STRAW
Internet-Draft
Intended status: Standards Track
Expires: January 5, 2015

R. Ravindranath
T. Reddy
G. Salgueiro
Cisco

July 4, 2014

Session Traversal Utilities for NAT (STUN) Message Handling for Session Initiation Protocol (SIP) Back-to-Back User Agents (B2BUAs)

draft-ram-straw-b2bua-stun-00

Abstract

Session Initiation Protocol (SIP) Back-to-Back User Agents (B2BUAs) are often designed to be on the media path, rather than just intercepting signaling. This means that B2BUAs often act on the media path leading to separate media legs that the B2BUA correlates and bridges together. When acting on the media path, B2BUAs are likely to receive Session Traversal Utilities for NAT (STUN) packets as part of Interactive Connectivity Establishment (ICE) processing. It is critical that B2BUAs handle these STUN messages properly.

This document defines behavior for a B2BUA performing ICE processing.

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 January 5, 2015.

Copyright Notice

Table of Contents

1. Introduction

In many SIP deployments, SIP entities exist in the SIP signaling path between the originating and final terminating endpoints, which go beyond the definition of a SIP proxy, performing functions not defined in Standards Track RFCs. These SIP entities, commonly known as Back-to-Back User Agents (B2BUAs) are described in [RFC7092].

The Session Initiation Protocol (SIP) [RFC3261], and other session control protocols that try to use direct path for media, are typically difficult to use across Network Address Translators (NATs). These protocols use IP addresses and transport port numbers encoded in the signaling, such as the Session Description Protocol (SDP) [RFC4566] and, in the case of SIP, various header fields. Such addresses and ports are unreachable unless all peers in a session are located behind the same NAT.

Mechanisms such as Session Traversal Utilities for NAT (STUN) [RFC5389], Traversal Using Relays around NAT (TURN) [RFC5766], and Interactive Connectivity Establishment (ICE) [RFC5245] did not exist when protocols like SIP began being deployed. Some mechanisms, such as the early versions of STUN [RFC3489], started appearing but they were unreliable and suffered a number of issues typical for

UNilateral Self-Address Fixing (UNSAF) and described in [RFC3424]. For these and other reasons, Session Border Controllers (SBCs) that were already being used by SIP domains for other SIP and media-related purposes began to use proprietary mechanisms to enable SIP devices behind NATs to communicate across the NAT. [I-D.ietf-mmusic-latching] describes how B2BUAs can perform Hosted NAT Traversal (HNT) to solve the NAT traversal problem.

Section 5 of [I-D.ietf-mmusic-latching] describes some of the issues with SBCs implementing HNT and offers some mitigation strategies. The most commonly used approach to solve these issues is "restricted-latching", whereby the B2BUA will not latch to any packets from a source public IP address other than the one the SIP UA uses for SIP signaling. However, this is susceptible to attacks, where an attacker who is able to see the source IP address of the SIP UA may generate packets using the same IP address. There are other threats described in Section 5 of [I-D.ietf-mmusic-latching] for which Secure Real-time Transport Protocol (SRTP) can be used as a solution. However, this would require the B2BUAs to be terminating/re-originating SRTP, which is not always possible. A B2BUA can use ICE [RFC5245], which provides authentication tokens (conveyed in the ice-ufrag and ice-pwd attributes) that allow the identity of a peer to be confirmed before engaging in media exchange. This can solve some of the security concerns with HNT solution. Further, ICE has other benefits like selecting an address when more than one address is available (e.g. dual-stack), verifying that a path works before connecting the call etc. For these reasons endpoints often use ICE to pick a candidate pair for media traffic between two agents.

B2BUAs often operate on the media path and have the ability to modify SIP headers and SDP bodies as part of their normal operation. Such entities, when present on the media path, are likely to take an active role in the session signaling depending on their level of activity on the media path. For example, some B2BUAs modify portions of the SDP body (e.g., IP address, port) and subsequently modify the media packet headers as well. There are other types of B2BUAs that modify the media payload (e.g., a media transcoder). Section 18.6 of ICE [RFC5245] explains two different behaviors when B2BUAs are present. Some B2BUAs are likely to remove all the SDP ICE attributes before sending the SDP across to the other side. Consequently, the call will appear to both endpoints as though the other side doesn't support ICE. There are other types of B2BUAs that pass the ICE attributes without modification, yet modify the default destination for media (contained in the m= and c= lines and rtcp attribute) This will be detected as an ICE mismatch and ICE processing is aborted for the call. The call may continue if the endpoints are able to reach each other over the default candidate (sent in m= and c= lines).

[RFC7092] describes three different categories of such B2BUAs, according to the level of activities performed on the media plane:

A B2BUA that acts as a simple media relay effectively unaware of anything that is transported and only modifies the transport header (could be UDP/IP) of the media packets.

A B2BUA that performs a media-aware role. It inspects and potentially modifies RTP or RTP Control Protocol (RTCP) headers; but it does not modify the payload of RTP/RTCP.

A B2BUA that performs a media-termination role and operates at the media payload layer, such as RTP/RTCP payload (e.g., a transcoder).

When such a B2BUA operating on a media plane is involved in a call between two endpoints performing ICE, then it SHOULD follow the behavior described in this specification.

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].

The following generalized terms are defined in [RFC3261], Section 6.

B2BUA: A SIP Back-to-Back User Agent, which is the logical combination of a User Agent Server (UAS) and User Agent Client (UAC).

UAS: A SIP User Agent Server.

UAC: A SIP User Agent Client.

All of the pertinent B2BUA terminology and taxonomy used in this document is based on [RFC7092].

Network Address Translators (NATs) are widely used in the Internet by consumers and organizations. Although specific NAT behaviors vary, this document uses the term "NAT", which maps to NAT and NAPT terms from [RFC3022], for devices that map any IPv4 or IPv6 address and transport port number to another IPv4 or IPv6 address and transport port number. This includes consumer NATs, Firewall-NATs, IPv4-IPv6 NATs, Carrier-Grade NATs (CGNs) [RFC6888], etc.

3. Media Plane B2BUAs

3.1. Overview

When one or both of the endpoints are behind a NAT, and there is a B2BUA between the endpoints, the B2BUAs MUST support ICE or at a minimum support ICE LITE functionality as described in [RFC5245]. Such B2BUAs MUST use the mechanism described in Section 2.2 of [RFC5245] to demultiplex STUN packets that arrive on the Real-time Transport Protocol(RTP)/RTP Control Protocol (RTCP) port.

The subsequent sections describe the behavior B2BUA's MUST follow for handling ICE messages. A B2BUA can terminate ICE and thus have two ICE contexts with either endpoint. The reason for ICE termination could be due to the need for B2BUA to be in the media path (e.g., media transcoding, media recording, address hiding etc.) A B2BUA can also be in ICE passthrough mode and passes across the candidate list from one endpoint to the other side. A B2BUA may be in ICE passthrough mode when it does not have a need to be on the media path. The below sections describes the behaviors for these two cases.

3.2. ICE Termination with B2BUA

A B2BUA that wishes to be in the media path follows the below steps:

When a B2BUA sends out SDP, it MUST advertise support for ICE and MAY include B2BUA candidates of different types for each component of each media stream.

If the B2BUA is in ICE lite mode as described in section 2.7 of [RFC5245] then it MUST send a=ice-lite attribute and MUST include B2BUAs host candidates for each component of each media stream.

If the B2BUA supports full ICE then it MAY include B2BUAs candidates of different types for each component of each media stream.

The B2BUA MUST generate new username, password values for ice-ufrag and ice-pwd attributes when it sends out the SDP and MUST NOT propagate the ufrag, password values it received in the incoming SDP. This ensures that the short-term credentials used for both the legs are different. The short-term credentials include authentication tokens (conveyed in the ice-ufrag and ice-pwd attributes), which the B2BUA can use to verify the identity of the peer. B2BUA terminates the ICE messages on each leg and does not propagate them.

The B2BUA MUST NOT propagate the candidate list received in the incoming SDP to the outbound SDP and instead only advertise its candidate list. In this way the B2BUA will be always in media path.

Depending on whether the B2BUA supports ICE lite or full ICE it implements the appropriate procedures mentioned in [RFC5245] for ICE connectivity checks.

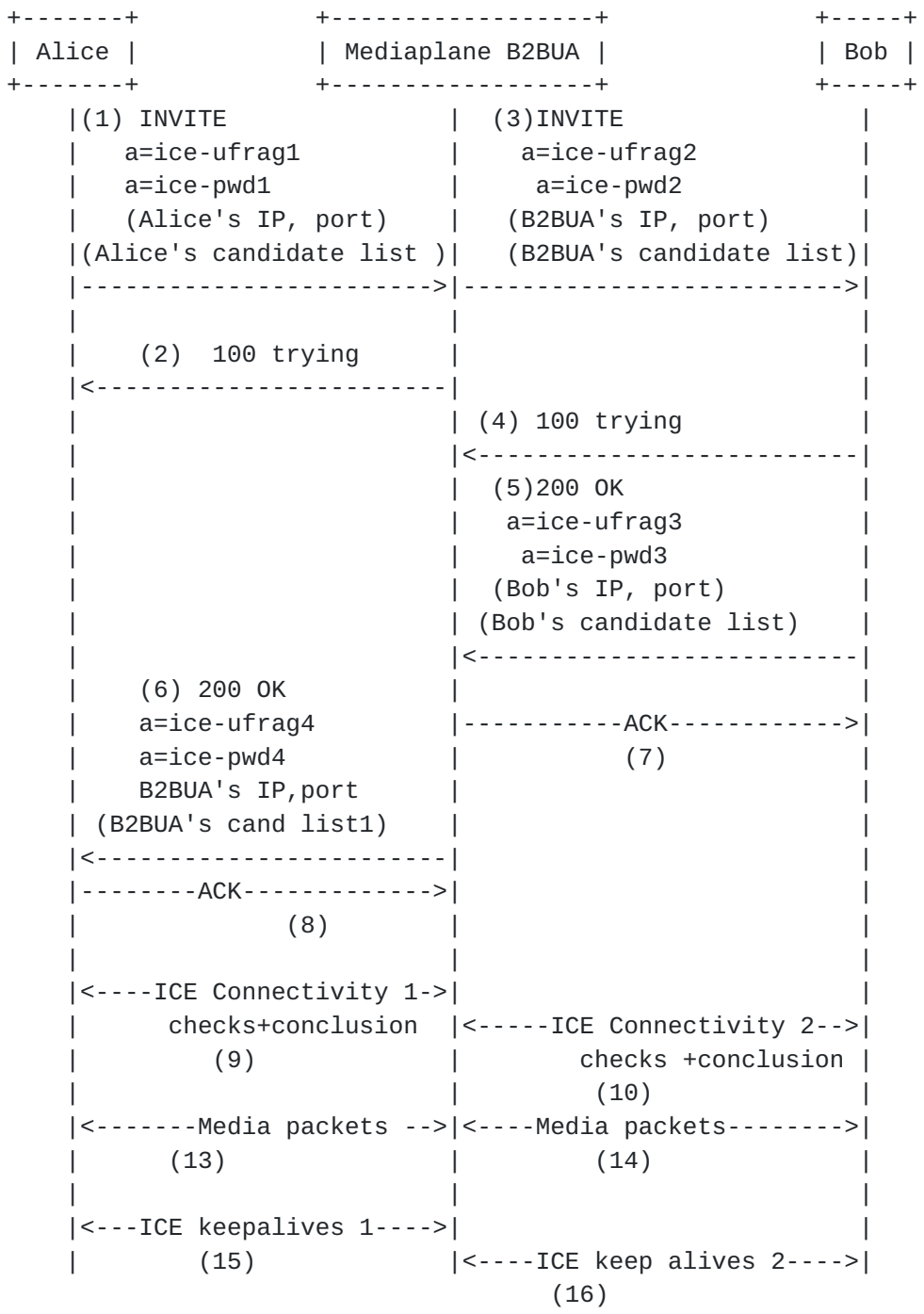

Figure 1: INVITE with SDP having ICE and with a Media Plane B2BUA

The above figure shows a sample call flow with two endpoints Alice and Bob doing ICE and a B2BUA handing STUN messages from both the endpoints. For the sake of brevity the entire ICE SDP attributes are not shown. Also the STUN messages exchanged as part of ICE connectivity checks are not shown. Key steps to note from the call flow are:

1. Alice sends an INVITE with SDP having ICE candidates.

2. B2BUA modifies the received SDP from Alice by removing the received candidate list, gathers its own candidates, generates new username, password values for ice-ufrag and ice-pwd attributes and forwards the INVITE (3) to Bob.

3. Bobs responds (5) to the INVITE with his own list of candidates.

4. B2BUA responds to the INVITE from Alice with SDP having B2BUA's candidate list. B2BUA generates new username, password values for ice-ufrag and ice-pwd attributes in the 200 OK response (6).

5. ICE Connectivity checks happen between Alice and the B2BUA in step 9. Depending on whether the B2BUA supports ICE or ICE lite it will follow the appropriate procedures mentioned in [RFC5245]. ICE Connectivity checks also happen between Bob and the B2BUA in step 10. Step 9 and 10 happen in parallel. The B2BUA always terminates the ICE messages on each leg and have two independent ICE contexts running.

6. Media flows between Alice and Bob via B2BUA (Step 13, 14).

7. STUN keepalives would be used between Alice and B2BUA (step 15) and between Bob and B2BUA (step 16) to keep NAT, Firewall bindings alive.

Since there are two independent ICE contexts on either side of the B2BUA it is possible that ICE checks will conclude on one side before concluding on the other side. This could result in an ongoing media session for one end, while the other is still being set up. Any such media received by the B2BUA would continue to be sent to the other side on the default candidate address (that was sent in c= line).

3.3. ICE Passthrough with B2BUAs

If a B2BUA does not see a need to be in media path, it can do the following steps mentioned in this section.

When a B2BUA receives an incoming SDP with ICE semantics it copies the received candidate list, adds its own candidate list in the outgoing SDP. The B2BUA also copies the ufrag/password values it received in the incoming SDP to the outgoing SDP and then sends out the SDP.

The B2BUAs candidates will have lower-priority than the candidates provided by the endpoint, this way endpoint and remote peer

candidate pairs are tested first before trying candidate pairs with B2BUA candidates.

After offer/answer is complete, the endpoints will have both the B2BUA's and remote peer candidates. It will then use ICE procedures described in [RFC5245] to nominate a candidate pair for sending and receiving media streams.

With this approach the B2BUA will be in media path only if the ICE checks between all the candidate pairs formed from the both the endpoints fails.

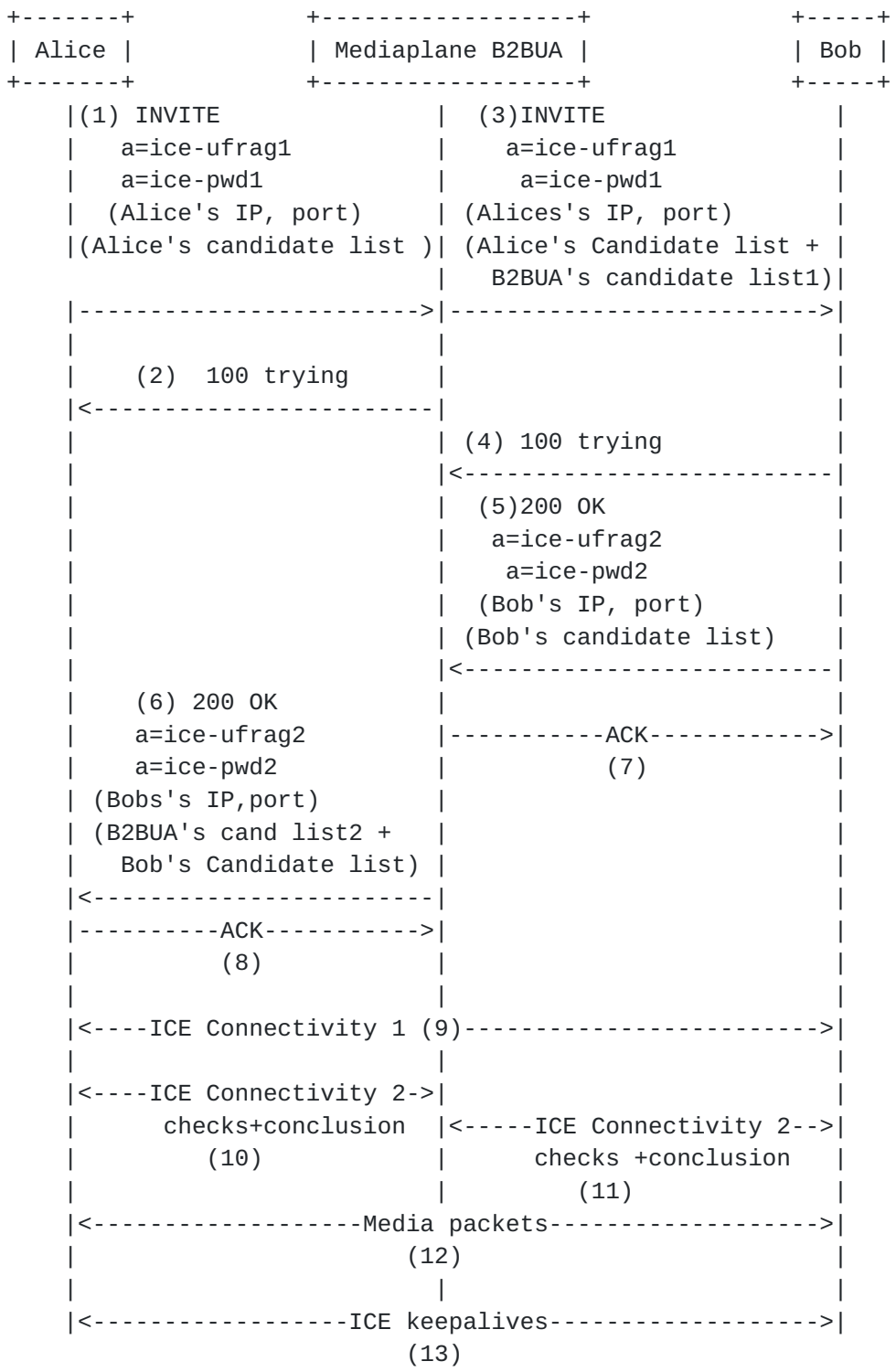

Figure 2: INVITE with SDP having ICE and with a Media Plane B2BUA in ICE Passthrough mode

The above figure shows a sample call flow with two endpoints Alice and Bob doing ICE and a B2BUA handing STUN messages from both the

endpoints. For the sake of brevity the entire ICE SDP attributes are not shown. Also the STUN messages exchanged as part of ICE connectivity checks are not shown. Key steps to note from the call flow are:

1. Alice sends an INVITE with an SDP having its own candidate list.

2. B2BUA propagates the received candidate list in incoming SDP from Alice after adding its own candidate list. The B2BUA also propagates the received ice-ufrag, ice-password attributes from Alice in the INVITE (3) to Bob.

3. Bob responds (5) to the INVITE with his own list of candidates.

4. B2BUA responds to the INVITE from Alice with an SDP having B2BUA's candidate list and the candidate list received from Bob. The B2BUA would also propagate the received ice-ufrag, ice-password attributes from Bob in step (5) to Alice in the 200 OK response (6).

5. ICE Connectivity checks happen between Alice and Bob in step 9. ICE Connectivity checks also happens between Alice and B2BUA and Bob and B2BUA as shown in step 10, 11. Step 9, 10 and 11 happen in parallel. In this example Alice and Bob conclude ICE with a candidate pair that enables them to send media directly.

6. Media flows between Alice and Bob in Step 12.

3.4. STUN Handling in B2BUA with Forked Signaling

Because of forking a B2BUA may receive multiple answers for a single outbound INVITE. When this occurs the B2BUA should follow section 3.2 or 3.3 for all of those received answers.

4. Security Considerations

TBD

5. IANA Considerations

This document makes no request of IANA.

6. Acknowledgments

Special thanks to Dan Wing, Pal Martinsen, Charles Eckel, Marc Petit-Huguenin, Simon Perreault and Lorenzo Miniero for their constructive comments, suggestions, and early reviews that were critical to the formulation and refinement of this document.

7. References

7.1. Normative References

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

[RFC3424] Daigle, L. and IAB, "IAB Considerations for UNilateral Self-Address Fixing (UNSAF) Across Network Address Translation", RFC 3424, November 2002.

[RFC3489] Rosenberg, J., Weinberger, J., Huitema, C., and R. Mahy, "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", RFC 3489, March 2003.

[RFC3711] Baugher, M., McGrew, D., Naslund, M., Carrara, E., and K. Norrman, "The Secure Real-time Transport Protocol (SRTP)", RFC 3711, March 2004.

[RFC4086] Eastlake, D., Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, June 2005.

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

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[RFC5763] Fischl, J., Tschofenig, H., and E. Rescorla, "Framework for Establishing a Secure Real-time Transport Protocol (SRTP) Security Context Using Datagram Transport Layer Security (DTLS)", RFC 5763, May 2010.

[RFC5764] McGrew, D. and E. Rescorla, "Datagram Transport Layer Security (DTLS) Extension to Establish Keys for the Secure Real-time Transport Protocol (SRTP)", RFC 5764, May 2010.

[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, April 2010.

7.2. Informative References

[I-D.ietf-mmusic-latching]
Ivov, E., Kaplan, H., and D. Wing, "Latching: Hosted NAT Traversal (HNT) for Media in Real-Time Communication", draft-ietf-mmusic-latching-08 (work in progress), June 2014.

[I-D.ram-straw-b2bua-dtls-srtp]
R, R., Reddy, T., Salgueiro, G., and V. Pascual, "DTLS-SRTP Handling in Session Initiation Protocol (SIP) Back-to-Back User Agents (B2BUAs)", draft-ram-straw-b2bua-dtls-srtp-00 (work in progress), June 2014.

[RFC3022] Srisuresh, P. and K. Egevang, "Traditional IP Network Address Translator (Traditional NAT)", RFC 3022, January 2001.

[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.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

[RFC6888] Perreault, S., Yamagata, I., Miyakawa, S., Nakagawa, A., and H. Ashida, "Common Requirements for Carrier-Grade NATs (CGNs)", BCP 127, RFC 6888, April 2013.

[RFC7092] Kaplan, H. and V. Pascual, "A Taxonomy of Session Initiation Protocol (SIP) Back-to-Back User Agents", RFC 7092, December 2013.

Authors' Addresses

Ram Mohan Ravindranath
Cisco
Cessna Business Park
Sarjapur-Marathahalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: rmohanr@cisco.com

Tirumaleswar Reddy
Cisco
Cessna Business Park, Varthur Hobli
Sarjapur Marathalli Outer Ring Road
Bangalore, Karnataka 560103
India

Email: tireddy@cisco.com

Gonzalo Salgueiro
Cisco Systems, Inc.
7200-12 Kit Creek Road
Research Triangle Park, NC 27709
US

Email: gsalguei@cisco.com