MMUSIC
Internet-Draft
Intended status: Standards Track
Expires: July 25, 2013

S. Loreto
G. Camarillo
Ericsson

January 21, 2013

Stream Control Transmission Protocol (SCTP)-Based Media Transport in the Session Description Protocol (SDP)

draft-ietf-mmusic-sctp-sdp-03

Abstract

SCTP (Stream Control Transmission Protocol) is a transport protocol used to establish associations between two endpoints. This document describes how to express media transport over SCTP in SDP (Session Description Protocol). This document defines the 'SCTP', 'SCTP/DTLS' and 'DTLS/SCTP' protocol identifiers for SDP.

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 July 25, 2013.

Table of Contents

1. Introduction

SDP (Session Description Protocol) [RFC4566] provides a general-purpose format for describing multimedia sessions in announcements or invitations. RFC4145 [RFC4145] specifies a general mechanism for describing and establishing TCP (Transmission Control Protocol) streams. RFC 4572 [RFC4572] extends RFC4145 [RFC4145] for describing TCP-based media streams that are protected using TLS (Transport Layer Security) [RFC5246].

This document defines a new protocol identifier, 'SCTP', to describe SCTP-based [RFC4960] media streams. Additionally, this document specifies the use of the 'setup' and 'connection' SDP attributes to establish SCTP associations. These attributes were defined in RFC4145 [RFC4145] for TCP. This document discusses their use with SCTP.

Additionally this document defines two new protocol identifiers:

SCTP/DTLS : to allow the usage of the Datagram Transport Layer Security (DTLS) [RFC4347] protocol over SCTP, as specified in [RFC6083], using SDP. DTLS over SCTP provides communications privacy for applications that use SCTP as their transport protocol.

DTLS/SCTP : to allow the usage of SCTP on top of the Datagram Transport Layer Security (DTLS) protocol, as defined in [I-D.tuexen-tsvwg-sctp-dtls-encaps], using SDP. SCTP over DTLS is used by the RTCWeb protocol suite for transporting non- media data between browsers.

The authentication certificates are interpreted and validated as defined in RFC4572 [RFC4572]. Self-signed certificates can be used securely, provided that the integrity of the SDP description is assured as defined in RFC4572 [RFC4572].

TLS is designed to run on top of a byte-stream oriented transport protocol providing a reliable, in-sequence delivery like TCP. Since no-one so far has implemented SCTP over TLS, due to some serious limitations described in [RFC6083], this document does not make use of TLS over SCTP as described in RFC3436 [RFC3436].

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as

described in BCP 14, RFC 2119 [RFC2119] and indicate requirement levels for compliant implementations.

3. Protocol Identifier

The following is the format for an 'm' line, as specified in RFC4566 [RFC4566]:

```
m=<media> <port> <proto> <fmt> ...
```

This document defines three new values for the 'proto' field: 'SCTP', 'SCTP/DTLS' and 'DTLS/SCTP'.

The 'SCTP', 'SCTP/DTLS' and 'DTLS/SCTP' protocol identifiers are similar to both the 'UDP' and 'TCP' protocol identifiers in that they only describe the transport protocol and not the upper-layer protocol.

Media described using an 'm' line containing the 'SCTP' protocol identifier are carried using SCTP [RFC4960].

The 'SCTP/DTLS' protocol identifier indicates that the media described will use the Datagram Transport Layer Security (DTLS) [RFC4347] over SCTP as specified in [RFC6083].

The 'DTLS/SCTP' protocol identifier indicates that the media described will use SCTP on top of the Datagram Transport Layer Security (DTLS) protocol as specified in [I-D.tuexen-tsvwg-sctp-dtls-encaps].

An 'm' line that specifies 'SCTP' or 'SCTP/DTLS' or 'DTLS/SCTP' MUST further qualify the application-layer protocol using an fmt identifier.

An 'm' line that specifies 'SCTP/DTLS' or 'DTLS/SCTP' MUST further provide a certificate fingerprint. An SDP attribute (an 'a' line) is used to transport and exchange end point certificate. The authentication certificates are interpreted and validated as defined in [RFC4572].

4. Media Formats

The SDP specification, [RFC4566], states that specifications defining new proto values, like the SCTP, SCTP/DTLS and DTLS/SCTP proto values defined in this RFC, must define the rules by which their media format (fmt) namespace is managed. Use of an existing MIME subtype

for the format is encouraged. If no MIME subtype exists, it is RECOMMENDED that a suitable one is registered through the IETF process [RFC4288] [RFC4289] by production of, or reference to, a standards-track RFC that defines the transport protocol for the format.

4.1. datachannels and m-line

SCTP defines a stream as an unidirectional logical channel existing within an SCTP association one to another SCTP endpoint. The streams are used to provide the notion of in-sequence delivery. Each user message is sent on a particular stream, either order or unordered. Ordering is preserved only for all ordered messages sent on the same stream.

A Data Channel may be:

unidirectional: formed by one single incoming or outgoing SCTP stream

bidirectional: a pair of one incoming stream and one outgoing SCTP stream

Using the format part of the m line for negotiating datachannel allows the intermediaries nodes to become aware of the kind of traffic actually exchanged on the datachannels, as well as the initial set of datachannels established within the association.

This document allows only one media format on top of an association. That means that all the datachannels must use the same media format.

[NOTE] The exact definition of datachannel depends on the actual media type using it.

4.2. Media Descriptions

The media description change slightly depending on the actual <proto>.

If the <proto> sub-field is 'SCTP' or 'SCTP/DTLS'

the <port> is the SCTP transport port and follows the same active/passive offer/answer model described in Section 4.1 of [RFC4145];

the <fmt> sub-field carries the same port number value specified in the <port> and the mandatory "a=sctpmap:" attribute contains the actual media format within the protocol parameter.

```
m=application 54111 SCTP/DTLS 54111
a=sctpmap:54111 t38 1
```

Running SCTP over DTLS make possible to have multiple SCTP associations on top of the same DTLS connection; each SCTP association make use of a distinct port number that is mainly used to demultiplex the associations.

If the <proto> sub-field is 'DTLS/SCTP'

the <port> is the UDP transport port;

the <fmt> sub-field carries the SCTP port number and the mandatory "a=sctpmap:" attribute contains the actual media format within the protocol parameter.

When a list of port number identifiers is given, this implies that all of these associations MUST run on top of the same DTLS connection. For the payload type assignments the "a=sctpmap:" attribute (see Section 5.1) SHOULD be used to map from a port number to a media encoding name that identifies the payload format transported by the association or the actual application protocol running on top of it.

```
m=application 54111 DTLS/SCTP 5000 5001 5002
c=IN IP4 79.97.215.79
a=sctpmap:5000 webrtc-datachannel 16
a=sctpmap:5001 bfcp 2
a=sctpmap:5002 t38 1
```

4.3. Predefinition of Data Channels

An 'm' line that specifies 'SCTP' or 'SCTP/DTLS' or 'DTLS/SCTP' MAY further provide the definition of channels running within the association. When provided the channel definition MUST include the "a=stream:" attribute (see Section 5.2) indicating the stream number used to form the channel, the "a=label:" attribute (see Section 5.3) indicating the name of the channel and the "a=subprotocol:" attribute (see Section 5.4) indicating which protocol the client would like to speak on the channel.

By default a channel is defined as 'reliable', however it is possible specify a channel as Partial Reliable indicating that the messages will not be retransmitted more times than specified in the "a=max_retr:" attribute (see Section 5.5) or indicating messages

might not be transmitted or retransmitted after a specified life-time given in milli-seconds in the "a=max_time:" attribute (see Section 5.6).

By default a channel is defined as 'ordered' (i.e. within a stream, an endpoint MUST deliver DATA chunks received to the upper layer according to the order of their Stream Sequence Number), however it is possible specify a channel as Unordered using the "a=unordered" attribute (see Section 5.7).

```
          m=application 54111 DTLS/SCTP 5000 5001 5002
          c=IN IP4 79.97.215.79
          a=sctpmap:5000 webrtc-DataChannel 2
          a=sctpmap:5001 bfcp 1
          a=sctpmap:5002 t38 1
          a=webrtc-DataChannel:5000 stream=1;label="channel
1";subprotocol="chat";
          a=webrtc-DataChannel:5000 stream=2;label="channel
2";subprotocol="file transfer";max_retr=3
```

5. Media attributes

5.1. sctpmap Attribute

The sctpmap attribute maps from a port number (as used in an "m=" line) to an encoding name denoting the payload format to be used on top of the SCTP association or the actual protocol running on top of it. It also can provide the number of streams to be supported by the association. If this attribute is not present, the implementation should provide a default, with a suggested value of 16.

```
    sctpmap-attr        =  "a=sctpmap:" sctpmap-number  protocol  [streams]
    sctpmap-number      =  1*DIGIT
    protocol            =  labelstring
      labelstring         =  text
      text                =  byte-string
    streams      =  1*DIGIT
```

5.2. stream Attribute

The 'stream' attribute indicates the actual stream number within the association used to form the channel.

```
stream-attr       =  "a=stream=" streamnumber
streamnumber      =  1*DIGIT
```

5.3. label Attribute

The 'label' attribute indicates the name of the channel. It represents a label that can be used to distinguish, in the context of the WebRTC API, an RTCDataChannel object from other RTCDataChannel objects.

```
label-attr       =  "a=label=" labelstring
labelstring      =  text
text             =  byte-string
```

5.4. subprotocol Attribute

The 'subprotocol' attribute indicates which protocol the client would like to speak on the channel.

```
subprotocol-attr       =  "a=subprotocol=" labelstring
labelstring            =  text
text                   =  byte-string
```

5.5. max_retr Attribute

The 'max_retr' attribute indicates the max times an user message will be retransmitted.

```
maxretr-attr       =  "a=maxretr=" maxretrvalue
maxretrvalue       =  1*DIGIT
```

5.6. max_time Attribute

An user messages might not be transmitted or retransmitted after a specified life-time given in milli-seconds in the 'max_time' attribute.

```
maxtime-attr       =  "a=maxtime=" maxtimevalue
maxtimevalue       =  1*DIGIT
```

5.7. unordered Attribute

The 'unordered' attribute indicates that DATA chunks in the channel MUST be dispatched to the upper layer by the receiver without any attempt to reorder.

6. The Setup and Connection Attributes and Association Management

The use of the 'setup' and 'connection' attributes in the context of an SCTP association is identical to the use of these attributes in the context of a TCP connection. That is, SCTP endpoints MUST follow the rules in Sections 4 and 5 of RFC 4145 [RFC4145] when it comes to the use of the 'setup' and 'connection' attributes in offer/answer [RFC3264] exchanges.

The management of an SCTP association is identical to the management of a TCP connection. That is, SCTP endpoints MUST follow the rules in Section 6 of RFC 4145 [RFC4145] to manage SCTP associations. Whether to use the SCTP ordered or unordered delivery service is up to the applications using the SCTP association.

7. Multihoming

An SCTP endpoint, unlike a TCP endpoint, can be multihomed. An SCTP endpoint is considered to be multihomed if it has more than one IP address. A multihomed SCTP endpoint informs a remote SCTP endpoint about all its IP addresses using the address parameters of the INIT or the INIT-ACK chunk (depending on whether the multihomed endpoint is the one initiating the establishment of the association). Therefore, once the address provided in the 'c' line has been used to establish the SCTP association (i.e., to send the INIT chunk), address management is performed using SCTP. This means that two SCTP endpoints can use addresses that were not listed in the 'c' line but that were negotiated using SCTP mechanisms.

During the lifetime of an SCTP association, the endpoints can add and remove new addresses from the association at any point [RFC5061]. If an endpoint removes the IP address listed in its 'c' line from the SCTP association, the endpoint SHOULD update the 'c' line (e.g., by sending a re-INVITE with a new offer) so that it contains an IP address that is valid within the SCTP association.

In some environments, intermediaries performing firewall control use the addresses in offer/answer exchanges to perform media authorization. That is, policy-enforcement network elements do not let media through unless it is sent to the address in the 'c' line.

In such network environments, the SCTP endpoints can only exchange media using the IP addresses listed in their 'c' lines. In these environments, an endpoint wishing to use a different address needs to update its 'c' line (e.g., by sending a re-INVITE with a new offer) so that it contains the new IP address.

It is worth to underline that when using SCTP on top of DTLS, only single homed SCTP associations can be used, since DTLS does not expose any address management to its upper layer.

8. Network Address Translation (NAT) Considerations

SCTP specific features (not present in UDP/TCP), such as the checksum (CRC32c) value calculated on the whole packet (not just the header) or its multihoming capabilities, present new challenges for NAT traversal. [I-D.ietf-behave-sctpnat] describes an SCTP specific variant of NAT, which provides similar features of Network Address and Port Translation (NAPT).

Current NATs do not typically support SCTP. As an alternative to design SCTP specific NATs, Encapsulating SCTP into UDP [I-D.tuexen-sctp-udp-encaps] makes it possible to use SCTP in networks with legacy NAT and firewalls not supporting SCTP.

At the time of writing, the work on NAT traversal for SCTP is still work in progress. Additionally, no extension has been defined to integrate ICE (Interactive Connectivity Establishment) [RFC5768] with SCTP and its multihoming capabilities either. Therefore, this specification does not define how to describe SCTP-over-UDP streams in SDP or how to establish and maintain SCTP associations using ICE. Should these features be specified for SCTP in the future, there will be a need to specify how to use them in an SDP environment as well.

9. Examples

The following examples show the use of the 'setup' and 'connection' SDP attributes. As discussed in Section 6, the use of these attributes with an SCTP association is identical to their use with a TCP connection. For the purpose of brevity, the main portion of the session description is omitted in the examples, which only show 'm' lines and their attributes (including 'c' lines).

9.1. Actpass/Passive

An offerer at 192.0.2.2 signals its availability for an SCTP association at SCTP port 54111. Additionally, this offerer is also

willing to initiate the SCTP association:

```
m=application 54111 SCTP 54111
c=IN IP4 192.0.2.2
a=setup:actpass
a=connection:new
a=sctpmap:54111 t38 1
```

Figure 1

The endpoint at 192.0.2.1 responds with the following description:

```
m=image 54321 SCTP 54321
c=IN IP4 192.0.2.1
a=setup:passive
a=connection:new
a=sctpmap:t54321 t38 1
```

Figure 2

This will cause the offerer (at 192.0.2.2) to initiate an SCTP association to port 54321 at 192.0.2.1.

9.2. Existing Connection Reuse

Subsequent to the exchange in Section 9.1, another offer/answer exchange is initiated in the opposite direction. The endpoint at 192.0.2.1, which now acts as the offerer, wishes to continue using the existing association:

```
m=application 54321 SCTP *
c=IN IP4 192.0.2.1
a=setup:passive
a=connection:existing
```

Figure 3

The endpoint at 192.0.2.2 also wishes to use the existing SCTP association and responds with the following description:

```
m=application 9 SCTP *
c=IN IP4 192.0.2.2
a=setup:active
a=connection:existing
```

Figure 4

The existing SCTP association between 192.0.2.2 and 192.0.2.1 will be

reused.

9.3. SDP description for DTLS Connection

An offerer at 192.0.2.2 signals the availability of a T.38 fax session over SCTP/DTLS.

```
        m=image 54111 DTLS/SCTP 5000
        c=IN IP4 192.0.2.2
        a=setup:actpass
        a=connection:new
        a=fingerprint:SHA-1 \
          4A:AD:B9:B1:3F:82:18:3B:54:02:12:DF:3E:5D:49:6B:19:E5:7C:AB
        a=sctpmap:5000 webrtc-DataChannel 16
        a=webrtc-DataChannel:5000 stream=1;label="channel
1";subprotocol="chat";
        a=webrtc-DataChannel:5000 stream=2;label="channel
2";subprotocol="file transfer"
```

Figure 5

10. Security Considerations

See RFC 4566 [RFC4566] for security considerations on the use of SDP in general. See RFC 3264 [RFC3264], RFC 4145 [RFC4145] and RFC 4572 [RFC4572] for security considerations on establishing media streams using offer/answer exchanges. See RFC 4960 [RFC4960] for security considerations on SCTP in general and [RFC6083] for security consideration using DTLS on top of SCTP. This specification does not introduce any new security consideration in addition to the ones discussed in those specifications.

11. IANA Considerations

This document defines three new proto values: 'SCTP', 'SCTP/DTLS' and 'DTLS/SCTP'. Their formats are defined in Section 3. These proto values should be registered by the IANA under "Session Description Protocol (SDP) Parameters" under "proto".

This document defines two SDP session and media-level attributes:

'sctpmap'. Its format is defined in Section 5.1. This attribute should be registered by IANA under "Session Description Protocol (SDP) Parameters" under "att-field" (both session and media level)".

12. References

12.1. Normative References

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

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC4145] Yon, D. and G. Camarillo, "TCP-Based Media Transport in the Session Description Protocol (SDP)", RFC 4145, September 2005.

[RFC4288] Freed, N. and J. Klensin, "Media Type Specifications and Registration Procedures", BCP 13, RFC 4288, December 2005.

[RFC4289] Freed, N. and J. Klensin, "Multipurpose Internet Mail Extensions (MIME) Part Four: Registration Procedures", BCP 13, RFC 4289, December 2005.

[RFC4347] Rescorla, E. and N. Modadugu, "Datagram Transport Layer Security", RFC 4347, April 2006.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC4572] Lennox, J., "Connection-Oriented Media Transport over the Transport Layer Security (TLS) Protocol in the Session Description Protocol (SDP)", RFC 4572, July 2006.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC5061] Stewart, R., Xie, Q., Tuexen, M., Maruyama, S., and M. Kozuka, "Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration", RFC 5061, September 2007.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[I-D.tuexen-tsvwg-sctp-dtls-encaps] Jesup, R., Loreto, S., Stewart, R., and M. Tuexen, "DTLS Encapsulation of SCTP Packets for RTCWEB", draft-tuexen-tsvwg-sctp-dtls-encaps-01 (work in progress), July 2012.

12.2. Informative References

[RFC3436] Jungmaier, A., Rescorla, E., and M. Tuexen, "Transport Layer Security over Stream Control Transmission Protocol", RFC 3436, December 2002.

[RFC6083] Tuexen, M., Seggelmann, R., and E. Rescorla, "Datagram Transport Layer Security (DTLS) for Stream Control Transmission Protocol (SCTP)", RFC 6083, January 2011.

[RFC5768] Rosenberg, J., "Indicating Support for Interactive Connectivity Establishment (ICE) in the Session Initiation Protocol (SIP)", RFC 5768, April 2010.

[I-D.ietf-behave-sctpnat] Stewart, R., Tuexen, M., and I. Ruengeler, "Stream Control Transmission Protocol (SCTP) Network Address Translation", draft-ietf-behave-sctpnat-04 (work in progress), December 2010.

[I-D.tuexen-sctp-udp-encaps] Tuexen, M. and R. Stewart, "UDP Encapsulation of SCTP Packets", draft-tuexen-sctp-udp-encaps-06 (work in progress), January 2011.

Authors' Addresses

Salvatore Loreto
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Salvatore.Loreto@ericsson.com

Gonzalo Camarillo
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: Gonzalo.Camarillo@ericsson.com