

DTNRG
Internet-Draft
Intended status: Experimental
Expires: August 5, 2013

H. Kruse
S. Jero
S. Ostermann
Ohio University
Feb 2013

**Datagram Convergence Layers for the DTN Bundle and LTP Protocols
draft-irtf-dtnrg-dgram-clayer-02**

Abstract

This document is a product of the DTNRG Research Group. It specifies the preferred method for transporting DTN protocol data over the Internet using datagrams. The specification covers convergence layers for the Bundle Protocol as well as the transportation of LTP segments. UDP and DCCP are the candidate datagram protocols discussed. UDP can only be used on a local network, or in cases where the DTN node implements explicit congestion control. DCCP addresses the congestion control problem, and its use is recommended whenever possible.

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 August 5, 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](#) [3](#)
- [1.1. Requirements Language](#) [3](#)
- [2. General Recommendation](#) [3](#)
- [3. Recommendations for Implementers](#) [4](#)
- [3.1. How and Where to Deal with Fragmentation](#) [4](#)
- [3.1.1. DCCP](#) [5](#)
- [3.1.2. UDP](#) [5](#)
- [3.2. Bundle Protocol over a Datagram Convergence Layer](#) [5](#)
- [3.2.1. DCCP](#) [5](#)
- [3.2.2. UDP](#) [5](#)
- [3.3. LTP over Datagrams](#) [6](#)
- [3.3.1. DCCP](#) [6](#)
- [3.3.2. UDP](#) [6](#)
- [3.4. Keep Alive Option](#) [6](#)
- [3.5. Checksums](#) [6](#)
- [3.5.1. DCCP](#) [7](#)
- [3.5.2. UDP](#) [7](#)
- [3.6. DCCP Congestion Control Modules](#) [7](#)
- [4. IANA Considerations](#) [7](#)
- [5. Security Considerations](#) [7](#)
- [6. References](#) [8](#)
- [6.1. Normative References](#) [8](#)
- [6.2. Informative References](#) [9](#)
- [Authors' Addresses](#) [10](#)

1. Introduction

Delay/Disruption Tolerant Network (DTN) communication protocols include the Bundle Protocol described in [RFC 5050](#) [[RFC5050](#)], which provides transmission of application data blocks (Bundles) through optional intermediate custody transfer, and the Licklider Transmission Protocol (LTP), RFCs 5325 (LTP Motivation) [[RFC5325](#)], 5326 (LTP Specification) [[RFC5326](#)], and 5327 (LTP Security) [[RFC5327](#)] which can be used to transmit Bundles reliably and efficiently over a point to point link. It is often desirable to test these protocols over Internet Protocol links. [draft-irtf-dtnrg-tcp-clayer](#) [[I-D.irtf-dtnrg-tcp-clayer](#)] defines a method for transporting Bundles over TCP. This draft specifies the preferred method for transmitting either Bundles or LTP blocks across the Internet using datagrams in place of TCP. This document represents the consensus of the DTN RG.

1.1. 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](#)].

2. General Recommendation

In order to utilize DTN protocols across the Internet, whether for testing purposes or as part of a larger network path, it is necessary to encapsulate them into a standard Internet protocol so that they travel easily across the Internet. This is particularly true for LTP, which provides no endpoint addressing. This encapsulation choice needs to be made carefully in order to avoid redundancy, since DTN protocols may provide their own reliability mechanisms.

Congestion control is vital to the continued functioning of the Internet, particularly for situations where data will be sent at arbitrarily fast data rates. The Bundle Protocol delegates provision of reliable delivery and, implicitly, congestion control to the convergence layer used ([Section 7.2 of RFC 5050](#) [[RFC5050](#)]). In situations where TCP will work effectively in communications between pairs of DTN nodes, use of the TCP convergence layer [draft-irtf-dtnrg-tcp-clayer](#) [[I-D.irtf-dtnrg-tcp-clayer](#)] will provide the required reliability and congestion control for transport of Bundles and would be the default choice in the Internet. Alternatives such as encapsulating Bundles directly in datagrams and using UDP or the Datagram Congestion Control Protocol (DCCP) are not generally appropriate because they offer limited reliability and, in the case of UDP, no congestion control.

LTP, on the other hand, offers its own form of reliability. Particularly for testing purposes, it makes no sense to run LTP over a protocol, like TCP, that offers reliability already. In addition, running LTP over TCP would reduce the flexibility available to users, since LTP offers more control over what data is delivered reliably and what data is delivered best effort, a feature that TCP lacks. As such, it would be better to run LTP over an unreliable protocol.

One solution would be to use UDP. UDP provides no reliability, allowing LTP to manage that itself. However, UDP does not provide congestion control. Because LTP is designed to run over fixed rate radio links it does provide rate control, but not congestion control. Lack of congestion control in network connections is a major problem that can cause artificially high loss rates and/or serious fairness issues. Previous standards documents are unanimous in recommending congestion control for protocols to be used on the Internet, see "Congestion Control Principles" [[RFC2914](#)], "Unicast UDP Usage Guidelines" [[RFC5405](#)], and "Queue Management and Congestion Avoidance" [[RFC2309](#)], among others. [RFC 5405](#), in particular, calls congestion control "vital" for "applications that can operate at higher, potentially unbounded data rates". Therefore, any Bundle Protocol implementation permitting the use of UDP to transport LTP segments or Bundles outside an isolated network for the transmission of any non-trivial amounts of data MUST implement congestion control consistent with [RFC 5405](#).

Alternatively, the Datagram Congestion Control Protocol (DCCP) [[RFC4340](#)] was designed specifically to provide congestion control without reliability for those applications that traverse the Internet but do not desire to retransmit lost data. As such, it is RECOMMENDED that, if possible, DCCP be used to transport LTP segments across the Internet.

3. Recommendations for Implementers

3.1. How and Where to Deal with Fragmentation

The Bundle Protocol allows Bundles with sizes limited only by node resource constraints. In IPv4, the maximum size of a UDP datagram is nearly 64KB. In IPv6, when using jumbograms [[RFC2675](#)], UDP datagrams can technically be up to 4GB in size [[RFC2147](#)], although this option is rarely used. It is well understood that sending large IP datagrams that must be fragmented by the network has enormous efficiency penalties [[Kent88](#)]. The Bundle protocol specification provides a Bundle fragmentation concept [[RFC5050](#)] that allows a large Bundle to be divided into Bundle fragments. If the Bundle Protocol is being encapsulated in DCCP or UDP, it therefore SHOULD create each

fragment of sufficiently small size that it can then be encapsulated into a datagram that will not need to be fragmented at the IP layer.

IP fragmentation can be avoided by using IP Path MTU Discovery [[RFC1191](#)][RFC1981], which depends on the deterministic delivery of ICMP Packet Too Big (PTB) messages from routers in the network. To bypass a condition referred to as a black hole [[RFC2923](#)], a newer specification is available in [[RFC4821](#)] to determine the IP Path MTU without the use of PTB messages.

[3.1.1.](#) DCCP

Because DCCP implementations are not required to support IP fragmentation and are not allowed to enable it by default, a DCCP Convergence Layer (we will use "CL" from here on) MUST NOT accept data segments that cannot be sent as one MTU sized datagram.

[3.1.2.](#) UDP

When an LTP CL is using UDP for datagram delivery, it SHOULD NOT create segments that will result in UDP datagrams that will need to be fragmented, as discussed above.

[3.2.](#) Bundle Protocol over a Datagram Convergence Layer

In general, the use of the Bundle protocol over a datagram CL is discouraged in IP networks. Bundles can be of (almost) arbitrary length, and the Bundle protocol does not include an effective retransmission mechanism. Whenever possible the Bundle protocol SHOULD be operated over the TCP Convergence Layer or over LTP.

If a datagram CL is used for transmission of Bundles, every datagram MUST contain exactly one Bundle or four zero octets as a keep-alive. Bundles that are too large for the path MTU SHOULD be fragmented at the Bundle protocol layer to prevent IP fragmentation.

[3.2.1.](#) DCCP

The DCCP CL for Bundle protocol use SHOULD use the IANA assigned port 4556/DCCP and service code 1685351985; the use of other port numbers and service codes is implementation specific.

[3.2.2.](#) UDP

The UDP CL for Bundle protocol use SHOULD use the IANA assigned port 4556/UDP; the use of other port numbers is implementation specific.

3.3. LTP over Datagrams

LTP is designed as a point to point protocol within DTN, and it provides intrinsic acknowledgement and retransmission facilities. LTP segments are transported over a "local data link layer" ([RFC 5325](#) [[RFC5325](#)]); we will use the term "transport" from here on. Transport of LTP using datagrams is an appropriate choice. When a datagram transport is used to send LTP segments, every datagram MUST contain exactly one LTP segment or four zero octets as a keep-alive. LTP MUST perform segmentation in such a way as to ensure that every LTP segment fits into a single packet which will not require IP fragmentation as discussed above.

3.3.1. DCCP

The DCCP transport for LTP SHOULD use the IANA assigned port 1113/DCCP and service code 7107696; the use of other port numbers and service codes is implementation specific.

3.3.2. UDP

The UDP transport for LTP SHOULD use the IANA assigned port 1113/UDP; the use of other port numbers is implementation specific.

3.4. Keep Alive Option

It may be desirable for a UDP or DCCP CL or transport to send "keep-alive" packets during extended idle periods. This may be needed to refresh a contact table entry at the destination, or to maintain an address mapping in a NAT or a dynamic access rule in a firewall. Therefore, the CL or transport MAY send a datagram containing exactly 4 octets of zero bits. The CL or transport receiving such a packet MUST discard this packet; the receiving CL or transport may then perform local maintenance of its state tables, these maintenance functions are not covered in this draft. Note that packets carrying Bundles or segments will always contain more than 4 octets of information (either the Bundle or the LTP header); keep-alive packets will therefore never be mistaken for actual data packets. If UDP or DCCP are being used for communication in both directions between a pair of Bundle agents, transmission and processing of keep-alives in the two directions occurs independently. Keep-alive intervals SHOULD be configurable, SHOULD default to 15 sec, and MUST NOT be configured shorter than 15 sec.

3.5. Checksums

Both the core Bundle protocol specification and core LTP specification assume that they are transmitting over an erasure

channel, i.e. a channel that either delivers packets correctly or not at all.

[3.5.1.](#) DCCP

A DCCP transmitter MUST, therefore, ensure that the entire packet is checksummed by setting the Checksum Coverage to 0. Likewise, the DCCP receiver MUST ignore all packets with partial checksum coverage.

[3.5.2.](#) UDP

A UDP transmitter therefore MUST NOT disable UDP checksums, and the UDP receiver MUST NOT disable checking of received UDP checksums.

Even when UDP checksums are enabled a small probability of UDP packet corruption remains. In some environments it may be acceptable for LTP or the Bundle protocol to occasionally receive corrupted input. In general, however, a UDP implementation SHOULD use optional security extensions available in the Bundle protocol or LTP to protect against message corruption.

[3.6.](#) DCCP Congestion Control Modules

DCCP supports pluggable congestion control modules in order to optimize its behavior to particular environments. The two most common congestion control modules (CCIDs) are TCP-like Congestion Control (CCID2) [[RFC4341](#)] and TCP-Friendly Rate Control (CCID3) [[RFC4342](#)]. TCP-like Congestion Control is designed to emulate TCP's congestion control as much as possible. It is recommended for applications that want to send data as quickly as possible, while TCP-Friendly Rate Control is aimed at applications that want to avoid sudden changes in sending rate. DTN use cases seem to fit more into the first case so DCCP CL's and transports SHOULD use TCP-like Congestion Control (CCID2) by default.

[4.](#) IANA Considerations

Port number assignments 1113/UDP and 4556/UDP have been registered with IANA. Assigned port numbers are 1113/DCCP for the transport of LTP, and 4556/DCCP for the transport of Bundles. Assigned DCCP Service Codes are 7107696 for tunneling LTP and 1685351985 for tunneling Bundle Protocol.

[5.](#) Security Considerations

This memo describes the use of datagrams to transport DTN application

data. Hosts may be in the position of having to accept and process packets from unknown sources; the DTN Endpoint ID can be discovered only after the Bundle has been retrieved from the DCCP or UDP packet. Hosts SHOULD use authentication methods available in the DTN specifications to prevent malicious hosts from inserting unknown data into the application.

Hosts need to listen for and process DCCP or UDP data on the known LTP or Bundle protocol ports. A denial of service scenario exists where a malicious host sends datagrams at a high rate, forcing the receiving hosts to use their resources to process and attempt to authenticate this data. Whenever possible, hosts SHOULD use IP address filtering to limit the origin of packets to known hosts.

6. References

6.1. Normative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [RFC1883] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 1883](#), December 1995.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2147] Borman, D., "TCP and UDP over IPv6 Jumbograms", [RFC 2147](#), May 1997.
- [RFC2675] Borman, D., Deering, S., and R. Hinden, "IPv6 Jumbograms", [RFC 2675](#), August 1999.
- [RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", [RFC 4340](#), March 2006.
- [RFC4341] Floyd, S. and E. Kohler, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 2: TCP-like Congestion Control", [RFC 4341](#), March 2006.
- [RFC5050] Scott, K. and S. Burleigh, "Bundle Protocol Specification", [RFC 5050](#), November 2007.
- [RFC5325] Burleigh, S., Ramadas, M., and S. Farrell, "Licklider Transmission Protocol - Motivation", [RFC 5325](#), September 2008.

- [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.

6.2. Informative References

- [I-D.irtf-dtnrg-tcp-clayer] Demmer, M., Ott, J., and S. Perreault, "Delay Tolerant Networking TCP Convergence Layer Protocol", [draft-irtf-dtnrg-tcp-clayer-05](#) (work in progress), January 2013.
- [Kent88] Kent, C. and J. Mogul, "Fragmentation considered harmful.", 1988, <<http://doi.acm.org/10.1145/55482.55524>>.
- [RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", [RFC 1191](#), November 1990.
- [RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", [RFC 1981](#), August 1996.
- [RFC2309] Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., and L. Zhang, "Recommendations on Queue Management and Congestion Avoidance in the Internet", [RFC 2309](#), April 1998.
- [RFC2914] Floyd, S., "Congestion Control Principles", [BCP 41](#), [RFC 2914](#), September 2000.
- [RFC2923] Lahey, K., "TCP Problems with Path MTU Discovery", [RFC 2923](#), September 2000.
- [RFC4342] Floyd, S., Kohler, E., and J. Padhye, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 3: TCP-Friendly Rate Control (TFRC)", [RFC 4342](#), March 2006.
- [RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", [RFC 4821](#), March 2007.
- [RFC5405] Eggert, L. and G. Fairhurst, "Unicast UDP Usage Guidelines for Application Designers", [BCP 145](#), [RFC 5405](#),

November 2008.

Authors' Addresses

Hans Kruse
Ohio University
292 Lindley Hall
Athens, OH 45701
United States

Phone: +1 740 593 4891
Email: kruse@ohiou.edu

Samuel Jero
Ohio University
Athens, Ohio 45701
United States

Email: sj323707@ohio.edu

Shawn Ostermann
Ohio University
Stocker Engineering Center
Athens, OH 45701
United States

Phone: +1 740 593 1566
Email: ostermann@eecs.ohiou.edu

