

Internet Engineering Task Force  
Internet Draft  
Expiration Date: August 2003

S. Venaas  
UNINETT

February 2003

An IPv4 - IPv6 multicast gateway

[draft-venaas-mboned-v4v6mcastgw-00.txt](#)

#### Status of this Memo

This document is an Internet-Draft and is subject to all provisions of [Section 10 of RFC2026](#).

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/lid-abstracts.txt>.

To view the list Internet-Draft Shadow Directories, see <http://www.ietf.org/shadow.html>.

#### Abstract

This document describes an IPv4 - IPv6 gateway solution that embeds all IPv4 multicast group addresses into IPv6, and allows IPv6 hosts to receive from and send to IPv4 multicast groups.

---

Internet Draft    [draft-venaas-mboned-v4v6mcastgw-00.txt](#)    February 2003

## Table of Contents

<a href="#">1.</a>	<a href="#">Introduction .....</a>	<a href="#">2</a>
<a href="#">2.</a>	<a href="#">Embedding IPv4 multicast group addresses into IPv6 .....</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">Architecture .....</a>	<a href="#">3</a>
<a href="#">4.</a>	<a href="#">Address rewriting .....</a>	<a href="#">4</a>
<a href="#">5.</a>	<a href="#">Examples .....</a>	<a href="#">4</a>
<a href="#">5.1.</a>	<a href="#">IPv6 host joining a group inside the /96 prefix .....</a>	<a href="#">4</a>
<a href="#">5.2.</a>	<a href="#">IPv6 host sending to group inside the /96 prefix .....</a>	<a href="#">5</a>
<a href="#">6.</a>	<a href="#">Issues .....</a>	<a href="#">5</a>
<a href="#">7.</a>	<a href="#">Acknowledgments .....</a>	<a href="#">5</a>
<a href="#">8.</a>	<a href="#">Security Considerations .....</a>	<a href="#">5</a>
<a href="#">9.</a>	<a href="#">References .....</a>	<a href="#">6</a>
<a href="#">9.1.</a>	<a href="#">Normative References .....</a>	<a href="#">6</a>
<a href="#">9.2.</a>	<a href="#">Informative References .....</a>	<a href="#">6</a>
	<a href="#">Author's Address .....</a>	<a href="#">6</a>
	<a href="#">Appendix A: Source addressing issues .....</a>	<a href="#">6</a>
	<a href="#">Appendix B: Possible enhancements .....</a>	<a href="#">7</a>
	<a href="#">Appendix C: Comparison with MTP .....</a>	<a href="#">7</a>

## [1.](#) Introduction

IPv4 and IPv6 will co-exist for many years, possibly decades. There are several solutions for how IPv4 and IPv6 hosts and networks can inter-operate. This is usually easy if a host is dual stack. If however an IPv6-only host needs to communicate with an IPv4-only host, then somewhere along the data path there must be some form of translation. There are several ways of doing this for unicast, while for multicast the only mechanism known to the author is [[MTP](#)].

Here we describe a possible multicast gateway solution. This gateway could be placed at the border between IPv6-only and IPv4-only networks to allow multicast access between them. The goal is to give an IPv6 host full access to send to and receive from any IPv4 multicast group by using the usual IPv6 multicast protocols and applications which will then operate on the respective IPv6 groups. The gateway solution can be used with no changes to other infrastructure.

We will define a one-to-one mapping of IPv4 addresses onto a subset

of the IPv6 multicast addresses. An IPv6 host will then be able to receive data from any IPv4 multicast group by joining the corresponding IPv6 group. An IPv6 host can also send, without necessarily joining, to any IPv4 multicast group by sending to the corresponding IPv6 group.

---

Internet Draft     [draft-venaas-mboned-v4v6mcastgw-00.txt](#)     February 2003

## [2.](#) Embedding IPv4 multicast group addresses into IPv6

We need a way of referring to an IPv4 multicast group using an IPv6 address. We do this by embedding IPv4 multicast addresses into IPv6 by prepending them with a specific /96 IPv6 prefix such that for each IPv4 multicast address we have a respective IPv6 multicast address. Depending on the prefix they might be of any scope desired.

An administrator deploying a gateway needs to choose a /96 prefix in accordance with the IPv6 multicast address format defined in [section 2.7](#) of [\[ADDRARCH\]](#).

The addresses used will then be of the form FFxx:<blah>:<IPv4> where flags, scope and the value of "blah" are chosen by the administrator. "IPv4" is the last 32 bits specifying the IPv4 address of the IPv4 multicast group. The administrator may choose to use Unicast-Prefix-based multicast addresses as defined in [\[UNIPRFXM\]](#).

## [3.](#) Architecture

The gateway makes use of PIM Sparse Mode [\[PIM-SM\]](#). It is a complete IPv6 PIM-SM router that also is the RP for the /96 IPv6 prefix. With respect to the IPv4 network, it behaves as an IPv4 multicast host. When it receives a PIM join message for a new IPv6 group inside the /96 prefix, it will join the IPv4 multicast group specified by the last 32 bits in the address. It should also do this if it gets MLD listener reports for such groups on links where it is the DR.

When an IPv6 source starts sending, the data will reach the gateway. If the gateway is the DR for the source, it will receive the packets natively and resend them as IPv4 multicast provided that the multicast address is within the /96 prefix, and the last 32 bits form a valid IPv4 multicast address. If the gateway is not the DR, then it will receive PIM register messages. Again if the multicast address fulfills the requirements above, the multicast packets are resent as IPv4 multicast. When receiving register messages, it may, according

to normal PIM behaviour, join the IPv6 group to receive packets natively instead. These packets are also resent.

Note that by being the Rendezvous Point, it can keep track of all IPv6 sources and receive all their data. And it will also know which groups there are listeners for. It does not however, have knowledge of IPv4 sources and listeners. A drawback with this, is that it will resend IPv6 data even if there are no IPv4 listeners, and it will also join and wait for IPv4 data even if there are no sources.

#### [4.](#) Address rewriting

When IPv4 packets are resent as IPv6 we will need to replace the source and destination addresses with suitable IPv6 addresses. And similar replacement going from IPv6 to IPv4.

The destination address is easy. That is the multicast address. As described above, we map IPv4 multicast addresses into IPv6 by prepending them with a /96-prefix. And going the other direction, we simply extract the last 32 bits.

For the source address we propose using one fixed IPv4 unicast address, and one fixed IPv6 unicast address. There are no special requirements, they might be any unicast addresses assigned to the router. From the perspective of an IPv6 receiver, the gateway will look like the source of all data resent from IPv4. Similarly in the other direction.

#### [5.](#) Examples

To illustrate how the gateway works, we will look at two examples. In both examples we assume that there are no previous state in the gateway.

##### [5.1.](#) IPv6 host joining a group inside the /96 prefix

An IPv6 host joins the group FFxx:<blah>:a.b.c.d. If the gateway is the DR for the host, it will receive an MLD membership report. If not, it will receive a PIM join since it is the RP for the group. The

gateway will then get (\*, G) state for the group. So far this is normal PIM behaviour. The gateway checks whether the address is inside the /96 prefix, and whether the last 32 bits (a.b.c.d) is an IPv4 multicast address. If it is, it joins a.b.c.d using IGMP, and stays joined as long as it has state for the group.

When the gateway receives a multicast packet for a.b.c.d it prepends the /96 prefix to form the IPv6 address FFxx:<blah>a.b.c.d. If the gateway has outgoing interfaces for this group, it will send an IPv6 packet to the same interfaces to which it would have forwarded an IPv6 packet for the group. The destination address will be FFxx:<blah>a.b.c.d, and the source address will be the fixed IPv6 unicast address used for all resent packets.

## [5.2](#). IPv6 host sending to group inside the /96 prefix

An IPv6 host sends to the group FFxx:<blah>a.b.c.d. If the gateway is the DR for the host, it will receive the data natively. If not, it will receive PIM register messages containing the data since it's the RP. For each packet received, either natively or inside register messages, it will first check that the destination address is inside the /96 prefix and that the last 32 bits (a.b.c.d) is an IPv4 multicast address. If this is okay, it will resend the packet to the IPv4 address a.b.c.d. The source address is the fixed IPv4 unicast address used for all resent packets.

## [6](#). Issues

The gateway should work well for most multicast protocols and applications. Since addresses are rewritten, there might, as with [\[NAT-PT\]](#), be problems with application protocols carrying IP addresses though. There might also be issues with using the same IP source address when resending packets from different sources, see [appendix A](#).

## [7](#). Acknowledgments

The author wishes to thank Michal Przybylski and Pekka Savola for valuable comments, and also people from the M6Bone community for testing a prototype implementation.

## 8. Security Considerations

The gateway as specified in this document does not take scoping into account. Hence there is a danger that multicast content that is supposed to be available only in a small scope on one side of the gateway, becomes available in a larger scope on the other side. The gateway could possibly try to translate IPv6 scopes into IPv4 ttl values and vice versa. In order to support multiple scopes one would then use multiple /96 multicast prefixes.

One may wish to limit who can access the gateway. If for instance one wishes to restrict it to a site, one can use a /96 prefix of site-local scope, and then filter at the site border, just like one would for multicast in general. A gateway implementation could also offer a way of restricting which groups and sources should be accepted.

## 9. References

### 9.1. Normative References

- [ADDRARCH]    Hinden, R., Deering, S., "IP Version 6 Addressing Architecture", [RFC 2373](#), July 1998.
- [UNIPRFXM]    Haberman, B., Thaler, D., "Unicast-Prefix-based IPv6 Multicast Addresses", [RFC 3306](#), August 2002.
- [PIM-SM]      Estrin, D., Farinacci, D., Helmy, A., Thaler, D., Deering, S., Handley, M., Jacobson, V., Liu, C., Sharma, P. and L. Wei, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", [RFC 2362](#), June 1998.

## [9.2.](#) Informative References

- [MTP] Tsuchiya, K., Higuchi, H., Sawada, S., Nozaki, S., "An IPv6/IPv4 Multicast Translator based on IGMP/MLD Proxying (mtp)", work-in-progress, [draft-ietf-ngtrans-mtp-03.txt](#), October 2002.
- [NAT-PT] Tsirtsis, G., Srisuresh, P., "Network Address Translation - Protocol Translation (NAT-PT)", [RFC 2766](#), February 2000.

### Author's Address

Stig Venaas  
UNINETT  
Trondheim, Norway  
Email: venaas@uninett.no

### Appendix A: Source addressing issues

As specified above, we use one fixed IPv4 unicast address and one fixed IPv6 unicast address for all multicast packets resent by the gateway. This works fine for common mbone tools like vic and rat. It may cause problems for some applications though. If there are multiple sources sending to the same group on one side of the gateway, an application running on a host on the other side may be unable to know which packets come from which source.

The idea is that for packets with different source addresses on one side of the gateway, there should also be different source addresses on the other side. This can be done by rewriting these unicast address like one would do for [\[NAT-PT\]](#). Rewriting IPv4 into IPv6 is

simple, we can use the same trick as for multicast and use a /96 prefix. Note that this must be different from the prefix used for multicast since the first bits are different in unicast and multicast addresses. Going from IPv6 to IPv4 we have a much bigger problem. We suggest using a pool of IPv4 addresses being dynamically allocated to the different IPv6 sources. If the total amount of IPv6 sources is larger than the number of addresses in the pool, one might reuse addresses between groups, so that the size of the pool would only

need to be as large as the largest number of sources in each group. Note that these unicast addresses are only used by the gateway as source addresses, but they must still have valid routes for PIM with its RPF checks to work.

## Appendix B: Possible enhancements

The main draft documents what we see as a complete working gateway solution. There are however several enhancements possible.

As specified, the gateway operates as an IPv4 host using IGMP. One could possibly let the gateway be an IPv4 PIM router. It could then stop sending IPv4 packets if it receives register stop messages from the RP. When it receives register stop messages, it could itself send register stop messages for IPv6 sources.

One could possibly add SSM support. One might consider using SSM to reach the gateway, and not necessarily let it be an RP. One could also allow IPv6 hosts to join specific IPv4 sources, by using some /96 IPv6 unicast prefix to embed IPv4 addresses into IPv6. In the first case we go from IPv6 SSM to IPv4 ASM. In the second we go from IPv6 SSM to IPv4 SSM.

## Appendix C: Comparison with MTP

The gateway solution described in this draft has some resemblance to [\[MTP\]](#). They both try to offer multicast connectivity between IPv4-only and IPv6-only hosts with some sort of translation device placed at the border between the IPv4 and IPv6 domains.

The main difference between the two is perhaps that MTP only operates at the IGMP and MLD level. This solution uses standard PIM and MLD mechanisms to know which groups to resend, while MTP as specified, requires an administrator to configure which groups to resend. This might limit the number of groups that can be resent, and there is a risk that one keeps resending data when there are no receivers present. MTP might be used together with some out-of-band mechanism for the user or application to signal interest, this will however make the solution less transparent to the users, or require software modifications.