

Independent Submission
Request for Comments: 6947
Category: Informational
ISSN: 2070-1721

M. Boucadair
France Telecom
H. Kaplan
Acme Packet
R. Gilman
Independent
S. Veikkolainen
Nokia
May 2013

The Session Description Protocol (SDP)
Alternate Connectivity (ALTC) Attribute

Abstract

This document proposes a mechanism that allows the same SDP offer to carry multiple IP addresses of different address families (e.g., IPv4 and IPv6). The proposed attribute, the "altc" attribute, solves the backward-compatibility problem that plagued Alternative Network Address Types (ANAT) due to their syntax.

The proposed solution is applicable to scenarios where connectivity checks are not required. If connectivity checks are required, Interactive Connectivity Establishment (ICE), as specified in [RFC 5245](#), provides such a solution.

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not a candidate for any level of Internet Standard; see [Section 2 of RFC 5741](#).

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc6947>.

Copyright Notice

Copyright (c) 2013 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.

Table of Contents

1. Introduction	3
1.1. Overall Context	3
1.2. Purpose	4
1.3. Scope	5
1.4. Requirements Language	5
2. Use Cases	5
3. Overview of the ALTC Mechanism	6
3.1. Overview	6
3.2. Rationale for the Chosen Syntax	7
4. Alternate Connectivity Attribute	8
4.1. ALTC Syntax	8
4.2. Usage and Interaction	9
4.2.1. Usage	9
4.2.2. Usage of ALTC in an SDP Answer	11
4.2.3. Interaction with ICE	11
4.2.4. Interaction with SDP-Cap-Neg	11
5. IANA Considerations	11
6. Security Considerations	12
7. Acknowledgements	12
8. References	12
8.1. Normative References	12
8.2. Informative References	12
Appendix A. ALTC Use Cases	15
A.1. Terminology	15
A.2. Multicast Use Case	16
A.3. Introducing IPv6 into SIP-Based Architectures	17
A.3.1. Avoiding Crossing CGN Devices	17
A.3.2. Basic Scenario for IPv6 SIP Service Delivery	17
A.3.3. Avoiding IPv4/IPv6 Interworking	18
A.3.4. DBE Bypass Procedure	20
A.3.5. Direct Communications between IPv6-Enabled User Agents	22

1. Introduction

1.1. Overall Context

Due to the IPv4 address exhaustion problem, IPv6 deployment is becoming an urgent need, along with the need to properly handle the coexistence of IPv6 and IPv4. The reality of IPv4-IPv6 coexistence introduces heterogeneous scenarios with combinations of IPv4 and IPv6 nodes, some of which are capable of supporting both IPv4 and IPv6 dual-stack (DS) and some of which are capable of supporting only IPv4 or only IPv6. In this context, Session Initiation Protocol (SIP) [RFC3261] User Agents (UAs) need to be able to indicate their available IP capabilities in order to increase the ability to establish successful SIP sessions, to avoid invocation of adaptation functions such as Application Layer Gateways (ALGs) and IPv4-IPv6 interconnection functions (e.g., NAT64 [RFC6146]), and to avoid using private IPv4 addresses through consumer NATs or Carrier-Grade NATs (CGNs) [RFC6888].

In the meantime, service providers are investigating scenarios to upgrade their service offering to be IPv6 capable. The current strategies involve either offering IPv6 only, for example, to mobile devices, or providing both IPv4 and IPv6, but with private IPv4 addresses that are NATed by CGNs. In the latter case, the end device may be using "normal" IPv4 and IPv6 stacks and interfaces, or it may tunnel the IPv4 packets through a Dual-Stack Lite (DS-Lite) stack that is integrated into the host [RFC6333]. In either case, the device has both address families available from a SIP and media perspective.

Regardless of the IPv6 transition strategy being used, it is obvious that there will be a need for dual-stack SIP devices to communicate with IPv4-only legacy UAs, IPv6-only UAs, and other dual-stack UAs. It may not be possible, for example, for a dual-stack UA to communicate with an IPv6-only UA unless the dual-stack UA has a means of providing the IPv6-only UA with an IPv6 address, while clearly it needs to provide a legacy IPv4-only device an IPv4 address. The communication must be possible in a backward-compatible fashion, such that IPv4-only SIP devices need not support the new mechanism to communicate with dual-stack UAs.

The current means by which multiple address families can be communicated are through ANAT [RFC4091] or ICE [RFC5245]. ANAT has serious backward-compatibility problems, as described in [RFC4092], which effectively make it unusable, and it has been deprecated by the IETF [RFC5245]. ICE at least allows interoperability with legacy devices. But, ICE is a complicated and processing-intensive mechanism and has seen limited deployment and implementation in SIP applications.

ALTC has been implemented as reported in [NAT64-EXP]. No issues have been reported in that document.

1.2. Purpose

This document proposes a new alternative: a backward-compatible syntax for indicating multiple media connection addresses and ports in an SDP offer, which can immediately be selected from and used in an SDP answer.

The proposed mechanism is independent of the model described in [RFC5939] and does not require implementation of SDP Capability Negotiations (a.k.a., SDPCapNeg) to function.

It should be noted that "backward-compatible" in this document generally refers to working with legacy IPv4-only devices. The choice has to be made, one way or the other, because to interoperate with legacy devices requires constructing SDP bodies that they would understand and support, such that they detect their local address family in the SDP connection line. It is not possible to support interworking with both legacy IPv4-only and legacy IPv6-only devices with the same SDP offer. Clearly, there are far more legacy IPv4-only devices in existence, and thus those are the ones assumed in this document. However, the syntax allows for a UA to choose which address family to be backward-compatible with, in case it has some means of determining it.

Furthermore, even for cases where both sides support the same address family, there should be a means by which the "best" address family transport is used, based on what the UAs decide. The address family that is "best" for a particular session cannot always be known a priori. For example, in some cases the IPv4 transport may be better, even if both UAs support IPv6.

The proposed solution provides the following benefits:

- o Allows a UA to signal more than one IP address (type) in the same SDP offer.
- o Is backward compatible. No parsing or semantic errors will be experienced by a legacy UA or by intermediary SIP nodes that do not understand this new mechanism.
- o Is as lightweight as possible to achieve the goal, while still allowing and interoperating with nodes that support other similar or related mechanisms.
- o Is easily deployable in managed networks.

- o Requires minimal increase of the length of the SDP offer (i.e., minimizes fragmentation risks).

ALTC may also be useful for the multicast context (e.g., Section 3.4 of [MULTRANS-FW] or Section 3.3 of [ADDR-ACQ]).

More detailed information about ALTC use cases is provided in Appendix A.

1.3. Scope

This document proposes an alternative scheme, as a replacement to the ANAT procedure [RFC4091], to carry several IP address types in the same SDP offer while preserving backward compatibility.

While two UAs communicating directly at a SIP layer clearly need to be able to support the same address family for SIP itself, current SIP deployments almost always have proxy servers or back-to-back user agents (B2BUAs) in the SIP signaling path, which can provide the necessary interworking of the IP address family at the SIP layer (e.g., [RFC6157]). SIP-layer address family interworking is out of scope of this document. Instead, this document focuses on the problem of communicating media address family capabilities in a backward-compatible fashion. Because media can go directly between two UAs, without a priori knowledge by the User Agent Client (UAC) of which address family the far-end User Agent Server (UAS) supports, it has to offer both, in a backward-compatible fashion.

1.4. Requirements Language

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 RFC 2119 [RFC2119].

2. Use Cases

The ALTC mechanism defined in this document is primarily meant for managed networks. In particular, the following use cases were explicitly considered:

- o A dual-stack UAC that initiates a SIP session without knowing the address family of the ultimate target UAS.
- o A UA that receives a SIP session request with SDP offer and that wishes to avoid using IPv4 or IPv6.
- o An IPv6-only UA that wishes to avoid using a NAT64 [RFC6146].

- o A SIP UA behind a DS-Lite CGN [RFC6333].
- o A SIP service provider or enterprise domain of an IPv4-only and/or IPv6-only UA that provides interworking by invoking IPv4-IPv6 media relays and that wishes to avoid invoking such functions and to let media go end to end as much as possible.
- o A SIP service provider or enterprise domain of a UA that communicates with other domains and that wishes either to avoid invoking IPv4-IPv6 interworking or to let media go end to end as much as possible.
- o A SIP service provider that provides transit peering services for SIP sessions that may need to modify SDP in order to provide IPv4-IPv6 interworking, but would prefer to avoid such interworking or to avoid relaying media in general, as much as possible.
- o SIP sessions that use the new mechanism when crossing legacy SDP-aware middleboxes, but that may not understand this new mechanism.

3. Overview of the ALTC Mechanism

3.1. Overview

The ALTC mechanism relies solely on the SDP offer/answer mechanism, with specific syntax to indicate alternative connection addresses. The basic concept is to use a new SDP attribute, "altc", to indicate the IP addresses for potential alternative connection addresses. The address that is most likely to get chosen for the session is in the normal "c=" line. Typically, in current operational networks, this would be an IPv4 address. The "a=altc" lines contain the alternative addresses offered for this session. This way, a dual-stack UA might encode its IPv4 address in the "c=" line, while possibly preferring to use an IPv6 address by explicitly indicating the preference order in the corresponding "a=altc" line. One of the "a=altc" lines duplicates the address contained in the "c=" line, for reasons explained in [Section 3.2](#). The SDP answerer would indicate its chosen address by simply using that address family in the "c=" line of its response.

An example of an SDP offer using this mechanism is as follows when IPv4 is considered most likely to be used for the session, but IPv6 is preferred:

```
v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=
c=IN IP4 192.0.2.1
t=0 0
m=audio 12340 RTP/AVP 0 8
a=altc:1 IP6 2001:db8::1 45678
a=altc:2 IP4 192.0.2.1 12340
```

If IPv6 were considered more likely to be used for the session, the SDP offer would be as follows:

```
v=0
o=- 25678 753849 IN IP6 2001:db8::1
s=
c=IN IP6 2001:db8::1
t=0 0
m=audio 45678 RTP/AVP 0 8
a=altc:1 IP6 2001:db8::1 45678
a=altc:2 IP4 192.0.2.1 12340
```

Since an alternative address is likely to require an alternative TCP/UDP port number as well, the new "altc" attribute includes both an IP address and a transport port number (or multiple port numbers). The ALTC mechanism does not itself support offering a different transport type (i.e., UDP vs. TCP), codec, or any other attribute. It is intended only for offering an alternative IP address and port number.

3.2. Rationale for the Chosen Syntax

The use of an "a=" attribute line is, according to [RFC4566], the primary means for extending SDP and tailoring it to particular applications or media. A compliant SDP parser will ignore the unsupported attribute lines.

The rationale for encoding the same address and port in the "a=altc" line as in the "m=" and "c=" lines is to provide detection of legacy SDP-changing middleboxes. Such systems may change the connection address and media transport port numbers, but not support this new mechanism, and thus two UAs supporting this mechanism would try to connect to the wrong addresses. Therefore, this document requires the SDP processor to proceed to the matching rules defined in [Section 4.2.1](#).

4. Alternate Connectivity Attribute

4.1. ALTC Syntax

The "altc" attribute adheres to the [RFC4566] "attribute" production. The ABNF syntax [RFC5234] of altc is provided below.

```
altc-attr = "altc" ":" att-value
att-value = altc-num SP addrtype SP connection-address SP port
           [ "/" rtcp-port ]
altc-num  = 1*DIGIT
rtcp-port = port
```

Figure 1: Connectivity Capability Attribute ABNF

The meaning of the fields are as follows:

- o altc-num: digit to uniquely refer to an address alternative. It indicates the preference order, with "1" indicated the most preferred address.
- o addrtype: the addrtype field as defined in [RFC4566] for connection data.
- o connection-address: a network address as defined in [RFC4566] corresponding to the address type specified by addrtype.
- o port: the port number to be used, as defined in [RFC4566]. Distinct port numbers may be used for each IP address type. If the specified address type does not require a port number, a value defined for that address type should be used.
- o rtcp-port: including an RTP Control Protocol (RTCP) port is optional. An RTCP port may be indicated in the alternative "c=" line when the RTCP port cannot be derived from the RTP port.

The "altc" attribute is applicable only in an SDP offer. The "altc" attribute is a media-level-only attribute and MUST NOT appear at the SDP session level. (Because it defines a port number, it is inherently tied to the media level.) There MUST NOT be more than one "altc" attribute per addrtype within each media description. This restriction is necessary so that the addrtype of the reply may be used by the offerer to determine which alternative was accepted.

The "addrtype"s of the altc MUST correspond to the "nettype" of the current connection ("c=") line.

A media description MUST contain two "altc" attributes: the alternative address and an alternative port. It must also contain an address and a port that "duplicate" the address/port information from the current "c=" and "m=" lines. Each media level MUST contain at least one such duplicate "altc" attribute, of the same IP address family, address, and transport port number as those in the SDP connection and media lines of its level. In particular, if a "c=" line appears within a media description, the addrtype and connection-address from that "c=" line MUST be used in the duplicate "altc" attribute for that media description. If a "c=" line appears only at the session level and a given media description does not have its own connection line, then the duplicate "altc" attribute for that media description MUST be the same as the session-level address information.

The "altc" attributes appearing within a media description MUST be prioritized. The explicit preference order is indicated in each line ("1" indicates the address with the highest priority). Given this rule, and the requirement that the address information provided in the "m=" line and "o=" line must be provided in an "altc" attribute as well, it is possible that the addresses in the "m=" line and "o=" line are not the preferred choice.

If the addrtype of an "altc" attribute is not compatible with the transport protocol or media format specified in the media description, that "altc" attribute MUST be ignored.

Note that "a=altc" lines describe alternative connection addresses, NOT addresses for parallel connections. When several "altc" lines are present, establishing multiple sessions MUST be avoided. Only one session is to be maintained with the remote party for the associated media description.

4.2. Usage and Interaction

4.2.1. Usage

In an SDP offer/answer model, the SDP offer includes "altc" attributes to indicate alternative connection information (i.e., address type, address and port numbers), including the "duplicate" connection information already identified in the "c=" and "m=" lines.

Additional, subsequent offers MAY include "altc" attributes again, and they may change the IP address, port numbers, and order of preference, but they MUST include a duplicate "altc" attribute for the connection and media lines in that specific subsequent offer. In other words, every offered SDP media description with an alternative address offer with an "altc" attribute has two "altc" attributes:

- one duplicating the "c=" and "m=" line information for that media description
- one for the alternative

These need not be the same as the original SDP offer.

The purpose of encoding a duplicate "altc" attribute is to allow receivers of the SDP offer to detect if a legacy SDP-changing middlebox has modified the "c=" and/or "m=" line address/port information. If the SDP answerer does not find a duplicate "altc" attribute value for which the address and port exactly match those in the "c=" line and "m=" line, the SDP answerer MUST ignore the "altc" attributes and use the "c=" and "m=" offered address/ports for the entire SDP instead, as if no "altc" attributes were present. The rationale for this is that many SDP-changing middleboxes will end the media sessions if they do not detect media flowing through them. If a middlebox modified the SDP addresses, media MUST be sent using the modified information.

Note that for RTCP, if applicable for the given media types, each side would act as if the chosen "altc" attribute's port number was in the "m=" media line. Typically, this would mean that RTCP is sent to the port number equal to "RTP port + 1", unless some other attribute determines otherwise. For example, the RTP/RTCP multiplexing mechanism defined in [RFC5761] can still be used with ALTC, such that if both sides support multiplexing, they will indicate so using the "a=rtcp-mux" attribute, as defined in [RFC5761], but the IP connection address and port they use may be chosen using the ALTC mechanism.

If the SDP offerer wishes to use the RTCP attribute defined in [RFC3605], a complication can arise, since it may not be clear which address choice the "a=rtcp" attribute applies to, relative to the choices offered by ALTC. Technically, RFC 3605 allows the address for RTCP to be indicated explicitly in the "a=rtcp" attribute itself, but this is optional and rarely used. For this reason, this document recommends using the "a=rtcp" attribute for the address choice encoded in the "m=" line and including an alternate RTCP port in the "a=altc" attribute corresponding to the alternative address. In other words, if the "a=rtcp" attribute explicitly encodes an address in its attribute, that address applies for ALTC, as per [RFC3605]. If it does not, then ALTC assumes that the "a=rtcp" attribute is for the address in the "m=" line, and the alternative "altc" attribute includes an RTCP alternate port number.

4.2.2. Usage of ALTC in an SDP Answer

The SDP answer SHOULD NOT contain "altc" attributes, because the answer's "c=" line implicitly and definitively chooses the address family from the offer and includes it in "c=" and "m=" lines of the answer. Furthermore, this avoids establishing several sessions simultaneously between the participating peers.

Any solution requiring the use of ALTC in the SDP answer SHOULD document its usage, in particular how sessions are established between the participating peers.

4.2.3. Interaction with ICE

Since ICE [RFC5245] also includes address and port number information in its candidate attributes, a potential problem arises: which one wins. Since ICE also includes specific ICE attributes in the SDP answer, the problem is easily avoided: if the SDP offerer supports both ALTC and ICE, it may include both sets of attributes in the same SDP offer. A legacy ICE-only answerer will simply ignore the "altc" attributes and use ICE. An ALTC-only answerer will ignore the ICE attributes and reply without them. An answerer that supports both MUST choose one and only one of the mechanisms to use: either ICE or ALTC. However, if the "m=" or "c=" line was changed by a middlebox, the rules for both ALTC and ICE would make the answerer revert to basic SDP semantics.

4.2.4. Interaction with SDP-Cap-Neg

The ALTC mechanism is orthogonal to SDPCapNeg [RFC5939]. If the offerer supports both ALTC and SDPCapNeg, it may offer both.

5. IANA Considerations

Per this document, the following new SDP attribute has been assigned.

SDP Attribute ("att-field"):

Attribute name	altc
Long form	Alternate Connectivity
Type of name	att-field
Type of attribute	Media level only
Subject to charset	No
Purpose	See Sections 1.2 and 3
Specification	Section 4

The contact person for this registration is Mohamed Boucadair (email: mohamed.boucadair@orange.com; phone: +33 2 99 12 43 71).

6. Security Considerations

The security implications for ALTC are effectively the same as they are for SDP in general [RFC4566].

7. Acknowledgements

Many thanks to T. Taylor, F. Andreassen, and G. Camarillo for their review and comments.

8. References

8.1. Normative References

- [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.
- [RFC3605] Huitema, C., "Real Time Control Protocol (RTCP) attribute in Session Description Protocol (SDP)", [RFC 3605](#), October 2003.
- [RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", [RFC 4566](#), July 2006.
- [RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, [RFC 5234](#), January 2008.
- [RFC5761] Perkins, C. and M. Westerlund, "Multiplexing RTP Data and Control Packets on a Single Port", [RFC 5761](#), April 2010.

8.2. Informative References

- [ADDR-ACQ] Tsou, T., Clauberg, A., Boucadair, M., Venaas, S., and Q. Sun, "Address Acquisition For Multicast Content When Source and Receiver Support Differing IP Versions", Work in Progress, January 2013.
- [ADDR-FORMAT] Boucadair, M., Ed., Qin, J., Lee, Y., Venaas, S., Li, X., and M. Xu, "IPv6 Multicast Address With Embedded IPv4 Multicast Address", Work in Progress, April 2013.

- [MULTRANS-FW]
Venaas, S., Li, X., and C. Bao, "Framework for IPv4/IPv6 Multicast Translation", Work in Progress, June 2011.
- [MULTRANS-PS]
Jacquenet, C., Boucadair, M., Lee, Y., Qin, J., Tsou, T., and Q. Sun, "IPv4-IPv6 Multicast: Problem Statement and Use Cases", Work in Progress, March 2013.
- [NAT64-EXP]
Abdesselam, M., Boucadair, M., Hasnaoui, A., and J. Queiroz, "PCP NAT64 Experiments", Work in Progress, September 2012.
- [RFC2871] Rosenberg, J. and H. Schulzrinne, "A Framework for Telephony Routing over IP", [RFC 2871](#), June 2000.
- [RFC4091] Camarillo, G. and J. Rosenberg, "The Alternative Network Address Types (ANAT) Semantics for the Session Description Protocol (SDP) Grouping Framework", [RFC 4091](#), June 2005.
- [RFC4092] Camarillo, G. and J. Rosenberg, "Usage of the Session Description Protocol (SDP) Alternative Network Address Types (ANAT) Semantics in the Session Initiation Protocol (SIP)", [RFC 4092](#), 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.
- [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.
- [RFC5939] Andreassen, F., "Session Description Protocol (SDP) Capability Negotiation", [RFC 5939](#), September 2010.
- [RFC6146] Bagnulo, M., Matthews, P., and I. van Beijnum, "Stateful NAT64: Network Address and Protocol Translation from IPv6 Clients to IPv4 Servers", [RFC 6146](#), April 2011.
- [RFC6157] Camarillo, G., El Malki, K., and V. Gurbani, "IPv6 Transition in the Session Initiation Protocol (SIP)", [RFC 6157](#), April 2011.

- [RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", [RFC 6333](#), August 2011.
- [RFC6406] Malas, D. and J. Livingood, "Session PEERing for Multimedia INTERconnect (SPEERMINT) Architecture", [RFC 6406](#), November 2011.
- [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.

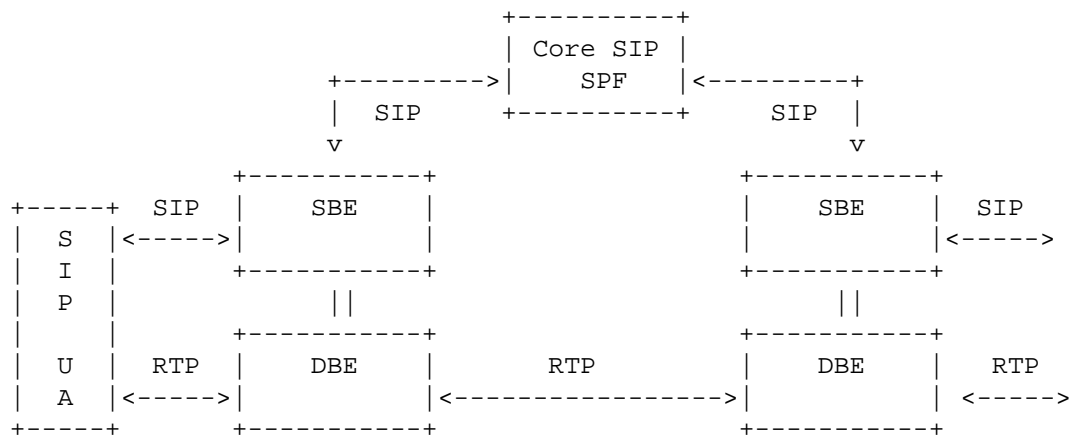
Appendix A. ALTC Use Cases

A.1. Terminology

The following terms are used when discussing the ALTC use cases:

- o SBE (Signaling Path Border Element) denotes a functional element, located at the boundaries of an ITAD (IP Telephony Administrative Domain) [RFC2871], that is responsible for intercepting signaling flows received from UAs and relaying them to the core service platform. An SBE may be located at the access segment (i.e., be the service contact point for UAs), or be located at the interconnection with adjacent domains [RFC6406]. An SBE controls one or more DBEs. The SBE and DBE may be located in the same device (e.g., the SBC [RFC5853]) or be separated.
- o DBE (Data Path Border Element) denotes a functional element, located at the boundaries of an ITAD, that is responsible for intercepting media/data flows received from UAs and relaying them to another DBE (or media servers, e.g., an announcement server or IVR). An example of a DBE is a media gateway that intercepts RTP flows. An SBE may be located at the access segment (i.e., be the service contact point for UAs) or be located at the interconnection with adjacent domains ([RFC6406]).
- o Core service platform ("core SPF") is a macro functional block including session routing, interfaces to advanced services, and access control.

Figure 2 provides an overview of the overall architecture, including the SBE, DBE, and core service platform.



SIP UA can be embedded in the CPE or in a host behind the CPE

Figure 2: Service Architecture Overview

A.2. Multicast Use Case

Recently, a significant effort has been undertaken within the IETF to specify new mechanisms to interconnect IPv6-only hosts to IPv4-only servers (e.g., [RFC6146]). This effort exclusively covered unicast transfer mode. An ongoing initiative, called "multrans", has been launched to cover multicast issues that are encountered during IPv6 transition. The overall problem statement is documented in [MULTRANS-PS].

A particular issue encountered in the context of IPv4/IPv6 coexistence and IPv6 transition of multicast services is the discovery of the multicast group and source (refer to Section 3.4 of [MULTRANS-PS]):

- o For an IPv6-only receiver requesting multicast content generated by an IPv4-only source:
 - * An ALG is required to help the IPv6 receiver select the appropriate IP address when only the IPv4 address is advertised (e.g., using SDP). Otherwise, access to the IPv4 multicast content cannot be offered to the IPv6 receiver. The ALG may be located downstream of the receiver. As such, the ALG does not know in advance whether the receiver is dual-stack or IPv6-only. The ALG may be tuned to insert both the original IPv4 address and the corresponding IPv6 multicast address using, for instance, the ALTIC SDP attribute.

- * To avoid involving an ALG in the path, an IPv4-only source can advertise both its IPv4 address and its IPv4-embedded IPv6 multicast address [[ADDR-FORMAT](#)] using, for instance, the ALTC SDP attribute.
- o For a dual-stack source sending its multicast content over IPv4 and IPv6, both IPv4 and IPv6 addresses need to be inserted in the SDP part. A means (e.g., ALTC) is needed for this purpose.

A.3. Introducing IPv6 into SIP-Based Architectures

A.3.1. Avoiding Crossing CGN Devices

Some service providers are in the process of enabling DS-Lite [[RFC6333](#)] as a means to continue delivering IPv4 services to their customers. To avoiding crossing four levels of NAT when establishing a media session (two NATs in the DS-Lite Address Family Transition Router (AFTR) and two NATs in the DBE), it is recommended to enable IPv6 functions in some SBEs/DBEs. Then, DS-Lite AFTRs will not be crossed for DS-Lite serviced customers if their UA is IPv6-enabled:

- o For a SIP UA embedded in the CPE, this is easy to implement since the SIP UA [[RFC3261](#)] can be tuned to behave as an IPv6-only UA when DS-Lite is enabled. No ALTC is required for this use case.
- o For SIP UAs located behind the CPE, a solution to indicate both IPv4 and IPv6 (e.g., ALTC) is required in order to avoid crossing the DS-Lite CGN.

A.3.2. Basic Scenario for IPv6 SIP Service Delivery

A basic solution to deliver SIP-based services using an IPv4-only core service platform to an IPv6-enabled UA is to enable the IPv4/IPv6 interworking function in the SBE/DBE. Signaling and media between two SBEs and DBEs is maintained over IPv4. IPv6 is used between an IPv6-enabled UA and an SBE/DBE.

Figure 3 shows the results of session establishment between UAs. In this scenario, the IPv4/IPv6 interworking function is invoked even when both involved UAs are IPv6-enabled.

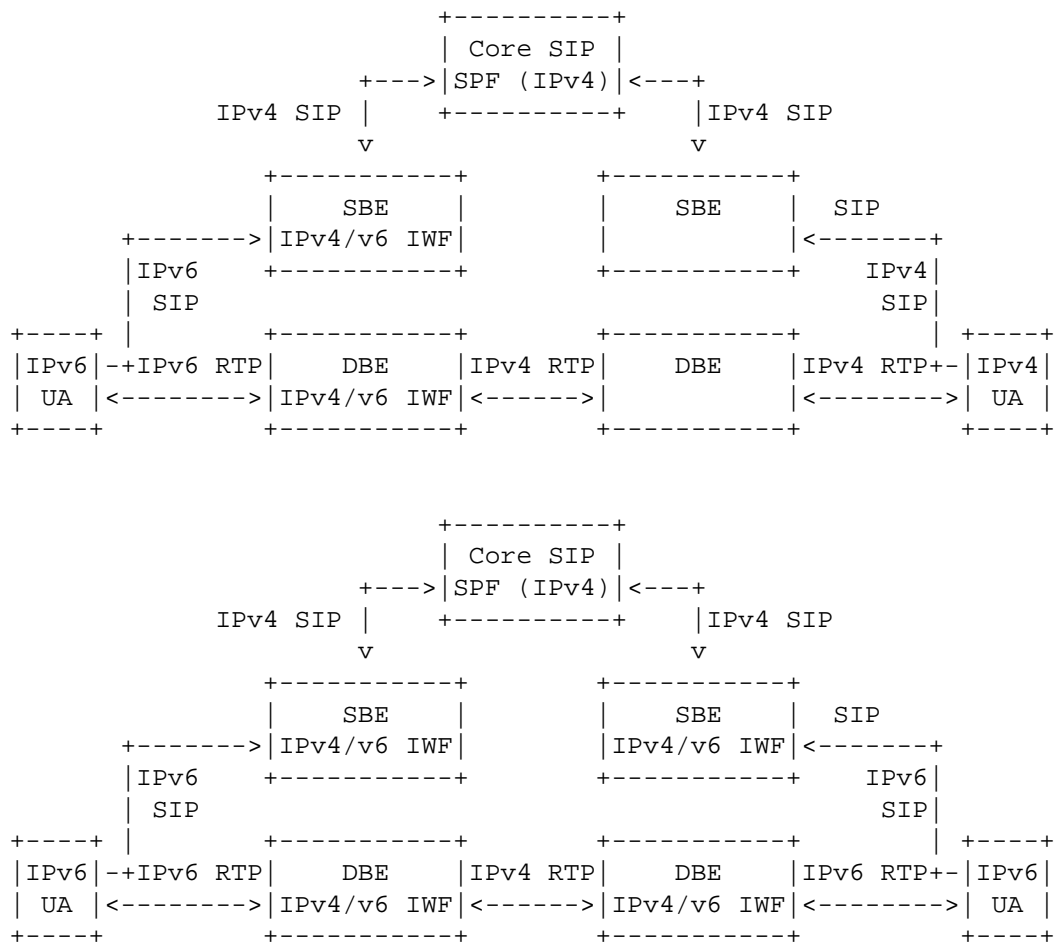


Figure 3: Basic Scenario

It may be valuable for service providers to consider solutions that avoid redundant IPv4/IPv6 NATs and that avoid involving several DBEs.

A.3.3. Avoiding IPv4/IPv6 Interworking

A solution to indicate both IPv4 and IPv4 addresses is required for service providers that want the following:

1. A means to promote the invocation of IPv6 transfer capabilities that can be enabled, while no parsing errors are experienced by core service legacy nodes.
2. To optimize the cost related to IPv4-IPv6 translation licenses.
3. To reduce the dual-stack lifetime.
4. To maintain an IPv4-only core.
5. To have a set of SBEs/DBEs that are IPv6-enabled.

This section provides an overview of the procedure to avoid IPv4/IPv6 interworking.

When an SBE receives an INVITE, it instantiates in its DBE an IPv6-IPv6 context and an IPv6-IPv4 context. Both an IPv6 address and an IPv4 address are returned, together with other information such as port numbers. The SBE builds an SDP offer, including both the IPv4 and IPv6-related information using the "altc" attribute. IPv6 is indicated as the preferred connectivity type; see Figure 4.

```
o=- 25678 753849 IN IP4 192.0.2.2
c=IN IP4 192.0.2.2
m=audio 12340 RTP/AVP 0 8
a=altc:1 IP6 2001:db8::2 6000
a=altc:2 IP4 192.0.2.2 12340
```

Figure 4: SDP Offer Updated by the SBE

The request is then forwarded to the core SPF, which, in turn, forwards it to the terminating SBE.

- o If this SBE is a legacy one, then it will ignore "altc" attributes and use the "c=" line.
- o If the terminating SBE is IPv6-enabled:
 - * If the called UA is IPv4 only, then an IPv6-IPv4 context is created in the corresponding DBE.
 - * If the called UA is IPv6-enabled, then an IPv6-IPv6 context is created in the corresponding DBE.

Figure 5 shows the results of the procedure when placing a session between an IPv4 and IPv6 UAs, while Figure 6 shows the results of establishing a session between two IPv6-enabled UAs. The result is still not optimal since redundant NAT66 is required (Appendix A.3.4).

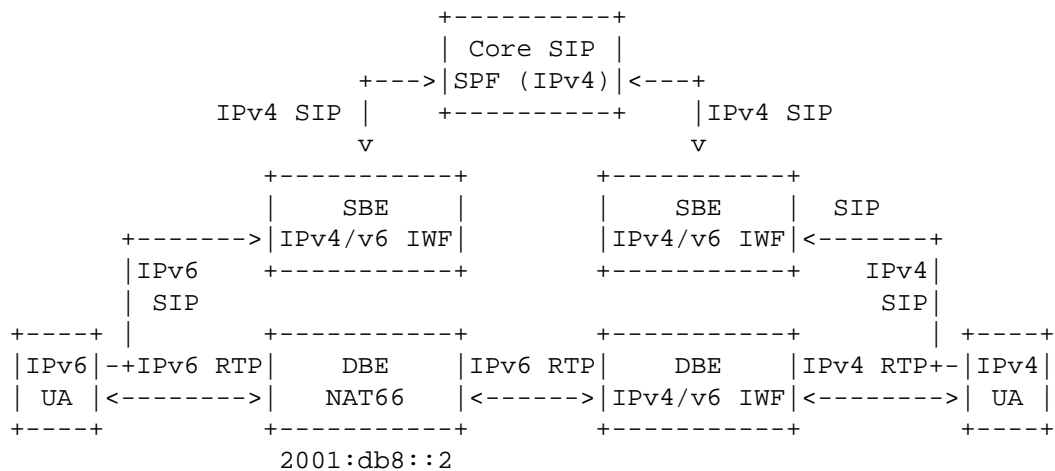


Figure 5: Session Establishment between IPv4 and IPv6 UAs

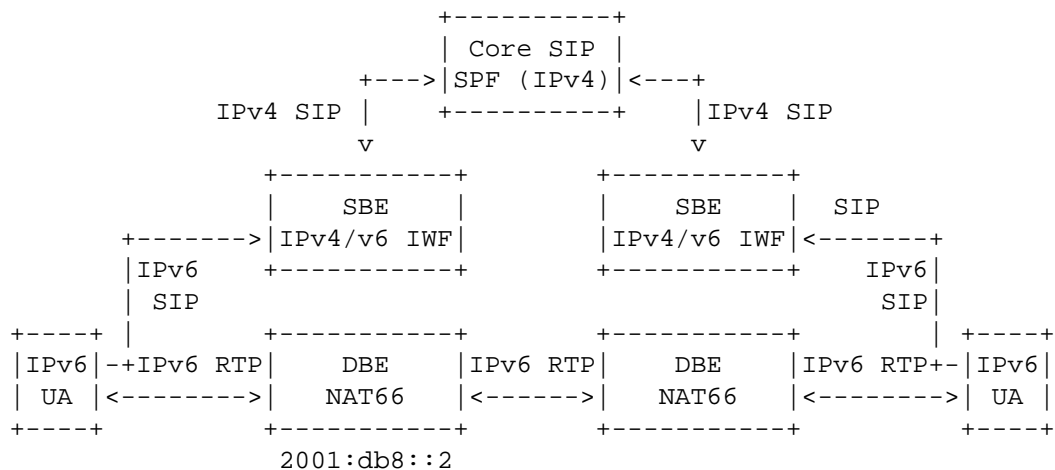


Figure 6: Session Establishment between IPv6 UAs

A.3.4. DBE Bypass Procedure

For service providers wanting to involve only one DBE in the media path when not all SBEs/DBEs and UAs are IPv6-enabled, a means to indicate both IPv4 and IPv6 addresses without inducing session failures is required. This section proposes an example procedure using the "altc" attribute.

When the originating SBE receives an INVITE from an IPv6-enabled UA, it instantiates in its DBE an IPv6-IPv6 context and an IPv6-IPv4 context. Both an IPv6 address and an IPv4 address are returned, together with other information, such as port numbers. The SBE

builds an SDP offer, including both IPv4 and IPv6-related information using the "altc" attribute (Figure 7). IPv6 is indicated as preferred connectivity type.

```
o=- 25678 753849 IN IP4 192.0.2.2
c=IN IP4 192.0.2.2
m=audio 12340 RTP/AVP 0 8
a=altc:1 IP6 2001:db8::2 6000
a=altc:2 IP4 192.0.2.2 12340
```

Figure 7: SDP Offer Updated by the SBE

The request is then forwarded to the core SPF, which, in turn, forwards it to the terminating SBE:

- o If the destination UA is IPv6 or reachable with a public IPv4 address, the SBEs only forwards the request without altering the SDP offer. No parsing error is experienced by core service nodes since ALTC is backward compatible.
- o If the terminating SBE does not support ALTC, it will ignore this attribute and use the legacy procedure.

As a consequence, only one DBE is maintained in the path when one of the involved parties is IPv6-enabled. Figure 8 shows the overall procedure when the involved UAs are IPv6-enabled.

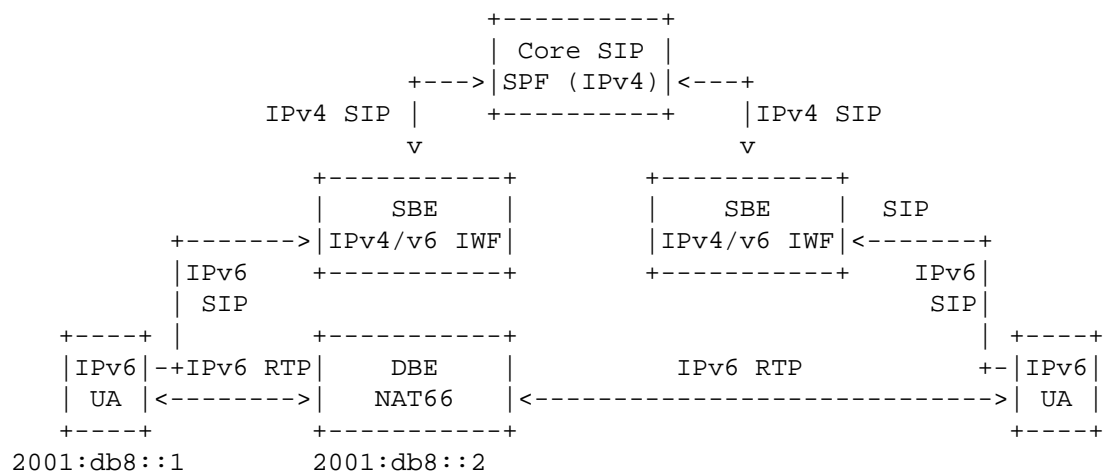


Figure 8: DBE Bypass Overview

The main advantages of such a solution are as follows:

- o DBE resources are optimized.
- o No redundant NAT is maintained in the path when IPv6-enabled UAs are involved.
- o End-to-end delay is optimized.
- o The robustness of the service is optimized since the delivery of the service relies on fewer nodes.
- o The signaling path is also optimized since no communication between the SBE and DBE at the terminating side is required for some sessions. (That communication would be through the Service Policy Decision Function (SPDF) in a Telecoms and Internet converged Services and Protocols for Advanced Networks/IP Multimedia Subsystem (TISPAN/IMS) context.)

A.3.5. Direct Communications between IPv6-Enabled User Agents

For service providers wanting to allow direct IPv6 communications between IPv6-enabled UAs, when not all SBEs/DBEs and UAs are IPv6-enabled, a means to indicate both the IPv4 and IPv6 addresses without inducing session failures is required. Below is an example of a proposed procedure using the "altc" attribute.

At the SBE originating side, when the SBE receives an INVITE from the calling IPv6 UA (Figure 9), it uses ALTC to indicate two IP addresses:

1. An IPv4 address belonging to its controlled DBE.
2. The same IPv6 address and port as received in the initial offer made by the calling IPv6.

Figure 9 shows an excerpted example of the SDP offer of the calling UA, and Figure 10 shows an excerpted example of the updated SDP offer generated by the originating SBE.

```
o=- 25678 753849 IN IP6 2001:db8::1
c=IN IP6 2001:db8::1
m=audio 6000 RTP/AVP 0 8
```

Figure 9: SDP Offer of the Calling UA

```
o=- 25678 753849 IN IP4 192.0.2.2
c=IN IP4 192.0.2.2
m=audio 12340 RTP/AVP 0 8
a=altc:1 IP6 2001:db8::1 6000
a=altc:2 IP4 192.0.2.2 12340
```

Figure 10: SDP Offer Updated by the SBE

The INVITE message will be routed appropriately to the destination SBE:

1. If the SBE is a legacy device (i.e., IPv4-only), it will ignore IPv6 addresses and will contact its DBE to instantiate an IPv4-IPv4 context.
2. If the SBE is IPv6-enabled, it will only forward the INVITE to the address of contact of the called party:
 - a. If the called party is IPv6-enabled, the communication will be placed using IPv6. As such, no DBE is involved in the data path, as illustrated in Figure 11.
 - b. Otherwise, IPv4 will be used between the originating DBE and the called UA.

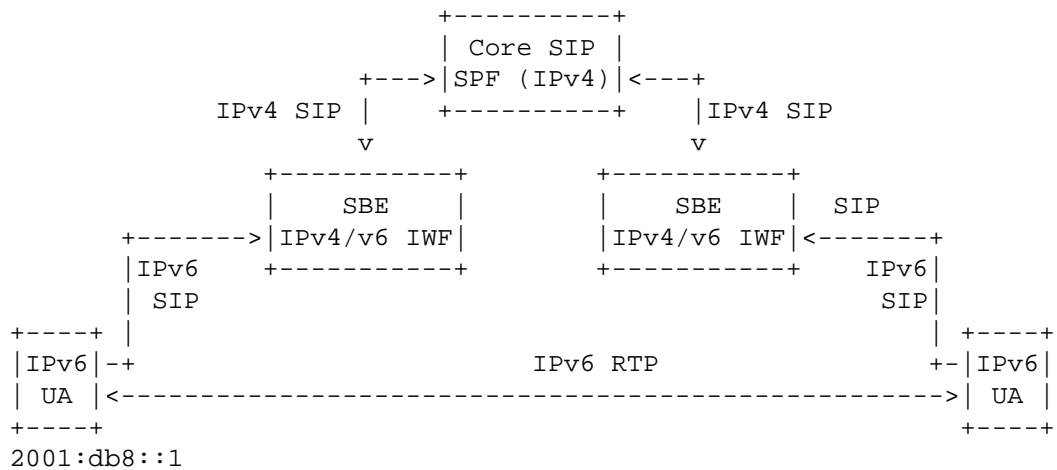


Figure 11: Direct IPv6 Communication

Authors' Addresses

Mohamed Boucadair
France Telecom
Rennes 35000
France

EMail: mohamed.boucadair@orange.com

Hadriel Kaplan
Acme Packet
71 Third Ave.
Burlington, MA 01803
USA

EMail: hkaplan@acmepacket.com

Robert R Gilman
Independent

EMail: bob_gilman@comcast.net

Simo Veikkolainen
Nokia

EMail: Simo.Veikkolainen@nokia.com