

Network Working Group  
Internet-Draft  
Intended status: Informational  
Expires: June 18, 2015

G. Fairhurst, Ed.  
University of Aberdeen  
B. Trammell, Ed.  
ETH Zurich  
December 15, 2014

Services provided by IETF transport protocols and congestion control  
mechanisms  
draft-ietf-taps-transport-00

## Abstract

This document describes services provided by existing IETF protocols and congestion control mechanisms. It is designed to help application and network stack programmers and to inform the work of the IETF TAPS Working Group.

## 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 June 18, 2015.

## Copyright Notice

Copyright (c) 2014 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

Internet-Draft

TAPS Transports

December 2014

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## 1. Introduction

Most Internet applications make use of the Transport Services provided by TCP (a reliable, in-order stream protocol) or UDP (an unreliable datagram protocol). We use the term "Transport Service" to mean an end-to-end facility provided by the transport layer. That service can only be provided correctly if information is supplied from the application. The application may determine the information to be supplied at design time, compile time, or run time and may include guidance on whether an aspect of the service is required, a preference by the application, or something in between. Examples of Transport service facilities are reliable delivery, ordered delivery, content privacy to in-path devices, integrity protection, and minimal latency.

Transport protocols such as SCTP, DCCP, MPTCP, UDP and UDP-Lite have been defined at the transport layer.

In addition, a transport service may be built on top of these transport protocols, using a framework such as WebSockets, or RTP. Service built on top of UDP or UDP-Lite typically also need to specify a congestion control mechanism, such as TFRC or the LEDBAT congestion control mechanism. This extends the set of available Transport Services beyond those provided to applications by TCP and UDP.

Transport services can also be differentiated by the services they provide: for instance, SCTP offers a message-based service that does not suffer head-of-line blocking when used with multiple stream, because it can accept blocks of data out of order, UDP-Lite provides partial integrity protection when used over link-layer services that can support this, and LEDBAT can provide low-priority "scavenger" communication.

## 2. Terminology

The following terms are defined throughout this document, and in subsequent documents produced by TAPS describing the composition and decomposition of transport services.

The terminology below is that as was presented at the TAPS WG meeting in Honolulu. While the factoring of the terminology seems uncontroversial, there may be some entities which still require names (e.g. information about the interface between the transport and lower

layers which could lead to the availability or unavailability of certain transport protocol features)

**Transport Service Feature:** a specific feature a transport service provides to its clients end-to-end. Examples include confidentiality, reliable delivery, ordered delivery, message-versus-stream orientation, etc.

**Transport Service:** a set of transport service features, without an association to any given framing protocol, which provides a complete service to an application.

**Transport Protocol:** an implementation that provides one or more different transport services using a specific framing and header format on the wire.

**Transport Protocol Component:** an implementation of a transport service feature within a protocol

**Transport Service Instance:** an arrangement of transport protocols with a selected set of features and configuration parameters that implements a single transport service, e.g. a protocol stack (RTP over UDP)

**Application:** an entity that uses the transport layer for end-to-end delivery data across the network.

### [3.](#) Transport Protocols

This section provides a list of known IETF transport protocol and transport protocol frameworks.

#### [3.1.](#) Transport Control Protocol (TCP)

TCP [[RFC0793](#)] provides a bidirectional byte-oriented stream over a connection-oriented protocol. The protocol and API use the byte-

stream model.

[EDITOR'S NOTE: Mirja Kuehlewind signed up as contributor for this section.]

### [3.1.1.](#) Multipath TCP (MPTCP)

### [3.2.](#) Stream Control Transmission Protocol (SCTP)

SCTP [[RFC4960](#)] provides a bidirectional set of logical unicast streams over one a connection-oriented protocol. The protocol and API use messages, rather than a byte-stream. Each stream of messages

is independently managed, therefore retransmission does not hold back data sent using other logical streams.

[EDITOR'S NOTE: Michael Tuexen and Karen Nielsen signed up as contributors for these sections.]

#### [3.2.1.](#) Partial Reliability for SCTP (PR-SCTP)

PR-SCTP [[RFC3758](#)] is a variant of SCTP that provides partial reliability.

### [3.3.](#) User Datagram Protocol (UDP)

The User Datagram Protocol (UDP) [[RFC0768](#)] provides a unidirectional minimal message-passing transport that has no inherent congestion control mechanisms. The service may be multicast and/or unicast.

[EDITOR'S NOTE: Kevin Fall signed up as contributor for this section.]

#### [3.3.1.](#) UDP-Lite

A special class of applications can derive benefit from having partially-damaged payloads delivered, rather than discarded, when using paths that include error-prone links. Such applications can tolerate payload corruption and may choose to use the Lightweight User Datagram Protocol [[RFC3828](#)]. The service may be multicast and/or unicast.

### [3.4.](#) Datagram Congestion Control Protocol (DCCP)

The Datagram Congestion Control Protocol (DCCP) [[RFC4340](#)] is a bidirectional transport protocol that provides unicast connections of congestion-controlled unreliable messages. DCCP is suitable for applications that transfer fairly large amounts of data and that can benefit from control over the tradeoff between timeliness and reliability.

### [3.5.](#) Realtime Transport Protocol (RTP)

RTP provides an end-to-end network transport service, suitable for applications transmitting real-time data, such as audio, video or data, over multicast or unicast network services, including TCP, UDP, UDP-Lite, DCCP.

[EDITOR'S NOTE: Varun Singh signed up as contributor for this section.]

### [3.6.](#) Hypertext Transport Protocol (HTTP) as a pseudotransport

[RFC3205]

#### [3.6.1.](#) WebSockets

[RFC6455]

## [4.](#) Transport Service Features

Features as derived from the subsections above.

This section is blank for now.

## [5.](#) IANA Considerations

This document has no considerations for IANA.

## [6.](#) Security Considerations

This document surveys existing transport protocols and protocols providing transport-like services. Confidentiality, integrity, and

authenticity are among the features provided by those services. This document does not specify any new components or mechanisms for providing these features. Each RFC listed in this document discusses the security considerations of the specification it contains.

## 7. Contributors

Non-editor contributors of text will be listed here, as in the authors section.

## 8. Acknowledgments

This work is partially supported by the European Commission under grant agreement FP7-ICT-318627 mPlane; support does not imply endorsement. Special thanks to Mirja Kuehlewind for the terminology section and for leading the terminology discussion in Honolulu.

## 9. References

### 9.1. Normative References

- [RFC0791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.

### 9.2. Informative References

- [RFC0768] Postel, J., "User Datagram Protocol", STD 6, [RFC 768](#), August 1980.
- [RFC0793] Postel, J., "Transmission Control Protocol", STD 7, [RFC 793](#), September 1981.
- [RFC0896] Nagle, J., "Congestion control in IP/TCP internetworks", [RFC 896](#), January 1984.
- [RFC1122] Braden, R., "Requirements for Internet Hosts - Communication Layers", STD 3, [RFC 1122](#), October 1989.
- [RFC2018] Mathis, M., Mahdavi, J., Floyd, S., and A. Romanow, "TCP

- Selective Acknowledgment Options", [RFC 2018](#), October 1996.
- [RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", [RFC 3168](#), September 2001.
- [RFC3205] Moore, K., "On the use of HTTP as a Substrate", [BCP 56](#), [RFC 3205](#), February 2002.
- [RFC3390] Allman, M., Floyd, S., and C. Partridge, "Increasing TCP's Initial Window", [RFC 3390](#), October 2002.
- [RFC3758] Stewart, R., Ramalho, M., Xie, Q., Tuexen, M., and P. Conrad, "Stream Control Transmission Protocol (SCTP) Partial Reliability Extension", [RFC 3758](#), May 2004.
- [RFC3828] Larzon, L-A., Degermark, M., Pink, S., Jonsson, L-E., and G. Fairhurst, "The Lightweight User Datagram Protocol (UDP-Lite)", [RFC 3828](#), July 2004.
- [RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", [RFC 4340](#), March 2006.
- [RFC4960] Stewart, R., "Stream Control Transmission Protocol", [RFC 4960](#), September 2007.
- [RFC5348] Floyd, S., Handley, M., Padhye, J., and J. Widmer, "TCP Friendly Rate Control (TFRC): Protocol Specification", [RFC 5348](#), September 2008.
- [RFC5405] Eggert, L. and G. Fairhurst, "Unicast UDP Usage Guidelines for Application Designers", [BCP 145](#), [RFC 5405](#), November 2008.

- [RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", [RFC 5925](#), June 2010.
- [RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", [RFC 5681](#), September 2009.
- [RFC6093] Gont, F. and A. Yourtchenko, "On the Implementation of the TCP Urgent Mechanism", [RFC 6093](#), January 2011.

- [RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent, "Computing TCP's Retransmission Timer", [RFC 6298](#), June 2011.
- [RFC6455] Fette, I. and A. Melnikov, "The WebSocket Protocol", [RFC 6455](#), December 2011.
- [RFC6691] Borman, D., "TCP Options and Maximum Segment Size (MSS)", [RFC 6691](#), July 2012.
- [RFC7323] Borman, D., Braden, B., Jacobson, V., and R. Scheffenegger, "TCP Extensions for High Performance", [RFC 7323](#), September 2014.

#### Authors' Addresses

Godred Fairhurst (editor)  
University of Aberdeen  
School of Engineering, Fraser Noble Building  
Aberdeen AB24 3UE

Email: [gorry@erg.abdn.ac.uk](mailto:gorry@erg.abdn.ac.uk)

Brian Trammell (editor)  
ETH Zurich  
Gloriastrasse 35  
8092 Zurich  
Switzerland

Email: [ietf@trammell.ch](mailto:ietf@trammell.ch)