Network Working Group
Internet-Draft
Intended status: Experimental
Expires: August 25, 2021

H. Chen
M. McBride
Futurewei
Y. Fan
Casa Systems
M. Toy
Verizon
A. Wang
China Telecom
L. Liu
Fujitsu
X. Liu
Volta Networks

February 21, 2021

Stateless SRv6 Point-to-Multipoint Path
draft-chen-pim-srv6-p2mp-path-02

Abstract

This document describes a solution for a SRv6 Point-to-Multipoint (P2MP) Path/Tree to deliver the traffic from the ingress of the path to the multiple egresses/leaves of the path in a SR domain. There is no state stored in the core of the network for a SR P2MP path like a SR Point-to-Point (P2P) path in this solution.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 https://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 August 25, 2021.

Table of Contents

1. Introduction

The Segment Routing (SR) for unicast or Point-to-Point (P2P) path is described in [RFC8402]. For SR multicast or Point-to-Multipoint (P2MP) path/tree, it may be implemented through using multiple SR P2P paths. The function of a SR P2MP path/tree from an ingress node to multiple (say n) egress/leaf nodes is implemented by n SR P2P paths. These n P2P paths are from the ingress to those n egress/leaf nodes

of the P2MP path/tree. This solution may waste some network resources such as link bandwidth.

An alternative solution proposed in [I-D.shen-spring-p2mp-transport-chain] uses a number of P2MP chain tunnels to implement a P2MP path/tree from an ingress to n egress/ leaf nodes. Each P2MP chain tunnel is a tunnel from the ingress to a leaf node as its tail end and may have some leaf nodes as its bud nodes along the tunnel. This alternative solution improves the usage of network resources over the solution above using pure P2P paths. However, these two solutions are based on SR P2P paths.

A solution for a SR P2MP path/tree using a P2MP multicast tree is proposed in [I-D.ietf-pim-sr-p2mp-policy]. For a SR P2MP path/tree from an ingress/root to multiple egress/leaf nodes, a multicast P2MP tree is created to deliver the traffic from the ingress/root to the egress/leaf nodes. The state of the tree is instantiated in the forwarding plane by a controller such as PCE at Root node, intermediate Replication nodes and Leaf nodes of the tree. This is not consistent with the SR principles in which no state is stored at the core of the network.

This document describes a new solution for a SRv6 Point-to-Multipoint (P2MP) Path/Tree to deliver the traffic from the ingress of the path to the multiple egresses/leaves of the path in a SR domain. This solution uses a P2MP multicast tree without storing its state in the core of the network for a SR P2MP path/tree like a SR P2P path. For distinguishing a SRv6 P2MP path/tree used in the other solutions with storing some states in the core, a new name, called stateless SRv6 P2MP path/tree, is used in the solution in this document. Even though SRv6 P2MP path/tree and stateless SRv6 P2MP path/tree are used interchangably in the document, they both mean stateless SRv6 P2MP path/tree.

2. Overview of P2MP Multicast Tree

For a SR P2P path from its ingress to its egress, a segment list for the path is provided to the ingress. The ingress pushes the list into a packet, and the packet is delivered to the egress according to the segment list without any state in the core of the network.

For a SR P2MP path from its ingress to multiple egress/leaf nodes, a segment list for the P2MP path is provided to the ingress. The ingress pushes the list into a packet, and the packet is delivered to the multiple egress/leaf nodes according to the segment list without any state in the core of the network.

Figure 1 shows a SR P2MP path from ingress/root R to four egress/leaf nodes L1, L2, L3 and L4. Nodes P1, P2, P3 and P4 are the transit nodes of the P2MP path.

Suppose that X-m is the segment identifier (SID) of node X. X-m is an adjacent SID or node SID. For simplicity, we assume X-m is a node SID in the illustrations below. R-m, P1-m, P2-m, P3-m, P4-m, L1-m, L2-m, L3-m and L4-m are the SIDs of the nodes on the SR P2MP path. They are multicast SIDs or replication SIDs in general.

A multicast SID is a SID from a multicast SID block. In a SR domain supporting SR multicast, each node has a multicast node SID, which is globally significant; each adjacency of a node has a multicast adjacency SID, which is locally significant. A multicast SID of a node on a SR P2MP path is associated with the SIDs of the next hop (or say downstream) nodes. When the node receives a packet with its multicast SID, it duplicates and sends the packet to each of the next hop nodes according to their SIDs.

If node P on a SR P2MP path has B (B > 1) next hop nodes along the path, the SID of node P, P-m, MUST be a multicast SID when it is in the segment list for the P2MP path. The SIDs of the B next hop nodes just follow P-m in the segment list. When node P receives the packet with P-m, it duplicates and sends the packet to each of the B next hop nodes along the P2MP path.

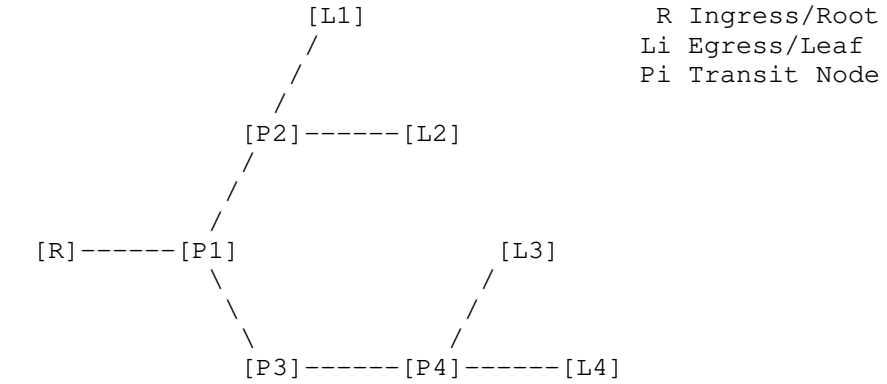


Figure 1: SR P2MP Path from R to L1, L2, L3 and L4

<P1-m, P2-m, P3-m, L1-m, L2-m, P4-m, L3-m, L4-m> is a segment list for the SR P2MP path in Figure 1 to be pushed into a packet at ingress/root R. Node P1 has 2 next hop nodes P2 and P3 along the P2MP path. The next hop nodes' SIDs P2-m and P3-m follow P1-m, which is P1's multicast SID. When P1 receives a packet transported by the

P2MP path, it duplicates and sends the packet to the next hop nodes P2 and P3 according to P1-m, P2-m and P3-m.

The number of branches or next hops from node P1 is a value of one argument in P1-m, called N-Branches. The value of N-Branches in P1-m is 2. With this information, node P1 duplicates and sends the packet to 2 next hop nodes P2 and P3, which are indicated by the 2 SIDs P2-m and P3-m following P1-m.

The number of SIDs of the nodes under node P1 is a value of another argument in P1-m, called N-SIDs. The value of N-SIDs in P1-m is 7, indicating that there are 7 SIDs following P1-m in the segment list.

There are 2 branches or next hops (i.e., L1 and L2) from node P2 and 2 SIDs (i.e., L1-m and L2-m) of the nodes under node P2. The values of N-Branches and N-SIDs in P2-m are 2 and 2. with this information, before sending the packet to node P2, node P1 pushes the SIDs under node P2 into the packet (i.e., the packet has a new segment list with the SIDs under node P2. The new segment list replaces the old one in the packet).

There are 1 branch or next hop (i.e., P4) from node P3 and 3 SIDs (i.e., P4-m, L3-m and L4-m) of the nodes under node P3. The values of N-Branches and N-SIDs in P3-m are 1 and 3 respectively. with this information, before sending the packet to node P3, node P1 pushes the SIDs under node P3 into the packet.

Each node on the SR P2MP path sends the packet to its next hop nodes according to the segment list and no state is stored in any transit node (i.e., the core of the network). The packet is delivered to the egress/leaf nodes from the ingress.

3. Encoding P2MP Multicast Tree

For each sub-tree STi of a SR P2MP path from the ingress node of the P2MP path, suppose that

- the multicast SID of the next hop node NHi is mSIDi;
- there are Bi branches (i.e., outgoing interfaces) to the next hop node BNHj (j = 1, ..., Bi) from node NHi along the sub-tree, the multicast SID of BNHj is mSIDij;
- the number of branches (i.e., outgoing interfaces) under the node BNHj (j = 1, ..., Bi) is BBj; and the number of SIDs of the nodes under each of the Bi branches from node BNHj is NSj (j = 1, ..., Bi).

Sub-tree STi is encoded as segment list

```
      < mSIDi,  mSIDi1, ..., mSIDiBi,   SegSeq1, ...,  SegSeqBi  >,
        \___/  \_____________________/   \______/      \________/
SIDs of  NHi    Bi branches/next-hops   sub-tree       sub-tree
                BNHj of node NHi        from BNH1      from BNHBi
```

where mSIDi contains the number of branches, Bi, in its N-Branches field, and the number of SIDs under mSIDi in its N-SIDs field; mSIDij (j = 1, ..., Bi) contains the number of branches, BBj, in its N-Branches field and the number of SIDs, NSj, in its N-SIDs field; SegSeqj (j = 1, ..., Bi) is the SID sequence in the segment list encoding the sub-trees from node BNHj.

For the P2MP path in Figure 1 from ingress node R to egress nodes L1, L2, L3 and L4, there is one sub-tree from R.

For this sub-tree,

- the next hop node is P1 and the multicast SID of P1 is P1-m;
- there are 2 branches to the next hop nodes P2 and P3 from node P1 along the sub-tree; the number of SIDs of the nodes under P1 is 7; the multicast SIDs of P2 and P3 are P2-m and P3-m respectively;
- the numbers of SIDs of the nodes under these two branches are 2 and 3 respectively. The SIDs of the nodes under P2 are L1-m and L2-m. The SIDs of the nodes under P3 are P4-m, L3-m and L4-m.

The sub-tree is encoded as segment list

```
      < P1-m,  P2-m, P3-m,          L1-m, L2-m,  P4-m, L3-m, L4-m >,
        \__/  \___________/         \________/   \_______________/
SIDs of  P1   2 branches/next-hops   sub-tree     sub-tree
              P2 and P3 of node P1   from P2      from P3
```

where

P1-m's N-Branches field is set to 2 and its N-SIDs field to 7;
P2-m's N-Branches field is set to 2 and its N-SIDs field to 2;
P3-m's N-Branches field is set to 1 and its N-SIDs field to 3;

L1-m and L2-m are the SID sequence in the segment list encoding the sub-trees from P2;

P4-m, L3-m and L4-m are the SID sequence in the segment list encoding the sub-trees from P3; and

P4-m's N-Branches field is set to 2 and its N-SIDs field to 2.

Figure 2 shows in details the segment list, which is an encoding of the P2MP multicast tree for the SR P2MP path from R to L1, L2, L3 and L4.

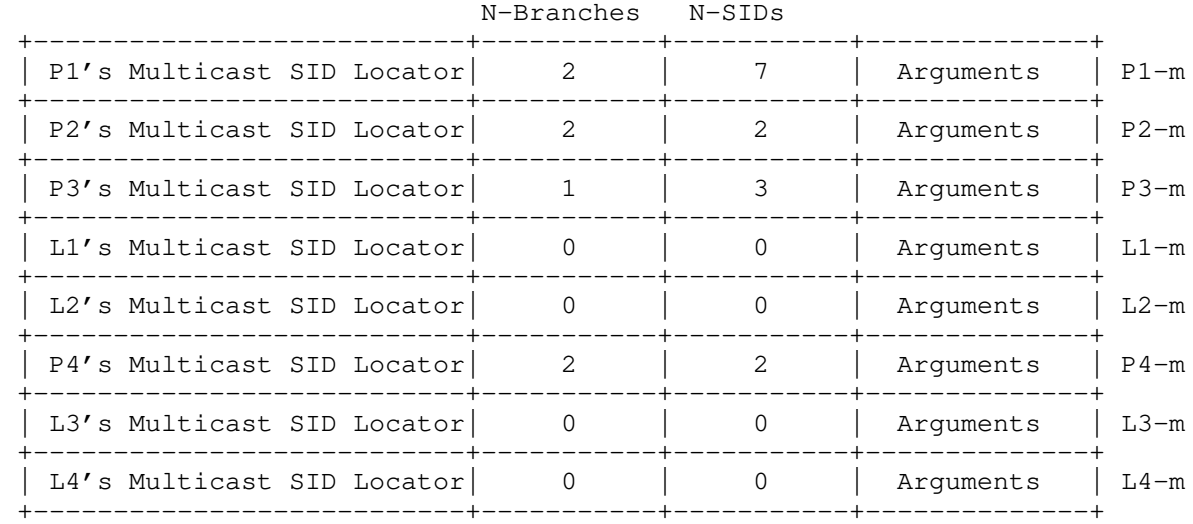


Figure 2: Encoding of P2MP Multicast Tree from R to L1 - L4

SID P1-m indicates that there are 2 branches and 7 SIDs under P1. SID P2-m indicates that there are 2 branches and 2 SIDs under P2. SID P3-m indicates that there are 1 branch and 3 SIDs under P3. SIDs L1-m and L2-m indicate that there is no branch under them. SID P4-m indicates that there are 2 branches and 2 SIDs under P4. L3-m and L4-m indicate that there is no branch under them.

4. Procedures/Behaviors

This section describes the procedures or behaviors on the ingress, transit and egress/leaf node of a SR P2MP path to deliver a packet received from the path to its destinations.

4.1. Procedure/Behavior on Ingress Node

For a packet to be transported by a SR P2MP Path, the ingress of the P2MP path duplicates the packet for each sub-tree of the SR P2MP path branching from the ingress, pushes the segment list encoding the sub-tree into the packet by executing H.Encaps [I-D.ietf-spring-srv6-network-programming] and sends the packet to the next hop node along the sub-tree.

Regarding to the finite size of the segment list, a sub-tree can be "split" into multiple sub-trees such that each of the sub-trees can be encoded in the segment list of the finite size.

For example, there is one sub-tree from the ingress R of the SR P2MP path in Figure 1 via next hop node P1 towards egress/leaf nodes L1, L2, L3 and L4.

For this sub-tree, the ingress R duplicates the packet, set the destination address (DA) to P1-m (i.e., multicast SID of node P1), pushes the segment list without P1-m (i.e., <P2-m, P3-m, L1-m, L2-m, P4-m, L3-m, L4-m>) encoding the sub-tree in a Segment Routing Header (SRH) of the packet by executing H.Encaps and sends the packet to DA (i.e., node P1). The contents of the multicast SIDs P1-m, P2-m, P3-m, L1-m, L2-m, P4-m, L3-m, L4-m are shown in Figure 2.

Suppose that the duplicated packet is Pkt0 for the sub-tree. The execution of H.Encaps pushes an IPv6 header (i.e., SRH) to Pkt0 and sets some fields in the header to produce an encapsulated packet Pkt'. Pkt' is represented in the following:

```
Pkt' = (SA=R, DA=P1-m)( L4-m, L3-m,..., P3-m,P2-m; SL=7)Pkt0
                       \_____________________________/
        corresponds to: <P2-m,P3-m, ..., L3-m,L4-m>
```

where DA=P1-m means that the destination address (DA) is set to P1-m; SA=R means that the source address (SA) is set to R; SL=7 means that the number of Segments Left (SL) is 7.

4.2. Procedure/Behavior on Transit Node

When a transit node of a SR P2MP path receives a packet transported by the P2MP path, the DA of the packet is a multicast SID of the node and the packet contains a segment list for the sub-trees under the transit node. The DA and the segment list comprise the information for encoding the sub-trees.

For example, when node P1 receives a packet transported by the SR P2MP path in Figure 1, the packet's DA is P1-m (which is a multicast SID of node P1) and the segment list in the packet is <P2-m, P3-m, L1-m, L2-m, P4-m, L3-m, L4-m>.

The N-Branches field (which has value of n) of the DA indicates that there are n branches (or say sub-trees) under the transit node. The N-SIDs field of the DA indicates the number of SIDs for these n sub-trees under the transit node. The multicast SIDs of the next hop nodes of these n sub-trees are the first n multicast SIDs of the segment list in the packet.

For example, the N-Branches field (which has value of 2) of DA = P1-m indicates that there are 2 branches (or say sub-trees) under node P1. The N-SIDs field (which has value of 7) of the DA = P1-m indicates that there are 7 SIDs for these 2 sub-trees under node P1.

The first multicast SID (P2-m) of the segment list is the SID of the next hop node (P2) of the first sub-tree; The second multicast SID (P3-m) of the segment list is the SID of the next hop node (P3) of the second sub-tree.

After the multicast SIDs of the next hop nodes, there are n blocks of SIDs for those n sub-trees. The N-SIDs field (which has value of B1) of the first multicast SID of the next hop nodes indicates that there are B1 SIDs in the first block for the first sub-tree; the N-SIDs field (which has value of B2) of the second multicast SID of the next hop nodes indicates that there are B2 SIDs in the second block for the second sub-tree after the first block; and so on.

For example, there are 2 blocks of SIDs for the 2 sub-trees under node P1 after the multicast SIDs P2-m and P3-m of the next hop nodes P2 and P3. The N-SIDs field of P2-m (the first multicast SID of the next hop nodes) has value of 2, indicating that there are 2 SIDs in the first block for the first sub-tree, which are L1-m and L2-m.

The N-SIDs field of P3-m (the second multicast SID of the next hop nodes) has value of 3, indicating that there are 3 SIDs in the second block for the second sub-tree after the first block, which are P4-m, L3-m and L4-m.

The transit node duplicates the packet without top header for each sub-tree under it and adds a new header with a new segment list built from the SID block for the sub-tree to the duplicated packet by executing H.Encaps. It sets the DA of the packet to the multicast SID of the next hop node along the sub-tree and sends the packet to the DA.

For example, node P1 duplicates the packet for the first sub-tree towards L1 and L2 and adds a new header with a new segment list <L1-m, L2-m>. It sets DA = P2-m (multicast SID of next hop P2), and sends the packet to the DA (i.e., P2).

Suppose that the duplicated packet is Pkt0 for the sub-tree. The execution of H.Encaps pushes a new IPv6 header (i.e., SRH) to Pkt0 and sets some fields in the header to produce an encapsulated packet Pkt'. Pkt' is represented in the following:

```
Pkt' = (SA=P1, DA=P2-m)( L2-m, L1-m;  SL=2)Pkt0.
                         \___________/
       corresponds to:   <L1-m, L2-m>
```

where DA=P2-m means that the destination address (DA) is set to P2-m; SA=P1 means that the source address (SA) is set to P1; SL=2 means that the number of Segments Left (SL) is 2.

Node P1 duplicates the packet for the second sub-tree via P3 towards L3 and L4 and adds a new header with a new segment list <P4-m, L3-m, L4-m>. It sets DA = P3-m (multicast SID of next hop P3), and sends the packet to the DA (i.e., P3).

4.3. Procedure/Behavior on Egress Node

When an egress node of a SR P2MP path receives a packet transported by the P2MP path, the DA of the packet is a SID of the egress node. The egress node sends the packet to its destination accordingly. If the SID is a multicast SID of the egress, the N-Branches field and N-SIDs field are all zeros.

5. Stateless SRv6 P2MP Path for Ingress

A controller such as PCE can compute a stateless SRv6 P2MP path and send it to its ingress. For a packet to be transported by the path, the ingress encapsulates the packet with the path and the packet will be delivered to the egresses of the path without any states in the network core.

An example architecture using PCE as a controller is illustrated in Figure 3. There is a connection (i.e., PCE session) between the PCE and (the PCC running on) each of the PEs, which are possible ingress nodes in the network domain. Note that some of connections between the PCE and PEs are not shown in the figure.

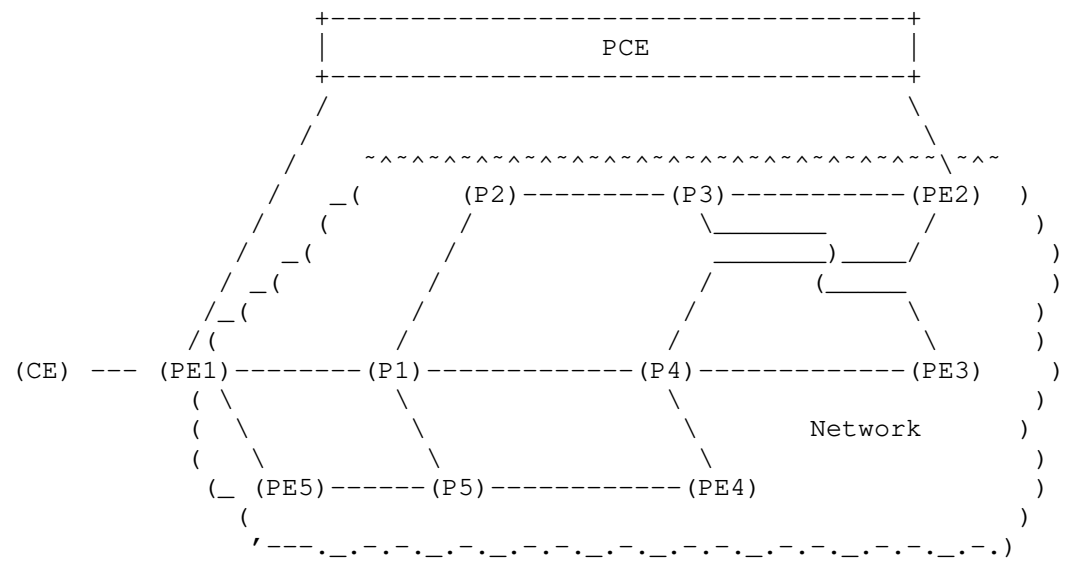


Figure 3: Architecture using PCE

The PCE has the information about the network domain from the IGP or BGP (BGP-LS). The information includes link bandwidth, link colors, node SIDs, and so on. A separate multicast SID could be provisioned on every replication node and the PCE gets the SID on the node from IGP or BGP.

The PCE maintains the current status of the network resource usage in its local TED (Traffic Engineering Database), and the status of every stateless SRv6 P2MP path in its local LSP-DB (Label Switch Path Database).

Upon receiving a request for a stateless SRv6 P2MP path from a user or application, the PCE computes a path based on the network resource availability stored in the TED. After a path satisfying the given constraints is found, the PCE constructs a stateless SRv6 P2MP path using the multicast SIDs of the nodes on the path and encodes the structure of the P2MP path/tree into the parameters of the SIDs. In fact, the stateless SRv6 P2MP path is a segment list consisting of multicast SIDs with parameter values.

And then the PCE sends the segment list representing the path to the ingress node of the path in a PCEP message such as PCInitiate. After receiving the path from the PCE, the ingress node establishes the path by creating a forwarding entry in its FIB. For every multicast packet to be transported by the path, the forwarding entry encapsulates the packet with the segment list and the packet will be

delivered to the egress nodes of the path along the path without any state in the core of the network.

6. Protection

Protections for a SR P2MP path can be classified into two types: global protection and local protection.

6.1. Global Protection

For a primary SR P2MP path from an ingress node R1 to multiple egress nodes Li (i = 1, ..., n), a backup SR P2MP path from an ingress node R1' to multiple egress nodes Li' (i = 1, ..., n) is set up to provide global protection for the primary SR P2MP path. If R1' is the same as R1, the failure of the ingress node R1 of the primary SR P2MP path is not protected; otherwise (i.e., R1' and R1 are different and connected to the same traffic source), the failure of the ingress node R1 is protected. If Li' is the same as Li (i = 1, ..., n), the failure of the egress nodes Li (i = 1, ..., n) of the primary SR P2MP path is not protected; otherwise (i.e., Li' and Li are different and connected to the same destination), the failure of the egress nodes Li is protected.

When a failure happens on the primary SR P2MP path and is detected by the source of the traffic or other entity, the traffic to be transported by the primary SR P2MP path is switched to the backup SR P2MP path, which sends the traffic from its ingress node R1' to its egress nodes Li' (i = 1, ..., n).

6.2. Local Protection

Local protection or say Fast Reroute (FRR) of a node and adjacency segment on a SR P2P path is proposed in [I-D.ietf-rtgwg-segment-routing-ti-lfa] and [I-D.ietf-rtgwg-srv6-egress-protection]. It can be applied to FRR of a node and adjacency segment on a SR P2MP path in a similar way. But FRR for SR P2MP path is more complicated.

More details will be added later.

7. IANA Considerations

TBD

8. Security Considerations

TBD

9. Acknowledgements

The authors would like to thank Acee Lindem, Jeffrey Zhang and Daniel Voyer for their valuable comments and suggestions on this draft.

10. References

10.1. Normative References

[I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-26 (work in progress), October 2019.

[I-D.ietf-rtgwg-segment-routing-ti-lfa]
Litkowski, S., Bashandy, A., Filsfils, C., Decraene, B., and D. Voyer, "Topology Independent Fast Reroute using Segment Routing", draft-ietf-rtgwg-segment-routing-ti-lfa-05 (work in progress), November 2020.

[I-D.ietf-rtgwg-srv6-egress-protection]
Hu, Z., Chen, H., Chen, H., Wu, P., Toy, M., Cao, C., He, T., Liu, L., and X. Liu, "SRv6 Path Egress Protection", draft-ietf-rtgwg-srv6-egress-protection-02 (work in progress), November 2020.

[I-D.ietf-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-ietf-spring-srv6-network-programming-28 (work in progress), December 2020.

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

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

10.2. Informative References

[I-D.ietf-pim-sr-p2mp-policy]
Voyer, D., Filsfils, C., Parekh, R., Bidgoli, H., and Z. Zhang, "Segment Routing Point-to-Multipoint Policy", draft-ietf-pim-sr-p2mp-policy-01 (work in progress), October 2020.

[I-D.ietf-spring-sr-replication-segment]
Voyer, D., Filsfils, C., Parekh, R., Bidgoli, H., and Z. Zhang, "SR Replication Segment for Multi-point Service Delivery", draft-ietf-spring-sr-replication-segment-02 (work in progress), October 2020.

[I-D.shen-spring-p2mp-transport-chain]
Shen, Y., Zhang, Z., Parekh, R., Bidgoli, H., and Y. Kamite, "Point-to-Multipoint Transport Using Chain Replication in Segment Routing", draft-shen-spring-p2mp-transport-chain-03 (work in progress), October 2020.

Appendix A. Example IPv6 Header using G-SRv6

For simplicity, 64 bits for Common Prefix, 16 bits for Node ID, 8 bits for the number of branches (N-Branches) and 8 bits for the number of SIDs (N-SIDs) are used when G-SRv6 compression method is applied for <P1-m, P2-m, P3-m, L1-m, L2-m, P4-m, L3-m, L4-m> at ingress node R in Figure 1. The Destination Address (DA) is illustrated below in Figure 4. It contains the Common Prefix of 64 bits, node P1's ID of 16 bits, the value 2 for the number of branches (N-Branches) of 8 bits, and the value 7 for the number of SIDs (N-SIDs) of 8 bits.

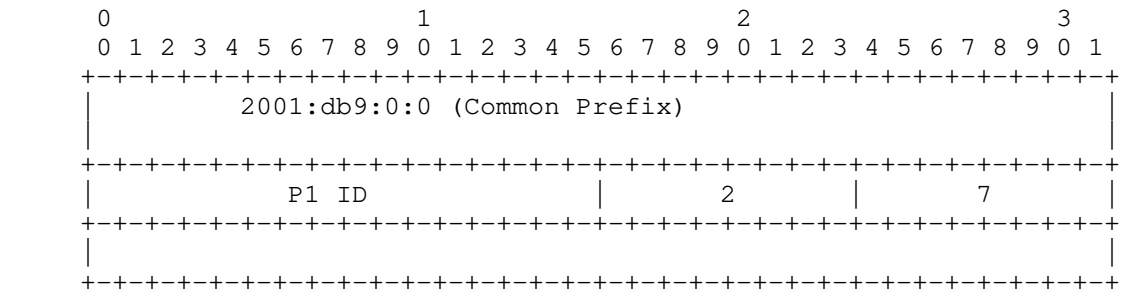


Figure 4: Destination Address (DA)

The IPv6 header is shown in Figure 5. Ingress node R sends a packet with the IPv6 header to the DA.

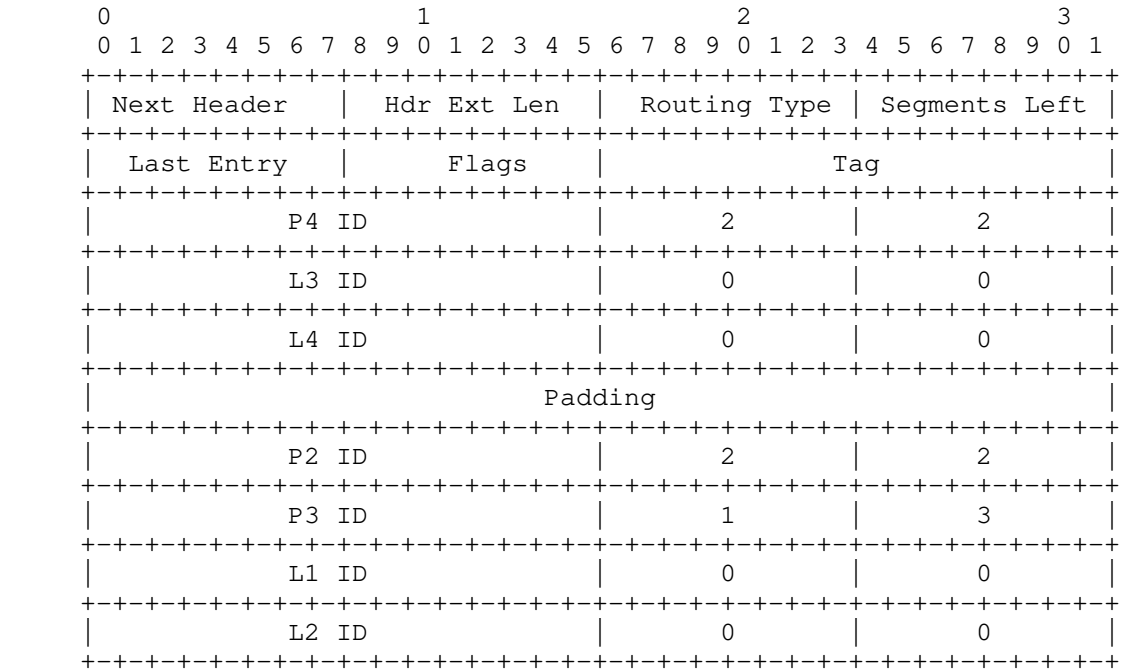


Figure 5: IPv6 Header

Authors' Addresses

Huaimo Chen
Futurewei
Boston, MA
USA

Email: Huaimo.chen@futurewei.com

Mike McBride
Futurewei

Email: michael.mcbride@futurewei.com

Yanhe Fan
Casa Systems
USA

Email: yfan@casa-systems.com

Mehmet Toy
Verizon
USA

Email: mehmet.toy@verizon.com

Aijun Wang
China Telecom
Beiqijia Town, Changping District
Beijing, 102209
China

Email: wangaj3@chinatelecom.cn

Lei Liu
Fujitsu

USA

Email: liulei.kddi@gmail.com

Xufeng Liu
Volta Networks

McLean, VA
USA

Email: xufeng.liu.ietf@gmail.com

PIM Working Group
Internet Draft
Intended status: Standards Track
Expires: August 21, 2021

Yisong Liu
China Mobile
M. McBride
T. Eckert
Futurewei
Z. Zhang
ZTE

Feb 21, 2021

PIM Assert Message Packing
draft-ietf-pim-assert-packing-01

Status of this Memo

Copyright Notice

Abstract

In PIM-SM shared LAN networks, there is typically more than one upstream router. When duplicate data packets appear on the LAN from different routers, assert packets are sent from these routers to elect a single forwarder. The PIM assert packets are sent periodically to keep the assert state. The PIM assert packet carries information about a single multicast source and group, along with the metric-preference and metric of the route towards the source or RP. This document defines a standard to send and receive multiple multicast source and group information in a single PIM assert packet in a shared network. This can be particularly helpful when there is traffic for a large number of multicast groups.

Table of Contents

1. Introduction

In PIM-SM shared LAN networks, there is typically more than one upstream router. When duplicate data packets appear on the LAN, from different upstream routers, assert packets are sent from these routers to elect a single forwarder according to [RFC7761]. The PIM assert packets are sent periodically to keep the assert state. The PIM assert packet carries information about a single multicast source and group, along with the corresponding metric-preference and metric of the route towards the source or RP.

This document defines a standard to send and receive multiple multicast source and group information in a single PIM assert packet in a shared LAN network. It can efficiently pack multiple PIM assert packets into a single message and reduce the processing pressure of the PIM routers. This can be particularly helpful when there is traffic for a large number of multicast groups.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

1.2. Terminology

RPF: Reverse Path Forwarding

RP: Rendezvous Point

SPT: Shortest Path Tree

RPT: RP Tree

DR: Designated Router

BDR: Backup Designated Router

2. Use Cases

PIM Assert will happen in many services where multicast is used and not limited to the examples described below.

2.1. Enterprise network

When an Enterprise network is connected through a layer-2 network, the intra-enterprise runs layer-3 PIM multicast. The different sites of the enterprise are equivalent to the PIM connection through the shared LAN network. Depending upon the locations and amount of groups there could be many asserts on the first hop routers.

2.2. Video surveillance

Video surveillance deployments have migrated from analog based systems to IP-based systems oftentimes using multicast. In the shared LAN network deployments, when there are many cameras streaming to many groups there may be issues with many asserts on first hop routers.

2.3. Financial Services

Financial services extensively rely on IP Multicast to deliver stock market data and its derivatives, and current multicast solution PIM is usually deployed. As the number of multicast flows grows, there are many stock data with many groups may result in many PIM asserts on a shared LAN network from publisher to the subscribers.

2.4. IPTV broadcast Video

PIM DR and BDR deployments are often used in host-side network for IPTV broadcast video services. Host-side access network failure scenario may be benefitted by assert packing when many groups are being used. According to [RFC7761] the DR will be elected to forward multicast traffic in the shared access network. When the DR recovers from a failure, the original DR starts to send traffic, and the current DR is still forwarding traffic. In the situation multicast traffic duplication maybe happen in the shared access network and can trigger the assert progress.

2.5. Summary

In the above scenarios, the existence of PIM assert process depends mainly on the network topology. As long as there is a layer 2 network between PIM neighbors, there may be multiple upstream routers, which can cause duplicate multicast traffic to be forwarded and assert process to occur.

Moreover as the multicast services become widely deployed, the number of multicast entries increases, and a large number of assert messages may be sent in a very short period when multicast data packets trigger PIM assert process in the shared LAN networks. The

PIM routers need to process a large number of PIM assert small packets in a very short time. As a result, the device load is very large. The assert packet may not be processed in time or even is discarded, thus extending the time of traffic duplication in the network.

Additionally, future backhaul, or fronthaul, networks may want to connect L3 across an L2 underlay supporting Time Sensitive Networks (TSN). The infrastructure may run DetNet over TSN. These transit L2 LANs would have multiple upstreams and downstreams. This document is taking a proactive approach to prevention of possible future assert issues in these types of environments.

3. Solution

The change to the PIM assert includes two elements: the PIM assert packing hello option and the PIM assert packing method.

There is no change required to the PIM assert state machine. Basically a PIM router can now be the assert winner or loser for multiple packed (S, G)’s in a single assert packet instead of one (S, G) assert at a time. An assert winner is now responsible for forwarding traffic from multiple (S, G)’s out of a particular interface based upon the multiple (S, G)’s packed in a single assert.

3.1. PIM Assert Packing Hello Option

The newly defined Hello Option is used by a router to negotiate the assert packet packing capability. It can only be used when all PIM routers, in the same shared LAN network, support this capability. This document defines two packing methods. One method is a simple merge of the original messages and the other is to extract the common message fields for aggregation.

3.2. PIM Assert Packing Simple Type

In this type of packing, the original assert message body is used as a record. The newly defined assert message can carry multiple assert records and identify the number of records.

This packing method is simply extended from the original assert packet, but, because the multicast service deployment often uses a small number of sources and RPs, there may be a large number of assert records with the same metric preference or route metric field, which would waste the payload of the transmitted message.

3.3. PIM Assert Packing Aggregation Type

When the source or RP addresses, in the actual deployment of the multicast service, are very few, this type of packing will combine the records related to the source address or RP address in the assert message.

* A (S, G) assert only can contain one SPT (S, G) entry, so it can be aggregated according to the same source address, and then all SPT (S, G) entries corresponding to the same source address are merged into one assert record.

* A (*, G) assert may contain a (*, G) entry or a RPT (S, G) entry, and both entry types actually depend on the route to the RP. So it can be aggregated further according to the same RP address, and then all (*, G) and RPT (S, G) entries corresponding to the same RP address are merged into one assert record.

This method can optimize the payload of the transmitted message by merging the same field content, but will add the complexity of the packet encapsulation and parsing.

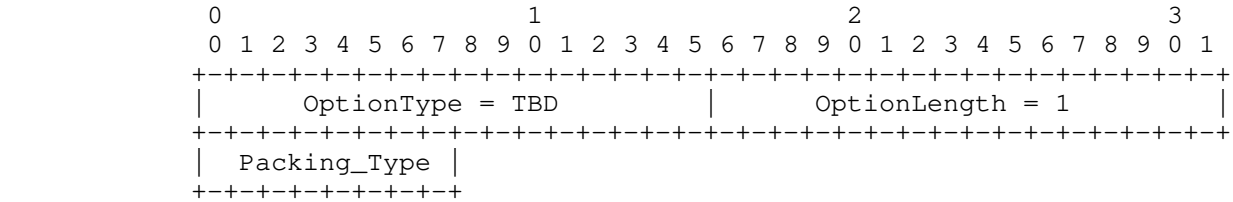
3.4. Assert Timer

This packing message takes no effect on the existed Assert Timer for (*,G) and (S,G). When the winner sends the assert message due to the local periodic timer expiration, the (*,G) and (S,G) which are expired at the same time will be sent by packing message instead of individual message.

4. Packet Format

This section describes the format of new PIM messages introduced by this document. The messages follow the same transmission order as the messages defined in [RFC7761].

4.1. PIM Assert Packing Hello Option



- OptionType: TBD

- OptionLength: 1

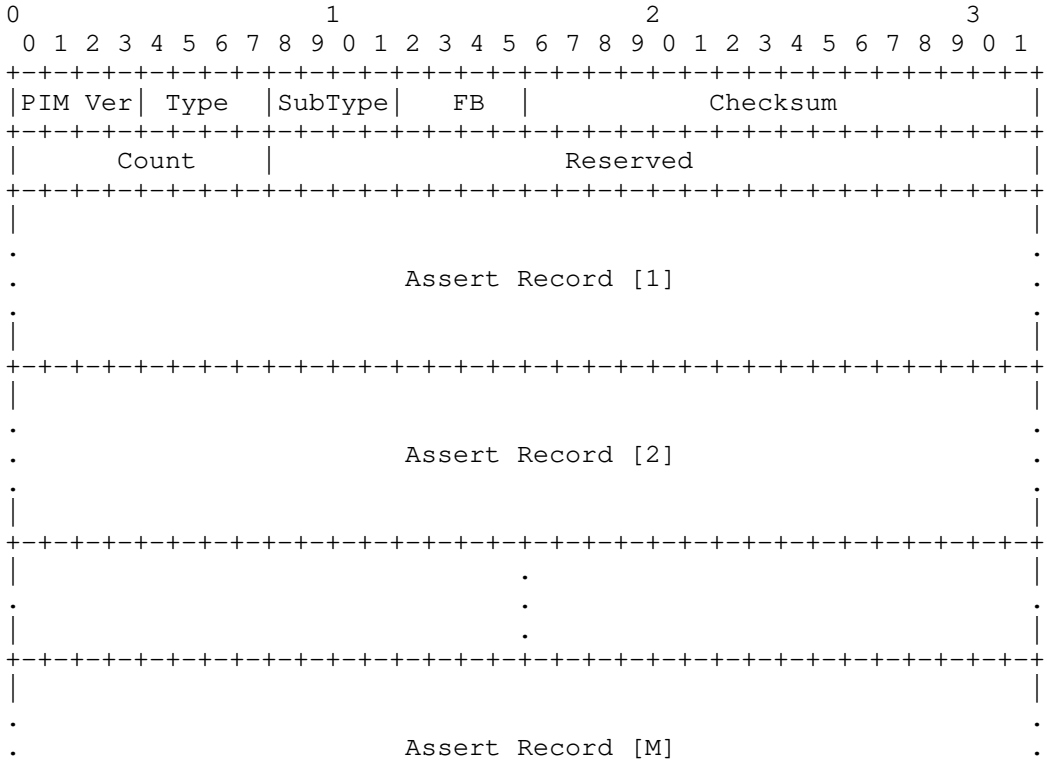
- Packing_Type: The specific packing mode is determined by the value of this field:

  1: indicates simple packing type as described in section 2.2

  2: indicates aggregating packing type as described in section 2.3

  3-255: reserved for future

4.2. PIM Assert Simple Packing Format



PIM Version, Reserved, Checksum

Same as [RFC7761] Section 4.9.6

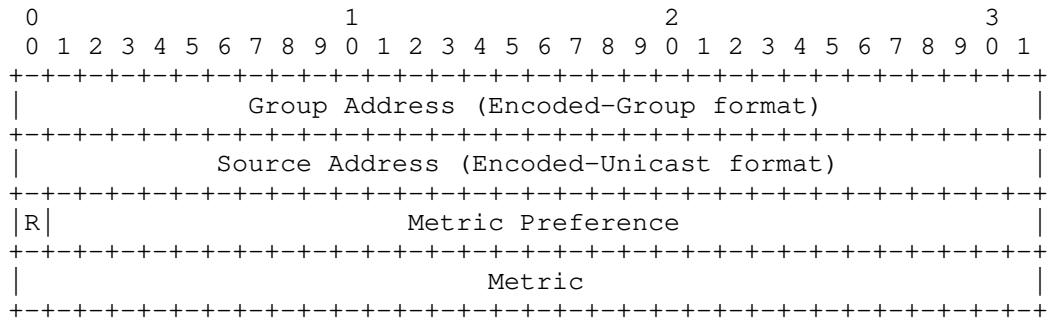
Type

The new Assert Type and SubType values TBD

Count

The number of packed assert records. A record consists of a single assert message body.

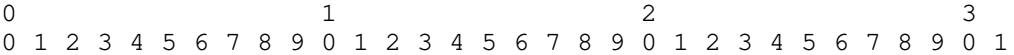
The format of each record is the same as the PIM assert message body of section 4.9.6 in [RFC7761].

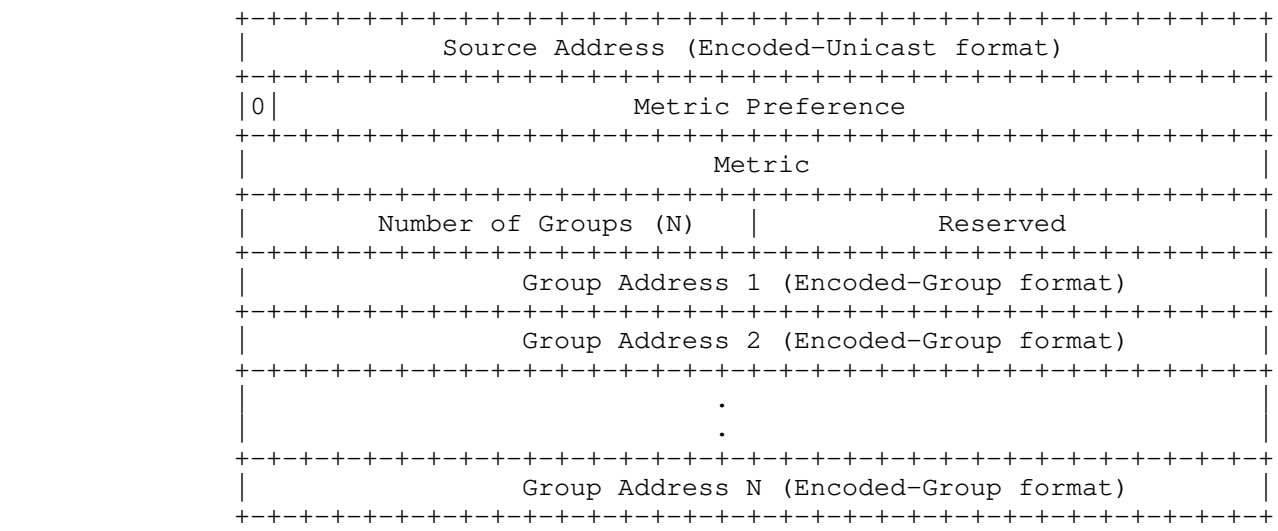


4.3. PIM Assert Aggregation Packing Format

This method also extends PIM assert packets to carry multiple records. The specific assert packet format is the same as section 4.2, but the records are divided into two types.

The (S, G) assert records are organized by the same source address, and the specific message format is:





Source Address, Metric Preference, Metric and Reserved

Same as [RFC7761] Section 4.9.6, but the source address MUST NOT be set to zero.

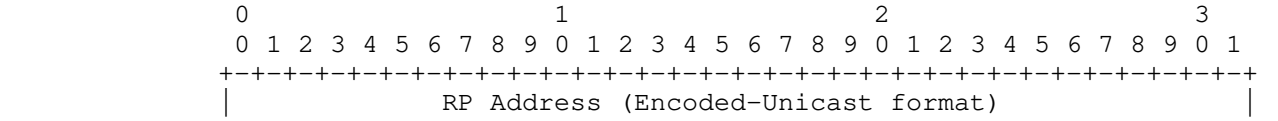
Number of Groups

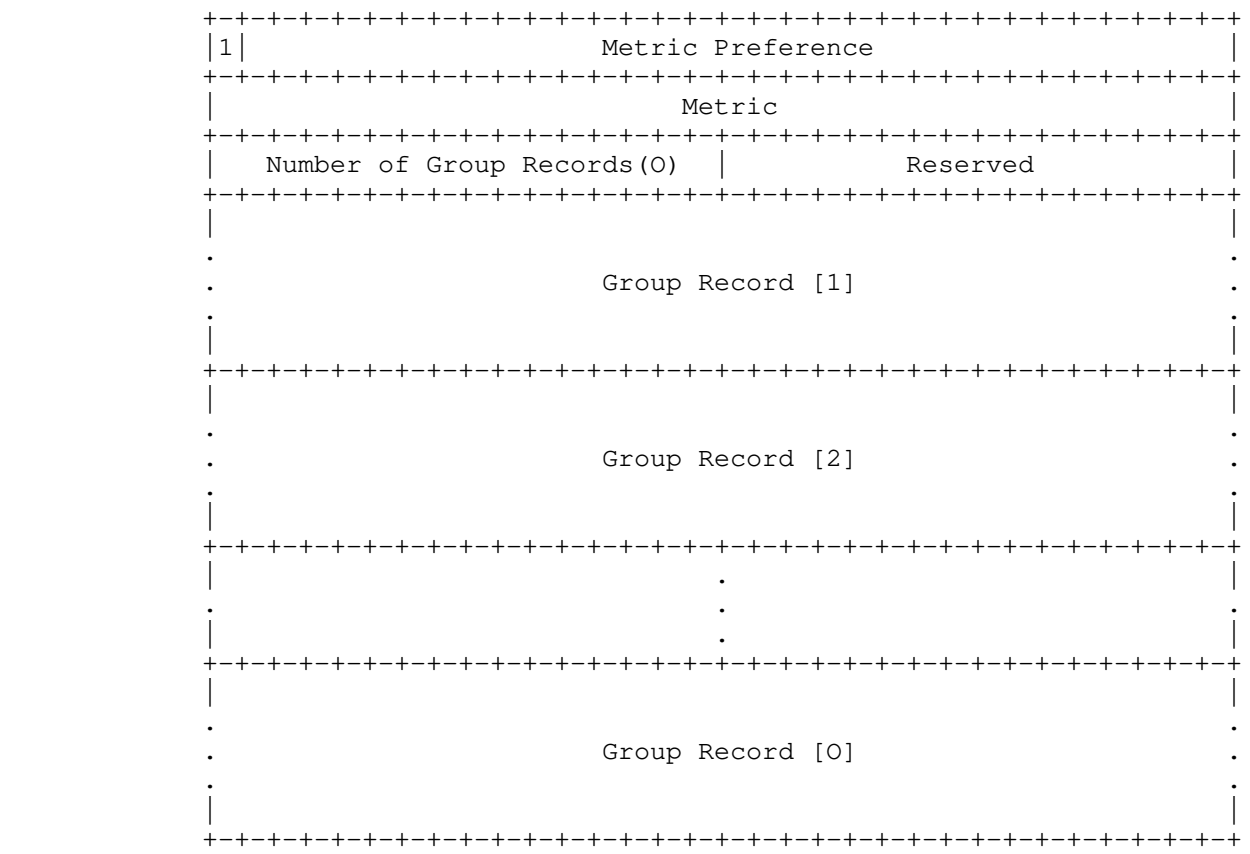
The number of group addresses corresponding to the source address field in the (S, G) assert record.

Group Address

Same as [RFC7761] Section 4.9.6, but there are multiple group addresses in the (S, G) assert record

The (*, G) assert records are organized in the same RP address and are divided into two levels of TLVs. The first level is the group record of the same RP address, and the second level is the source record of the same multicast group address, including (*, G) and RPT (S, G), and the specific message format is:





RP Address

The address of RP corresponding to all of the contained group records. The format for this address is given in the encoded unicast address in [RFC7761] Section 4.9.1

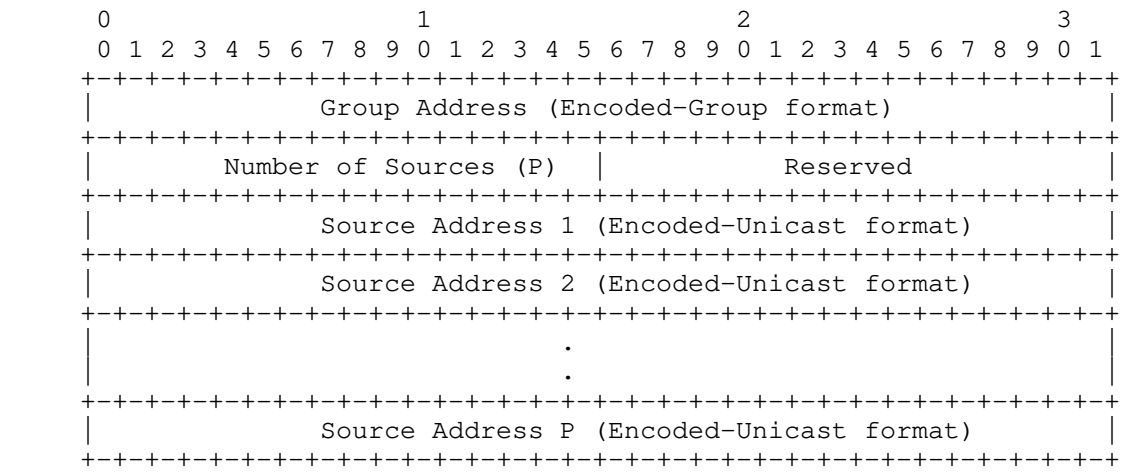
Metric Preference, Metric and Reserved

Same as [RFC7761] Section 4.9.6

Number of Group Records

The number of packed group records. A record consists of a group address and a source address list.

The format of each group record is:



Group Address and Reserved

Same as [RFC7761] Section 4.9.6

Number of Sources

The number of source addresses corresponding to the group address field in the group record.

Source Address

Same as [RFC7761] Section 4.9.6, but there are multiple source addresses in the group record.

5. IANA Considerations

This document requests IANA to assign a registry for PIM assert packing Hello Option in the PIM-Hello Options and new PIM assert packet type and subtype. The assignment is requested permanent for IANA when this document is published as an RFC. The string TBD should be replaced by the assigned values accordingly.

6. Security Considerations

For general PIM-SM protocol Security Considerations, see [RFC7761].

TBD

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC7761] Fenner, B., Handley, M., Holbrook, H., Kouvelas, I.,Parekh, R., Zhang, Z., and L. Zheng, "Protocol IndependentMulticast - Sparse Mode (PIM-SM): Protocol Specification(Revised)", RFC 7761, March 2016

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, May 2017

7.2. Informative References

TBD

8. Acknowledgments

The authors would like to thank the following for their valuable contributions of this document:

TBD

Authors' Addresses

Yisong Liu
China Mobile

Email: liuyisong@chinamobile.com

Mike McBride
Futurewei

Email: michael.mcbride@futurewei.com

Toerless Eckert
Futurewei

Email: tte+ietf@cs.fau.de

Zheng(Sandy) Zhang
ZTE Corporation

Email: zhang.zheng@zte.com.cn

PIM WG
Internet-Draft
Intended status: Standards Track
Expires: August 21, 2021

Z. Zhang
ZTE Corporation
F. Hu
Individual
B. Xu
ZTE Corporation
M. Mishra
Cisco Systems

February 17, 2021

Protocol Independent Multicast - Sparse Mode (PIM-SM) Designated Router (DR) Improvement
draft-ietf-pim-dr-improvement-11

Abstract

Protocol Independent Multicast - Sparse Mode (PIM-SM) is a widely deployed multicast protocol. As deployment for the PIM protocol is growing day by day, a user expects lower packet loss and faster convergence regardless of the cause of the network failure. This document defines an extension to the existing protocol, which improves the PIM's stability with respect to packet loss and convergence time when the PIM Designated Router (DR) role changes.

Table of Contents

1. Introduction

Multicast technology, with PIM-SM ([RFC7761]), is used widely in Modern services. Some events, such as changes in unicast routes, or a change in the PIM-SM DR, may cause the loss of multicast packets.

The PIM DR has two responsibilities in the PIM-SM protocol. For any active sources on a LAN, the PIM DR is responsible for registering with the Rendezvous Point (RP). Also, the PIM DR is responsible for tracking local multicast listeners and forwarding data to these listeners.

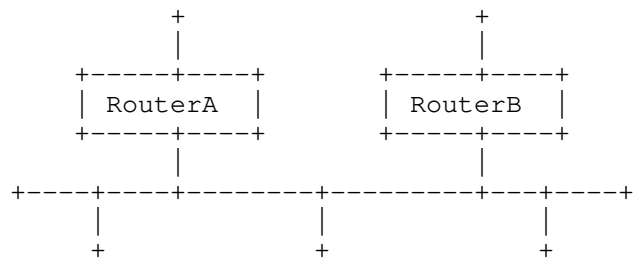


Figure 1: An example of multicast network

The simple network in Figure 1 presents two routers (A and B) connected to a shared-media LAN segment. Two different scenarios are described to illustrate potential issues.

(a) Both routers are on the network, and RouterB is elected as the DR. If RouterB then fails, multicast packets are discarded until RouterA is elected as DR, and it assumes the multicast flows on the LAN. As detailed in [RFC7761], a DR's election is triggered after the current DR's Hello_Holdtime expires. The failure detection and election procedures may take several seconds. That is too long for modern multicast services.

(b) Only RouterA is initially on the network, making it the DR. If RouterB joins the network with a higher DR Priority. Then it will be elected as DR. RouterA will stop forwarding multicast packets, and the flows will not recover until RouterB assumes them.

In either of the situations listed, many multicast packets may be lost, and the quality of the services noticeably affected. To increase the stability of the network this document introduces the Designated DR (DR) and Backup Designated Router (BDR) options, and specifies how the identity of these nodes is explicitly advertised.

1.1. Keywords

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

Modern services: The real time multicast services, such like IPTV, Net-meeting, etc.

Backup Designated Router (BDR): Immediately takes over all DR functions ([RFC7761]) on an interface once the DR is no longer present. A single BDR SHOULD be elected per interface.

Designated Router Other (DROther): A router which is neither a DR nor a BDR.

0x0: 0.0.0.0 if IPv4 addresses are in use or 0:0:0:0:0:0:0:0/128 if IPv6 addresses are in use. To simplify, 0x0 is used in abbreviation in this draft.

Sticky: The DR doesn't change unnecessarily when routers, even with higher priority, go down or come up.

3. Protocol Specification

The router follows the following procedures, these steps are to be used when a router starts, or the interface is enabled:

(a). When a router first starts or its interface is enabled, it includes the DR and BDR Address options with the OptionValue set to 0x0 in its Hello messages (Section 4). At this point the router considers itself a DROther, and starts a timer set to Default_Hello_Holdtime [RFC7761].

(b). When the router receives Hello messages from other routers on the same shared-media LAN, the router checks the value of DR/BDR address option. If the value is filled with a non-zero IP address, the router stores the IP address.

(c). After the timer expires, the router first executes the algorithm defined in section 3.1. After that, the router acts as one of the roles in the LAN: DR, BDR, or DROther.

If the router is elected the BDR, it takes on all the functions of a DR as specified in [RFC7761], but it SHOULD NOT actively forward multicast flows or send a register message to avoid duplication.

If the DR becomes unreachable on the LAN, the BDR MUST take over all the DR functions, including multicast flow forwarding and sending the Register messages. Mechanisms outside the scope of this specification, such as [I-D.ietf-pim-bfd-p2mp-use-case] or BFD Asynchronous mode [RFC5880] can be used for faster failure detection.

For example, there are three routers: A, B, and C. If all three were in the LAN, then their DR preference would be A, B, and C, in that order. Initially, only C is on the LAN, so C is DR. Later, B joins;

C is still the DR, and B is the BDR. Later A joins, then A becomes the BDR, and B is simply DROther.

3.1. Election Algorithm

The DR and BDR election refers the DR election algorithm defined in section 9.4 in [RFC2328], and updates the election function defined in section 4.3.2 in [RFC7761].

- The DR is elected among the DR candidates directly. If there is no DR candidates, i.e., all the routers advertise the DR Address options with zero OptionValue, the elected BDR will be the DR. And then the BDR is elected again from the other routers in the LAN.

- The BDR election is not sticky. Whatever there is a router that advertise the BDR Address option, the router which has the highest priority, expect for the elected DR, is elected as the BDR. That is the BDR may be the router which has the highest priority in the LAN.

- The advertisement is through PIM Hello message.

Except for the information recorded in section 4.3.2 in [RFC7761], the DR/BDR OptionValue from the neighbor is also recorded:

- neighbor.dr: The DR Address OptionValue that presents in the Hello message from the PIM neighbor.

- neighbor.bdr: The BDR Address OptionValue that presents in the Hello message from the PIM neighbor.

The pseudocode is shown below: A BDR election function is added, and the DR function is updated. The validneighbor function means that a valid Hello message has been received from this neighbor.

```
BDR(I) {
    bdr = NULL
    for each neighbor on interface I {
        if ( neighbor.bdr != NULL ) {
            if (validneighbor (neighbor.bdr) == TRUE) {
                if bdr == NULL
                    bdr = neighbor.bdr
                else (dr_is_better( neighbor.bdr, bdr, I ) == TRUE ) {
                    bdr = neighbor.bdr
                }
            }
        }
    }
    return bdr
}

DR(I) {
    dr = NULL
    for each neighbor on interface I {
        if ( neighbor.dr != NULL ) {
            if (validneighbor (neighbor.dr) == TRUE) {
                if (dr == NULL)
                    dr = neighbor.dr
                else (dr_is_better( neighbor.dr, dr, I ) == TRUE ) {
                    dr = neighbor.dr
                }
            }
        }
    }
    if (dr == NULL) {
        dr = bdr
    }
    if (dr == NULL) {
        dr = me
    }
    return dr
}
```

Compare to the DR election function defined in section 4.3.2 in [RFC7761] the differences include:

- The router, that can be elected as DR, has the highest priority among the DR candidates. The elected DR may not be the one that has the highest priority in the LAN.

- The router that supports the election algorithm defined in section 3.1 MUST advertise the DR Address option defined in section 4.1 in PIM Hello message, and SHOULD advertise the BDR Address option

defined in section 4.2 in PIM Hello message. In case a DR is elected and no BDR is elected, only the DR Address option is advertised in the LAN.

3.2. Sending Hello Messages

When PIM is enabled on an interface or a router first starts, Hello messages MUST be sent with the OptionValue of the DR Address option set to 0x0. The BDR Address option SHOULD also be sent, the OptionValue MUST be set to 0x0. Then the interface starts a timer which value is set to Default_Hello_Holdtime. When the timer expires, the DR and BDR will be elected on the interface according to the DR election algorithm (Section 3.1).

After the election, if there is one existed DR in the LAN, the DR remains unchanged. If there is no existed DR in the LAN, a new DR is elected, the routers in the LAN MUST send the Hello message with the OptionValue of DR Address option set to the elected DR. If there are more than one routers with non-zero DR priority in the LAN, a BDR is also elected. Then the routers in the LAN MUST send the Hello message with the OptionValue of BDR Address option set to the elected BDR. Any DROther router MUST NOT use its IP addresses in the DR/BDR Address option.

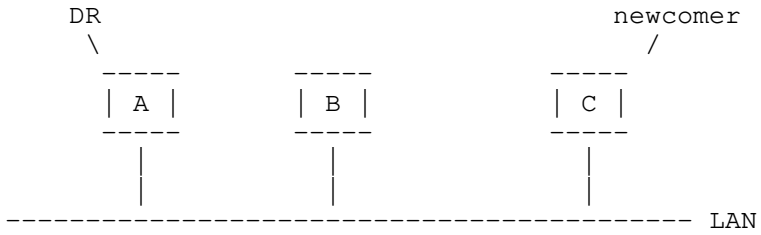


Figure 2

For example, there is a stable LAN that includes RouterA and RouterB. RouterA is the DR that has the highest priority. RouterC is a newcomer. RouterC sends a Hello message with the OptionValue of DR/ BDR Address option set to zero. RouterA and RouterB sends the Hello message with the DR OptionValue set to RouterA, the BDR OptionValue set to RouterB.

In case RouterC has a higher priority than RouterB, RouterC elects itself as the BDR after it runs the election algorithm, then RouterC sends Hello messages with the DR OptionValue set to the IP address of current DR (RouterA), and the BDR OptionValue set to RouterC.

In case RouterB has a higher priority than RouterC, RouterC finds that it can not be the BDR after it runs the election algorithm, it

sets the status to DROther. Then RouterC sends Hello messages with the DR OptionValue set to RouterA and the BDR OptionValue set to RouterB.

3.3. Receiving Hello Messages

When a Hello message is received, the OptionValue of DR/BDR is checked. If the OptionValue of DR is not zero and it isn't the same with local stored values, or the OptionValue of DR is zero but the advertising router is the stored DR, the interface timer of election MAY be set/reset.

Before the election algorithm runs, the validity check MUST be done. The DR/BDR OptionValue in the Hello message MUST match with a known neighbor, otherwise the DR/BDR OptionValue can not become the DR/BDR candidates.

If there is one or more candidates which are different from the stored DR/BDR value after the validity check, the election MUST be taken. The new DR/BDR will be elected according to the rules defined in section 3.1.

3.4. Working with the DRLB function

A network can use the enhancement described in this document with the DR Load Balancing (DRLB) mechanism [RFC8775]. The DR MUST send the DRLB-List Hello Option defined in [RFC8775]. If the DR becomes unreachable, the BDR will take over all the multicast flows on the link, which may result in duplicated traffic as it may not have been a Group DR (GDR). The new DR MUST then follow the procedures in [RFC8775].

In case the DR, or the BDR which becomes DR after the DR failure, doesn't support the mechanism defined in [RFC8775], the DRLB-List Hello Option can not be advertised, then the DRLB mechanism takes no effection.

4. PIM Hello message format

Two new PIM Hello Options are defined, which conform to the format defined in [RFC7761].

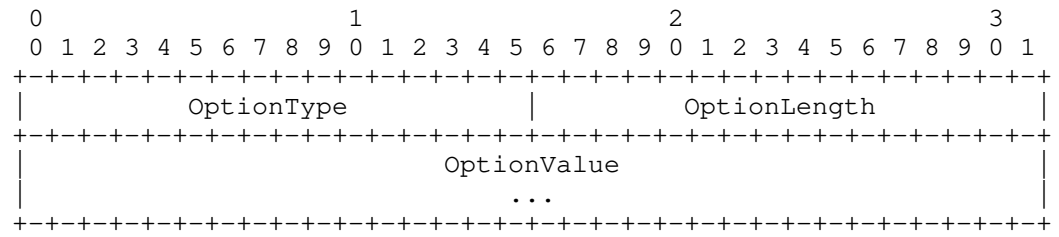


Figure 3: Hello Option Format

4.1. DR Address Option format

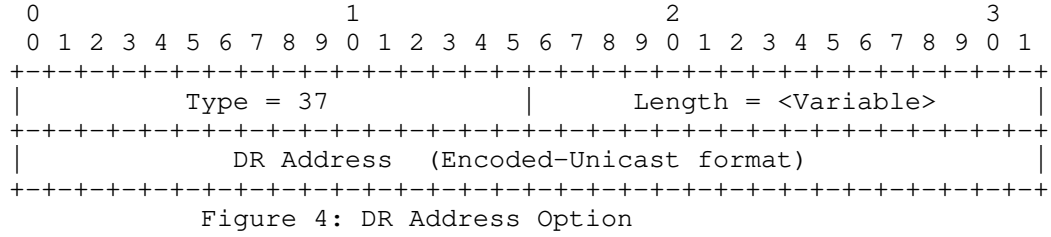


Figure 4: DR Address Option

- OptionType : The value is 37.
- OptionLength: 4 bytes if using IPv4 and 16 bytes if using IPv6.
- DR Address: If the IP version of the PIM message is IPv4, the value MUST be the IPv4 address of the DR. If the IP version of the PIM message is IPv6, the value MUST be the link-local address of the DR.

4.2. BDR Address Option format

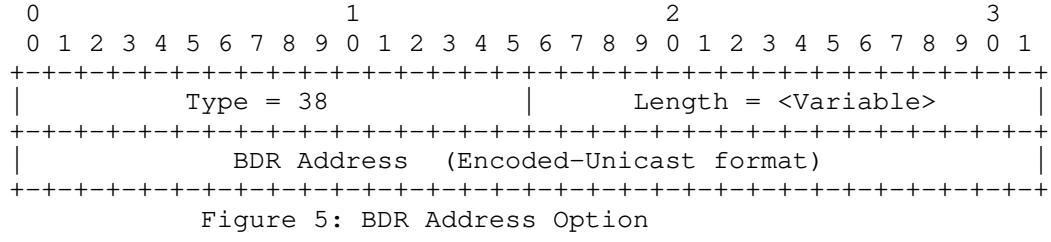


Figure 5: BDR Address Option

- OptionType : The value is 38.
- OptionLength: 4 bytes if using IPv4 and 16 bytes if using IPv6.
- BDR Address: If the IP version of the PIM message is IPv4, the value MUST be the IPv4 address of the BDR. If the IP version of

the PIM message is IPv6, the value MUST be the link-local address of the BDR.

4.3. Error handling

The DR and BDR addresses MUST correspond to an address used to send PIM Hello messages by one of the PIM neighbors on the interface. If that is not the case then the OptionValue of DR/BDR MUST be ignored as described in section 3.3.

An option with unexpected values MUST be ignored. For example, a DR Address option with an IPv4 address received while the interface only supports IPv6 is ignored.

5. Backwards Compatibility

Any router using the DR and BDR Address Options MUST set the corresponding OptionValues. If at least one router on a LAN doesn't send a Hello message, including the DR Address Option, then the specification in this document MUST NOT be used. For example, the routers in a LAN all support the options defined in this document, the DR/BDR is elected. A new router which doesn't support the options joins, when the hello message without DR Address Option is received, all the router MUST switch the election function back immediately. This action results in all routers using the DR election function defined in [RFC7761] or [I-D.mankamana-pim-bdr]. Both this draft and the draft [I-D.mankamana-pim-bdr], introduce a backup DR. The later draft does this without introducing new options but does not consider the sticky behavior. In case there is router which doesn't support the DR/BDR Address Option defined in this document, the routers SHOULD take the function defined in [I-D.mankamana-pim-bdr] if all the routers support it, otherwise the router SHOULD used the function defined in [RFC7761].

A router that does not support this specification ignores unknown options according to section 4.9.2 defined in [RFC7761]. So the new extension defined in this draft will not influence the stability of neighbors.

6. Security Considerations

[RFC7761] describes the security concerns related to PIM-SM. A rogue router can become the DR/BDR by appropriately crafting the Address options to include a more desirable IP address or priority. Because the election algorithm makes the DR role be non-preemptive, an attacker can then take control for long periods of time. The effect of these actions can result in multicast flows not being forwarded (already considered in [RFC7761]).

Some security measures, such as IP address filtering for the election, may be taken to avoid these situations. For example, the Hello message received from an untrusted neighbor is ignored by the election process.

7. IANA Considerations

IANA is requested to allocate two new code points from the "PIM-Hello Options" registry.

Type	Description	Reference
37	DR Address Option	This Document
38	BDR Address Option	This Document

Table 1

8. Acknowledgements

The authors would like to thank Alvaro Retana, Greg Mirsky, Jake Holland, Stig Venaas for their valuable comments and suggestions.

9. References

9.1. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, DOI 10.17487/RFC2328, April 1998, <https://www.rfc-editor.org/info/rfc2328>.

[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, <https://www.rfc-editor.org/info/rfc7761>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8775] Cai, Y., Ou, H., Vallepalli, S., Mishra, M., Venaas, S., and A. Green, "PIM Designated Router Load Balancing", RFC 8775, DOI 10.17487/RFC8775, April 2020, <https://www.rfc-editor.org/info/rfc8775>.

9.2. Informative References

[I-D.ietf-pim-bfd-p2mp-use-case]
Mirsky, G. and J. Xiaoli, "Bidirectional Forwarding Detection (BFD) for Multi-point Networks and Protocol Independent Multicast - Sparse Mode (PIM-SM) Use Case", draft-ietf-pim-bfd-p2mp-use-case-05 (work in progress), November 2020.

[I-D.mankamana-pim-bdr]
mishra, m., Goh, J., and G. Mishra, "PIM Backup Designated Router Procedure", draft-mankamana-pim-bdr-04 (work in progress), April 2020.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

Authors' Addresses

Zheng(Sandy) Zhang
ZTE Corporation
No. 50 Software Ave, Yuhuatai Distinct
Nanjing
China

Email: zhang.zheng@zte.com.cn

Fangwei Hu
Individual
Shanghai
China

Email: hufwei@gmail.com

Benchong Xu
ZTE Corporation
No. 68 Zijinghua Road, Yuhuatai Distinct
Nanjing
China

Email: xu.benchong@zte.com.cn

Mankamana Mishra
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: mankamis@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: September 23, 2021

V. Kamath
VMware
R. Chokkanathapuram Sundaram
Cisco Systems, Inc.
R. Banthia
Apstra
A. Gopal
Cisco Systems, Inc.

March 22, 2021

PIM Null-Register packing
draft-ietf-pim-null-register-packing-08

Abstract

In PIM-SM networks PIM Register messages are sent by the Designated Router (DR) to the Rendezvous Point (RP) to signal the presence of Multicast sources in the network. There are periodic PIM Null-Registers sent from the DR to the RP to keep the state alive at the RP as long as the source is active. The PIM Null-Register message carries information about a single Multicast source and group.

This document defines a standard to send multiple multicast source and group information in a single PIM Null-Register message, in a packed format. This document also discusses the interoperability between the PIM routers which do not understand the packed message format with multiple multicast source and group details.

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 https://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 September 23, 2021.

Table of Contents

1. Introduction

PIM Null-Registers are sent by the DR periodically for Multicast streams to keep the states active on the RP, as long as the multicast source is alive. As the number of multicast sources increases, the number of PIM Null-Register messages that are sent also increases. This results in more PIM packet processing at the RP and the DR.

The control plane policing (COPP), monitors the packets that are processed by the control plane. The high rate at which Null-Registers are received at the RP can lead to COPP drops of Multicast PIM Null-Register messages. This draft proposes a method to efficiently pack multiple PIM Null-Registers [[RFC7761] (Section 4.4)] and Register-Stops [[RFC7761] (Section 3.2)] into a

single message as these packets anyway do not contain encapsulated data.

The draft also discusses interoperability with PIM routers that do not understand the new packet format.

1.1. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] when, and only when, they appear in all capitals, as shown here.

1.2. Terminology

RP: Rendezvous Point

DR: Designated Router

2. Packed Null-Register Capability

A router (DR) can decide to pack multiple Null-Register messages based on the capability received from the RP as part of Register-Stop. This ensures compatibility with routers that do not support processing of the new format. The capability information can be indicated by the RP via the PIM Register-Stop message sent to the DR. Thus a DR will switch to the new format only when it learns that the RP is capable of handling the packed Null-Register messages.

Conversely, a DR that does not support the new format can continue generating the PIM Null-Register the current way. To exchange the capability information in the Register-Stop message, the "reserved" field can be used to indicate this capability in those Register-Stop messages. One bit of the reserved field is used to indicate the "packing" capability (P bit). The rest of the bits in the "Reserved" field will be retained for future use.

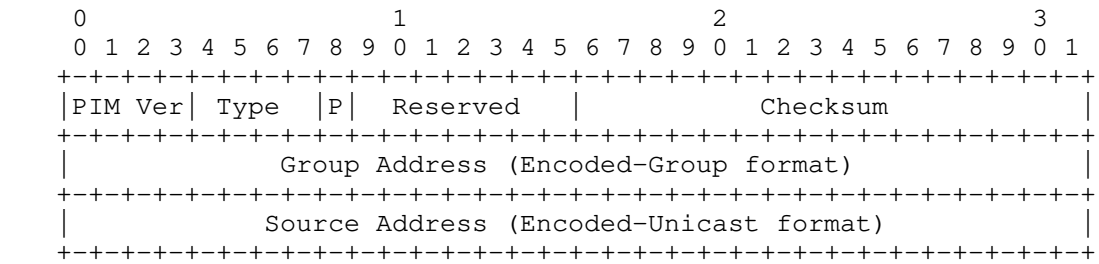


Figure 1: PIM Register-Stop message with capability option

PIM Version, Type, Checksum, Group Address, Source Address:

Same as [RFC7761] (Section 4.9.4)

P:

Capability bit (flag bit 7) used to indicate support for the Packed Null-Register Capability

3. PIM Packed Null-Register message

PIM Packed Null-Register message format includes a count to indicate the number of Null-Register records in the message.

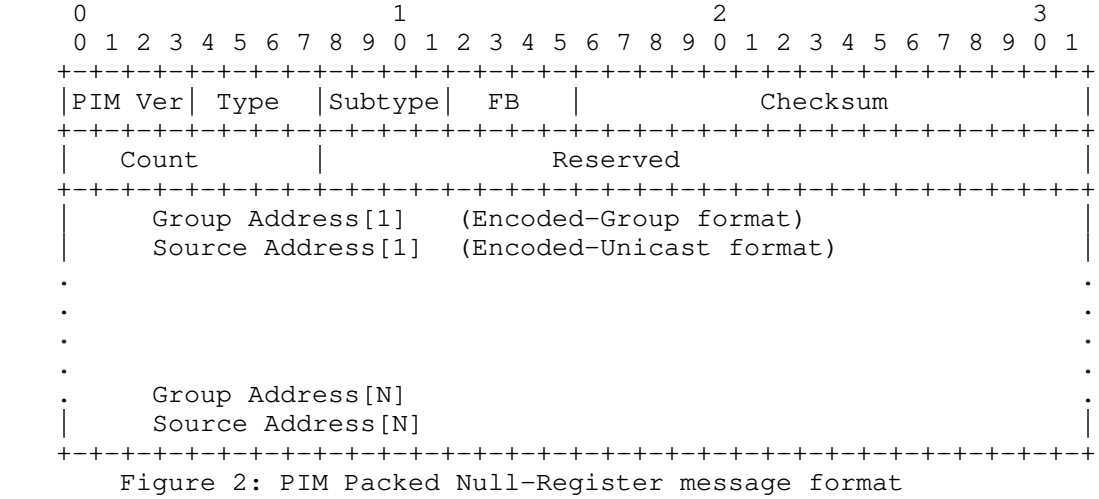


Figure 2: PIM Packed Null-Register message format

PIM Version, Reserved, Checksum:

Same as [RFC7761] (Section 4.9.3)

Type, SubType:

The new packed Null-Register Type and SubType values TBD. [RFC8736]

Count:

The number of packed Null-Register records. A record consists of a Group Address and Source Address pair.

Group Address, Source Address:

Same as [RFC7761] (Section 4.9.4)

4. PIM Packed Register-Stop message format

The PIM Packed Register-Stop message includes a count to indicate the number of records that are present in the message.

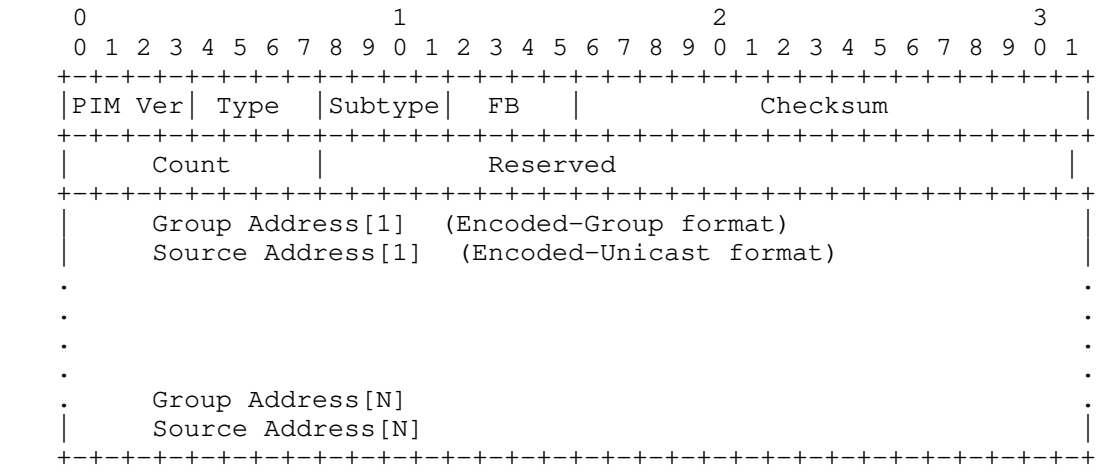


Figure 3: PIM Packed Register-Stop message format

PIM Version, Reserved, Checksum:

Same as [RFC7761] (Section 4.9.4)

Type:

The new Register Stop Type and SubType values TBD

Count:

The number of packed Register-Stop records. A record consists of a Group Address and Source Address pair.

Group Address, Source Address:

Same as [RFC7761] (Section 4.9.4)

5. Protocol operation

The following combinations exist -

1. DR and RP both support the PIM Packed Null-Register format

   * As specified in [[RFC7761]], the DR sends PIM Register messages towards the RP when a new source is detected.

   * An RP supporting this specification MUST set the P-bit in the corresponding Register-Stop messages.

   * When a Register-Stop message with the P-bit set is received, the DR MAY send Packed Null-Register messages (Section 3) to the RP instead of multiple Register messages with the N-bit set ([[RFC7761]]).

   * The RP, after receiving a Packed Null-Register message MAY start sending Packed Register-Stop messages (Section 4) to the corresponding DR instead of individual Register-Stop messages.

2. DR supports but RP does not support PIM Packed Null-Register format

   * As specified in [[RFC7761]], DR sends PIM Register towards the RP.

   * RP sends a normal Register-Stop without any capability information.

   * DR then sends Null-Registers in the old format. [[RFC7761]]

3. RP supports but DR doesn’t support the new PIM Packed Null-Register format

   * As specified in [[RFC7761]], DR sends the PIM Register towards the RP.

   * P sends a PIM Packed Register-Stop towards the DR that includes capability information.

* Since DR does not support the new format, it sends Null-Registers in the old format. [[RFC7761]]

6. PIM Anycast RP considerations

The PIM Packed Null-Register format should be enabled only if it is supported by all PIM Anycast RP [[RFC4610]] members in the RP set for the RP address.

7. PIM RP router version downgrade

Consider a PIM RP router that supports PIM Register Packing and then downgrades to a software version which does not support PIM Register Packing. The DR that sends the PIM Packed Null-Register message will not get a PIM Register-Stop message back. In such scenarios the DR can send an unpacked PIM Null-Register and check the PIM Register-Stop to see if the capability bit (P-bit) for PIM Packed Null-Register is set or not. If it is not set then the DR will continue sending unpacked PIM Null-Register messages.

8. Fragmentation consideration

When building a PIM Packed Null-Register message or PIM Packed Register-Stop message, a router should include as many records as possible based on the path MTU towards RP, if path MTU discovery is done. Otherwise, the number of records should be limited by the MTU of the outgoing interface.

9. Security Considerations

General Register messages security considerations from RFC7761 apply. As mentioned in RFC7761, PIM Null-Register messages and Register-Stop messages are forwarded by intermediate routers to their destination using normal IP forwarding. Without data origin authentication, an attacker who is located anywhere in the network may be able to forge a Null-Register or Register-Stop message. We next consider the effect of a forgery of each of these messages. By forging a Register message, an attacker can cause the RP to inject forged traffic onto the shared multicast tree.

By forging a Register-Stop message, an attacker can prevent a legitimate DR from registering packets to the RP. This can prevent local hosts on that LAN from sending multicast packets. The above two PIM messages are not changed by intermediate routers and need only be examined by the intended receiver. Thus, these messages can be authenticated end-to-end. Attacks on Register and Register-Stop messages do not apply to a PIM-SSM-only implementation, as these messages are not used in PIM-SSM.

There is another case where a spoofed Register-Stop can be sent to make it appear that is from the RP, and that the RP supports this new packed capability when it does not. This can cause Null-Registers to be sent to an RP that doesnt support this packed format. But standard methods to prevent spoofing should take care of this case. For example, uRPF can be used to filter out packets coming from the outside from addresses that belong to routers inside.

10. IANA Considerations

This document requires the assignment of Capability bit (P-bit), flag bit 7 in the PIM Register-Stop message.

This document requires the assignment of 2 new PIM message types for the PIM Packed Null-Register and PIM Packed Register-Stop.

11. Acknowledgments

The authors would like to thank Stig Venaas, Anish Peter, Zheng Zhang and Umesh Dudani for their helpful comments on the draft.

12. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4610] Farinacci, D. and Y. Cai, "Anycast-RP Using Protocol Independent Multicast (PIM)", RFC 4610, DOI 10.17487/RFC4610, August 2006, <https://www.rfc-editor.org/info/rfc4610>.

[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, <https://www.rfc-editor.org/info/rfc7761>.

[RFC8736] Venaas, S. and A. Retana, "PIM Message Type Space Extension and Reserved Bits", RFC 8736, DOI 10.17487/RFC8736, February 2020, <https://www.rfc-editor.org/info/rfc8736>.

Authors' Addresses

Vikas Ramesh Kamath
VMware
3401 Hillview Ave
Palo Alto CA 94304
USA

Email: vkamath@vmware.com

Ramakrishnan Chokkanathapuram Sundaram
Cisco Systems, Inc.
Tasman Drive
San Jose CA 95134
USA

Email: ramaksun@cisco.com

Raunak Banthia
Apstra
333 Middlefield Rd STE 200
Menlo Park CA 94025
USA

Email: rbanthia@apstra.com

Ananya Gopal
Cisco Systems, Inc.
Tasman Drive
San Jose CA 95134
USA

Email: ananygop@cisco.com

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: August 22, 2021

D. Voyer, Ed.
Bell Canada
C. Filsfils
R. Parekh
Cisco Systems, Inc.
H. Bidgoli
Nokia
Z. Zhang
Juniper Networks
February 18, 2021

Segment Routing Point-to-Multipoint Policy
draft-ietf-pim-sr-p2mp-policy-02

Abstract

This document describes an architecture to construct a Point-to-Multipoint (P2MP) tree to deliver Multi-point services in a Segment Routing domain. A SR P2MP tree is constructed by stitching a set of Replication segments together. A SR Point-to-Multipoint (SR P2MP) Policy is used to define and instantiate a P2MP tree which is computed by a PCE.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 https://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 August 22, 2021.

Table of Contents

1. Introduction

A Multi-point service delivery could be realized via P2MP trees in a Segment Routing domain [RFC8402]. A P2MP tree spans from a Root node to a set of Leaf nodes via intermediate Replication Nodes. It consists of a Replication segment [I-D.ietf-spring-sr-replication-segment] at the root node, one or more Replication segments at Leaf nodes and intermediate Replication Nodes. The Replication segments are stitched together.

A Segment Routing P2MP policy, a variant of the SR Policy [I-D.ietf-spring-segment-routing-policy], is used to define a P2MP tree. A PCE is used to compute the tree from the Root node to the set of Leaf nodes via a set of Replication Nodes. The PCE then instantiates the P2MP tree in the SR domain by signaling Replication segments to Root, replication and Leaf nodes using various protocols (PCEP, BGP, NetConf etc.). Replication segments of a P2MP tree can be instantiated for SR-MPLS and SRv6 dataplanes.

2. P2MP Tree

A P2MP tree in a SR domain connects a Root to a set of Leaf nodes via a set of intermediate Replication Nodes. It consists of a Replication segment at the root stitched to Replication segments at intermediate Replication Nodes eventually reaching the Leaf nodes.

The Replication SID of the Replication segment at Root node is called Tree-SID. The Tree-SID SHOULD also be used as Replication SID of Replication segments at Replication and Leaf nodes. The Replication segments at Replication and Leaf nodes MAY use Replication SIDs that are not same as the Tree-SID.

The Replication segment at Root of a P2MP tree MUST be associated with that P2MP tree (i.e. <Root, Tree-ID> identifier in SR P2MP policy section below) to map a Multi-point service to the tree. A Replication segment that terminates a P2MP tree at a Leaf node MUST be associated with the P2MP tree to determine the context for a Multi-point service. The The information that can be used to derive this association is specific to encoding of the protocol (PCEP, BGP, NetConf etc.) used to instantiate the Replication segment for a P2MP tree. Replication segments at intermediate Replication Nodes of a tree are also associated with that tree.

For SR-MPLS, a PCE MAY decide not to instantiate Replication segments at Leaf nodes of a P2MP tree if it is known a priori that Multi-point

services mapped to the P2MP tree can be identified using a context that is globally unique in SR domain. In this case, Replication Nodes connecting to Leaf nodes effectively does Penultimate-Hop Pop (PHP) behavior to pop Tree-SID from a packet. A Multi-point service context assigned from "Domain-wide Common Block" (DCB) [I-D.ietf-bess-mvpn-evpn-aggregation-label] is an example of globally unique context.

A packet steered into a P2MP tree is replicated by the Replication segment at Root node to each downstream node in the Replication segment, with the Replication SID of the Replication segment at the downstream node. A downstream node could be a Leaf node or an intermediate Replication Node. In the latter case, replication continues with the Replication segments until all Leaf nodes are reached. A packet is steered into a P2MP tree in two ways:

- Based on a local policy-based routing at the Root node.
- Based on steering via the Tree-SID at the Root node.

2.1. Sharing Replication segments across P2MP trees

Two or more P2MP trees MAY share a Replication segment at Root or Replication Nodes if at minimum the first condition below is satisfied. A tree always has its own Replication segment at its root even if shares another Replication segment. A tree that shares another Replication segment may or may not have its own Replication segment on its Leaf nodes. If not, the second and third conditions apply to such situations.

1. The Leaf nodes reached via a shared Replication segment must be subset of Leaf or Replication Nodes of the P2MP trees that share this segment. Note if a Replication segment is shared, all its downstream Replication segments are also shared.

2. Some Multi-point services realized by the P2MP trees may need service context (e.g. packets are for certain VPNs, and/or from certain nodes). If the trees do not have their own Replication segments at their Leaf nodes then the packets transported on the P2MP trees MUST carry a service context that does not rely on the tree or root identification, e.g. a service label assigned from Domain-wide Common Block or common SRGB for SR-MPLS.

3. For some Multi-point services using P2MP trees that share Replication segments, packets transported on these trees MAY require a Tree context (e.g. MVPN Extranet [RFC7900] to avoid certain ambiguities - see Section 2.3.1 of RFC 7900). In this case, the trees MUST have their own Replication segments on the

Leaf nodes. For SR-MPLS, this is similar to "tunnel stacking" concept.

Sharing of a Replication segment for P2MP trees is OPTIONAL. Exact procedures to ensure validity of above conditions across PM2P services on nodes of a Segment Routing domain are outside the scope of this document.

3. SR P2MP Policy

The SR P2MP policy is a variant of an SR policy[I-D.ietf-spring-segment-routing-policy] and is used to instantiate SR P2MP trees.

A SR P2MP Policy is identified by the tuple <Root, Tree-ID>, where:

- Root: The address of Root node of P2MP tree instantiated by the SR P2MP Policy
- Tree-ID: A identifier that is unique in context of the Root. This is an unsigned 32-bit number.

A SR P2MP Policy is defined by following elements:

- Leaf nodes: A set of nodes that terminate the P2MP trees.
- Candidate Paths: See below.

A SR P2MP policy is provisioned on a PCE to instantiate the P2MP tree. The Tree-SID SHOULD be used as Binding SID of the P2MP policy. A PCE computes the P2MP tree and instantiates Replication segments at Root, Replication and Leaf nodes. When Replication segments are not shared across P2MP trees, the Root and Tree-ID of the SR P2MP policy are mapped to Replication-ID element of the Replication segment identifier i.e the SR Replication segment identifier is <Root, Tree-ID, Node-ID>. A shared Replication segment MAY be identified with zero Root-ID address (0.0.0.0 for IPv4 and :: for IPv6) and a Replication-ID that is unique in context of Node address where the Replication segment is instantiated when it is not associated a particular tree.

A SR P2MP Policy has one or more Candidate paths. The active Candidate path is selected based on the tie breaking rules amongst the candidate-paths as specified in[I-D.ietf-spring-segment-routing-policy]. Each candidate path has a set of topological/resource constraints and/or optimization objectives which determine the P2MP tree for that Candidate path. Tree-SID is an identifier of the P2MP tree of the candidate path in

the forwarding plane. It is instantiated in the forwarding plane at Root node, intermediate Replication Nodes and Leaf nodes. The Tree-SID MAY be different at Replication and Leaf nodes.

4. Using Controller to build a P2MP Tree

A P2MP tree can be built using a Path Computation Element (PCE). This section outlines a high-level architecture for such an approach.

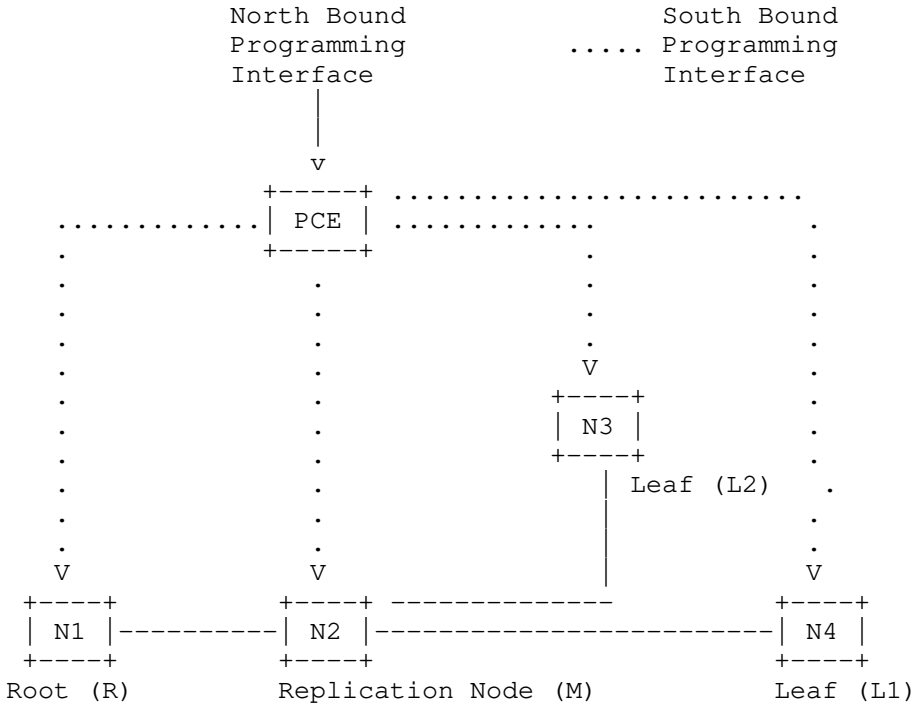


Figure 1: Centralized Control Plane Model

4.1. Provisioning SR P2MP Policy Creation

A SR P2MP policy can be instantiated and maintained in a centralized fashion using a Path Computation Element (PCE).

4.1.1. API

North-bound APIs on a PCE can be used to:

1. Create SR P2MP policy: CreateSRP2MPPolicy<Root, Tree-ID>

2. Delete SR P2MP policy: DeleteSRP2MPPolicy<Root, Tree-ID>

3. Modify SR P2MP policy Leaf Set: SRP2MPPolicyLeafSetModify<Root, Tree-ID, {Leaf Set}>

4. Create a Candidate Path for SR P2MP policy: CreateSRP2MPCandidatePath<Root, Tree-ID, <CP-ID>>

5. Delete a Candidate Path for SR P2MP policy: DeleteSRP2MPCandidatePath<Root, Tree-ID, <CP-ID>>

6. Update a Candidate Path for SR P2MP policy: UpdateSRP2MPCandidatePath<Root, Tree-ID, <CP-ID>, Preference, [Constraints], [Optimization], ...>

CP-ID is identifier of a Candidate Path within a SR P2MP policy. One possible identifier is the tuple <Protocol-Origin, originator, discriminator> as specified in [I-D.ietf-spring-segment-routing-policy].

Note these are conceptual APIs. Actual implementations may offer different APIs as long as they provide same functionality. For example, API might allow symbolic name to be assigned for a P2MP policy or APIs might allow individual Leaf nodes to be added or deleted from a policy instead of an update operation.

4.1.2. Invoking API

Interaction with a PCE can be via PCEP, REST, Netconf, gRPC, CLI. Yang model shall be be developed for this purpose as well.

4.2. P2MP Tree Computation

An entity (an operator, a network node or a machine) provisions a SR P2MP policy by specifying the addresses of the root (R) and set of leaves {L} as well as Traffic Engineering (TE) attributes of Candidate paths via a suitable North-Bound API. The PCE computes the tree of Active candidate path. The PCE MAY compute P2MP trees for all Candidate paths., If tree computation is successful, PCE instantiates the P2MP tree(s) using Replication segments on Root, Replication, and Leaf nodes.

Candidate path constraints shall include link color affinity, bandwidth, disjointness (link, node, SRLG), delay bound, link loss, etc. Candidate path shall be optimized based on IGP or TE metric or link latency.

The Tree SID of Candidate path of a SR P2MP policy can be either dynamically allocated by the PCE or statically assigned by entity provisioning the SR P2MP policy. Ideally, same Tree-SID SHOULD be used for Replication segments at Root, Replication, and Leaf nodes. Different Tree-SIDs MAY be used at Replication Node(s) if it is not feasible to use same Tree SID.

A PCE can modify a P2MP tree following network element failure or in case a better path can be found based on the new network state. In this case, the PCE may want to setup the new instance of the tree and remove the old instance of the tree from the network in order to minimize traffic loss. In this case, the instances of trees for all the Candidate paths of a P2MP policy can be identified by an Instance-ID which is unique in context of the P2MP policy. As such, the identifier of non-shared Replication segments used to instantiate these trees becomes <Root-ID, Tree-ID, Node-ID, Instance-ID>.

A PCE shall be capable of computing paths across multiple IGP areas or levels as well as Autonomous Systems (ASs).

4.2.1. Topology Discovery

A PCE shall learn network topology, TE attributes of link/node as well as SIDs via dynamic routing protocols (IGP and/or BGP-LS). It may be possible for entities to pass topology information to PCE via north-bound API.

4.2.2. Capability and Attribute Discovery

It shall be possible for a node to advertise SR P2MP tree capability via IGP and/or BGP-LS. Similarly, a PCE can also advertise its P2MP tree computation capability via IGP and/or BGP-LS. Capability advertisement allows a network node to dynamically choose one or more PCE(s) to obtain services pertaining to SR P2MP policies, as well a PCE to dynamically identify SR P2MP tree capable nodes.

4.3. Instantiating P2MP tree on nodes

Once a PCE computes a P2MP tree for Candidate path of SR P2MP policy, it needs to instantiate the tree on the relevant network nodes via Replication segments. The PCE can use various protocols to program the Replication segments as described below.

4.3.1. PCEP

PCE Protocol (PCEP)has been traditionally used:

1. For a head-end to obtain paths from a PCE.

2. A PCE to instantiate SR policies.

PCEP protocol can be stateful in that a PCE can have a stateful control of an SR policy on a head-end which has delegated the control of the SR policy to the PCE. PCEP shall be extended to provision and maintain SR P2MP trees in a stateful fashion.

4.3.2. BGP

BGP has been extended to instantiate and report SR policies. It shall be extended to instantiate and maintain P2MP trees for SR P2MP policies.

4.3.3. NetConf

TBD

4.4. Protection

4.4.1. Local Protection

A network link, node or path on the tree of a P2MP tree can be protected using SR policies computed by PCE. The backup SR policies shall be programmed in forwarding plane in order to minimize traffic loss when the protected link/node fails. It is also possible to use node local Fast Re-Route protection mechanisms (LFA) to protect link/ nodes of P2MP tree.

4.4.2. Path Protection

It is possible for PCE create a disjoint backup tree for providing end-to-end path protection.

5. IANA Considerations

This document makes no request of IANA.

6. Security Considerations

There are no additional security risks introduced by this design.

7. Acknowledgements

The authors would like to acknowledge Siva Sivabalan, Mike Koldychev and Vishnu Pavan Beeram for their valuable inputs..

8. Contributors

Clayton Hassen
Bell Canada
Vancouver
Canada

Email: clayton.hassen@bell.ca

Kurtis Gillis
Bell Canada
Halifax
Canada

Email: kurtis.gillis@bell.ca

Arvind Venkateswaran
Cisco Systems, Inc.
San Jose
US

Email: arvvenka@cisco.com

Zafar Ali
Cisco Systems, Inc.
US

Email: zali@cisco.com

Swadesh Agrawal
Cisco Systems, Inc.
San Jose
US

Email: swaagraw@cisco.com

Jayant Kotalwar
Nokia
Mountain View
US

Email: jayant.kotalwar@nokia.com

Tanmoy Kundu
Nokia
Mountain View
US

Email: tanmoy.kundu@nokia.com

Andrew Stone
Nokia
Ottawa
Canada

Email: andrew.stone@nokia.com

Tarek Saad
Juniper Networks
Canada

Email:tsaad@juniper.net

9. References

9.1. Normative References

[I-D.ietf-spring-segment-routing-policy]
Filsfils, C., Talaulikar, K., Voyer, D., Bogdanov, A., and P. Mattes, "Segment Routing Policy Architecture", draft-ietf-spring-segment-routing-policy-09 (work in progress), November 2020.

[I-D.ietf-spring-sr-replication-segment]
Voyer, D., Filsfils, C., Parekh, R., Bidgoli, H., and Z. Zhang, "SR Replication Segment for Multi-point Service Delivery", draft-ietf-spring-sr-replication-segment-02 (work in progress), October 2020.

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

[RFC8402] Filsfils, C., Ed., Previdi, S., Ed., Ginsberg, L., Decraene, B., Litkowski, S., and R. Shakir, "Segment Routing Architecture", RFC 8402, DOI 10.17487/RFC8402, July 2018, <https://www.rfc-editor.org/info/rfc8402>.

9.2. Informative References

[I-D.filsfils-spring-srv6-net-pgm-illustration]
Filsfils, C., Camarillo, P., Li, Z., Matsushima, S., Decraene, B., Steinberg, D., Lebrun, D., Raszuk, R., and J. Leddy, "Illustrations for SRv6 Network Programming", draft-filsfils-spring-srv6-net-pgm-illustration-03 (work in progress), September 2020.

[I-D.ietf-bess-mvpn-evpn-aggregation-label]
Zhang, Z., Rosen, E., Lin, W., Li, Z., and I. Wijnands, "MVPN/EVPN Tunnel Aggregation with Common Labels", draft-ietf-bess-mvpn-evpn-aggregation-label-05 (work in progress), January 2021.

[I-D.ietf-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., Voyer, D., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-ietf-spring-srv6-network-programming-28 (work in progress), December 2020.

[RFC7900] Rekhter, Y., Ed., Rosen, E., Ed., Aggarwal, R., Cai, Y., and T. Morin, "Extranet Multicast in BGP/IP MPLS VPNs", RFC 7900, DOI 10.17487/RFC7900, June 2016, <https://www.rfc-editor.org/info/rfc7900>.

Appendix A. Illustration of SR P2MP Policy and P2MP Tree

Consider the following topology:

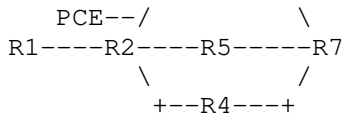


Figure 1

In these examples, the Node-SID of a node Rn is N-SIDn and Adjacency-SID from node Rm to node Rn is A-SIDmn. Interface between Rm and Rn is Lmn.

For SRv6, the reader is expected to be familiar with SRv6 Network Programming [I-D.ietf-spring-srv6-network-programming] to follow the examples. We use SID allocation scheme, reproduced below, from Illustrations for SRv6 Network Programming [I-D.filsfils-spring-srv6-net-pgm-illustration]

2001:db8::/32 is an IPv6 block allocated by a RIR to the operator

2001:db8:0::/48 is dedicated to the internal address space

2001:db8:cccc::/48 is dedicated to the internal SRv6 SID space

We assume a location expressed in 64 bits and a function expressed in 16 bits

Node k has a classic IPv6 loopback address 2001:db8::k/128 which is advertised in the IGP

Node k has 2001:db8:cccc:k::/64 for its local SID space. Its SIDs will be explicitly assigned from that block

Node k advertises 2001:db8:cccc:k::/64 in its IGP

Function :1:: (function 1, for short) represents the End function with PSP support

Function :Cn:: (function Cn, for short) represents the End.X function to Node n

Function :C1n: (function C1n for short) represents the End.X function to Node n with USD

Each node k has:

An explicit SID instantiation 2001:db8:cccc:k:1::/128 bound to an End function with additional support for PSP

An explicit SID instantiation 2001:db8:cccc:k:Cj::/128 bound to an End.X function to neighbor J with additional support for PSP

An explicit SID instantiation 2001:db8:cccc:k:C1j::/128 bound to an End.X function to neighbor J with additional support for USD

Assume PCE is provisioned following SR P2MP policy at Root R1 with Tree-ID T-ID:

```
SR P2MP Policy <R1,T-ID>:
 Leaf Nodes: {R2, R6, R7}
 Candidate-path 1:
   Optimize: IGP metric
   Tree-SID: T-SID1
```

The PCE is responsible for P2MP tree computation. Assume PCE instantiates P2MP trees by signalling non-shared Replication segments i.e. Replication-ID of these Replication segments is <Root, Tree-ID>. If a Candidate-path can have multiple instances of P2MP trees, the

Replication-ID is <Root, Tree-ID, Instance-ID>. In this example, we assume one instance of P2MP tree for a candidate-path. All Replication segments use the Tree-SID T-SID1 as Replication-SID. For SRv6, assume the Replication SID at node k, bound to an End.Replcate function, is 2001:db8:cccc:k:FA::/128.

A.1. P2MP Tree with non-adjacent Replication Segments

Assume PCE computes a P2MP tree with Root node R1, Intermediate and Leaf node R2, and Leaf nodes R6 and R7. The PCE instantiates the P2MP tree by stitching Replication segments at R1, R2, R6 and R7. Replication segment at R1 replicates to R2. Replication segment at R2 replicates to R6 and R7. Note nodes R3, R4 and R5 do not have any Replication segment state for the tree.

A.1.1. SR-MPLS

The Replication segment state at nodes R1, R2, R6 and R7 is shown below.

Replication segment at R1:

```
Replication segment <R1,T-ID,R1>:
 Replication SID: T-SID1
 Replication State:
   R2: <T-SID1->L12>
```

Replication to R2 steers packet directly to the node on interface L12.

Replication segment at R2:

```
Replication segment <R1,T-ID,R2>:
 Replication SID: T-SID1
 Replication State:
   R2: <Leaf>
   R6: <N-SID6, T-SID1>
   R7: <N-SID7, T-SID1>
```

R2 is a Bud-Node. It performs role of Leaf as well as a transit node replicating to R6 and R7. Replication to R6, using N-SID6, steers packet via IGP shortest path to that node. Replication to R7, using N-SID7, steers packet via IGP shortest path to R7 via either R5 or R4 based on ECMP hashing.

Replication segment at R6:

```
Replication segment <R1,T-ID,R6>:
 Replication SID: T-SID1
 Replication State:
   R6: <Leaf>

Replication segment at R7:

Replication segment <R1,T-ID,R7>:
 Replication SID: T-SID1
 Replication State:
   R7: <Leaf>
```

When a packet is steered into the SR P2MP Policy at R1:

- Since R1 is directly connected to R2, R1 performs PUSH operation with just <T-SID1> label for the replicated copy and sends it to R2 on interface L12.

- R2, as Leaf, performs NEXT operation, pops T-SID1 label and delivers the payload. For replication to R6, R2 performs a PUSH operation of N-SID6, to send <N-SID6,T-SID1> label stack to R3. R3 is the penultimate hop for N-SID6; it performs penultimate hop popping, which corresponds to the NEXT operation and the packet is then sent to R6 with <T-SID1> in the label stack. For replication to R7, R2 performs a PUSH operation of N-SID7, to send <N-SID7,T-SID1> label stack to R4, one of IGP ECMP nexthops towards R7. R4 is the penultimate hop for N-SID6; it performs penultimate hop popping, which corresponds to the NEXT operation and the packet is then sent to R7 with <T-SID1> in the label stack.

- R6, as Leaf, performs NEXT operation, pops T-SID1 label and delivers the payload.

- R7, as Leaf, performs NEXT operation, pops R-SID7 label and delivers the payload.

A.1.2. SRv6

For SRv6, the replicated packet from R2 to R7 has to traverse R4 using a SR-TE policy, Policy27. The policy has one SID in segment list: End.X function with USD of R4 to R7 . The Replication segment state at nodes R1, R2, R6 and R7 is shown below.

```
Policy27: <2001:db8:cccc:4:C17::>

Replication segment at R1:
```

```
Replication segment <R1,T-ID,R1>:
 Replication SID: 2001:db8:cccc:1:FA::
 Replication State:
   R2: <2001:db8:cccc:2:FA::->L12>
```

Replication to R2 steers packet directly to the node on interface L12.

Replication segment at R2:

```
Replication segment <R1,T-ID,R2>:
 Replication SID: 2001:db8:cccc:2:FA::
 Replication State:
   R2: <Leaf>
   R6: <2001:db8:cccc:6:FA::>
   R7: <2001:db8:cccc:7:FA:: -> Policy27>
```

R2 is a Bud-Node. It performs role of Leaf as well as a transit node replicating to R6 and R7. Replication to R6, steers packet via IGP shortest path to that node. Replication to R7, via SR-TE policy, first encapsulates the packet using H.Encaps and then steers the outer packet to R4. End.X USD on R4 decapsulates outer header and sends the original inner packet to R7.

Replication segment at R6:

```
Replication segment <R1,T-ID,R6>:
 Replication SID: 2001:db8:cccc:6:FA::
 Replication State:
   R6: <Leaf>
```

Replication segment at R7:

```
Replication segment <R1,T-ID,R7>:
 Replication SID: 2001:db8:cccc:7:FA::
 Replication State:
   R7: <Leaf>
```

When a packet (A,B2) is steered into the SR P2MP Policy at R1 using H.Encaps.Replicate behavior:

- Since R1 is directly connected to R2, R1 sends replicated copy (2001:db8::1, 2001:db8:cccc:2:FA::) (A,B2) to R2 on interface L12.

- R2, as Leaf removes outer IPv6 header and delivers the payload. R2, as a bud node, also replicates the packet.

-

* For replication to R6, R2 sends (2001:db8::1, 2001:db8:cccc:6:FA::) (A,B2) to R3. R3 forwards the packet using 2001:db8:cccc:6::/64 packet to R6.

* For replication to R7 using Policy27, R2 encapsulates and sends (2001:db8::2, 2001:db8:cccc:4:C17::) (2001:db8::1, 2001:db8:cccc:7:FA::) (A,B2) to R4. R4 performs End.X USD behavior, decapsulates outer IPv6 header and sends (2001:db8::1, 2001:db8:cccc:7:FA::) (A,B2) to R7.

- R6, as Leaf, removes outer IPv6 header and delivers the payload.

- R7, as Leaf, removes outer IPv6 header and delivers the payload.

A.2. P2MP Tree with adjacent Replication Segments

Assume PCE computes a P2MP tree with Root node R1, Intermediate and Leaf node R2, Intermediate nodes R3 and R5, and Leaf nodes R6 and R7. The PCE instantiates the P2MP tree by stitching Replication segments at R1, R2, R3, R5, R6 and R7. Replication segment at R1 replicates to R2. Replication segment at R2 replicates to R3 and R5. Replication segment at R3 replicates to R6. Replication segment at R5 replicates to R7. Note node R4 does not have any Replication segment state for the tree.

A.2.1. SR-MPLS

The Replication segment state at nodes R1, R2, R3, R5, R6 and R7 is shown below.

Replication segment at R1:

```
Replication segment <R1,T-ID,R1>:
 Replication SID: T-SID1
 Replication State:
   R2: <T-SID1->L12>
```

Replication to R2 steers packet directly to the node on interface L12.

Replication segment at R2:

```
Replication segment <R1,T-ID,R2>:
 Replication SID: T-SID1
 Replication State:
   R2: <Leaf>
   R3: <T-SID1->L23>
   R5: <T-SID1->L25>
```

R2 is a Bud-Node. It performs role of Leaf as well as a transit node replicating to R3 and R5. Replication to R3, steers packet directly to the node on L23. Replication to R5, steers packet directly to the node on L25.

Replication segment at R3:

```
Replication segment <R1,T-ID,R3>:
 Replication SID: T-SID1
 Replication State:
   R6: <T-SID1->L36>
```

Replication to R6, steers packet directly to the node on L36.

Replication segment at R5:

```
Replication segment <R1,T-ID,R5>:
 Replication SID: T-SID1
 Replication State:
   R7: <T-SID1->L57>
```

Replication to R7, steers packet directly to the node on L57.

Replication segment at R6:

```
Replication segment <R1,T-ID,R6>:
 Replication SID: T-SID1
 Replication State:
   R6: <Leaf>
```

Replication segment at R7:

```
Replication segment <R1,T-ID,R7>:
 Replication SID: T-SID1
 Replication State:
   R7: <Leaf>
```

When a packet is steered into the SR P2MP Policy at R1:

- Since R1 is directly connected to R2, R1 performs PUSH operation with just <T-SID1> label for the replicated copy and sends it to R2 on interface L12.

- R2, as Leaf, performs NEXT operation, pops T-SID1 label and delivers the payload. It also performs PUSH operation on T-SID1 for replication to R3 and R5. For replication to R6, R2 sends <T-SID1> label stack to R3 on interface L23. For replication to R5, R2 sends <T-SID1> label stack to R5 on interface L25.

- R3 performs NEXT operation on T-SID1 and performs a PUSH operation for replication to R6 and sends <T-SID1> label stack to R6 on interface L36.

- R5 performs NEXT operation on T-SID1 and performs a PUSH operation for replication to R7 and sends <T-SID1> label stack to R7 on interface L57.

- R6, as Leaf, performs NEXT operation, pops T-SID1 label and delivers the payload.

- R7, as Leaf, performs NEXT operation, pops R-SID7 label and delivers the payload.

A.2.2. SRv6

The Replication segment state at nodes R1, R2, R3, R5, R6 and R7 is shown below.

Replication segment at R1:

```
Replication segment <R1,T-ID,R1>:
 Replication SID: 2001:db8:cccc:1:FA::
 Replication State:
   R2: <2001:db8:cccc:2:FA::->L12>
```

Replication to R2 steers packet directly to the node on interface L12.

Replication segment at R2:

```
Replication segment <R1,T-ID,R2>:
 Replication SID: 2001:db8:cccc:2:FA::
 Replication State:
   R2: <Leaf>
   R3: <2001:db8:cccc:3:FA::->L23>
   R5: <2001:db8:cccc:5:FA::->L25>
```

R2 is a Bud-Node. It performs role of Leaf as well as a transit node replicating to R3 and R5. Replication to R3, steers packet directly to the node on L23. Replication to R5, steers packet directly to the node on L25.

Replication segment at R3:

```
Replication segment <R1,T-ID,R3>:
 Replication SID: 2001:db8:cccc:3:FA::
 Replication State:
   R6: <2001:db8:cccc:6:FA::->L36>
```

Replication to R6, steers packet directly to the node on L36.

Replication segment at R5:

```
Replication segment <R1,T-ID,R5>:
 Replication SID: 2001:db8:cccc:5:FA::
 Replication State:
   R7: <2001:db8:cccc:7:FA::->L57>
```

Replication to R7, steers packet directly to the node on L57.

Replication segment at R6:

```
Replication segment <R1,T-ID,R6>:
 Replication SID: 2001:db8:cccc:6:FA::
 Replication State:
   R6: <Leaf>
```

Replication segment at R7:

```
Replication segment <R1,T-ID,R7>:
 Replication SID: 2001:db8:cccc:7:FA::
 Replication State:
   R7: <Leaf>
```

When a packet (A,B2) is steered into the SR P2MP Policy at R1 using H.Encaps.Replicate behavior:

- Since R1 is directly connected to R2, R1 sends replicated copy (2001:db8::1, 2001:db8:cccc:2:FA::) (A,B2) to R2 on interface L12.

- R2, as Leaf, removes outer IPv6 header and delivers the payload. R2, as a bud node, also replicates the packet. For replication to R3, R2 sends (2001:db8::1, 2001:db8:cccc:3:FA::) (A,B2) to R3 on interface L23. For replication to R5, R2 sends (2001:db8::1, 2001:db8:cccc:5:FA::) (A,B2) to R5 on interface L25.

- R3 replicates and sends (2001:db8::1, 2001:db8:cccc:6:FA::) (A,B2) to R6 on interface L36.

- R5 replicates and sends (2001:db8::1, 2001:db8:cccc:7:FA::) (A,B2) to R7 on interface L57.

- R6, as Leaf, removes outer IPv6 header and delivers the payload.
- R7, as Leaf, removes outer IPv6 header and delivers the payload.

Authors' Addresses

Daniel Voyer (editor)
Bell Canada
Montreal
CA

Email: daniel.voyer@bell.ca

Clarence Filsfils
Cisco Systems, Inc.
Brussels
BE

Email: cfilsfil@cisco.com

Rishabh Parekh
Cisco Systems, Inc.
San Jose
US

Email: riparekh@cisco.com

Hooman Bidgoli
Nokia
Ottawa
CA

Email: hooman.bidgoli@nokia.com

Zhaohui Zhang
Juniper Networks

Email: zzhang@juniper.net

Network Working Group
Internet-Draft
Intended status: Informational
Expires: October 10, 2021

M. Mishra
S. Santhanam
A. Paramasivam
J. Goh
Cisco Systems
G. Mishra
Verizon Communications Inc. (VZ)

April 8, 2021

PIM Backup Designated Router Procedure
draft-mankamana-pim-bdr-05

Abstract

On a multi-access network, one of the PIM routers is elected as a Designated Router (DR). On the last hop LAN, the PIM DR is responsible for tracking local multicast listeners and forwarding traffic to these listeners if the group is operating in PIM-SM. In this document, we propose a mechanism to elect backup DR on a shared LAN. A backup DR on LAN would be useful for faster convergence. This draft introduces the concept of a Backup Designated Router (BDR) and the procedure to implement it.

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 https://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 October 10, 2021.

Copyright Notice

Table of Contents

1. Introduction

On a multi-access LAN such as an Ethernet, one of the PIM routers is elected as a DR. The PIM DR has two roles in the PIM-SM protocol. On the first hop network, the PIM DR is responsible for registering an active source with the Rendezvous Point (RP) if the group is operating in PIM-SM. On the last hop LAN, the PIM DR is responsible for tracking local multicast listeners and forwarding to these listeners if the group is operating in PIM-SM.

Consider the following last hop LAN in Figure 1:

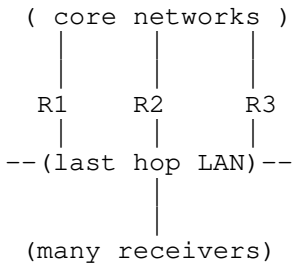


Figure 1: Last Hop LAN

Assume R1 is elected as the Designated Router. According to [RFC4601], R1 will be responsible for forwarding traffic to that LAN on behalf of any local member. In addition to keeping track of IGMP and MLD membership reports, R1 is also responsible for initiating the creation of source and/or shared trees towards the senders or the RPs.

There are multiple reasons for why network could potentially trigger DR re-election. Some of the reasons are

1. R1 going down

2. Access interface towards shared LAN going down

3. Config changed with lower DR priority

When any of above network event occurs, PIM DR re-election would be triggered. When a new DR is elected in shared LAN, new DR would be responsible to build a multicast tree towards source / RP. There are some cases, where traffic is crucial and the operator wants to have minimum traffic loss with DR failure. To address this requirement, this draft introduces a backup DR election procedure which would minimize traffic loss during PIM DR failure.

2. Terminology

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] .

BDR - PIM Backup DR

With respect to PIM, this document follows the terminology that has been defined in [RFC4601] .

3. Applicability and deviation from draft PIM DR Improvement

[I-D.ietf-pim-dr-improvement] defines procedure to solve same problem which was stated in the introduction section of this draft. [I-D.ietf-pim-dr-improvement] introduces new PIM Hello options for election of backup PIM DR.

This draft provides mechanism to elect BDR without using any new PIM Hello.

4. Protocol Specification

4.1. PIM Backup DR (BDR) election procedure

[RFC7761] defines procedure for PIM DR election. PIM DR is elected on interface "I" among all PIM routers for which "I" has received PIM Hello. BDR election follows the exact same procedure and the second best PIM DR on shared LAN to be chosen as BDR on interface "I"

BDR would perform each of the responsibility of PIM DR except it would not forward traffic on shared LAN.

4.2. Existing PIM DR failure

When PIM DR fails, PIM DR re-election is triggered on shared LAN. Since BDR is second best DR in LAN, it MUST take over immediately and MUST start forwarding multicast traffic on shared LAN.

Again on a shared LAN, new BDR would be elected. and current BDR would be the new DR.

4.3. Existing PIM BDR failure

When an existing PIM BDR fails, the shared LAN MUST have BDR re-election using the DR election procedure from [RFC7761].

4.4. New PIM Router addition in network

When a new PIM router is added in shared LAN, It could be either one of the below defined roles.

4.4.1. New PIM router eligible to be PIM DR on shared LAN

When a new PIM router is added in a shared LAN and has the highest PIM DR priority configured, if a new router starts propagating its configured DR priority right away, the existing PIM DR would give up its role. Then there would be potential traffic loss till the new DR

learns about membership states and builds a multicast tree to the source or RP.

To avoid any such traffic loss situation, new PIM router SHOULD send a PIM Hello with priority 0. After 2 (default value, SHOULD have way to configure) PIM Hello interval or IGMP Query Interval (Which ever is higher) it SHOULD start propagating its original configured DR priority.

Even though a new PIM router propagating its priority as 0, it MUST start building a multicast tree towards source / RP, This is So that traffic loss could be minimized once it starts sending Hello with configured DR priority.

For a brief amount of time, there would be multiple copies of flows present in the multicast core, but a user SHOULD be able to configure whether to send hello with 0 priority or a configured priority. Depending on the application tolerance (Traffic loss Vs Extra traffic in core) the operator can choose option whichever is suitable for network.

After a PIM Hello or IGMP Query interval, the network would get stable with only one DR and one BDR.

4.4.2. New PIM router eligible to be PIM BDR on shared LAN

It SHOULD follow the exact same procedure defined in the previous section.

4.4.3. New PIM router is not eligible to be PIM DR or BDR on shared LAN

First a PIM Hello MUST be sent with priority 0. Once it has gotten Hello from other PIM neighbors, it knows that it is not eligible to be PIM DR or BDR. It MUST send configured PIM DR priority immediately. It MUST not wait for next hello interval.

4.5. Initial case, All new PIM router coming up in shared LAN

In this case, initially each of the PIM routers would send Hellos with priorities of 0. If a PIM router receives all Hellos with priorities 0, it MUST send out a Hello with a configured PIM DR priority. Since it is initial startup case, it would take up to one Hello interval to converge.

4.6. Benefit

1. Easy to implement as it uses an existing PIM procedure to elect DR.

2. Does not introduce any new Hello option

5. Compatibility

6. Manageability Considerations

7. IANA Considerations

8. Security Considerations

9. Acknowledgement

The author would like to thank Stig Venaas, Tharak Abraham, Anish Kachinthaya, Anvitha Kachinthaya for helping with original idea.

10. Normative References

[I-D.ietf-pim-dr-improvement]
Zhang, Z., hu, f., Xu, B., and m. mishra, "PIM DR Improvement", draft-ietf-pim-dr-improvement-04 (work in progress), December 2017.

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

[RFC4601] Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, DOI 10.17487/RFC4601, August 2006, <https://www.rfc-editor.org/info/rfc4601>.

[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, <https://www.rfc-editor.org/info/rfc7761>.

Authors' Addresses

Mankamana Mishra
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: mankamis@cisco.com

Sridhar Santhanam
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: sridsant@cisco.com

Aravind Paramasivam
Cisco Systems
821 Alder Drive,
MILPITAS, CALIFORNIA 95035
UNITED STATES

Email: arparama@cisco.com

Joseph Goh
Cisco Systems
SINGAPORE

Email: hocgoh@cisco.com

Gyan S. Mishra
Verizon Communications Inc. (VZ)
13101 Columbia Pike FDC1 Rm 304-D
Silver Spring MD 20904
UNITED STATES

Email: gyan.s.mishra@verizon.com

Internet Engineering Task Force
Internet-Draft
Intended status: Experimental
Expires: 25 August 2021

V. Govindan
Cisco

21 February 2021

PIM Join/ Prune Attributes for LISP Environments using Underlay Multicast
draft-vgovindan-pim-jp-extensions-lisp-00

Abstract

This document specifies an extension to PIM Join/ Prune messages. This document defines one PIM Join/ Prune attribute that support the construction of multicast distribution trees where the root and receivers are located in different Locator/ID Separation Protocol (LISP) sites using underlay IP Multicast. This attribute allows the receiver site to signal the underlay multicast group to the control plane of the root ITR (Ingress Tunnel Router).

Table of Contents

1. Introduction

The construction of multicast distribution trees where the root and receivers are located in different LISP sites [RFC6830] is defined in [RFC6831].

[RFC6831] specifies that (root-EID,G) data packets are to be LISP-encapsulated into (root-RLOC,G) multicast packets. This document defines a TLV that facilitates the construction of trees for (root-RLOC, G).

Specifically, the assignment of the underlay multicast group needs to be done in consonance with the downstream xTR nodes and avoid unnecessary replication or traffic hairpinning.

Since the Receiver RLOC Attribute TLV defined in [RFC8059] only addresses the Ingress Replication case, an additional TLV is defined by this draft to include scenarios where the underlay uses Multicast transport. The TLV definition proposed here complies with the base specification [RFC5384].

This document uses terminology defined in [RFC6830], such as EID, RLOC, ITR, and ETR.

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 RFC 2119 [RFC2119].

2. The case for requiring a new PIM Join/ Prune Extension

When LISP based Multicast trees are built using IP Multicast in the underlay, the mapping between the overlay group address and the underlay group address becomes a very crucial. It is possible that under certain circumstances, differnt subsets of xTRs subscribing to the same overlay multicast stream would be constrained to use different underlay multicast mapping ranges. This definitely involves a trade-off between replication and the flexibility in assigning address ranges and could be required in certain situations as below:

Inter-site PxTR:
When multiple LISP sites are connected through a LISP based transit, the site border node interconnects the site-facing interfaces and the external LISP based core. Under such circumstances, there could be different ranges of multicast group addresses used for building the (S-RLOC, G) trees inside the LISP site and the external LISP based core. This is desired for various reasons:

Other Use-cases:
TBD

Editorial Note: Comments from Stig: There should be some text indicating that the group address used should ideally only be used for LISP encapsulation (if ASM), and perhaps that it is preferrable to use an SSM group. Also, that the group obviously must be a group that the underlay supports/allows. I think it is also worth noting that ideally, different ETRs should request the same group.

3. Receiver ETR Group Address Attribute

3.1. Receiver Group Address Attribute Format

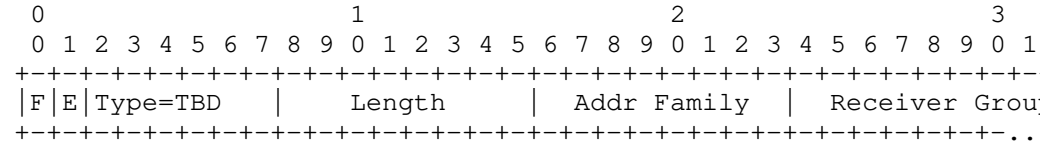


Figure 1

F-bit:
The Transitive bit. Specifies whether this attribute is transitive or non-transitive. MUST be set to zero. This attribute is ALWAYS non-transitive.

E-bit:
End-of-Attributes bit. Specifies whether this attribute is the last. Set to zero if there are more attributes. Set to 1 if this is the last attribute.

Type:
The Receiver Group Attribute type is TBD.

Length:
The length in octets of the attribute value. MUST be set to the length in octets of the receiver group address plus one octet to account for the Address Family field.

Addr Family:
The PIM Address Family of the receiver group as defined in [RFC7761].

Receiver Group:
The Multicast Group address on which the receiver ETR wishes to receive the IP multicast encapsulated flow.

4. Acknowledgements

The authors would like to thank Stig Venaas for his valuable comments.

5. Contributors

Sankaralingam
Cisco

Email: sankt@cisco.com

Amit Kumar
Cisco

Email: kumaram3@cisco.com

6. IANA Considerations

This memo includes the following request to IANA: One new PIM Join/ Prune attribute types have been requested: value TBD for the Receiver Group Attribute.

7. Security Considerations

There is perhaps a new attack vector where an attacker can send a bunch of joins with different group addresses. It may interfere with other multicast traffic if those group addresses overlap. Also, it may take up a lot of resources if replication for thousands of groups are requested. However PIM authentication (?) should come to the rescue here. TBD Since explicit tracking would be done, perhaps it is worth enforcing that for each ETR RLOC (the RLOC used as the source of the overlay join), there should be only one group, whatever is in the last join would override what was there earlier? Or is it to strict to only allow a single group? Might there be reasons to maybe split different LISP payload into different groups in some cases. TBD.

Ed Note: To be addressed - Comments from Stig: Regarding security considerations and PIM authentication. The only solution we have here is to use IP-Sec to sign the J/P messages. I don’t know if anyone has tried to us IPSec between LISP RLOCs. Are there any LISP security mechanisms that would help here for authenticating LISP encapsulated messages between xTRs?

8. 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, <https://www.rfc-editor.org/info/rfc2119>.

[RFC5384] Boers, A., Wijnands, I., and E. Rosen, "The Protocol Independent Multicast (PIM) Join Attribute Format", RFC 5384, DOI 10.17487/RFC5384, November 2008, <https://www.rfc-editor.org/info/rfc5384>.

[RFC6830] Farinacci, D., Fuller, V., Meyer, D., and D. Lewis, "The Locator/ID Separation Protocol (LISP)", RFC 6830, DOI 10.17487/RFC6830, January 2013, <https://www.rfc-editor.org/info/rfc6830>.

[RFC6831] Farinacci, D., Meyer, D., Zwiebel, J., and S. Venaas, "The Locator/ID Separation Protocol (LISP) for Multicast Environments", RFC 6831, DOI 10.17487/RFC6831, January 2013, <https://www.rfc-editor.org/info/rfc6831>.

[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, <https://www.rfc-editor.org/info/rfc7761>.

[RFC8059] Arango, J., Venaas, S., Kouvelas, I., and D. Farinacci, "PIM Join Attributes for Locator/ID Separation Protocol (LISP) Environments", RFC 8059, DOI 10.17487/RFC8059, January 2017, <https://www.rfc-editor.org/info/rfc8059>.

Author's Address

Vengada Prasad Govindan
Cisco

Email: venggovi@cisco.com