RTO considerations in LPWAN
draft-gomez-lpwan-rto-considerations-01

Authors:
Carles Gomez <carlesgo@entel.upc.edu>
Universitat Politècnica de Catalunya (Spain)
Jon Crowcroft <jon.crowcroft@cl.cam.ac.uk>
University of Cambridge (UK)

105th IETF, Montreal, July 26th, 2019
1. Introduction

• Long or very long RTTs in many LPWANs:
  • In ideal scenarios: in the order of seconds or tens of seconds
  • Higher order RTTs: up to several minutes or even more

• RTT (and its variance) in LPWAN, much greater than typical one on
  the Internet
  • Default RTO in TCP, currently: 1 second
  • Default RTO in CoAP: between 2 and 3 seconds

• In LPWAN, RTOs:
  • When using CoAP, for CON messages
  • In SCHC fragmentation (ACK-Always, ACK-on-Error)
2. Status

• Version -00 presented in Prague (IETF 104)
  • Uplink-RTT (U-RTT) analysis
  • Proposal of an algorithm for the RTO

• New version: -01
  • Added terminology: U-RTT, Downlink-RTT (D-RTT)
  • Added D-RTT analysis

• Additionally
  • Preliminary evaluation results of proposed algorithm
3. D-RTT analysis

• Components
  • Wait time until next uplink transmission
    • Depends on app, might be minutes, hours...
    • May be zero for ideal, scheduled uplink transmissions
  • Time since uplink completed until D-RTT completed
    • Basic D-RTT (BD-RTT)
4. Approaches for the RTO

- If delay is not relevant, set the (default) RTO to the highest expected RTT
- If delay is relevant, and higher order RTTs expected:
  - Dual-RTO algorithm

```
if N_THRESH_HIGH
  consecutive RTT samples > THRESH_HIGH_RTT
  +-----+  if N_THRESH_LOW
  | High RTO |-----+  consecutive RTT samples < THRESH_LOW_RTT
  +-----+  | Low RTO |-----+

```
5. Simulation results

- Dual RTO using the TCP RTO in each state
- Scenario with high RTT intervals
- High RTT value known a priori: time between uplink messages
- Improvement depends on the duration of high and low RTT intervals
6. Questions

• Interest in this work?

• Way forward?
  • Different kinds of contributions:
    • Guidance for RTO settings
    • Proposal of an algorithm