AVTEXT Working Group   J. Xia
INTERNET-DRAFT   R. Even
Intended Status: Standards Track   R. Huang
Expires: June 13, 2015   Huawei
                                                                 L. Deng
                                                            China Mobile
                                                       December 10, 2014

RTP/RTCP extension for RTP Splicing Notification
draft-ietf-avtext-splicing-notification-01

Abstract

Content splicing is a process that replaces the content of a main multimedia stream with other multimedia content, and delivers the substitutive multimedia content to the receivers for a period of time. The splicer is designed to handle RTP splicing and needs to know when to start and end the splicing.

This memo defines two RTP/RTCP extensions to indicate the splicing related information to the splicer: an RTP header extension that conveys the information in-band and an RTCP packet that conveys the information out-of-band.

Copyright and License Notice

Table of Contents

1 Introduction 3
1.1 Terminology 3
2 Overview of RTP Splicing Notification 4
3 Conveying Splicing Interval in RTP/RTCP extensions 5
3.1 RTP Header Extension 5
3.2 RTCP Splicing Notification Message 6
4 Reducing Splicing Latency 7
5 Failure Cases 8
6 SDP Signaling 8
6.1 Declarative SDP 9
6.2 Offer/Answer without BUNDLE 9
6.3 Offer/Answer with BUNDLE: All Media are spliced 10
6.4 Offer/Answer with BUNDLE: a Subset of Media are Spliced . . 12
7 Security Considerations 13
8 IANA Considerations 14
8.1 RTCP Control Packet Types 14
8.2 RTP Compact Header Extensions 14
8.3 SDP Grouping Semantic Extension 14
9 Acknowledges 15
10 References 15
10.1 Normative References 15
10.2 Informative References 15
Authors' Addresses 16

1 Introduction

Splicing is a process that replaces some multimedia content with other multimedia content and delivers the substitutive multimedia content to the receivers for a period of time. In some predictable splicing cases, e.g., advertisement insertion, the splicing duration MUST be inside of the specific, pre-designated time slot. Certain timing information about when to start and end the splicing must be first acquired by the splicer in order to start the splicing. This document refers to this information as Splicing Interval.

[SCTE35] provides a method that encapsulates the Splicing Interval inside the MPEG2-TS layer in cable TV systems. But in the RTP splicing scenario described in [RFC6828], the RTP mixer designed as the splicer has to decode the RTP packets and search for the Splicing Interval inside the payloads. The need for such processing increases the workload of the mixer and limits the number of RTP sessions the mixer can support.

The document defines an RTP header extension [RFC5285] used by the main RTP sender to provide the Splicing Interval by including it in the RTP packets.

Nevertheless, the Splicing Interval conveyed in the RTP header extension might not reach the mixer successfully, any splicing un-aware middlebox on the path between the RTP sender and the mixer might strip this RTP header extension.

To increase robustness against such case, the document also defines a new RTCP packet type in a complementary fashion to carry the same Splicing Interval to the mixer.

1.1 Terminology

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].

The terminology defined in "Content Splicing for RTP Sessions" [RFC6828] applies to this document and in addition, we define:

Splicing Interval:

The NTP timestamps for the Splicing-In point and Splicing-Out point per [RFC6828] allowing the mixer to know when to start and end the RTP splicing.

2 Overview of RTP Splicing Notification

According to RTP Splicing draft [RFC6828], a mixer is designed to handle splicing on the RTP layer at the reserved time slots set by the main RTP sender. This implies that the mixer must first know the Splicing Interval from the main RTP sender before it can start splicing.

When a new splicing is forthcoming, the main RTP sender MUST send the Splicing Interval to the mixer. Usually, the Splicing Interval SHOULD be sent more than once to mitigate the possible packet loss. To enable the mixer to get the substitutive content before the splicing starts, the main RTP sender MUST send the Splicing Interval far ahead. For example, the main RTP sender can estimate when to send the Splicing Interval based on the round-trip time (RTT) following the mechanisms in section 6.4.1 of [RFC3550] when the mixer sends RTCP RR to the main sender.

The substitutive sender also needs to learn the Splicing Interval from the main RTP sender in advance, and thus estimates when to transfer the substitutive content to the mixer. The Splicing Interval could be transmitted from the main RTP sender to the substitutive content using some out-of-band mechanisms, the details how to achieve that are beyond the scope of this memo. To ensure the Splicing Interval is valid for both the main RTP sender and the substitutive RTP sender, the two senders MUST share a common reference clock, so the mixer can achieve accurate splicing.

In this document, the main RTP sender uses a couple of NTP-format timestamps, derived from the common reference clock, to indicate when to start and end the splicing to the mixer: the timestamp of the first substitutive RTP packet at the splicing in point, and the timestamp of the first main RTP packet at the splicing out point.

When the substitutive RTP sender gets the Splicing Interval, it must prepare the substitutive stream. The mixer MUST ensure that the RTP timestamp of the first substitutive RTP packet that would be presented to the receivers corresponds to the same time instant as the former NTP timestamp in the Splicing Interval. To enable the mixer to know the first substitutive RTP packet it needs to send, the substitutive RTP sender MUST send the substitutive RTP packet ahead of the Splicing In point, sllowing the mixer to find out the timestamp of this first RTP packet in the substitutive RTP stream, e.g., using a prior RTCP SR message.

When the splicing will end, the mixer MUST ensure that the RTP timestamp of the first main RTP packet that would be presented on the receivers corresponds to the same time instant as the latter NTP

timestamp in the Splicing Interval.

3 Conveying Splicing Interval in RTP/RTCP extensions

This memo defines two backwards compatible RTP extensions to convey the Splicing Interval to the mixer: an RTP header extension and an RTCP splicing notification message.

3.1 RTP Header Extension

The RTP header extension mechanism defined in [RFC5285] can be adapted to carry the Splicing Interval consisting of a couple of NTP-format timestamps.

One variant is defined for this header extension. It carries the 7 octets splicing-out NTP timestamp (lower 24-bit part of the Seconds of a NTP-format timestamp and the 32 bits of the Fraction of a NTP-format timestamp as defined in [RFC5905]), followed by the 8 octets splicing-in NTP timestamp (64-bit NTP-format timestamp as defined in [RFC5905]). The top 8 bits of the splicing-out NTP timestamp are referred from the top 8 bits of the splicing-in NTP timestamp. This is unambiguous, under the assumption that the splicing-out time is after the splicing-in time, and the splicing interval is less than 2^25 seconds.

The format is shown in Figures 1.

```
     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
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |       0xBE    |    0xDE       |           length=4            |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+E
    |   ID  | L=15  | OUT NTP timestamp format - Seconds (bit 8-31) |x
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+t
    |       OUT NTP timestamp format - Fraction (bit 0-31)          |e
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+n
    |       IN  NTP timestamp format - Seconds (bit 0-31)           |s
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+i
    |       IN  NTP timestamp format - Fraction (bit 0-31)          |o
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+n
```

Figure 1: Sample hybrid NTP Encoding Using
the One-Byte Header Format

Note that the inclusion of an RTP header extension will reduce the efficiency of RTP header compression. It is RECOMMENDED that the main

sender begins to insert the RTP header extensions into a number of RTP packets prior to the splicing in, while leaving the remaining RTP packets unmarked.

After the mixer intercepts the RTP header extension and derives the Splicing Interval, it will generate its own stream and SHOULD NOT include the RTP header extension in outgoing packets to reduce header overhead.

Furthermore, whether the in-band NTP-format timestamps are included or not, RTCP splicing notification message, specified in the next section, MUST be sent to provide robustness in case of any splicing-unaware middlebox that might strip RTP header extensions.

3.2 RTCP Splicing Notification Message

In addition to the RTP header extension, the main RTP sender includes the Splicing Interval in an RTCP splicing notification message.

The RTCP splicing notification message is a new RTCP packet type. It has a fix header followed by a couple of NTP-format timestamps:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|V=2|P|reserved |    PT=TBA     |             length            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             SSRC                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             IN NTP Timestamp (most significant word)          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             IN NTP Timestamp (least significant word)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             OUT NTP Timestamp (most significant word)         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             OUT NTP Timestamp (least significant word)        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: RTCP Splicing Notification Message

The RSI packet includes the following fields:

Length: 16 bits

As defined in [RFC3550], the length of the RTCP packet in 32-bit words minus one, including the header and any padding.

SSRC: 32 bits

The SSRC of the Main RTP Sender.

Timestamp: 64 bits

Indicates the wallclock time when this splicing starts and ends. The full-resolution NTP timestamp is used, which is a 64-bit, unsigned, fixed-point number with the integer part in the first 32 bits and the fractional part in the last 32 bits. This format is similar to RTCP Sender Report (Section 6.4.1 of [RFC3550]).

The RTCP splicing notification message can be appended to RTCP SR the main RTP sender generates in compound RTCP packets, and hence follows the compound RTCP rules defined in Section 6.1 in [RFC3550].

If the use of non-compound RTCP [RFC5506] was previously negotiated between the sender and the mixer, the RTCP splicing notification message may be sent as non-compound RTCP packets.

When the mixer intercepts the RTCP splicing notification message, it MAY NOT forward the message to the receivers in order to reduce RTCP bandwidth consumption or to avoid downstream receivers from detecting splicing defined in Section 4.5 in [RFC6828].

4 Reducing Splicing Latency

When splicing starts or ends, the mixer outputs the multimedia content from another sender to the receivers. Given that the receivers must first acquire certain information ([RFC6285] refers to this information as Reference Information) to start processing the multimedia data, either the main RTP sender or the substitutive sender SHOULD provide the Reference Information align with its multimedia content to reduce the delay caused by acquiring the Reference Information. The methods by which the Reference Information is distributed to the receivers is out of scope of this memo.

Another latency element is synchronization caused delay. The receivers must receive enough synchronization metadata prior to synchronizing the separate components of the multimedia streams when splicing starts or ends. Either the main RTP sender or the substitutive sender SHOULD send the synchronization metadata early enough so that the receivers can play out the multimedia in a synchronized fashion. The mechanisms defined in [RFC6051] are RECOMMENDED to be adopted to reduce the possible synchronization delay.

5 Failure Cases

This section examines the implications of losing RTCP splicing notification message and other failure case, e.g., the RTP header extension is stripped on the path.

Given that there may be splicing un-aware middlebox on the path between the main RTP sender and the mixer, one heuristics will be used to verify whether or not the Splicing Interval reaches the mixers.

If the mixer does not get the Splicing Interval when the splicing starts, it will still output the main content to the downstream receivers and forward the RTCP RR packets sent from downstream receivers to the main RTP sender (see section 4.2 of [RFC6828]). In such case, the main RTP sender can learn that splicing failed.

In a similar manner, the substitutive sender can learn that splicing failed if it does not receive any RTCP RR packets from downstream receivers when the splicing starts.

Upon the detection of a failure, the main RTP sender or the substitutive sender SHOULD check the path to the failed mixer, or fallback to the payload specific mechanisms, e.g., MPEG-TS splicing solution defined in [SCTE35].

6 SDP Signaling

This document defines the URI for declaring this header extension in an extmap attribute to be "urn:ietf:params:rtp-hdrext:splicing-interval".

This document extended the standard semantics defined in SDP Grouping Framework [RFC5888] with a new semantic: SPLICE to represent the relationship between the main RTP stream and the substitutive RTP stream. The main RTP sender provides the information about both main and substitutive sources.

The extended SDP attribute specified in this document is applicable for offer/answer content [RFC3264] and do not affect any rules when negotiating offer and answer. When used with multiple media, substitutive RTP MUST be applied only to the RTP packets whose SDP m-line is in the same group with the substitutive stream using FID and has the extended splicing extmap attribute. This semantics is to have splicing applicable for BUNDLE cases.

The following examples show how SDP signaling could be used for splicing in different cases.

6.1 Declarative SDP

```
v=0
o=xia 1122334455 1122334466 IN IP4 splicing.example.com
s=RTP Splicing Example
t=0 0
a=group:SPLICE 1 2
m=video 30000 RTP/AVP 100
i=Main RTP Stream
c=IN IP4 233.252.0.1/127
a=rtpmap:100 MP2T/90000
a=extmap:1 urn:ietf:params:rtp-hdrext:splicing-interval
a=mid:1
m=video 30002 RTP/AVP 100
i=Substitutive RTP Stream
c=IN IP4 233.252.0.2/127
a=sendonly
a=rtpmap:100 MP2T/90000
a=mid:2
```

Figure 3: Example SDP for a single-channel splicing scenario

The mixer receiving the SDP message above receives one MPEG2-TS stream (payload 100) from the main RTP sender (with multicast destination address of 233.252.0.1) on port 30000, and/or receives another MPEG2-TS stream from the substitutive RTP sender (with multicast destination address of 233.252.0.2) on port 30002. But at a particular point in time, the mixer only selects one stream and outputs the content from the chosen stream to the downstream receivers.

6.2 Offer/Answer without BUNDLE

SDP Offer - from main RTP sender

```
v=0
o=xia 1122334455 1122334466 IN IP4 splicing.example.com
s=RTP Splicing Example
t=0 0
a=group:SPLICE 1 2
m=video 30000 RTP/AVP 31 100
i=Main RTP Stream
c=IN IP4 splicing.example.com
a=rtpmap:31 H261/90000
```

```
a=rtpmap:100 MP2T/90000
a=extmap:1 urn:ietf:params:rtp-hdrext:splicing-interval
```



```
a=mid:1
m=video 40000 RTP/AVP 31 100
i=Substitutive RTP Stream
c=IN IP4 substitutive.example.com
a=rtpmap:31 H261/90000
a=rtpmap:100 MP2T/90000
a=sendonly
a=mid:2
```

SDP Answer - from splicer

```
v=0
o=xia 1122334455 1122334466 IN IP4 splicer.example.com
s=RTP Splicing Example
t=0 0
a=group:SPLICE 1 2
m=video 30000 RTP/AVP 100
i=Main RTP Stream
c=IN IP4 splicer.example.com
a=rtpmap:100 MP2T/90000
a=extmap:1 urn:ietf:params:rtp-hdrext:splicing-interval
a=mid:1
m=video 40000 RTP/AVP 100
i=Substitutive RTP Stream
c=IN IP4 splicer.example.com
a=rtpmap:100 MP2T/90000
a=recvonly
a=mid:2
```

Only codecs that are supported both by the main RTP stream and the substitutive RTP stream could be negotiated with SDP O/A. And the mixer MUST choose the same codec for both of these two streams.

6.3 Offer/Answer with BUNDLE: All Media are spliced

In this example, the bundled audio and video media have their own substitutive media for splicing:

1. An Offer, in which the offerer assigns a unique address and a substitutive media to each bundled "m="line for splicing within the BUNDLE group.

2. An answer, in which the answerer selects its own BUNDLE address, and leave the substitutive media untouched.

SDP Offer - from main RTP sender


```
v=0
o=alice 1122334455 1122334466 IN IP4 splicing.example.com
s=RTP Splicing Example
c=IN IP4 splicing.example.com
t=0 0
a=group:SPLICE foo 1
a=group:SPLICE bar 2
a=group:BUNDLE foo bar
m=audio 10000 RTP/AVP 0 8 97
a=mid:foo
b=AS:200
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
a=extmap:1 urn:ietf:params:rtp-hdrext:splicing-interval
m=video 10002 RTP/AVP 31 32
a=mid:bar
b=AS:1000
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
a=extmap:2 urn:ietf:params:rtp-hdrext:splicing-interval
m=audio 20000 RTP/AVP 0 8 97
i=Substitutive audio RTP Stream
c=IN IP4 substitive.example.com
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
a=sendonly
a=mid:1
m=video 20002 RTP/AVP 31 32
i=Substitutive video RTP Stream
c=IN IP4 substitive.example.com
```

```
        a=rtpmap:31 H261/90000
        a=rtpmap:32 MPV/90000
        a=mid:2
```

SDP Answer - from the splicer

```
        v=0
        o=bob 2808844564 2808844564 IN IP4 splicer.example.com
        s=RTP Splicing Example
        c=IN IP4 splicer.example.com
        t=0 0
        a=group:SPLICE foo 1
        a=group:SPLICE bar 2
        a=group:BUNDLE foo bar
        m=audio 30000 RTP/AVP 0
        a=mid:foo
```



```
        b=AS:200
        a=rtpmap:0 PCMU/8000
        a=extmap:1 urn:ietf:params:rtp-hdrext:splicing-interval
        m=video 30000 RTP/AVP 32
        a=mid:bar
        b=AS:1000
        a=rtpmap:32 MPV/90000
        a=extmap:2 urn:ietf:params:rtp-hdrext:splicing-interval
        m=audio 30002 RTP/AVP 0
        i=Substitutive audio RTP Stream
        c=IN IP4 splicer.example.com
        a=rtpmap:0 PCMU/8000
        a=sendonly
        a=mid:1
        m=video 30004 RTP/AVP 32
        i=Substitutive video RTP Stream
        c=IN IP4 splicer.example.com
        a=rtpmap:32 MPV/90000
        a=mid:2
```

6.4 Offer/Answer with BUNDLE: a Subset of Media are Spliced

In this example, the substitutive media only applies for video when splicing:

1. An Offer, in which the offerer assigns a unique address to each bundled "m="line within the BUNDLE group, and assigns a substitutive media to the bundled video "m=" line for splicing.

2. An answer, in which the answerer selects its own BUNDLE address, and leave the substitutive media untouched.

SDP Offer - from the main RTP sender:

```
v=0
o=alice 1122334455 1122334466 IN IP4 splicing.example.com
s=RTP Splicing Example
c=IN IP4 splicing.example.com
t=0 0
a=group:SPLICE bar 2
a=group:BUNDLE foo bar
m=audio 10000 RTP/AVP 0 8 97
a=mid:foo
b=AS:200
a=rtpmap:0 PCMU/8000
a=rtpmap:8 PCMA/8000
a=rtpmap:97 iLBC/8000
m=video 10002 RTP/AVP 31 32
```



```
a=mid:bar
b=AS:1000
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
a=extmap:2 urn:ietf:params:rtp-hdrext:splicing-interval
m=video 20000 RTP/AVP 31 32
i=Substitutive video RTP Stream
c=IN IP4 substitutive.example.com
a=rtpmap:31 H261/90000
a=rtpmap:32 MPV/90000
a=mid:2
```

SDP Answer - from the splicer:

```
v=0
o=bob 2808844564 2808844564 IN IP4 splicer.example.com
s=RTP Splicing Example
c=IN IP4 splicer.example.com
```

```
t=0 0
a=group:SPLICE bar 2
a=group:BUNDLE foo bar
m=audio 30000 RTP/AVP 0
a=mid:foo
b=AS:200
a=rtpmap:0 PCMU/8000
m=video 30000 RTP/AVP 32
a=mid:bar
b=AS:1000
a=rtpmap:32 MPV/90000
a=extmap:2 urn:ietf:params:rtp-hdrext:splicing-interval
m=video 30004 RTP/AVP 32
i=Substitutive video RTP Stream
c=IN IP4 splicer.example.com
a=rtpmap:32 MPV/90000
a=mid:2
```

7 Security Considerations

The security considerations of the RTP specification [RFC3550], the general mechanism for RTP header extensions [RFC5285] and the security considerations of the RTP splicing specification [RFC6828] apply.

The RTP header extension defined in Section 4.1 include two NTP-format timestamps. In the Secure Real-time Transport Protocol (SRTP)[RFC3711], RTP header extensions are authenticated but not encrypted. A malicious endpoint possessing the SRTP key could choose to set the values in this header extension falsely, so as to falsely

claim the splicing time. Also, such a malicious endpoint could cause any arbitrary content it wishes spliced into the main RTP stream.

In scenarios where this is a concern, additional mechanisms MUST be used to protect the confidentiality of the header extension. This mechanism could be header extension encryption [SRTP-ENCR-HDR], or a lower-level security and authentication mechanism such as IPsec [RFC4301].

8 IANA Considerations

8.1 RTCP Control Packet Types

Based on the guidelines suggested in [RFC5226], a new RTCP packet format has been registered with the RTCP Control Packet Type (PT) Registry:

Name: SNM

Long name: Splicing Notification Message

Value: TBA

Reference: This document

8.2 RTP Compact Header Extensions

The IANA has also registered a new RTP Compact Header Extension [RFC5285], according to the following:

Extension URI: urn:ietf:params:rtp-hdrext:splicing-interval

Description: Splicing Interval

Contact: Jinwei Xia <xiajinwei@huawei.com>

Reference: This document

8.3 SDP Grouping Semantic Extension

This document request IANA to register the new SDP grouping semantic extension called "SPLICE".

Semantics: Splice

Token:SPLICE

Reference: This document

Contact: Jinwei Xia <xiajinwei@huawei.com>

9 Acknowledges

TBD

10 References

10.1 Normative References

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

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

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

[RFC5285] Singer, D. and H. Desineni, "A General Mechanism for RTP Header Extensions", RFC 5285, July 2008.

[RFC5888] Camarillo, G. and H. Schulzrinne, "The Session Description Protocol (SDP) Grouping Framework", RFC 5888, June 2010.

[RFC5905] Mills, D., Martin, J., Ed., Burbank, J., and W. Kasch, "Network Time Protocol Version 4: Protocol and Algorithms Specification", RFC 5905, June 2010.

[RFC6051] Perkins, C. and T. Schierl, "Rapid Synchronisation of RTP Flows", RFC 6051, November 2010.

[RFC6828] Xia, J., "Content Splicing for RTP Sessions", RFC 6828, January 2013.

10.2 Informative References

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, May 2008.

[RFC5506] Johansson, I. and M. Westerlund, "Support for Reduced-Size Real-Time Transport Control Protocol (RTCP): Opportunities and Consequences", RFC 5506, April 2009.

[RFC6285] Ver Steeg, B., Begen, A., Van Caenegem, T., and Z. Vax, "Unicast-Based Rapid Acquisition of Multicast RTP Sessions", RFC 6285, June 2011.

[RFC6904] Lennox, J.,"Encryption of Header Extensions in the Secure Real-Time Transport Protocol (SRTP)", April 2013.

[SCTE35] Society of Cable Telecommunications Engineers (SCTE), "Digital Program Insertion Cueing Message for Cable", 2011.

Authors' Addresses

Jinwei Xia
Huawei

Email: xiajinwei@huawei.com

Roni Even
Huawei

Email: ron.even.tlv@gmail.com

Rachel Huang
Huawei

Email: rachel.huang@huawei.com

Lingli Deng
China Mobile

Email: denglingli@chinamobile.com