Stream: Internet Engineering Task Force (IETF)
RFC: 8840
Category: Standards Track
Published: January 2021
ISSN: 2070-1721
Authors: E. Ivov *Jitsi* T. Stach *Unaffiliated* E. Marocco *Telecom Italia* C. Holmberg *Ericsson*

RFC 8840
A Session Initiation Protocol (SIP) Usage for Incremental Provisioning of Candidates for the Interactive Connectivity Establishment (Trickle ICE)

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). The document also defines a new SIP Info Package to support this usage together with the corresponding media type. Additionally, a new Session Description Protocol (SDP) "end-of-candidates" attribute and a new SIP option tag "trickle-ice" are defined.

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc8840.

Copyright Notice

Table of Contents

1. Introduction

2. Terminology

3. Protocol Overview

  3.1. Discovery Issues

  3.2. Relationship with the Offer/Answer Model

4. Incremental Signaling of ICE Candidates

  4.1. Initial Offer/Answer Exchange

    4.1.1. Sending the Initial Offer

    4.1.2. Receiving the Initial Offer

    4.1.3. Sending the Initial Answer

    4.1.4. Receiving the Initial Answer

  4.2. Subsequent Offer/Answer Exchanges

  4.3. Establishing the Dialog

    4.3.1. Establishing Dialog State through Reliable Offer/Answer Delivery

    4.3.2. Establishing Dialog State through Unreliable Offer/Answer Delivery

    4.3.3. Initiating Trickle ICE without an SDP Answer

  4.4. Delivering Candidates in INFO Requests

5. Initial Discovery of Trickle ICE Support

  5.1. Provisioning Support for Trickle ICE

  5.2. Trickle ICE Discovery with Globally Routable User Agent URIs (GRUUs)

- 5.3. Fall Back to Half Trickle
- 6. Considerations for RTP and RTCP Multiplexing
- 7. Considerations for Media Multiplexing
- 8. SDP "end-of-candidates" Attribute
  - 8.1. Definition
  - 8.2. Offer/Answer Procedures
- 9. Content Type "application/trickle-ice-sdpfrag"
  - 9.1. Overall Description
  - 9.2. Grammar
- 10. Info Package
  - 10.1. Rationale -- Why INFO?
  - 10.2. Overall Description
  - 10.3. Applicability
  - 10.4. Info Package Name
  - 10.5. Info Package Parameters
  - 10.6. SIP Option Tags
  - 10.7. INFO Request Body Parts
  - 10.8. Info Package Usage Restrictions
  - 10.9. Rate of INFO Requests
  - 10.10. Info Package Security Considerations
- 11. Deployment Considerations
- 12. IANA Considerations
  - 12.1. SDP "end-of-candidates" Attribute
  - 12.2. Media Type "application/trickle-ice-sdpfrag"
  - 12.3. SIP Info Package "trickle-ice"
  - 12.4. SIP Option Tag "trickle-ice"
- 13. Security Considerations
- 14. References
  - 14.1. Normative References
  - 14.2. Informative References

Acknowledgements

Authors' Addresses

1. Introduction

The Interactive Connectivity Establishment (ICE) protocol [RFC8445] 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, 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. Once the candidate information is available, a third phase starts in parallel 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 [RFC8445], the three phases above happen consecutively, in a blocking way, which can introduce undesirable setup delay during session establishment. The Trickle ICE extension [RFC8838] defines generic semantics required for these ICE phases to happen in a parallel, non-blocking way and hence speeds 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 [RFC8838] 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 together with the corresponding media type, SDP attribute, and SIP option tag.

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 ICE protocol in [RFC8445] and by its Trickle ICE extension in [RFC8838]. It is assumed that the reader is familiar with the terminology from both documents.

[RFC8445] also describes how ICE makes use of the Session Traversal Utilities for NAT (STUN) protocol [RFC5389] and its extension Traversal Using Relays around NAT (TURN) [RFC5766].

3. Protocol Overview

When using ICE for SIP according to [RFC8839], 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 also allows ICE candidates 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 and 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 (GRUUs) [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 [RFC8838] provides one way around this, by requiring the first Offer to contain a complete set of local ICE candidates and using only 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. Therefore, Section 5 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 online provisioning as a means of allowing the 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 middleboxes and proxies, the Offer/Answer exchange for Trickle ICE looks partly similar to the Offer/Answer exchange for regular ICE for SIP [RFC8839]. 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 UAs. 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 [RFC8838] 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 [RFC8838].
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 already included in the initial Offer, it 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 [RFC8839].

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 RTP Control Protocol (RTCP) transport address; thus, it **MUST NOT** include the "rtcp" attribute [RFC3605]. This avoids potential ICE mismatch (see [RFC8839]) for the RTCP transport address.

If the Offerer wants to use RTCP multiplexing [RFC5761] and/or exclusive RTCP multiplexing [RFC8858], it still will include the "rtcp-mux" and/or "rctp-mux-only" attribute in the initial Offer.

In any case, the Offerer **MUST** include the "ice-options:trickle" attribute in accordance to [RFC8838] and **MUST** include in each "m=" line a "mid" attribute in accordance to [RFC5888]. The "mid" attribute identifies the "m=" line to which a candidate belongs and helps in case of multiple "m=" lines, when candidate gathering could occur in an order different from the order of the "m=" lines.

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 [RFC8838].

If the initial Offer included the "ice-options:trickle" attribute, 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 [RFC8839]).

4.1.3. Sending the Initial Answer

If the Answerer includes candidates in its initial Answer, it **MUST** encode these candidates as specified in [RFC8839].

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; thus, it **MUST NOT** include the "rtcp" attribute [RFC6086]. This avoids potential ICE mismatch (see [RFC8839]) for the RTCP transport address.

If the Answerer accepts the use of RTCP multiplexing [RFC5761] and/or exclusive RTCP multiplexing [RFC8858], it will include the "rtcp-mux" attribute in the initial Answer.

In any case, the Answerer **MUST** include the "ice-options:trickle" attribute in accordance to [RFC8838] and **MUST** include in each "m=" line 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 [RFC8838].

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 [RFC8839]).

4.2. Subsequent Offer/Answer Exchanges

Subsequent Offer/Answer exchanges are handled the same as regular ICE (see Section 4.4 of [RFC8839]).

If an Offer or Answer needs to be sent while the ICE Agents are in the middle of trickling, Section 4.4 of [RFC8839] applies. This means that an ICE Agent includes candidate attributes for all local candidates it had trickled previously for a specific media stream.

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:

- Trickle ICE support at the peer agent **MUST** be confirmed.
- A dialog **MUST** have been created between the peers.

Section 5 discusses in detail the various options for satisfying the first of the above conditions. However, regardless of those mechanisms, 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: A 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 (or lack thereof) for Trickle ICE at the Answerer 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 or 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 backoff timers described in [RFC3262]. These retransmissions **MUST** cease on receipt of an INFO request carrying a "trickle-ice" Info Package body, on receipt of any other in-dialog request from the Offerer, or on transmission of the Answer in a 2xx response. The Offerer cannot send in-dialog requests until it receives a response, so the arrival of such a request proves that the response has arrived. Using the INFO request for dialog confirmation is similar to the procedure described in Section 7.1.1 of [RFC8839], 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 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). The first case could happen when Half Trickle is used and all candidates are already in the initial offer. The second case could happen when Full Trickle is used and the Offerer is currently gathering additional candidates but did not yet get them. Also, if the initial Offer did not contain any candidates, depending on how the Offerer gathers its candidates and how long it takes to do so, this INFO could still contain no candidates.

When Full Trickle is used and if newly learned candidates are available, the Offerer **SHOULD** also deliver these candidates in said INFO request, unless it wants to hold back some candidates in reserve, e.g., in case 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 meantime 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 trickling can begin (Figure 5).

Note: The "+SRFLX" in Figure 5 indicates that additional 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 backoff timers described in [RFC3262]. These retransmissions **MUST** cease on receipt of an INFO request carrying a "trickle-ice" Info Package body, on receipt of any in-dialog requests from the Offerer, or on transmission of the Answer in a 2xx response. The Offerer cannot send in-dialog requests until it receives a response, so the arrival of such a request proves that the response has arrived. This is again similar to the procedure described in Section 6.1.1 of [RFC8839], except that an Answer is not yet provided.

Note: The "+SRFLX" in Figure 6 indicates that additional 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 an 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.

4.4. Delivering Candidates in INFO Requests

Whenever new ICE candidates become available for sending, agents encode them in "candidate" attributes as described by [RFC8839]. For example:

```
a=candidate:1 1 UDP 2130706432 200a0b:12f0::1 5000 typ host
```

The use of SIP INFO requests happens within the context of the Info Package as defined in 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 "candidate" nor the "end-of-candidates" attributes contain information that would allow correlating them to a specific "m=" line, it is handled through the use of pseudo "m=" lines.

Pseudo "m=" lines follow the SDP syntax for "m=" lines as defined in [RFC4566] and are linked to the corresponding "m=" line in the SDP Offer or Answer via the identification tag in a "mid" attribute [RFC5888]. A pseudo "m=" line does not provide 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, the content might be unknown to the agent. In this case, the agent **MUST** include the following default values:

- The media field is set to 'audio'.
- The port value is set to '9'.
- The proto value is set to 'RTP/AVP'.
- 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 "mid" attribute for every "m=" line whose candidate list they intend to update. Such "mid" attributes **MUST** immediately precede the list of candidates for that specific "m=" line.

All "candidate" or "end-of-candidates" attributes following 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 "mid" attributes since candidates are grouped in sections headed by "pseudo" "m=" lines. These sections contain "mid" attribute values that point back to the true "m=" line.

An "end-of-candidates" attribute, preceding the first 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 "end-of-candidates" attribute.

Refer to Figure 7 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 the first 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 "mid" attribute from [RFC5888], it does not define any grouping semantics.

All INFO requests **MUST** carry the "ice-pwd" and "ice-ufrag" attributes that allow mapping them to a specific ICE generation. An agent **MUST** discard any received INFO requests containing "ice-pwd" and "ice-ufrag" attributes that do not match those of the current ICE Negotiation Session.

The "ice-pwd" and "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 the first pseudo "m=" line. 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 when candidates are trickled, [RFC8838] requires that 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. That is, the agent **MUST** repeat in the INFO request body all candidates that were previously sent under the same combination of "ice-pwd" and "ice-ufrag" in the same order as they were sent before. 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.

In addition, an ICE Agent needs to adhere to Section 17 of [RFC8838] on preserving candidate order while trickling.

When receiving INFO requests carrying any candidates, agents **MUST** 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 actual 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 [RFC8838].

Receiving an "end-of-candidates" attribute in an INFO request body -- with the "ice-ufrag" and "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 the session level, i.e., before any pseudo "m=" line, this indication applies to the whole session; when included at the media level, the indication applies only to the corresponding "m=" line. Handling of such end-of-candidates indications is defined in [RFC8838].

The example in Figure 7 shows the content of a candidate delivering INFO request. In the example, the "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 2130706432 2001:db8:a0b:12f0::1 5000 typ host
a=candidate:1 2 UDP 2130706432 2001:db8:a0b:12f0::1 5001 typ host
a=candidate:1 1 UDP 2130706431 192.0.2.1 5010 typ host
a=candidate:1 2 UDP 2130706431 192.0.2.1 5011 typ host
a=candidate:2 1 UDP 1694498815 192.0.2.3 5010 typ srflx
   raddr 192.0.2.1 rport 8998
a=candidate:2 2 UDP 1694498815 192.0.2.3 5011 typ srflx
   raddr 192.0.2.1 rport 8998
a=end-of-candidates
m=audio 9 RTP/AVP 0
a=mid:2
a=candidate:1 1 UDP 2130706432 2001:db8:a0b:12f0::1 6000 typ host
a=candidate:1 2 UDP 2130706432 2001:db8:a0b:12f0::1 6001 typ host
a=candidate:1 1 UDP 2130706431 192.0.2.1 6010 typ host
a=candidate:1 2 UDP 2130706431 192.0.2.1 6011 typ host
a=candidate:2 1 UDP 1694498815 192.0.2.3 6010 typ srflx
   raddr 192.0.2.1 rport 9998
a=candidate:2 2 UDP 1694498815 192.0.2.3 6011 typ srflx
   raddr 192.0.2.1 rport 9998
a=end-of-candidates

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

Figure 7: An Example for the Content of an INFO Request

5. Initial Discovery of Trickle ICE Support

SIP UAs are required by [RFC8838] to indicate their support of and intent to use Trickle ICE in their Offers and Answers by using the "ice-options:trickle" attribute, and they **MUST** include the SIP option-tag "trickle-ice" in a SIP Supported: or Require: 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 (SBCs) 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.

However, an Offerer assuming Trickle ICE support **MUST** include a SIP Require: trickle-ice header field. That way, if the provisioned assumption of Trickle ICE support ends up being incorrect, the failure is (a) operationally easy to track down and (b) recoverable by the client, i.e., they can resend the request without the SIP Require: header field and without the assumption of Trickle ICE support.

5.2. Trickle ICE Discovery with Globally Routable User Agent URIs (GRUUs)

[RFC3840] provides a way for SIP UAs to query for support of specific capabilities using, among others, OPTIONS requests. On the other hand, support for GRUU according to [RFC5627] 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 8). 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  |
            |             Supported: trickle-ice                 |
            |                      (Offer)                       |
            |--------------------------------------------------->|
            |                                                    |
            |                  183 (Answer)                      |
            |<---------------------------------------------------|
            |                INFO/OK (Trickling)                 |
            |<-------------------------------------------------->|
            |                                                    |
            |                       ...                          |
            |                                                    |
```

Figure 8: Trickle ICE Support Discovery with OPTIONS and GRUU

Confirming support for Trickle ICE through [RFC3840] gives SIP UAs the option 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 [RFC8838]. With Half Trickle, agents initiate sessions the same way they would when using ICE for SIP [RFC8839]. 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 [RFC8839] and legacy endpoints [RFC3264].

```
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 9: Example of a Typical (Half) Trickle ICE Exchange with SIP

As a reminder, once a single Offer or Answer has 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 describes options for Trickle ICE in order to give some guidance to implementers on how trickling can be optimized with respect to providing RTCP candidates.

Handling of the "rtcp" attribute [RFC3605] and the "rtcp-mux" attribute for RTP/RTCP multiplexing [RFC5761] is already considered in Section 5.1.1.1 of [RFC8445] and in [RFC5761]. 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 [RFC8858], the procedures in that document apply for the handling of the "rtcp-mux-only", "rtcp", and "rtcp-mux" attributes.

While a Half Trickle Offerer has to send an Offer compliant to [RFC8839] and [RFC5761] including candidates for all components, the flexibility of a Full Trickle Offerer allows the sending of 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 "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 "rtcp" attribute as given in [RFC8839] 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 the 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 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 RTCP candidates or releasing already gathered RTCP candidates. If the INFO request did not contain the "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. In case the Offerer already sent RTCP candidates in a previous INFO request, it still needs to repeat them in subsequent INFO requests, even when that support for RTCP multiplexing was confirmed by the Answerer and the Offerer has released its RTCP candidates.

7. Considerations for Media Multiplexing

The following considerations describe options for Trickle ICE in order to give some guidance to implementers on how trickling can be optimized with respect to providing candidates in case of Media Multiplexing [RFC8843]. It is assumed that the reader is familiar with [RFC8843].

ICE candidate exchange is already considered in Section 10 of [RFC8843]. 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 that is 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 a "group:BUNDLE" attribute [RFC8843] at session level 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 behavior is illustrated by the following example Offer that indicates support for Media Multiplexing.

If the Offerer already sent candidates for "m=" lines in a bundle in a previous INFO request, it still needs to repeat them in subsequent INFO requests, even when that support for bundling was confirmed by the Answerer and the Offerer has released candidates that are no longer needed.

```
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 "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
```

This INFO request indicates that the Answerer supports and uses Media Multiplexing as well. Note that the Answerer only includes a single pseudo "m=" line since candidates matching those from the second "m=" line in the offer are not needed from the Answerer.

The INFO request also indicates that the Answerer accepted the suggested Offerer BUNDLE address. This allows the Offerer to omit gathering RTP and RTCP candidates for the other "m=" lines or releasing already gathered candidates. If the INFO request did not contain the "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 [RFC8859] 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 the new SDP media-level and session-level [RFC4566] "end-of-candidates" attribute. "end-of-candidates" is a property attribute [RFC4566]; hence, it 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 the session level, this indication applies to the whole session; when included at the 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 "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 "end-of-candidates" attribute together with the "ice-ufrag" and "ice-pwd" attributes of the current ICE generation as required by [RFC8838]. When included at the session level, this indication applies to the whole session; when included at the media level, the indication applies only to the corresponding media description.

Receipt of an "end-of-candidates" attribute at an Offerer or Answerer -- with the "ice-ufrag" and "ice-pwd" attributes matching the current ICE generation -- indicates that the gathering of candidates has ended at the peer, for either the session or only 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 [RFC8838].

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

9.1. Overall Description

An "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 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-sensitive %s, as defined in [RFC7405], but it also imports grammar for other SDP attributes that precede the production of [RFC7405]. A sender **SHOULD** use lower case for attributes from such earlier grammar, but a receiver **MUST** treat them as case insensitive.

```
;  Syntax
trickle-ice-sdpfrag =   session-level-fields
                  pseudo-media-descriptions
session-level-fields = *(session-level-field CRLF)

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

pseudo-media-descriptions    =  *( media-field
                           trickle-ice-attribute-fields )
trickle-ice-attribute-fields = *(trickle-ice-attribute-field CRLF)
trickle-ice-attribute-field = mid-attribute /
                        candidate-attributes /
                        ice-pwd-attribute  /
                        ice-ufrag-attribute /
                        remote-candidate-attribute /
                        end-of-candidates-attribute /
                        rtcp-attribute /
                        rtcp-mux-attribute /
                        rtcp-mux-only-attribute /
                        extension-attribute-fields
                                        ; for future extensions

rtcp-attribute                = %s"a" "=" %s"rtcp"
rtcp-mux-attribute            = %s"a" "=" %s"rtcp-mux"
rtcp-mux-only-attribute       = %s"a" "=" %s"rtcp-mux-only"
candidate-attributes          = %s"a" "=" candidate-attribute
remote-candidate-attribute    = %s"a" "=" remote-candidate-att
```

ice-lite, ice-pwd-att, remote-candidate-att, ice-ufrag-att, ice-pacing-att, ice-options, candidate-attribute, and remote-candidate-att are from [RFC8839]; identification-tag and mid-attribute are from [RFC5888]; and media-field and attribute-fields are from [RFC4566]. The "rtcp" attribute is defined in [RFC3605], the "rtcp-mux" attribute is defined in [RFC5761], and the "rtcp-mux-only" attribute is defined in [RFC8858]. The latter attributes lack formal grammar in their corresponding RFCs and are reproduced here.

The "ice-pwd" and "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.

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, the 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: Offer/Answer is defined as a strictly sequential mechanism in [RFC3264]. 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 and the average two-second backoff intervals to occur 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 GRUU [RFC5627] that, 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 UAs implementing Trickle ICE. INFO requests carry ICE candidates discovered after the peer UAs 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: or Require: 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: or Require: header field.

10.7. INFO Request Body Parts

Entities implementing this specification **MUST** include a payload of type "application/trickle-ice-sdpfrag" in SIP INFO requests as defined in Section 9.2. 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. An implementation **MUST** aggregate ICE candidates in case an unreliable transport protocol such as UDP is used. A Trickle ICE Agent **MUST NOT** have more than one INFO request pending at any one time. When INFO messages are sent over an unreliable transport, they are retransmitted according to the rules specified in [RFC3261], Section 17.1.2.1.

If the INFO requests are sent on top of TCP, which is probably the standard way, it 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 mechanisms for the 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 understand the INFO request body.

12. IANA Considerations

12.1. SDP "end-of-candidates" Attribute

This section defines a new SDP media-level and session-level [RFC4566] "end-of-candidates" attribute, which 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: RFC 8840 defines the detailed SDP Offer/Answer procedures for the "end-of-candidates" attribute.

Mux Category: IDENTICAL

Reference: RFC 8840

Example: a=end-of-candidates

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

This document defines the 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 "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 RFC 8840

Interoperability considerations: See RFC 8840

Published specification: See RFC 8840

Applications that 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 RFC 8840.

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	RFC 8840

Table 1

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" subregistry of the SIP Parameters 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.	RFC 8840

Table 2

13. Security Considerations

The Security Considerations of [RFC6086], [RFC8838], and [RFC8839] 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. References

14.1. Normative References

[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>.

[RFC8445] Keranen, A., Holmberg, C., and J. Rosenberg, "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal", RFC 8445, DOI 10.17487/RFC8445, July 2018, <https://www.rfc-editor.org/info/rfc8445>.

[RFC8838] Ivov, E., Uberti, J., and P. Saint-Andre, "Trickle ICE: Incremental Provisioning of Candidates for the Interactive Connectivity Establishment (ICE) Protocol", RFC 8838, DOI 10.17487/RFC8838, January 2021, <https://www.rfc-editor.org/info/rfc8838>.

[RFC8839] Petit-Huguenin, M., Nandakumar, S., Holmberg, C., Keränen, A., and R. Shpount, "Session Description Protocol (SDP) Offer/Answer Procedures for Interactive Connectivity Establishment (ICE)", RFC 8839, DOI 10.17487/RFC8839, January 2021, <https://www.rfc-editor.org/info/rfc8839>.

[RFC8843] Holmberg, C., Alvestrand, H., and C. Jennings, "Negotiating Media Multiplexing Using the Session Description Protocol (SDP)", RFC 8843, DOI 10.17487/RFC8843, January 2021, <https://www.rfc-editor.org/info/rfc8843>.

[RFC8858] Holmberg, C., "Indicating Exclusive Support of RTP and RTP Control Protocol (RTCP) Multiplexing Using the Session Description Protocol (SDP)", RFC 8858, DOI 10.17487/RFC8858, January 2021, <https://www.rfc-editor.org/info/rfc8858>.

[RFC8859] Nandakumar, S., "A Framework for Session Description Protocol (SDP) Attributes When Multiplexing", RFC 8859, DOI 10.17487/RFC8859, January 2021, <https://www.rfc-editor.org/info/rfc8859>.

14.2. 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>.

[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>.

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. In addition, the authors thank Benjamin Kaduk, Adam Roach, Mirja Kühlewind, and Eric Rescorla for their comments and/or text proposals for improving the document during IESG review.

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

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

The authors thank Shawn Emery for the Security Directorate review.

Authors' Addresses

Emil Ivov
Jitsi
67000 Strasbourg
France
Phone: +33 6 72 81 15 55
Email: emcho@jitsi.org

Thomas Stach
Unaffiliated
1130 Vienna
Austria
Email: thomass.stach@gmail.com

Enrico Marocco
Telecom Italia
Via G. Reiss Romoli, 274
10148 Turin
Italy
Email: enrico.marocco@telecomitalia.it

Christer Holmberg
Ericsson
Hirsalantie 11
FI-02420 Jorvas
Finland
Email: christer.holmberg@ericsson.com