Prague requirements survey
based on draft-ietf-tsvwg-ecn-l4s-id-12
updates in draft-ietf-tsvwg-ecn-l4s-id-14

Koen De Schepper, Nokia Bell Labs
Bob Briscoe, independent
TSVWG @ IETF110
March 9, 2021
Survey for Prague Congestion Controls

• Goal: are Prague requirements feasible/realizable and supported by a broad community (allows several different CCs)?

• Template provided:
  • List of all normative requirements
  • List of 3 performance improvement suggestions (no normative text)

• Targeting Congestion Control developers having a Prague CC, or that plan to support L4S using the L4S-ID ECT(1)

• 2 questions asked:

<table>
<thead>
<tr>
<th>Compliant / Partially Compliant / Non-compliant</th>
<th>Explain at what level you (plan to) meet the requirement</th>
<th>Any description/limitations/remarks/explanation related to evaluation, implementation and plans (will implement or will not implement) can be explained here. Any expected or experienced issues and any objections/disagreements to the requirement can be explained and colored appropriately here.</th>
</tr>
</thead>
</table>
Multiple responses received

3 were publicly shared:
• Linux TCP-Prague by L4Steam
• SCReAM by Ingemar Johansson
• GeforceNow by NVIDIA

→ Listed in https://l4steam.github.io/#prague-requirements-compliance

Other responses shared privately:

→ consolidated summary available at:
  https://l4steam.github.io/PragueReqs/Prague_requirements_consolidated.pdf
Compliant/supported or planned by all

Requirements:
• An L4S sender **MUST** set the ECN field to ECT(1) → OS APIs and Kernels need to support it
• **MUST NOT** set ECT(1) unless it complies with ...
• A sender that sets ECT(1) **SHOULD** implement a scalable congestion control
• **MUST** provide feedback of the extent of CE marking ... → Some remaining concerns with Accurate ECN → tcpm
• **MUST** reduce RTT bias ... → Also, more throughput is planned for longer RTTs
• **SHOULD** detect loss by counting in time-based units ...

Non-Normative performance suggestions:
• Setting ECT(1) in TCP Control Packets and Retransmissions
• Faster than Additive Increase
• Faster Convergence at Flow Start

Actions on the draft:
→ OK after minor clarifications
Strong objections on documentation-only reqs

- The specification **MUST** describe in detail ...
- The specification **MUST** define, quantify and justify burst limit approach ...

- Are these documentation requirements really needed?
- How can it be enforced?
- May not be possible (proprietary).

Actions on the draft:
→ These requirements have been removed
Needs experimental data

• SHOULD scale down to fractional congestion window ...

- Not all convinced if it will be a problem on the Internet, and might not implement
- Multiple research implementations exist; others support it or plan to implement

→ Not a safety issue, but would prevent extra latency on L4S-only queues and drop on Coupled-AQMs
→ Propose: Keep SHOULD. Develop further during experiment as needed.

Actions on the draft:
→ Updated based on discussions on the mailing list (further refinement/clarifications)
Needs experimental data

• **MUST** implement monitoring to detect non-L4S ECN AQM...
  - Is detection itself required?
  - Robust detection scheme needs real deployment experience.
  - Combination with delay-based control could minimize potential issues
  - Develop during experiment as needed.

• **SHOULD** be capable to automatically fall back ...
• **MUST** be capable of being replaced (operator action) by a Classic congestion control ...
  - Is "replace" required or can it disable L4S part to reduce to Classic response only
  - On active flows or new flows

→ If L4S Operational guidelines draft is adopted, these requirements will need to be aligned with it.

Actions on the draft:
→ Todo: further refinement/clarifications
Compliant (to intent) by all: Needs Clarification

- MUST react to packet loss in a way that will coexist safely with a TCP Reno congestion control [RFC5681] ...

  - Not clear what it means “coexist safely with a TCP Reno congestion control”
  - Don’t want to be as degraded as Reno for long RTTs

  ➔ Seeking input from WG on correct wording for this requirement e.g. RFC5033
  ➔ Discussion started on the mailing list

Balance between openness to innovations and guidance/recommendations
  ➔ keep open during experiment, not the mechanism but the result is important
  ➔ Practical example in TCP-Prague CC draft
Conclusion

• **Strong objections against “MUST document”** → all removed

• Develop during experiment to determine need and get real live data:
  • Scaling down to fractional windows
  • Classic ECN bottleneck detection → align with L4S-ops if adopted

• Others already have implementations, or req’s are seen as feasible and are planned to be implemented

• Other inputs are still welcome (public or private)
Backup
<table>
<thead>
<tr>
<th>Requirement</th>
<th>Compliance Details</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>An L4S sender <strong>MUST</strong> set the ECN field to ECT(1)</td>
<td>- Compliant or planned&lt;br&gt;- OS APIs and Kernels need to support it&lt;br&gt;(can RFC8311 be used to justify API updates)</td>
<td>None, OK as is</td>
</tr>
<tr>
<td>A sender that sets ECT(1) <strong>SHOULD</strong> implement a scalable congestion control</td>
<td>- Compliant or planned&lt;br&gt;- More clarification needed to align marking rate to throughput</td>
<td>Improve informative text for rate convergence of long flows</td>
</tr>
<tr>
<td><strong>MUST</strong> eliminate RTT ...</td>
<td>- Compliant or planned&lt;br&gt;- Also for longer RTTs more throughput is planned</td>
<td>None, OK as is</td>
</tr>
<tr>
<td><strong>SHOULD</strong> detect loss by counting in time-based units ...</td>
<td>- Compliant or planned</td>
<td>None, OK as is</td>
</tr>
<tr>
<td><strong>MUST NOT</strong> set ECT(1) unless it complies with following ...</td>
<td>- Compliant to this requirement&lt;br&gt;- Comments were on referred requirements</td>
<td>None, OK as is</td>
</tr>
</tbody>
</table>
All agreed (non-normative): Supported or planned

<table>
<thead>
<tr>
<th>Feature</th>
<th>Status</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting ECT(1) in TCP Control Packets and Retransmissions</td>
<td>Supported or planned</td>
<td>RTP/RTCP clarifications will be added</td>
</tr>
<tr>
<td>Faster than Additive Increase</td>
<td>Supported or planned</td>
<td>None, OK as is</td>
</tr>
<tr>
<td>Faster Convergence at Flow Start</td>
<td>Research code exists and planned</td>
<td>None, OK as is</td>
</tr>
</tbody>
</table>
## Questioned and Strong objections

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Question/Objection</th>
<th>Action</th>
</tr>
</thead>
</table>
| The specification **MUST** describe in detail ... | - Is this requirement really needed?  
- How can it be enforced?  
- May not be possible (proprietary). | This requirement is removed                   |
| **SHOULD** scale down to fractional congestion window ... | - Multiple research codes exist  
- Not all convinced if this is needed, others support it and plan to implement  
- Develop during experiment as needed. | Keep **SHOULD**. The need for this requirement should be observed during the experiment |
| limit bursts ...  
The specification **MUST** define, quantify and justify its approach ... | - Normative requirement is mainly documentation related, see above  
- Can more clear guidelines be given? | The normative **MUST** is removed.  
Warning text still present. |
### Clarification needed

<table>
<thead>
<tr>
<th>Requirement</th>
<th>Compliance Details</th>
<th>Remarks</th>
</tr>
</thead>
</table>
| **MUST** provide feedback of the extent of CE marking... | - Compliant  
- Clarification needed for feedback timing and RTT requirements  
- Some remaining concerns with Accurate ECN | - Appropriate feedback timing depends on the proprietary protocol and needs to be tuned to it  
- Remaining concerns about Accurate ECN needs to be dealt with in tcpm |
| **MUST** react to packet loss in a way that will coexist safely with a TCP Reno congestion control [RFC5681]... | - Compliant to the intent  
- Not clear what it means "coexist safely with a TCP Reno congestion control"  
- Don't want to be as degraded as Reno for long RTTs | - Seeking input from WG on clarification to this requirement e.g. RFC5033 |
| **MUST** implement monitoring to detect non_L4S ECN AQM...  
**SHOULD** be capable to automatically fall back...  
**MUST** be capable of being replaced by a Classic congestion control... | - Robust detection scheme needs real deployment experience.  
- Develop during experiment as needed.  
- Combination with delay-based control could minimize potential issues  
- Clarification: is detection itself required? | - If L4S Operational guidelines draft is adopted, these requirements will need to be aligned with it |