Network Working Group	D. Farinacci
Internet-Draft	IJ. Wijnands
Intended status: Experimental	S. Venaas
Expires: April 28, 2011	cisco Systems
	M. Napierala
	AT&T Labs
	October 25, 2010

A Reliable Transport Mechanism for PIM
draft-ietf-pim-port-04.txt

Abstract

This draft describes how a reliable transport mechanism can be used by the PIM protocol to optimize CPU and bandwidth resource utilization by eliminating periodic Join/Prune message transmission. This draft proposes a modular extension to PIM to use either the TCP or SCTP transport protocol.

Status of this Memo

Copyright Notice

Table of Contents

1. Introduction

TOC

The goals of this specification are:

* To create a simple incremental mechanism to provide reliable PIM message delivery in PIM version 2 for use with PIM Sparse-Mode [RFC4601] (Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)," August 2006.) (including Source-Specific Multicast) and Bidirectional PIM [RFC5015] (Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)," October 2007.).

- The reliable transport mechanism will be used for Join-Prune message transmission only.

- When a router supports this specification, it need not use the reliable transport mechanism with every neighbor. That is, negotiation on a per neighbor basis will occur.

The explicit non-goals of this specification are:

- Changes to the PIM message formats as defined in [RFC4601] (Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, “Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised),” August 2006.).

- Provide support for automatic switching between Datagram mode and Transport mode. Two routers that are PIM neighbors on a link will always use Transport mode if and only if both have Transport mode enabled.

This document will specify how periodic Join/Prune message transmission can be eliminated by using TCP [RFC0793] (Postel, J., “Transmission Control Protocol,” September 1981.) or SCTP [RFC4960] (Stewart, R., “Stream Control Transmission Protocol,” September 2007.) as the reliable transport mechanism for Join/Prune messages.
This specification enables greater scalability in terms of control traffic overhead. However, for routers connected to multi-access links that comes at the price of increased control plane state overhead and the control plane overhead required to maintain this state.
In many existing and emerging networks, particularly wireless and mobile satellite systems, link degradation due to weather, interference, and other impairments can result in temporary spikes in the packet loss. In these environments, periodic PIM joining can cause join latency when messages are lost causing a retransmission only 60 seconds later. By applying a reliable transport, a lost join is retransmitted rapidly. Furthermore, when the last user leaves a multicast group, any lost prune is similarly repaired and the multicast stream is quickly removed from the wireless/satellite link. Without a reliable transport, the multicast transmission could otherwise continue until it timed out, roughly 3 minutes later. As network resources are at a premium in many of these environments, rapid termination of the multicast stream is critical to maintaining efficient use of bandwidth.

1.1. Requirements Notation

TOC

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] (Bradner, S., “Key words for use in RFCs to Indicate Requirement Levels,” March 1997.).

1.2. Definitions

TOC

PORT: Stands for PIM Over Reliable Transport. Which is the short form for describing the mechanism in this specification where PIM can use the TCP or SCTP transport protocol.

Periodic Join/Prune message: A Join/Prune message sent periodically to refresh state.

Incremental Join/Prune message: A Join/Prune message sent as a result of state creation or deletion events. Also known as a triggered message.

Native Join/Prune message: A Join/Prune message which is carried with an IP protocol type of PIM.

PORT Join/Prune message: A Join/Prune message using TCP or SCTP for transport.

Datagram Mode: The current procedures PIM uses by encapsulating Join/Prune messages in IP packets sent either triggered or periodically.

PORT Mode: Procedures used by PIM defined in this specification for sending Join/Prune messages over the TCP or SCTP transport layer.

2. Protocol Overview

TOC

PIM Over Reliable Transport (PORT) is a simple extension to PIMv2 for refresh reduction of PIM Join/Prune messages. It involves sending incremental rather than periodic Join/Prune messages over a TCP/SCTP connection between PIM neighbors.
PORT only applies to PIM Sparse-Mode [RFC4601] (Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, “Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised),” August 2006.) and Bidirectional PIM [RFC5015] (Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, “Bidirectional Protocol Independent Multicast (BIDIR-PIM),” October 2007.) Join/Prune messages.
This document does not restrict PORT to any specific link types. However, the use of PORT on e.g. multi-access LANs with many PIM

neighbors should be carefully evaluated. This due to the fact that there may be a full mesh of PORT connections, and that explicit tracking of all PIM PORT routers is required.

PORT can be incrementally used on a link between PORT capable neighbors. Routers which are not PORT capable can continue to use PIM in Datagram Mode. PORT capability is detected using new PORT Capable PIM Hello Options.

Once PORT is enabled on an interface and a PIM neighbor also announces that it is PORT enabled, only PORT Join/Prune messages will be used. That is, only PORT Join/Prune messages are accepted from, and sent to, that particular neighbor. Native Join/Prune messages may still be used for other neighbors.

PORT Join/Prune messages are sent using a TCP/SCTP connection. When two PIM neighbors are PORT enabled, both for TCP or both for SCTP, they will immediately, or on-demand, establish a connection. If the connection goes down, they will again immediately, or on-demand, try to reestablish the connection. No Join/Prune messages (neither Native nor PORT) are sent while there is no connection.

When PORT is used, only incremental Join/Prune messages are sent from downstream routers to upstream routers. As such, downstream routers do not generate periodic Join/Prune messages for state for which the RPF neighbor is PORT-capable.

For Joins and Prunes, which are received over a TCP/SCTP connection, the upstream router does not start or maintain timers on the outgoing interface entry. Instead, it keeps track of which downstream routers have expressed interest. An interface is deleted from the outgoing interface list only when all downstream routers on the interface, no longer wish to receive traffic.

There is no change proposed for the PIM Join/Prune packet format. However, for Join/Prune messages sent over TCP/SCTP connections, no IP Header is included. The message begins with the PIM common header, followed by the Join/Prune message. See section Section 5 (Common Header Definition) for details on the common header.

---

3. New PIM Hello Options TOC

3.1. PIM over the TCP Transport Protocol TOC

Option Type: PIM-over-TCP Capable

```
 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 = 27           |         Length = X + 8        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    TCP Connection ID AFI      |          Reserved     |  Exp  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       TCP Connection ID                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Interface ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Allocated Hello Type values can be found in [HELLO-OPT] (IANA, “PIM Hello Options,” March 2007.).
When a router is configured to use PIM over TCP on a given interface, it MUST include the PIM-over-TCP Capable hello option in its Hello messages for that interface. If a router is explicitly disabled from using PIM over TCP it MUST NOT include the PIM-over-TCP Capable hello option in its Hello messages. When the router cannot setup a TCP connection, it will refrain from including this option.
Implementations may provide a configuration option to enable or disable PORT functionality. We recommend that this capability be disabled by default.

Length: In bytes for the value part of the Type/Length/Value encoding. Where X is 4 bytes if AFI of value 1 (IPv4) is used and 16 bytes when AFI of value 2 (IPv6) is used [AFI] (IANA, “Address Family Indicators (AFIs),” February 2007.).

TCP Connection ID AFI: The AFI value to describe the address-family of the address of the TCP Connection ID field. When this field is 0, a mechanism outside the scope of this spec is used to obtain the addresses used to establish the TCP connection.

Reserved: Set to zero on transmission and ignored on receipt.

Exp: For experimental use [RFC3692] (Narten, T., “Assigning Experimental and Testing Numbers Considered Useful,” January 2004.).

TCP Connection ID: An IPv4 or IPv6 address used to establish the TCP connection. This field is omitted (length 0) for the Connection ID AFI 0.

Interface ID: An Interface ID is used to associate the connection a Join/Prune message is received over with an interface which is added or removed from an oif-list. When unnumbered interfaces are

> used or when a single Transport connection is used for sending and receiving Join/Prune messages over multiple interfaces, the Interface ID is used convey the interface from Join/Prune message sender to Join/Prune message receiver. When a PIM router sets a locally generated value for the Interface ID in the Hello TLV, it must send the same Interface ID value in all Join/Prune messages it is sending to the PIM neighbor.

3.2. PIM over the SCTP Transport Protocol

TOC

Option Type: PIM-over-SCTP Capable

```
 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 = 28           |         Length = X + 8        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   SCTP Connection ID AFI      |          Reserved       |  Exp |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       SCTP Connection ID                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Interface ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Allocated Hello Type values can be found in [HELLO-OPT] (IANA, “PIM Hello Options,” March 2007.).
When a router is configured to use PIM over SCTP on a given interface, it MUST include the PIM-over-SCTP Capable hello option in its Hello messages for that interface. If a router is explicitly disabled from using PIM over SCTP it MUST NOT include the PIM-over-SCTP Capable hello option in its Hello messages. When the router cannot setup a SCTP connection, it will refrain from including this option.
Implementations may provide a configuration option to enable or disable PORT functionality. We recommend that this capability be disabled by default.

> **Length:** In bytes for the value part of the Type/Length/Value encoding. Where X is 4 bytes if AFI of value 1 (IPv4) is used and 16 bytes when AFI of value 2 (IPv6) is used [AFI] (IANA, “Address Family Indicators (AFIs),” February 2007.).
>
> **SCTP Connection ID AFI:** The AFI value to describe the address-family of the address of the SCTP Connection ID field. When this

field is 0, a mechanism outside the scope of this spec is used to obtain the addresses used to establish the SCTP connection.

Reserved: Set to zero on transmission and ignored on receipt.

Exp: For experimental use [RFC3692] (Narten, T., "Assigning Experimental and Testing Numbers Considered Useful," January 2004.).

SCTP Connection ID: An IPv4 or IPv6 address used to establish the SCTP connection. This field is omitted (length 0) for the Connection ID AFI 0.

Interface ID: An Interface ID is used to associate the connection a Join/Prune message is received over with an interface which is added or removed from an oif-list. When unnumbered interfaces are used or when a single Transport connection is used for sending and receiving Join/Prune messages over multiple interfaces, the Interface ID is used convey the interface from Join/Prune message sender to Join/Prune message receiver. When a PIM router sets a locally generated value for the Interface ID in the Hello TLV, it must send the same Interface ID value in all Join/Prune messages it is sending to the PIM neighbor.

---

4. Establishing Transport Connections

TOC

While a router interface is PORT enabled, a PIM-over-TCP or a PIM-over-SCTP option is included in the PIM Hello messages sent on that interface. When a router on a PORT-enabled interface receives a Hello message containing a PIM-over-TCP/PIM-over-SCTP Option from a new neighbor, or an existing neighbor that did not previously include the option, it switches to PORT mode for that particular neighbor.
When a router switches to PORT mode for a neighbor, it stops sending and accepting Native Join/Prune messages for that neighbor. Any state from previous Native Join/Prune messages is left to expire as normal. It will also attempt to establish a Transport connection (TCP or SCTP) with the neighbor. If both the router and its neighbor have announced both PIM-over-TCP and PIM-over-SCTP options, SCTP MUST be used.
When the router is using TCP it will compare the TCP Connection ID it announced in the PIM-over-TCP Capable Option with the TCP Connection ID in the Hello received from the neighbor. The router with the lower Connection ID will do an active Transport open to the neighbor Connection ID. The router with the higher Connection ID will do a passive Transport open. An implementation may open connections only on-demand, in that case it may be that the neighbor with the higher Connection ID does the active open, see Section 4.3 (On-demand versus

Pre-configured Connections). Note that the source address of the active open must be the announced Connection ID.

When the router is using SCTP, the IP address comparison need not be done since the SCTP protocol can handle call collision.

If PORT is used both for IPv4 and IPv6, both IPv4 and IPv6 PIM Hello messages are sent, both containing PORT Hello options. If two neighbors announce the same transport (TCP or SCTP) and the same Connection ID in the IPv4 and IPv6 Hello messages, then only one connection is established and is shared. Otherwise, two connections are established and are used separately.

The PIM router that performs the active open initiates the connection with a locally generated source transport port number and a well-known destination transport port number. The PIM router that performs the passive open listens on the well-known local transport port number and does not qualify the remote transport port number. See Section 5 (Common Header Definition) for well-known port number assignment for PORT.

When a Transport connection is established (or reestablished), the two routers MUST both send a full set of Join/Prune messages for state for which the other router is the upstream neighbor. This is needed to ensure that the upstream neighbor has the correct state. When moving from Datagram mode, or when the connection has gone down, the router cannot be sure that all the previous Join/Prune state was received by the neighbor. Any state received while in Datagram mode that is not refreshed, will be left to expire.

When a Transport connection goes down, Join/Prune state that was sent over the Transport connection is still retained. The neighbor should not be considered down until the neighbor timer has expired. This allows routers to do a control-plane switchover without disrupting the network. If a Transport connection is reestablished before the neighbor timer expires, the previous state is intact and any new Join/Prune messages sent cause state to be created or removed (depending on if it was a Join or Prune). If the neighbor timer does expire, only the upstream router, that has oif-list state, to the expired downstream neighbor will need to clear state. A downstream router, when an upstream neighboring router has expired, will simply update the RPF for the corresponding state to a new neighbor where it would trigger Join/Prune messages like it would in [RFC4601] (Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, “Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised),” August 2006.). It is required of a PIM router to clear its neighbor table for a neighbor who has timed out due to neighbor holdtime expiration.

Note that, a Join sent over a Transport connection will only be seen by the upstream router, and thus will not cause routers on the link that do not use PIM PORT with the upstream router to possibly delay the refresh of Join state for the same state. Similarly, a Prune sent over a Transport connection will only be seen by the upstream router, and will thus never cause routers on the link on the link that do not use

PIM PORT with the upstream router, to send a Join to override this Prune.

Note also, that a datagram PIM Join/Prune message for a said (S,G) or (*,G) sent by some router on a link will not cause routers on the same link that use a Transport connection with the upstream router for that state, to suppress the refresh of that state to the usptream router (because they don't need to periodically refresh this state) or to send a Join to override a Prune (as the upstream router will only stop forwarding the traffic when all joined routers that use a Transport connection have explicitly sent a Prune for this state, as explained in Section 6 (Explicit Tracking)).

4.1. TCP Connection Maintenance

TOC

TCP is designed to keep connections up indefinitely during a period of network disconnection. If a PIM-over-TCP router fails, the TCP connection may stay up until the neighbor actually reboots, and even then it may continue to stay up until you actually try to send the neighbor some information. This is particularly relevant to PIM, since the flow of Join/Prune messages might be in only one direction, and the downstream neighbor might never get any indication via TCP that the other end of the connection is not really there.

Implementations SHOULD support the use of TCP Keep-Alives, see [RFC1122] (Braden, R., "Requirements for Internet Hosts - Communication Layers," October 1989.) section 4.2.3.6. We recommend the use of Keep-Alives to be optional, allowing network administrators to use it as needed. Note that Keep-Alives can be used by a peer, independently of whether the other peer supports it. With the use of Keep-Alives one can detect that a connection is not working without sending any TCP data.

Most applications using TCP want to detect when a neighbor is no longer there, so that the associated application state can be released. Also, one wants to clean up the TCP state, and not keep half-open connections around indefinitely. This is accomplished by using PIM Hellos and by not introducing an application-specific or new PIM keep-alive message. Therefore, when a GENID changes from a received PIM Hello message, and a TCP connection is established or attempting to be established, the local side will tear down the connection and attempt to reopen a new one for the new instance of the neighbor coming up. However, if the connection is shared by multiple interfaces and the GENID changes only for one of them, then there was not a full reboot and the connection is likely to still work. In that case, the router should just resend all Join/Prune state for that particular neighbor. This is similar to how state is refreshed when GENID changes for PIM in datagram mode.

There may be situations where a router ignores some joins or prunes. E.g. due to wrong RP information or receiving joins on an RPF interface. A router may try to cache such messages and apply them later if only a temporary error. It may however also ignore the message, and later change its GENID for that interface to make the neighbor resend all state, including any that may have been previously ignored. It is possible that one receives Join/Prune messages for an interface/link that is down. As long as the neighbor has not expired, we recommend processing those messages as usual. If they are ignored, then the router should change the GENID for that interface when it comes back up, in order to get a full update.

4.2. Moving from PORT to Datagram Mode

TOC

There may be situations where an administrator decides to stop using PORT. If PORT is disabled on a router interface, we start expiry timers with the respective neighbor holdtimes as the initial values. Similarly if we receive a Hello message without a PORT Capable option from a neighbor, we start expiry timers for all Join/Prune state we have for that particular neighbor. The Transport connection should be shut down as soon as there are no more PIM neighborships using it. That is, for the connection we have associated local and remote Connection IDs. When there is no PIM neighbor with that particular remote connection ID on any interface where we announce the local connection ID, the connection should be shut down.

4.3. On-demand versus Pre-configured Connections

TOC

Transport connections could be established when they are needed or when a router interface to other PIM neighbors has come up. The advantage of on-demand Transport connection establishment is the reduction of router resources. Especially in the case where there is no need for n^2 connections on a network interface. The disadvantage is additional delay and queueing when a Join/Prune message needs to be sent and a Transport connection is not established yet.

If a router interface has become operational and PIM neighbors are learned from Hello messages, at that time, Transport connections may be established. The advantage is that a connection is ready to transport data by the time a Join/Prune message needs to be sent. The disadvantage is there can be more connections established than needed. This can occur when there is a small set of RPF neighbors for the active distribution trees compared to the total number of neighbors.

Even when Transport connections are pre-established before they are needed, a connection can go down and an implementation will have to deal with an on-demand situation.

Note that for TCP, it is the router with the lower Connection ID that decides whether to open a connection immediately, or on-demand. The

router with the higher Connection ID should only initiate a connection on-demand. That is, if it needs to send a Join/Prune message and there is no currently established connection.
Therefore, this specification recommends but does not mandate the use of on-demand Transport connection establishment.

4.4. Possible Hello Suppression Considerations

TOC

This specification indicates that a Transport connection cannot be established until a Hello message is received. One reason for this is to determine if the PIM neighbor supports this specification and the other is to determine the remote address to use to establish the Transport connection.
There are cases where it is desirable to suppress entirely the transmission of Hello messages. In this case, it is outside the scope of this document on how to determine if the PIM neighbor supports this specification as well as an out-of-band (outside of the PIM protocol) method to determine the remote address to establish the Transport connection.

4.5. Avoiding a Pair of Connections between Neighbors

TOC

To ensure there are not two connections between a pair of PIM neighbors, the following set of rules must be followed. Let A and B be two PIM neighbors where A's Connection ID is numerically smaller than B's Connection ID, and each is known to the other as having a potential PIM adjacency relationship.
At node A:

- If there is already an established TCP connection to B, on the PIM-over-TCP port, then A MUST NOT attempt to establish a new connection to B. Rather it uses the established connection to send Join/Prune messages to B. (This is independent of which node initiated the connection.)

- If A has initiated a connection to B, but the connection is still in the process of being established, then A MUST refuse any connection on the PIM-over-TCP port from B.

- At any time when A does not have a connection to B which is either established or in the process of being established, A MUST accept connections from B.

At node B:

- If there is already an established TCP connection to A, on the PIM-over-TCP port, then B MUST NOT attempt to establish a new connection to A. Rather it uses the established connection to send Join/Prune messages to A. (This is independent of which node initiated the connection.)

- If B has initiated a connection to A, but the connection is still in the process of being established, then if A initiates a connection too, B MUST accept the connection initiated by A and must release the connection which it (B) initiated.

---

5. Common Header Definition

TOC

It may be desirable for scaling purposes to allow Join/Prune messages from different PIM protocol instances to be sent over the same Transport connection. Also, it may be desirable to have a set of Join/Prune messages for one address-family sent over a Transport connection that is established over a different address-family network layer.

To be able to do this we need a common header that is inserted and parsed for each PIM Join/Prune message that is sent on a Transport connection. This common header will provide both record boundary and demux points when sending over a stream protocol like Transport.

Each Join/Prune message will have in front of it the following common header in Type/Length/Value format. And multiple different TLV types can be sent over the same Transport connection.

To make sure PIM Join/Prune messages are delivered as soon as the TCP transport layer receives the Join/Prune buffer, the TCP Push flag will be set in all outgoing Join/Prune messages sent over a TCP transport connection.

PIM messages will be sent using destination TCP port number 8471. When using SCTP as the reliable transport, destination port number 8471 will be used. See Section 10 (IANA Considerations) for IANA considerations.

Join/Prune messages are error checked. This includes a bad PIM checksum, illegal type fields, illegal addresses or a truncated message. If any parsing errors occur in a Join/Prune message, it is skipped, and we proceed processing any following TLVs.

The TLV type field is 16 bits. The range 61440 - 65535 is for experimental use [RFC3692] (Narten, T., “Assigning Experimental and Testing Numbers Considered Useful,” January 2004.).

The current list of defined TLVs are:

IPv4 Join/Prune Message

```
 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 = 1             |        Length = X + 16        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Reserved               |  Exp  |I-Type |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Interface ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      Instance ID . . .                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      . . . Instance ID                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    PIMv2 Join/Prune Message                   |
|                               .                               |
|                               .                               |
|                               .                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The IPv4 Join/Prune common header is used when a Join/Prune message is sent that has all IPv4 encoded addresses in the PIM payload.

Length: In bytes for the value part of the Type/Length/Value encoding. Where X is the number of bytes that make up the PIMv2 Join/Prune message.

Reserved: Set to zero on transmission and ignored on receipt.

Exp: For experimental use [RFC3692] (Narten, T., “Assigning Experimental and Testing Numbers Considered Useful,” January 2004.).

I-Type: Defines the encoding and semantics of the Instance ID field. Instance Type 0 means Instance ID is not used. Other values are not defined in this specification. A message with an unknown Instance Type MUST be ignored.

Interface ID: This is the Interface ID from the Hello TLV, defined in this specification, the PIM router is sending to the PIM neighbor. It indicates to the PIM neighbor what interface to associate the Join/Prune with.

Instance ID: This document only defines this for Instance Type 0. For type 0 the field should be set to zero on transmission and ignored on receipt. This field is always 64 bits.

PIMv2 Join/Prune Message: PIMv2 Join/Prune message and payload with no IP header in front of it. As you can see from the packet format diagram, multiple Join/Prune messages can go into one TCP/

SCTP stream from the same or different Interface and Instance IDs.

IPv6 Join/Prune Message

```
 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 = 2             |         Length = X + 16       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Reserved                |  Exp  |I-Type |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Interface ID                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Instance ID . . .                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     . . . Instance ID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   PIMv2 Join/Prune Message                    |
|                               .                               |
|                               .                               |
|                               .                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The IPv6 Join/Prune common header is used when a Join/Prune message is sent that has all IPv6 encoded addresses in the PIM payload.

Length: In bytes for the value part of the Type/Length/Value encoding. Where X is the number of bytes that make up the PIMv2 Join/Prune message.

Reserved: Set to zero on transmission and ignored on receipt.

Exp: For experimental use [RFC3692] (Narten, T., “Assigning Experimental and Testing Numbers Considered Useful,” January 2004.).

I-Type: Defines the encoding and semantics of the Instance ID field. Instance Type 0 means Instance ID is not used. Other values are not defined in this specification.

Interface ID: This is the Interface ID from the Hello TLV, defined in this specification, the PIM router is sending to the PIM neighbor. It indicates to the PIM neighbor what interface to associate the Join/Prune with.

Instance ID: This document only defines this for Instance Type 0. For type 0 the field should be set to zero on transmission and ignored on receipt.

PIMv2 Join/Prune Message:

> PIMv2 Join/Prune message and payload with no IP header in front of it. As you can see from the packet format diagram, multiple Join/Prune messages can go into one TCP/SCTP stream from the same or different Interface and Instance IDs.

---

6. Explicit Tracking

TOC

When explicit tracking is used, a router keeps track of join state for individual downstream neighbors on a given interface. This is done for all PORT joins and prunes. It may also be done for native join/prune messages, if all neighbors on the LAN have set the T bit of the LAN Prune Delay option. In the discussion below we will talk about ET (explicit tracking) neighbors, and non-ET neighbors. The set of ET neighbors always includes the PORT neighbors. The set of non-ET neighbors consists of all the non-PORT neighbors unless all neighbors have set the LAN Prune Delay T bit. Then the ET neighbors set contains all neighbors.

For some link-types, e.g. point-to-point, tracking neighbors is no different than tracking interfaces. It may also be possible for an implementation to treat different downstream neighbors as being on different logical interfaces, even if they are on the same physical link. Exactly how this is implemented and for which link types, is left to the implementer.

For (*,G) and (S,G) state, the router starts forwarding traffic on an interface when a Join is received from a neighbor on such an interface. When a non-ET neighbor sends a Prune, there is generally a small delay to see if another non-ET neighbor sends a Join to override the Prune. If there is no override, one should note that no non-ETP neighbor is interested. If no ET neighbors are interested, the interface can be removed from the oif-list. When a ET neighbor sends a Prune, one removes the join state for that neighbor. If no other ET or non-ET neighbors are interested, the interface can be removed from the oif-list. When a PORT neighbor sends a prune, there can be no Prune Override, since the Prune is not visible to other neighbors.

For (S,G,R) state, the router needs to track Prune state on the shared tree. It needs to know which ET neighbors have sent prunes, and whether any non-ET neighbors have sent prunes. Normally one would forward a packet from a source S to a group G out on an interface if a (*,G)-join is received, but no (S,G,R)-prune. With ET one needs to do this check per ET neighbor. That is, the packet should be forwarded unless all ET neighbors that have sent (*,G)-joins have also sent (S,G,R)-prunes, and if a non-ET neighbor has sent a (*,G)-join, whether there also is non-ET (S,G,R)-prune state.

7. Multiple Instances and Address-Family Support

TOC

Multiple instances of the PIM protocol may be used to support e.g. multiple address families. Multiple instances can cause a multiplier effect on the number of router resources consumed. To be able to have an option to use router resources more efficiently, muxing Join/Prune messages over fewer Transport connections can be performed.

There are two ways this can be accomplished, one using a common header format over a TCP connection and the other using multiple streams over a single SCTP connection.

Using the Common Header format described previously in this specification, using different TLVs, both IPv4 and IPv6 based Join/Prune messages can be encoded within a Transport connection. Likewise, within a TLV, multiple occurrences of Join/Prune messages can occur and are tagged with an instance-ID so multiple Join/Prune messages for different instances can use a single Transport connection.

When using SCTP multi-streaming, the common header is still used to convey instance information but an SCTP association is used, on a per-instance basis, to send data concurrently for multiple instances. When data is sent concurrently, head of line blocking, which can occur when using TCP, is avoided.

8. Miscellany

TOC

No changes expected in processing of other PIM messages like PIM Asserts, Grafts, Graft-Acks, Registers, and Register-Stops. This goes for BSR and Auto-RP type messages as well.

This extension is applicable only to PIM-SM, PIM-SSM and Bidir-PIM. It does not take requirements for PIM-DM into consideration.

9. Security Considerations

TOC

Transport connections can be authenticated using HMACs MD5 and SHA-1 similar to use in BGP [RFC4271] (Rekhter, Y., Li, T., and S. Hares, “A Border Gateway Protocol 4 (BGP-4),” January 2006.) and MSDP [RFC3618] (Fenner, B. and D. Meyer, “Multicast Source Discovery Protocol (MSDP),” October 2003.).

When using SCTP as the transport protocol, [RFC4895] (Tuexen, M., Stewart, R., Lei, P., and E. Rescorla, “Authenticated Chunks for the Stream Control Transmission Protocol (SCTP),” August 2007.) can be used, on a per SCTP association basis to authenticate PIM data.

10. IANA Considerations

TOC

This specification makes use of a TCP port number and a SCTP port number for the use of PIM-Over-Reliable-Transport that has been allocated by IANA. It also makes use of IANA PIM Hello Options allocations that should be made permanent. In addition, a registry for PORT message types is requested. The registry should cover the range 0 - 61439. An RFC is required for assignments in that range. This document defines two PORT message types. Type 1, IPv4 Join/Prune Message; and Type 2, IPv6 Join/Prune Message. The type range 61440 - 65535 is for experimental use [RFC3692] (Narten, T., “Assigning Experimental and Testing Numbers Considered Useful,” January 2004.).

11. Contributors

TOC

In addition to the persons listed as authors, significant contributions were provided by Apoorva Karan and Arjen Boers.

12. Acknowledgments

TOC

The authors would like to give a special thank you and appreciation to Nidhi Bhaskar for her initial design and early prototype of this idea. Appreciation goes to Randall Stewart for his authoritative review and recommendation for using SCTP.

Thanks also goes to the following for their ideas and commentary review of this specification, Mike McBride, Toerless Eckert, Yiqun Cai, Albert Tian, Suresh Boddapati, Nataraj Batchu, Daniel Voce, John Zwiebel, Yakov Rekhter, Lenny Giuliano, Gorry Fairhurst, Sameer Gulrajani, Thomas Morin and Dimitri Papadimitriou.

A special thank you goes to Eric Rosen for his very detailed review and commentary. Many of his comments are reflected as text in this specification.

13. References

TOC

13.1. Normative References

TOC

[RFC0793]	Postel, J., "Transmission Control Protocol," STD 7, RFC 793, September 1981 (TXT).
[RFC1122]	Braden, R., "Requirements for Internet Hosts - Communication Layers," STD 3, RFC 1122, October 1989 (TXT).
[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, March 1997 (TXT, HTML, XML).
[RFC3618]	Fenner, B. and D. Meyer, "Multicast Source Discovery Protocol (MSDP)," RFC 3618, October 2003 (TXT).
[RFC4271]	Rekhter, Y., Li, T., and S. Hares, "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, January 2006 (TXT).
[RFC4601]	Fenner, B., Handley, M., Holbrook, H., and I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)," RFC 4601, August 2006 (TXT, PDF).
[RFC4895]	Tuexen, M., Stewart, R., Lei, P., and E. Rescorla, "Authenticated Chunks for the Stream Control Transmission Protocol (SCTP)," RFC 4895, August 2007 (TXT).
[RFC4960]	Stewart, R., "Stream Control Transmission Protocol," RFC 4960, September 2007 (TXT).
[RFC5015]	Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)," RFC 5015, October 2007 (TXT).

13.2. Informative References

TOC

[AFI]	IANA, "Address Family Indicators (AFIs)," ADDRESS FAMILY NUMBERS http://www.iana.org/numbers.html, February 2007.
[HELLO-OPT]	IANA, "PIM Hello Options," PIM-HELLO-OPTIONS per RFC4601 http://www.iana.org/assignments/pim-hello-options, March 2007.
[RFC3692]	Narten, T., "Assigning Experimental and Testing Numbers Considered Useful," BCP 82, RFC 3692, January 2004 (TXT).

Authors' Addresses

TOC

	Dino Farinacci
	cisco Systems
	Tasman Drive
	San Jose, CA 95134
	USA

Email:	dino@cisco.com
	IJsbrand Wijnands
	cisco Systems
	Tasman Drive
	San Jose, CA 95134
	USA
Email:	ice@cisco.com
	Stig Venaas
	cisco Systems
	Tasman Drive
	San Jose, CA 95134
	USA
Email:	stig@cisco.com
	Maria Napierala
	AT&T Labs
	200 Laurel Drive
	Middletown, New Jersey 07748>
	USA
Email:	mnapierala@att.com