

Delay-Tolerant Networking LTP Convergence Layer (LTPCL) Adapter
draft-burleigh-dtnrg-ltpcl-05

Abstract

This document describes the procedures by which the Licklider Transmission Protocol (LTP) is used as a "convergence-layer" protocol to convey Delay-Tolerant Networking (DTN) "bundles" between DTN nodes.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC 2119](#) [[RFC2119](#)].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on October 27, 2013.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents

carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	Specification	3
2.1.	LTP Engine Configuration	4
2.2.	Bundle Transmission	4
2.3.	Bundle Reception	5
3.	IANA Considerations	7
4.	Security Considerations	7
5.	Acknowledgment	7
6.	Normative References	7
	Author's Address	8

[1.](#) Introduction

This document describes the procedures by which the Licklider Transmission Protocol (LTP)) [[RFC5326](#)] is used as a "convergence-layer" protocol to convey Delay-Tolerant Networking (DTN) Bundle Protocol (BP) [[RFC5050](#)] "bundles" between DTN nodes.

BP is designed to enable end-to-end forwarding of a unit of user data, encapsulated in a bundle, from one DTN node to another, possibly indirectly through the agency of other DTN nodes that relay the bundle among themselves toward the destination node. Conveyance of a bundle directly from one node to a second node -- either a relay node or the final destination -- is accomplished by the sending node's invocation of the transmission services of some underlying "convergence-layer" protocol stack.

A virtually limitless variety of convergence-layer stacks may be utilized in support of BP for this purpose, each one achieving inter-node bundle flow in a way that is suited to the particular communications infrastructure to which both the sending and receiving nodes have access.

Convergence-layer stacks are typically characterized by the highest-layer standard protocol in the stack, i.e., the protocol that is immediately below BP, which is commonly called the "convergence-layer protocol." To assure interoperability among nodes that utilize a common convergence-layer protocol, it is necessary to agree on the procedures by which bundles are encapsulated in the protocol data units of the convergence-layer protocol and reconstructed from those

protocol data units upon reception; these procedures are performed by a "convergence-layer adapter" that conforms to a standardized convergence-layer adapter specification. (Note that convergence-layer adapter standardization is necessary for BP node interoperation but is in itself not sufficient: agreement on the configuration of protocols at all layers below the convergence-layer protocol is also necessary. Mechanisms for achieving this agreement are beyond the scope of this document.)

This document, the LTP convergence-layer adapter specification, defines standard procedures for encapsulating bundles in LTP segments and reconstructing bundles from received LTP segments.

2. Specification

In general, LTP operates as follows: an LTP "sender" engine generates one or more data "segments" from a "block" of client service data and conducts a segment transmission "session" that ultimately enables reconstruction of the client service data block, from received segments, at the "receiver" engine.

Each block comprises a "red-part" of zero or more octets, to which reliable transmission procedures must be applied, immediately followed by a "green-part" of zero or more octets, to which only "best efforts" transmission procedures need be applied. The length of the red-part of a block is termed the block's "red length." The length of the green-part of a block is termed the block's "green length." The length of a block is the sum of its red length and green length. Each LTP data segment encapsulates part (or all) of the red-part of a block or part (or all) of the green-part of a block, but never both. LTP data segments that encapsulate red-part data are termed "red segments." LTP data segments that encapsulate green-part data are termed "green segments."

The LTP specification implicitly mandates the reassembly of the red-part of a block into a contiguous byte array from received red segments, but it does not mandate the reassembly of the green-part from received green segments. That is, any required reassembly of "green" service data is the responsibility of the client service - in this case, the LTP convergence-layer adapter.

Specific procedures for invoking LTP to send bundles are as follows.

2.1. LTP Engine Configuration

If the BP node that utilizes a given LTP engine is a member of one or more endpoints identified by CBHE-conformant endpoint IDs, then the engine number of that LTP engine MAY be identical to the node number that is common to those endpoint IDs. This convention can simplify network management in some environments.

Otherwise, the LTP engine number MAY be interpreted as the node number that identifies this node, in any context where some such identifying node number may be useful.

2.2. Bundle Transmission

The client service data block that the LTP convergence-layer adapter (LTPCLA) passes to the LTP sender engine for transmission MUST comprise only one or more complete "conformant" bundles. ("Conformant" bundles are bundles that conform to the Bundle Protocol specification [[RFC5050](#)].) The total length of the block MUST be equal to the sum of the lengths of all bundles in the block.

The client service data block passed to the LTP sender engine by LTPCLA MUST contain exactly one of the following:

- o A single conformant bundle of length L for which only best-efforts transmission is intended. In this case, the red length of the block is zero and the green length of the block is L.
- o A single conformant bundle of length L comprising N bytes ($0 < N < L$) for which reliable transmission is intended followed by (L - N) bytes for which only best-efforts transmission is intended. In this case, the red length of the block is N and the green length of the block is (L - N).
- o A single conformant bundle of length L for which reliable transmission is intended. In this case, the red length of the block is L and the green length of the block is zero.
- o The concatenation of two or more conformant bundles for which reliable transmission is intended, the sum of whose lengths is L. In this case, the red length of the block is L and the green length of the block is zero.

Note that the length of a bundle in a client service data block is not constrained in any way by the lengths of the LTP data segments in which portions of the block will be transmitted.

An LTP sender engine MAY impose an upper limit on the red length of a block. Mechanisms by which LTPCLA may determine the block red length limit are an implementation matter.

The client service ID number passed by LTPCLA to LTP with each block MUST be 1, signifying that the client service is the Bundle Protocol.

Upon reception of a transmission-session completion notice from the LTP receiver engine, LTPCLA MUST inform the bundle protocol agent that data sending procedures with regard to all bundles in the corresponding block have concluded.

Procedures to be performed upon reception of a transmission-session cancellation notice from the LTP receiver engine are an implementation matter, but in particular LTPCLA MUST NOT inform the bundle protocol agent that data sending procedures with regard to all bundles in the corresponding block have concluded. LTPCLA MAY advise the bundle protocol agent to repeat steps 2 through 6 of the Bundle Forwarding procedure defined in [section 5.4 of RFC 5050](#), for each bundle in the block, in this event.

Procedures to be performed by LTPCLA upon reception of transmission-session start notices and initial-transmission completion notices are an implementation matter.

2.3. Bundle Reception

Upon reception of a red-part reception notice from the LTP receiver engine:

- a. The red-part reception notice's client service data is the reassembled red-part of a client service data block.
- b. If the last byte of the red-part is the last byte of the block, LTPCLA MUST initiate bundle reception procedures (complying with [RFC 5050](#)) once for each conformant bundle in the continuous sequence of complete conformant bundles beginning at the first octet of the red-part reception notice's client service data (the reassembled red-part of the block). The total length of a conformant bundle is the sum of the lengths of all BP blocks (not to be confused with LTP blocks) in that bundle, and the length of each conformant BP block can always be determined by inspection of the initial octets of the block. Inability to determine the length of a BP block in the red-part of an LTP block signifies that the immediately prior complete bundle, if any, was the last conformant bundle in the block's red-part; in this case, LTPCLA MUST discard all data in the block following that last conformant bundle.

- c. Otherwise, LTPCLA MUST interpret the red-part as the initial N bytes of a single bundle that is being conveyed via this block, where N is the length of the red-part. LTPCLA SHOULD retain these N bytes of data for the purpose of reconstructing the bundle; alternatively, LTPCLA MAY discard the entire red-part. The latter option is preferable when there is a non-negligible probability that subsequent arrival of green-part segments will not occur in time to prevent block reassembly storage resource consumption from exhausting available storage, as might result from some kinds of denial-of-service attack as discussed in [section 4](#).

Upon reception of a green-part segment arrival notice from the LTP receiver engine:

- a. LTPCLA SHOULD discard all data retained for the purpose of reconstructing the bundle(s) encapsulated in all other blocks for which green data are still unreceived. Green segments nominally arrive in transmission order, and they are never retransmitted, so the arrival of a green segment for one block nominally indicates that no further green segments will be received for any other block that is still pending reassembly.
- b. The "expected offset" for the first green segment received for any block is the red length of the block; the expected offset for every other green segment received for a block is the sum of the offset and length of whichever previously received green segment for that block had the largest offset.
 - * When a green-part segment arrival notice indicates an offset that is less than the expected offset for that segment, the notice's client service data MAY be discarded or it MAY be retained for the purposes of reconstructing the bundle encapsulated in the affected block, an implementation decision.
 - * When a green-part segment arrival notice indicates an offset that is equal to the expected offset for that segment, the notice's client service data MUST be retained for the purposes of reconstructing the bundle encapsulated in the affected block.
 - * When a green-part segment arrival notice indicates an offset that is greater than the expected offset for that segment, the notice's client service data MUST be retained for the purposes of reconstructing the bundle encapsulated in the affected block and the fact that a gap in client service data reception was detected SHOULD be reported if and when the bundle

Burleigh

Expires October 27, 2013

[Page 6]

encapsulated in that block is received. The manner in which this information is reported is an implementation matter.

- c. When a green-part segment arrival notice indicates that the last byte of the notice's client service data is the last byte of the block, LTPCLA MUST initiate bundle reception procedures (complying with [RFC 5050](#)) for the bundle that is encapsulated in this block (regardless of whether or not gaps in the block were detected) and then MUST discard this notice's client service data and all retained data for this block.

Upon reception of a reception-session cancellation notice, LTPCLA MUST discard all data retained for the purpose of reconstructing the bundle conveyed in the affected block.

Procedures to be performed by LTPCLA upon reception of reception-session start notices are an implementation matter.

[3.](#) IANA Considerations

This document has no IANA considerations.

[4.](#) Security Considerations

As noted in [section 2.3](#) above, LTPCLA reception of a partially-red bundle introduces the possibility of a denial-of-service attack. An attacker could transmit an unlimited number of LTP red segments, each of which is flagged as end-of-red-part but not end-of-block, without ever sending any green segments at all. If the attacked node chooses to retain all of these completely received red-parts in hopes of reassembling the original bundles that they are apparently pieces of, without ever receiving any green segments that would cause the retained data to be discarded, it may eventually exhaust its storage resources and cease operation. LTP authentication as documented in [\[RFC5327\]](#) could reduce the chance that such an attack might succeed, making partially-red bundle reception safer.

Otherwise, LTPCLA introduces no new security considerations beyond those discussed in the DTN Bundle Protocol and Licklider Transmission Protocol specifications.

[5.](#) Acknowledgment

Stephen Farrell's and Keith Scott's many helpful observations on this specification are gratefully acknowledged.

[6.](#) Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, [RFC 3986](#), January 2005.
- [RFC5050] Scott, K. and S. Burleigh, "Bundle Protocol Specification", [RFC 5050](#), November 2007.
- [RFC5326] Ramadas, M., Burleigh, S., and S. Farrell, "Licklider Transmission Protocol - Specification", [RFC 5326](#), September 2008.
- [RFC5327] Farrell, S., Ramadas, M., and S. Burleigh, "Licklider Transmission Protocol - Security Extensions", [RFC 5327](#), September 2008.

Author's Address

Scott Burleigh
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, m/s 301-490
Pasadena, CA 91109
USA

Phone: +1 818 393 3353
Email: Scott.C.Burleigh@jpl.nasa.gov

