Network Working Group
Internet-Draft
Intended status: Experimental
Expires: September 18, 2016

IJ. Wijnands
S. Venaas
Cisco Systems, Inc.
M. Brig
Aegis BMD Program Office
A. Jonasson
Swedish Defence Material Administration (FMV)

March 17, 2016

PIM flooding mechanism and source discovery
draft-ietf-pim-source-discovery-bsr-04

Abstract

PIM Sparse-Mode uses a Rendezvous Point and shared trees to forward multicast packets from new sources. Once last hop routers receive packets from a new source, they may join the Shortest Path Tree for the source for optimal forwarding. This draft defines a new protcol that provides a way to support PIM Sparse Mode (SM) without the need for PIM registers, RPs or shared trees. Multicast source information is flooded throughout the multicast domain using a new generic PIM flooding mechanism. This allows last hop routers to learn about new sources without receiving initial data packets.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

PIM Sparse-Mode uses a Rendezvous Point (RP) and shared trees to forward multicast packets to Last Hop Routers (LHR). After the first packet is received by a LHR, the source of the multicast stream is learned and the Shortest Path Tree (SPT) can be joined. This draft defines a new protocol that provides a way to support PIM Sparse Mode (SM) without the need for PIM registers, RPs or shared trees. Multicast source information is flooded throughout the multicast domain using a new generic PIM flooding mechanism. This mechanism is defined in this document, and is modeled after the Bootstrap Router protocol [RFC5059]. By removing the need for RPs and shared trees,

the PIM-SM procedures are simplified, improving router operations, management and making the protocol more robust. Also the data packets are only sent on the SPTs, providing optimal forwarding.

1.1. Conventions used in this document

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].

1.2. Terminology

RP: Rendezvous Point.

BSR: Bootstrap Router.

RPF: Reverse Path Forwarding.

SPT: Shortest Path Tree.

FHR: First Hop Router, directly connected to the source.

LHR: Last Hop Router, directly connected to the receiver.

SG Mapping: Multicast source to group mapping.

SG Message: A PIM message containing SG Mappings.

2. Testing and deployment experiences

A prototype of this specification has been implemented and there has been some limited testing in the field. The prototype was tested in a network with low bandwidth radio links. In this network with frequent topology changes and link or router failures PIM-SM with RP election is found to be too slow. With PIM-DM issues were observed with new multicast sources starving low bandwidth links even when there are no receivers, in some cases such that there were no bandwidth left for prune message. For the tests, all routers were configured to send PFP-SA for directly connected source and to cache received announcements. Applications such as SIP with multicast subscriber discovery, multicast voice conferencing, position tracking and NTP were successfully tested. The tests went quite well. Packets were rerouted as needed and there were no unnecessary forwarding of packets. Ease of configuration was seen as a plus.

3. A generic PIM flooding mechanism

The Bootstrap Router protocol (BSR) [RFC5059] is a commonly used protocol for distributing dynamic Group to RP mappings in PIM. It is responsible for flooding information about such mappings throughout a PIM domain, so that all routers in the domain can have the same information. BSR as defined, is only able to distribute Group to RP mappings. We are defining a more generic mechanism that can flood any kind of information throughout a PIM domain. It is not necessarily a domain though, it depends on the administrative boundaries being configured. The forwarding rules are identical to BSR, except that there is no BSR election and that one can control whether routers should forward messages of unsupported types. For some types of information it is quite useful that it can be distributed without all routers having to support the particular type, while there may also be types where it is necessary for every single router to support it. The protocol includes an originator address which is used for RPF checking to restrict the flooding, just like BSR. Just like BSR it is also sent hop by hop. Note that there is no built in election mechanism as in BSR, so there can be multiple originators. It is still possible to add such an election mechanism on a type by type bases if this protocol is used in scenarios where this is desirable. We include a type field, which can allow boundaries to be defined, and election to take place, independently per type. We call this protocol the PIM Flooding Protocol (PFP).

3.1. PFP message format

```
        0                   1                   2                   3
       0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |PIM Ver| Type  |N|  Reserved   |           Checksum            |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            Originator Address (Encoded-Unicast format)        |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            PFP Type           |         Reserved            |U|
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          Type 1               |          Length 1             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                            Value 1                            |
      |                               .                               |
      |                               .                               |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                               .                               |
      |                               .                               |
      |          Type n               |          Length n             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                            Value n                            |
      |                               .                               |
      |                               .                               |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PIM Version: Reserved, Checksum Described in [RFC7761].

Type: PIM Message Type. Value (pending IANA) for a PFP message.

[N]o-Forward bit: When set, this bit means that the PFP message is not to be forwarded. This is irrespective of the value of the Unsupported-No-Forwarding bit defined below.

Originator Address: The address of the router that originated the message. This can be any address assigned to the originating router, but MUST be routable in the domain to allow successful forwarding (just like BSR address). The format for this address is given in the Encoded-Unicast address in [RFC7761].

PFP Type: There may be different sub protocols or different uses for this generic protocol. The PFP Type specifies which sub protocol it is used for.

[U]nsupported-No-Forwarding bit: When the No-Forward bit defined above is not set, whether to forward the message depends on whether the PFP type is supported and the setting of the Unsupported-No-Forwarding bit. Some sub protocols may require that each router do some processing of the contents and not simply forward the message. When Unsupported-No-Forwarding bit is set, a

router MUST NOT forward the message when the PFP Type is unsupported. When it is not set, a router MUST forward the message when possible. If the PFP Type is supported, then the specification of that type will specify how to handle the message, including whether the message should be forwarded.

Type 1..n: A message contains one or more TLVs, in this case n TLVs. The Type specifies what kind of information is in the Value. Note that the Type space is shared between all PFP types. Not all types make sense for all PFP types though.

Length 1..n: The length of the the value field.

Value 1..n: The value associated with the type and of the specified length.

3.2. Processing PFP messages

A router that receives an PFP message MUST perform the initial checks specified here. If the checks fail, the message MUST be dropped. An error MAY be logged, but otherwise the message MUST be dropped silently. If the checks pass, the contents is processed according to the PFP type if supported. If the type is unsupported it may still be forwarded if neither the No-Forward bit nor the Unsupported-No-Forwarding bit are set.

3.2.1. Initial checks

In order to do further processing, a message MUST meet the following requirements. The message MUST be from a directly connected neighbor for which we have active Hello state, and it MUST have been sent to the ALL-PIM-ROUTERS group. Also, the interface MUST NOT be an administrative boundary for the message's PFP type. If No-Forward is not set, it MUST have been sent by the RPF neighbor towards the router that originated the message. If No-Forward is set, we MUST have restarted within 60 seconds. In pseudo-code the algorithm is as follows:

```
              if ((DirectlyConnected(PFP.src_ip_address) == FALSE) OR
                  (we have no Hello state for PFP.src_ip_address) OR
                  (PFP.dst_ip_address != ALL-PIM-ROUTERS) OR
                  (Incoming interface is admin boundary for PFP.type)) {
                   drop the message silently, optionally log error.
              }
              if (PFP.no_forward_bit == 0) {
                  if (PFP.src_ip_address !=
                      RPF_neighbor(PFP.originator_ip_address)) {
                      drop the message silently, optionally log error.
                  }
              } else if (more than 60 seconds elapsed since startup)) {
                  drop the message silently, optionally log error.
              }
```

3.2.2. Processing messages of supported PFP type

When the message is received, the initial checks above must be performed. If it passes the checks, then we continue as follows. If the PFP type is supported by the implementation, the processing and potential forwarding is done according to the specification for that PFP type. If the PFP type specification does not specify any particular forwarding rules, the message is forwarded out of all interfaces with PIM neighbors (including the interface it was received on).

3.2.3. Processing messages of unsupported PFP type

When the message is received, the initial checks above must be performed. If it passes the checks, then we continue as follows. If the PFP type is unsupported, the message MUST be dropped if the Unsupported-No-Forwarding bit is set. If the bit is not set, the message is forwarded out of all interfaces with PIM neighbors (including the interface it was received on).

4. Distributing Source to Group Mappings

We want to provide information about active multicast sources throughout a PIM domain by making use of the generic flooding mechanism defined in the previous section. We request PFP Type 0 to be assigned for this purpose. We call a message with PFP Type 0 an SG Message. We also define a PFP TLV which we request to be type 0. How this TLV is used with PFP Type 0 is defined in the next section. Other PFP Types may specify the use of this TLV for other purposes. For PFP Type 0 the U-bit MUST NOT be set. This means that routers not supporting PFP Type 0 would still forward the message.

4.1. Group Source Holdtime TLV

```
        0                   1                   2                   3
       0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |          Type = 0               |          Length             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |              Group Address (Encoded-Group format)             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            Src Count          |        Src Holdtime           |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            Src Address 1 (Encoded-Unicast format)             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            Src Address 2 (Encoded-Unicast format)             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |                               .                               |
      |                               .                               |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
      |            Src Address m (Encoded-Unicast format)             |
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Type: This TLV has type 0.

Length: The length of the value.

Group Address: The group we are announcing sources for. The format for this address is given in the Encoded-Group format in [RFC7761].

Src Count: How many unicast encoded sources address encodings follow.

Src Holdtime: The Holdtime (in seconds) for the corresponding source(s).

Src Address: The source address for the corresponding group. The format for these addresses is given in the Encoded-Unicast address in [RFC7761].

4.2. Originating SG messages

An SG Message, that is a PFP message of Type 0, MAY contain one or more Group Source Holdtime TLVs. This is used to flood information about active multicast sources. Each FHR that is directly connected to an active multicast source originates SG BSR messages. How a multicast router discovers the source of the multicast packet and when it considers itself the FHR follows the same procedures as the registering process described in [RFC7761]. When a FHR has decided

that a register needs to be sent per [RFC7761], the SG is not registered via the PIM SM register procedures, but the SG mapping is included in an SG message. Note, only the SG mapping is distributed in the message, not the entire packet as would have been done with a PIM register. The router originating the SG messages includes one of its own addresses in the originator field. Note that this address SHOULD be routeable due to RPF checking. The SG messages are periodically sent for as long as the multicast source is active, similar to how PIM registers are periodically sent. The default announcement period is 60 seconds, which means that as long as the source is active, it is included in an SG message originated every 60 seconds. The holdtime for the source is by default 210 seconds. Other values MAY be configured, but the holdtime MUST be either zero, or larger than the announcement period. It is RECOMMENDED to be 3.5 times the announcement period. A source MAY be announced with a holdtime of zero to indicate that the source is no longer active.

4.3. Processing SG messages

A router that receives an SG message SHOULD parse the message and store the SG mappings with a holdtimer started with the advertised holdtime for that group. For each group that has directly connected receivers, this router SHOULD send PIM (S,G) joins for all the SG mappings advertised in the message for the group. The SG mappings are kept alive for as long as the holdtimer for the source is running. Once the holdtimer expires a PIM router SHOULD send a PIM (S,G) prune to remove itself from the tree. Note that a holdtime of zero has a special meaning. It is to be treated as if the source just expired, causing a prune to be sent and state to be removed. Source information MUST NOT be removed due to it being omitted in a message. For instance, if there are a large number of sources for a group, there may be multiple SG messages for the same group, each message containing a different list of sources.

4.4. The first packets and bursty sources

The PIM register procedure is designed to deliver Multicast packets to the RP in the absence of a Shortest Path Tree (SPT) from the RP to the source. The register packets received on the RP are decapsulated and forwarded down the shared tree to the LHRs. As soon as an SPT is built, multicast packets would flow natively over the SPT to the RP or LHR and the register process would stop. The PIM register process ensures packet delivery until an SPT is in place reaching the FHR. If the packets were not unicast encapsulated to the RP they would be dropped by the FHR until the SPT is setup. This functionality is important for applications where the initial packet(s) must be received for the application to work correctly. Another reason would be for bursty sources. If the application sends out a multicast

packet every 4 minutes (or longer), the SPT is torn down (typically after 3:30 minutes of inactivity) before the next packet is forwarded down the tree. This will cause no multicast packet to ever be forwarded. A well behaved application should be able to deal with packet loss since IP is a best effort based packet delivery system. But in reality this is not always the case.

With the procedures defined in this document the packet(s) received by the FHR will be dropped until the LHR has learned about the source and the SPT is built. That means for bursty sources or applications sensitive for the delivery of the first packet this solution would not be very applicable. This solution is mostly useful for applications that don't have strong dependency on the initial packet(s) and have a fairly constant data rate, like video distribution for example. For applications with strong dependency on the initial packet(s) we recommend using PIM Bidir [RFC5015] or SSM [RFC4607]. The protocol operations are much simpler compared to PIM SM, it will cause less churn in the network and both guarantee best effort delivery for the initial packet(s).

Another solution to address the problems described above is documented in [I-D.ietf-magma-msnip]. This proposal allows for a host to tell the FHR its willingness to act as Source for a certain Group before sending the data packets. LHRs have time to join the SPT before the host starts sending which would avoid packet loss. The SG mappings announced by [I-D.ietf-magma-msnip] can be advertised directly in SG messages, allowing a nice integration of both proposals. The life time of the SPT is not driven by the liveliness of Multicast data packets (which is the case with PIM SM), but by the announcements driven via [I-D.ietf-magma-msnip]. This will also prevent packet loss due to bursty sources.

4.5. Resiliency to network partitioning

In a PIM SM deployment where the network becomes partitioned, due to link or node failure, it is possible that the RP becomes unreachable to a certain part of the network. New sources that become active in that partition will not be able to register to the RP and receivers within that partition are not able to receive the traffic. Ideally you would want to have a candidate RP in each partition, but you never know in advance which routers will form a partitioned network. In order to be fully resilient, each router in the network may end up being a candidate RP. This would increase the operational complexity of the network.

The solution described in this document does not suffer from that problem. If a network becomes partitioned and new sources become active, the receivers in that partitioned will receive the SG

Mappings and join the source tree. Each partition works independently of the other partition(s) and will continue to have access to sources within that partition. As soon as the network heals, the SG Mappings are re-flooded into the other partition(s) and other receivers can join to the newly learned sources.

5. Security Considerations

The security considerations are mainly similar to what is documented in [RFC5059]. It is a concern that rogue devices can inject packets that are flooded throughout a domain. PFP packets must only be accepted from a PIM neighbor. Deployments may use mechanisms for authenticating PIM neighbors. For PFP-SA it is an issue that injected packets from a rogue device could send SG mappings for a large number of source addresses, causing routers to use memory storing these mappings, and also if they have interest in the groups, build Shortest Path Trees for sources that are not actually active.

6. IANA considerations

This document requires the assignment of a new PIM Protocol type for the PIM Flooding Protocol (PFP). IANA is also requested to create a registry for PFP Types with type 0 assigned to "Source-Group Message". IANA is also requested to create a registry for PFP TLVs, with type 0 assigned to the "Source Group Holdtime" TLV. Assignments for both registries are to be made according to the policy "IETF Review" as defined in [RFC5226].

7. Acknowledgments

The authors would like to thank Arjen Boers for contributing to the initial idea and Yiqun Cai for his comments on the draft.

8. References

8.1. 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>.

[RFC5059] Bhaskar, N., Gall, A., Lingard, J., and S. Venaas, "Bootstrap Router (BSR) Mechanism for Protocol Independent Multicast (PIM)", RFC 5059, DOI 10.17487/RFC5059, January 2008, <http://www.rfc-editor.org/info/rfc5059>.

[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>.

8.2. Informative References

[RFC4607] Holbrook, H. and B. Cain, "Source-Specific Multicast for IP", RFC 4607, DOI 10.17487/RFC4607, August 2006, <http://www.rfc-editor.org/info/rfc4607>.

[RFC5015] Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)", RFC 5015, DOI 10.17487/RFC5015, October 2007, <http://www.rfc-editor.org/info/rfc5015>.

[RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 5226, DOI 10.17487/RFC5226, May 2008, <http://www.rfc-editor.org/info/rfc5226>.

[I-D.ietf-magma-msnip]
Fenner, B., Haberman, B., Holbrook, H., Kouvelas, I., and S. Venaas, "Multicast Source Notification of Interest Protocol (MSNIP)", draft-ietf-magma-msnip-06 (work in progress), March 2011.

Authors' Addresses

IJsbrand Wijnands
Cisco Systems, Inc.
De kleetlaan 6a
Diegem 1831
Belgium

Email: ice@cisco.com

Stig Venaas
Cisco Systems, Inc.
Tasman Drive
San Jose CA 95134
USA

Email: stig@cisco.com

Michael Brig
Aegis BMD Program Office
17211 Avenue D, Suite 160
Dahlgren VA 22448-5148
USA

Email: michael.brig@mda.mil

Anders Jonasson
Swedish Defence Material Administration (FMV)
Loennvaegen 4
Vaexjoe 35243
Sweden

Email: anders@jomac.se