Network Working Group
Internet-Draft
Intended status: Experimental
Expires: August 19, 2011

D. Farinacci
IJ. Wijnands
S. Venaas
cisco Systems
M. Napierala
AT&T Labs

February 15, 2011

A Reliable Transport Mechanism for PIM
draft-ietf-pim-port-05.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

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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

This Internet-Draft will expire on August 19, 2011.

Table of Contents

1. Introduction

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] (including Source-Specific Multicast) and Bidirectional PIM [RFC5015].

- 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].

- Provide support for automatic switching between the reliable transport mechanism and the regular PIM mechanism defined in [RFC4601]. Two routers that are PIM neighbors on a link will always use the reliable transport mechanism if and only if both have reliable transport enabled.

This document will specify how periodic Join/Prune message transmission can be eliminated by using TCP [RFC0793] or SCTP [RFC4960] 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 for maintaining efficient use of bandwidth.

1.1. Requirements Notation

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].

1.2. Definitions

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

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] and Bidirectional PIM [RFC5015] 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 are still used for PIM neighbors that are not PORT enabled.

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. Also, any received native Join/Prune messages from that neighbor are discarded, even when the connection is down.

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. If there also are native joins/ prunes from non-PORT neighbor, then one can maintain timers on the

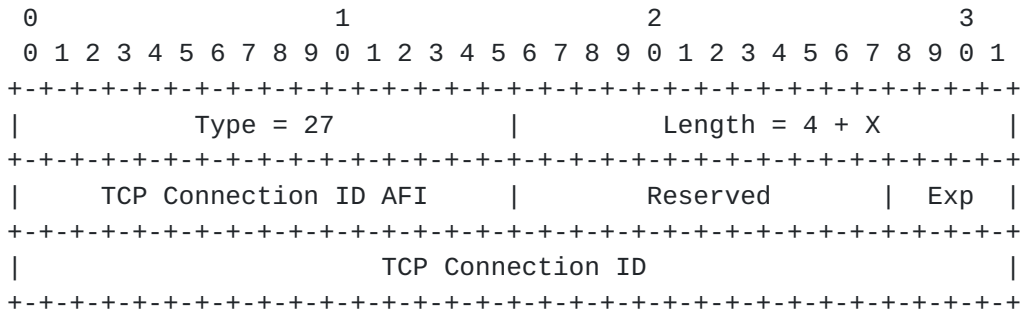
outgoing interface entry as usual, while at the same time keep track of each of the downstream PORT joins/prunes.

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. Each message is contained in a PORT message. See section Section 5 for details on the PORT message.

3. PIM Hello Options

3.1. PIM over the TCP Transport Protocol

Option Type: PIM-over-TCP Capable



Allocated Hello Type values can be found in [HELLO-OPT].

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.

All Hello messages containing the PIM-over-TCP Capable hello option, MUST also contain the Interface ID hello option, see section .

Implementations MAY provide a configuration option to enable or disable PORT functionality. We RECOMMEND that this capability be disabled by default.

Length: Length in bytes for the value part of the Type/Length/Value encoding; where X is the number of bytes that make up the Connection ID field. X is 4 when AFI of value 1 (IPv4) is used, 16 when AFI of value 2 (IPv6) is used, and 0 if AFI of value 0 is used [AFI].

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 document is used to obtain the addresses used to establish the TCP connection.

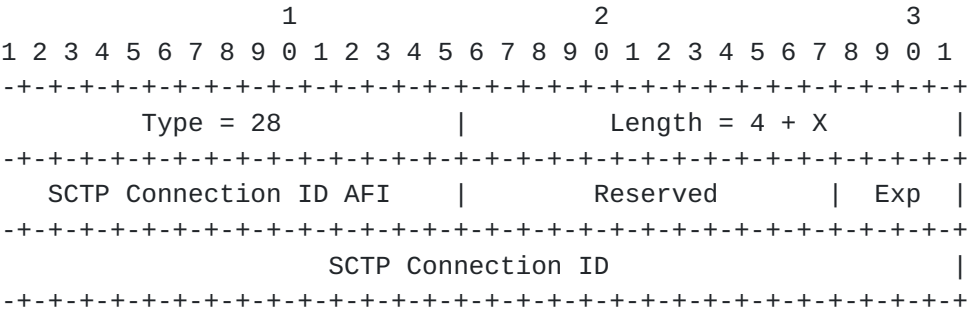
Reserved: Set to zero on transmission and ignored on receipt.

Exp: For experimental use [RFC3692].

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.

3.2. PIM over the SCTP Transport Protocol

Option Type: PIM-over-SCTP Capable



Allocated Hello Type values can be found in [HELLO-OPT].

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.

All Hello messages containing the PIM-over-SCTP Capable hello option, MUST also contain the Interface ID hello option, see section .

Implementations MAY provide a configuration option to enable or disable PORT functionality. We RECOMMEND that this capability be disabled by default.

Length: Length in bytes for the value part of the Type/Length/Value encoding; where X is the number of bytes that make up the Connection ID field. X is 4 when AFI of value 1 (IPv4) is used, 16 when AFI of value 2 (IPv6) is used, and 0 if AFI of value 0 is used [AFI].

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 document 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].

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.

3.3. Interface ID

All Hello messages containing PIM-over-TCP Capable or PIM-over-SCTP Capable hello options, MUST also contain the Interface ID hello option [I-D.gulrajani-pim-hello-intid].

The 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 to convey the interface from Join/Prune message sender to Join/Prune message receiver. The value of the Interface ID hello option in Hellos sent on an interface, must be the same as the Interface ID value in all PORT Join/Prune messages sent to a PIM neighbor on that interface.

The Interface ID need only uniquely identify an interface of a router, it does not need to identify which router the interface belongs to. This means that the Router ID part of the Interface ID MAY be 0. For details on the Router ID and the value 0, see [I-D.gulrajani-pim-hello-intid].

4. Establishing Transport Connections

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 4.5. 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 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.

It is possible that a router starts sending Hello messages with a new Connection ID, e.g. due to configuration changes. One MUST always use the last announced and last seen Connection IDs. When a Connection ID changes, if the previously used connection is not needed (there are no other PIM neighborships using the same pair of Connection IDs), both peers MUST attempt a graceful shutdown of the connection. Next (even if the old connection is still needed), they MUST, unless a connection already exists with the new Connection IDs, immediately or on-demand attempt to establish a new connection with the new Connection IDs.

Normally the Interface ID would not change while a connection is up. However, if it does, it should not affect the connection. It just means that when subsequent PORT join/prune messages are received, they should be matched against the last seen Interface ID.

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 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 upstream 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).

4.1. Connection Security

TCP/SCTP packets MUST be sent with a TTL/Hop Limit of 255 to facilitate enabling of the Generalized TTL Security Mechanism (GTSM) [RFC5082]. Implementations SHOULD provide a configuration option to enable the GTSM check. This means checking that inbound packets from directly connected neighbors have a TTL/Hop Limit of 255, but MAY

also allow for a different TTL/Hop Limit threshold to check that the sender is within a certain number of router hops. The GTSM check SHOULD be disabled by default.

Implementations SHOULD support the TCP Authentication Option (TCP-AO) [RFC5925].

4.2. Connection Maintenance

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.

One can quicker detect that a PORT connection is not working by regularly sending PORT messages. PORT in itself does not require any periodic signaling. PORT Join/Prune messages are only sent when there is a state change. If the state changes are not frequent enough, a PORT Keep-Alive message can be sent instead. E.g. if an implementation wants to send a PORT message, to check that the connection is working, at least every 60 seconds, then whenever there is 60 seconds since the the previous message, a Keep-Alive message could be sent. If there were less than 60 seconds between each Join/Prune, no Keep-Alive messages would be needed. Implementations SHOULD support the use of PORT Keep-Alive messages. We RECOMMEND this 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.

As described in the previous paragraph, an implementation can make use of Keep-Alives to regularly send messages and detect when a connection is not working. For TCP the connection will be reset if no TCP ACKs are received. A quicker and more reliable way of detecting that a connection is not working, is to send regular PORT messages, and have our peer take down the connection if it doesn't receive them. This can be done by sending Keep-alive messages with a non-zero holdtime value. If the last received Keep-alive message had a non-zero holdtime, one tears down the connection if the time measured in seconds since the last processed PORT message exceeds the specified holdtime.

Implementations SHOULD support Keep-Alive messages. An implementation that supports Keep-Alive messages acts as follows when processing a received PORT message. When processing a Keep-Alive

message with a non-zero Holdtime value, it MUST set a timer to the value. We call this timer Connection Expiry Timer (CET). If the CET is already running, it MUST be reset to the new value. When processing a Keep-Alive message with a zero Holdtime value, the CET MUST be stopped if running. When processing a PORT message other than Keep-Alive, the CET MUST be reset to the last received Holdtime value if running. If the CET is not running, no action is taken. If the CET expires, the connection SHOULD be shut down.

It is possible that a router 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 ensure it gets a full update for that interface when it comes back up. This can be done by changing the GenID, or by terminating and reestablishing the connection.

If a PORT neighbor changes its GenID and a connection is established or attempting to be established, the local side should generally tear down the connection and do as described in Section 4.3. However, if the connection is shared by multiple interfaces and the GenID changes only for one of them, then there was not a full restart, and one may simply send a full update similar to other cases when a GenID changes for an upstream neighbor.

4.3. Actions When a Connection Goes Down

A connection may go down for a variety of reasons. It may be due to an error condition, or a configuration change. A 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. This may happen when a new connection ID is configured, PORT is disabled, or a PIM neighbor expires.

If a PIM neighbor expires, one should free connection state and downstream oif-list state for the neighbor. 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]. It is required of a PIM router to clear its neighbor table for a neighbor who has timed out due to neighbor holdtime expiration.

When a connection is no longer available between two PORT enabled PIM neighbors, they MUST immediately, or on-demand, try to reestablish the connection following the normal rules for connestion establishment. The neighbors MUST also start expiry timers so that

all oif-list state for the neighbor using the connection, gets expired after JP_HOLDTIME, unless it later gets refreshed by receiving new Join/Prunes.

The value of JP_HOLDTIME is 215 seconds. This value is based on section 4.11 of [RFC4601] which says that J/P_HoldTime should be 3.5 * t_periodic where the default for t_periodic is 60 seconds.

4.4. Moving from PORT to Datagram Mode

There may be situations where an administrator decides to stop using PORT. If PORT is disabled on a router interface, or a previously PORT enabled neighbor no longer announces any of the PORT Hello options, one follows the rules in Section 4.3 for taking down connections and starting timers. Next, one should trigger a full state update similar to what would be done if the GenID changed in Datagram Mode. This means sending joins for any state where we switched from PORT to Datagram Mode for the upstream neighbor.

4.5. On-demand versus Pre-configured Connections

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 a full mesh of 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.

4.6. Possible Hello Suppression Considerations

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.

4.7. Avoiding a Pair of TCP Connections between Neighbors

To ensure that there is only one TCP connection between a pair of PIM neighbors, the following set of rules must be followed. Note that this section applies only to TCP, for SCTP this is not an issue. 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. PORT Message Definition

It may be desirable for scaling purposes to allow Join/Prune messages from different PIM protocol families 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 PORT message message format. This will provide both record boundary and demux points when sending over a stream protocol like TCP/SCTP.

A PORT message may contain PORT options, see Section 5.3. We will define two PORT options for carrying PIM Join/Prune messages. One for IPv4 and one for IPv6. For each PIM Join/Prune message to be sent over the Transport connection, we send a PORT Join/Prune message containing exactly one such option.

Each PORT message will have the below Type/Length/Value format. 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.

PORT 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 for IANA considerations.

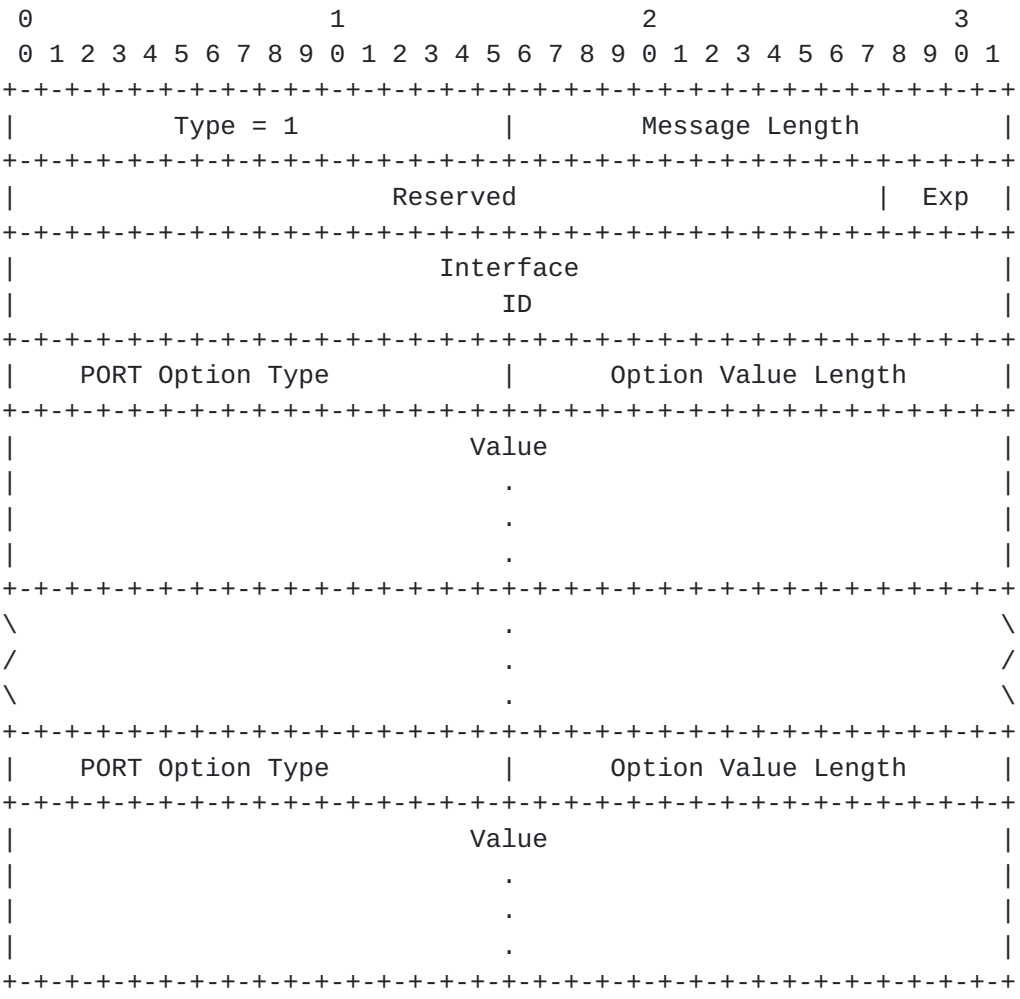
PORT messages are error checked. This includes a bad PIM checksum, illegal type fields, illegal addresses or a truncated message. If any parsing errors occur in a Join/Prune message, it is skipped, and we proceed processing any following PORT messages.

The TLV type field is 16 bits. The range 61440 - 65535 is for experimental use [RFC3692].

This document defines two message types.

5.1. PORT Join/Prune Message

PORT Join/Prune Message



The PORT Join/Prune Message is used for sending a PIM Join/Prune.

Message Length: Length in bytes for the value part of the Type/ Length/Value encoding. If no PORT Options were included, the length would be 12. If n PORT Options with Option Value lengths L1, L2, ..., Ln are included, the message length will be 12 + 4*n + L1 + L2 + ... + Ln.

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

Exp: For experimental use [RFC3692].

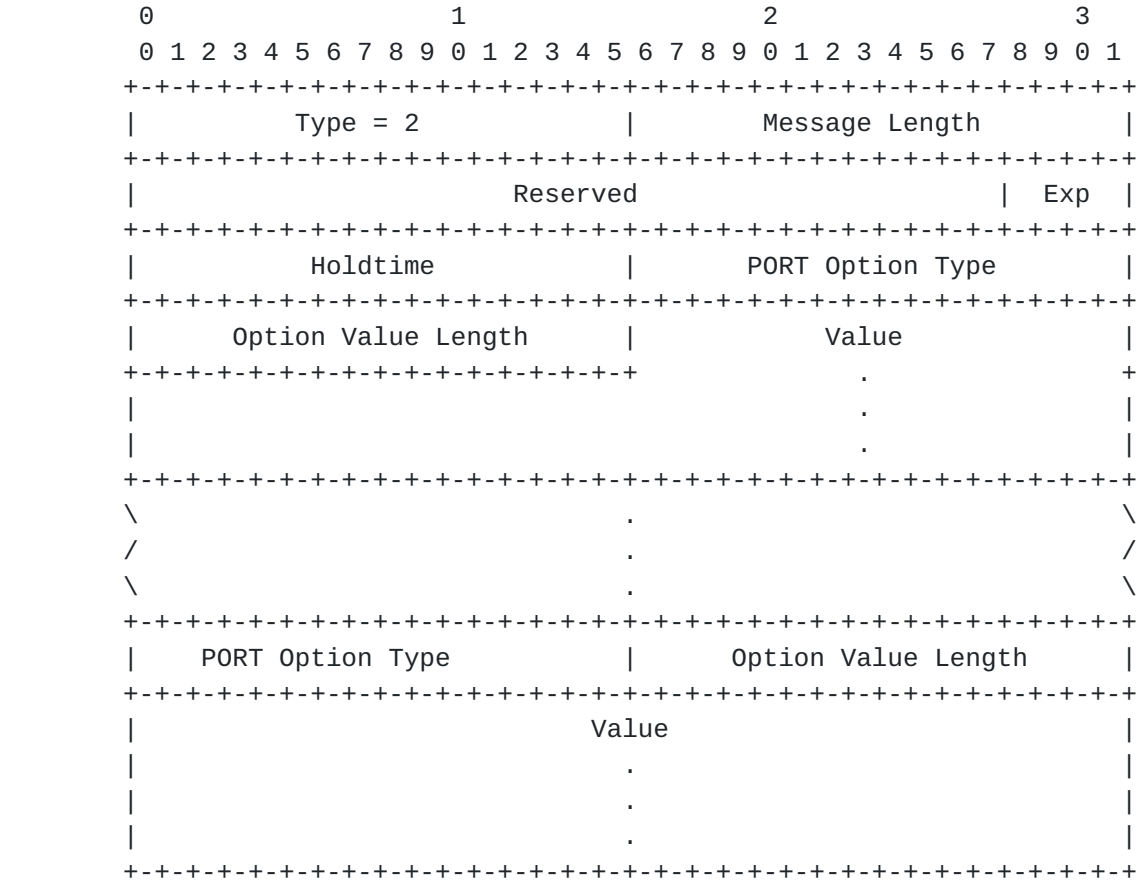
Interface ID: This is the Interface ID of the Interface ID Hello option contained in the PIM Hello messages the PIM router is sending to the PIM neighbor. It indicates to the PIM neighbor what interface to associate the Join/Prune with.

PORT Options: The message MUST contain exactly one PIM Join/Prune Port Option, either one PIM IPv4 Join/Prune or one PIM IPv6 Join/Prune. It MUST NOT contain both. It MAY contain additional options not defined in this document. A router receiving a PORT Join/Prune message containing unknown options MUST ignore the entire PORT message. See Section 5.3 for option definitions.

As can be seen from the packet format diagram, multiple Join/Prune messages can go into one TCP/SCTP stream from the same or different Interface IDs.

5.2. PORT Keep-alive Message

PORT Keep-alive Message



The PORT Keep-alive Message is used to regularly send PORT messages to verify that a connection is alive. They are used when other PORT

messages are not sent of the desired frequency.

Message Length: Length in bytes for the value part of the Type/ Length/Value encoding. If no PORT Options were included, the length would be 6. If n PORT Options with Option Value lengths L1, L2, ..., Ln are included, the message length will be 6 + 4*n + L1 + L2 + ... + Ln.

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

Exp: For experimental use [RFC3692].

Holdtime: This specifies a holdtime in seconds for the connection. A non-zero value means that the connection SHOULD be gracefully shut down if no further PORT messages are received within the specified time. This is measured on the receiving side by measuring the time from one PORT message has been processed until the next has been processed. Note that this is done for any PORT message, not just keep-alive messages. A hold time of 0 disables the keep-alive mechanism.

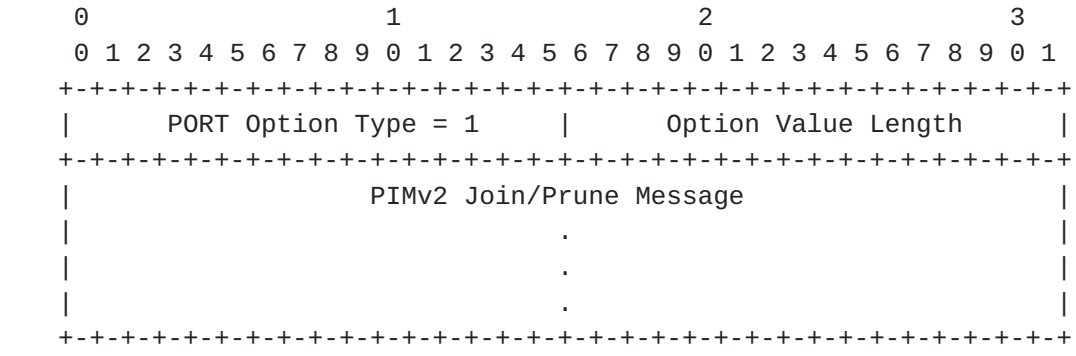
PORT Options: A keep-alive message MUST NOT contain any of the options defined in this document. It MAY contain other options not defined in this document. See Section 5.3 for option definitions.

5.3. PORT Options

Each PORT Option is a TLV. The type is 16 bits. PORT Option types are assigned by IANA, except the range 61440 - 65535 which is for experimental use [RFC3692]. The length specifies the length of the value in bytes. Below are the two options defined in this document.

PIM IPv4 Join/Prune Option

PIM IPv4 Join/Prune Option Format



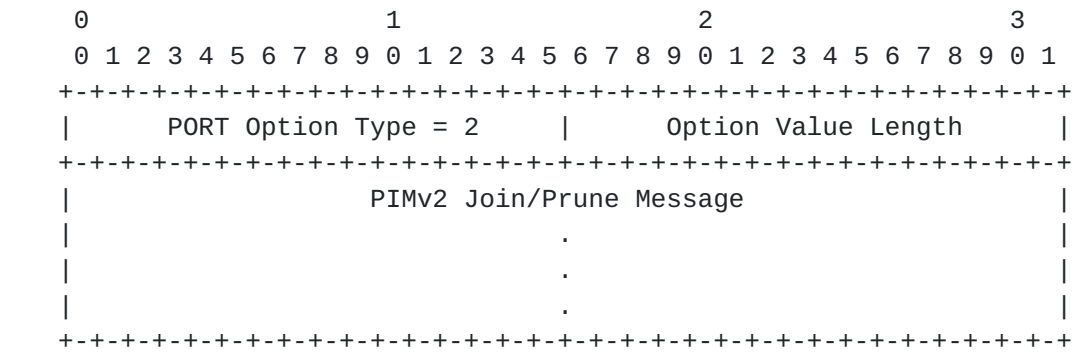
The IPv4 Join/Prune Option is used to carry a PIMv2 Join/Prune message that has all IPv4 encoded addresses in the PIM payload.

Option Value Length: The number of bytes that make up the PIMv2 Join/Prune message.

PIMv2 Join/Prune Message: PIMv2 Join/Prune message and payload with no IP header in front of it.

PIM IPv6 Join/Prune Option

PIM IPv6 Join/Prune Option Format



The IPv6 Join/Prune Option is used to carry a PIMv2 Join/Prune message that has all IPv6 encoded addresses in the PIM payload.

Option Value Length: The number of bytes that make up the PIMv2 Join/Prune message.

PIMv2 Join/Prune Message: PIMv2 Join/Prune message and payload with no IP header in front of it.

6. Explicit Tracking

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, as specified [RFC4601], if no Join is sent to override this Prune before the expiration of the Override Timer, the upstream router concludes that no non-ET neighbor is interested. If no ET neighbors are interested, the interface can be removed from the oif-list. When an 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,rpt) 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,rpt)-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,rpt)-prunes, and if a non-ET neighbor has sent a (*,G)-join, whether there also is non-ET (S,G,rpt)-prune state.

7. Multiple Address-Family Support

To allow for efficient use of router resources, one can mux Join/ Prune messages of different address families on the same Transport connections. There are two ways this can be accomplished, one using a common message format over a TCP connection and the other using multiple streams over a single SCTP connection.

Using the common message format described previously in this specification, using different PORT options, both IPv4 and IPv6 based Join/Prune messages can be encoded within the same Transport connection.

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

8. Miscellany

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

TCP connections can be authenticated using TCP-AO [RFC5925]. When using SCTP, [RFC4895] can be used for authentication on a per SCTP association basis. Also GTSM [RFC5082] can be used to help prevent spoofing.

10. IANA Considerations

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.

10.1. PORT Message Type Registry

A registry for PORT message types is requested. The message type is a 16-bit integer, with values from 0 to 65535. An RFC is required for assignments in the range 0 - 61439. This document defines one PORT message type. Type 1, PORT Join/Prune Message. The type range 61440 - 65535 is for experimental use [RFC3692].

The initial content of the registry should be as follows:

Type	Name	Reference
0	Reserved	this document
1	Join/Prune	this document
2	Keep-alive Message	this document
3-61439	Unassigned	
61440-65535	Experimental	this document

10.2. PORT Option Type Registry

A registry for PORT option types is requested. The option type is a 16-bit integer, with values from 0 to 65535. An RFC is required for assignments in the range 0 - 61439. This document defines two PORT option types. Type 1, PIM IPv4 Join/Prune Message; and Type 2, PIM IPv6 Join/Prune Message. The type range 61440 - 65535 is for experimental use [RFC3692].

The initial content of the registry should be as follows:

Type	Name	Reference
0	Reserved	this document
1	PIM IPv4 Join/Prune Message	this document
2	PIM IPv6 Join/Prune Message	this document
3-61439	Unassigned	
61440-65535	Experimental	this document

11. Contributors

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

12. Acknowledgments

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, Dimitri Papadimitriou, Bharat Joshi, Rishabh Parekh, Manav Bhatia and Pekka Savola.

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

13.1. Normative References

[I-D.gulrajani-pim-hello-intid]
Gulrajani, S. and S. Venaas, "An Interface ID Hello Option for PIM", draft-gulrajani-pim-hello-intid-00 (work in progress), February 2011.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[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 I. Kouvelas, "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", RFC 4601, August 2006.

[RFC4895] Tuexen, M., Stewart, R., Lei, P., and E. Rescorla, "Authenticated Chunks for the Stream Control Transmission Protocol (SCTP)", RFC 4895, August 2007.

[RFC4960] Stewart, R., "Stream Control Transmission Protocol", RFC 4960, September 2007.

[RFC5015] Handley, M., Kouvelas, I., Speakman, T., and L. Vicisano, "Bidirectional Protocol Independent Multicast (BIDIR-PIM)", RFC 5015, October 2007.

[RFC5082] Gill, V., Heasley, J., Meyer, D., Savola, P., and C. Pignataro, "The Generalized TTL Security Mechanism (GTSM)", RFC 5082, October 2007.

[RFC5925] Touch, J., Mankin, A., and R. Bonica, "The TCP Authentication Option", RFC 5925, June 2010.

13.2. Informative References

[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.

Authors' Addresses

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