Overview

• Background
• Discussion of open questions
• Way forward for TCPCL and BPbis
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, I have a need for TLS-based authentication and integrity. TCPCLv3 mentions TLS but does not specify its use.

3. Reduced sequencing variability from TCPCLv3

4. Allow an endpoint to positively reject a message (rather than simply ignoring it).
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.

- Re-use existing encoding, type and reason codes.
  - Avoid duplication of IANA assignments.
  - Since workflow is preserved, majority of message types are retained.
  - This inherits limitations from TCPCLv3 for the sake simpler implementation changes.
Remaining Open Comments

• Last questions raised on WG mailing list:
  ◦ Should TLS be made mandatory in TCPCLv4?
  ◦ Is it better to re-use existing TCPCLv3 message/reason codes and names, or create new IANA registries with new names? Path from v3 or clean start with consistent naming?
  ◦ What about TCPCL extensibility? Allow more information exchange between TCPCL endpoints?
  ◦ What about neighbor discovery at the CL?
  ◦ There are still a few typographical nits in the current I-D
Security, TLS, DTN, and IETF

• Should TLS be made mandatory in TCPCLv4?

• Author preference: yes. There is no reason for any plaintext on-the-wire, for a public network. Parts of the current CL protocol work-around exposing sensitive info.

• If TLS was part of CL but unwanted for certain users, the requirement can be ignored for a private network.
  ◦ Unsecured TCPCLv4 may be helpful for troubleshooting situations.
Type and Reason Codes

• Current TCLCLv4 spec uses the same IANA registry as TCPCLv3 for:
  ◦ Message type codes
  ◦ Refuse-bundle reason codes
  ◦ Shutdown reason codes

• Additional TCPCLv4 codes:
  ◦ Reject-message reason codes

• What is WG preference: create four new registries exclusively for TCPCLv4 codes and apply more consistent naming scheme?
  ◦ Is there a relevant BCP for this type of thing?
CL Protocol Extensibility

• How can the TCPCL be made adaptable by future applications?

• An earlier TCPCLv4 draft included all negotiation parameters within a type-length-value (TLV) sequence header

• Later drafts reduced the negotiation parameters and removed the TLV encoding in favor of fixed field placements

• There is no reason why a fixed-encoding (non-SDNV) TLV header sequence could not be introduced
  ◦ Current TCPCL would simply leave the sequence empty
  ◦ Type codes could be segmented similar to other protocols: one range for IANA approved, one range for experimental
Neighbor Discovery

• Is there a need and/or desire for neighbor discovery at the Convergence Layer?

• Author comment: what, exactly, is being discovered? Mapping of peer network (IP) address to DTN endpoint identifiers (EIDs)?

• Author preference: no. This belongs at a lower layer where there is some broadcast/multicast capability.
Conclusions

- There has not been any follow-on WG discussion other than the immediate replies to Rick Taylor’s original questions in February.
- Do we have current consensus on the open questions?