

DISPATCH  
Internet-Draft  
Intended status: Standards Track  
Expires: June 10, 2013

Shi Tao. Li  
Huawei Technologies  
December 7, 2012

SIP signalling for ICE trickling  
draft-li-dispatch-ice-trickling-signalling-00

Abstract

Trickle ICE is a mechanism that allows ICE agents to send and receive candidates incrementally rather than exchanging complete lists. This document gives several solutions for trickle ICE with SIP protocol.

Note

Discussion and suggestions for improvement are requested, and should be sent to [dispatch@ietf.org](mailto:dispatch@ietf.org).

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on June 10, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must

include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Terminology . . . . .	3
3. Overview . . . . .	3
4. Solutions . . . . .	3
4.1. Using UPDATE method . . . . .	3
4.2. Using INFO method . . . . .	5
4.3. Using SUBSCRIBE/NOTIFY . . . . .	7
5. Comparison . . . . .	10
6. Security Considerations . . . . .	12
7. IANA Considerations . . . . .	12
8. Acknowledgements . . . . .	12
9. Normative References . . . . .	12
Author's Address . . . . .	13

## 1. Introduction

Trickle ICE is a mechanism that allows ICE agents to send and receive candidates incrementally rather than exchanging complete lists. The purpose of this mechanism is to reduce session establishment times, and trickle ICE had already been included in XMPP Jingle [XEP-0176] and it has been used in rtcweb.

[I-D.rescorla-mmusic-ice-trickle] describes the basic mechanism about trickle ICE. And this document illustrates several possible methods that can be used for trickle ICE with SIP protocol.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

## 3. Overview

The general description about trickle ICE as described in [I-D.rescorla-mmusic-ice-trickle] is that the initiator first include an initial ICE description (such as ICE credentials and host candidates) in their Offers. Answerers in this situation MAY send their ICE description at any point after receiving that of the offerer. After sending an ICE description each agent can continue exchanging of additional candidates

## 4. Solutions

### 4.1. Using UPDATE method

This method complies the SDP Offer/Answer model as described in RFC3264. After the first SDP O/A exchange for the initial ICE description, each side can start to send the SIP UPDATE request which contains the newly collected candidates.

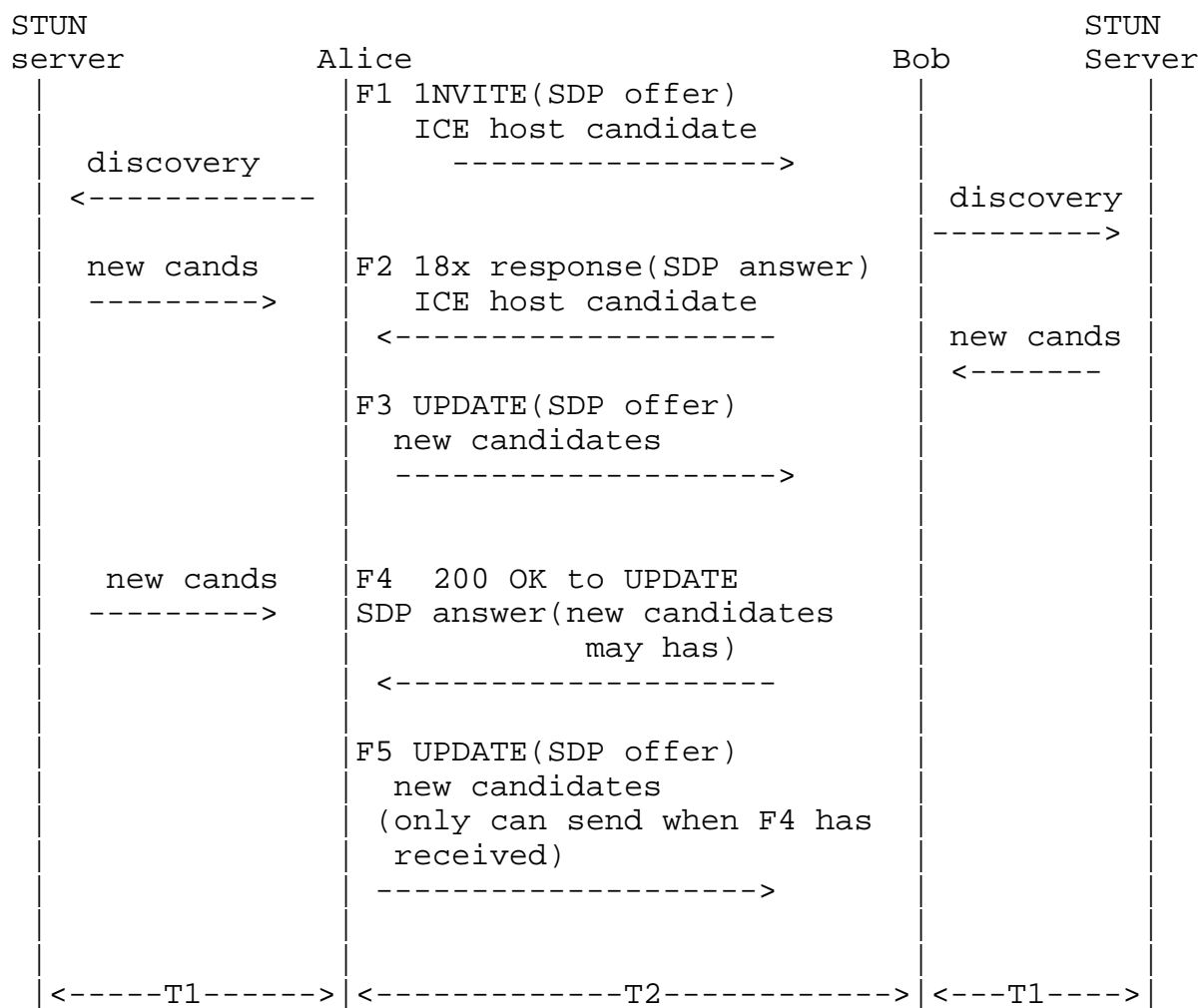


Figure 1: Using UPDATE request for ICE trickling

As described above, each side can start to send the SIP UPDATE request which contains the newly collected candidates after the first SDP O/A exchange. This may cause the "glare" problem as described in RFC3264. In SIP protocol, as specified in RFC3261, when "glare" problem happened, the user agent MUST return a 491 (Request Pending) response, then waits for the timer to expire, so it can start to send another new UPDATE request. Although the "glare" problem can be solved, it will increase the session setup times.

Another issue for the UPDATE method is that, for example, if Alice wants to send another UPDATE request (message F5 in Figure 1) to carry another new candidates, she has to wait until the SDP answer in SIP 200 OK for the previous UPDATE request (message F4 in figure 1) has received from BOB. This does not completely comply with the trickle ICE mechanism, because, trickle ICE requires the participants to send the additional candidates as soon as the new candidate has

been collected.

Assuming the candidate collecting time is  $T_1$ , message transfer time from Alice to Bob is  $T_2$ , and  $T_1$  smaller than  $2 * T_2$ .

For the best case, without glare happened:

The fastest time for Bob to receive the first additional candidates information (message F3 in Figure 1) after received the first SIP INVITE request (message F1 in Figure 1) is  $2 * T_2$ .

The fast time for Alice to receive the first additional candidates information (message F4 in Figure 1) after received the first response message (message F2 in Figure 1) is  $2 * T_2$ .

Using UPDATE method, in some cases, may has some effect to reduce the session establishment, but in most cases, it can not guarantee it works, for example, when "glare" happened.

#### 4.2. Using INFO method

This method use SIP INFO message to carry the additional candidates. Either side can start to send the SIP INFO message without any collision problem.

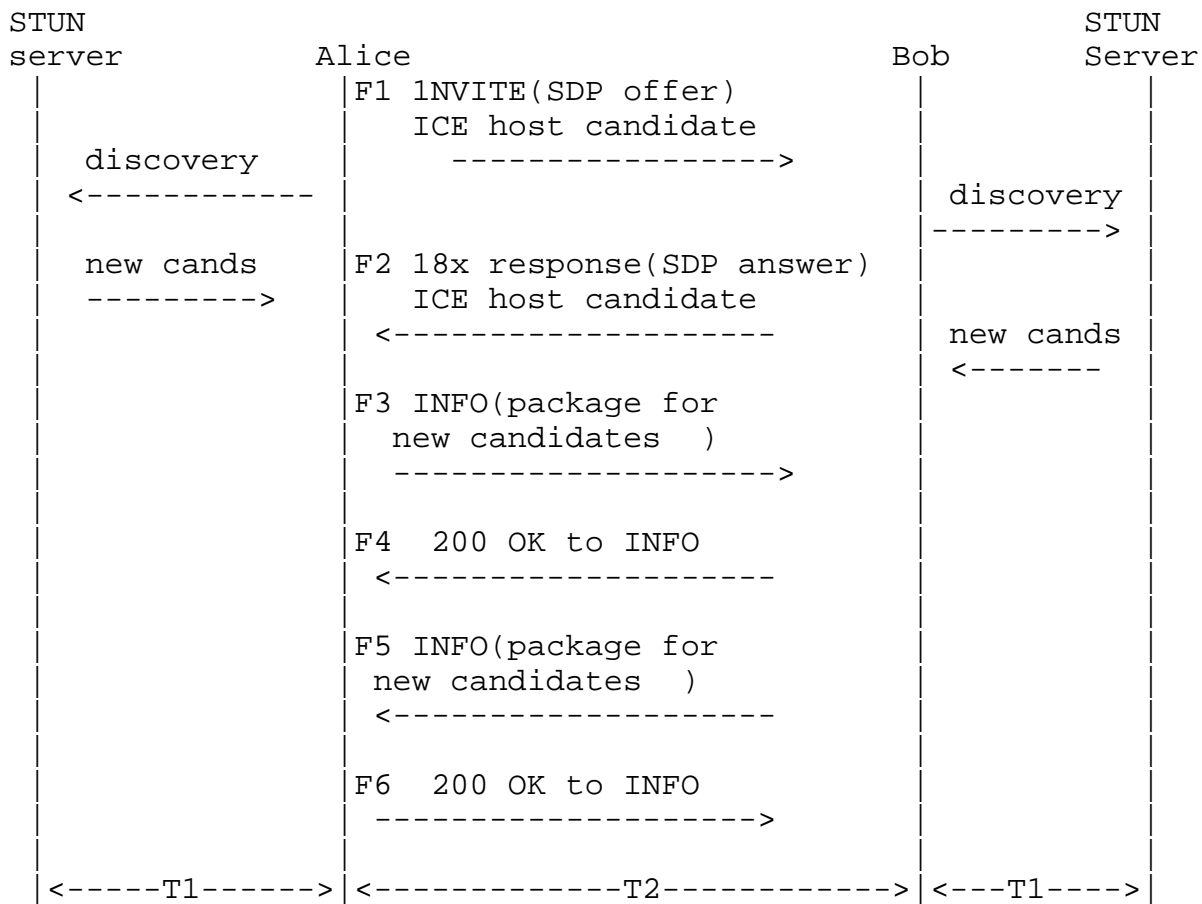


Figure 2: Using INFO request for ICE trickling

Using SIP INFO method to carry additional candidates needs to define a new info package as described in RFC6086.

The issue for INFO method is that, INFO message has to use the existing SIP dialog, that is to say, as the caller, Alice can not send the INFO message(message F3 in Figure 2) until the first response has received from Bob(message F2 in Figure 2). When the first response from Bob has received, Alice can start to send INFO message any time it needed. At Bob side, it can send the INFO message just after it send out the first response.

Using INFO method, only the candidates information needs to be exchanged between the two participants.

The only delay for session setup using INFO method is at the caller side, and only for the first INFO message. The other procedures comply the trickle ICE mechanism.

Assuming the candidate collecting time is T1, message transfer time from Alice to Bob is T2, and T1 smaller than 2\*T2.

For most cases, the fastest time for Bob to receive the first additional candidates information (messsage F3 in Figure 2) after received the first SIP INVITE request (message F1 in Figure 2) is  $2 * T2$ .

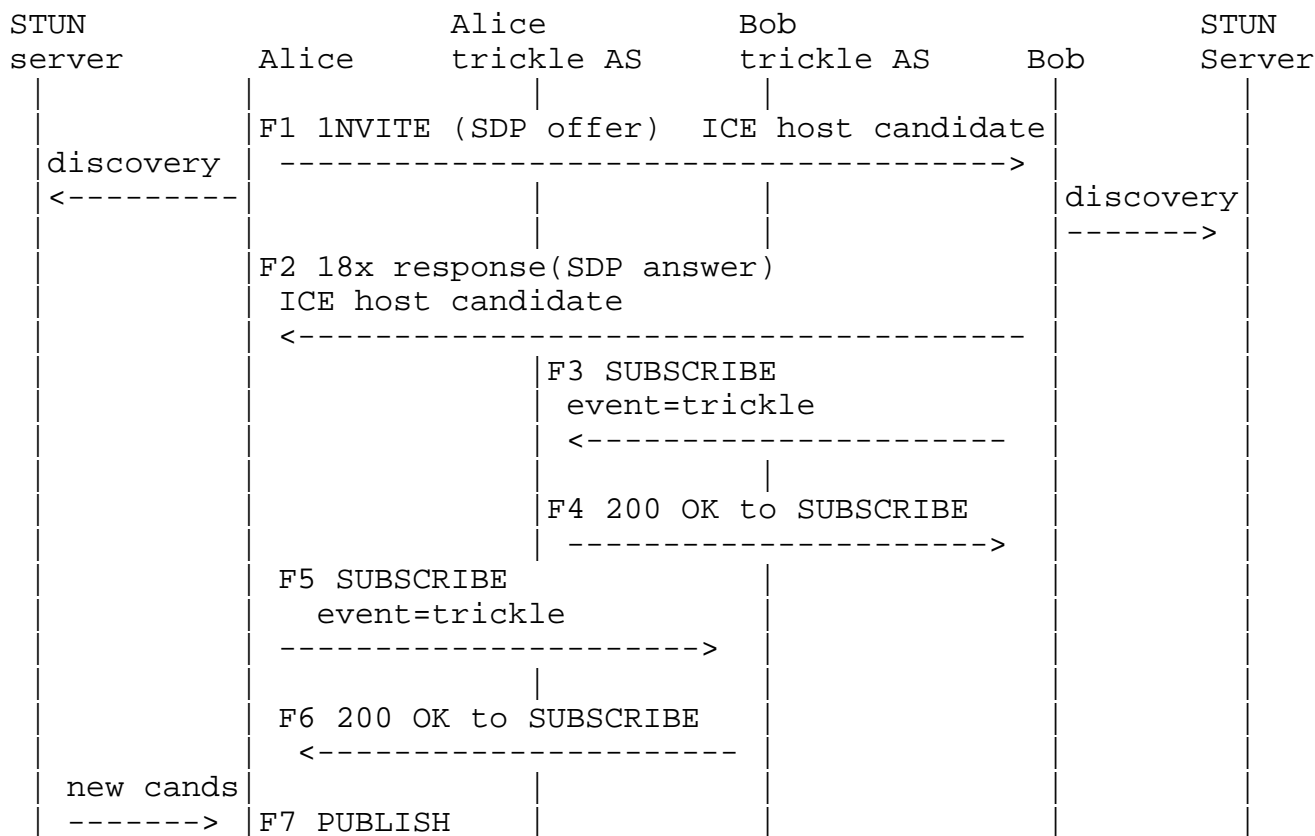
The fast time for Alice to receive the first additional candidates information (message F5 in Figure 2) after received the first response message (message F2 in Figure 2) is  $T1 + T2 - T2 = T1$ .

Using INFO method can reduce the session establishment times, has better effect than using UPDATE method.

### 4.3. Using SUBSCRIBE/NOTIFY

SUBSCRIBE/NOTIFY method is similar to the Presence service, each side has to implement a trickle AS. One participant has to subscribe to the trickle AS that belong to another participant.

At one side, whenever it collects a new candidate, it uses PUBLISH message to send the new candidates to its trickle AS, the trickle AS will then send a NOTIFY message that also contain the new candidates to the other side.



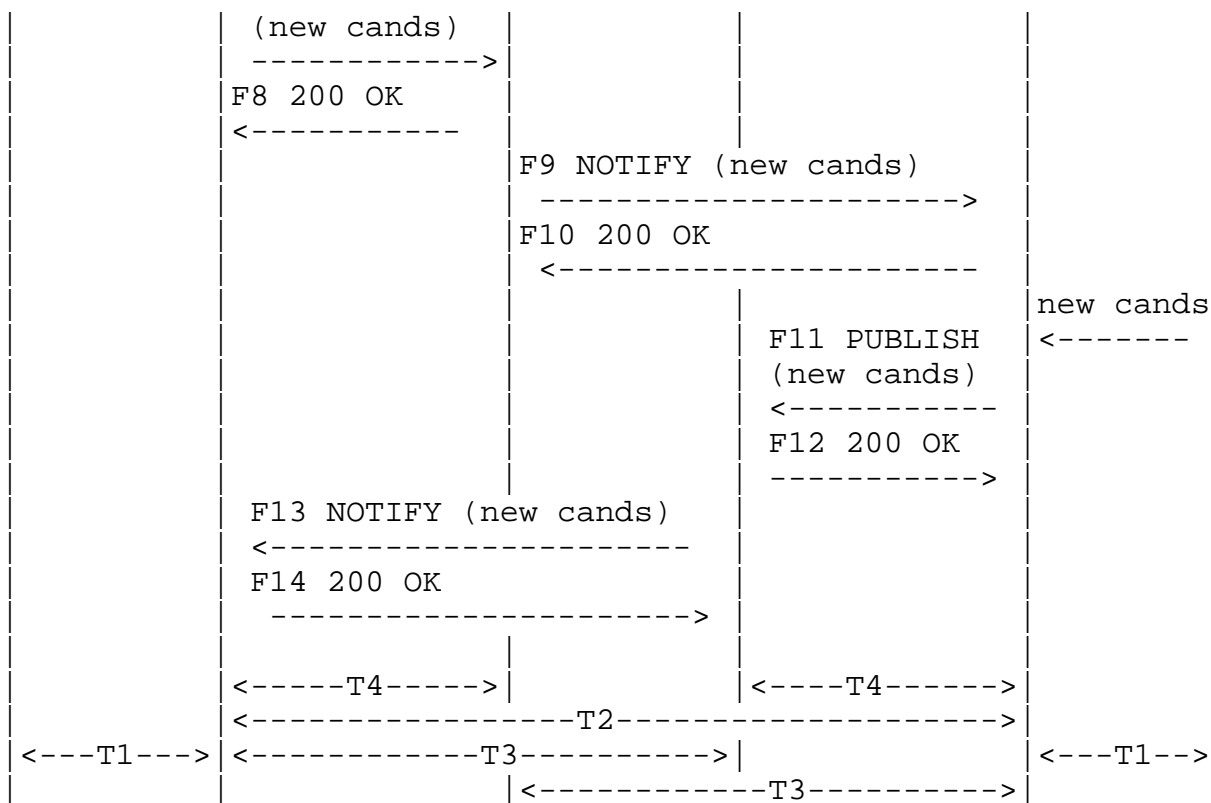


Figure 3: Using SUBSCRIBE/NOTIFY for ICE trickling

Using SUBSCRIBE/NOTIFY method needs to define a new subscribe event package as described in RFC3265.

At the callee side, Bob can start to send SUBSCRIBE request (message F3 in Figure 3) when the first SIP INVITE request has received (message F1 in Figure 3), the request-URI of the SUBSCRIBE request contains the AOR of Alice, and the event header field contains the new value "trickle", the SUBSCRIBE request will route to the trickle AS of Alice. Whenever Bob has collected a new candidate, it can send a PUBLISH request (message 11 in Figure 3) which contain the new candidate to its own trickle AS. When Bob trickle AS has received the SUBSCRIBE request from Alice (message F5 in Figure 3), it can start to send the NOTIFY request to Alice which including the new candidate information of Bob. (message F13 in Figure 3).

At the caller side, if Alice has already known that Bob supports trickle ICE before the call, Alice can send the SUBSCRIBE request (message F5 in Figure 3) just after send out the first SIP INVITE request, if Alice does know whether Bob supports trickle ICE before the call, she has to wait until it has received the first response from Bob (message F2 in Figure 3). Alice can send the PUBLISH request (message F7 in Figure 3) which contain the additional candidate to its



trickle AS only after the first SIP INVITE request (message F1 in Figure 3) has sent out.

Assuming that the candidate collecting time is  $T_1$ , message transfer time from Alice to Bob is  $T_2$ , and  $T_1$  smaller than  $2 * T_2$ , message transfer time from Alice to Bob trickle AS and the time from Bob to Alice trickle AS is  $T_3$ , obviously  $T_3$  smaller than  $T_2$ . The time from Alice to Alice trickle AS and the time from Bob to Bob trickle AS is  $T_4$ , but it can be ignored.

If Alice has already known that Bob supports trickle ICE before the call:

The fastest time for Bob to receive the first additional candidates information (message F9 in Figure 3) after received the first SIP INVITE request (message F1 in Figure 3) is happened at the case that when F7 sends after F1, and arrives at Alice trickle AS before F3, so the fastest time is  $2 * T_3$ .

The fast time for Alice to receive the first additional candidates information (message F13 in Figure 3) after received the first response message (message F2 in Figure 3) is happened at the case that when Alice send SUBSCRIBE request (F5) after F1, and do not need to wait F2, so the fastest time is  $T_2 + T_1 + T_4 + T_3 - 2 * T_2 = T_1$ .

If Alice does know whether Bob support trickle ICE before the call:

The fastest time for Bob to receive the first additional candidates information (message F9 in Figure 3) after received the first SIP INVITE request (message F1 in Figure 3) is also  $2 * T_3$ ,

The fast time for Alice to receive the first additional candidates information (message F13 in Figure 3) after received the first response message (message F2 in Figure 3) is happened at the case that when Alice sends F5 only when F2 has received, and knows that Bob support trickle ICE, so the fastest time is " $2 * T_3$ " when F11 arrives before F5 ( $T_2 + T_1 + T_4 + T_3 + (T_2 + T_3 - (T_1 + T_4)) - 2 * T_2 = 2 * T_3$ ), or " $T_1$ " when F5 arrives before F11 ( $T_2 + T_1 + T_4 + T_3 - 2 * T_2 = T_1$ ).

For the SUBSCRIBE/NOTIFY method, the effect to reduce session setup time is better than using INFO method, because PUBLISH request can be sent in a separate dialog. However, this method also has some issues if one participant is involved in multiple calls. Then there would be multiple subscriptions to the trickle event. And the trickle AS would somehow need to figure out which subscriber should receive a particular published candidate.

## 5. Comparison

This section lists some criteria for comparison for the above three methods.

	UPDATE method	INFO method	SUBSCRIBE/NOTIFY
Using SDP O/A	Yes	No	No
initiator can send new cands only when first answer has received from the remote side	Yes	Yes	No
glare issue	Yes	No	No
inforamtion exchange	All the SDP information has to exchange	Only new cands infor needs to exchange	Only new cands infor needs to exchange
needs to wait the answer so another new cands can be send	Yes	No	No
new parameter needed	No	Yes (new info package)	YES (new event)
more signalling	less signalling added	need more signalling than UPDATE method	need more signalling than INFO method
effect of increasing the setup time	Not too much	better than UPDATE	better then INFO
multiple call interworking problem	No	No	Yes

Table 1: methods comparison

## 6. Security Considerations

To do

## 7. IANA Considerations

Needed if INFO method or SUBSCRIBE/NOTIFY method has been chosen as the final method.

## 8. Acknowledgements

Thanks to Paul Kyzivat for providing useful comments to the document.

## 9. Normative References

[I-D.rescorla-mmusic-ice-trickle]

Rescorla, E., Uberti, J., and E. Ivov, "Trickle ICE: Incremental Provisioning of Candidates for the Interactive Connectivity Establishment (ICE) Protocol", draft-rescorla-mmusic-ice-trickle-01 (work in progress), October 2012.

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

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC3265] Roach, A., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.

[RFC6086] Holmberg, C., Burger, E., and H. Kaplan, "Session Initiation Protocol (SIP) INFO Method and Package Framework", RFC 6086, January 2011.

Author's Address

Shitao Li  
Huawei Technologies  
Huawei Base  
101 Software Avenue, Yuhua District  
Nanjing, Jiangsu 210012  
China

Phone: +86-25-56624157  
Email: lishitao@huawei.com