Network Working Group   E. Ivov
Internet-Draft   Jitsi
Intended status: Standards Track   T. Stach
Expires: August 27, 2018   Unaffiliated
                                                               E. Marocco
                                                           Telecom Italia
                                                              C. Holmberg
                                                                 Ericsson
                                                        February 23, 2018

               A Session Initiation Protocol (SIP) Usage for Trickle ICE
                     draft-ietf-mmusic-trickle-ice-sip-14

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) and defines a new SIP Info Package
   to support this usage.

Table of Contents

1. Introduction

   The Interactive Connectivity Establishment (ICE) protocol
   [I-D.ietf-ice-rfc5245bis] describes a mechanism for Network Address
   Translator (NAT) traversal that consists of three main phases.

   During the first phase an agent gathers a set of candidate transport
   addresses (source IP address, port and transport protocol). This is
   followed by a second phase where these candidates are sent to a
   remote agent within the Session Description Protocol (SDP) body of a
   SIP message. At the remote agent the gathering procedure is repeated
   and candidates are sent to the first agent. Finally, a third phase
   starts where connectivity between all candidates in both sets is
   checked (connectivity checks). Once these phases have been
   completed, and only then, both agents can begin communication.

   According to [I-D.ietf-ice-rfc5245bis] the three phases above happen
   consecutively, in a blocking way, which can introduce undesirable
   setup delay during session establishment. The Trickle ICE extension
   [I-D.ietf-ice-trickle] defines generic semantics required for these

ICE phases to happen in a parallel, 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 exchanged incrementally using SIP INFO requests [RFC6086]
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", "NOT RECOMMENDED", "MAY", and
"OPTIONAL" in this document are to be interpreted as described in BCP
14 [RFC2119], [RFC8174] when, and only when, they appear in all
capitals, as shown here.

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

[I-D.ietf-ice-rfc5245bis] also describes how ICE makes use of the
Session Traversal Utilities for NAT (STUN) protocol [RFC5389] and its
extension Traversal Using Relay NAT (TURN) [RFC5766].

3. Protocol Overview

When using ICE for SIP according to [I-D.ietf-mmusic-ice-sip-sdp] the
ICE candidates are exchanged solely via SDP Offer/Answer as per
[RFC3264]. This specification defines an additional mechanism where
candidates can be exchanged using SIP INFO messages and a newly
defined Info Package [RFC6086]. This allows ICE candidates also to
be sent in parallel to an ongoing Offer/Answer negotiation and/or
after the completion of the Offer/Answer negotiation.

Typically, in cases where Trickle ICE is fully supported, the Offerer
sends an INVITE request containing a subset of candidates. Once an
early dialog is established the Offerer can continue sending
candidates in INFO requests within that dialog.

Similarly, an Answerer can send ICE candidates using INFO requests
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      |            |
       |                 |----------------------->|            |
       |                 |                   |            |
       |                 |<===== MEDIA FLOWS =====>|            |
       |                 |                   |            |
```
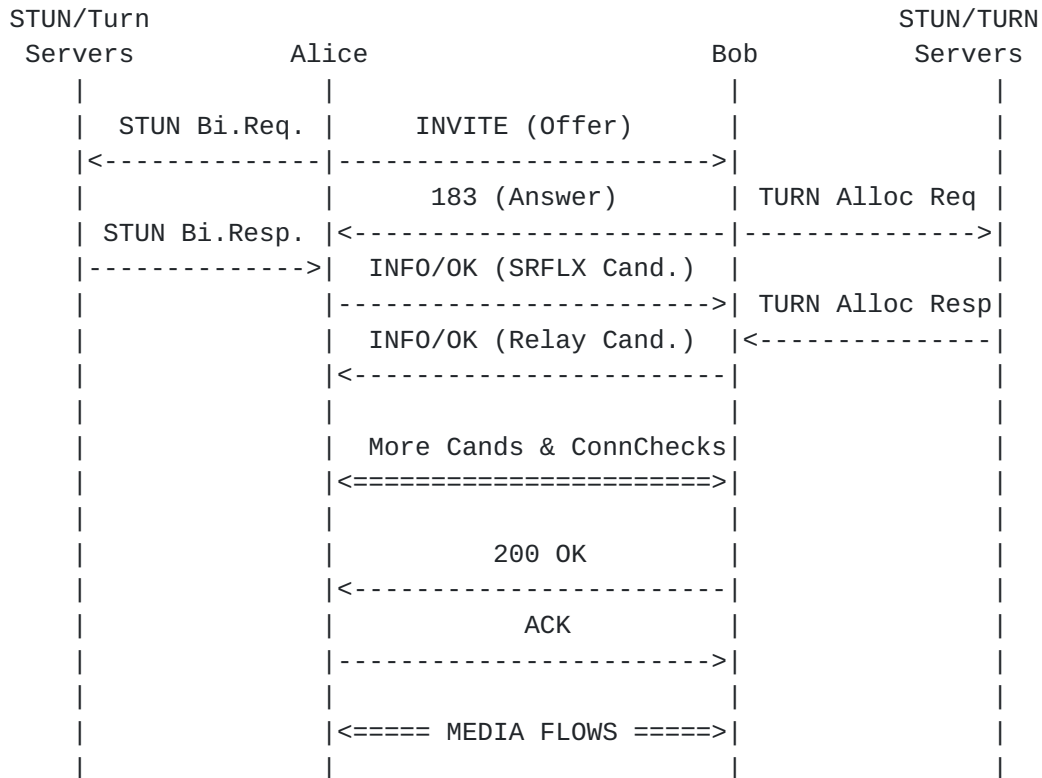
               Note: SRFLX denotes server-reflexive candidates

                  Figure 1: Sample Trickle ICE scenario with SIP

3.1. 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 using it.

   Contrary to other protocols, where "in advance" capability discovery
   is widely implemented, the mechanisms that allow this for SIP (i.e.,
   a combination of UA Capabilities [RFC3840] and Globally Routable User
   Agent URIs (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 the first Offer to contain a complete set of local ICE
candidates and only using incremental provisioning of remote
candidates for the rest of the session.

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 5.3 discusses the implementation
and use of Half Trickle by SIP UAs where none of the above are an
option.

3.2. Relationship with the Offer/Answer Model

From the perspective of SIP middle boxes and proxies the Offer/Answer
exchange for Trickle ICE looks partly similar to the Offer/Answer
exchange for regular ICE for SIP [I-D.ietf-mmusic-ice-sip-sdp].
However, in order to have the full picture of the candidate exchange,
the newly introduced INFO messages need to be considered as well.

```
    +------------------------------+  +------------------------------+
    |   Alice      +--------------+ |  | +--------------+    Bob      |
    |              | Offer/Answer | |  | | Offer/Answer |            |
    | +--------+   |    Module    | |  | |    Module    |  +--------+ |
    | |  ICE   |   +--------------+ |  | +--------------+  |  ICE   | |
    | | Module |        |          |  | |        |        | Module | |
    | +--------+        |          |  | |        |        +--------+ |
    +------------------------------+  +------------------------------+
         |             |                     |                  |
         |             |    INVITE (Offer)   |                  |
         |             |-------------------->|                  |
         |             |     183 (Answer)    |                  |
         |             |<--------------------|                  |
         |             |                     |                  |
         |                                                      |
         |          SIP INFO (more candidates)                  |
         |----------------------------------------------------->|
         |          SIP INFO (more candidates)                  |
         |<-----------------------------------------------------|
         |                                                      |
         |        STUN Binding Requests/Responses               |
         |----------------------------------------------------->|
         |        STUN Binding Requests/Responses               |
         |<-----------------------------------------------------|
         |                                                      |
```
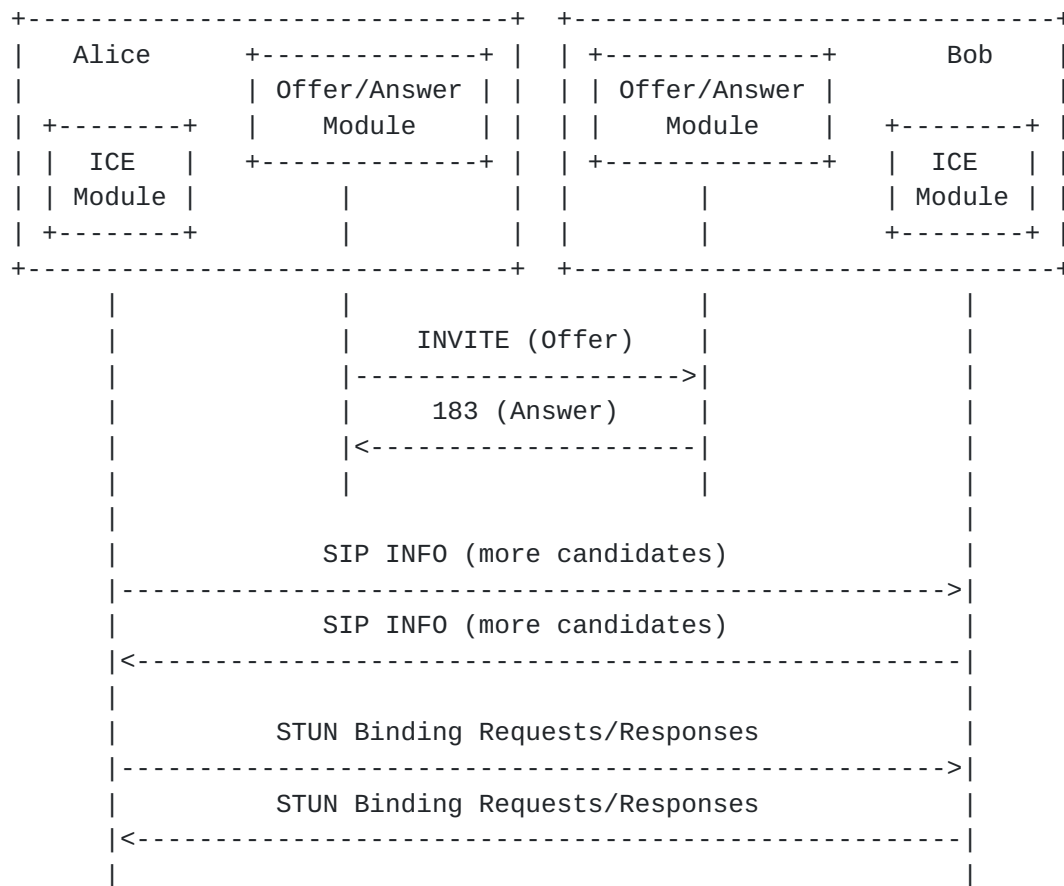
           Figure 2: Distinguishing between Trickle ICE and traditional
                              signaling.

   From an architectural viewpoint, as displayed in Figure 2, exchanging
   candidates through SIP INFO requests could be represented as
   signaling between ICE modules and not between 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, INFO requests are not considered Offers or
   Answers. Nevertheless, candidates that have been exchanged using
   INFO requests SHALL be included in subsequent Offers or Answers. The
   version number in the "o=" line of that subsequent Offer needs to be
   incremented by 1 per the rules in [RFC3264].

4. Incremental Signaling of ICE candidates

   Trickle ICE Agents will exchange ICE descriptions compliant to
   [I-D.ietf-ice-trickle] via Offer/Answer procedures and/or INFO
   request bodies. This requires the following SIP-specific extensions:

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

2. Trickle ICE Agents MUST indicate support for Trickle ICE by
    including the ice-option 'trickle' within all SDP Offers and
    Answers in accordance to [I-D.ietf-ice-trickle].

3. Trickle ICE Agents MAY include any number of ICE candidates, i.e.
    from zero to the complete set of candidates, in their initial
    Offer or Answer. If the complete candidate set is included
    already in the initial Offer, this is called Half-Trickle.

4. 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 requests
    carry an Info-Package: trickle-ice. Trickle ICE Agents MUST be
    prepared to receive INFO requests within that same dialog usage,
    containing additional candidates and/or an indication that
    trickling of such candidates has ended.

5. 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 may exist.

The following sections provide further details on how Trickle ICE
Agents perform the initial Offer/Answer exchange (Section 4.1),
perform subsequent Offer/Answer exchanges (Section 4.2) and establish
the INVITE dialog usage (Section 4.3) such that they can
incrementally trickle candidates (Section 4.4).

4.1. Initial Offer/Answer Exchange

4.1.1. Sending the Initial Offer

If the Offerer includes candidates in its initial Offer, it MUST
encode these candidates as specified in
[I-D.ietf-mmusic-ice-sip-sdp].

If the Offerer wants to send its initial Offer before knowing any
candidate for one or more media descriptions, it MUST set the port to
the default value '9' for these media descriptions. If the Offerer
does not want to include the host IP address in the corresponding
c-line, e.g. due to privacy reasons, it SHOULD include a default
address in the c-line, which is set to the IPv4 address 0.0.0.0 or to
the IPv6 equivalent ::.

In this case, the Offerer obviously cannot know the RTCP transport
address and, thus, MUST NOT include the "a=rtcp" attribute [RFC6086].
This avoids potential ICE mismatch (see
[I-D.ietf-mmusic-ice-sip-sdp]) for the RTCP transport address.

If the Offerer wants to use RTCP multiplexing [RFC5761] and/or
exclusive RTCP multiplexing [I-D.ietf-mmusic-mux-exclusive], it still
will include the "a=rtcp-mux" and/or "a=rctp-mux-only" attribute in
the initial Offer.

In any case, the Offerer MUST include the attribute "a=ice-
options:trickle" in accordance to [I-D.ietf-ice-trickle] and MUST
include in each "m="-line a "a=mid:" attribute in accordance to
[RFC5888].

4.1.2. Receiving the Initial Offer

If the initial Offer included candidates, the Answerer uses these
candidates to start ICE processing as specified in
[I-D.ietf-ice-trickle].

If the initial Offer included the attribute a=ice-options:trickle,
the Answerer MUST be prepared for receiving trickled candidates later
on.

In case of a "m/c=" line with default values none of the eventually
trickled candidates will match the default destination. This
situation MUST NOT cause an ICE mismatch (see
[I-D.ietf-mmusic-ice-sip-sdp]).

4.1.3. Sending the Initial Answer

If the Answerer includes candidates in its initial Answerer, it MUST
encode these candidates as specified in
[I-D.ietf-mmusic-ice-sip-sdp].

If the Answerer wants to send its initial Answer before knowing any
candidate for one or more media descriptions, it MUST set the port to
the default value '9' for these media descriptions. If the Answerer
does not want to include the host IP address in the corresponding
c-line, e.g. due to privacy reasons, it SHOULD include a default
address in the c-line, which is set to the IPv4 address 0.0.0.0 or to
the IPv6 equivalent ::.

In this case, the Answerer obviously cannot know the RTCP transport
address and, thus, MUST NOT include the "a=rtcp" attribute [RFC6086].
This avoids potential ICE mismatch (see
[I-D.ietf-mmusic-ice-sip-sdp]) for the RTCP transport address.

If the Answerer accepts to use RTCP multiplexing [RFC5761] and/or
exclusive RTCP multiplexing [I-D.ietf-mmusic-mux-exclusive], it will
include the "a=rtcp-mux" attribute in the initial Answer.

In any case, the Answerer MUST include the attribute "a=ice-
options:trickle" in accordance to [I-D.ietf-ice-trickle] and MUST
include in each "m="-line a "a=mid:" attribute in accordance to
[RFC5888].

4.1.4. Receiving the Initial Answer

If the initial Answer included candidates, the Offerer uses these
candidates to start ICE processing as specified in
[I-D.ietf-ice-trickle].

If the initial Answer included the attribute a=ice-options:trickle,
the Offerer MUST be prepared for receiving trickled candidates later
on.

In case of a "m/c=" line with default values none of the eventually
trickled candidates will match the default destination. This
situation MUST NOT cause an ICE mismatch (see
[I-D.ietf-mmusic-ice-sip-sdp]).

4.2. Subsequent Offer/Answer Exchanges

Subsequent Offer/Answer exchanges are handled as for regular ICE (see
section 4.2 of [I-D.ietf-mmusic-ice-sip-sdp]).

If an Offer or Answer needs to be sent while the ICE agents are in
the middle of trickling section 4.2.1.2.1 of
[I-D.ietf-mmusic-ice-sip-sdp]) applies. This means that an ICE agent
includes candidate attributes for all local candidates it had
trickled previously for a specific media stream.

[RFC EDITOR NOTE: The section 4.2.1.2.1 in above sentence is correct
for version 16 of said I-D. Authors need to cross-check during
Auth48 since it could have have changed in the meantime.]

4.3. Establishing the Dialog

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

o Trickle ICE support at the peer agent MUST be confirmed.

o The dialog at both peers 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).

4.3.1. Establishing 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.)   |
          |<---------------------------|
          |                            |

       Note: SRFLX denotes server-reflexive candidates
```

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

As shown in Figure 3 satisfying both conditions is 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 at the Answerer
(or lack thereof) and to have fully initialized the SIP dialog at
both ends. Offerers and Answerers (after receipt of the PRACK
request) in the above situation can therefore freely commence
trickling within the newly established dialog.

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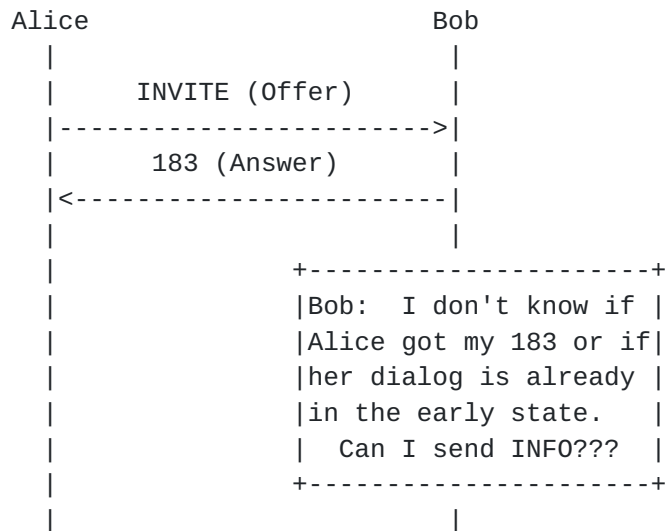
        Figure 4: A SIP UA that sent an Answer in an unreliable provisional
        response does not know if it was received and if the dialog at the
                side of the Offerer has entered the early state

   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]. These retransmissions MUST
   cease on receipt of an INFO request carrying a 'trickle-ice' Info
   Package body or on transmission of the Answer in a 2xx response.
   This is similar to the procedure described in section 8.1.1 of
   [I-D.ietf-mmusic-ice-sip-sdp] except that the STUN binding Request is
   replaced by the INFO request.

   [RFC EDITOR NOTE: The section 8.1.1 in above sentence is correct for
   version 16 of said I-D. Authors need to cross-check during Auth48
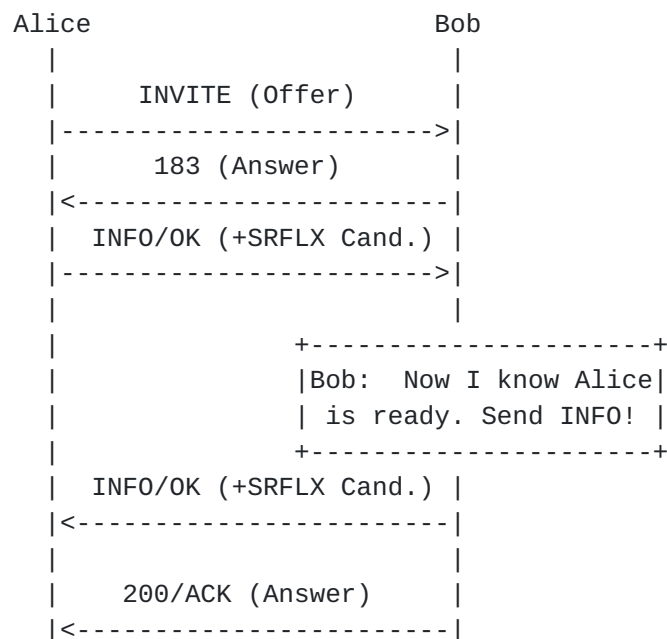   since it could have have changed in the meantime.]

   The Offerer MUST send a Trickle ICE INFO request as soon as it
   receives an SDP Answer in an unreliable provisional response. This
   INFO request 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).

If available, the Offerer SHOULD also deliver newly learned
candidates in this INFO request, unless it wants to hold back some
candidates in reserve, e.g. in case that these candidates are
expensive to use and would only be trickled if all other candidates
failed.

The Offerer SHOULD include an end-of-candidates attribute in case
candidate discovery has ended in the mean time and no further
candidates are to be trickled.

As soon as an Answerer has received such an INFO request, the
Answerer has an indication that a dialog is established at both ends
and can begin trickling (Figure 5).

Note: The +SRFLX in Figure 5 indicates that additionally newly
learned server-reflexive candidates are included.

```
              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)     |
                |<-------------------------|


        Note: SRFLX denotes server-reflexive candidates
```

   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 to the INVITE request,
the Answerer needs to repeat exactly the same Answer that was
previously sent in the unreliable provisional response in order to
fulfill the corresponding requirements in [RFC3264]. Thus, the
Offerer needs to be prepared for receiving a different number of

candidates in that repeated Answer than previously exchanged via
trickling and MUST ignore the candidate information in that 200 OK
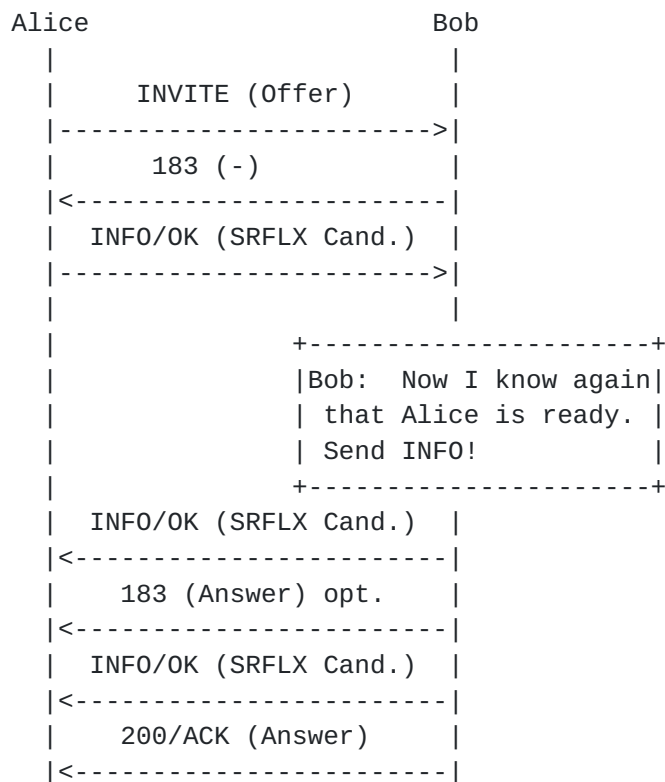response.

4.3.3. Initiating Trickle ICE without an SDP Answer

The ability to convey arbitrary candidates in INFO message bodies
allows ICE Agents to initiate trickling without actually sending an
Answer. Trickle ICE Agents can therefore respond to an INVITE
request with provisional responses without an SDP Answer [RFC3261].
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]. These retransmissions 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 8.1.1 of [I-D.ietf-mmusic-ice-sip-sdp] except that an Answer
is not yet provided.

[RFC EDITOR NOTE: The section 8.1.1 in above sentence is correct for
version 16 of said I-D. Authors need to cross-check during Auth48
since it could have have changed in the meantime.]

Note: The +SRFLX in Figure 6 indicates that additionally newly
learned server-reflexive candidates are included.

```
                     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)    |
                       |<-------------------------|


           Note: SRFLX denotes server-reflexive candidates
```

   Figure 6: A SIP UA sends an unreliable provisional response without
              an Answer for establishing an early dialog
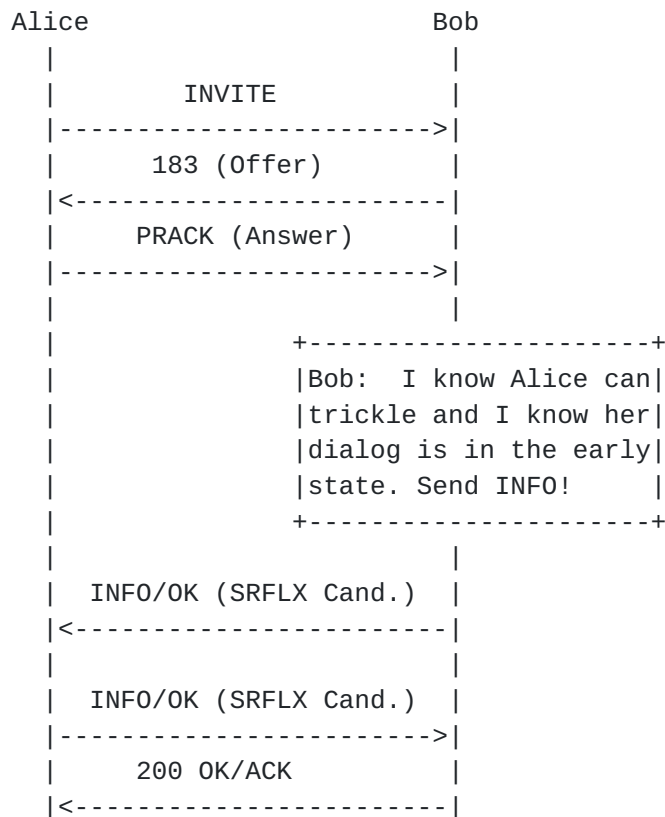
   When sending the Answer, the agent MUST repeat all currently known
   and used candidates, if any, and MAY include all newly gathered
   candidates since the last INFO request was sent. However, if that
   Answer was already sent in a unreliable provisional response, the
   Answerers MUST repeat exactly the same Answer in the 200 OK response
   to the INVITE request in order to fulfill the corresponding
   requirements in [RFC3264]. In case that trickling continued, an
   Offerer needs to be prepared for receiving fewer candidates in that
   repeated Answer than previously exchanged via trickling and MUST
   ignore the candidate information in that 200 OK response.

4.3.4. Considerations for Third Party Call Control

   Third Party Call Control (3PCC) for SIP can be performed using
   several signaling variants as described in [RFC3725]. We give
   specific consideration for 3PCC that starts with an offerless INVITE
   request [RFC3261]. As specified in Section 4 this offerless INVITE
   MUST include the SIP option-tag 'trickle-ice' in a SIP Supported:
   header field in order to indicate support for Trickle-ICE to the

Offerer (at the User Agent Server (UAS)). Then, a UAS has the option
to send its Offer in a reliable provisional response [RFC3262] or in
the 200 OK response to the INVITE request.

Agents that had sent an Offer in a reliable provisional response and
that received an Answer in a PRACK request [RFC3262] are also in a
situation where support for Trickle ICE is confirmed and the SIP
dialog is guaranteed to be in a state that allows in-dialog INFO
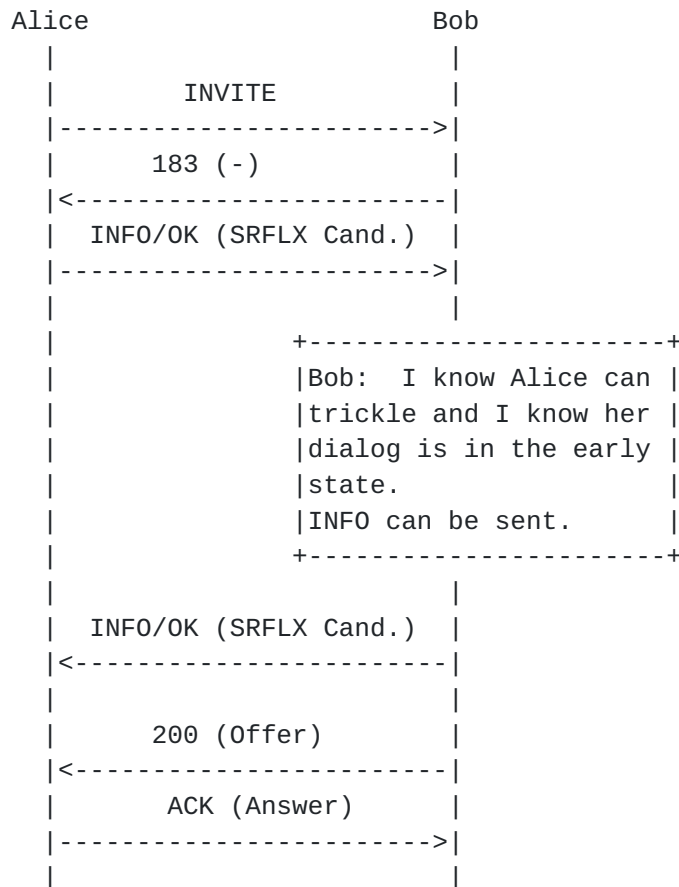requests (see Figure 7).

```
                    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      |
                      |<----------------------|


          Note: SRFLX denotes server-reflexive candidates
```


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

Trickle ICE Agents that send an Offer in a 200 OK response and
receive an Answer in an ACK message can still create a dialog and
confirm support for Trickle ICE by sending an unreliable provisional
response similar to Section 4.3.3. As specified in Section 4 this
unreliable provisional response MUST include the SIP option-tag
'trickle-ice' in a SIP Supported: header field in order to indicate
support for Trickle-ICE to the UAC. According to [RFC3261], this
unreliable response cannot contain an Offer.

The Trickle ICE Agent, i.e. the user Agent server (UAS), retransmits
the provisional response 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. The peer
Trickle ICE Agent (the UAC) MUST send a Trickle ICE INFO request as
soon as it receives 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)      |
                    |-------------------------->|
                    |                           |
```

        Note: SRFLX denotes server-reflexive candidates

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

4.4. Delivering Candidates in INFO Requests

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

```
   a=candidate:2 1 UDP 1694498815 2001:db8:a0b:12f0::3 5000 typ srflx
      raddr 2001:db8:a0b:12f0::1 rport 8998
```

The use of SIP INFO requests happens within the context of the Info
Package as defined Section 10. The Media Type [RFC6838] for their
payload MUST be set to 'application/trickle-ice-sdpfrag' as defined
in Section 9. The Info request body adheres to the grammar as
specified in Section 9.2.

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 any remaining content of the pseudo "m=" line,
which is not defined in this document. 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
"m=" line in 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 field is set to 'audio'.

o The port value is set to '9'.

o The proto value is set to 'RTP/AVP'.

o The fmt field MUST 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 a
pseudo "m=" line, pertain to the "m=" line identified by that
identification tag.

Note, that there is no requirement that the Info request body
contains as many pseudo m= lines as the Offer/Answer contains m=
lines, nor that the pseudo m= lines be in the same order as the
m=lines that they pertain to. The correspondence can be made via the
"a=mid:" attributes.

An "a=end-of-candidates" attribute, preceding any pseudo "m=" line,
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.

Refer to Figure 9 for an example of the INFO request content.

The use of pseudo "m=" lines 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 pseudo "m=" line are considered to be session-level.
Lines appearing in between or after pseudo "m=" lines 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, 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 allow mapping them to a specific ICE generation. An
agent MUST discard any received INFO requests containing "a=ice-pwd:"
and "a=ice-ufrag:" attributes that do not match those of the current
ICE Negotiation Session.

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, they will also
appear at the beginning of all INFO request payloads, i.e. preceding
all pseudo "m=" lines. If they were originally exchanged as media
level attributes, potentially overriding session-level values, then
they will also be included in INFO request payloads following the
corresponding pseudo "m=" lines.

Note that [I-D.ietf-ice-trickle] requires that when candidates are
trickled, each candidate must be delivered to the receiving Trickle
ICE implementation not more than once and in the same order as it was
conveyed. If the signaling protocol provides any candidate
retransmissions, they need to be hidden from the ICE implementation.
This requirement is fulfilled as follows.

Since the agent is not fully aware of the state of the ICE
Negotiation Session at its peer it MUST include all currently known
and used local candidates in every INFO request. I.e. the agent MUST
repeat in the INFO request body all candidates that were previously
sent under the same combination of "a=ice-pwd:" and "a=ice-ufrag:" in
the same order as they were gathered. In other words, the sequence
of a previously sent list of candidates MUST NOT change in subsequent
INFO requests and newly gathered candidates MUST be added at the end
of that list. Although repeating all candidates creates some
overhead, it also allows easier handling of problems that could arise
from unreliable transports, like e.g. loss of messages and
reordering, which can be detected through the CSeq: header field in
the INFO request.

When receiving INFO requests carrying any candidates, agents MUST
therefore first identify and discard the attribute lines containing
candidates they have already received in previous INFO requests or in
the Offer/Answer exchange preceding them.

Such candidates are considered to be equal if their IP address port,
transport and component ID are the same. After identifying and
discarding the known candidates, the agents MUST forward the actually
new candidates to the ICE Agents in the same order as they were
received in the INFO request body. The ICE Agents will then process
the new candidates according to the rules described in
[I-D.ietf-ice-trickle].

Receiving an "a=end-of-candidates" attribute in an INFO request body
- with the "a=ice-ufrag" and "a=ice-pwd" attributes matching the
current ICE generation - is an indication from the peer agent that it
will not send any further candidates. When included at session
level, i.e. before any pseudo "m=" line, this indication applies to
the whole session; when included at media level the indication
applies only to the corresponding "m=" line. Handling of such end-
of-candidates indications is defined in [I-D.ietf-ice-trickle].

Note: At the time of writing this specification there were ongoing
discussions if a functionality for removing already exchanged
candidates would be useful. Such a functionality is out of the scope
of this specification and most likely needs to be signaled by means
of a yet to be defined ICE extension, although it could in principle
be achieved quite easily, e.g. without anticipating any solution by
simply omitting a previously sent candidate from a subsequent INFO
request. However, if an implementation according to this
specification receives such an INFO request with a missing candidate
it would have to treat that as an exceptional case. Implementing
appropriate recovery procedures at the receiving side is advisable

for this situation. Ignoring that a candidate was missing might be a
sensible strategy.

The example in Figure 9 shows the content of a candidate delivering
INFO request. In the example the "a=end-of-candidates" attributes
indicate that the candidate gathering is finished and that no further
INFO requests follow.

```
INFO sip:alice@example.com SIP/2.0
...
Info-Package: trickle-ice
Content-type: application/trickle-ice-sdpfrag
Content-Disposition: Info-Package
Content-length: 862

a=ice-pwd:asd88fgpdd777uzjYhagZg
a=ice-ufrag:8hhY
m=audio 9 RTP/AVP 0
a=mid:1
a=candidate:1 1 UDP 2130706431 2001:db8:a0b:12f0::1 5000 typ host
a=candidate:1 2 UDP 2130706431 2001:db8:a0b:12f0::1 5001 typ host
a=candidate:2 1 UDP 1694498815 2001:db8:a0b:12f0::3 5000 typ srflx
   raddr 2001:db8:a0b:12f0::1 rport 8998
a=candidate:2 2 UDP 1694498815 2001:db8:a0b:12f0::3 5001 typ srflx
   raddr 2001:db8:a0b:12f0::1 rport 8998
a=end-of-candidates
m=audio 9 RTP/AVP 0
a=mid:2
a=candidate:1 1 UDP 2130706431 2001:db8:a0b:12f0::1 6000 typ host
a=candidate:1 2 UDP 2130706431 2001:db8:a0b:12f0::1 6001 typ host
a=candidate:2 1 UDP 1694498815 2001:db8:a0b:12f0::3 6000 typ srflx
   raddr 2001:db8:a0b:12f0::1 rport 9998
a=candidate:2 2 UDP 1694498815 2001:db8:a0b:12f0::3 6001 typ srflx
   raddr 2001:db8:a0b:12f0::1 rport 9998
a=end-of-candidates
```

Note: In a real INFO request there will be no line breaks
        in the a=candidate: attributes

     Figure 9: An Example for the Content of an INFO Request

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 attribute "a=ice-options:trickle" and MUST
   include the SIP option-tag "trickle-ice" in a SIP Supported: header
   field. This makes discovery fairly straightforward for Answerers or
   for cases where Offers need to be generated within existing dialogs
   (i.e., when sending UPDATE or re-INVITE requests). In both scenarios
   prior SDP bodies will have provided the necessary information.

   Obviously, such information 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 the
   case, for example, if Session Border Controllers (SBC) with support
   for this specification are used to connect to UAs that do not support
   Trickle ICE.

   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 Globally Routable User Agent URIs
      (GRUU)

   [RFC3840] provides a way for SIP User Agents to query for support of
   specific capabilities using, among others, OPTIONS requests. Support
   for GRUU according to [RFC5627] 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 10.6 provides SIP UAs with a way of
   learning the capabilities of specific SIP UA instances and then
   addressing them directly with INVITE requests that require Trickle
   ICE support.

   Such learning of capabilities may happen in different ways. One
   option for a SIP UA is to learn the GRUU instance ID of a peer
   through presence and then to query its capabilities with an OPTIONS
   request. Alternatively, it can also just send an OPTIONS request to
   the Address of Record (AOR) it intends to contact and then inspect
   the returned response(s) for support of both GRUU and Trickle ICE

(Figure 10). It is noted that using the GRUU means that the INVITE
request can go only to that particular device. This prevents the use
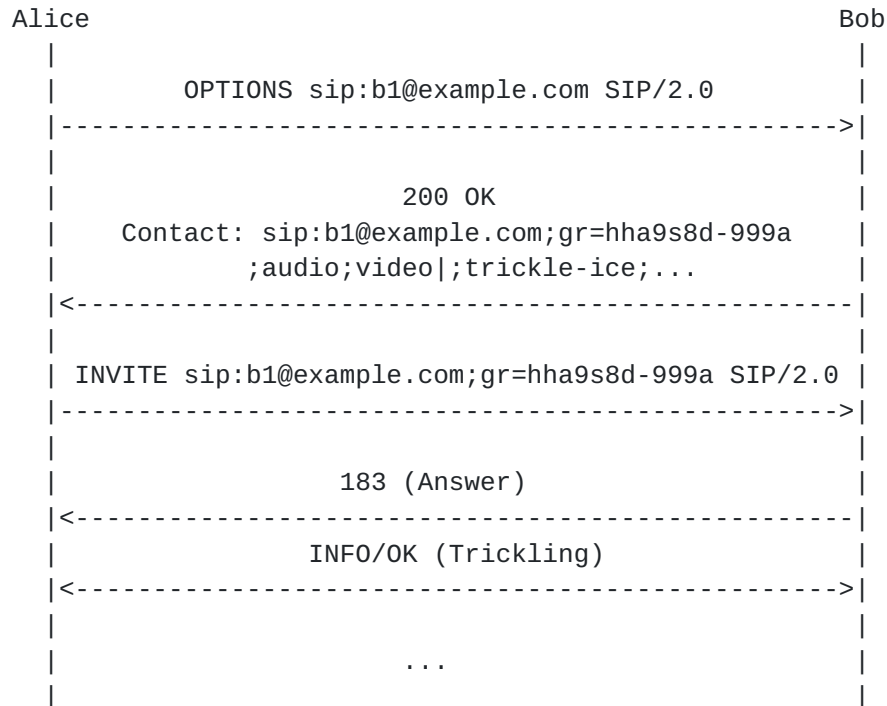of forking for that request.

```
         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 10: 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. 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 ICE for SIP
[I-D.ietf-mmusic-ice-sip-sdp]. This means that, prior to actually
sending an Offer, agents first gather ICE candidates in a blocking
way and then send them all in that Offer. The blocking nature of the
process implies that some amount of latency will be accumulated and
it is advised that agents try to anticipate it where possible, 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 results in Offers that are compatible with both
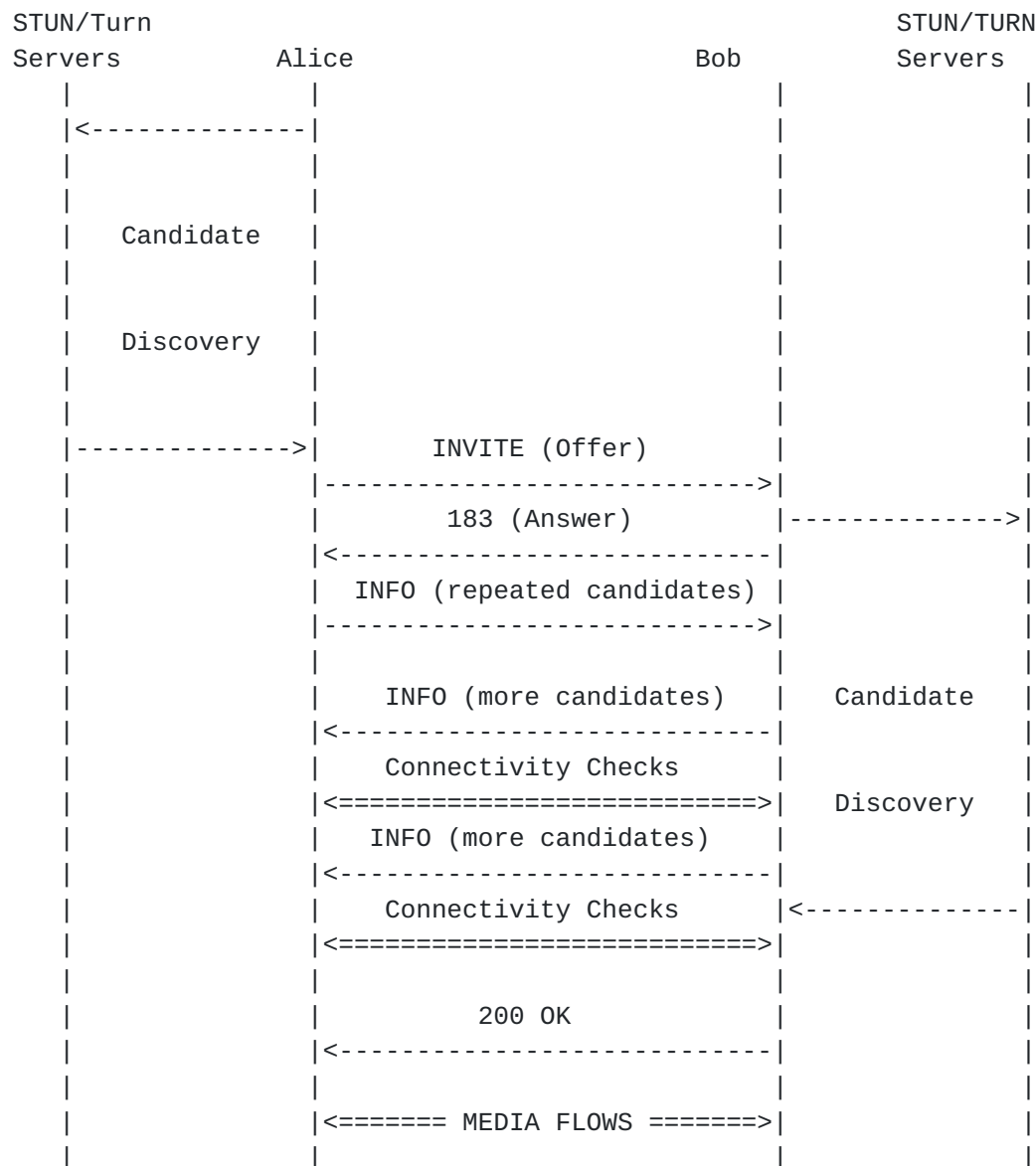ICE SIP endpoints and legacy [RFC3264] endpoints.

```
     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          |         |
        |                |<----------------------------|         |
        |                |                       |         |
        |                |<======= MEDIA FLOWS ======>|         |
        |                |                       |         |
```

   Figure 11: 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 5.1.1.1. of [I-D.ietf-ice-rfc5245bis] and 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.

[RFC EDITOR NOTE: The section 5.1.1.1 in above sentence is correct for version 15 of said I-D. Authors need to cross-check during Auth48 since it could have have changed in the meantime.]

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 has to send an Offer compliant to [I-D.ietf-mmusic-ice-sip-sdp] and [RFC5761] including candidates for all components, the flexibility of a Full Trickle Offerer allows to send only RTP candidates (component 1) in the initial Offer assuming 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 if it supports and uses RTP and RTCP multiplexing. The Trickle Answerer needs to follow the guidance on the usage of the "a=rtcp" attribute as given in [I-D.ietf-mmusic-ice-sip-sdp] and [RFC3605]. 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 IP6 atlanta.example.com
s=
c=IN IP6 2001:db8:a0b:12f0::3
t=0 0
a=ice-pwd:777uzjYhagZgasd88fgpdd
a=ice-ufrag:Yhh8
m=audio 5000 RTP/AVP 0
a=mid:1
a=rtcp-mux
a=candidate:1 1 UDP 1658497328 2001:db8:a0b:12f0::3 5000 typ host
```

Once the dialog is established as described in section [Section 4.3](#)
the Answerer sends the following INFO request.

```
INFO sip:alice@example.com SIP/2.0
...
Info-Package: trickle-ice
Content-type: application/trickle-ice-sdpfrag
Content-Disposition: Info-Package
Content-length: 161

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 1658497382 2001:db8:a0b:12f0::4 6000 typ host
```

This INFO request indicates that the Answerer supports and uses RTP
and RTCP multiplexing as well. It allows the Offerer to omit
gathering of RTCP candidates or releasing already gathered RTCP
candidates. If the INFO request did not contain the a=rtcp-mux
attribute, the Offerer has 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 considerations 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 has 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.

On receipt of the Answer, the Offerer will detect if BUNDLE is
supported by the Answerer 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 an Offerer BUNDLE address
different from the 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 IP6 atlanta.example.com
s=
c=IN IP6 2001:db8:a0b:12f0::3
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=rtcp-mux
a=rtpmap:0 PCMU/8000
a=extmap 1 urn:ietf:params:rtp-hdrext:sdes:mid
a=candidate:1 1 UDP 1658497328 2001:db8:a0b:12f0::3 10000 typ host
m=video 10002 RTP/AVP 31
a=mid:bar
a=rtcp-mux
a=rtpmap:31 H261/90000
a=extmap 1 urn:ietf:params:rtp-hdrext:sdes:mid
```

The example Offer indicates support for RTP and RTCP multiplexing and
contains a "a=candidate:" attribute only for the m-line with the
suggested Offerer bundle address. Once the dialog is established as
described in Section 4.3 the Answerer sends the following INFO
request.

```
INFO sip:alice@example.com SIP/2.0
...
Info-Package: trickle-ice
Content-type: application/trickle-ice-sdpfrag
Content-Disposition: Info-Package
Content-length: 219

a=group:BUNDLE foo bar
a=ice-pwd:asd88fgpdd777uzjYhagZg
a=ice-ufrag:8hhY
m=audio 9 RTP/AVP 0
a=mid:foo
a=rtcp-mux
a=candidate:1 1 UDP 1658497328 2001:db8:a0b:12f0::3 5000 typ host
m=audio 9 RTP/AVP 0
a=mid:bar
```

This INFO request 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 4.4, e.g. media set
'audio' although 'video' was offered. The receiving ICE Agents MUST
ignore these default values in the pseudo "m=" lines.

The INFO request 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 request did not
contain the a=group:BUNDLE attribute, the Offerer has 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 as specified in Section 9.2 in the
application/trickle-ice-sdpfrag body.

8. SDP 'end-of-candidates' Attribute

8.1. Definition

This section defines a new SDP media-level and session-level
attribute [RFC4566] 'end-of-candidates'. 'end-of-candidates' 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. When included at session
level this indication applies to the whole session, when included at
media level the indication applies only to the corresponding media
description.

    Name: end-of-candidates

    Value: N/A

    Usage Level: media and session-level

    Charset Dependent: no

    Mux Category: IDENTICAL

    Example: a=end-of-candidates

8.2. Offer/Answer Procedures

   The Offerer or Answerer MAY include an "a=end-of-candidates"
   attribute in case candidate discovery has ended and no further
   candidates are to be trickled. The Offerer or Answerer MUST provide
   the "a=end-of-candidates" attribute together with the "a=ice-ufrag"
   and "a=ice-pwd" attributes of the current ICE generation as required
   by [I-D.ietf-ice-trickle]. When included at session level this
   indication applies to the whole session; when included at media level
   the indication applies only to the corresponding media description.

   Receipt of an "a=end-of-candidates" attribute at an Offerer or
   Answerer - with the "a=ice-ufrag" and "a=ice-pwd" attributes matching
   the current ICE generation - indicates that gathering of candidates
   has ended at the peer, either for the session or only for the
   corresponding media description as specified above. The receiving
   agent forwards an end-of-candidates indication to the ICE Agent,
   which in turn acts as specified in [I-D.ietf-ice-trickle].

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

9.1. Overall Description

   A application/trickle-ice-sdpfrag body is used exclusively by the
   'trickle-ice' Info Package. Other SDP related applications need to
   define their own media type. The INFO request body uses a subset of
   the possible SDP lines as defined by the grammar defined in
   [RFC4566]. A valid body uses only pseudo "m=" lines 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 on
   the following ABNF [RFC5234]. It specifies the subset of existing
   SDP attributes, that is needed or useful for trickling candidates.
   The grammar uses the indicator for case-sensitivity %s as defined in
   [RFC7405], but also imports grammars for other SDP attributes that
   precede the production of [RFC7405]. A sender SHOULD use lower-case
   for attributes from such earlier grammars, but a receiver MUST treat
   them case-insensitively.

```
         ;  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 = %s"a" "=" %s"group:" bundle-semantics
                              *(SP identification-tag)
         bundle-semantics = "BUNDLE"
         end-of-candidates-attribute  = %s"a" "=" end-of-candidates
         end-of-candidates            = %s"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 = %s"a" "="  mid-attribute CRLF
                             [%s"a" "=" %s"rtcp" CRLF]
                             [%s"a" "=" %s"rtcp-mux" CRLF]
                             [%s"a" "=" %s"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
```

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
"a=rtcp" attribute is defined in [RFC3605], the "a=rtcp-mux"
attribute in [RFC5761] and the "a=rtcp-mux-only" attribute in

[I-D.ietf-mmusic-mux-exclusive]. The latter attributes lack a formal
grammar in their corresponding RFC and are reproduced here.

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

10. Info Package

10.1. 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 requests can be exchanged both ways with no
restrictions on timing and frequency and no risk of collision.

A critical fact is that the sending of Trickle ICE candidates in one
direction is entirely uncoupled from sending candidates in the other
direction. Thus, the sending of candidates in each direction can be
done by a stream of INFO requests that is not correlated with the
stream of INFO requests in the other direction. And since each INFO
request cumulatively includes the contents of all previous INFO
requests in that direction, ordering between INFO requests need not
be preserved. All of this permits using largely-independent INFO
requests.

Contrarily, UPDATE or other offer/answer mechanisms assume that the
messages in each direction are tightly coupled with messages in the
other direction. Using Offer/Answer and UPDATE requests [RFC3311]
would introduce the following complications:

Blocking of messages: [RFC3264] defines Offer/Answer as a strictly
   sequential mechanism. There can only be a maximum of one active
   exchange at any point of time. Both sides cannot simultaneously
   send Offers nor can they generate multiple Offers prior to
   receiving an Answer. Using UPDATE requests 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 implies that the duration of Trickle ICE processing
would not only fail to improve but actually exceed those of
regular 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. Out-of-
dialog alternatives 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 a constraint to
the adoption of Trickle ICE.

10.2. 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.

10.3. 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.

10.4. Info Package Name

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

10.5. Info Package Parameters

   This document does not define any Info Package parameters.

10.6. 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.

10.7. Info Request Body Parts

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

10.8. Info Package Usage Restrictions

   This document does not define any Info Package Usage Restrictions.

10.9. Rate of INFO Requests

   Given that IP addresses may be gathered rapidly 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. Implementors MUST consider aggregating ICE
   candidates in case that UDP is used as transport protocol and send
   INFOs only at some configurable intervals.

   If the INFO requests are sent on top of TCP, which is probably the
   standard way, this is not an issue for the network anymore, but it
   can remain one for SIP proxies and other intermediaries forwarding
   the SIP INFO messages. Also, an endpoint may not be able to tell
   that it has congestion controlled transport all the way.

[10.10](#). **Info Package Security Considerations**

   See [Section 13](#)

[11](#). **Deployment Considerations**

   Trickle ICE uses two mechanism for exchange of candidate information.
   This imposes new requirements to certain middleboxes that are used in
   some networks, e.g. for monitoring purposes. While the first
   mechanism, SDP Offers and Answers, is already used by regular ICE and
   is assumed to be supported, the second mechanism, INFO request
   bodies, needs to be considered by such middleboxes as well when
   trickle ICE is used. Such middleboxes need to make sure that they
   remain in the signaling path of the INFO requests and need to
   understand the INFO request body.

[12](#). **IANA Considerations**

   [RFC EDITOR NOTE: Please replace RFCXXXX with the RFC number of this
   document.]

[12.1](#). **SDP 'end-of-candidates' Attribute**

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

Name: end-of-candidates

Value: N/A

Usage Level: media and session

Charset Dependent: no

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

O/A Procedures: RFCXXX defines the detailed
                SDP Offer/Answer procedures for
                the 'end-of-candidates' attribute.

Mux Category: IDENTICAL

Reference: RFCXXXX

Example:

a=end-of-candidates

12.2. Media Type 'application/trickle-ice-sdpfrag'

This document defines a new Media Type 'application/trickle-ice-
sdpfrag' in accordance with [RFC6838].

Type name: application

Subtype name: trickle-ice-sdpfrag

Required parameters: None.

Optional parameters: None.

Encoding considerations:

The media contents follow the same rules as SDP, except as
noted in this document. The media contents are text, with the
grammar specified in Section 9.2.

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:

Trickle-ICE

Fragment identifier considerations: N/A

Additional information:

Deprecated alias names for this type: N/A

Magic number(s): N/A

File extension(s): N/A

Macintosh File Type Code(s): N/A

Person and email address to contact for further information:

The IESG (iesg@ietf.org)

Intended usage:

Trickle-ICE for SIP as specified in RFCXXXX.

Restrictions on usage: N/A

Author/Change controller:

The IESG (iesg@ietf.org)

Provisional registration? (standards tree only): N/A

12.3. 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] |
|             |           |
+-------------+-----------+
```

12.4. 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 | [RFCXXXX] |
|             | that a UA supports and understands  |           |
|             | Trickle-ICE.                        |           |
+-------------+-------------------------------------+-----------+
```

13. Security Considerations

The Security Considerations of [I-D.ietf-mmusic-ice-sip-sdp],
[RFC6086] and [I-D.ietf-ice-trickle] apply. This document clarifies

how the above specifications are used together for trickling
candidates and does not create additional security risks.

The new Info Package 'trickle-ice' and the new Media Type
'application/trickle-ice-sdpfrag' do not introduce additional
security considerations when used in the context of Trickle ICE.
Both are not intended to be used for other applications, so any
security considerations for its use in other contexts is out of the
scope of this document

14. Acknowledgements

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

The authors also like to thank Flemming Andreasen for shepherding
this document and Ben Campbell for his AD review and suggestions.

Many thanks to Dale Worley for Gen-Art review and proposed
enhancements for several sections.

Many thanks to Joerg Ott for TSV-Art review and suggested
improvements.

The authors thank Shawn Emery for Security Directorate review.

15. Change Log

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

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

o Editorial Clean up

o IANA Consideration added

o Security Consideration added

o RTCP and BUNDLE Consideration added with rules for including
   "a=rtcp-mux" and "a=group: BUNDLLE" attributes

o 3PCC Consideration added

o Clarified that 18x w/o answer is sufficient to create a dialog
   that allows for trickling to start

o Added remaining Info Package definition sections as outlined in
   section 10 of [RFC6086]

o Added definition of application/sdpfrag making draft-ivov-mmusic-
   sdpfrag obsolete

o Added pseudo m-lines as additional separator in sdpfrag bodies for
   Trickle ICE

o Added ABNF for sdp-frag bodies and Trickle-ICE package

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

o Removed definition of application/sdpfrag

o Replaced with new type application/trickle-ice-sdpfrag

o RTCP and BUNDLE Consideration enhanced with some examples

o draft-ietf-mmusic-sdp-bundle-negotiation and RFC5761 changed to
   normative reference

o Removed reference to 4566bis

o Addressed review comment from Simon Perreault

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

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

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

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

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

o 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

o considered comments from Christer Holmberg

o 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]

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

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

Changes from draft-ietf-mmusic-trickle-ice-sip-05

o considered further comments from Christer Holmberg

o editorial comments on section 3 addressed

o moved section 3.1 to section 10.1 and applied some edits

o replaced the term "previously sent candidates" with "currently
   known and used candidates".

Changes from draft-ietf-mmusic-trickle-ice-sip-06

o editorial fixes

o additional text on the content of the INFO messages.

o recommendation on what to do if a previously sent candidate is
   unexpectedly missing in a subsequent INFO

o terminology alignment with draft-ietf-ice-trickle-07

Changes from draft-ietf-mmusic-trickle-ice-sip-07

o editorial fixes

o clarification on ordering of candidates for alignment with draft-
   ietf-ice-trickle-12

o O/A procedures for end-of-candidates attribute described here
   after corresponding procedures have been removed from draft-ietf-
   ice-trickle-11

o using IPv6 addresses in examples

Changes from draft-ietf-mmusic-trickle-ice-sip-08

o editorial fixes/clarification based on Flemmings review

o Description of Trickle specifics in O/A procedures for initial O/A
   exchange and specification of ICE mismatch exception

Changes from draft-ietf-mmusic-trickle-ice-sip-09

o editorial fixes/correction of references

o adding missing Ref to RFC3605 in section 6, 5th para

o replaced remaining IPv4 adresses with IPv6

o Added text for handling a=rtcp in case of default RTP address
   0.0.0.0:9 based on comment from Roman Shpount.

Changes from draft-ietf-mmusic-trickle-ice-sip-10

o editorial fixes due to idnits output

Changes from draft-ietf-mmusic-trickle-ice-sip-11

o addressing comments from Ben Campell's AD review and Christer's
   review

o Numerous editorial improvements/corrections

o Added [RFC8174] boiler plate and adapted usage of normative
   language

o Clarified terminology ICE modules .vs. ICE agent

o Added more detailed OA procedures

o Corrected default values in m-line and usage of "a=mid:" attribute
   explicitly mentioned for offer/answer

o Removed explicit mentioning of XMPP

o Added Deployment Considerations section

o Fixed ref for rfc5245bis

Changes from draft-ietf-mmusic-trickle-ice-sip-12

o addressing comments from Gen-Art review, TSV-Art review and
   Security Directorate review

o Numerous editorial improvements/corrections/clarifications

Changes from draft-ietf-mmusic-trickle-ice-sip-13

o added expansions for SDP,GRUU, AOR, STUN, TURN

o some editorial corrections

[16](16). References

[16.1](16.1).   Normative References

[I-D.ietf-ice-rfc5245bis]
          Keranen, A., Holmberg, C., and J. Rosenberg, "Interactive
          Connectivity Establishment (ICE): A Protocol for Network
          Address Translator (NAT) Traversal", draft-ietf-ice-rfc5245bis-17 (work in progress), February 2018.

[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-16 (work in progress),
          February 2018.

[I-D.ietf-mmusic-ice-sip-sdp]
          Petit-Huguenin, M., Keranen, A., and S. Nandakumar,
          "Session Description Protocol (SDP) Offer/Answer
          procedures for Interactive Connectivity Establishment
          (ICE)", draft-ietf-mmusic-ice-sip-sdp-16 (work in
          progress), November 2017.

[I-D.ietf-mmusic-mux-exclusive]
          Holmberg, C., "Indicating Exclusive Support of RTP/RTCP
          Multiplexing using SDP", draft-ietf-mmusic-mux-exclusive-12 (work in progress), May 2017.

[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-48 (work in progress), January 2018.

[I-D.ietf-mmusic-sdp-mux-attributes]
          Nandakumar, S., "A Framework for SDP Attributes when
          Multiplexing", draft-ietf-mmusic-sdp-mux-attributes-16
          (work in progress), December 2016.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119,
              DOI 10.17487/RFC2119, March 1997,
              <https://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,
              <https://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,
              <https://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,
              <https://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,
              <https://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, <https://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,
              <https://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,
              <https://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,
              <https://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,
               <https://www.rfc-editor.org/info/rfc6086>.

   [RFC6838]   Freed, N., Klensin, J., and T. Hansen, "Media Type
               Specifications and Registration Procedures", BCP 13,
               RFC 6838, DOI 10.17487/RFC6838, January 2013,
               <https://www.rfc-editor.org/info/rfc6838>.

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

   [RFC8174]   Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC
               2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174,
               May 2017, <https://www.rfc-editor.org/info/rfc8174>.

16.2. Informative References

   [RFC3311]   Rosenberg, J., "The Session Initiation Protocol (SIP)
               UPDATE Method", RFC 3311, DOI 10.17487/RFC3311, October
               2002, <https://www.rfc-editor.org/info/rfc3311>.

   [RFC3725]   Rosenberg, J., Peterson, J., Schulzrinne, H., and G.
               Camarillo, "Best Current Practices for Third Party Call
               Control (3pcc) in the Session Initiation Protocol (SIP)",
               BCP 85, RFC 3725, DOI 10.17487/RFC3725, April 2004,
               <https://www.rfc-editor.org/info/rfc3725>.

   [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,
               <https://www.rfc-editor.org/info/rfc3840>.

   [RFC5389]   Rosenberg, J., Mahy, R., Matthews, P., and D. Wing,
               "Session Traversal Utilities for NAT (STUN)", RFC 5389,
               DOI 10.17487/RFC5389, October 2008,
               <https://www.rfc-editor.org/info/rfc5389>.

   [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,
               <https://www.rfc-editor.org/info/rfc5627>.

   [RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using
              Relays around NAT (TURN): Relay Extensions to Session
              Traversal Utilities for NAT (STUN)", RFC 5766,
              DOI 10.17487/RFC5766, April 2010,
              <https://www.rfc-editor.org/info/rfc5766>.

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