Network Working Group
Internet-Draft
Expires: July 19, 2014

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

January 15, 2014

Softwire Mesh Multicast
draft-ietf-softwire-mesh-multicast-06

Abstract

The Internet needs to support IPv4 and IPv6 packets. Both address families and their attendant protocol suites support multicast of the single-source and any-source varieties. As part of the transition to IPv6, 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 to E-IP unicast and multicast support across an I-IP backbone. This document describes the mechanisms for supporting Internet-style multicast across a set of E-IP and I-IP networks supporting softwire mesh.

Table of Contents

1. Introduction

The Internet needs to support IPv4 and IPv6 packets. Both address families and their attendant protocol suites support multicast of the single-source and any-source varieties. As part of the transition to IPv6, 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).

The preferred solution is to leverage the multicast functions inherent in the I-IP backbone, to efficiently and scalably forward client E-IP multicast packets inside an I-IP core tree, which roots 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 including the multicast. It is straightforward to envisage 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 that the client E-IP source-rooted or shared tree should traverse the I-IP backbone network.

One method to accomplish 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 a large amount of (S,G) states. Aggregation at the edge contains the (S,G) states that need to be maintained by the network operator supporting the customer VPNs. The disadvantage of this approach is the possible inefficient bandwidth and resource utilization when multicast packets are delivered to a receiver AFBR with no attached E-IP receivers.

Internet-style multicast is somewhat different in that the trees tend to be relatively sparse and source-rooted. The need for multicast aggregation at the edge (where many customer multicast trees are mapped into a few or one 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.

A framework on how to support such methods is described in [RFC5565]. In this document, a more detailed discussion supporting the "one-to-one" mapping schemes for the IPv6 over IPv4 and IPv4 over IPv6 scenarios will be discussed.

2. Terminology

An example of a softwire mesh network supporting multicast is illustrated in Figure 1. 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 some 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 a simple case the resources of the I-IP core are managed by a single operator although the inter-provider case is not precluded.

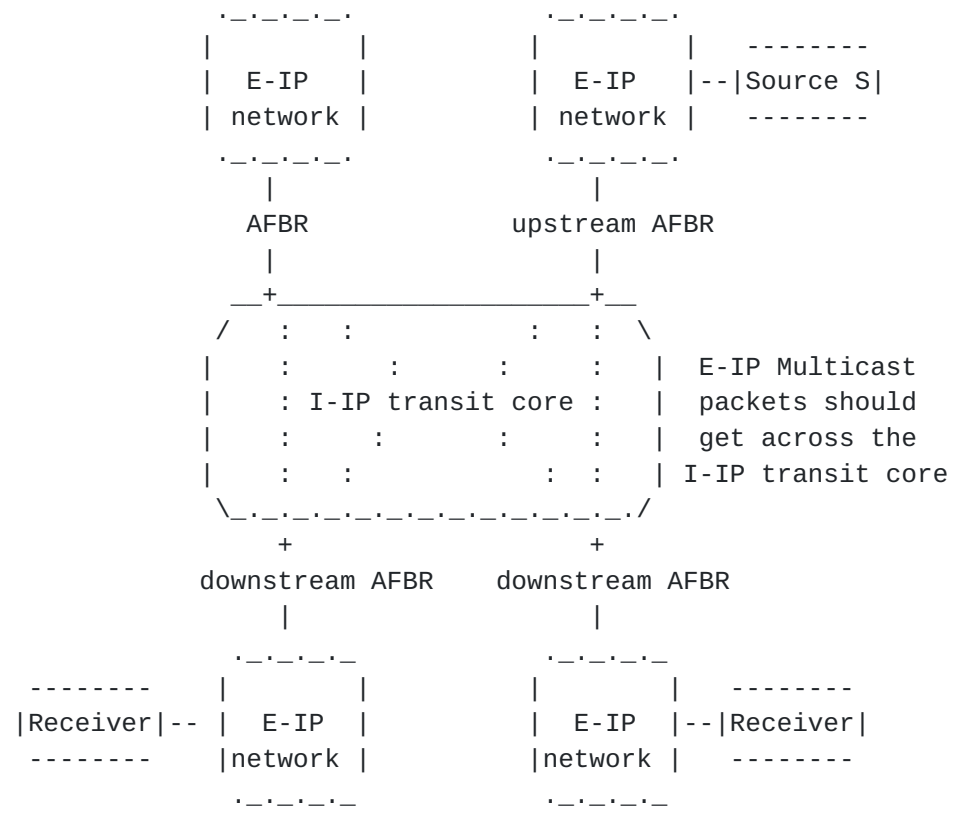

Figure 1: Softwire Mesh Multicast Framework

Terminology used in this document:

o Address Family Border Router (AFBR) - A dual-stack 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 the form of IP (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 form of IP (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 IPv4-embedded IPv6 source address.

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 where softwires mesh multicast will apply.

3.1. IPv4-over-IPv6

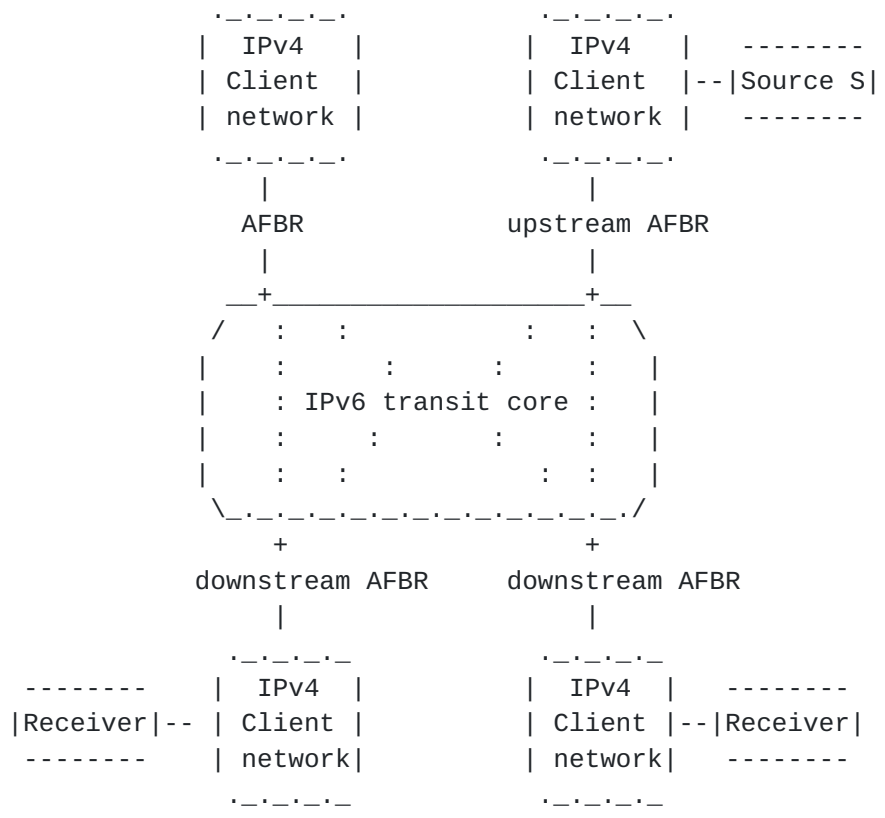

Figure 2: IPv4-over-IPv6 Scenario

In this scenario, the E-IP client networks run IPv4 and I-IP core runs IPv6. This scenario is illustrated in Figure 2.

Because of the much larger IPv6 group address space, it will not be a problem to map individual client E-IPv4 tree 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 naturally support native IPv6 services and applications but it is with near 100% certainty that legacy IPv4 networks handling unicast and multicast should be accommodated.

3.2. IPv6-over-IPv4

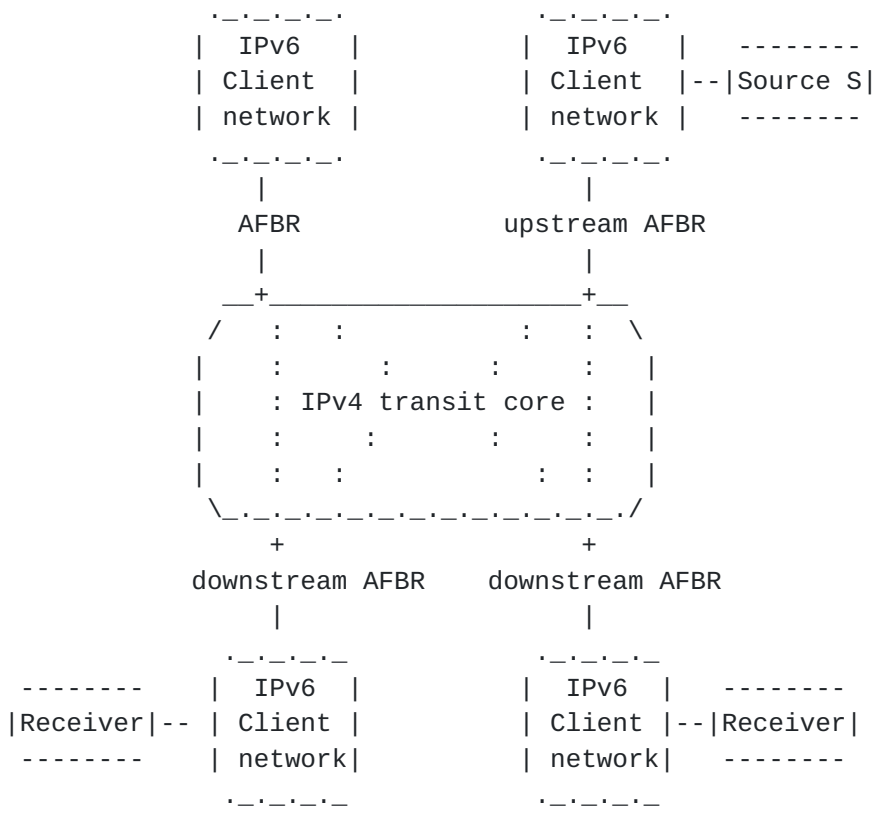

Figure 3: IPv6-over-IPv4 Scenario

In this scenario, the E-IP Client Networks run IPv6 while the I-IP core runs IPv4. This scenario is illustrated in Figure 3.

IPv6 multicast group addresses are longer than IPv4 multicast group addresses. It will not be 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.

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 contain routes to all other client E-IPv4 networks. Through the set of known and deployed mechanisms, 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 I-IP IPv6 address of the upstream AFBR. When the I-IPv6 PIM message arrives at the upstream AFBR, it should 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.

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. It becomes possible to devise an algorithmic one-to-one IPv4-to-IPv6 address mapping at AFBRs.

4.2. Group Address Mapping

For IPv4-over-IPv6 scenario, a simple algorithmic mapping between IPv4 multicast group addresses and IPv6 group addresses is supported. [I-D.ietf-mboned-64-multicast-address-format] has already defined an applicable format. Figure 4 is the reminder of the format:

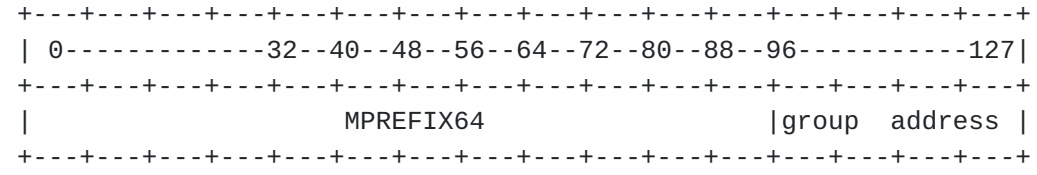

Figure 4: IPv4-Embedded IPv6 Multicast Address Format

The MPREFIX64 for SSM mode is also defined in [I-D.ietf-mboned-64-multicast-address-format] :

o ff3x:0:8000::/96 ('x' is any valid scope)

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 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 kind of multicast, the source address translation rules could be very complex to support all possible scenarios. But since SSM can be implemented with a strict subset of the PIM-SM protocol mechanisms [RFC4601], we can treat I-IP core as SSM-only to make it as simple as possible, then there remains only two scenarios to be discussed in detail:

- 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]:

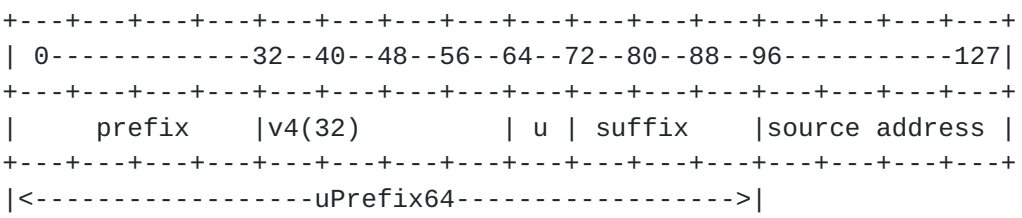

Figure 5: IPv4-Embedded IPv6 Virtual Source Address Format

  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]; "v4" field is the IP address of one of upstream AFBR's E-IPv4 interfaces; "u" field is defined in [RFC4291], and MUST be set to zero; "suffix" field is reserved for future extensions and SHOULD be set to zero; "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) "uPrefix64".

- E-IP network supports ASM

The (S,G) source list entry and the (*,G) source list entry only differ in that the latter have both the WC and RPT bits of the Encoded-Source-Address set, while the former all cleared (See Section 4.9.5.1 of [RFC4601]). 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 translate them back at upstream AFBRs vice-versa.

4.4. 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 /32 prefix in "IPv4-Embedded IPv6 Virtual Source Address Format" must be known to every AFBR, and every AFBR should not only announce the IP address of one of its E-IPv4 interfaces presented in the "v4" field to other AFBRs by MPBGP, but also announce the corresponding uPrefix64 to the I-IPv6 network. Since every IP address of upstream AFBR's E-IPv4 interface is different from each other, every uPrefix64 that AFBR announces should be different either, and uniquely identifies each AFBR. "uPrefix64" is an IPv6 prefix, and the distribution of it is the same as the distribution in the traditional mesh unicast scenario. But since "v4" field is an E-IPv4 address, and BGP messages are NOT tunneled through softwires or through any other mechanism as 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 uPrefix64 of S' is unique, and is known to every router in the I-IPv6 network, the translated message will eventually arrive at 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 eventually arrive at 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 [RFC4601]), when this upstream AFBR checks the "source address" field of the message, it'll find the IPv4 address of RP, so this upstream AFBR judges that this is originally a

(*,G) message, then it translates the message back to the (*,G) message and processes it.

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 the set of known and deployed mechanisms, 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 certainly separated from E-IP multicast state.

This particular scenario introduces unique challenges. Unlike the IPv4-over-IPv6 scenario, it's 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.

A recommended multicast address format is defined in [I-D.ietf-mboned-64-multicast-address-format]. 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 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 could be very complex to support all possible scenarios. But since SSM can be implemented with a strict subset of the PIM-SM protocol mechanisms [RFC4601], we can treat I-IP core as SSM-only to make it as simple as possible, then there remains only two scenarios to be discussed in detail:

- 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:

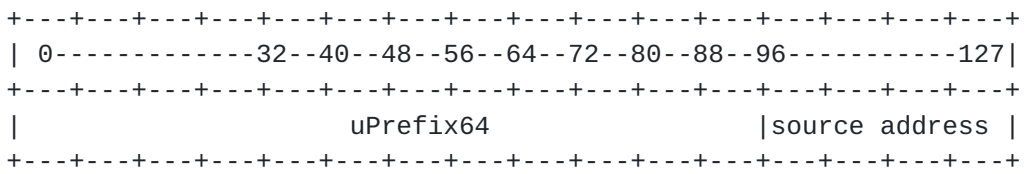

Figure 6: IPv4-Embedded IPv6 Source Address Format

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]; "source address" field is the corresponding I-IPv4 source address.

o E-IP network supports ASM

The (S,G) source list entry and the (*,G) source list entry only differ in that the latter have both the WC and RPT bits of the Encoded-Source-Address set, while the former all cleared (See Section 4.9.5.1 of [RFC4601]). 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 translate them back at upstream AFBRs vice-versa. 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 should 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 eventually arrive at 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 eventually arrive at 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 [RFC4601]), 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

The 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 there runs a non-transit I-IP PIM-SSM in the I-IP transit core. Since the translated source address starts with the unique "Well-Known" prefix or the ISP-defined prefix that should not

be used otherwise, mesh multicast won't influence non-transit PIM-SM multicast at all. When one AFBR receives an I-IP (S',G') message, it should check S'. If S' starts with the unique prefix, it means that this message is actually a translated E-IP (S,G) or (*,G) message, then the AFBR should 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 do 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 [RFC4601], it should propagate a Prune(S,G,rpt) message along with the periodical Join(*,G) message upstream towards RP. Unfortunately, routers in I-IP transit core are not supposed to understand (S,G,rpt) messages since I-IP transit core is treated as SSM-only. As a result, this downstream AFBR is unable to prune S from this RPT, then 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 RPT.

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

When RP' receives this encapsulated message, it should decapsulate this message as what it does in the unicast scenario, and get the original (S,G,rpt) message. The incoming interface of this message may be different from 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 [RFC4601] on the incoming interface.

To solve this problem, and keep the solution as simple 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 wants to receive multicast data of (S,G) along the RPT. In this way, we do insure that downstream AFBRs won't miss any multicast data that they needs, at the cost of duplicated multicast data of (S,G) along the RPT received by SPT-switched-over downstream AFBRs, if there exists at least one

downstream AFBR that hasn't 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:

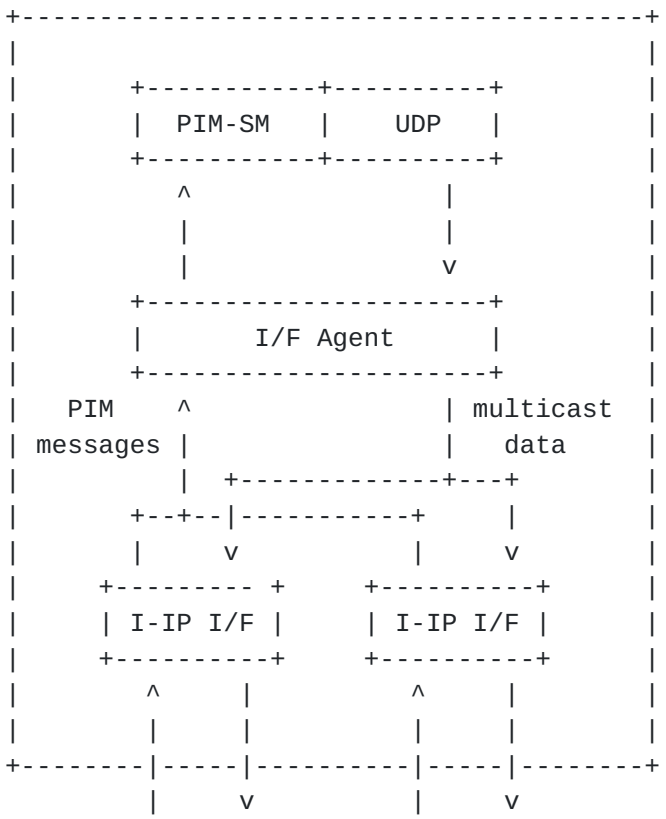

Figure 7: Interface Agent Implementation Example

In this example, the interface agent has two responsibilities: In the control plane, it should work as a real interface that has joined (*,G) in representative of all the I-IP interfaces who should have been outgoing interfaces of (*,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 wants 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 changed it mind to receive multicast data of (S,G) along the RPT again, the interface agent should send a Join(S,G,rpt) to 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 onto every I-IP interface.

NOTICE: There may exist 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 still take effect.

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 spit of whether they have switched over to SPT of some source(s) or not.

To minimize this redundancy, it's recommended that every AFBR's SwitchToSptDesired(S,G) function employs the "switch on first packet" policy. In this way, the delay of switchover to SPT is kept as little 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, there exists other message types 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 dierctly linked routers to negotiate with each other. It's not necessary to translate them for forwarding, thus the process of these messages is out of scope for this document.

6.8. Other PIM States Maintenance

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

7. Data Plane Functions of AFBR

7.1. Process and Forward Multicast Data

On receiving multicast data from upstream routers, the AFBR looks up its forwarding table to check the IP address of each outgoing interface. If there exists at least one outgoing interface whose IP address family is different from the incoming interface, the AFBR should encapsulate/decapsulate this packet and forward it to such

outgoing interface(s), then forward the data to other outgoing interfaces without encapsulation/decapsulation.

When a downstream AFBR that has already switched over to 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 at AFBRs. It's recommended that all AFBRs use the same technology, otherwise some AFBRs may 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 is out of scope of this document.

7.4. Fragmentation

The encapsulation performed by upstream AFBR will increase the size of packets. As a result, the outgoing I-IP link MTU may not accommodate the extra size. As it's not always possible for core operators to increase the MTU of every link. Fragmentation and reassembling of encapsulated packets MUST be supported by AFBRs.

8. Security Considerations

The AFBR routers could maintain secure communications within Security Architecture for the Internet Protocol as described in [RFC4301]. To protect against unwanted forged PIM protocol messages, the PIM messages can be authenticated using IPsec as described in [RFC4601].

But when adopting some schemes that will cause heavy burden on routers, some attacker may use it as a tool for DDoS attack. Compared with [RFC4301], the security concerns should be more carefully considered. The attackers can set up many multicast trees in the edge networks, causing too many multicast trees to get set up in the core network.

9. IANA Considerations

When AFBRs perform address mapping, they should follow some predefined rules, especially the IPv6 prefix for source address mapping should be predefined, such that ingress AFBRs and egress AFBRs can finish the mapping procedure correctly. The IPv6 prefix for translation can be unified within only the transit core, or

within global area. In the later condition, the prefix should be assigned by IANA.

10. References

10.1. Normative References

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC4925] Li, X., Dawkins, S., Ward, D., and A. Durand, "Softwire Problem Statement", RFC 4925, July 2007.

[RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", RFC 5565, June 2009.

[RFC6052] Bao, C., Huitema, C., Bagnulo, M., Boucadair, M., and X. Li, "IPv6 Addressing of IPv4/IPv6 Translators", RFC 6052, October 2010.

[RFC6513] Rosen, E. and R. Aggarwal, "Multicast in MPLS/BGP IP VPNs", RFC 6513, February 2012.

10.2. Informative References

[I-D.ietf-mboned-64-multicast-address-format] Boucadair, M., Qin, J., Lee, Y., Venaas, S., Li, X., and M. Xu, "IPv6 Multicast Address With Embedded IPv4 Multicast Address", draft-ietf-mboned-64-multicast-address-format-05 (work in progress), April 2013.

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
Department of Computer Science, Tsinghua University
Beijing 100084
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