

Internet Engineering Task Force (IETF)  
Request for Comments: 7891  
Category: Standards Track  
ISSN: 2070-1721

J. Asghar  
IJ. Wijnands, Ed.  
S. Krishnaswamy  
A. Karan  
Cisco Systems  
V. Arya  
DIRECTV Inc.  
June 2016

## Explicit Reverse Path Forwarding (RPF) Vector

### Abstract

The PIM Reverse Path Forwarding (RPF) Vector TLV defined in RFC 5496 can be included in a PIM Join Attribute such that the RPF neighbor is selected based on the unicast reachability of the RPF Vector instead of the source or Rendezvous Point associated with the multicast tree.

This document defines a new RPF Vector Attribute type such that an explicit RPF neighbor list can be encoded in the PIM Join Attribute, thus bypassing the unicast route lookup.

### Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <http://www.rfc-editor.org/info/rfc7891>.

## Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

## Table of Contents

1. Introduction . . . . .	3
2. Specification of Requirements . . . . .	3
3. Motivation . . . . .	3
4. Use of the PIM Explicit RPF Vector . . . . .	4
5. Explicit RPF Vector Attribute TLV Format . . . . .	5
6. Mixed Vector Processing . . . . .	5
7. Conflicting RPF Vectors . . . . .	5
8. PIM Asserts . . . . .	6
9. Join Suppression . . . . .	6
10. Unsupported Explicit Vector Handling . . . . .	7
11. IANA Considerations . . . . .	7
12. Security Considerations . . . . .	7
13. References . . . . .	8
13.1. Normative References . . . . .	8
13.2. Informative References . . . . .	8
Acknowledgements . . . . .	8
Authors' Addresses . . . . .	9

## 1. Introduction

The procedures in [RFC5496] define how an RPF Vector can be used to influence the path selection in the absence of a route to the source. The same procedures can be used to override a route to the source when it exists. It is possible to include multiple RPF Vectors in the list where each router along the path will perform a unicast route lookup on the first Vector in the attribute list. Once the router owning the address of the RPF Vector is reached, following the procedures in [RFC5496], the RPF Vector will be removed from the attribute list. This will result in a 'loosely' routed path that still depends on unicast reachability to the RPF Vector(s).

In some scenarios, the network administrators don't want to rely on the unicast reachability to the RPF Vector address and want to build a path strictly based on the RPF Vectors. In that case, the RPF Vectors represent a list of directly connected PIM neighbors along the path. For these Vectors, the router would not do a route lookup in the unicast routing table. These Vectors are referred to as 'Explicit' RPF Vector addresses. If a router receiving an Explicit RPF Vector does not have a PIM neighbor matching the Explicit RPF Vector address, it does not fall back to loosely routing the Join. Instead, it could process the packet and store the RPF Vector list so that the PIM Join can be sent out as soon as the neighbor comes up. Since the behavior of the Explicit RPF Vector differs from the 'loose' RPF Vector as defined in [RFC5496], a new attribute called the Explicit RPF Vector is defined.

This document defines a new TLV in the PIM Join Attribute message [RFC5384] for specifying the explicit path.

## 2. Specification of Requirements

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].

## 3. Motivation

Some broadcast video transport networks use a multicast PIM Live-Live resiliency model for video delivery based on PIM Source-Specific Multicast (PIM-SSM) or PIM Any-Source Multicast (PIM-ASM). Live-Live implies using two active, spatially diverse multicast trees to transport video flows from root to leaf multicast routers. The leaf multicast router receives two copies from the PIM multicast core and will replicate one copy towards the receivers [RFC7431].



In comparison, when the procedures specified in [RFC5496] are used, if the R5-R6 link fails, then the Join may be rerouted using the R6-R8-R7 path to reach R5.

## 5. Explicit RPF Vector Attribute TLV Format

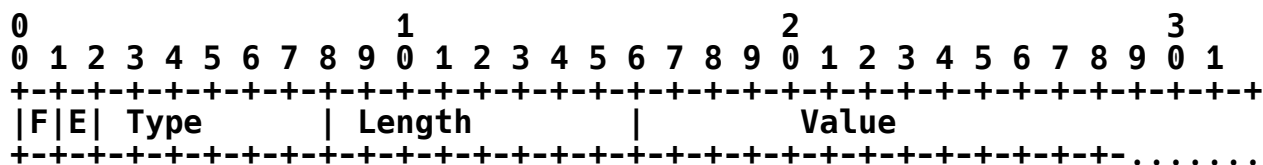


Figure 2

**F bit:** 'Transitive Attribute' bit. The F bit **MUST** be set to 0. Otherwise, there could be loops.

**E bit:** 'End of Attributes' bit. If the E bit is set, then this is the last TLV specified in the list.

**Type:** 4 (Explicit RPF Vector)

**Length:** The length depending on the Address Family (IPv4 or IPv6) of the Encoded-Unicast address.

**Value:** Encoded-Unicast address. This **SHOULD** be a valid IPv4 or IPv6 address of an upstream router.

## 6. Mixed Vector Processing

The Explicit RPF Vector Attribute does not impact or restrict the functionality of other RPF Vector Attributes in a PIM Join. It is possible to mix Vectors of different types such that some part of the tree is explicit and other parts are loosely routed. RPF Vectors are processed in the order in which they are specified.

## 7. Conflicting RPF Vectors

It is possible that a PIM router has multiple downstream neighbors. If for the same multicast route there is an inconsistency between the Explicit RPF Vector lists provided by the downstream PIM neighbor, the procedures as documented in Section 3.3.3 of [RFC5384] apply.

The conflict resolution procedures in Section 3.3.3 of [RFC5384] only apply to attributes of the same Join Attribute type. Join Attributes that have a different type can't be compared because the content of the Join Attribute may have a totally different meaning and/or encoding. This may cause a problem if a mix of Explicit RPF Vectors

(this document) and 'loose' RPF Vectors [RFC5496] is received from two or more downstream routers. The order in which the RPF Vectors are encoded may be different, and/or the combination of RPF Vectors may be inconsistent. The procedures in Section 3.3.3 of [RFC5384] would not resolve the conflict. The following procedures **MUST** be applied to deal with this scenario.

When a PIM Join with a Join Attribute list is received from a downstream neighbor, the router **MUST** verify that the order in which the RPF Vector types appear in the PIM Join Attribute list matches what is stored as the Join Attribute list for reaching the source or Rendezvous Point listed in the PIM Join. Once it is determined that the RPF Vector types on the stack are equal, the content of the RPF Vectors **MUST** be compared ([RFC5384]). If it is determined that there is either a conflict with RPF Vector types or the RPF Vector content, the router uses the RPF Vector stack from the PIM adjacency with the numerically smallest IP address. In the case of IPv6, the link-local address will be used. When two neighbors have the same IP address, either for IPv4 or IPv6, the interface index **MUST** be used as a tie breaker. It's **RECOMMENDED** that the router doing the conflict resolution log a message.

## 8. PIM Asserts

Section 3.3.3 of [RFC5496] specifies the procedures for how to deal with PIM Asserts when RPF Vectors are used. The same procedures apply to the Explicit RPF Vector. There is a minor behavioral difference: the route 'metric' that is included in the PIM Assert should be the route metric of the first Explicit RPF Vector address in the list. However, the first Explicit Vector should always be directly connected, so the metric may likely be zero. The metric will therefore not be a tie breaker in the PIM Assert selection procedure.

## 9. Join Suppression

Section 3.3.4 of [RFC5496] specifies the procedures for how to apply Join Suppression when an RPF Vector Attribute is included in the PIM Join. The same procedure applies to the Explicit RPF Vector Attribute. The procedure **MUST** match against all the Explicit RPF Vectors in the PIM Join before a PIM Join can be suppressed.

## 10. Unsupported Explicit Vector Handling

The F bit MUST be set to 0 in all Explicit RPF Vectors in case the upstream router receiving the Join does not support the TLV. As described in Section 3.3.2 of [RFC5384], routers that do not understand the type of a particular attribute that has the F bit clear will discard it and continue to process the Join.

This processing is particularly important when the routers that do not support the Explicit RPF TLV are identified as hops in the Explicit RPF list because failing to remove the RPF Vectors could cause upstream routers to send the Join back toward these routers causing loops.

As the administrator is manually specifying the path that the Joins need to be sent on, it is recommended that the administrator computes the path to include routers that support the Explicit Vector and check that the state is created correctly on each router along the path. Tools like mtrace can be used for debugging and to ensure that the Join state is setup correctly.

## 11. IANA Considerations

In the "PIM Join Attribute Types" registry, IANA has assigned the value 4 to the Explicit RPF Vector Attribute.

## 12. Security Considerations

Security of the Explicit RPF Vector Attribute is only guaranteed by the security of the PIM packet, so the security considerations for PIM Join packets as described in PIM-SM [RFC7761] apply here. A malicious downstream node can attempt a denial-of-service attack by sending PIM Join packets with invalid addresses listed in the RPF Vector stack with an intent to stop the propagation of the Joins to the correct upstream node. Another denial-of-service attack would be a malicious downstream node targeting all Joins to a specific node with an intent to overload the bandwidth on that node by making it responsible for forwarding multicast traffic for more streams that it can handle. In order to minimize the risk of a denial-of-service attack from forged PIM Join packets with Explicit RPF Vector stack, it should be used within a single trusted management domain.

If a router finds that it cannot use the Vector list due to the next hop router not being a PIM neighbor, it may log an error. Also, if a router is receiving two conflicting Vectors, it may log an error. It is up to the mechanisms that produced the Explicit RPF Vector to ensure that the PIM tree is built correctly and to monitor any error logs.

## 13. References

### 13.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.
- [RFC5384] Boers, A., Wijnands, I., and E. Rosen, "The Protocol Independent Multicast (PIM) Join Attribute Format", RFC 5384, DOI 10.17487/RFC5384, November 2008, <<http://www.rfc-editor.org/info/rfc5384>>.
- [RFC5496] Wijnands, IJ., Boers, A., and E. Rosen, "The Reverse Path Forwarding (RPF) Vector TLV", RFC 5496, DOI 10.17487/RFC5496, March 2009, <<http://www.rfc-editor.org/info/rfc5496>>.
- [RFC7761] Fenner, B., Handley, M., Holbrook, H., Kouvelas, I., Parekh, R., Zhang, Z., and L. Zheng, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", STD 83, RFC 7761, DOI 10.17487/RFC7761, March 2016, <<http://www.rfc-editor.org/info/rfc7761>>.

### 13.2. Informative References

- [MTRACE-V2] Asaeda, H., Meyer, K., and W. Lee, "Mtrace Version 2: Traceroute Facility for IP Multicast", Work in Progress, draft-ietf-mboned-mtrace-v2-13, June 2016.
- [RFC7431] Karan, A., Filsfils, C., Wijnands, IJ., Ed., and B. Decraene, "Multicast-Only Fast Reroute", RFC 7431, DOI 10.17487/RFC7431, August 2015, <<http://www.rfc-editor.org/info/rfc7431>>.

## Acknowledgements

The authors would like to thank Vatsa Kumar, Nagendra Kumar, and Bharat Joshi for their comments on the document.



**Authors' Addresses**

Javed Asghar  
Cisco Systems  
725, Alder Drive  
Milpitas, CA 95035  
United States

Email: [jasghar@cisco.com](mailto:jasghar@cisco.com)

IJsbrand Wijnands (editor)  
Cisco Systems  
De Kleetlaan 6a  
Diegem 1831  
Belgium

Email: [ice@cisco.com](mailto:ice@cisco.com)

Sowmya Krishnaswamy  
Cisco Systems  
3750 Cisco Way  
San Jose, CA 95134  
United States

Email: [sowkrish@cisco.com](mailto:sowkrish@cisco.com)

Apoorva Karan  
Cisco Systems  
3750 Cisco Way  
San Jose, CA 95134  
United States

Email: [apoorva@cisco.com](mailto:apoorva@cisco.com)

Vishal Arya  
DIRECTV Inc.  
2230 E Imperial Hwy  
El Segundo, CA 90245  
United States

Email: [varya@directv.com](mailto:varya@directv.com)