Advertising Interactive Connectivity Establishment (ICE) Services with Presence
draft-peterson-pidf-ice-00

Status of this Memo

By submitting this Internet-Draft, I certify that any applicable patent or other IPR claims of which I am aware have been disclosed, and any of which I become aware will be disclosed, in accordance with RFC 3668.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on January 10, 2005.

Copyright Notice

Copyright (C) The Internet Society (2004). All Rights Reserved.

Abstract

Many presence applications use availability information about a presentity to show a watcher their readiness to communicate. In the case of real-time multimedia communications, the readiness of the presentity to communicate does not entail that the network topography will permit communication to occur. Accordingly, this document specifies a means to integrate Interactive Connectivity Establishment (ICE) with presence. Presentities employing the Presence Information Data Format (PIDF) can use the extensions in this draft to advertise their ability to undergo a preliminary ICE check, and thus allow
watchers to instigate ICE tests to ascertain whether or not the intervening network is friendly to real-time multimedia communication. In a presence application, the results of this test could be displayed to watchers as the relative "connectivity status" of the presentity.

Table of Contents

1. Introduction .............................................. 3
2. The 'ice' Presence Capability .............................. 3
3. Using ICE Prior to Session Establishment .................. 4
4. Using the Results of ICE in a Presence Application ....... 5
5. XML Schema for the 'ice' Stanza ............................ 6
6. Security Considerations .................................... 7
7. Examples ....................................................... 7
8. IANA Considerations .......................................... 7
   Author's Address ........................................... 9
9. References .................................................... 7
9.1 Normative References ....................................... 7
9.2 Informative References ...................................... 8
A. Acknowledgments ............................................. 9
   Intellectual Property and Copyright Statements ............... 10
1. Introduction

Interactive Connectivity Establishment (ICE, [1]) is a mechanism that employs the testing algorithms of STUN (Simple Traversal of UDP for NATs, [2]) to determine how two applications on the Internet can exchange real-time multimedia traffic through network topologies that include Network Address Translators (NATs, [8]) and other impediments to direct end-to-end connectivity. It is based on the use of a higher-layer signaling protocol (such as SIP, [9]) which carries session descriptions (such as SDP, [10]) between endpoints. The description of the session, which includes the network address and port at which media will be received by the respective endpoints, serves as input to the ICE process, and is used to make a bilateral determination of endpoint reachability. The results of these ICE tests might influence the selection of a network address (when multiple network interfaces are present), the usage of relays to circumvent NATs, and so on.

In its initially-envisioned use, ICE is employed immediately prior to the establishment of a real-time media session between endpoints. For example, when a SIP INVITE is sent, and an offer/answer exchange of SDP is shared, both parties to a session employ the ICE mechanism to determine whether or not end-to-end connectivity is possible.

Presence [11]) information depicts the status and disposition of a presentity towards a particular watcher. It encompasses availability on the network (can I be reached now?), attitude towards communication (do I want to be reached now?), and to some degree application capability (what kinds of communication do I support now?). However, these traditional attributes do not give any indication of whether or not network topology allows communication to be possible.

Accordingly, this specification provides a means for presence to communicate a presentity's capability to perform ICE tests prior to any session establishment. The results of ICE tests could be rendered to the watcher to give them some idea of whether or not the initiation of a real-time multimedia session with a protocol like SIP is likely to be successful. This mechanism communicates a presentity's ability to receive ICE tests by extending the Presence Information Data Format (PIDF), a document format for transmitting presence information.

2. The 'ice' Presence Capability

This specification extends the Presence Capabilities (or 'prescaps') framework [7] for PIDF with a new element called 'ice'. Transmission of the 'ice' stanza in a PIDF presence notification takes the place
of the 'initiate' message in ICE processing. The 'ice' element contains an XML representation of all the data that would appear in an 'initiate' message - in fact, this schema is imported directly from the canonical syntax of the 'initiate' message.

The construction of the 'ice' element follows the rules of the ICE specification, section 5.4. Since the presentity is not actually formulating an SDP message with a bounded set of desired media streams, the presentity must find some other way of determining how to populate the media-streams element. When the 'ice' element is used in concert with the remainder of the prescaps framework, implementations SHOULD advertise the availability of one media stream per media capability advertised by prescaps. The advertised media capabilities of prescaps are, broadly, the top-level MIME types advertised in the m= lines of an SDP offer. Consequently, this is roughly comparable to mirroring what an SDP offer might show.

Network addresses and ports to be used in the media-streams element SHOULD be selected in the same manner as the presentity application would ordinarily select ports for constructing a session description. Note that this can entail the use of STUN or other mechanisms that help endpoints to unilaterally discover appropriate relay addresses.

3. Using ICE Prior to Session Establishment

As stated above, the transmission of the 'ice' stanza of prescaps serves the same purpose as the ICE 'initiate' message, and contains the same information. The remaining ICE procedures are largely identical to those described in the ICE specification - except that ICE tests are only performed by the responder.

Prior to sending a presence notification with an 'ice' stanza, presentities MUST perform the steps described in Section 5.1, 5.2 and 5.3 of the ICE specification. The steps in Section 5.4 are followed as the presence notification is constructed. Similarly, when a compliant watcher receives a NOTIFY containing a PIDF document with an 'ice' stanza, it MUST perform the steps described in Section 5.5 of the ICE specification. However, the watcher SHOULD NOT perform the responder steps described in Section 5.6 of the ICE specification, including the transmission of an ICE 'accept' message, nor any of the steps specified in subsequent subsections of Section 5.

While this makes the use of ICE in this context somewhat unidirectional, even the unidirectional ICE tests determine whether or not round-trip connectivity is possible. For presence, only a watcher requires that the connectivity status of their presence-list be determined and displayed; since presence subscriptions are not
necessarily reciprocal, it might not be the case that any given presentity would be interested to know the connectivity status of some of their watchers. Presentities that are interested in the connectivity status of watchers should maintain reciprocal presence subscriptions with those watchers (this situation is, in most presence applications, the norm anyway).

Note that the mechanism above will trigger an ICE test in a watcher whenever a presence notification containing a PIDF document with an 'ice' stanza is received - no more or less frequently. Thus, if the network topology surrounding the watcher or presentity changes without the transmission of any new presence notification from the presentity, the results of the ICE test may become outdated. There is, however, no reasonable way for the presentity to monitor overall network topology with respect to each potential watcher in order to time the transmission of presence notifications. This is, therefore, a limitation of the mechanism. It is RECOMMENDED that applications transmit new presence notifications when there are known changes to the manner in which they interface with the network, including expiration of DHCP leases or acquisition of new network interfaces, including VPNs. However, implementations SHOULD NOT attempt to monitor network topology through other means, and MUST NOT use constant pings or similar invasive monitoring techniques to determine the necessity of triggering a new ICE test.

On the flip side, sending presence notifications expressing ICE capability too frequently could result in redundant and unnecessary ICE tests, which could unnecessarily burden the applications or intervening network. The frequency of presence notifications is generally bounded by applications, but could have a minimum interval as low as a second. Presence notifications containing a 'ice' stanza SHOULD NOT be sent more frequently than once a minute.

4. Using the Results of ICE in a Presence Application

Once the ICE tests have been performed, a watcher's application MAY render the results of the tests to the watcher. The manner in which it does so is application-specific, and not a subject of standardization. For informative purposes, an example application result is given here.

A watcher application might express four states associated with a presentity in a presence-list: question-mark, green, yellow, and red. These states could be displayed as icons alongside the presentity's name, where:

- question-mark signifies that the presentity does not support the 'ice' prescap, or that no presence notifications have been received containing that prescap, and therefore the connectivity
status of the presentity is unknown
green signifies that there is a clean path of connectivity between
the presentity and the watcher; the establishment of real-time
multimedia sessions between the two would most likely be
successful
yellow signifies that there are network or transport-layer
disparities between the presentity and the watcher, but that known
and available relays can be used to traverse these disparities;
connectivity is possible through these relays
red signifies that there are network or transport-layer
disparities between the presentity and the watcher than cannot be
circumvented with the tools known to the watcher; new relays or
other middleboxes would need to be engaged to make real-time
multimedia sessions possible

When a presentity publishes presence from multiple sources, rendering
of the results of an ICE test naturally lends itself to the 'device
view' of presence. For example, it could be used in concert with
other preference information in presence to prioritize the devices at
which the presentity could be reached.

5. XML Schema for the 'ice' Stanza

The ICE specification already gives an XML Schema for the ICE
'initiate' message (in Section 7). Accordingly, this schema is
imported into the 'ice' prescaps element. Note that the only valid
message type that can appear in the 'ice' prescap element is the
'initiate' message.

```xml
<?xml version="1.0" encoding="UTF-8"?>
    xmlns:icens="urn:ietf:params:xml:ns:ice"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    elementFormDefault="qualified"
    attributeFormDefault="unqualified">
    <xs:element name="ice" type="tns:ice-elem"/>
</xs:schema>
```
6. Security Considerations

Providing network addresses and ports in presence potentially exposes device information that end users might not want to divulge. In this respect, the information provided in the 'ice' presence capability shares the privacy concerns common to most forms of presence information. Fortunately, presence authorization has been well-studied, and numerous mechanisms exist to prevent the distribution of presence attributes to undesired parties. Implementers and end-users of presence applications employing this mechanism should be careful to treat the 'ice' presence capability with the same privacy preferences as other forms of presence. Presence notifications in SIP can supply confidentiality properties (through baseline security mechanisms, including S/MIME) that prevent eavesdroppers from overhearing addresses sent to authorization watchers.

The Security Considerations of the ICE specification note the importance of integrity protection to the ICE process. Without integrity protection, it would be possible for an attacker to modify the 'ice' stanza of a presence notification in transit, and by substituting bad addresses the attacker might persuade the watcher that connectivity with the presentity is impossible. Presence notifications in SIP can supply integrity properties (through baseline security mechanisms, including S/MIME) that prevent men-in-the-middle from modifying the contents of a presence notification.

7. Examples

[Ed. Note: TBD]

8. IANA Considerations

This document registers a new XML namespaces for the 'ice' stanza of the prescaps extension to PIDF.

[Ed. Note: Registration details TBD]

9. References

9.1 Normative References

9.2 Informative References


Appendix A. Acknowledgments

Thanks to Jonathan Rosenberg for some useful advice on ICE.
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2004). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.