

Virtual IPv6 Connectivity for IPv4-Only Networks
draft-vogt-durand-virtual-ip6-connectivity-03

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of [BCP 78](#) and [BCP 79](#). This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

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 April 29, 2010.

Copyright Notice

Copyright (c) 2009 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 in effect on the date of publication of this document (<http://trustee.ietf.org/license-info>). Please review these documents carefully, as they describe your rights and restrictions with respect to this document.

Abstract

Although the impetus to invest in interworking between IP versions 4 and 6 is initially on the side of early IPv6 adopters, more substantial IPv6 deployment in the future will shift this impetus towards the side of the legacy IPv4 Internet. However, interworking techniques for IPv4-only networks are as yet largely unexplored. This document proposes Virtual IPv6 Connectivity, a technique for IPv4-only networks to communicate with the IPv6 Internet.

Table of Contents

1.	Introduction	3
2.	Conceptual Overview	4
3.	Virtual IP Addresses	6
4.	Discovery of Virtual IP Addresses	7
5.	IP Protocol Translation	13
6.	Multi-Homing Support	18
7.	Side-Effects	20
8.	Security Considerations	21
9.	IANA Considerations	21
10.	Acknowledgment	21
11.	References	22
Appendix A.	Pending Document Changes	22
Appendix B.	Log of Document Changes	22
	Authors' Addresses	24

1. Introduction

The deployment of IP version 6 is challenging. Since it requires changes to applications, host stacks, and networks end-to-end, IPv6 deployment involves multiple stakeholders. These stakeholders are in general independent and not coordinated, so a simultaneously transition to IPv6 by all stakeholders is practically impossible. It is therefore necessary to enable the stakeholders to deploy IPv6 independently of each other. This requires interworking between those stakeholders who have already adopted IPv6, and those stakeholders who have not yet.

There are two basic approaches to enable interworking between IP versions:

- o Make IPv6 backwards compatible with IPv4
- o Make IPv4 forward compatible with IPv6

Which of the approaches is feasible depends on where the impetus on interworking is -- on the side of early IPv6 adopters, or on the side of the legacy IPv4 Internet. This impetus will shift over time. Initially, the impetus for interworking will naturally be high on the side of early IPv6 adopters. Services and peers will at that point almost exclusively be IPv4-only, so IPv6 adopters will have to make sure they can communicate via IPv4. For the same reason, initial incentives to invest in interworking will be very low on the side of the legacy IPv4 Internet: Why would the legacy IPv4 Internet want to communicate via IPv6 if everything they need is available also via IPv4?

However, as IPv6 deployment progresses, more and more services and peers will become IPv6-capable. The interworking impetus on the side of IPv6 adopters will therefore shrink. At some point, a critical mass of IPv6 deployment will have been reached to make it feasible for services and peers to be IPv6-only. And as there are more and more IPv6-only services and peers, the impetus for interworking will grow on the side of the legacy IPv4 Internet, so that those IPv6-only services and peers can be reached.

Consequently, interworking between IP versions will initially be realized foremost through backwards compatibility of IPv6 with IPv4, but over time increasingly also through forward compatibility of IPv4 with IPv6. Due to the immediate need for IPv6 backwards compatibility, the Internet Engineering Task Force has made the design of suitable techniques a task of high priority. However, techniques for IPv4 forward compatibility are as yet largely unexplored.

This document proposes Virtual IPv6 Connectivity, a technique to make IPv4 forward compatible with IPv6. Virtual IPv6 Connectivity uses IP protocol translators on the border of IPv4-only networks to enable local hosts to communicate with IPv6-only peers on the Internet. It makes IPv4-only hosts reachable by peers on the Internet via "virtual IPv6 addresses", and it makes IPv6-only peers reachable by IPv4-only hosts via "virtual IPv4 addresses". A virtual IPv6 address is statically mapped to each IPv4 address, and made retrievable via the DNS or DNSSEC. A virtual IPv4 address is mapped to a peer's IPv6 address dynamically and on demand, and made retrievable via DNS proxies or DNSSEC proxies. The IP protocol translators can alternatively be operated by IPv4-only networks directly, or by providers on behalf of IPv4-only networks.

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](#)].

2. Conceptual Overview

To enable hosts in an IPv4-only network to communicate with remote IPv6 peers, Virtual IPv6 Connectivity employs four components:

- o Virtual IPv4 and IPv6 addresses to represent, respectively, the IPv6 addresses of remote peers to local IPv4-only hosts, and the IPv4 addresses of local hosts to remote IPv6 peers.
- o IP protocol translators, placed on border links of the IPv4-only network, to translate packets exchanged with remote IPv6 peers.
- o Support in authoritative DNS servers for returning virtual IPv6 addresses in place of the IPv4 addresses of local IPv4-only hosts when responding to queries from remote IPv6-only peers.
- o Support in recursive DNS servers for returning virtual IPv4 addresses in place of the IPv6 addresses of remote IPv6-only peers when responding to queries from local IPv4-only hosts.

Figure 1 depicts an IPv4-only network that uses Virtual IPv6 Connectivity. The network has an IP protocol translator, denoted "xlate" in the figure, on its border link to the IPv6 Internet. The network also has sets of virtual IPv6 and IPv4 addresses, which, in this example, are taken from the IPv6 address prefix 2001:DB8::/96 and from the IPv4 address prefix 10.0.0.0/8, respectively. The virtual IPv6 addresses represent the regular IPv4 addresses of the IPv4-only network to remote IPv6-only peers. The virtual IPv4 addresses represent the regular IPv6 addresses of remote IPv6-only

peers to hosts in the IPv4-only network. Furthermore, the IPv4-only network has authoritative and recursive DNS servers that can return, respectively, virtual IPv4 addresses in place of local IPv4 addresses, and virtual IPv4 addresses in place of remote IPv6 addresses.

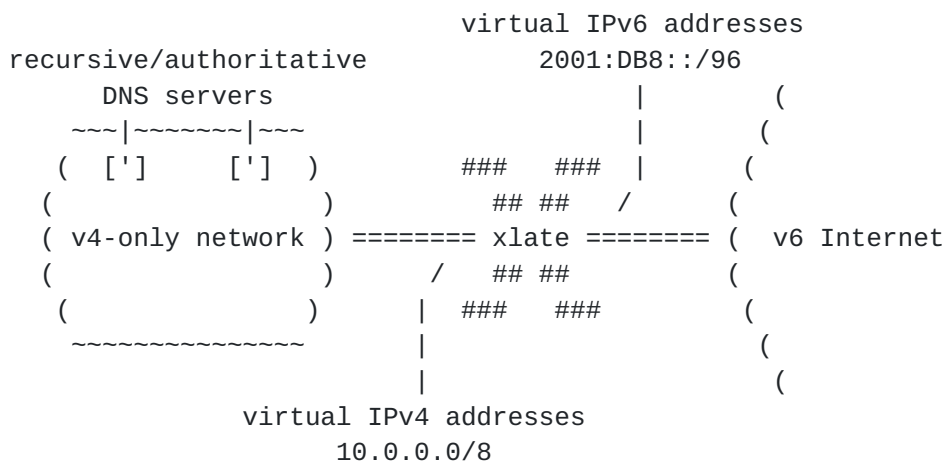


Figure 1: Conceptual overview of Virtual IPv6 Connectivity

Packets exchanged between an IPv4 host and an IPv6 peer are always destined to a virtual IP address, and sourced from a regular IP address, when initially dispatched by the sender. Virtual IP addresses route to an IP protocol translator, which performs three tasks for each received packet exchanged between an IPv4 host and an IPv6 peer:

- o It maps the IP source and destination addresses in a packet, replacing the virtual IP destination address with the represented regular IP address, and replacing the regular IP source address with the representing virtual IP address.
- o It uses the Stateless IP/ICMP Translation algorithm [[RFC2765](#)] to translate the IP header of the packet into an IP header from the respective other IP version. The new IP header uses the IP source and destination addresses determined in the foregoing mapping.
- o It forwards the packet towards the new IP destination address.

Translated packets are consequently always destined to a regular IP address and sourced from a virtual IP address.

The pair of a virtual IP address and the represented regular IP address is called a "mapping". Mappings are used in IP protocol

translators and DNS servers as a basis for translating packets between IP versions, and for provisioning virtual IP addresses in the DNS. Mappings between IPv4 addresses and virtual IPv6 addresses are static. Each IPv4 address is mapped to a separate virtual IPv6 address by appending it to a 96-bit virtual IPv6 address prefix. This enables stateless, algorithmic mapping in IP protocol translators and authoritative DNS servers. On the other hand, mappings between IPv6 addresses and virtual IPv4 addresses are dynamic and on-demand to enable reuse of virtual IPv4 addresses. This is inevitable because any set of virtual IPv4 addresses is smaller than the global set of regular IPv6 addresses for which mappings are potentially needed. As a consequence, mappings between IPv6 addresses and virtual IPv4 addresses must be statefully maintained by IP protocol translators and recursive DNS servers per communication session. Mapping creation is triggered by a recursive DNS server when receiving a query for an IPv6-only peer's DNS name from an IPv4-only host. The mappings expire when no longer used by a communication session.

3. Virtual IP Addresses

For the local hosts in an IPv4-only network to communicate with remote IPv6 correspondent hosts, local hosts and remote correspondent hosts must be able to reach their respective peer with an IP version they support: The local hosts must be able to reach the remote correspondent hosts via an IPv4 address, and the remote correspondent hosts must be able to reach the local hosts via an IPv6 address. Virtual IPv6 Connectivity achieves this reachability across IP versions by means of virtual IPv4 and IPv6 addresses, which respectively map onto the regular IPv6 addresses of remote correspondent hosts and the regular IPv4 addresses of local hosts.

Virtual IPv6 addresses represent local hosts in an IPv4-only network to remote IPv6 correspondent hosts. They are ordinary, global IPv6 addresses, which the IPv4-only network obtains either from its Internet service provider or from an Internet registry. Virtual IPv6 addresses are announced externally in the global routing system, but are not used internally within the IPv4-only network. They map one-to-one onto the regular IPv4 addresses in the IPv4-only network, i.e., there is one unique virtual IPv6 address for each regular IPv4 address. There are no restrictions as to how to realize this one-to-one mapping. A simple, recommended way is to reserve a 96-bit IPv6 address prefix, and to map each regular IPv4 address to the virtual IPv6 address that is defined by attaching the 32 bits of the regular IPv4 address to the 96 bits of the reserved 96-bit IPv6 address prefix.

Virtual IPv4 addresses represent remote IPv6 correspondent hosts to local hosts in an IPv4-only network. They are ordinary IPv4 addresses that map onto the regular IPv6 addresses of the remote correspondent hosts. Virtual IPv4 addresses are announced internally in the local routing system of an IPv4-only network, but are not used outside the IPv4-only network. Since virtual IPv4 addresses are used only locally within an IPv4-only network, they can be non-global. This makes them re-usable in different IPv4-only networks with Virtual IPv6 Connectivity. As an example, virtual IPv4 addresses could be taken from the private net-10 or 192.168.0.0/16 address spaces [###[RFC1918](#)], or from a new IPv4 address block specifically allocated for use in IPv4-only networks with Virtual IPv6 Connectivity.

Different than mappings between regular IPv4 addresses and virtual IPv6 addresses, mappings between regular IPv6 addresses and virtual IPv4 addresses cannot be one-to-one because the set of regular IPv6 addresses that potentially need to be mapped is larger than any pool of virtual IPv4 addresses. Mappings between regular IPv6 addresses and virtual IPv4 addresses may instead change dynamically depending on the set of remote IPv6 correspondent hosts that communicate with an IPv4-only network. Mappings between regular IPv6 addresses and virtual IPv4 addresses are therefore created in an on-demand manner, and they have a lifetime after which they may be replaced.

In the example of Figure 1, the IPv4-only network has obtained the 96-bit IPv6 address prefix 2001:DB8::/96 for virtual IPv6 addresses, so any IPv4 address a.b.c.d from the IPv4-only network can be mapped one-to-one to a virtual IPv6 address 2001:DB8::a.b.c.d. The IPv4-only network furthermore uses the private net-10 address space, 10.0.0.0/8, for virtual IPv4 addresses. So for any remote IPv6 correspondent host that communicates with a local host in the IPv4-only network, a virtual IPv4 address is allocated from the 8-bit IPv4 address prefix 10.0.0.0/8, and is temporarily mapped to the remote correspondent host's regular IPv6 address.

4. Discovery of Virtual IP Addresses

Communications between between an IPv4-only host and an IPv6-only peer require that the initiator of a new communication obtains a virtual IP address for its peer: An IPv6-only peer intending to reach an IPv4-only host must obtain a virtual IPv6 address for the IPv4-only host; an IPv4-only host intending to reach an IPv6-only peer must obtain a virtual IPv4 address for the IPv6-only peer. To support the discovery of virtual IP addresses, Virtual IPv6 Connectivity makes virtual IPv4 and IPv6 addresses available via the DNS. This section describes the extensions for recursive and

authoritative DNS servers in an IPv4-only network that are necessary for this.

4.1. Discovery of Virtual IPv6 Addresses

For an IPv6-only peer to obtain via the DNS the virtual IPv6 addresses of a host in an IPv4-only network, the authoritative DNS servers in the IPv4-only network must provide records of type AAAA containing the host's virtual IPv6 addresses. That is, whenever the DNS resolves a host's DNS name into an IPv4 address of that host, the same DNS name must also resolve to the corresponding virtual IPv6 address. The retrieval of virtual IPv6 addresses via DNS then does not differ from the retrieval of regular IPv4 addresses.

Furthermore, the virtual IPv6 addresses maintained by an authoritative DNS server for an IPv4-only host should automatically adapt when the host's regular IPv4 addresses change in order to support dynamic DNS updates: Since the mapping between IPv4 addresses and the corresponding virtual IPv6 addresses is static, a change in a host's IPv4 address implies that also a virtual IPv6 address of the host changes. The change of the corresponding type-AAAA record should be pursued by the authoritative DNS server. Hosts cannot be expected to update records of type AAAA because they are generally unaware of their virtual IP addresses.

To enable accurate provisioning of virtual IPv6 addresses via the DNS, Virtual IPv6 Connectivity calls for authoritative DNS servers to synthesize type-AAAA records on demand whenever an IPv6 address is being requested by a remote peer. This guarantees that the type-AAAA records in a DNS reply contain exactly those virtual IPv6 addresses that correspond to the IPv4 addresses of the host whose DNS name is being resolved.

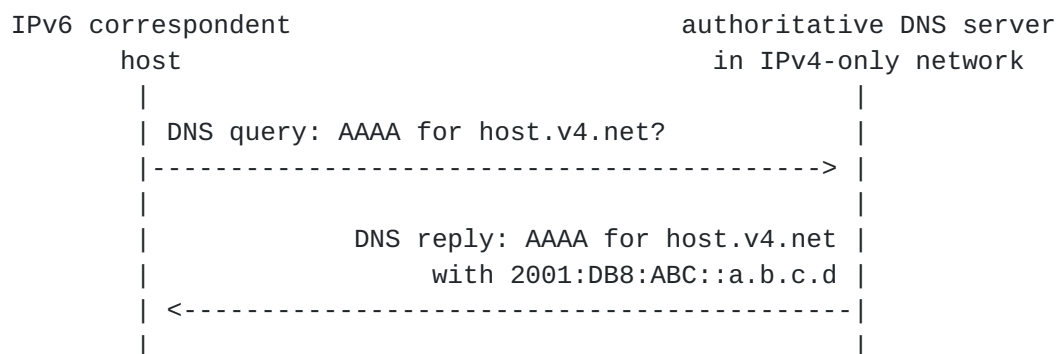


Figure 2: Discovery of virtual IPv6 addresses via the DNS

Figure 4 shows the resolution of an IPv4-only host's DNS name into a virtual IPv6 address. In this example, the DNS name is "host.v4.net", and it resolves into the regular IPv4 address a.b.c.d. The virtual IPv6 address prefix is 2001:DB8:ABC::/96 in this example, so the host's IPv4 address maps to the virtual IPv6 address 2001:DB8:ABC::a.b.c.d. The message exchange in the figure begins where a remote peer queries an authoritative DNS server in the IPv4-only network for a record of type AAAA associated with DNS name "host.v4.net". The authoritative DNS server replies with a type-AAAA record containing the virtual IPv6 address 2001:DB8:ABC::a.b.c.d.

4.2. Discovery of Virtual IPv4 Addresses

For an IPv4-only host to obtain via the DNS the virtual IPv4 addresses of a remote IPv6-only peer, the DNS must provide records of type A containing the peer's virtual IPv4 addresses. However, different than virtual IPv6 addresses, virtual IPv4 addresses generally cannot be provisioned directly by the DNS server that is authoritative for the corresponding regular IP addresses. A DNS server handing out virtual IPv4 addresses must have an administrative relationship with the IPv4-only network to which the virtual IPv4 addresses belong. This relationship generally does not exist for DNS servers authoritative for a remote peer's regular IPv6 addresses.

To enable the retrieval of virtual IPv4 addresses via the DNS despite the missing administrative relationship between the authoritative DNS server and the IPv4-only network, Virtual IPv6 Connectivity calls for recursive DNS servers in IPv4-only networks to serve as proxies for the authoritative DNS servers of IPv6-only peers. Thus, when requested by an IPv4-only host to resolve the DNS name of an IPv6-only peer, a recursive DNS server retrieves the peer's regular IPv6 addresses, triggers the creation of a mapping for those regular IPv6 addresses, and returns the corresponding virtual IPv4 addresses on behalf of the peer's authoritative DNS server.

Since mappings between virtual IPv4 addresses and regular IPv6 addresses are dynamic, they must be carefully coordinated between IP protocol translators and recursive DNS servers if created during DNS name resolution. Lack of coordination can lead to loss of packets exchanged between IPv4-only hosts and IPv6-only peers due to incorrect translation. To guarantee the coordination of mappings, Virtual IPv6 Connectivity requires all mappings between virtual IPv4 addresses and regular IPv6 addresses to be created by IP protocol translators. Accordingly, recursive DNS servers must request a mapping from an IP protocol translator when needed for a peer's regular IPv6 address. Given such a request, an IP protocol translator first checks if a mapping already exists for the peer's regular IPv6 address. If so, the IP protocol translator returns this

mapping to the recursive DNS server. If no mapping exists for the peer's regular IPv6 address, the IP protocol translator creates a new mapping, stores the mapping locally for subsequent translation of packets, and returns a copy of the mapping to the requesting recursive DNS server. The recursive DNS server finally generates a DNS reply and removes the mapping afterwards. IP protocol translators maintain pools of virtual IPv4 addresses that are currently not used in a mapping. They use virtual IPv4 addresses from these pools to create new mappings.

The reason for requiring IP protocol translators, rather than recursive DNS servers, to be in control of mappings is to spare the overhead of coordinating mappings that are created based on packets received by an IP protocol translator. These mappings are created for the first packet in communication sessions initiated by a remote IPv6-only peer. Having the IP protocol translator create the mapping in those cases removes the need to coordinate the mapping with a recursive DNS server. Coordination between the IP protocol translator and the recursive DNS server is then only necessary for mappings created during DNS name resolution initiated by IPv4-only hosts.

The overall operation of a recursive DNS server in an IPv4-only network more specifically proceeds as follows:

- o Upon receiving a DNS query from a local IPv4-only host for records of type A associated with a peer's DNS name, the recursive DNS server first attempts to retrieve the records of type A directly from the peer's authoritative DNS server. This is the standard operation of a recursive DNS server.
- o If the recursive DNS server receives a DNS reply with one or more records of type A, it returns the DNS reply unchanged to the resolving IPv4-only host. The peer in this case is reachable via IPv4, and it is not necessary to allocate a virtual IPv4 address for the peer.
- o If the recursive DNS server fails to receive a DNS reply with a record of type A, the peer cannot be reached at a regular IPv4 address. The recursive DNS server in this case attempts to retrieve records of type AAAA associated with the peers's DNS name.
- o If the recursive DNS server receives a DNS reply with one or more records of type AAAA, the peer can be reached via IPv6. The recursive DNS server in this case requests a mapping for each of the regular IPv6 addresses in those records from the IP protocol translator. It then synthesizes a DNS reply with a record of type

A for each virtual IPv4 address in a received mapping, and it returns this DNS reply to the resolving IPv4-only host. The recursive DNS server does not store the mappings received from the IP protocol translator.

- o If the recursive DNS server fails to receive a DNS reply with records of either type A or type AAAA, it returns an error notification to the resolving IPv4-only host. The peer is in this case unreachable from the IPv4-only host.

As an optimization to reduce latency, a recursive DNS server may send DNS queries for type-A and type-AAAA records simultaneously to the DNS server authoritative for the peer. With this optimization, if no type-A records can be retrieved, the latency related to retrieving type-AAAA records can be reduced or eliminated.

IP protocol translators remove mappings between virtual IPv4 addresses and regular IPv6 addresses that appear to be no longer in use. To keep track of which mappings are in use, IP protocol translators should maintain an expiration timer for each mapping. The expiration timer for a mapping is set at the time the mapping is created. It is reset whenever a mapping for the same regular IPv6 address is requested, and whenever the mapping is used to translate a packet. After a mapping's expiration timer has reached zero, the mapping should be removed, and the virtual IPv4 address from the mapping should be returned to the IP protocol translator's pool of unallocated virtual IPv4 addresses.

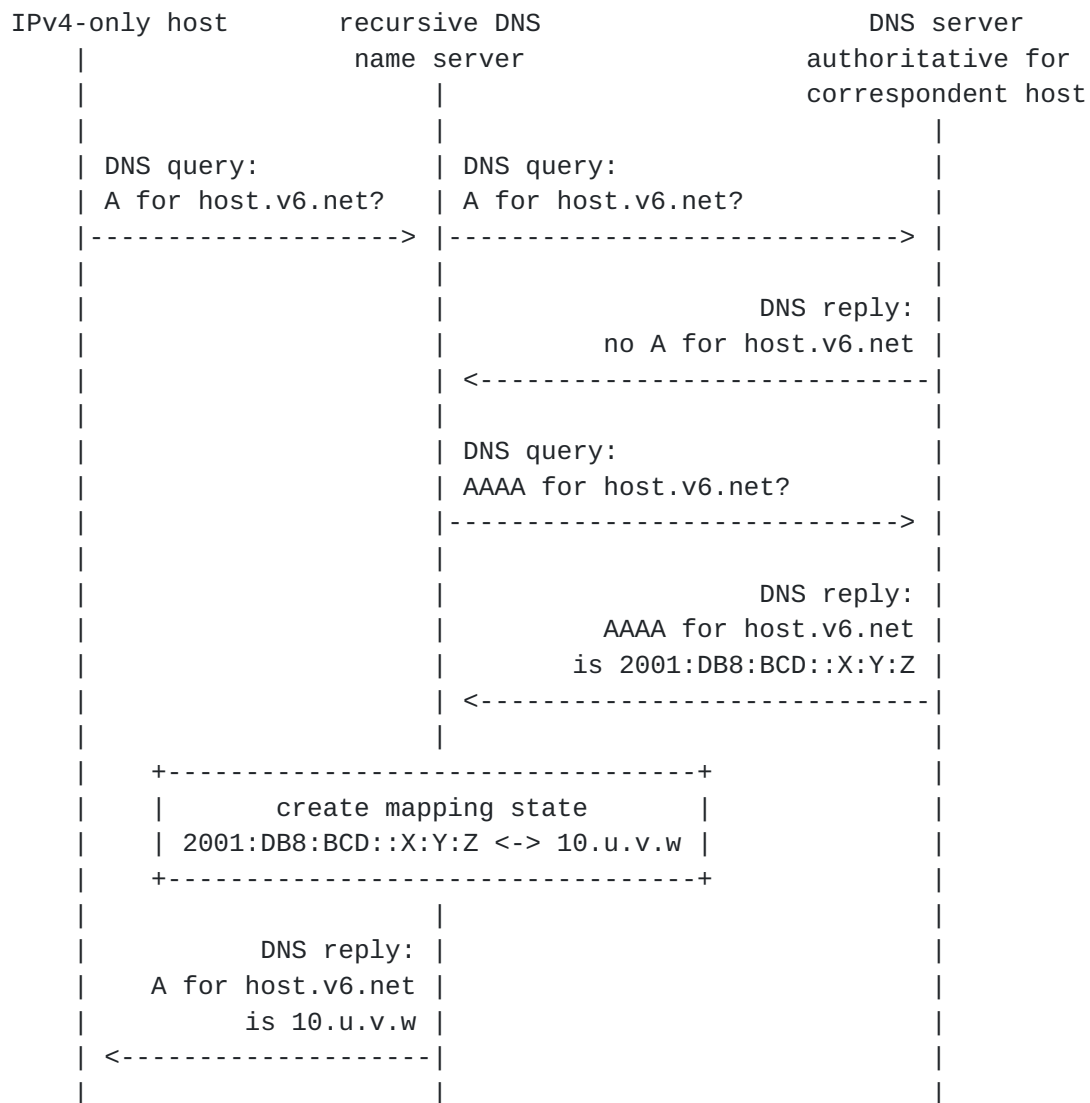


Figure 3: Discovery of virtual IPv4 addresses via the DNS

Figure 3 illustrates the operation of a recursive DNS server in an IPv4-only network with Virtual IPv6 Connectivity. The IPv4-only host on the left of the figure queries a recursive DNS server in the middle for records of type A associated with the DNS name of an IPv6-only peer. The DNS name of the peer is "host.v6.net" in this example, and this is associated with a single record of type AAAA for IPv6 address 2001:DB8:BCD::X:Y:Z. Since only IPv4 addresses are of use for the IPv4-only host, the host naturally does not send a DNS query for records of type AAAA. The recursive DNS server first processes the DNS query unchanged, so it tries to retrieve records of type A from the DNS server that is authoritative for the peer, as represented on the right of Figure 3. This fails because the peer does not have an IPv4 address. In a second attempt, the recursive

DNS name server queries for records of type AAAA. This is successful; the recursive DNS server receives a record with IPv6 address 2001:DB8:BCD::X:Y:Z. The recursive DNS server then obtains from an IP protocol translator in the IPv4-only network a mapping between the regular IPv6 address 2001:DB8:BCD::X:Y:Z and a virtual IPv4 address that routes to this IP protocol translator. In the example of Figure 3, this virtual IPv4 address is 10.u.v.w. The recursive DNS server finally returns a DNS reply to the resolving IPv4-only host that contains a record of type A containing the virtual IPv4 address 10.u.v.w.

5. IP Protocol Translation

In addition to the mapping between regular IP addresses from one IP version and virtual IP addresses from the respective other IP version, packets exchanged between the local hosts in an IPv4-only network and remote IPv6 correspondent hosts must be converted from IPv4 to IPv6 and vice versa on the border between the IPv4-only network and the IPv6 Internet. For this purpose, IPv4-only networks with Virtual IPv6 Connectivity deploy IP protocol translators on each border link that shall be enabled for communications with remote IPv6 correspondent host. An IP protocol translator is a router with the extra capabilities to map between regular and virtual IP addresses, and to convert packets between IPv4 and IPv6 using the appropriate mapping. Figure 1 illustrates an IPv4-only network with one border link, which is endowed with an IP protocol translator to achieve Virtual IPv6 connectivity for the IPv4-only network.

Since in general not all packets traversing a border link need to be translated between IPv4 and IPv6, IP protocol translators must have a means to distinguish packets that require translation from packets that can be forwarded without modification. IP protocol translators perform this differentiation based on the packets' IP destination address: Packets destined to a regular IP address do not require translation, whereas packets destined to a virtual IP address do. This rule is independent of whether packets originate within and leave the IPv4-only network, or whether the packets originate outside and enter the IPv4-only network. For packets that are destined to a regular IP address, an IP protocol translator behaves as an ordinary router. For packets that are destined to a virtual IP address, the IP protocol translator performs three tasks:

- o It maps the IP source and destination addresses in a packet, replacing the virtual IP destination address with the represented regular IP address, and replacing the regular IP source address with the representing virtual IP address.

- o It uses the Stateless IP/ICMP Translation Algorithm [SIIT] to translate the IP header of the packet into an IP header from the respective other IP version. The new IP header uses the IP source and destination addresses determined in the foregoing mapping.
- o It forwards the packet towards the new (regular) IP destination address.

DESCRIBE WHEN TRANSLATOR CREATES MAPPING?

To ensure that packets pass through an IP protocol translator if exchanged between a host in an IPv4-only network and a remote IPv6 correspondent host, the routes towards virtual IPv4 and IPv6 addresses must lead to an IP protocol translator. Routes to virtual IPv4 addresses must be available within the IPv4-only network; they lead to the interface of an IP protocol translator that faces the IPv4-only network. Routes to virtual IPv6 addresses must be available in the Internet; they lead to the Internet-facing interface of an IP protocol translator. Thus, when a remote IPv6 correspondent host sends a packet to the virtual IPv6 address of a local host in the IPv4-only network, the packet routes to an IP protocol translator, which translates the packet's virtual IPv6 destination address into the mapped regular IPv4 destination address, and the packet's regular IPv6 source address into a mapped virtual IPv4 source address. Vice versa, when the local host in the IPv4-only network sends a packet to a remote IPv6 correspondent host, an IP protocol translator translates the packet's regular IPv4 source address into the mapped virtual IPv6 source address, and the packet's virtual IPv4 destination address into the mapped regular IPv6 destination address. Both local host and remote correspondent host therefore use only their respective IP version; the peer's regular IP address from the other IP version is transparent to them.

There are different means to provision routes to virtual IP addresses. One is through dynamic route announcements by IP protocol translators. IP protocol translators then announce routes towards virtual IPv6 addresses on their interfaces towards the IPv6 Internet, and routes towards virtual IPv4 addresses on their interfaces towards the IPv4-only network. Static configuration is an alternative for part of the routes. Routes to virtual IPv4 addresses can be statically configured in routers of the IPv4-only network. Routes to virtual IPv6 addresses can be statically configured between the IPv4-only network and its provider, if the IP protocol translators are located on the side of the IP-only network. In the latter case, it is the IPv4-only network's provider that dynamically announces a route to virtual IPv6 addresses towards the Internet.

Routes to virtual IP addresses may be aggregated with routes to other

Since the communication between a host in an IPv4-only network and a remote IPv6 correspondent host would break if the mapping between either host's regular and virtual IP addresses changed throughout the course of the communication, IP protocol translators must ensure that mappings are stable for the duration of the communications in which they are used. This is trivial for the mappings between regular IPv4 addresses and virtual IPv6 addresses because these mappings are one-to-one and therefore constant. However, as one-to-one mappings are not possible between the large set of regular IPv6 addresses and the necessarily smaller set of virtual IPv4 addresses, IP protocol translators need state to ensure that these mappings remain constant during the communications in which they are used. An IP protocol translator establishes such state in either of two ways, depending on which way a communication is established:

- o For communications that are initiated by a local host in an IPv4-only network, mapping state is created at the time the local host retrieves the remote IPv6 correspondent host's virtual IPv4 address via the DNS. This ensures that the IP protocol translator can map the virtual IPv4 address in the first packet that the local host sends to the remote correspondent host. Mapping state creation is in this case initiated by a recursive DNS name server from the IPv4-only network. This will be explained further in [Section 4](#).
- o For communications that are initiated by a remote IPv6 correspondent host, the creation of mapping state is postponed to the time the IP protocol translator receives the first packet that the remote correspondent host sends to the local host in the IPv4-only network. It is then the IP protocol translator which initiates mapping state creation. Postponing the creation of mapping state is possible in this case because the remote correspondent host does not use the virtual IPv4 address for which mapping state is created. The postponement then has the advantage that both the creation of mapping state as well as the allocation of a virtual IPv4 address happens only when there is actually a communication using it.

The involvement of both IP protocol translators and recursive DNS name servers in the creation of mapping state and in the allocation of virtual IPv4 addresses calls for coordination between IP protocol translators and recursive DNS name servers: IP protocol translators must have relevant mapping state even if the creation of the state was initiated by a recursive DNS name server; and simultaneous allocation of a given virtual IPv4 address by both an IP protocol translator and a recursive DNS name server must be avoided. This document assumes that sufficient coordination is in place between IP protocol translators and recursive DNS name servers. IP protocol translators and recursive DNS name servers may be co-located to simplify their coordination, or they may coordinate by means of a protocol specified outside of this document. It is recommended that the maintenance of virtual IPv4 addresses is with the IP protocol translator to which they route. This accelerates packet forwarding when the creation of mapping state is initiated by an IP protocol translator, and it hence accounts for the commonly stricter delay constraints on packet forwarding in routers compared to those on DNS lookups.

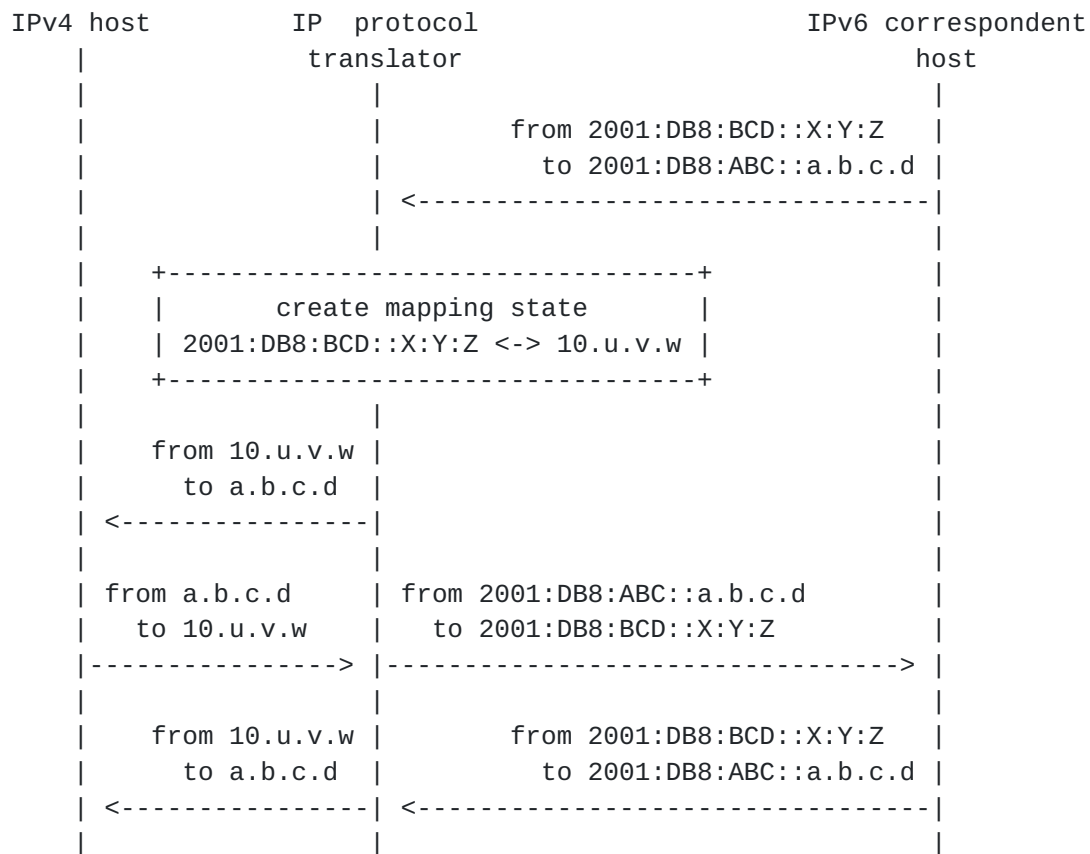


Figure 5: Mapping state creation and packet translation for a communication initiated by an IPv6 correspondent host

Figure 5 illustrates the mapping state creation and the translation of packets between IPv4 and IPv6 for a communication that is initiated by a remote IPv6 correspondent host. The packets are exchanged between a local host in an IPv4-only network on the left, and the remote IPv6 correspondent host on the right. The IPv4-only network uses Virtual IPv6 Connectivity, so the packets traverse an IP protocol translator. Continuing from previous examples, virtual IPv6 addresses are taken from the IPv6 address prefix 2001::DB8:ABC::/96, and virtual IPv4 addresses are taken from the private net-10 address space, 10.0.0.0/8. The local host's regular IPv4 address, a.b.c.d, maps to the virtual IPv6 address 2001:DB8:ABC::a.b.c.d. This mapping is constant. The remote correspondent host's regular IPv6 address, 2001:DB8:BCD::X:Y:Z, maps to the virtual IPv4 address 10.u.v.w. This mapping is temporary and created in on-demand manner. The dynamic state for the mapping between the regular IPv6 address 2001:DB8:BCD::X:Y:Z and the virtual IPv4 address 10.u.v.w is created when the first packet in the communication reaches the IP protocol translator. Once the state is created, subsequent packets from the same communication can be translated from IPv6 to IPv4 and vice versa, as shown in the

figure.

6. Multi-Homing Support

A technique for networks to increase the performance and reliability of their Internet connectivity is to establish multiple border links to either the same or different providers. This technique is called "multi-homing". To combine multi-homing with Virtual IPv6 Connectivity, an IPv4-only network must ensure that packets exchanged between local IPv4 hosts and remote IPv6 correspondent hosts pass through an IP protocol translator that is aware of the mappings for the virtual IP addresses used in the packets. This can be achieved in one of the following two ways:

- o Shared virtual IP addresses -- A route to a virtual IPv4 or IPv6 address originates at all IP protocol translators, and all IP protocol translators can handle the mapping for all virtual IP addresses. This requires synchronization of mapping state across IP protocol translators.
- o Unshared virtual IP addresses -- A route to each virtual IPv4 or IPv6 address originates at exactly one IP protocol translator, and only that IP protocol translator can handle the mapping for the virtual IP address. It is then sufficient to maintain mapping state for a given virtual IPv4 address in only one IP protocol translator.

Sharing virtual IP addresses across multiple IP protocol translators provides a basis for automated fail-over and dynamic load balancing between border links. Fail-over from a failing IP protocol translator to a backup happens automatically due to the redundancy of routes for a given virtual IP address. Dynamic load balancing can be achieved if the routes to virtual IP addresses are dynamically announced. Virtual IP addresses can then be dynamically divided across available IP protocol translators. On the other hand, shared virtual IP addresses require IP protocol translators to map the complete set of virtual IP addresses in a consistent manner. This implies two costs: First, it involves more mapping state for each IP protocol translator, since mappings must be stored even when unused. Second, it necessitates synchronization between IP protocol translators to ensure mapping consistency. Whereas the mapping consistency for virtual IPv6 addresses can be guaranteed by pre-configuration due to the staticness of these mappings, mappings for virtual IPv4 addresses are dynamic and must hence be synchronized across IP protocol translators.

Synchronization of mapping state for virtual IPv4 addresses may be

proactive or on-demand. Proactive synchronization ensures that IP protocol translators have a consistent view of all mappings at any time. This requires immediate mapping updates whenever a new mapping is created. On-demand synchronization enables IP protocol translators to retrieve any missing mapping they may need to process a packet. The mappings of virtual IPv4 addresses may then be centrally maintained in a single database that IP protocol translators can query when needed.

Unshared virtual IP addresses can be used without synchronization between IP protocol translators. It is instead sufficient to ensure that the virtual IPv4 and IPv6 addresses that are used in the mappings for a particular communication belong to partitions from the same IP protocol translator. The disjoint route announcements of the IP protocol translators then ensure that packets exchanged between IPv4 and IPv6 hosts always route to the IP protocol translator that can map the IP addresses in the packets. A disadvantage of partitioned virtual IP addresses is that active communications cannot be switched to a different IP protocol translator for the purpose of fail-over or load balancing because they are bound to a route via a particular IP protocol translator.

Although active communications cannot be switched between IP protocol translators, communications that are newly established should be able to go via any IP protocol translator of an IPv4-only network. The partitioning of virtual IPv4 addresses does not prevent this: Since the mappings for virtual IPv4 addresses are dynamically created, they can be created with a virtual IPv4 address that routes to a preferred IP protocol translator. However, the mappings for virtual IPv6 addresses cannot reflect a current preference for a specific IP protocol translator because those mappings are constant. So to enable the establishment of a new communication via any IP protocol translator, each regular IPv4 address of a host within the IPv4-only network must be mapped to multiple virtual IPv6 addresses, one for each IP protocol translator.

The decision of whether to use shared vs. partitioned virtual IP addresses may have an impact on whether a multi-homed IPv4 network can use virtual IPv6 addresses that have been allocated to its provider: If the border links of the multi-homed IPv4 network connect to different providers, virtual IPv6 addresses must be provider-independent in order to be shared across IP protocol translators. This may cause additional load for the global routing system because routes towards the virtual IPv6 addresses could not be announced in aggregated manner. It is therefore recommended that multi-homed IPv4 networks with border links to multiple providers use virtual IPv6 addresses that are provider-allocated, and hence by necessity partitioned virtual IP addresses.

7. Side-Effects

The translation of IP addresses in the network without explicit coordination with applications and host stacks inevitably leads to side-effects. The following explains the side-effects of Virtual IPv6 Connectivity, and shows why those are unavoidable given the objectives for which Virtual IPv6 Connectivity was designed.

The side-effects of Virtual IPv6 Connectivity are three-fold:

- o It may interfere with applications that expect addresses not to change in packets en route.
- o It may cause the establishment of new communication sessions to fail due to the unavailability or staleness of a virtual IPv4 address.
- o It requires the use of a recursive DNS server for name resolution by IPv4-only hosts.

Interferences may affect applications that use address referrals inside packet payloads. Since the translation between IPv4 and IPv6 addresses affects only the IP headers in packets, but no address referrals inside packet payloads, addresses referenced in packet payloads may become unreachable when a packet traverses an address translator. Special support, either in the applications themselves or in the address translator, can mitigate the problem. But such special support cannot be guaranteed for every application.

Unavailability of virtual IPv4 addresses may cause the establishment of new communication sessions to fail if a large number of IPv6-only hosts communicate with an IPv4-only network. In such a case, the pool of virtual IPv4 addresses may temporarily be exhausted. Staleness of virtual IPv4 addresses can cause session establishment failures if an application uses a virtual IPv4 address a long time after it was mapped to the corresponding regular IPv6 address. The virtual IPv4 address may at that point have been mapped to a different regular IPv6 address, as mappings between regular IPv6 addresses and virtual IPv4 addresses are dynamic.

The need for recursive DNS servers in IPv4-only networks precludes direct resolution of DNS names by hosts, and requires configuration of hosts in IPv4-only networks with a recursive DNS server. While this requirement is often negligible due to the widespread use of recursive DNS servers, preventing direct DNS name resolution altogether is typically difficult.

All of these side-effects are unavoidable, since they are a

consequence of the objectives for which Virtual IPv6 Connectivity was designed, that is, to enable bilateral communications between IPv4-only hosts and IPv6-only peers with upgrades only to IPv4-only networks:

- o The objective to upgrade only IPv4-only networks requires that addresses are translated, in the network, without explicit coordination with host stacks and applications. This may cause interference with applications that expect addresses not to change in packets en route.
- o The objective to enable bilateral communications between IPv4-only hosts and IPv6-only peers requires that IPv6-only peers must be represented to IPv4-only hosts via virtual IPv4 addresses. The mappings between regular IPv6 addresses and virtual IPv4 addresses must be dynamic, since no set of IPv4 addresses can be large enough to provide for static virtual IPv4 addresses for all potential IPv6-only peers. This may cause the establishment of new communication sessions to fail due to the unavailability or staleness of a virtual IPv4 address.
- o The need to translate addresses without explicit coordination with host stacks and applications calls for the use of recursive DNS servers by IPv4-only hosts, in order to handle the translation of DNS messages from regular IPv6 addresses to virtual IPv4 addresses. Even though it would be possible to translate DNS messages without recursive DNS server, doing so would defeat the use of DNS security extensions.

8. Security Considerations

No security considerations yet; to be written down.

9. IANA Considerations

This document has no actions for IANA.

10. Acknowledgment

Many ideas in Virtual IPv6 Connectivity were originally proposed in 2002 by Alain Durand [[draft-durand-ngtrans-nat64-nat46](#)].

The authors would like to thank Edward Jankiewicz, Dave Thaler, and Brian Carpenter for their reviews of this document and their valuable comments.

This document was generated using the xml2rfc tool.

11. References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2765] Nordmark, E., "Stateless IP/ICMP Translation Algorithm (SIIT)", [RFC 2765](#), February 2000.

Appendix A. Pending Document Changes

This section identifies pending document changes, proposed by a reviewer or author, that have not been addressed so far.

- o Describe the extent to which Virtual IPv6 Connectivity supports DNSSEC, and explain how DNSSEC can be put to use. (Comment from Brian Carpenter.)
- o Describe how reverse DNS lookups function for virtual IP addresses. (Comment from Brian Carpenter.)
- o Integrate support for IPv4/IPv4 address translation for communication sessions with IPv4 peers? (Comment from Edward Jankiewicz. Needs working group feedback.)

Appendix B. Log of Document Changes

The following is a list of technical changes that were made from version 00 to version 01 of the document. Editorial revisions are not explicitly mentioned. Part of the technical and editorial document changes address review comments received from Dave Thaler.

- o Revised abstract and introduction to emphasize that the impetus to invest in IPv4/IPv6 interworking will shift towards the legacy IPv4 Internet as IPv6 becomes more deployed.
- o Clarified that virtual IP address prefixes used in figures and examples throughout the document are not mandatory. Which prefixes are used is up to the network operator/administrator deploying Virtual IPv6 Connectivity.
- o Clarified in which situations an a-priori DNS name resolution is needed to initiate a new communication session.

- o Elaborated on how routes towards virtual IP addresses can be established. Besides announcement of these routes by IP address translators, the document now also mentions static configuration, and aggregation of the routes within a default route.
- o Highlighted the importance of coordination between IP protocol translators and recursive DNS servers. Still need to specify such coordination in more detail.
- o Explained how dynamic DNS updates work. This required a clarification of how authoritative DNS servers generate type-AAAA records for virtual IPv6 addresses.
- o Elaborated on the pros and cons of the two multi-homing techniques described in [section 6](#), "Multi-Homing". [Section 6](#) still requires more work.

The following is a list of technical changes that were made from version 01 to version 02 of the document. Editorial revisions are not explicitly mentioned. Part of the technical and editorial document changes address review comments received from Edward Jankiewicz.

- o Revised the conceptual overview.
- o Swapped the order of sections [4](#) and [5](#), "IP Protocol Translation" and "Discovery of Virtual IP Addresses", respectively. The reason for originally having the former section first was that it included information about address mappings that was useful for the latter section. However, the comment that the new order is easier to follow for the reader is a good one.
- o Revised sections [4](#) and [5](#), "IP Protocol Translation" and "Discovery of Virtual IP Addresses", to accommodate the change of order of these sections (see previous bullet).
- o Clarified that address mappings are maintained by IP protocol translators, not by recursive DNS servers. Recursive DNS servers may only trigger the creation of a new address mapping during DNS name resolution. Elaborated further on the coordination between IP protocol translators and recursive DNS servers.
- o Included a note that the attempts to retrieve regular IPv4 and IPv6 addresses by a recursive DNS server could be initiated simultaneously to reduce latency.
- o Elaborated on the mapping management for the case when two IPv4-only hosts want to communicate with the same remote IPv6-only

peer. In this case, the same mapping should be used for both communication sessions to reduce the number of virtual IPv4 addresses needed.

- o Further review comments on abstract and introduction were already addressed by the document revision from version 00 to version 01.

Authors' Addresses

Christian Vogt
Ericsson Research, NomadicLab
Hirsalantie 11
02420 Jorvas
Finland

Email: christian.vogt@ericsson.com

Alain Durand
Comcast
1500 Market Street
Philadelphia, PA 19102
USA

Email: alain_durand@cable.comcast.com

