TCP over Constrained-Node Networks

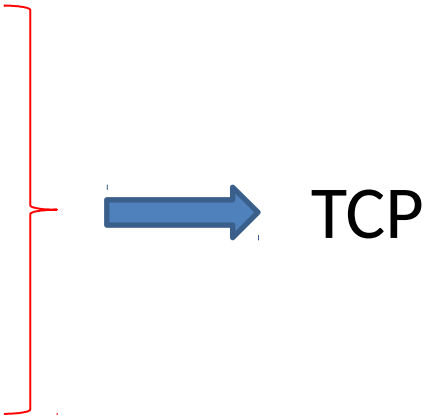
draft-gomez-lwig-tcp-constrained-node-networks-03

Carles Gomez
Universitat Politècnica de Catalunya
Jon Crowcroft
University of Cambridge
Michael Scharf
Nokia

IETF 99 – Prague, July 2017

Motivation

- Several application layer protocols being used for the Internet of Things (IoT)
  - Constrained Application Protocol (CoAP)
    - Originally over UDP
    - CoAP over TCP in progress
      - To overcome middlebox problems
  - HTTP/2 and HTTP/1.1
  - XMPP
  - MQTT



- TCP is being / will be used in many IoT scenarios
  - Offer simple measures for suitable TCP implementation/operation over CNNs

Status

- draft-gomez-core-tcp-constrained-node-networks-00
  - Presented in IETF 96 (LWIG and TCPM WGs)
- draft-gomez-lwig-tcp-...-01
  - Presented in IETF 97
- draft-gomez-lwig-tcp-...-02
  - Presented in IETF 98
- draft-gomez-lwig-tcp-...-03
  - Feedback from IETF 98
  - Details on RIOT and OpenWSN TCP implementations
    - Thanks to Simon Brummer and Xavi Vilajosana

Acknowledgment

- Carsten Bormann, Zhen Cao, Wei Genyu, Ari Keranen, Abhijan Bhattacharyya, Andres Arcia-Moret, Yoshifumi Nishida, Joe Touch, Fred Baker, Nik Sultana, Kerry Lynn, Erik Nordmark, Simon Brummer, Xavi Vilajosana, Rahul Jadhav.

Updates in -03 (I/V)

- 4.2. Maximum Segment Size (MSS)
  - RFC 1981: technologies that support an MTU > 1280 bytes
    - SHOULD support PMTU discovery
      - A minimal IPv6 implementation may choose to omit implementation of PMTU discovery
  - Unless applications require handling large data units (IPv6 datagram size > 1280 bytes)
    - Desirable to limit the MTU to 1280 bytes

Updates in -03 (II/V)

- 4.8. Delayed Acknowledgments
  - Constrained device sending data to a peer
  - If Delayed ACKs are enabled at the peer
    - ACKs may be delayed by, typically, 200 ms
    - Transactional-type traffic
      - Unnecessary delay
      - Disabling Delayed ACKs is recommended
      - Possible if the peer is administered by the same entity managing the CNN

Updates in -02 (III/V)

- 7.3. RIOT TCP implementation
  - Designed for Class 1 devices
  - Targets are 8- and 16-bit microcontrollers
  - Single-MSS window
    - Simplifies implementation
  - By default, only enough memory for a single TCP connection
  - Similar to uIP, but:
    - Memory allocated can be increased to support multiple parallel connections
      - Provides independent buffer for each connection
    - Retransmission handled by TCP

Updates in -02 (IV/V)

- 7.4. OpenWSN
  - Mostly equivalent to uIP implementation
  - Only supports minimum functionality
  - E.g. does not perform retransmissions

Updates in -03 (V/V)

More details welcome!

- Annex

		uIP	lwIP orig	lwIP 2.0	RIOT	OpenWSN	TinyOS
Memory	Data size	*	*	*	*	*	*
	Code size (kB)	< 5	~9 to ~14	*	*	*	*
TCP features	Window size (MSS)	1	Multiple	Multiple	1	1	*
	Slow start	No	Yes	Yes	No	No	*
	Fast rec/retx	No	Yes	Yes	No	No	*
	Keep-alive	No	*	*	No	No	*
	TFO	No	No	*	No	No	*
	ECN	No	No	*	No	No	*
	Window Scale	No	No	Yes	No	No	*
	TCP timestamps	No	No	Yes	No	No	*
	SACK	No	No	Yes	No	No	*
	Delayed ACKs	No	Yes	Yes	No	No	*

WG adoption ?