Network Working Group
Internet-Draft
Intended status: Informational
Expires: March 23, 2013

E. Ivov
Jitsi
H. Kaplan
Acme Packet
D. Wing
Cisco

September 19, 2012

Latching: Hosted NAT Traversal (HNT) for Media in Real-Time Communication
draft-ivov-mmusic-latching-01

Abstract

This document describes behavior of signalling intermediaries in RTC deployments, sometimes referred to as Session Border Controllers (SBCs), when performing Hosted NAT Traversal (HNT). HNT is a set of mechanisms, such as media relaying and latching, that such intermediaries use to enable other RTC devices behind NATs to communicate with each other. This document is non-normative, and is only written to explain HNT in order to provide a reference to the IETF community, as well as an informative description to manufacturers, and users.

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 March 23, 2013.

Table of Contents

1. Introduction

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" 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 NAPTs, Firewall-NATs, IPv4-IPv6 NATs, Carrier-Grade NATs, etc.

Protocols like SIP [RFC3261], and others that try to use a more direct path for media than with signalling, are difficult to use across NATs. They use IP addresses and transport port numbers encoded in bodies such as SDP [RFC4566]> as well as, in the case of SIP, various header fields. Such addresses and ports are unusable unless all peers in a session are located behind the same NAT.

Mechanisms such as STUN [RFC5389], TURN [RFC5766], and ICE [RFC5245], did not exist when protocols like SIP began being deployed. 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 NATs.

The term often used for this behavior is Hosted NAT Traversal (HNT), although some manufacturers sometimes use other names such as "Far-end NAT Traversal" or "NAT assist" instead. The systems which perform HNT are frequently SBCs as described in [RFC5853], although other systems such as media gateways and "media proxies" sometimes perform the same role. For the purposes of this document, all such systems are referred to as SBCs, and the NAT traversal behavior is called HNT.

As of this document's creation time, a vast majority of SIP domains use HNT to enable SIP devices to communicate across NATs, despite the publication of ICE. There are many reasons for this, but those reasons are not relevant to this document's purpose and will not be discussed. It is however worth pointing out that the current deployment levels of HNT and NATs themselves make an exclusive adoption of ICE highly unlikely in the foreseeable future.

The purpose of this document is to describe the mechanisms often used for HNT at the SDP and media layer, in order to aid understanding the implications and limitations imposed by it. Although the mechanisms used in HNT are not novel to experts, publication in an IETF document is useful as a means of providing common terminology and a reference for related documents.

In no way does this document try to make a case for HNT or present it

as a solution that is somehow better than alternatives such as ICE. The mechanisms described here, popular as they may be, are not necessarily considered best practice or recommended operation.

It is also worth mentioning that there are purely signaling-layer components of HNT as well. One such component is briefly described for SIP in [RFC5853], but that is not the focus of this document. The SIP signaling-layer component of HNT is typically more implementation-specific and deployment-specific than the SDP and media components. For the purposes of this document it is hence assumed that signaling intermediaries handle traffic in way that allows protocols such as SIP to function correctly across the NATs.

The rest of this document is going to focus primarily on use of HNT for SIP. However, the mechanisms described here are relatively generic and are often used with other protocols, such as XMPP [RFC6120], MGCP, H.248/MEGACO, and H.323.

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. Background

The general problems with NAT traversal for protocols such as SIP are:

1. The addresses and port numbers encoded in SDP bodies (or their equivalents) by NATed User Agents (UAs) are not usable across the Internet, because they represent the private addressing information of the UA rather than the addresses/ports that will be mapped to/from by the NAT.
2. The policies inherent in NATs, and explicit in Firewalls, are such that packets from outside the NAT cannot reach the UA until the UA sends packet out first.
3. Some NATs apply endpoint dependent filtering on incoming packets, as described in [RFC4787] and thus a UA may only be able to receive packets from the same remote peer IP:port as it sends packets out to.

In order to overcome these issues, signaling intermediaries such as SIP SBCs on the public side of the NATs perform HNT for both signaling and media. An example deployment model of HNT and SBCs is shown in Figure 1.

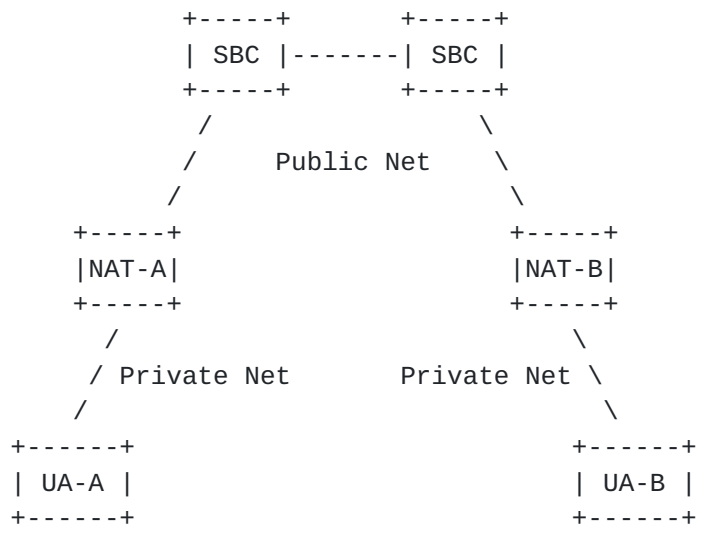

Figure 1: Logical Deployment Paths

4. Impact on Signaling

Along with codec and other media-layer information, session establishment signaling also conveys, potentially private and non-globally routable addressing information. Signaling intermediaries would hence modify such information so that peer UAs are given the (public) addressing information of a media relay controlled by the intermediary.

Quite often, the IP address of the newly introduced media relay may be the same as that of the signaling intermediary (e.g. the SIP SBC) or it may be a completely different one. In almost all cases however, the new address would belong to the same IP address family as the one used for signaling, since it is known to work for that UA.

The port numbers introduced in the signaling by the intermediary are typically allocated dynamically. Allocation strategies are entirely implementation dependent and they often vary from one product to the next.

The offer/answer media negotiation model [RFC3264] is such that once an offer is sent, the generator of the offer needs to be prepared to receive media on the advertised address/ports. In practice such media may or may not be received, depending on the implementations participating in a given session, local policies, and call scenario. For example if a SIP SDP Offer originally came from a UA behind a NAT, the SIP SBC cannot send media to it until an SDP Answer is given to the UA and latching (Section 5) occurs. Another example is when a

SIP SBC sends an SDP Offer in a SIP INVITE to a residential customer's UA and receives back SDP in a 18x response, the SBC may decide not to send media to that customer UA until a SIP 200 response for policy reasons, to prevent toll-fraud.

5. Media Behavior, Latching

An UA behind a NAT streams media from a private address:port set that once packets cross the NAT, will be mapped to a public set. The UA however is not typically aware of the public mapping and would often advertise in the private address:port couple in signaling. This way, when the signalling intermediary performing HNT receives the private addressing information from the UA it will not know what address/ ports to send media to. Therefore media relays used in HNT would often use a mechanism called "latching".

Historically, "latching" only referred to the process by which SBCs "latch" onto UDP packets from a given UA for security purposes, and "symmetric-latching" is when the latched address:ports are used to send media back to the UA. Today most people talk about them both as "latching", and thus this document does as well.

The latching mechanism works as follows:

1. After receiving an offer from a NATed UA, a signaling intermediary located on the public Internet would allocate a set of IP address:ports on a media relay. The set would then be advertised to the remote party so that it would use it for all media it wished to send toward the UA.
2. Next, after receiving an answer to its offer, the signaling server would allocate a second address:port set on the media relay. It would advertise this second set to the UA and use it for all media traffic to and from the UA.
3. The media relay receives the media packets on the allocated ports, and uses their source address and port as a destination for all media bound in the opposite direction. In other words, it "latches" or locks on these source address:port set.
4. This way all media streamed by the UA would be received on the second address:port set. The source addresses and ports of the traffic would belong to the public interface of the NAT in front of the UA and anything that the relay sends there would find its way to it.
5. Similarly the source of the stream originating at the remote party would be latched upon and used for media going in that direction.
6. Latching is usually done only once per peer and not allowed to change or cause a re-latching until a new offer and answer get exchanged.

Figure 2 describes how latching occurs for SIP where HNT is provided by an SBC connected to two networks: 38.2.2/24 facing towards the UAC network and 198.51.100/24 facing towards the UAS network.

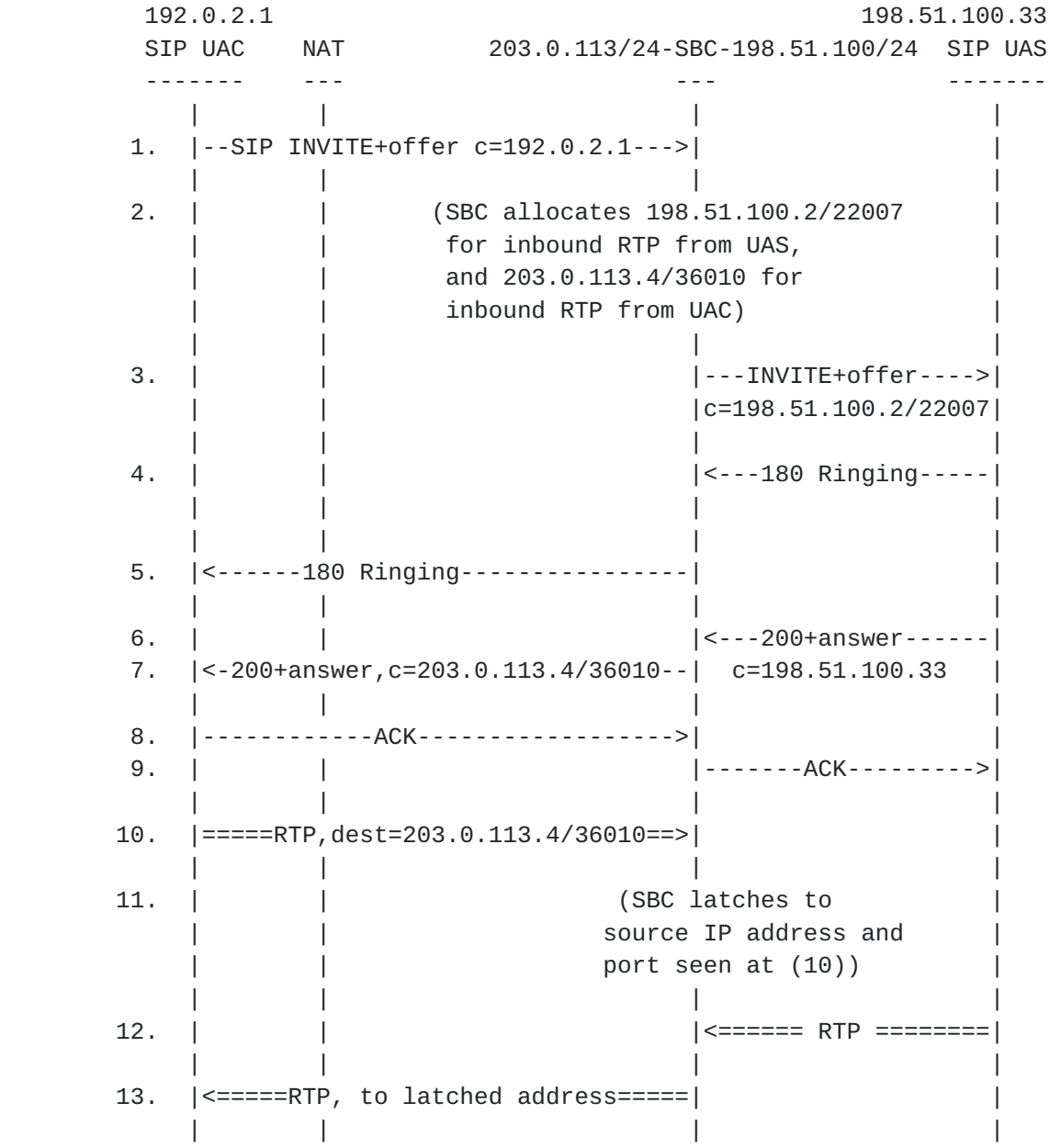

Figure 2: Latcing by a SIP SBC across two interfaces

While XMPP implementations often rely on ICE to handle NAT traversal, there are some that also support a non-ICE transport called Raw UDP [XEP-0177]. Figure 3 describes how latching occurs for one such XMPP implementate where HNT is provided by an XMPP server on the public

internet.

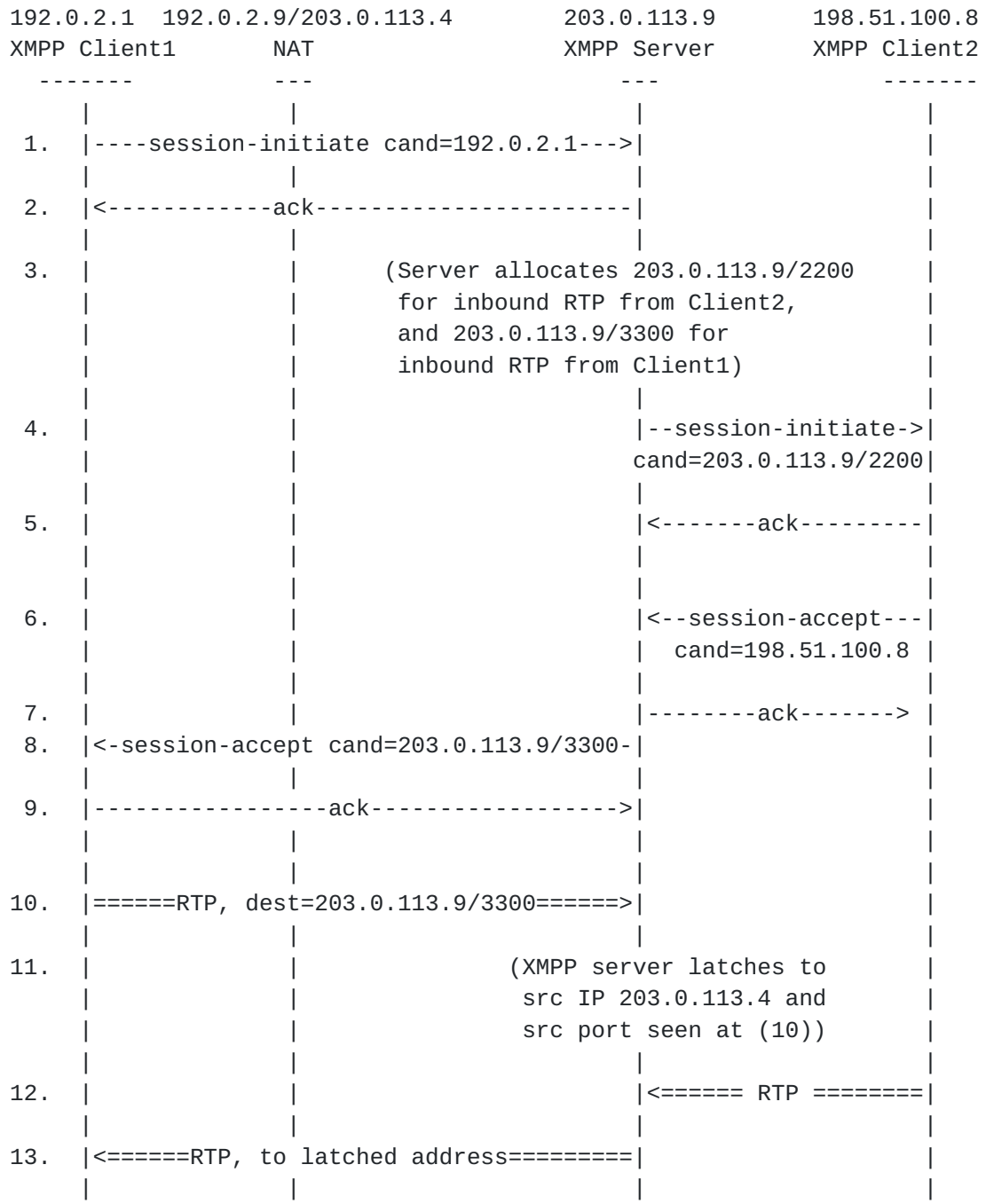

Figure 3: Latcing by a SIP SBC across two interfaces

The above is a general description, and some details vary between implementations or configuration settings. For example, some intermediaries perform additional logic before latching on received

packet source information to prevent malicious attacks or latching erroneously to previous media senders - often called "rogue-rtp" in the industry.

It is worth pointing out that latching is not an exclusively "server affair" and some clients may also use it in cases where they are configured with a public IP address and they are contacted by a NATed client with no other NAT traversal means.

In order for latching to function correctly, the UA behind the NAT needs to support symmetric RTP. That is, it needs to use the same ports for sending data as the ones it listens on for inbound packets. Today this is the case for with, for example, almost all SIP and XMPP clients. Also UAs need to make sure they can begin sending media packets independently and without waiting for packets to arrive first. In theory, it is possible that some UAs would not send packets out first; for example if a SIP session begins in 'inactive' or 'recvonly' SDP mode from the UA behind the NAT. In practice, however, SIP sessions from regular UAs (the kind that one could find behind a NAT) virtually never begin in an inactive or recvonly mode, for obvious reasons. The media direction would also be problematic if the SBC side indicated 'inactive' or 'sendonly' modes when it sent SDP to the UA. However SBCs providing HNT would always be configured to avoid this.

Given that, in order for latching to work properly, media relays need to begin receiving media before they start sending, it is possible for deadlocks to occur. This can happen when the UAC and the UAS in a session are connected to different signalling intermediaries that both provide HNT. In this case the media relays controled by the signalling servers could end up each waiting upon the other to initiate the streaming. To prevent this relays would often attempt to start streaming toward the address:port sets provided in the offer/answer even before receiving any inbound traffic. If the entity they are streaming to is another HNT performing server it would have provided its relay's public address and ports and the early stream would find its target.

Although many SBCs only support UDP-based media latching, and in particular RTP/RTCP, many SBCs support TCP-based media latching as well. TCP-based latching is more complicated, and involves forcing the UA behind the NAT to be the TCP client and sending the initial SYN-flagged TCP packet to the SBC (i.e., be the 'active' mode side of a TCP-based media session). If both UAs of a TCP-based media session are behind NATs, then SBCs typically force both UAs to be the TCP clients, and the SBC splices the TCP connections together. TCP splicing is a well-known technique, and described in [tcp-splicing].

HNT and latcing in particular are generally found to be working reliably but they do have obvious caveats. The first one usually raised by IETF members is that UAs are not aware of it occurring. This makes it impossible for the mechanism to be used with protocols such as ICE that try various traversal techniques in an effort to choose the one the best suits a particular situation. Overwriting address information in in offers and answers may actually completely prevent UAs from using ICE because of the ice-mismatch rules described in [RFC5245]

The second issue raised by IETF members is that it causes media to go through a relay instead of directly over the IP-routed path between the two participating UAs. While this adds obvious drawbacks such as reduced scalability and often increased latency, it is also considered a benefit by SBC administrators: if a customer pays for "phone" service, for example, the media is what is truly being paid for, and the administrators usually like to be able to detect that media is flowing correctly, evaluate its quality, know if and why it failed, etc. Also in some cases routing media through operator controlled relays may route media over paths explicitly optimized for media and hence offer better performance than regular Internet routing.

6. Security Considerations

A common concern is that an SBC that implements HNT may latch to incorrect and possibly malicious sources. A malicious source could, for example, attempt a resource exhaustion attack by flooding all possible media-latching UDP ports on the SBC in order to prevent calls from succeeding. SBCs have various mechanisms to prevent this from happening, or alert an administrator when it does. Still, a sufficiently sophisticated attacker may be able to bypass them for some time. The most common example is typically referred to as "restricted-latching", whereby the SBC will not latch to any packets from a source public IP other than the one the SIP UA uses for SIP signaling. In some cases the limitation may be loosened to allow media from a range of IP addresses belonging to the same network. This way the SBC simply ignores and does not latch onto packets coming from the attacker. If the attacker knows the public source IP of the legitimate SIP UA that is actually making the call, then they could still flood all of the SBC's public IP addresses and ports with packets spoofing that SIP UA's public source IP address. However, this would only disturb media from that IP (or range of IP addresses) rather than all calls that the SBC is servicing.

A malicious source could send media packets to an SBC media-latching UDP port in the hopes of being latched-to for the purpose of

receiving media for a given SIP session. SBCs have various mechanisms to prevent this as well. Restricted latching for example would also help in this case since the attacker can't make the SBC send media packets back to themselves since the SBC will not latch onto the attacker's packets. There could still be an issue if the attacker happens to be either (1) in the IP routing path and thus can spoof the same IP as the real UA and get the media coming back, in which case the attacker hardly needs to attack at all to begin with, or (2) the attacker is behind the same NAT as the legitimate SIP UA, in which case the attacker's packets will be latched-to by the SBC and the SBC will send media back to the attacker. In this latter case, which is a corner-case to begin with, in practice the legitimate SIP UA will end the call anyway, because a human user would not hear anything and will hang up. In the case of a non-human call participant, such as an answering machine, this may not happen (although many such automated UAs would also hang up when they do not receive any media). The attacker could also redirect all media to the real SIP UA after receiving it, in which case the attack would likely remain undetected and succeed. Naturally, SRTP [RFC3711] would prevent such an attack from succeeding, and should be used independently of HNT.

For SIP clients, HNT is usually transparent in the sense that the SIP UA does not know it occurs. In certain cases it may be detectable, such as when ICE is supported by the SIP UA and the SBC modifies the default connection address and media port numbers in SDP, thereby disabling ICE due to the mismatch condition. Even in that case, however, the SIP UA only knows a middlebox is relaying media, but not necessarily that it is performing latching/HNT.

In order to perform HNT, the SBC has to modify SDP to and from the SIP UA behind a NAT, and thus the SIP UA cannot use S/MIME [RFC5751], and it cannot sign a sending request or verify a received request using [RFC4474] unless the SBC re-signs the request. However it is sometimes argued that, neither S/MIME nor [RFC4474] are widely deployed and that this may not be a real concern.

From a privacy perspective, media relaying is sometimes seen as a way of protecting one's IP address and not revealing it to the remote party. That kind of IP address masking is often perceived as important. However, this is no longer an exclusive advantage of HNT since it can also be accomplished by client-controlled relaying mechanisms such as TURN [RFC5766], if the client explicitly wishes to do so.

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.

7.2. Informative References

[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.

[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.

[RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", RFC 4474, August 2006.

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

[RFC4787] Audet, F. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, January 2007.

[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.

[RFC5751] Ramsdell, B. and S. Turner, "Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 3.2 Message Specification", RFC 5751, January 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.

[RFC5853] Hautakorpi, J., Camarillo, G., Penfield, R., Hawrylyshen, A., and M. Bhatia, "Requirements from Session Initiation Protocol (SIP) Session Border Control (SBC) Deployments", RFC 5853, April 2010.

[RFC6120] Saint-Andre, P., "Extensible Messaging and Presence Protocol (XMPP): Core", RFC 6120, March 2011.

[RFC6189] Zimmermann, P., Johnston, A., and J. Callas, "ZRTP: Media Path Key Agreement for Unicast Secure RTP", RFC 6189, April 2011.

[XEP-0177]
Beda, J., Saint-Andre, P., Hildebrand, J., and S. Egan, "XEP-0177: Jingle Raw UDP Transport Method", XEP XEP-0177, December 2009.

Authors' Addresses

Emil Ivov
Jitsi
Strasbourg 67000
France

Email: emcho@jitsi.org

Hadriel Kaplan
Acme Packet
100 Crosby Drive
Bedford, MA 01730
USA

Email: hkaplan@acmepacket.com

Dan Wing
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134
USA

Email: dwing@cisco.com