RTCWeb Working Group Internet-Draft Intended status: Standards Track Expires: September 7, 2012 R. Jesup Mozilla S. Loreto Ericsson M. Tuexen Muenster University of Applied Sciences March 6, 2012

WebRTC Data Channel Protocol draft-jesup-rtcweb-data-protocol-00.txt

Abstract

The Web Real-Time Communication (WebRTC) working group is charged to provide protocols to support for direct interactive rich communication using audio, video, and data between two peers' webbrowsers. This document specifies an actual (minor) protocol for how the JS-layer dataChannel objects provide the data channels between the peers.

This minor protocol has applicability to other bidirectional uses of SCTP.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of $\underline{BCP 78}$ and $\underline{BCP 79}$.

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 <u>http://datatracker.ietf.org/drafts/current/</u>.

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 September 7, 2012.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to <u>BCP 78</u> and the IETF Trust's Legal

Jesup, et al.

Expires September 7, 2012

[Page 1]

Provisions Relating to IETF Documents

(<u>http://trustee.ietf.org/license-info</u>) 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

<u>1</u> . Introduction	<u>3</u>
<u>2</u> . Opening handshake	<u>3</u>
<u>3</u> . Conventions	<u>3</u>
<u>3.1</u> . Terminology	<u>3</u>
<u>4</u> . Control Messages	<u>3</u>
<u>4.1</u> . Adding a Channel	<u>6</u>
<u>4.2</u> . Closing a Channel	<u>6</u>
5. Sending and Receiving Data	<u>7</u>
<u>6</u> . Security Considerations	<u>7</u>
$\underline{7}$. IANA Considerations	7
<u>8</u> . Acknowledgments	<u>7</u>
<u>9</u> . References	7
<u>9.1</u> . Normative References	<u>7</u>
<u>9.2</u> . Informational References	<u>8</u>
Authors' Addresses	<u>8</u>

<u>1</u>. Introduction

The Data Channel Protocol is designed to provide, in the WebRTC context [I-D.ietf-rtcweb-overview], a generic transport service allowing Web Browser to exchange generic data in a bidirectional peer to peer fashion. As discussed in [I-D.jesup-rtcweb-data] the protocol uses Stream Control Transmission Protocol (SCTP) [RFC4960] encapsulated on Datagram Transport Layer Security (DTLS) [RFC6347] to benefit from their transport and security already standardized features.

2. Opening handshake

The opening handshake is based on the multimedia session description exchange that happens between the browsers, typically through a Web Server acting as the signaling service.

The draft [<u>I-D.ietf-mmusic-sctp-sdp</u>] defines the protocol identifier, 'SCTP/DTLS', and defines how to establish an SCTP association over DTLS using the Session Description Protocol (SDP).

The SCTP association is created with the number of streams specified by the application, and if not specified, then it shall default to 16 streams.

It is recommended that additional streams be available dynamically based on [<u>RFC6525</u>].

3. Conventions

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 [RFC2119].

<u>3.1</u>. Terminology

This document uses the following terms:

Association: An SCTP association. Stream: A unidirectional stream of an SCTP association. It is uniquely identified by a stream identifier.

<u>4</u>. Control Messages

Data Channel Control Messages are sent to manage opening

bidirectional channels. A DATA_CHANNEL_OPEN is sent on the Stream that is intended to be used to send in that direction, and a response (DATA_CHANNEL_OPEN_RESPONSE) using the same structure is sent back on the Stream to be used for the other direction, with a reverse_direction_stream entry holding the Stream number the DATA_CHANNEL_OPEN was sent on. This allows association of the Streams that define the bidirectional channel.

Errors are returned by setting the first 16 bits of the msg_type_data field to a non-O value. In this case the original sender of DATA_CHANNEL_OPEN shall close the channel.

0 1 2 3 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 Type = 0 | Reserved|U|B|E| Length TSN Stream Identifier S | Stream Sequence Number n | PPID = Data Channel Control | msg_type | channel_type | flags reverse_direction_stream | reliability_parameters | msg_type_data \ \ 'msq_type' contains: DATA_CHANNEL_OPEN 0 DATA CHANNEL OPEN RESPONSE 1 'channel_type' contains: DATA_CHANNEL_RELIABLE 0 DATA_CHANNEL_RELIABLE_STREAM 1 2 DATA_CHANNEL_UNRELIABLE DATA_CHANNEL_PARTIAL_RELIABLE_REXMIT 3 DATA_CHANNEL_PARTIAL_RELIABLE_TIMED 4 'flags' contains: DATA_CHANNEL_FLAG_OUT_OF_ORDER_ALLOWED 0x0001 all other bits are reserved 'msg_type_data' contains: for DATA_CHANNEL_OPEN: a DOMString label for the data channel for DATA_CHANNEL_OPEN_RESPONSE: a 16-bit value for errors or 0 for no error 'reliability_parameters' contains (as unsigned 16-bit numbers): for DATA_CHANNEL_PARTIAL_RELIABLE_REXMIT: the maximum number of retransmits or 0 for none for DATA_CHANNEL_PARTIAL_RELIABLE_TIMED: the maximum time in ms to attempt to deliver the data or 0 for no attempts.

Figure 1: Data Channel control messages: OPEN and OPEN_RESPONSE Messages

Note that the forward and reverse-direction Stream numbers may be different if both sides attempt to open the same Stream number at the same time, and the protocol does not require they be the same.

Control messages shall be sent on the relevant Streams, and shall be sent using reliable, in-order delivery.

4.1. Adding a Channel

When one side wants to add a channel, it picks an unused Stream, and if need be, negotiates an increase in the number of Streams available. It then sends a DATA_CHANNEL_OPEN control message on the Stream, with reverse_direction_stream set to 0, though that value MUST be ignored by the recipient.

When a DATA_CHANNEL_OPEN is received on a Stream, an unused Stream is picked, and if need be, negotiates an increase in the number of Streams available. A DATA_CHANNEL_OPEN_RESPONSE is sent on the Stream, with reverse_direction_stream set to the Stream the DATA_CHANNEL_OPEN request came in on.

When a DATA_CHANNEL_OPEN_RESPONSE is received, the reverse_direction_stream value is matched against all pending DATA_CHANNEL_OPENS. If no match can be found, the DATA_CHANNEL_OPEN_RESPONSE should be ignored. If a match is found, then if the error value is 0 the bidirectional Data Channel is now established and ready for use. If the error value is non-zero, the open failed, and the originator should close down the originallyselected Stream and notify the application.

An alternative to embedding the control messages on the Streams used for data transfer would be to put them on a reserved Stream (0 for example), and add an additional field for the forward_direction_stream.

The channel_type and reliability_parameters fields of the DATA_CHANNEL_OPEN message MUST be used to set up the reverse side of the Data Channel so that both directions use the same options.

<u>4.2</u>. Closing a Channel

Data Channels shall be closed by doing an SCTP Stream Sequence Number reset, which should guarantee that all data sent before closing is delivered or has been abandoned (for partially-reliable Data Channels).

Internet-Draft

data protocol P2P in RTCWEB

<u>5</u>. Sending and Receiving Data

Data shall be sent using PPID's other than the Data Channel Control PPID. These PPID's should be registered with IANA via (TBD). The meaning of these data PPIDs and the format of the data shall be specific to the usage of this protocol, and typically shall be provided to the higher layers to allow proper decoding of the data.

For WebRTC, data PPID's for DOMStrings and binary data blobs shall be created.

All data sent on a Data Channel in both directions MUST be sent over the underlying Stream using the reliability defined when the Data Channel was opened.

It is recommended that message size be kept within certain bounds (TBD).

<u>6</u>. Security Considerations

To be done.

7. IANA Considerations

This document also defines three new SCTP Payload Protocol Identifiers (PPIDs). <u>RFC 4960</u> [<u>RFC4960</u>] creates the registry from which these identifiers have been assigned. The following values have been reserved:

WebRTC Control - #To Be Assigned DOMString - #To Be Assigned Binary Data - #To Be Assigned

8. Acknowledgments

Many thanks for comments, ideas, and text ... (TBD)

9. References

<u>9.1</u>. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", <u>BCP 14</u>, <u>RFC 2119</u>, March 1997.

[I-D.ietf-mmusic-sctp-sdp]

Loreto, S. and G. Camarillo, "Stream Control Transmission Protocol (SCTP)-Based Media Transport in the Session Description Protocol (SDP)", <u>draft-ietf-mmusic-sctp-sdp-00</u> (work in progress), July 2011.

- [RFC6347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security Version 1.2", <u>RFC 6347</u>, January 2012.
- [RFC6525] Stewart, R., Tuexen, M., and P. Lei, "Stream Control Transmission Protocol (SCTP) Stream Reconfiguration", RFC 6525, February 2012.

<u>9.2</u>. Informational References

```
[I-D.ietf-rtcweb-overview]
Alvestrand, H., "Overview: Real Time Protocols for Brower-
based Applications", <u>draft-ietf-rtcweb-overview-02</u> (work
in progress), September 2011.
```

[I-D.jesup-rtcweb-data]

Jesup, R., Loreto, S., and M. Tuexen, "RTCWeb Datagram Connection", <u>draft-jesup-rtcweb-data-01</u> (work in progress), October 2011.

Authors' Addresses

Randell Jesup Mozilla USA

Email: randell-ietf@jesup.org

Salvatore Loreto Ericsson Hirsalantie 11 Jorvas 02420 Finland

Email: salvatore.loreto@ericsson.com

Michael Tuexen Muenster University of Applied Sciences Stegerwaldstrasse 39 Steinfurt 48565 Germany

Email: tuexen@fh-muenster.de