Softwire WG
Internet-Draft
Intended status: Standards Track
Expires: May 17, 2017

M. Xu
Y. Cui
J. Wu
S. Yang
Tsinghua University
C. Metz
G. Shepherd
Cisco Systems

November 13, 2016

Softwire Mesh Multicast
draft-ietf-softwire-mesh-multicast-14

Abstract

The Internet needs to support IPv4 and IPv6 packets. Both address families and their related protocol suites support multicast of the single-source and any-source varieties. During IPv6 transition, there will be scenarios where a backbone network running one IP address family internally (referred to as internal IP or I-IP) will provide transit services to attached client networks running another IP address family (referred to as external IP or E-IP). It is expected that the I-IP backbone will offer unicast and multicast transit services to the client E-IP networks.

Softwire Mesh is a solution providing E-IP unicast and multicast support across an I-IP backbone. This document describes the mechanism for supporting Internet-style multicast across a set of E-IP and I-IP networks supporting softwire mesh.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 17, 2017.

Table of Contents

1. Introduction

The Internet needs to support IPv4 and IPv6 packets. Both address families and their related protocol suites support multicast of the single-source and any-source varieties. During IPv6 transition, there will be scenarios where a backbone network running one IP address family internally (referred to as internal IP or I-IP) will provide transit services to attached client networks running another IP address family (referred to as external IP or E-IP).

One solution is to leverage the multicast functions inherent in the I-IP backbone, to efficiently forward client E-IP multicast packets inside an I-IP core tree, which is rooted at one or more ingress AFBR nodes and branches out to one or more egress AFBR leaf nodes.

[RFC4925] outlines the requirements for the softwires mesh scenario and includes support for multicast traffic. It is likely that client E-IP multicast sources and receivers will reside in different client E-IP networks connected to an I-IP backbone network. This requires the client E-IP source-rooted or shared tree to traverse the I-IP backbone network.

One method of accomplishing this is to re-use the multicast VPN approach outlined in [RFC6513]. MVPN-like schemes can support the softwire mesh scenario and achieve a "many-to-one" mapping between the E-IP client multicast trees and the transit core multicast trees. The advantage of this approach is that the number of trees in the I-IP backbone network scales less than linearly with the number of E-IP client trees. Corporate enterprise networks and by extension multicast VPNs have been known to run applications that create too many (S,G) states. Aggregation at the edge contains the (S,G) states for customer's VPNs and these need to be maintained by the network operator. The disadvantage of this approach is the possibility of inefficient bandwidth and resource utilization when multicast packets are delivered to a receiving AFBR with no attached E-IP receivers.

Internet-style multicast is somewhat different in that the trees are source-rooted and relatively sparse. The need for multicast aggregation at the edge (where many customer multicast trees are

mapped into one or more backbone multicast trees) does not exist and to date has not been identified. Thus the need for a basic or closer alignment with E-IP and I-IP multicast procedures emerges.

[RFC5565] describes the "Softwire Mesh Framework". This document provides a more detailed description of how one-to-one mapping schemes ([RFC5565], Section 11.1) for IPv6 over IPv4 and IPv4 over IPv6 can be achieved.

1.1. 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 [RFC2119].

2. Terminology

Figure 1 shows an example of how a softwire mesh network can support multicast traffic. A multicast source S is located in one E-IP client network, while candidate E-IP group receivers are located in the same or different E-IP client networks that all share a common I-IP transit network. When E-IP sources and receivers are not local to each other, they can only communicate with each other through the I-IP core. There may be several E-IP sources for a single multicast group residing in different client E-IP networks. In the case of shared trees, the E-IP sources, receivers and RPs might be located in different client E-IP networks. In the simplest case, a single operator manages the resources of the I-IP core, although the inter-operator case is also possible and so not precluded.

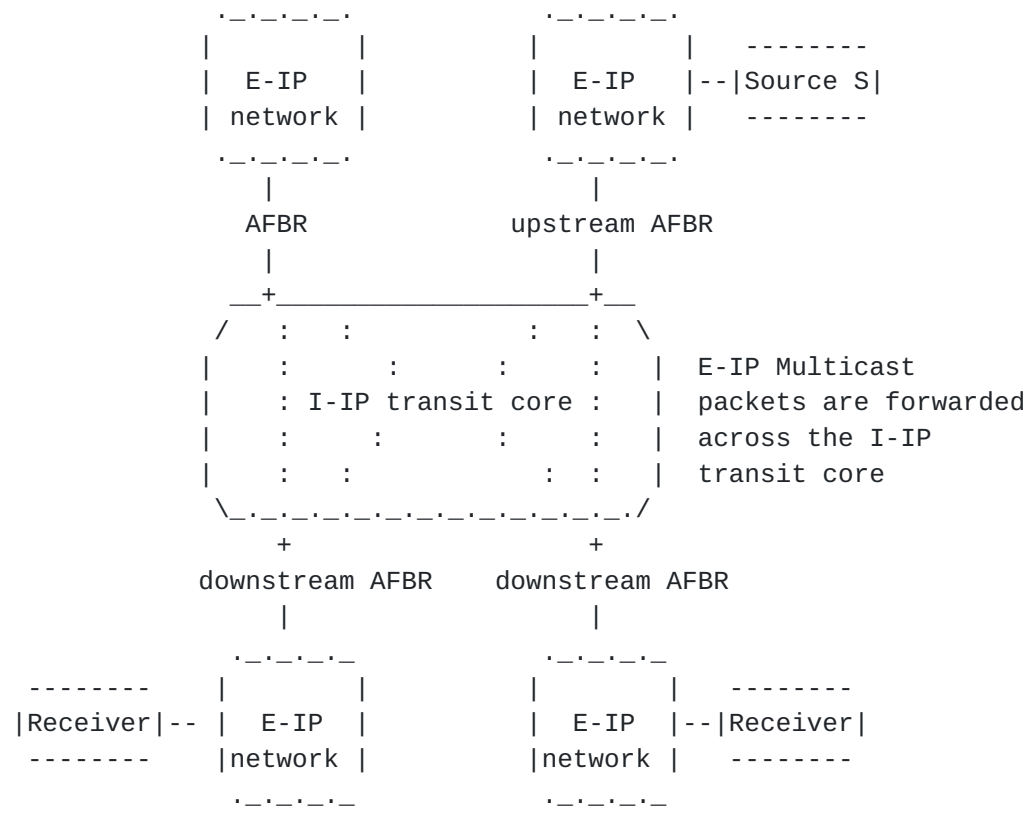


Figure 1: Softwire Mesh Multicast Framework

Terminology used in this document:

o Address Family Border Router (AFBR) - A router interconnecting two or more networks using different IP address families. In the context of softwire mesh multicast, the AFBR runs E-IP and I-IP control planes to maintain E-IP and I-IP multicast states respectively and performs the appropriate encapsulation/decapsulation of client E-IP multicast packets for transport across the I-IP core. An AFBR will act as a source and/or receiver in an I-IP multicast tree.

o Upstream AFBR: The AFBR router that is located on the upper reaches of a multicast data flow.

o Downstream AFBR: The AFBR router that is located on the lower reaches of a multicast data flow.

o I-IP (Internal IP): This refers to IP address family (i.e., either IPv4 or IPv6) that is supported by the core (or backbone) network. An I-IPv6 core network runs IPv6 and an I-IPv4 core network runs IPv4.

o E-IP (External IP): This refers to the IP address family (i.e. either IPv4 or IPv6) that is supported by the client network(s) attached to the I-IP transit core. An E-IPv6 client network runs IPv6 and an E-IPv4 client network runs IPv4.

o I-IP core tree: A distribution tree rooted at one or more AFBR source nodes and branched out to one or more AFBR leaf nodes. An I-IP core tree is built using standard IP or MPLS multicast signaling protocols operating exclusively inside the I-IP core network. An I-IP core tree is used to forward E-IP multicast packets belonging to E-IP trees across the I-IP core. Another name for an I-IP core tree is multicast or multipoint softwire.

o E-IP client tree: A distribution tree rooted at one or more hosts or routers located inside a client E-IP network and branched out to one or more leaf nodes located in the same or different client E-IP networks.

o uPrefix64: The /96 unicast IPv6 prefix for constructing an IPv4-embedded IPv6 source address in IPv6-over-IPv4 scenario.

o uPrefix46: The /96 unicast IPv6 prefix for constructing an IPv4-embedded IPv6 source address in IPv4-over-IPv6 scenario.

o mPrefix46: The /96 multicast IPv6 prefix for constructing an IPv4-embedded IPv6 multicast address in IPv4-over-IPv6 scenario.

o Inter-AFBR signaling: A mechanism used by downstream AFBRs to send PIM messages to the upstream AFBR.

3. Scenarios of Interest

This section describes the two different scenarios that softwires mesh multicast is appliacable to.

3.1. IPv4-over-IPv6

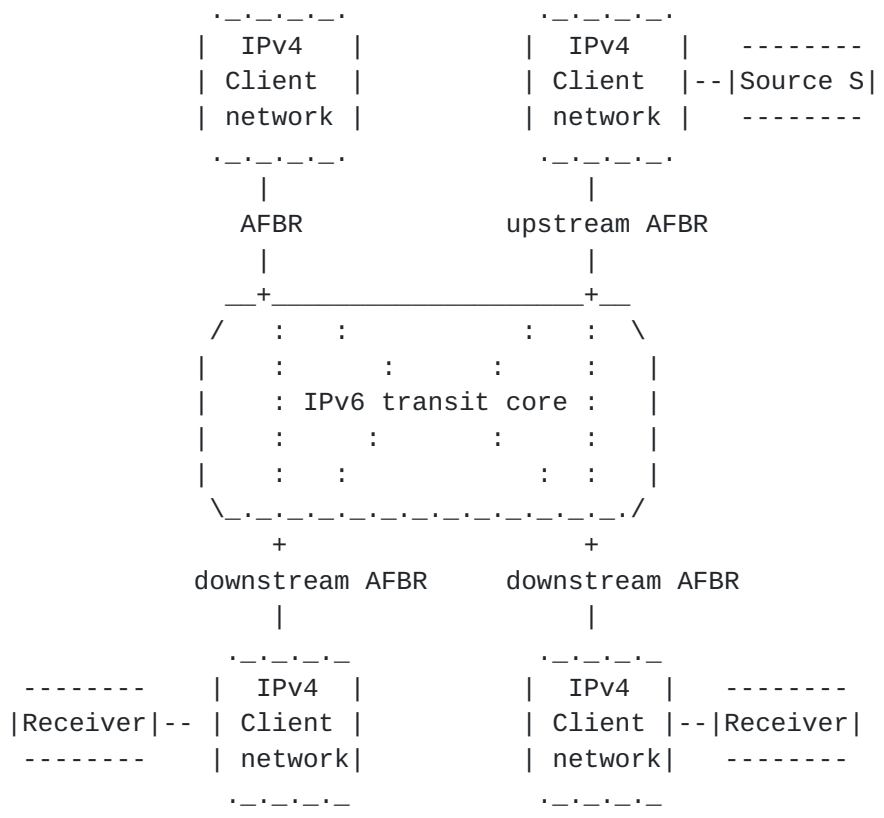


Figure 2: IPv4-over-IPv6 Scenario

In Figure 2, the E-IP client networks run IPv4 and the I-IP core runs IPv6.

Because of the much larger IPv6 group address space, the client E-IPv4 tree can be mapped to a specific I-IPv6 core tree. This simplifies operations on the AFBR because it becomes possible to algorithmically map an IPv4 group/source address to an IPv6 group/source address and vice-versa.

The IPv4-over-IPv6 scenario is an emerging requirement as network operators build out native IPv6 backbone networks. These networks support native IPv6 services and applications but in many cases, support for legacy IPv4 unicast and multicast services will also need to be accomodated.

3.2. IPv6-over-IPv4

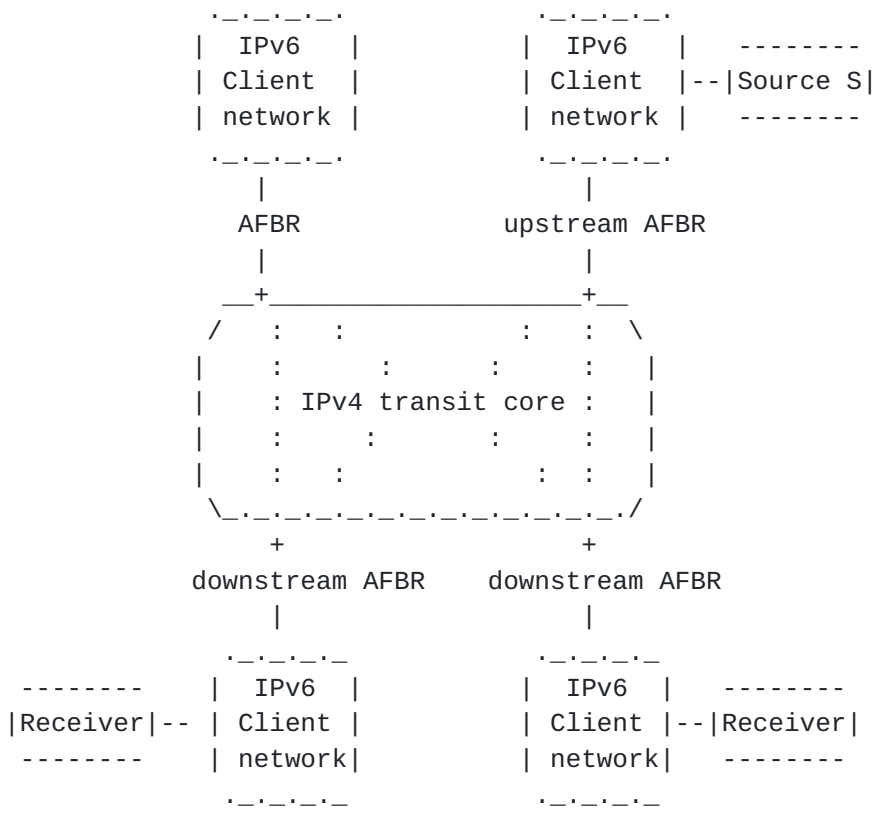


Figure 3: IPv6-over-IPv4 Scenario

In Figure 3, the E-IP Client Networks run IPv6 while the I-IP core runs IPv4.

IPv6 multicast group addresses are longer than IPv4 multicast group addresses so it is not possible to perform an algorithmic IPv6 to IPv4 address mapping without the risk of multiple IPv6 group addresses mapped to the same IPv4 address, resulting in unnecessary bandwidth and resource consumption.Therefore, additional efforts will be required to ensure that client E-IPv6 multicast packets can be injected into the correct I-IPv4 multicast trees at the AFBRs. This clear mismatch in IPv6 and IPv4 group address lengths means that it will not be possible to perform a one-to-one mapping between IPv6 and IPv4 group addresses unless the IPv6 group address is scoped, such as applying a "Well-Known" prefix or an ISP-defined prefix.

As mentioned earlier, this scenario is common in the MVPN environment. As native IPv6 deployments and multicast applications emerge from the outer reaches of the greater public IPv4 Internet, it is envisaged that the IPv6 over IPv4 softwire mesh multicast scenario will be a necessary feature supported by network operators.

4. IPv4-over-IPv6 Mechanism

4.1. Mechanism Overview

Routers in the client E-IPv4 networks have routes to all other client E-IPv4 networks. Through PIM messages, E-IPv4 hosts and routers have discovered or learnt of (S,G) or (*,G) IPv4 addresses. Any I-IPv6 multicast state instantiated in the core is referred to as (S',G') or (*,G') and is certainly separated from E-IPv4 multicast state.

Suppose a downstream AFBR receives an E-IPv4 PIM Join/Prune message from the E-IPv4 network for either an (S,G) tree or a (*,G) tree. The AFBR can translate the E-IPv4 PIM message into an I-IPv6 PIM message with the latter being directed towards the I-IP IPv6 address of the upstream AFBR. When the I-IPv6 PIM message arrives at the upstream AFBR, it MUST be translated back into an E-IPv4 PIM message. The result of these actions is the construction of E-IPv4 trees and a corresponding I-IP tree in the I-IP network. An example of the packet format and traslation is provided in Section 8.

In this case, it is incumbent upon the AFBR routers to perform PIM message conversions in the control plane and IP group address conversions or mappings in the data plane. The AFBRs perform an algorithmic, one-to-one mapping of IPv4-to-IPv6.

4.2. Group Address Mapping

For the IPv4-over-IPv6 scenario, a simple algorithmic mapping between IPv4 multicast group addresses and IPv6 group addresses is performed. Figure 4 shows the reminder of the format:

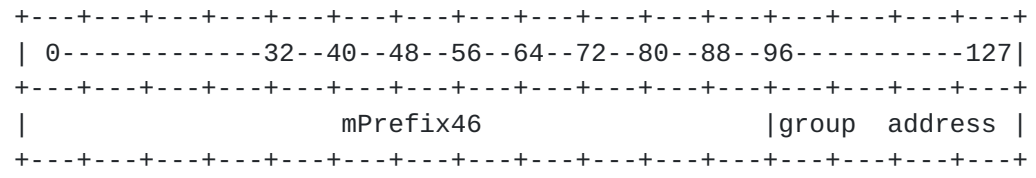


Figure 4: IPv4-Embedded IPv6 Multicast Address Format

An IPv6 multicast prefix (mPrefix46) is assigned to each AFBR. AFBRs will prepend the prefix to an IPv4 multicast group address when translating it to an IPv6 multicast group address.

The mPrefix46 for SSM mode is also defined in Section 4.1 of [RFC7371]

With this scheme, each IPv4 multicast address can be mapped into an IPv6 multicast address (with the assigned prefix), and each IPv6 multicast address with the assigned prefix can be mapped into an IPv4 multicast address.

4.3. Source Address Mapping

There are two kinds of multicast: ASM and SSM. Considering that I-IP network and E-IP network may support different kinds of multicast, the source address translation rules needed to support all possible scenarios may become very complex. But since SSM can be implemented with a strict subset of the PIM-SM protocol mechanisms [RFC7761], we can treat the I-IP core as SSM-only to make it as simple as possible. There then remain only two scenarios to be discussed in detail:

o E-IP network supports SSM

One possible way to make sure that the translated I-IPv6 PIM message reaches upstream AFBR is to set S' to a virtual IPv6 address that leads to the upstream AFBR. Figure 5 is the recommended address format based on [RFC6052]:

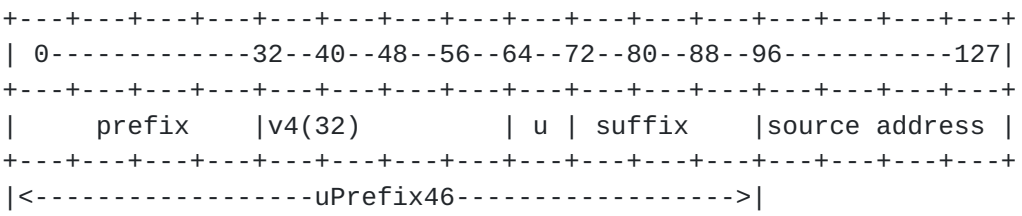


Figure 5: IPv4-Embedded IPv6 Virtual Source Address Format

In this address format,

* The "prefix" field contains a "Well-Known" prefix or an ISP-defined prefix. An existing "Well-Known" prefix is 64:ff9b, which is defined in [RFC6052];

* The "v4" field is the IP address of one of upstream AFBR's E-IPv4 interfaces;

* The "u" field is defined in [RFC4291], and MUST be set to zero;

* The "suffix" field is reserved for future extensions and SHOULD be set to zero;

* The "source address" field stores the original S.

We call the overall /96 prefix ("prefix" field and "v4" field and "u" field and "suffix" field altogether) "uPrefix46".

o E-IP network supports ASM

The (S,G) source list entry and the (*,G) source list entry only differ in that the latter has both the WC and RPT bits of the Encoded-Source-Address set, while the former is all cleared (See Section 4.9.5.1 of [RFC7761]). So we can translate source list entries in (*,G) messages into source list entries in (S'G') messages by applying the format specified in Figure 5 and clearing both the WC and RPT bits at downstream AFBRs, and vice-versa for the reverse translation at upstream AFBRs.

4.4. Routing Mechanism

In the mesh multicast scenario, routing information is REQUIRED to be distributed among AFBRs to make sure that the PIM messages that a downstream AFBR propagates reach the right upstream AFBR.

Every AFBR MUST know the /32 prefix in "IPv4-Embedded IPv6 Virtual Source Address Format". To achieve this, every AFBR should announce one of its E-IPv4 interfaces in the "v4" field, and the corresponding uPrefix46. The announcement SHOULD be sent to the other AFBRs through MBGP. Since every IP address of upstream AFBR's E-IPv4 interface is different from each other, every uPrefix46 that AFBR announces MUST be different, and uniquely identifies each AFBR. "uPrefix46" is an IPv6 prefix, and the distribution mechanism is the same as the traditional mesh unicast scenario. But "v4" field is an E-IPv4 address, and BGP messages are NOT tunneled through softwires or any other mechanism specified in [RFC5565], AFBRs MUST be able to transport and encode/decode BGP messages that are carried over I-IPv6, whose NLRI and NH are of E-IPv4 address family.

In this way, when a downstream AFBR receives an E-IPv4 PIM (S,G) message, it can translate this message into (S',G') by looking up the IP address of the corresponding AFBR's E-IPv4 interface. Since the uPrefix46 of S' is unique, and is known to every router in the I-IPv6 network, the translated message will be forwarded to the corresponding upstream AFBR, and the upstream AFBR can translate the message back to (S,G). When a downstream AFBR receives an E-IPv4 PIM (*,G) message, S' can be generated according to the format specified

in Figure 4, with "source address" field set to *(the IPv4 address of RP). The translated message will be forwarded to the corresponding upstream AFBR. Since every PIM router within a PIM domain MUST be able to map a particular multicast group address to the same RP (see Section 4.7 of [RFC7761]), when the upstream AFBR checks the "source address" field of the message, it finds the IPv4 address of the RP, and assertains that this is originally a (*,G) message. This is then translated back to the (*,G) message and processed.

5. IPv6-over-IPv4 Mechanism

5.1. Mechanism Overview

Routers in the client E-IPv6 networks contain routes to all other client E-IPv6 networks. Through PIM messages, E-IPv6 hosts and routers have discovered or learnt of (S,G) or (*,G) IPv6 addresses. Any I-IP multicast state instantiated in the core is referred to as (S',G') or (*,G') and is separated from E-IP multicast state.

This particular scenario introduces unique challenges. Unlike the IPv4-over-IPv6 scenario, it is impossible to map all of the IPv6 multicast address space into the IPv4 address space to address the one-to-one Softwire Multicast requirement. To coordinate with the "IPv4-over-IPv6" scenario and keep the solution as simple as possible, one possible solution to this problem is to limit the scope of the E-IPv6 source addresses for mapping, such as applying a "Well-Known" prefix or an ISP-defined prefix.

5.2. Group Address Mapping

To keep one-to-one group address mapping simple, the group address range of E-IP IPv6 can be reduced in a number of ways to limit the scope of addresses that need to be mapped into the I-IP IPv4 space.

For example, the high order bits of the E-IPv6 address range will be fixed for mapping purposes. With this scheme, each IPv4 multicast address can be mapped into an IPv6 multicast address (with the assigned prefix), and each IPv6 multicast address with the assigned prefix can be mapped into an IPv4 multicast address.

5.3. Source Address Mapping

There are two kinds of multicast --- ASM and SSM. Considering that I-IP network and E-IP network may support different kind of multicast, the source address translation rules needed to support all possible scenarios may become very complex. But since SSM can be implemented with a strict subset of the PIM-SM protocol mechanisms [RFC7761], we can treat the I-IP core as SSM-only to make it as

simple as possible. There then remain only two scenarios to be discussed in detail:

o E-IP network supports SSM

To make sure that the translated I-IPv4 PIM message reaches the upstream AFBR, we need to set S' to an IPv4 address that leads to the upstream AFBR. But due to the non-"one-to-one" mapping of E-IPv6 to I-IPv4 unicast address, the upstream AFBR is unable to remap the I-IPv4 source address to the original E-IPv6 source address without any constraints.

We apply a fixed IPv6 prefix and static mapping to solve this problem. A recommended source address format is defined in [RFC6052]. Figure 6 is the reminder of the format:

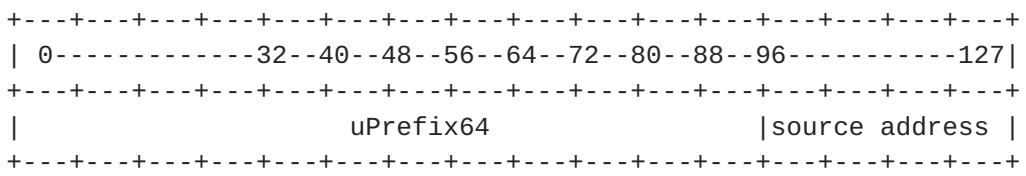


Figure 6: IPv4-Embedded IPv6 Source Address Format

In this address format, the "uPrefix64" field starts with a "Well-Known" prefix or an ISP-defined prefix. An existing "Well-Known" prefix is 64:ff9b/32, which is defined in [RFC6052]; The "source address" field is the corresponding I-IPv4 source address.

o The E-IP network supports ASM

The (S,G) source list entry and the (*,G) source list entry only differ in that the latter has both the WC and RPT bits of the Encoded-Source-Address set, while the former is all cleared (See Section 4.9.5.1 of [RFC7761]). So we can translate source list entries in (*,G) messages into source list entries in (S',G') messages by applying the format specified in Figure 5 and setting both the WC and RPT bits at downstream AFBRs, and vice-versa for the reverse translation at upstream AFBRs. Here, the E-IPv6 address of RP MUST follow the format specified in Figure 6. RP' is the upstream AFBR that locates between RP and the downstream AFBR.

5.4. Routing Mechanism

In the mesh multicast scenario, routing information is REQUIRED to be distributed among AFBRs to make sure that PIM messages that a downstream AFBR propagates reach the right upstream AFBR.

To make it feasible, the /96 uPrefix64 MUST be known to every AFBR, every E-IPv6 address of sources that support mesh multicast MUST follow the format specified in Figure 6, and the corresponding upstream AFBR of this source MUST announce the I-IPv4 address in "source address" field of this source's IPv6 address to the I-IPv4 network. Since uPrefix64 is static and unique in IPv6-over-IPv4 scenario, there is no need to distribute it using BGP. The distribution of "source address" field of multicast source addresses is a pure I-IPv4 process and no more specification is needed.

In this way, when a downstream AFBR receives a (S,G) message, it can translate the message into (S',G') by simply taking off the prefix in S. Since S' is known to every router in I-IPv4 network, the translated message will be forwarded to the corresponding upstream AFBR, and the upstream AFBR can translate the message back to (S,G) by appending the prefix to S'. When a downstream AFBR receives a (*,G) message, it can translate it into (S',G') by simply taking off the prefix in *(the E-IPv6 address of RP). Since S' is known to every router in I-IPv4 network, the translated message will be forwarded to RP'. And since every PIM router within a PIM domain MUST be able to map a particular multicast group address to the same RP (see Section 4.7 of [RFC7761]), RP' knows that S' is the mapped I-IPv4 address of RP, so RP' will translate the message back to (*,G) by appending the prefix to S' and propagate it towards RP.

6. Control Plane Functions of AFBR

AFBRs are responsible for the following functions:

6.1. E-IP (*,G) State Maintenance

When an AFBR wishes to propagate a Join/Prune(*,G) message to an I-IP upstream router, the AFBR MUST translate Join/Prune(*,G) messages into Join/Prune(S',G') messages following the rules specified above, then send the latter.

6.2. E-IP (S,G) State Maintenance

When an AFBR wishes to propagate a Join/Prune(S,G) message to an I-IP upstream router, the AFBR MUST translate Join/Prune(S,G) messages into Join/Prune(S',G') messages following the rules specified above, then send the latter.

6.3. I-IP (S',G') State Maintenance

It is possible that the I-IP transit core runs another non-transit I-IP PIM-SSM instance. Since the translated source address starts with the unique "Well-Known" prefix or the ISP-defined prefix that SHOULD NOT be used by other service provider, mesh multicast will not influence non-transit PIM-SSM multicast at all. When an AFBR receives an I-IP (S',G') message, it MUST check S'. If S' starts with the unique prefix, then the message is actually a translated E-IP (S,G) or (*,G) message, and the AFBR MUST translate this message back to E-IP PIM message and process it.

6.4. E-IP (S,G,rpt) State Maintenance

When an AFBR wishes to propagate a Join/Prune(S,G,rpt) message to an I-IP upstream router, the AFBR MUST operate as specified in Section 6.5 and Section 6.6.

6.5. Inter-AFBR Signaling

Assume that one downstream AFBR has joined a RPT of (*,G) and a SPT of (S,G), and decide to perform a SPT switchover. According to [RFC7761], it SHOULD propagate a Prune(S,G,rpt) message along with the periodical Join(*,G) message upstream towards RP. However, routers in the I-IP transit core do not process (S,G,rpt) messages since the I-IP transit core is treated as SSM-only. As a result, the downstream AFBR is unable to prune S from this RPT, so it will receive two copies of the same data of (S,G). In order to solve this problem, we introduce a new mechanism for downstream AFBRs to inform upstream AFBRs of pruning any given S from an RPT.

When a downstream AFBR wishes to propagate a (S,G,rpt) message upstream, it SHOULD encapsulate the (S,G,rpt) message, then send the encapsulated unicast message to the corresponding upstream AFBR, which we call "RP'".

When RP' receives this encapsulated message, it SHOULD decapsulate the message as in the unicast scenario, and retrieve the original (S,G,rpt) message. The incoming interface of this message may be different to the outgoing interface which propagates multicast data to the corresponding downstream AFBR, and there may be other downstream AFBRs that need to receive multicast data of (S,G) from this incoming interface, so RP' SHOULD NOT simply process this message as specified in [RFC7761] on the incoming interface.

To solve this problem as simply as possible, we introduce an "interface agent" to process all the encapsulated (S,G,rpt) messages the upstream AFBR receives, and prune S from the RPT of group G when

no downstream AFBR is subscribed to receive multicast data of (S,G) along the RPT. In this way, we ensure that downstream AFBRs will not miss any multicast data that they need, at the cost of duplicated multicast data of (S,G) along the RPT received by SPT-switched-over downstream AFBRs, if at least one downstream AFBR exists that has not yet sent Prune(S,G,rpt) messages to the upstream AFBR. The following diagram shows an example of how an "interface agent" MAY be implemented:

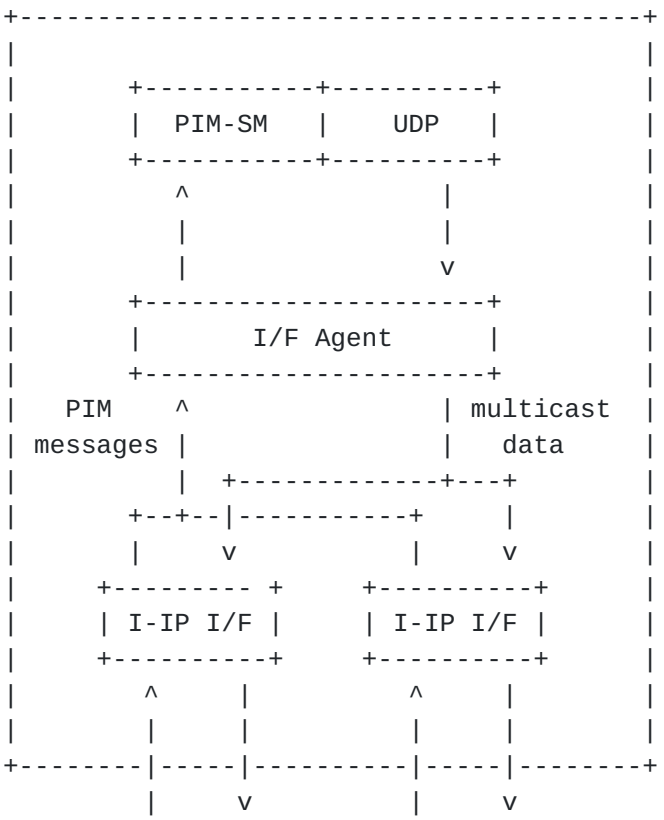


Figure 7: Interface Agent Implementation Example

Figure 7 shows an example of interface agent implementation using UDP encapsulation. The interface agent has two responsibilities: In the control plane, it SHOULD work as a real interface that has joined (*,G), representing of all the I-IP interfaces which are outgoing interfaces of the (*,G) state machine, and process the (S,G,rpt) messages received from all the I-IP interfaces. The interface agent maintains downstream (S,G,rpt) state machines of every downstream AFBR, and submits Prune (S,G,rpt) messages to the PIM-SM module only when every (S,G,rpt) state machine is at Prune(P) or PruneTmp(P')

state, which means that no downstream AFBR is subscribed to receive multicast data of (S,G) along the RPT of G. Once a (S,G,rpt) state machine changes to NoInfo(NI) state, which means that the corresponding downstream AFBR has switched to receive multicast data of (S,G) along the RPT again, the interface agent SHOULD send a Join (S,G,rpt) to the PIM-SM module immediately; In the data plane, upon receiving a multicast data packet, the interface agent SHOULD encapsulate it at first, then propagate the encapsulated packet from every I-IP interface.

NOTICE: It is possible that an E-IP neighbor of RP' that has joined the RPT of G, so the per-interface state machine for receiving E-IP Join/Prune (S,G,rpt) messages SHOULD keep alive.

6.6. SPT Switchover

After a new AFBR expresses its interest in receiving traffic destined for a multicast group, it will receive all the data from the RPT at first. At this time, every downstream AFBR will receive multicast data from any source from this RPT, in spite of whether they have switched over to an SPT of some source(s) or not.

To minimize this redundancy, it is recommended that every AFBR's SwitchToSptDesired(S,G) function employs the "switch on first packet" policy. In this way, the delay in switchover to SPT is kept as small as possible, and after the moment that every AFBR has performed the SPT switchover for every S of group G, no data will be forwarded in the RPT of G, thus no more redundancy will be produced.

6.7. Other PIM Message Types

Apart from Join or Prune, other message types exist, including Register, Register-Stop, Hello and Assert. Register and Register-Stop messages are sent by unicast, while Hello and Assert messages are only used between directly linked routers to negotiate with each other. It is not necessary to translate these for forwarding, thus the processing of these messages is out of scope for this document.

6.8. Other PIM States Maintenance

Apart from states mentioned above, other states exist, including (*,*,RP) and I-IP (*,G') state. Since we treat the I-IP core as SSM-only, the maintenance of these states is out of scope for this document.

7. Data Plane Functions of the AFBR

7.1. Process and Forward Multicast Data

On receiving multicast data from upstream routers, the AFBR checks its forwarding table to find the IP address of each outgoing interface. If there is at least one outgoing interface whose IP address family is different from the incoming interface, the AFBR MUST encapsulate/decapsulate this packet and forward it via the outgoing interface(s), then forward the data via other outgoing interfaces without encapsulation/decapsulation.

When a downstream AFBR that has already switched over to the SPT of S receives an encapsulated multicast data packet of (S,G) along the RPT, it SHOULD silently drop this packet.

7.2. Selecting a Tunneling Technology

Choosing tunneling technology depends on the policies configured on AFBRs. It is REQUIRED that all AFBRs use the same technology, otherwise some AFBRs SHALL not be able to decapsulate encapsulated packets from other AFBRs that use a different tunneling technology.

7.3. TTL

Processing of TTL depends on the tunneling technology, and it is out of scope of this document.

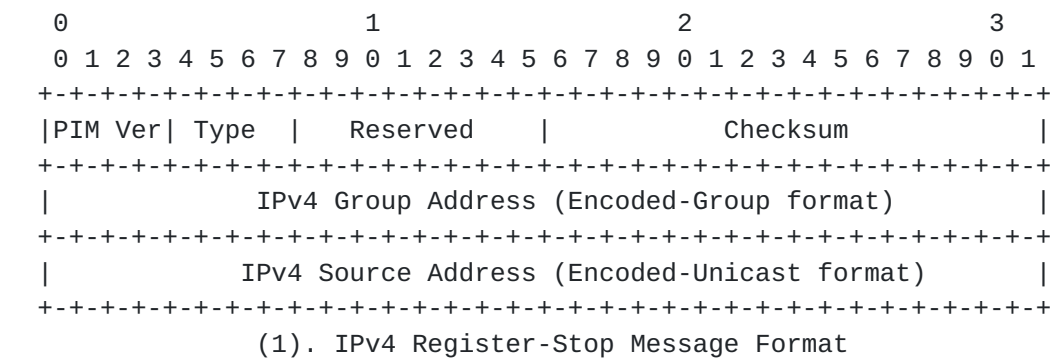
7.4. Fragmentation

The encapsulation performed by an upstream AFBR will increase the size of packets. As a result, the outgoing I-IP link MTU may not accommodate the larger packet size. As it is not always possible for core operators to increase the MTU of every link. Fragmentation after encapsulation and reassembling of encapsulated packets MUST be supported by AFBRs [RFC5565].

8. Packet Format and Translation

Because the PIM-SM Specification is independent of the underlying unicast routing protocol, the packet format in Section 4.9 of [RFC7761] remains the same, except that the group address and source address MUST be translated when traversing AFBR.

For example, Figure 8 shows the register-stop message format in IPv4 and IPv6 address family.



(1). IPv4 Register-Stop Message Format

(2). IPv6 Register-Stop Message Format

Figure 8: Register-Stop Message Format

In Figure 8, the semantics of fields "PIM Ver", "Type", "Reserved", and "Checksum" remain the same.

IPv4 Group Address (Encoded-Group format): The encoded-group format of the IPv4 group address described in Section 4.2 and 5.2.

IPv4 Source Address (Encoded-Group format): The encoded-unicast format of the IPv4 source address described in Section 4.3 and 5.3.

IPv6 Group Address (Encoded-Group format): The encoded-group format of the IPv6 group address described in Section 4.2 and 5.2.

IPv6 Source Address (Encoded-Group format): The encoded-unicast format of the IPv6 source address described in Section 4.3 and 5.3.

9. Softwire Mesh Multicast Encapsulation

Softwire mesh multicast encapsulation does not require the use of any one particular encapsulation mechanism. Rather, it must accommodate a variety of different encapsulation mechanisms, and allow the use of encapsulation mechanisms mentioned in [RFC4925]. Additionally, all of the AFBRs attached to the I-IP network MUST implement the same encapsulation mechanism.

10. Security Considerations

The security concerns raised in [RFC4925] and [RFC7761] are applicable here. In addition, the additional workload associated with some schemes could be exploited by an attacker to perform a out DDoS attack. Compared with [RFC4925], the security concerns SHOULD be considered more carefully: an attacker could potentially set up many multicast trees in the edge networks, causing too many multicast states in the core network.

11. IANA Considerations

This document includes no request to IANA.

12. References

12.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <http://www.rfc-editor.org/info/rfc4291>.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <http://www.rfc-editor.org/info/rfc4301>.

[RFC4925] Li, X., Ed., Dawkins, S., Ed., Ward, D., Ed., and A. Durand, Ed., "Softwire Problem Statement", RFC 4925, DOI 10.17487/RFC4925, July 2007, <http://www.rfc-editor.org/info/rfc4925>.

[RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, DOI 10.17487/RFC5565, June 2009, <http://www.rfc-editor.org/info/rfc5565>.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, DOI 10.17487/RFC6052, October 2010, <http://www.rfc-editor.org/info/rfc6052>.

[RFC6513] Rosen, E., Ed. and R. Aggarwal, Ed., "Multicast in MPLS/ BGP IP VPNs", RFC 6513, DOI 10.17487/RFC6513, February 2012, <http://www.rfc-editor.org/info/rfc6513>.

[RFC7761] Fenner, B., Handley, M., Holbrook, H., Kouvelas, I., Parekh, R., Zhang, Z., and L. Zheng, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", STD 83, RFC 7761, DOI 10.17487/RFC7761, March 2016, <http://www.rfc-editor.org/info/rfc7761>.

12.2. Informative References

[RFC7371] Boucadair, M. and S. Venaas, "Updates to the IPv6 Multicast Addressing Architecture", RFC 7371, DOI 10.17487/RFC7371, September 2014, <http://www.rfc-editor.org/info/rfc7371>.

Appendix A. Acknowledgements

Wenlong Chen, Xuan Chen, Alain Durand, Yiu Lee, Jacni Qin and Stig Venaas provided useful input into this document.

Authors' Addresses

Mingwei Xu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China

Phone: +86-10-6278-5822
Email: xmw@cernet.edu.cn

Yong Cui
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China

Phone: +86-10-6278-5822
Email: cuiyong@tsinghua.edu.cn

Jianping Wu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China

Phone: +86-10-6278-5983
Email: jianping@cernet.edu.cn

Shu Yang
Tsinghua University
Graduate School at Shenzhen
Shenzhen 518055
P.R. China

Phone: +86-10-6278-5822
Email: yangshu@csnet1.cs.tsinghua.edu.cn

Chris Metz
Cisco Systems
170 West Tasman Drive
San Jose, CA 95134
USA

Phone: +1-408-525-3275
Email: chmetz@cisco.com

Greg Shepherd
Cisco Systems
170 West Tasman Drive
San Jose, CA 95134
USA

Phone: +1-541-912-9758
Email: shep@cisco.com