Operational Guidance for Deployment of L4S in the Internet
draft-ietf-tsvwg-l4sops-01

Greg White, Editor

TSVWG @ IETF111

July 26, 2021

Contributors include: Bob Briscoe, Jake Holland, Koen De Schepper, Olivier Tilmans, Tom Henderson, Asad Ahmed, Gorry Fairhurst, Sebastian Moeller, Pete Heist

Scope & Status

- Addresses the concerns raised about possible rate-imbalance in shared-queue RFC3168 bottlenecks
  - Guidance for Operators of End-hosts, Operators of Networks, Researchers

- WG Draft Adopted on March 26
  - Draft-00 (May 2021)
  - Draft-01 (July 2021)
  - Not seeking WGLC currently

Summary of Deltas in Draft-01

- Organization of Network Options into:
  - Preferred, Less Preferred, Last Resort
- Addition of AFD (approximate fair dropping)
- Mention of who is impacted (i.e. it isn't the experimenter)
- More discussion of FQ
  - Incl. configuration as "per-user" queuing rather than per-flow
- More detail on recent CE detection experiments (Apple, Akamai, P.Heist) & mention of Roddav paper

Outline

1. Introduction

2. Per-Flow Fairness

3. Flow Queuing Systems

4. Detection of Classic ECN Bottlenecks

    4.1. Recent Studies

    4.2. Future Experiments

5. Operator of an L4S host

    5.1 Server Type

    5.2 Server deployment environment

6. Operator of a Network Employing RFC3168 FIFO Bottlenecks

    6.1 Preferred Options

    6.2 Less Preferred Options

    6.3 Last Resort Options

7. Operator of a Network Employing RFC3168 FQ Bottlenecks

8. Conclusion of the L4S experiment

    8.1. Termination of a successful L4S experiment

    8.2. Termination of an unsuccessful L4S experiment

6. Options for existing RFC3168 networks

- Preferred Options
  - Upgrade bottlenecks to be L4S-aware
  - Configure Non-Coupled Dual Queue

  Addresses fairness imbalance &
  L4S traffic can achieve low delay, low loss

  - Enable Approximate Fair Dropping
  - Replace single-queue 3168 with FQ 3168

  Addresses fairness imbalance &
  L4S traffic can achieve low loss

  - Do Nothing

  Where fairness imbalance is not an issue.
  L4S traffic can achieve low loss

- Less Preferred Options
  - Treat ECT1 as NotECT (several configuration options)

  Addresses fairness imbalance.
  Disables low loss for L4S but not classic.

- Last Resort Options
  - Turn off 3168 support

  Addresses fairness imbalance, but disables
  low loss for L4S & Classic

  - Re-mark ECT1 packets to NotECT

  Addresses fairness imbalance, but disables
  L4S on the remainder of the path

Expectations on Experimental Deployment

- Networks enable L4S support (dual-queue or L4S-aware FQ)

- Application & OS add support for L4S congestion control
  - Receiver congestion feedback available by default
  - Sender behavior default off

- No Flag Day! End-host operators selectively enable L4S on senders
  - "Canary-based" methods:
    - Selectively enable L4S/3168/NotECT on a subset of paths
    - Monitor whether there are positive or negative effects
  - Progressive-deployment (lab tests, limited field tests, large scale field test, etc)

TODOs & Discussion

- Should L4Sops include more guidance on canary-based methods?
- Guidance on short flows:
  - Draft suggests that short flows need not be concerned with the steady-state unfairness issue. How short is short?