Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: February 10, 2017

E. Ivov
Jitsi
T. Stach
Unaffiliated
E. Marocco
Telecom Italia
C. Holmberg
Ericsson
August 9, 2016

A Session Initiation Protocol (SIP) usage for Trickle ICE
draft-ietf-mmusic-trickle-ice-sip-05

Abstract

The Interactive Connectivity Establishment (ICE) protocol describes a Network Address Translator (NAT) traversal mechanism for UDP-based multimedia sessions established with the Offer/Answer model. The ICE extension for Incremental Provisioning of Candidates (Trickle ICE) defines a mechanism that allows ICE agents to shorten session establishment delays by making the candidate gathering and connectivity checking phases of ICE non-blocking and by executing them in parallel.

This document defines usage semantics for Trickle ICE with the Session Initiation Protocol (SIP).

Table of Contents

1. Introduction

The Interactive Connectivity Establishment protocol [I-D.ietf-mmusic-rfc5245bis] (a.k.a. Vanilla ICE) describes a mechanism for NAT traversal that consists of three main phases: a phase where an agent gathers a set of candidate transport addresses (source IP address, port and transport protocol), a second phase where these candidates are sent to a remote agent and this gathering procedure is repeated and, finally, a third phase where connectivity between all candidates in both sets is checked (connectivity checks). Once these phases have been completed, and only then, can both agents begin communication. According to the Vanilla ICE specification the three phases above happen consecutively, in a blocking way, which can introduce undesirable latency during session establishment.

The Trickle ICE extension defined in [I-D.ietf-ice-trickle] defines generic semantics required for these ICE phases to happen simultaneously, in a non-blocking way and hence speed up session establishment.

This specification defines a usage of Trickle ICE with the Session Initiation Protocol (SIP)[RFC3261]. It describes how ICE candidates are to be incrementally exchanged with SIP INFO requests and how the Half Trickle and Full Trickle modes defined in [I-D.ietf-ice-trickle] are to be used by SIP User Agents (UAs) depending on their expectations for support of Trickle ICE by a remote agent.

This document defines a new Info Package as specified in [RFC6086] for use with Trickle ICE.

2. 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 [RFC2119].

This specification makes use of all terminology defined by the protocol for Interactive Connectivity Establishment in [I-D.ietf-mmusic-rfc5245bis] and its Trickle ICE extension [I-D.ietf-ice-trickle]. It is assumed that the reader will be familiar with the terminology from both of them.

3. Protocol Overview

The semantics that Vanilla ICE for SIP [I-D.ietf-mmusic-ice-sip-sdp] defines for exchanging ICE candidates are exclusively based on use of Offers and Answers as per [RFC3264] over the Session Description Protocol (SDP) [RFC4566]. This specification extends these mechanism by allowing ICE candidates to also be sent in parallel to the Offer/Answer negotiation or after the completion of Offer/Answer negotiation. This extension is done through the use of SIP INFO messages and a newly defined Info Package [RFC6086].

Typically, in cases where Trickle ICE is fully supported, a candidate exchange would happen along the following lines: The Offerer would send an INVITE containing a subset of candidates and then wait for an early dialog to be established. Once that happens, it will be able to continue sending candidates through in INFO requests and within the same dialog.

Similarly, an Answerer can start or continue "trickling" ICE candidates using INFO messages within the dialog established by its 18x provisional response. Figure 1 shows such a sample exchange:

```
      STUN/Turn                                                STUN/TURN
       Servers          Alice                      Bob          Servers
          |               |                         |                |
          |  STUN Bi.Req. |      INVITE (Offer)      |                |
          |<--------------|------------------------>|                |
          |               |      183 (Answer)       | TURN Alloc Req |
          | STUN Bi.Resp. |<------------------------|--------------->|
          |-------------->|  INFO/OK (SRFLX Cand.)  |                |
          |               |------------------------>| TURN Alloc Resp|
          |               |  INFO/OK (Relay Cand.)  |<---------------|
          |               |<------------------------|                |
          |               |                         |                |
          |               |  More Cands & ConnChecks|                |
          |               |<=======================>|                |
          |               |                         |                |
          |               |          200 OK         |                |
          |               |<------------------------|                |
          |               |            ACK          |                |
          |               |------------------------>|                |
          |               |                         |                |
          |               |    5245 SIP re-INVITE   |                |
          |               |------------------------>|                |
          |               |          200 OK         |                |
          |               |<------------------------|                |
          |               |            ACK          |                |
          |               |------------------------>|                |
          |               |                         |                |
          |               |<===== MEDIA FLOWS =====>|                |
          |               |                         |                |
```

Figure 1: Sample Trickle ICE scenario with SIP

3.1. Rationale - Why INFO?

The decision to use SIP INFO requests as a candidate transport method is based primarily on their lightweight nature. Once a dialog has been established, INFO messages can be exchanged both ways with no restrictions on timing and frequency and no risk of collision.

On the other hand, using Offer/Answer and UPDATE requests, which from an [I-D.ietf-mmusic-ice-sip-sdp] perspective is the traditional way of transporting ICE candidates, introduces the following complications:

Need for a non-blocking mechanism: [RFC3264] defines Offer/Answer as a strictly sequential mechanism. There can only be a maximum of one exchange at any point of time. Both sides cannot simultaneously send Offers nor can they generate multiple Offers

prior to receiving an Answer. Using UPDATEs for candidate transport would therefore imply the implementation of a candidate pool at every agent where candidates can be stored until it is once again that agent's "turn" to emit an Answer or a new Offer. Such an approach would introduce non-negligible complexity for no additional value.

Elevated risk of glare: The sequential nature of Offer/Answer also makes it impossible for both sides to send Offers simultaneously. What's worse is that there are no mechanisms in SIP to actually prevent that. [RFC3261], where the situation of Offers crossing on the wire is described as "glare", only defines a procedure for addressing the issue after it has occurred. According to that procedure both Offers are invalidated and both sides need to retry the negotiation after a period between 0 and 4 seconds. The high likelihood for glare to occur and the average two second back-off intervals would imply Trickle ICE processing duration would not only fail to improve but actually exceed those of Vanilla ICE.

INFO messages decouple the exchange of candidates from the Offer/Answer negotiation and are subject to none of the glare issues described above, which makes them a very convenient and lightweight mechanism for asynchronous delivery of candidates.

Using in-dialog INFO messages also provides a way of guaranteeing that candidates are delivered end-to-end, between the same entities that are actually in the process of initiating a session. The alternative would have implied requiring support for Globally Routable UA URI (GRUU) [RFC5627] which, given GRUUs relatively low adoption levels, would have constituted too strong of constraint to the adoption of Trickle ICE.

3.2. Discovery issues

In order to benefit from Trickle ICE's full potential and reduce session establishment latency to a minimum, Trickle ICE agents need to generate SDP Offers and Answers that contain incomplete, potentially empty sets of candidates. Such Offers and Answers can only be handled meaningfully by agents that actually support incremental candidate provisioning, which implies the need to confirm such support before actually using it.

Contrary to other protocols, like XMPP [RFC6120], where "in advance" capability discovery is widely implemented, the mechanisms that allow this for SIP (i.e., a combination of UA Capabilities [RFC3840] and GRUU [RFC5627]) have only seen low levels of adoption. This presents an issue for Trickle ICE implementations as SIP UAs do not have an

obvious means of verifying that their peer will support incremental candidate provisioning.

The Half Trickle mode of operation defined in the Trickle ICE specification [I-D.ietf-ice-trickle] provides one way around this, by requiring first Offers to contain a complete set of ICE candidates and only using incremental provisioning for the rest of the sessions.

While using Half Trickle does provide a working solution it also comes at the price of increased latency. Section 5 therefore makes several alternative suggestions that enable SIP UAs to engage in Full Trickle right from their first Offer: Section 5.1 discusses the use of on-line provisioning as a means of allowing use of Trickle ICE for all endpoints in controlled environments. Section 5.2 describes anticipatory discovery for implementations that actually do support GRUU and UA Capabilities and Section 5.4 discusses the implementation and use of Half Trickle by SIP UAs where none of the above are an option.

3.3. Relationship with the Offer/Answer Model

It is important to note that this specification does not require, define, or even assume any mechanisms that would have an impact on the Offer/Answer model beyond the way it is already used by Vanilla ICE for SIP [I-D.ietf-mmusic-ice-sip-sdp]. From the perspective of all SIP middle boxes and proxies, and with the exception of the actual INFO messages, signaling in general and Offer/Answer exchanges in particular would look the same way for Trickle ICE as they would for Vanilla ICE for SIP.

```
   +-------------------------------+  +-------------------------------+
   |   Alice      +--------------+ |  | +--------------+       Bob    |
   |              | Offer/Answer | |  | | Offer/Answer |              |
   | +-------+    |    Module    | |  | |    Module    |    +-------+ |
   | |  ICE  |    +--------------+ |  | +--------------+    |  ICE  | |
   | | Agent |           |         |  |         |           | Agent | |
   | +-------+           |         |  |         |           +-------+ |
   +-------------------------------+  +-------------------------------+
         |               |                      |               |
         |               |    INVITE (Offer)    |               |
         |               |--------------------->|               |
         |               |     183 (Answer)     |               |
         |               |<---------------------|               |
         |               |                      |               |
         |                                                      |
         |             SIP INFO (more candidates)               |
         |----------------------------------------------------->|
         |             SIP INFO (more candidates)               |
         |<-----------------------------------------------------|
         |                                                      |
         |          STUN Binding Requests/Responses             |
         |----------------------------------------------------->|
         |          STUN Binding Requests/Responses             |
         |<-----------------------------------------------------|
         |                                                      |
         |               |                      |               |
         |               |  5245 SIP re-INVITE  |               |
         |               |--------------------->|               |
         |               |        200 OK        |               |
         |               |<---------------------|               |
```

Figure 2: Distinguishing between Trickle ICE and traditional signaling.

It is important to note that, as displayed on Figure 2, exchanging candidates through SIP INFO messages are best represented as signaling between ICE agents and not between the traditional SIP and Offer/Answer modules of SIP User Agents. Then, such INFO requests do not impact the state of the Offer/Answer transaction other than providing additional candidates. Consequently, if a new offer is to be send at some point in time it would include the candidates of the previous offer and the candidates that were trickled in the meantime. The version number in the "o=" line of that new offer would need to be incremented by 1 per the rules in [RFC3264].

4. Incremental Signaling of ICE candidates

Trickle ICE agents will construct Offers and Answers as specified in [I-D.ietf-ice-trickle] with the following additional SIP-specific additions:

1. Trickle ICE agents MUST indicate support for Trickle ICE by including the option-tag 'trickle-ice' in a SIP Supported: header field within all SIP INVITE requests and responses.

2. Trickle ICE agents MAY exchange additional ICE candidates using INFO requests within an existing INVITE dialog usage (including an early dialog) as specified in [RFC6086]. The INFO messages carry an Info-Package: trickle-ice. Trickle ICE agents MUST be prepared to receive INFO requests within that same dialog usage, containing additional candidates or an indication for the end of such candidates

3. Trickle ICE agents MAY exchange additional ICE candidates before the Answerer has sent the Answer provided that an invite dialog usage is established at both Trickle ICE agents. Note that in case of forking multiple early dialogs will exist.

The following section provide further details on how Trickle ICE agents establish the INVITE dialog usage such that they can trickle candidates.

4.1. Establishing the dialog

In order for SIP UAs to be able to start trickling, the following two conditions need to be satisfied:

- Trickle ICE support in the peer agent MUST be confirmed.

- The dialog at both sides MUST be in early or confirmed state.

Section 5 discusses in detail the various options for satisfying the first of the above conditions. Regardless of those mechanisms however, agents are certain to have a clear understanding of whether their peers support trickle ICE once an Offer and an Answer have been exchanged, which also allows for ICE processing to commence (see Figure 3).

4.1.1. Asserting dialog state through reliable Offer/Answer delivery

```
                Alice                      Bob
                  |                         |
                  |     INVITE (Offer)      |
                  |------------------------>|
                  |      183 (Answer)       |
                  |<------------------------|
                  |        PRACK/OK         |
                  |------------------------>|
                  |                         |
          +----------------------------------------+
          |Alice and Bob know that both can trickle|
          |and know that the dialog is in the early|
          |state. Send INFO!                       |
          +----------------------------------------+
                  |                         |
                  |  INFO/OK (+SRFLX Cand.) |
                  |------------------------>|
                  |  INFO/OK (+SRFLX Cand.) |
                  |<------------------------|
                  |                         |
```

Figure 3: SIP Offerer can freely trickle as soon as it receives an Answer.

Satisfying both conditions is also relatively trivial for ICE agents that have sent an Offer in an INVITE and that have received an Answer in a reliable provisional response. It is guaranteed to have confirmed support for Trickle ICE within the Answerer (or lack thereof) and to have fully initialized the SIP dialog at both ends. Offerers and Answerers in the above situation can therefore freely commence trickling within the newly established dialog.

4.1.2. Asserting dialog state through unreliable Offer/Answer delivery

The situation is a bit more delicate for agents that have received an Offer in an INVITE request and have sent an Answer in an unreliable provisional response because, once the response has been sent, the Answerer does no know when or if it has been received (Figure 4).

```
                   Alice                      Bob
                     |                          |
                     |     INVITE (Offer)       |
                     |------------------------->|
                     |      183 (Answer)        |
                     |<-------------------------|
                     |                          |
                     |               +-----------------------+
                     |               |Bob:  I don't know if |
                     |               |Alice got my 183 or if|
                     |               |her dialog is already |
                     |               |in the early state.   |
                     |               |  Can I send INFO???  |
                     |               +-----------------------+
                     |                          |
```

Figure 4: A SIP UA that sent an Answer in an unreliable provisional response does not know if it was received and if the dialog at the side of the Offerer has entered the early state

In order to clear this ambiguity as soon as possible, the answerer needs to retransmit the provisional response with the exponential back-off timers described in [RFC3262]. Retransmits MUST cease on receipt of a INFO request or on transmission of the answer in a 2xx response. This is similar to the procedure described in section 13.1.1 of [I-D.ietf-mmusic-ice-sip-sdp] except that the STUN binding Request is replaced by the INFO request.

The Offerer MUST send a Trickle ICE INFO request as soon as it receives an SDP Answer in an unreliable provisional response. This INFO message MUST repeat the candidates that were already provided in the Offer (as would be the case when Half Trickle is performed or when new candidates have not been learned since then) and/or they MAY also deliver new candidates (if available). An end-of-candidates indication MAY be included in case candidate discovery has ended in the mean time.

As soon as an Answerer has received such an INFO request, the Answerer has an indication that a dialog is well established at both ends and MAY begin trickling (Figure 5). Note: The +SRFLX in Figure 5 indicates that additionally newly learned server-reflexive candidates are includes.

```
                Alice                      Bob
                  |                         |
                  |     INVITE (Offer)      |
                  |------------------------>|
                  |      183 (Answer)       |
                  |<------------------------|
                  |  INFO/OK (+SRFLX Cand.) |
                  |------------------------>|
                  |                         |
                  |              +----------------------+
                  |              |Bob:  Now I know Alice|
                  |              | is ready. Send INFO! |
                  |              +----------------------+
                  |  INFO/OK (+SRFLX Cand.) |
                  |<------------------------|
                  |                         |
                  |    200/ACK (Answer)     |
                  |<------------------------|
```

Figure 5: A SIP UA that received an INFO request after sending an unreliable provisional response knows that the dialog at the side of the receiver has entered the early state

When sending the Answer in the 200 OK response, the Answerer MUST repeat exactly the same Answer that was previously sent in the unreliable provisional response in order to fulfill the corresponding requirements in [RFC3264]. In other words, that Offerer needs to be prepared to receive fewer candidates in that repeated Answer than previously exchanged via trickling.

4.1.3. Initiating Trickle ICE without an SDP Answer

The possibility to convey arbitrary candidates in INFO message bodies allows ICE agents to initiate trickling without actually sending an Answer. Trickle ICE Agents MAY therefore respond to INVITEs with provisional responses without an SDP Answer. Such provisional responses serve for establishing an early dialog.

Agents that choose to establish the dialog in this way, MUST retransmit these responses with the exponential back-off timers described in [RFC3262]. Retransmits MUST cease on receipt of an INFO request or on transmission of the answer in a 2xx response. This is again similar to the procedure described in section 12.1.1 of [I-D.ietf-mmusic-ice-sip-sdp] except that an Answer is not yet provided.

```
                   Alice                       Bob
                     |                          |
                     |     INVITE (Offer)       |
                     |------------------------->|
                     |      183 (-)             |
                     |<-------------------------|
                     |  INFO/OK (SRFLX Cand.)   |
                     |------------------------->|
                     |                          |
                     |               +----------------------+
                     |               |Bob:  Now I know again|
                     |               | that Alice is ready. |
                     |               | Send INFO!           |
                     |               +----------------------+
                     |  INFO/OK (SRFLX Cand.)   |
                     |<-------------------------|
                     |    183 (Answer) opt.     |
                     |<-------------------------|
                     |  INFO/OK (SRFLX Cand.)   |
                     |<-------------------------|
                     |    200/ACK (Answer)      |
                     |<-------------------------|
```

Figure 6: A SIP UA sends an unreliable provisional response without an Answer for establishing an early dialog

When sending the Answer the agent MUST repeat all previously sent candidates, if any, and MAY include all newly gathered candidates since the last INFO request was sent. If that answer was sent in a unreliable provisional response, the Answerers MUST repeat exactly the same Answer in the 200 OK response in order to fulfill the corresponding requirements in [RFC3264]. In other words, an Offerer needs to be prepared to receive fewer candidates in that repeated Answer than previously exchanged via trickling.

4.1.4. Considerations for 3PCC

Agents that have sent an Offer in a reliable provisional response and that receive an Answer in a PRACK are also in a situation where support for Trickle ICE is confirmed and the SIP dialog is guaranteed to be in a state that would allow in-dialog INFO requests (see Figure 7).

```
                  Alice                      Bob
                    |                          |
                    |         INVITE           |
                    |------------------------->|
                    |       183 (Offer)        |
                    |<-------------------------|
                    |      PRACK (Answer)      |
                    |------------------------->|
                    |                          |
                    |               +-----------------------+
                    |               |Bob:  I know Alice can|
                    |               |trickle and I know her|
                    |               |dialog is in the early|
                    |               |state. Send INFO!     |
                    |               +-----------------------+
                    |                          |
                    |  INFO/OK (SRFLX Cand.)   |
                    |<-------------------------|
                    |                          |
                    |  INFO/OK (SRFLX Cand.)   |
                    |------------------------->|
                    |      200 OK/ACK          |
                    |<-------------------------|
```

Figure 7: A SIP Offerer in a 3PCC scenario can also freely start trickling as soon as it receives an Answer.

Trickle Agents that send an Offer in a 200 OK and receive an Answer in an ACK can still create a dialog and confirm support for Trickle ICE by sending an unreliable provisional response similar to Section 4.1.3. According to [RFC3261], this unreliable response MUST NOT contain an Offer.

The Trickle Agent (at the UAS) retransmits the provisional response with the exponential back-off timers described in [RFC3262]. Retransmits MUST cease on receipt of a INFO request or on transmission of the answer in a 2xx response. The peer Trickle Agent (at the UAC) MUST send a Trickle ICE INFO request as soon as they receive an unreliable provisional response (see Figure 8).

```
                  Alice                       Bob
                    |                          |
                    |        INVITE            |
                    |------------------------>|
                    |      183 (-)             |
                    |<------------------------|
                    |  INFO/OK (SRFLX Cand.)   |
                    |------------------------>|
                    |                          |
                    |               +-----------------------+
                    |               |Bob:  I know Alice can |
                    |               |trickle and I know her |
                    |               |dialog is in the early |
                    |               |state.                 |
                    |               |INFO can be sent.      |
                    |               +-----------------------+
                    |                          |
                    |  INFO/OK (SRFLX Cand.)   |
                    |<------------------------|
                    |                          |
                    |      200 (Offer)         |
                    |<------------------------|
                    |       ACK (Answer)       |
                    |------------------------>|
                    |                          |
```

Figure 8: A SIP UAC in a 3PCC scenario can also freely start trickling as soon as it receives an unreliable provisional response.

4.2. Delivering candidates in INFO messages

Whenever new ICE candidates become available for sending, agents would encode them in "a=candidate" lines as described by [I-D.ietf-ice-trickle]. For example:

```
        a=candidate:2 1 UDP 1694498815 192.0.2.3 5000 typ srflx
                  raddr 10.0.1.1 rport 8998
```

The use of SIP INFO requests happens within the context of the Info Package as defined Section 10. The MIME type for their payload MUST be set to 'application/trickle-ice-sdpfrag' as defined in Section 9.

Since neither the "a=candidate" nor the "a=end-of-candidates" attributes contain information that would allow correlating them to a specific "m=" line, this is handled through the use of pseudo "m=" lines and identification tags in "a=mid:" attributes as defined in

[RFC5888]. Pseudo "m=" lines follow the SDP syntax for "m=" lines as defined in [RFC4566], but provide no semantics other than indicating to which "m=" line a candidate belongs. Consequently, the receiving agent MUST ignore the remaining content of the pseudo m-line. This guarantees that the 'application/trickle-ice-sdpfrag' bodies do not interfere with the Offer/Answer procedures as specified in [RFC3264].

When sending the INFO request, the agent MAY, if already known to the agent, include the same content into the pseudo m-line as for the corresponding Offer or Answer. However, since Trickle-ICE might be decoupled from the Offer/Answer negotiation this content might be unknown to the agent. In this case, the agent MUST include the following default values.

- o The media is set to 'audio'.
- o The port value is set to '9'.
- o The proto value is set to 'RTP/AVP'.
- o The fmt SHOULD appear only once and is set to '0'

Agents MUST include a pseudo "m=" line and an identification tag in a "a=mid:" attribute for every "m=" line whose candidate list they intend to update. Such "a=mid:" attributes MUST immediately precede the list of candidates for that specific "m=" line. All "a=candidate" or "a=end-of-candidates" attributes following an "a=mid:" attribute, up until (and excluding) the next occurrence of an "a=mid:" attribute, pertain to the "m=" line identified by that identification tag. An "a=end-of-candidates" attribute, preceding any "a=mid:" attributes, indicates the end of all trickling from that agent, as opposed to end of trickling for a specific "m=" line, which would be indicated by a media level "a=end-of-candidates" attribute.

The use of "a=mid:" attributes allows for a structure similar to the one in SDP Offers and Answers where separate media-level and session-level sections can be distinguished. In the current case, lines preceding any "a=mid:" attributes are considered to be session-level. Lines appearing in between or after "a=mid:" attributes will be interpreted as media-level.

> Note that while this specification uses the "a=mid:" attribute from [RFC5888], it does not define any grouping semantics. Consequently, using the "a=group:" attribute from that same specification is neither needed nor used in Trickle ICE for SIP.

All INFO requests MUST carry the "a=ice-pwd:" and "a=ice-ufrag:" attributes that would allow mapping them to a specific ICE

generation. INFO requests containing "a=ice-pwd:" and "a=ice-ufrag:" attributes that do not match those of the current ICE processing session MUST be discarded.

The "a=ice-pwd:" and "a=ice-ufrag:" attributes MUST appear at the same level as the ones in the Offer/Answer exchange. In other words, if they were present as session-level attributes there, they will also appear at the beginning of all INFO message payloads, preceding all "a=mid:" attributes. If they were originally exchanged as media level attributes, potentially overriding session-level values, then they will also be included in INFO message payloads, following the corresponding "a=mid:" attribute.

In every INFO request agents MUST include all local candidates they have signaled previously. This is necessary in order to more easily avoid problems that would arise from unreliable transports. Mis-ordering can be detected through the CSeq: header field in the INFO request. As a consequence candidates cannot be removed unless an ICE restart is performed. Note that extension might be specified in the future that enable such removal without a restart.

When receiving INFO requests carrying any candidates, agents will therefore first identify and discard the SDP lines containing candidates they have already received in previous INFO requests or in the Offer/Answer exchange preceding them. Two candidates are considered to be equal if their IP address port, transport and component ID are the same. After identifying and discarding known candidates, the ICE agents will then receive and process the remaining, actually new candidates according to the rules described in [I-D.ietf-ice-trickle].

The following example shows the content of one sample candidate delivering INFO request:

```
INFO sip:alice@example.com SIP/2.0
...
Info-Package: trickle-ice
Content-type: application/sdp
Content-Disposition: Info-Package
Content-length: ...

a=ice-pwd:asd88fgpdd777uzjYhagZg
a=ice-ufrag:8hhY
m=audio 9 RTP/AVP 0
a=mid:1
a=candidate:1 1 UDP 1658497328 192.168.100.33 5000 typ host
a=candidate:2 1 UDP 1658497328 96.1.2.3 5000 typ srflx
        raddr 10.0.1.1 rport 8998
a=end-of-candidates
m=audio 9 RTP/AVP 0
a=mid:2
a=candidate:2 1 UDP 1658497328 96.1.2.3 5002 typ srflx
        raddr 10.0.1.1 rport 9000
a=end-of-candidates
```

5. Initial discovery of Trickle ICE support

SIP User Agents (UAs) that support and intend to use trickle ICE are REQUIRED by [I-D.ietf-ice-trickle] to indicate that in their Offers and Answers using the following attribute: "a=ice-options:trickle". This makes discovery fairly straightforward for Answerers or for cases where Offers need to be generated within existing dialogs (i.e., when sending re-INVITE requests). In both scenarios prior SDP would have provided the necessary information.

Obviously, prior SDP is not available at the time a first Offer is being constructed and it is therefore impossible for ICE agents to determine support for incremental provisioning that way. The following options are suggested as ways of addressing this issue.

5.1. Provisioning support for Trickle ICE

In certain situations it may be possible for integrators deploying Trickle ICE to know in advance that some or all endpoints reachable from within the deployment will support Trickle ICE. This is likely to be the case, for example, for WebRTC clients that will always be communicating with other WebRTC clients or known Session Border Controllers (SBC) with support for this specification.

While the exact mechanism for allowing such provisioning is out of scope here, this specification encourages trickle ICE implementations to allow the option in the way they find most appropriate.

5.2. Trickle ICE discovery with GRUU

[RFC3840] provides a way for SIP user agents to query for support of specific capabilities using, among others, OPTIONS requests. GRUU support on the other hand allows SIP requests to be addressed to specific UAs (as opposed to arbitrary instances of an address of record). Combining the two and using the "trickle-ice" option tag defined in Section 10.5 provides SIP UAs with a way of learning the capabilities of specific US instances and then addressing them directly with INVITE requests that require SIP support.

Such targeted trickling may happen in different ways. One option would be for a SIP UA to learn the GRUU instance ID of a peer through presence and to then query its capabilities direction with an OPTIONS request. Alternately, it can also just send an OPTIONS request to the AOR it intends to contact and then inspect the returned response(s) for support of both GRUU and Trickle ICE (Figure 9).

```
         Alice                                                Bob
           |                                                   |
           |        OPTIONS sip:b1@example.com SIP/2.0         |
           |-------------------------------------------------->|
           |                                                   |
           |                      200 OK                       |
           |    Contact: sip:b1@example.com;gr=hha9s8d-999a    |
           |            ;audio;video|;trickle-ice;...          |
           |<--------------------------------------------------|
           |                                                   |
           | INVITE sip:b1@example.com;gr=hha9s8d-999a SIP/2.0 |
           |-------------------------------------------------->|
           |                                                   |
           |                  183 (Answer)                     |
           |<--------------------------------------------------|
           |                INFO/OK (Trickling)                |
           |<------------------------------------------------->|
           |                                                   |
           |                      ...                          |
           |                                                   |
```

Figure 9: Trickle ICE support discovery with OPTIONS and GRUU

Confirming support for Trickle ICE through [RFC3840] gives SIP UAs the options to engage in Full Trickle negotiation (as opposed to the more lengthy Half Trickle) from the very first Offer they send.

5.3. Trickle ICE discovery through other protocols

Protocols like XMPP [RFC6120] define advanced discovery mechanisms that allow specific features to be queried priory to actually attempting to use them. Solutions like [RFC7081] define ways of using SIP and XMPP together which also provides a way for dual stack SIP+XMPP endpoints to make use of such features and verify Trickle ICE support for a specific SIP endpoint through XMPP. [TODO expand on a specific way to do this or declare as out of scope]

5.4. Fall-back to Half Trickle

In cases where none of the other mechanisms in this section are acceptable, SIP UAs should use the Half Trickle mode defined in [I-D.ietf-ice-trickle]. With Half Trickle, agents initiate sessions the same way they would when using Vanilla ICE for SIP [I-D.ietf-mmusic-ice-sip-sdp]. This means that, prior to actually sending an Offer, agents would first gather ICE candidates in a blocking way and then send them all in that Offer. The blocking nature of the process would likely imply that some amount of latency will be accumulated and it is advised that agents try to anticipate it where possible, like for example, when user actions indicate a high likelihood for an imminent call (e.g., activity on a keypad or a phone going off-hook).

Using Half Trickle would result in Offers that are compatible with both Vanilla ICE SIP endpoints and legacy [RFC3264] endpoints.

```
   STUN/Turn                                                STUN/TURN
   Servers          Alice                      Bob          Servers
      |               |                             |               |
      |<--------------|                             |               |
      |               |                             |               |
      |               |                             |               |
      |   Candidate   |                             |               |
      |               |                             |               |
      |               |                             |               |
      |   Discovery   |                             |               |
      |               |                             |               |
      |               |                             |               |
      |-------------->|       INVITE (Offer)        |               |
      |               |---------------------------->|               |
      |               |        183 (Answer)         |-------------->|
      |               |<----------------------------|               |
      |               |  INFO (repeated candidates) |               |
      |               |---------------------------->|               |
      |               |                             |               |
      |               |    INFO (more candidates)   |   Candidate   |
      |               |<----------------------------|               |
      |               |    Connectivity Checks      |               |
      |               |<===========================>|   Discovery   |
      |               |   INFO (more candidates)    |               |
      |               |<----------------------------|               |
      |               |    Connectivity Checks      |<--------------|
      |               |<===========================>|               |
      |               |                             |               |
      |               |          200 OK             |               |
      |               |<----------------------------|               |
      |               |                             |               |
      |               |      5245 SIP re-INVITE     |               |
      |               |---------------------------->|               |
      |               |            200 OK           |               |
      |               |<----------------------------|               |
      |               |                             |               |
      |               |                             |               |
      |               |<======= MEDIA FLOWS =======>|               |
      |               |                             |               |
```

Figure 10: Example - A typical (Half) Trickle ICE exchange with SIP

It is worth reminding that once a single Offer or Answer had been exchanged within a specific dialog, support for Trickle ICE will have been determined. No further use of Half Trickle will therefore be necessary within that same dialog and all subsequent exchanges can use the Full Trickle mode of operation.

6. Considerations for RTP and RTCP multiplexing

The following consideration describe options for Trickle-ICE in order to give some guidance to implementors on how trickling can be optimized with respect to providing RTCP candidates.

Handling of the "a=rtcp" attribute [RFC3605] and the "a=rtcp-mux" attribute for RTP/RTCP multiplexing [RFC5761] is already considered in section 4.2. of [I-D.ietf-mmusic-ice-sip-sdp], respectively, as well in [RFC5761] itself. These considerations are still valid for Trickle ICE, however, trickling provides more flexibility for the sequence of candidate exchange in case of RTCP multiplexing.

If the Offerer supports RTP/RTCP multiplexing exclusively as specified in [I-D.ietf-mmusic-mux-exclusive], the procedures in that document apply for the handling of the "a=rtcp-mux-only", "a=rtcp" and the "a=rtcp-mux" attributes.

While a Half Trickle Offerer would have to send an offer compliant to [I-D.ietf-mmusic-ice-sip-sdp] and [RFC5761] including candidates for all components, this flexibility allows a Full Trickle Offerer to initially send only RTP candidates (component 1) if it assumes that RTCP multiplexing is supported by the Answerer. A Full Trickle Offerer would need to start gathering and trickling RTCP candidates (component 2) only after having received an indication in the answer that the answerer unexpectedly does not support RTCP multiplexing.

A Trickle answerer MAY include an "a=rtcp-mux" attribute [RFC5761] in the application/trickle-ice-sdpfrag body it supports and uses RTP and RTCP multiplexing. Trickle answerer MUST follow the guidance on the usage of the "a=rtcp" attribute as given in [I-D.ietf-mmusic-ice-sip-sdp] and Receipt of this attribute at the Offerer in an INFO request prior to the Answer indicates that the Answerer supports and uses RTP and RTCP multiplexing. The Offerer can use this information e.g. for stopping gathering of RTCP candidates and/or for freeing corresponding resources.

This behavior is illustrated by the following example offer that indicates support for RTP and RTCP multiplexing.

```
v=0
o=alice 2890844526 2890844526 IN IP4 atlanta.example.com
s=
c=IN IP4 atlanta.example.com
t=0 0
a=ice-pwd:777uzjYhagZgasd88fgpdd
a=ice-ufrag:Yhh8
m=audio 10000 RTP/AVP 0
a=mid:1
a=rtcp-mux
a=candidate:1 1 UDP 1658497328 192.168.100.33 5000 typ host
```

Once the dialog is established as described in section Section 4.1 the Answerer sends the following INFO message.

```
INFO sip:alice@example.com SIP/2.0
...
Info-Package: trickle-ice
Content-type: application/sdp
Content-Disposition: Info-Package
Content-length: ...

a=ice-pwd:asd88fgpdd777uzjYhagZg
a=ice-ufrag:8hhY
m=audio 9 RTP/AVP 0
a=mid:1
a=rtcp-mux
a=candidate:1 1 UDP 1658497328 192.168.100.33 5000 typ host
```

This INFO message indicates that the Answerer supports and uses RTP and RTCP multiplexing as well. This allows the Offerer to omit gathering of RTCP candidates or releasing already gathered RTCP candidates. If the INFO message did not contain the a=rtcp-mux attribute, the Offerer would have to gather RTCP candidates unless it wants to wait until receipt of an Answer that eventually confirms support or non-support for RTP and RTCP multiplexing.

7. Considerations for Media Multiplexing

The following consideration describe options for Trickle-ICE in order to give some guidance to implementors on how trickling can be optimized with respect to providing candidates in case of Media Multiplexing [I-D.ietf-mmusic-sdp-bundle-negotiation]. It is assumed that the reader is familiar with [I-D.ietf-mmusic-sdp-bundle-negotiation].

ICE candidate exchange is already considered in section 11 of [I-D.ietf-mmusic-sdp-bundle-negotiation]. These considerations are still valid for Trickle ICE, however, trickling provides more flexibility for the sequence of candidate exchange, especially in Full Trickle mode.

Except for bundle-only m-lines, a Half Trickle Offerer would have to send an offer with candidates for all bundled m-lines. The additional flexibility, however, allows a Full Trickle Offerer to initially send only candidates for the m-line with the suggested Offerer BUNDLE address.

Latest on receipt of the answer, the Offerer will detect if BUNDLE is supported and if the suggested Offerer BUNDLE address was selected. In this case the Offerer does not need to trickle further candidates for the remaining m-lines in a bundle. However, if BUNDLE is not supported, the Full Trickle Offerer needs to gather and trickle candidates for the remaining m-lines as necessary. If the answerer selects a Offerer BUNDLE address different from suggested Offerer BUNDLE address, the Full Trickle Offerer needs to gather and trickle candidates for the m-line that carries the selected Offerer BUNDLE address.

A Trickle Answerer SHOULD include an "a=group: BUNDLE" attribute [I-D.ietf-mmusic-sdp-bundle-negotiation] in the application/trickle-ice-sdpfrag body if it supports and uses bundling. When doing so, the Answerer MUST include all identification-tags in the same order that is used or will be used in the Answer.

Receipt of this attribute at the Offerer in an INFO request prior to the Answer indicates that the Answerer supports and uses bundling. The Offerer can use this information e.g. for stopping the gathering of candidates for the remaining m-lines in a bundle and/or for freeing corresponding resources.

This behaviour is illustrated by the following example offer that indicates support for Media Multiplexing.

```
v=0
o=alice 2890844526 2890844526 IN IP4 atlanta.example.com
s=
c=IN IP4 atlanta.example.com
t=0 0
a=group:BUNDLE foo bar
a=ice-pwd:777uzjYhagZgasd88fgpdd
a=ice-ufrag:Yhh8
m=audio 10000 RTP/AVP 0
a=mid:foo
a=rtpmap:0 PCMU/8000
a=extmap 1 urn:ietf:params:rtp-hdrext:sdes:mid
m=video 10002 RTP/AVP 31
a=mid:bar
a=rtpmap:31 H261/90000
a=extmap 1 urn:ietf:params:rtp-hdrext:sdes:mid
```

Once the dialog is established as described in section Section 4.1 the Answerer sends the following INFO message.

```
INFO sip:alice@example.com SIP/2.0
...
Info-Package: trickle-ice
Content-type: application/sdp
Content-Disposition: Info-Package
Content-length: ...

a=group:BUNDLE foo bar
a=ice-pwd:asd88fgpdd777uzjYhagZg
a=ice-ufrag:8hhY
m=audio 9 RTP/AVP 0
a=mid:1
a=rtcp-mux
a=candidate:1 1 UDP 1658497328 192.168.100.33 5000 typ host
m=audio 9 RTP/AVP 0
a=mid:bar
```

This INFO message indicates that the Answerer supports and uses Media Multiplexing as well. Note, that the second m-line shows the default values as specified in section Section 4.2, e.g. media set 'audio' although 'video' was offered. The receiving ICE agents needs to ignore these default values in the pseudo m-lines.

The INFO message also indicates that the Answerer accepted the suggested Offerer Bundle Address. This allows the Offerer to omit gathering of RTP and RTCP candidates for the other m-lines or releasing already gathered candidates. If the INFO message did not contain the a=group:BUNDLE attribute, the Offerer would have to gather RTP and RTCP candidates for the other m-lines unless it wants to wait until receipt of an Answer that eventually confirms support or non-support for Media Multiplexing.

Independent of using Full Trickle or Half Trickle mode, the rules from [I-D.ietf-mmusic-sdp-mux-attributes] apply to both, Offerer and Answerer, when putting attributes in the application/trickle-ice-sdpfrag body.

8. SDP 'end-of-candidate' Attribute

This section defines a new SDP media-level and session-level attribute [RFC4566] 'end-of-candidate'. 'end-of-candidate' is a property attribute [RFC4566], and hence has no value. By including this attribute in an Offer or Answer the sending agent indicates that it will not trickle further candidates. The detailed SDP Offer/Answer procedures for the 'end-of-candidate' attribute are specified in [I-D.ietf-ice-trickle].

Name: end-of-candidate

Value: N/A

Usage Level: media and session-level

Charset Dependent: no

Mux Category: IDENTICAL

Example: a=end-of-candidate

9. Content Type 'application/trickle-ice-sdpfrag'

9.1. Overall Description

A application/trickle-ice-sdpfrag body is used by the Trickle-ICE Info Package. It uses a subset of the possible SDP lines that are allowed based on the grammar defined in [RFC4566]. A valid body uses only media descriptions and certain attributes that are needed and/or useful for trickling candidates. The content adheres to the following grammar.

9.2. Grammar

The grammar of an 'application/trickle-ice-sdpfrag' body is based the following ABNF [RFC5234]. It specifies the subset of existing SDP attributes, that are needed or useful for trickling candidates.

```
; Syntax
trickle-ice-sdpfrag =   session-level-fields
                  pseudo-media-descriptions
session-level-fields = [bundle-group-attribute CRLF]
                  [ice-lite-attribute CRLF]
                  ice-pwd-attribute CRLF
                  ice-ufrag-attribute CRLF
                  [ice-options-attribute CRLF]
                  [ice-pacing-attribute CRLF]
                  [end-of-candidates-attribute CRLF]
                  extension-attribute-fields
                        ; for future extensions

ice-lite-attribute     = %s"a=" ice-lite
ice-pwd-attribute      = %s"a=" ice-pwd-att
ice-ufrag-attribute    = %s"a=" ice-ufrag-att
ice-pacing-attribute  =    %s"a="  ice-pacing-att
ice-options-attribute  =    %s"a="  ice-options
bundle-group-attribute = "a=group:" bundle-semantics
                           *(SP identification-tag)
bundle-semantics = "BUNDLE"
end-of-candidates-attribute  = %s"a=" end-of-candidates
extension-attribute-fields   = attribute-fields

pseudo-media-descriptions    =  *( media-field
                           trickle-ice-attribute-fields
                          [extension-attribute-fields] )
                              ; for future extensions
trickle-ice-attribute-fields =  mid-attribute CRLF
                        ["a=rtcp-mux" CRLF]
                        ["a=rtcp-mux-only" CRLF]
                        *(candidate-attributes CRLF)
                        [ice-pwd-attribute CRLF]
                        [ice-ufrag-attribute CRLF]
                        [remote-candidate-attribute CRLF]
                        [end-of-candidates-attribute CRLF]
remote-candidate-attribute   = %s"a=" remote-candidate-att
candidate-attributes         = %s"a=" candidate-attribute
end-of-candidates            = %s"end-of-candidates"
```

with ice-lite, ice-pwd-att, remote-candidate-att, ice-ufrag-att, ice-pacing-att, ice-options, candidate-attribute remote-candidate-att

from [I-D.ietf-mmusic-ice-sip-sdp], identification-tag, mid-attribute ; from [RFC5888], media-field, attribute-fields from [RFC4566]. The indicator for case-sensitivity %s is defined in [RFC7405].

An Agent MUST ignore any received unknown extension-attribute-fields.

10. Info Package

10.1. Overall Description

This specification defines an Info Package for use by SIP user agents implementing Trickle ICE. INFO requests carry ICE candidates discovered after the peer user agents have confirmed mutual support for Trickle ICE.

10.2. Applicability

The purpose of the ICE protocol is to establish a media path in the presence of NAT and firewalls. The candidates are transported in INFO requests and are part of this establishment.

Candidates sent by a Trickle ICE agent after the Offer, follow the same signaling path and reach the same entity as the Offer itself. While it is true that GRUUs can be used to achieve this, one of the goals of this specification is to allow operation of Trickle ICE in as many environments as possible including those without GRUU support. Using out-of-dialog SUBSCRIBE/NOTIFY requests would not satisfy this goal.

10.3. Info Package Name

This document defines a SIP Info Package as per [RFC6086]. The Info Package token name for this package is "trickle-ice"

10.4. Info Package Parameters

This document does not define any Info Package parameters.

10.5. SIP Option Tags

[RFC6086] allows Info Package specifications to define SIP option-tags. This specification extends the option-tag construct of the SIP grammar as follows:

```
option-tag /= "trickle-ice"
```

SIP entities that support this specification MUST place the 'trickle-ice' option-tag in a SIP Supported: header field within all SIP INVITE requests and responses.

When responding to, or generating a SIP OPTIONS request a SIP entity MUST also include the 'trickle-ice' option-tag in a SIP Supported: header field.

10.6. Info Message Body Parts

Entities implementing this specification MUST include a payload of type 'application/trickle-ice-sdpfrag' as defined in Section 9.2 all SIP INFO requests. The payload is used to convey SDP encoded ICE candidates.

10.7. Info Package Usage Restrictions

This document does not define any Info Package Usage Restrictions.

10.8. Rate of INFO Requests

A Trickle ICE Agent with many network interfaces might create a high rate of INFO requests if every newly detected candidate is trickled individually without aggregation. Implementor that are concerned about loss of packets in such a case might consider aggregating ICE candidates and sending INFOS only at some configurable intervals.

10.9. Info Package Security Considerations

See Section 12

11. IANA Considerations

[RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this document. Please replace "I-D.ietf-ice-trickle" with the RFC number of that document.]

11.1. SDP 'end-of-candidate' Attribute

This section defines a new SDP media-level and session-level attribute [RFC4566] , 'end-of-candidate'. 'end-of-candidate' is a property attribute [RFC4566] , and hence has no value.

Name: end-of-candidate

Value: N/A

Usage Level: media and session

Charset Dependent: no

Purpose: The sender indicates that it will not trickle further candidates.

O/A Procedures: "I-D.ietf-ice-trickle" defines the detailed SDP Offer/Answer procedures for the 'end-of-candidate' attribute.

Mux Category: IDENTICAL

Reference: RFCXXXX

Example:

a=end-of-candidate

11.2. application/trickle-ice-sdpfrag MIME Type

Type name: application

Subtype name: trickle-ice-sdpfrag

Required parameters: None.

Optional parameters: None.

Encoding considerations:

SDP files are primarily UTF-8 format text. Although the initially defined content of a trickle-ice-sdpfrag body does only include ASCII characters, UTF-8 encoded content might be introduced via extension attributes. The "a=charset:" attribute may be used to signal the presence of other character sets in certain parts of a trickle-ice-sdpfrag body (see [RFC4566]). Arbitrary binary content cannot be directly represented in SDP or a trickle-ice-sdpfrag body.

Security considerations:

See [RFC4566]) and RFCXXXX

Interoperability considerations:

See RFCXXXX

Published specification:

See RFCXXXX

Applications which use this media type:

Voice over IP, video teleconferencing, streaming media, instant messaging, Trickle-ICE among others.

Additional information:

Magic number(s): none

File extension(s): none

Macintosh File Type Code(s): none

Person and email address to contact for further information:

IETF MMUSIC working group mmusic@ietf.org

Intended usage:

Trickle-ICE for SIP as specified in RFCXXXX.

Author/Change controller:

IETF MMUSIC working group mmusic@ietf.org

11.3. SIP Info Package 'trickle-ice'

This document defines a new SIP Info Package named 'trickle-ice' and updates the Info Packages Registry with the following entry.

Name	Reference
trickle-ice	[RFCXXXX]

11.4. SIP Option Tag 'trickle-ice'

This specification registers a new SIP option tag 'trickle-ice' as per the guidelines in Section 27.1 of [RFC3261] and updates the "Option Tags" section of the SIP Parameter Registry with the following entry:

Name	Description	Reference
trickle-ice	This option tag is used to indicate that a UA supports and understands Trickle-ICE.	[RFCXXXX]

12. Security Considerations

The Security Considerations of [I-D.ietf-mmusic-ice-sip-sdp], [RFC6086], [I-D.ietf-ice-trickle] apply. This document clarifies how the above specifications are used together for trickling candidates and does not create addtitional security risks.

13. Acknowledgements

The authors would like to thank Ayush Jain, Paul Kyzivat, Jonathan Lennox, Simon Perreault and Martin Thomson for reviewing and/or making various suggestions for improvements and optimizations.

14. Change Log

[RFC EDITOR NOTE: Please remove this section when publishing].

Changes from draft-ietf-mmusic-trickle-ice-sip-01

- Editorial Clean up
- IANA Consideration added
- Security Consideration added
- RTCP and BUNDLE Consideration added with rules for including "a=rtcp-mux" and "a=group: BUNDLLE" attributes
- 3PCC Consideration added
- Clarified that 18x w/o answer is sufficient to create a dialog that allows for trickling to start
- Added remaining Info Package definition sections as outlined in section 10 of [RFC6086]
- Added definition of application/sdpfrag making draft-ivov-mmusic-sdpfrag obsolete
- Added pseudo m-lines as additional separator in sdpfrag bodies for Trickle ICE
- Added ABNF for sdp-frag bodies and Trickle-ICE package

Changes from draft-ietf-mmusic-trickle-ice-sip-02

- Removed definition of application/sdpfrag
- Replaced with new type application/trickle-ice-sdpfrag
- RTCP and BUNDLE Consideration enhanced with some examples
- draft-ietf-mmusic-sdp-bundle-negotiation and RFC5761 changed to normative reference
- Removed reference to 4566bis
- Addressed review comment from Simon Perreault

Changes from draft-ietf-mmusic-trickle-ice-sip-03

- replaced reference to RFC5245 with draft-ietf-mmusic-rfc5245bis and draft-ietf-mmusic-ice-sip-sdp

- Corrected Figure 10, credits to Ayush Jain for finding the bug

- Referencing a=rtcp and a=rtcp-mux handling from draft-ietf-mmusic-ice-sip-sdp

- Referencing a=rtcp-mux-exclusive handling from draft-ietf-mmusic-mux-exclusive, enahnced ABNF to support a=rtcp-mux-exclusive

- Clarifying that draft-ietf-mmusic-sdp-mux-attributes applies for the application/trickle-ice-sdpfrag body

Changes from draft-ietf-mmusic-trickle-ice-sip-04

- considered comments from Christer Holmberg

- corrected grammar for INFO package, such that ice-ufrag/pwd are also allowed on media-level as specified in [I-D.ietf-mmusic-ice-sip-sdp]

- Added new ice-pacing-attribute fom [I-D.ietf-mmusic-ice-sip-sdp]

- Added formal definition for the end-of-candidates attribute

15. References

15.1. Normative References

[I-D.ietf-ice-trickle]
Ivov, E., Rescorla, E., Uberti, J., and P. Saint-Andre, "Trickle ICE: Incremental Provisioning of Candidates for the Interactive Connectivity Establishment (ICE) Protocol", draft-ietf-ice-trickle-03 (work in progress), July 2016.

[I-D.ietf-mmusic-ice-sip-sdp]
Petit-Huguenin, M., Keranen, A., and S. Nandakumar, "Using Interactive Connectivity Establishment (ICE) with Session Description Protocol (SDP) offer/answer and Session Initiation Protocol (SIP)", draft-ietf-mmusic-ice-sip-sdp-10 (work in progress), July 2016.

[I-D.ietf-mmusic-mux-exclusive]
Holmberg, C., "Indicating Exclusive Support of RTP/RTCP Multiplexing using SDP", draft-ietf-mmusic-mux-exclusive-10 (work in progress), August 2016.

[I-D.ietf-mmusic-rfc5245bis]
Keranen, A. and J. Rosenberg, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal", draft-ietf-mmusic-rfc5245bis-05 (work in progress), September 2015.

[I-D.ietf-mmusic-sdp-bundle-negotiation]
Holmberg, C., Alvestrand, H., and C. Jennings, "Negotiating Media Multiplexing Using the Session Description Protocol (SDP)", draft-ietf-mmusic-sdp-bundle-negotiation-31 (work in progress), June 2016.

[I-D.ietf-mmusic-sdp-mux-attributes]
Nandakumar, S., "A Framework for SDP Attributes when Multiplexing", draft-ietf-mmusic-sdp-mux-attributes-13 (work in progress), June 2016.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, DOI 10.17487/RFC3261, June 2002, <http://www.rfc-editor.org/info/rfc3261>.

[RFC3262] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)", RFC 3262, DOI 10.17487/RFC3262, June 2002, <http://www.rfc-editor.org/info/rfc3262>.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, DOI 10.17487/RFC3264, June 2002, <http://www.rfc-editor.org/info/rfc3264>.

[RFC3605] Huitema, C., "Real Time Control Protocol (RTCP) attribute in Session Description Protocol (SDP)", RFC 3605, DOI 10.17487/RFC3605, October 2003, <http://www.rfc-editor.org/info/rfc3605>.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, DOI 10.17487/RFC4566, July 2006, <http://www.rfc-editor.org/info/rfc4566>.

[RFC5234] Crocker, D., Ed. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, DOI 10.17487/RFC5234, January 2008, <http://www.rfc-editor.org/info/rfc5234>.

[RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", RFC 5761, DOI 10.17487/RFC5761, April 2010, <http://www.rfc-editor.org/info/rfc5761>.

[RFC5888] Camarillo, G. and H. Schulzrinne, "The Session Description Protocol (SDP) Grouping Framework", RFC 5888, DOI 10.17487/RFC5888, June 2010, <http://www.rfc-editor.org/info/rfc5888>.

[RFC6086] Holmberg, C., Burger, E., and H. Kaplan, "Session Initiation Protocol (SIP) INFO Method and Package Framework", RFC 6086, DOI 10.17487/RFC6086, January 2011, <http://www.rfc-editor.org/info/rfc6086>.

[RFC7405] Kyzivat, P., "Case-Sensitive String Support in ABNF", RFC 7405, DOI 10.17487/RFC7405, December 2014, <http://www.rfc-editor.org/info/rfc7405>.

15.2. Informative References

[RFC3840] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", RFC 3840, DOI 10.17487/RFC3840, August 2004, <http://www.rfc-editor.org/info/rfc3840>.

[RFC5627] Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)", RFC 5627, DOI 10.17487/RFC5627, October 2009, <http://www.rfc-editor.org/info/rfc5627>.

[RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 6120, DOI 10.17487/RFC6120, March 2011, <http://www.rfc-editor.org/info/rfc6120>.

[RFC7081] Ivov, E., Saint-Andre, P., and E. Marocco, "CUSAX: Combined Use of the Session Initiation Protocol (SIP) and the Extensible Messaging and Presence Protocol (XMPP)", RFC 7081, DOI 10.17487/RFC7081, November 2013, <http://www.rfc-editor.org/info/rfc7081>.

Authors' Addresses

Emil Ivov
Jitsi
Strasbourg 67000
France

Phone: +33 6 72 81 15 55
Email: emcho@jitsi.org

Thomas Stach
Unaffiliated
Vienna 1130
Austria

Email: thomass.stach@gmail.com

Enrico Marocco
Telecom Italia
Via G. Reiss Romoli, 274
Turin 10148
Italy

Email: enrico.marocco@telecomitalia.it

Christer Holmberg
Ericsson
Hirsalantie 11
Jorvas 02420
Finland

Email: christer.holmberg@ericsson.com