3: Optional code (under development / evaluation) + ideas**

<table>
<thead>
<tr>
<th>Requirements</th>
<th>Linux code</th>
<th>Linux RFC</th>
<th>Linux mainline</th>
<th>L4Steam git</th>
<th>research private</th>
<th>research opened</th>
<th>none</th>
<th>none (simulated)</th>
</tr>
</thead>
<tbody>
<tr>
<td>L4S-ECN Packet Identification: ECT(1)</td>
<td>mandatory</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reno-friendly on loss</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reno-friendly if blank, ECN bottleneck</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reduce RTT dependence (low RTT dominance)</td>
<td>default off</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Scale-down to fractional window</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Detecting loss in units of time</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Linux code</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Performance improvements</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incooperative TCP control packets</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Faster flow start</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Faster than additive increase</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Continuous additive increase</td>
<td>default on</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reduce RTT dependence (high RTT weakness)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Burst avoidance for TSO sizing &amp; pacing (&lt;1ms)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Performance-bug fixes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integer scaling &amp; fractional carry counting</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PRR undershoot spikes</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Integer rounding &amp; fractional carry counting</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Reno-friendly TCP</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Prague CC – recent work

• Prague CC is not only about adding code to DCTCP for each Prague L4S req't
  • draft corrects this impression
  • subtle but critical changes to:
    – feedback averaging, AI, MD
    – burst limiting / pacing
    – segmentation offload
L4S Drafts – Summary

- Milestone for main drafts: Oct'21; aiming for earlier
- draft-ietf-tsvwg-l4s-arch-08
  - stable, complete, unchanged
- draft-ietf-tsvwg-ecn-l4s-id-14
  - Tom Henderson’s review
  - updates to L4S Transport (Prague) Reqs (§4) & supporting appendix (thx to survey respondents, Asad). Ongoing...
- draft-ietf-tsvwg-dualq-coupled-aqm-14: stable, except...
  - imported L4S Network Reqs by reference [ecn-l4s-id; §5]
  - Ingemar Johansson’s review
- draft-white-tsvwg-l4s-ops-02
  - updated – in adoption call (thx to Bob, Jake, Koen, Olivier, Tom, Asad)
- draft-briscoe-iccrg-prague-congestion-control-00
  - new