

PIM Working Group
Internet Draft
Intended status: Informational
Expires: January 2009

Rajiv Asati
Mike McBride
Cisco Systems
July 6, 2008

Multicast Routing Blackhole Avoidance
draft-asati-pim-multicast-routing-blackhole-avoid-00.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with [Section 6 of BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/lid-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>

This Internet-Draft will expire on January 6, 2007.

Copyright Notice

Copyright (C) The IETF Trust (2008).

Internet-Draft [draft-asati-pim-multicast-routing-blackhole](#) July 2008

Abstract

This document specifies a mechanism to avoid blackholing of IP Multicast traffic due to the disruption of multicast tree during the time when the RPF neighbor is yet to become the PIM neighbor. Such scenario is possible during the topology change (e.g. link UP) in an IP network that employs PIM-SM (or SSM) as the multicast routing protocol and a unicast routing protocol (including static routing).

Conventions used in this document

In examples, "C:" and "S:" indicate lines sent by the client and server respectively.

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](#) [[RFC2119](#)].

Table of Contents

1.	Introduction.....	3
2.	Problem Details.....	3
2.1.	Root-cause.....	4
3.	Solution.....	5
3.1.	Solution Detail.....	6
3.2.	Make-before-Break.....	7
4.	Other Solutions.....	8
5.	Interoperability Considerations.....	8
6.	Security Considerations.....	8
7.	IANA Considerations.....	8
8.	Conclusions.....	9
9.	Acknowledgments.....	9
10.	References.....	10
10.1.	Normative References.....	10
10.2.	Informative References.....	10
	Author's Addresses.....	10
	Intellectual Property Statement.....	10
	Disclaimer of Validity.....	11

Internet-Draft [draft-asati-pim-multicast-routing-blackhole](#) July 2008

1. Introduction

A Multicast distribution tree, as identified by the corresponding multicast route e.g. (S,G), (*,G), provides a pre-determined path for distributing the multicast traffic from the source (or RP) to the receivers through a set of routers.

PIM-SM [[RFC4601](#)] is a multicast routing protocol that uses the information from the underlying unicast routing information base (or its derivative) to determine the reverse path for the multicast distribution tree (be it a source tree (S,G) or shared tree (*,G)) and establishes the tree on that path. Specifically, the unicast routing information base (or its derivative) information is utilized to determine the next-hop neighbor and interface, commonly referred to as RPF neighbor and RPF interface respectively in PIM-SM [[RFC4601](#)], for the source of (S,G) or RP of (*,G).

PIM-SM [[RFC4601](#)] specifies that the PIM Join/Prune message for a multicast tree must be sent to the RPF neighbor on the RPF interface to join or prune a multicast tree. It also specifies that the PIM Join/Prune message must be sent to or received from only PIM neighbors. This means that the PIM neighbor relationship must be established with the RPF neighbor on the RPF interface prior to transmitting the PIM Join/Prune message after the link UP event. This ensures that the PIM neighbor becomes capable of processing the PIM Join/Prune messages and joining/pruning a multicast distribution tree.

However, such assumption also opens up the possibility of multicast traffic getting blackholed for brief time period after the link UP event when the PIM neighbor relationship MAY take a little longer to get established. This document describes this problem and proposes a solution for that.

2. Problem Details

Multicast distribution trees built using PIM-SM [[RFC4601](#)] in an IP network may experience an outage for brief period following the link UP event. The figure 1 and 2 below are used to explain this problem.

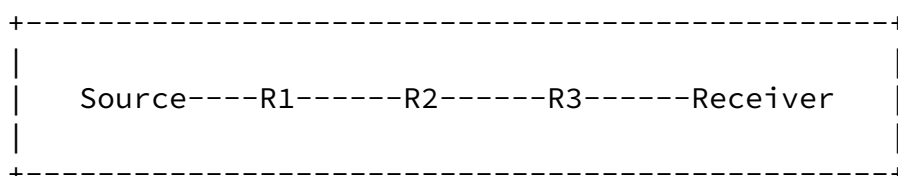


Figure 1 Topology prior to R1-R3 link Up

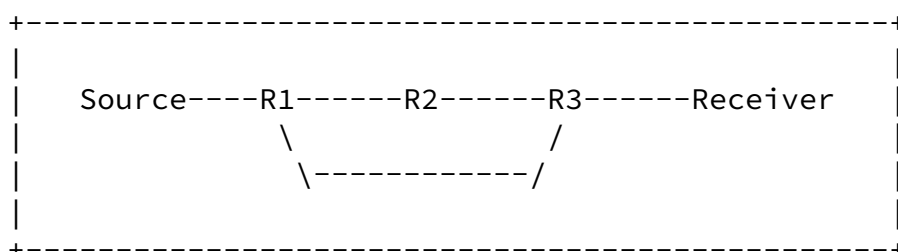


Figure 2 Topology after R1-R3 link Up

After the (R1-R3) link Up event, the unicast routing may likely converge quite fast and update the unicast routing information base to make use of that link. This may trigger the RPF neighbor calculation at router R3 for the particular multicast route(s). The determination of new RPF neighbor (R1, say) for a particular multicast route may further trigger an update of the multicast route in the multicast routing information base. This further triggers the PIM module to send PIM Prune message to the current RPF neighbor (R2, say) and PIM Join message to the new RPF neighbor (R1, say).

However, per PIM-SM [[RFC4601](#)], if the RPF neighbor is not yet the PIM

neighbor, then the PIM Join should not be sent to that neighbor. Even if the PIM Join was sent (by R3, say), then the upstream router (R1, say) may not process the PIM Join/Prune message until the downstream router is known as the PIM neighbor. This means that the multicast tree may be disrupted on R1-R3 link for some time until PIM neighbor relationship is established on R1-R3 link and PIM Join is sent and processed.

[2.1.](#) Root-cause

Firstly, after the link UP event, the PIM neighbor adjacency establishment on a link may take longer than the total time taken to

Asati & McBride

Expires January 6, 2009

[Page 4]

Internet-Draft [draft-asati-pim-multicast-routing-blackhole](#) July 2008

establish the unicast routing neighbor adjacency on that link, determine the new RPF neighbor, if any, and update the corresponding multicast route entry(s). The larger time-period may be due to any of the reasons such as -

- 1) PIM is misconfigured or not configured.
- 2) PIM Hellos are not exchanged immediately after the link UP.
- 3) PIM Hello is sent, but not received after the link UP.
- 4) PIM module experienced a software issue.

Secondly, a multicast routing entry is updated with the new RPF neighbor information as soon as that information becomes available after the unicast routing convergence. There is no check for the RPF neighbor being the PIM neighbor prior to this update. This is really what causes the disruption in multicast tree.

[3.](#) Solution

It may be somewhat obvious from [section 2.1](#) that one may attempt to solve the 1st problem, but it is difficult to solve for all of the scenarios. However, it is very much possible to sufficiently solve

the 2nd problem, and that will automatically bypass the 1st one.

The solution is somewhat simple and straight-forward - replace the current RPF neighbor with the new RPF neighbor for a multicast routing entry ONLY if the new RPF neighbor is determined to be the PIM neighbor. This means that the logic is infused to not propagate the new RPF neighbor information to the multicast routing information base unless the RPF neighbor is listed in the PIM neighbor database. If the RPF neighbor is not listed in the PIM neighbor database, then the PIM adjacency establishment procedure i.e. send PIM hello etc. must be initiated.

This means that the forwarding mroute entry may be left intact until the PIM neighbor adjacency is established with the RPF neighbor on the RPF interface. While this may keep the multicast distribution tree from taking advantage of the changed topology i.e. new link for brief time period, it ensures that the multicast distribution tree is not disrupted unnecessarily and the multicast traffic is not blackholed.

An advantage with this solution is that the proposed changes are local to a router.

[3.1.](#) Solution Detail

One may construct the state machine to be the modified as following (please refer to figure2) -

After the (R1-R3) link Up event, the unicast routing convergence triggers the RPF neighbor calculation at router R3 for the particular multicast route(s). After the RPF neighbor is determined, a check is inserted whether the RPF neighbor is also the PIM neighbor on R1-R3 link by looking up the PIM neighbor database

If such a check returns a positive match, then the corresponding multicast route in the multicast routing information base is(are) updated with the new RPF neighbor (R1, say) information. This further triggers the PIM module to send PIM Join message to the new RPF neighbor (R1, say), and PIM Prune message to the current RPF neighbor (R2, say).

If such a check returns a negative match, then the multicast route in the multicast routing information base is(are) NOT updated with the new RPF neighbor (R1, say) information, and no PIM Join/Prune messages are generated as a result. This results in the multicast distribution tree continuing to use the old RPF neighbor information and avoiding the multicast traffic blackholing.

The state machine may continue to check for RPF neighbor being the PIM neighbor either regularly or wait for the trigger, so that when the RPF neighbor indeed becomes the PIM neighbor, the check returns a positive match for the multicast routes to be updated as explained above.

In summary, the multicast routing sequence would be as follows -

- 1) New RPF neighbor is determined after the unicast routing convergence
- 2) Check whether the new RPF neighbor is also PIM neighbor on the RPF interface

- 3) If it is not, then initiate the PIM neighbor establishment procedure by sending PIM Hellos etc. on the new RPF interface
- 4) If it is, then update the multicast forwarding entry by replacing the old RPF neighbor (and interface) information with the new RPF neighbor (and interface) information
- 5) Send the PIM Join message for the related multicast routes (S,G or *,G).
- 6) Send the PIM Prune message to the old RPF neighbor for the related multicast routes (S,G or *,G)

[3.2](#). Make-before-Break

This solution also paves the way for true make-before-break behavior

for multicast forwarding if the PIM join for a particular mroute (*,G or S,G) is sent to the new RPF neighbor after establishing the PIM neighbor relationship, but before updating the related multicast forwarding entry. This increases the likelihood that the upstream RPF neighbor would have updated its multicast forwarding entry and started sending the multicast traffic downstream. Of course, the traffic would not get forwarded by the downstream router (thanks to RPF check failure) until the multicast forwarding entry is updated with the new RPF neighbor. This almost ensures that the multicast traffic is available on the new path before switching the RPF interface of the multicast forwarding entry.

The multicast routing sequence for make-before-break can be summarized as below -

- 1) New RPF neighbor is determined after the unicast routing convergence
- 2) Check whether the new RPF neighbor is also PIM neighbor on the RPF interface
- 3) If it is not, then initiate the PIM neighbor establishment procedure by sending PIM Hellos etc. on the new RPF interface
- 4) If it is, then send the PIM Join message for the related multicast routes (S,G or *,G).

- 5) Update the multicast forwarding entry by replacing the old RPF neighbor (and interface) information with the new RPF neighbor (and interface) information
- 6) Send the PIM Prune message to the old RPF neighbor for the related multicast routes (S,G or *,G)

It is envisioned that an implementation may provide a configuration option to enable the make-before-break functionality. Also note that this section refers to [section 4.5.7 of RFC4601](#).

[4. Other Solutions](#)

There are number of other approaches to solve this issue, however, they all attempt to solve the 1st problem as explained in the [section 2.1](#).

1. Ignore the PIM hello - This doesn't solve the issue completely. What if the PIM module is malfunctioning on the upstream router.
2. Triggered PIM Hello - This doesn't solve the issue completely, what if the PIM hellos are not received by the remote router. Speeding up Hellos may also penalize the processing resources at the router.
3. Multi-topology - A non-PIM solution that shifts the problem to a different topology, which becomes responsible for maintaining the PIM operation.

[5. Interoperability Considerations](#)

This mechanism introduces no interoperability issue since the changes are local to the router.

[6. Security Considerations](#)

None.

[7. IANA Considerations](#)

None.

[8. Conclusions](#)

IPTV deployment using multicast for Broadcast Video delivery desires multicast sub-second convergence and resiliency since the broadcast video such as MPEG video is loss intolerant. Hence, it is imperative to eliminate any unnecessary multicast traffic outage (blackholing) during the IP network topology changes such as Link UP event. The root-cause of the blackholing problem, as clarified in this document,

lies in the PIM protocol.

Although there are non-PIM solutions, having a PIM based solution that is simple, straight-forward, easier to implement and that requires no interoperability is the goal of this document. The simple change to the PIM state machinery, as proposed in this document, would prevent blackholing of multicast data.

[9.](#) Acknowledgments

TBD.

This document was prepared using 2-Word-v2.0.template.dot.

[10.](#) References

[10.1.](#) Normative References

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

[RFC4601] Fenner B., Handley M., Holbrook H., and Kouvelas I., "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", [RFC 2601](#), August 2006.

[10.2](#). Informative References

Author's Addresses

Rajiv Asati
Cisco Systems, 7025-6 Kit Creek Rd, RTP, NC, 27709-4987
Email: rajiva@cisco.com

Mike McBride
Cisco Systems, 121 Theory, IRVINE, CA, 92617-3045
Email: mmcbride@cisco.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in [BCP 78](#) and [BCP 79](#).

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The IETF Trust (2008).

This document is subject to the rights, licenses and restrictions contained in [BCP 78](#), and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.

