

Working Group Draft for TCPCLv4

Brian Sipos

RKF Engineering Solutions

IETF102



Overview

- Background
- Current state of TCPCL
- Last Changes to Draft
- Way Forward for TCPCL

Motivations for Updates to TCPCL

1. During implementation of TCPCLv3, Scott Burleigh found an ambiguity in bundle acknowledgment and refusal.
2. For use in a terrestrial WAN, author has a need for TLS-based authentication and integrity. TCPCLv3 mentions TLS but does not specify its use. IETF strongly in favor of TLS for new general-use protocols.
3. Reduced sequencing variability from TCPCLv3
4. Allow an endpoint to positively reject a message (rather than simply ignoring it).
5. Adding extension capability for TCPCL sessions and transfers.

Goals for TCPCLv4

- Do not change scope or workflow of TCPCL.
 - As much as possible, keep existing requirements and behaviors. The baseline spec was a copy-paste of TCPCLv3.
 - Still using single-phase contact negotiation, re-using existing headers and message type codes.
 - Allow existing implementations to be adapted for TCPCLv4.

Last Draft Edits

- Changes are in [draft-ietf-dtn-tcpclv4-09](#).
- Protocol description changes:
 - Added protocol entity and role names (“active” vs “passive” role).
 - Added session and transfer state names and transition diagrams.
- Split contact negotiation from session negotiation.
 - Only CAN_TLS flag is present in contact header.
 - No possibility of information leak when TLS is mandatory.
- Specified contact header and session initialization exchange order.
 - The active role always sends first.
- Added transfer extension capability similar to session extension.
- Removed unnecessary termination message on TLS failure.

Way Forward for TCPCLv4

- Current specification draft is complete
 - All comments to-date have been addressed and many have led to draft edits.
- Working implementation exists and is available for interoperability testing
 - Updated to current I-D content
 - Implemented in scapy/python for ease of understanding
 - Handles concurrent sessions
 - Does not implement BP agent behavior, only CL behavior