Stream Control Transmission Protocol (SCTP)-Based Media Transport in the
                 Session Description Protocol (SDP)
                    draft-ietf-mmusic-sctp-sdp-07

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 January 5, 2015.

Copyright Notice

Table of Contents

1. Introduction

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

This document defines three new protocol identifiers:

SCTP : to describe SCTP-based [RFC4960] media streams.

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

DTLS/SCTP : to describe media streams transported using SCTP on top
   of the Datagram Transport Layer Security (DTLS) protocol, as
   defined in [I-D.ietf-tsvwg-sctp-dtls-encaps].

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].

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.

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.ietf-tsvwg-sctp-dtls-encaps]. The actual layer below DTLS can be plain UDP or what ICE agrees on (in the case ICE is used to negotiate the actual transport flow). The lower layer used is identified from the elements present inside the m= line block.

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 provide a certificate fingerprint only if the endpoint supports, and is willing to use, a cipher suite with an associated certificate. 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.

An m-line with <proto> of 'SCTP', 'SCTP/DTLS' or 'DTLS/SCTP' always describe a single SCTP association.

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

The 'fmtp' attribute should be used to map from the 'association-usage' (as used in an "m=" line) to the max-message-size parameter

indicating the maximum message size, in bytes, the endpoint is
willing to accept.

The sctp-port attribute specifies the actual sctp port.


```
 m=application 12345 DTLS/SCTP webrtc-datachannel
 a=fmtp:webrtc-datachannel max-message-size=100000
 a=sctp-port 4060
```


4.1. Media Descriptions

An 'm' line containing the 'SCTP', 'SCTP/DTLS' or 'DTLS/SCTP'
protocol identifier has the following syntax:


```
 sctp-m-line = %x6d "="
  ("application" space sctp-port space "SCTP"    space sctp-fmt CRLF) /
  ("application" space sctp-port space "SCTP/DTLS" space sctp-fmt CRLF) /
  ("application" space udp-port  space "DTLS/SCTP" space sctp-fmt CRLF)

 sctp-port = port

 udp-port = port

 sctp-fmt = association-usage

 association-usage = token
```


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 (sctp-port) and follows the same active/
   passive offer/answer model described in Section 4.1 of [RFC4145].

   If the <proto> sub-field is 'DTLS/SCTP', the <port> is the UDP
   transport port (udp-port).

The <fmt> sub-field carries the parameter indicating the conventional
usage of an entire sctp association (association-usage).

association-usage:
   The association-usage token indicates the conventional usage of an
   entire sctp association including its streams (e.g. the pairing of
   certain streams, the protocol carried over certain streams, etc).

This parameter is a required parameter. [TBD a value for the
generic usage as defined in RFC 4960 [RFC4960]].

Any offered association MAY be rejected in the answer, for any
reason. If an association offer is rejected, the offerer and
answerer MUST NOT establish an SCTP association for it. To reject an
SCTP association, the <port> in the answer MUST be set to zero.

4.1.1. sctp-port

```
sctp-port-attr   =  "a=sctp-port=" portnumber
port-number      =  port
```

The sctp-port attribute specifies the actual sctp port. This
attribute is optional and is only meaningful and required if the the
<proto> sub-field is 'DTLS/SCTP'. If the attribute is not present,
the default value is 5000.

4.1.2. max-message-size

```
sctpmap-attr        =  "a=fmtp:" association-usage [max-message-size]
max-message-size    =  "max-message-size" EQUALS 1*DIGIT
```

The 'fmtp' attribute may be used to map from the 'association-usage'
(as used in an "m=" line) to the max-message-size parameter
indicating the maximum message size, in bytes, the endpoint is
willing to accept.

The max-message-size parameter indicates the maximum message size, in
bytes, the endpoint is willing to accept. The peer should assume
that larger message will be rejected by the endpoint, though it is up
to the endpoint decide the appropriate behaviour. A parameter with
value of '0' will signal a best effort attempt, subject to the
current endpoint memory capacity, to handle messages of any size. If
the parameter is not present, the implementation should provide a
default, with a suggested value of 64K.

5. 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.

6. 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.

7. **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 [RFC6951]
   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 establish and maintain SCTP
   associations using ICE. Should this feature be specified for SCTP in
   the future, there will be a need to specify how to use them in an SDP
   environment as well.

8. **Examples**

   The following examples show the use of the 'setup' and 'connection'
   SDP attributes. As discussed in Section 5, 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).

8.1. **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 t38
     c=IN IP4 192.0.2.2
     a=setup:actpass
     a=connection:new
```

                              Figure 1

   The endpoint at 192.0.2.1 responds with the following description:

```
m=application 54321 SCTP t38
c=IN IP4 192.0.2.1
a=setup:passive
a=connection:new
```

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.

8.2. Existing Connection Reuse

Subsequent to the exchange in Section 8.1, 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 54111 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.

8.3. SDP description for SCTP over DTLS Connection

This example shows the usage of SCTP over DTLS.

An offerer at 192.0.2.2 signals the availability of a webrtc-DataChannel session over SCTP/DTLS. The DTLS connection runs on top of port 54111. The sctp association runs on port 5000 (i.e. sctp-port) over DTLS. The maximum message size, in bytes, the endpoint is willing to accept is 100000 (i.e. max-message-size).

```
m=application 54111 DTLS/SCTP webrtc-datachannel
a=fmtp:webrtc-datachannel max-message-size=100000
    a=sctp-port 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
```

Figure 5

The endpoint at 192.0.2.1 responds with the following description:

```
m=application 62442 DTLS/SCTP webrtc-datachannel
    a=fmtp:webrtc-datachannel max-message-size=100000
    a=sctp-port 5000
c=IN IP4 192.0.2.1
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
```

Figure 6

8.4. SDP description for SCTP over DTLS Connection using default values

This example shows the usage of SCTP over DTLS when default values
are used.

An offerer at 192.0.2.2 signals the availability of a webrtc-
DataChannel session over SCTP/DTLS. The DTLS connection runs on top
of port 54111. As the sctp association runs on the default sct-port
number 5000 over DTLS ant the maximum message size, in bytes, the
endpoint is willing to accept is equal to the default value of 64K
both the parameters may be omitted.

Note that as the sctp association is meant to be used to transport
webrtc data channel, the association-usage parameter is present with
the webrtc-datachannel value.

```
m=application 54111 DTLS/SCTP webrtc-datachannel
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
```

                              Figure 7

   The endpoint at 192.0.2.1 responds with the following description,
   with default value for the sctp-port and max-message-size parameters:

```
m=application 62442 DTLS/SCTP webrtc-datachannel
c=IN IP4 192.0.2.1
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
```

                              Figure 8

9. 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.

10. 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".

   The "fmt" value, "association-usage", used with these "proto" is
   required. It is defined in section Section 4.1.

[Note] TBD whether a new registry is necessary to register the
different possible "association-usage" values.

10.1. sctp-port attribute

This document defines a new SDP session and media-level attribute:

sctp-port: Its format is define in section Section 4.1.This
    attribute should be registered by IANA under "Session Description
    Protocol (SDP) Parameters" under "att-field" (both session and
    media level)".

11. Acknowledgments

The authors wish to thank Harald Alvestrand, Randell Jesup, Paul
Kyzivat, Michael Tuexen for their comments and useful feedback.

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", 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.ietf-tsvwg-sctp-dtls-encaps]
               Tuexen, M., Stewart, R., Jesup, R., and S. Loreto, "DTLS
               Encapsulation of SCTP Packets", draft-ietf-tsvwg-sctp-
               dtls-encaps-04 (work in progress), May 2014.

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.

   [RFC6951]   Tuexen, M. and R. Stewart, "UDP Encapsulation of Stream
               Control Transmission Protocol (SCTP) Packets for End-Host
               to End-Host Communication", RFC 6951, May 2013.

   [I-D.ietf-behave-sctpnat]
               Stewart, R., Tuexen, M., and I. Ruengeler, "Stream Control
               Transmission Protocol (SCTP) Network Address Translation",
               draft-ietf-behave-sctpnat-09 (work in progress), September
               2013.

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