Internet Engineering Task Force — PIM WG
INTERNET-DRAFT
draft-ietf-pim-bidir-07.ps
Mark Handley/UCL
Isidor Kouvelas/Cisco
Tony Speakman/Cisco
Lorenzo Vicisano/Cisco
18 July 2004
Expires: January 2005

Bi-directional Protocol Independent Multicast (BIDIR-PIM)

Status of this Document

Abstract

This document discusses Bi-directional PIM, a variant of PIM Sparse-Mode that builds bi-directional shared trees connecting multicast sources and receivers. Bi-directional trees are built using a fail-safe Designated Forwarder (DF) election mechanism operating on each link of a multicast topology. With the assistance of the DF, multicast data is natively forwarded from sources to the Rendezvous-Point and hence along the shared tree to receivers without requiring source-specific state. The DF election takes place at RP discovery time and provides the route to the RP thus eliminating the requirement for data-driven protocol events.

Table of Contents

List of Figures

1. Introduction

This document specifies Bi-directional PIM (BIDIR-PIM), a variant of PIM Sparse-Mode (PIM-SM) [4] that builds bi-directional shared trees connecting multicast sources and receivers.

PIM-SM constructs uni-directional shared trees that are used to forward data from senders to receivers of a multicast group. PIM-SM also allows the construction of source specific trees, but this capability is not related to the protocol described in this document.

The shared tree for each multicast group is rooted at a multicast router called the Rendezvous Point (RP). Different multicast groups can use separate RPs within a PIM domain.

In unidirectional PIM-SM, there are two possible methods for distributing data packets on the shared tree. These differ in the way packets are forwarded from a source to the RP:

- Initially when a source starts transmitting, its first hop router encapsulates data packets in special control messages (Registers) which are unicast to the RP. After reaching the RP the packets are decapsulated and distributed on the shared tree.
- A transition from the above distribution mode can be made at a later stage. This is achieved by building source specific state on all routers along the path between the source and the RP. This state is then used to natively forward packets from that source.

Both these mechanisms suffer from problems. Encapsulation results in significant processing, bandwidth and delay overheads. Forwarding using source specific state has additional protocol and memory requirements.

Bi-directional PIM dispenses with both encapsulation and source state by allowing packets to be natively forwarded from a source to the RP using shared tree state. In contrast to PIM-SM this mode of forwarding does not require any data-driven events.

The protocol specification in this document assumes familiarity with the PIM-SM specification in [4]. Portions of the BIDIR-PIM protocol operation that are identical to that of PIM-SM are only defined by reference.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 and indicate requirement levels for compliant BIDIR-PIM implementations.

2.1. Definitions

This specification uses a number of terms to refer to the roles of routers participating in BIDIR-PIM. The following terms have special significance for BIDIR-PIM:

MRIB

Multicast Routing Information Base. This is the multicast topology table, which is typically derived from the unicast routing table, or routing protocols such as MBGP that carry multicast-specific topology information. It is used by PIM for establishing the RPF interface (used in the forwarding rules). In PIM-SM the MRIB is also used to make decisions regarding where to forward Join/Prune messages whereas in BIDIR-PIM it is used as a source for routing metrics for the DF election process.

Rendezvous Point Address (RPA):

An RPA is an address that is used as the root of the distribution tree for a range of multicast groups. The RPA must be routable from all routers in the PIM domain. The RPA does not need to correspond to an address for an interface of a real router. In this respect

BIDIR-PIM differs from PIM-SM which requires an actual router to be configured as the Rendezvous Point (RP). Join messages from receivers for a BIDIR-PIM group propagate hop-by-hop towards the RPA.

Rendezvous Point Link (RPL):
: An RPL for a particular RPA is the physical link to which the RPA belongs. In BIDIR-PIM all multicast traffic to groups mapping to a specific RPA is forwarded on the RPL of that RPA. The RPL is special within a BIDIR-PIM domain as it is the only link on which a Designated Forwarder election does not take place (see DF definition below).

Upstream
: Towards the root (RPA) of the tree. The direction used by packets traveling from sources to the RPL.

Downstream
: Away from the root of the tree. The direction on which packets travel from the RPL to receivers.

Designated Forwarder (DF):
: The protocol presented in this document is largely based on the concept of a Designated Forwarder (DF). A single DF exists for each RPA on every link within a BIDIR-PIM domain (this includes both multi-access and point-to-point links). The only exception is the RPL on which no DF exists. The DF is the router on the link with the best route to the RPA (determined by comparing MRIB provided metrics). A DF for a given RPA is in charge of forwarding downstream traffic onto its link, and forwarding upstream traffic from its link towards the RPL. It does this for all the bi-directional groups that map to the RPA. The DF on a link is also responsible for processing Join messages from downstream routers on the link as well as ensuring that packets are forwarded to local receivers (discovered through a local membership mechanism such as MLD [3] or IGMP [2]).

RPF Interface
: RPF stands for "Reverse Path Forwarding". The RPF Interface of a router with respect to an address is the interface that the MRIB indicates should be used to reach that address. In the case of a BIDIR-PIM multicast group, the RPF interface is determined by looking up the RPA in the MRIB. The RPF information determines the interface of the router that would be used to send packets towards the RPL for the group.

RPF Neighbor
: The RPF Neighbor of a router with respect to an address is the neighbor that the MRIB indicates should be used to reach that address. Note that in BIDIR-PIM, the RPF neighbor for a group is not necessarily the router on the RPF interface that Join messages for that group would be directed to (Join messages are only directed to the DF on the RPF interface for the group).

TIB
: Tree Information Base. This is the collection of state at a PIM router that has been created by receiving PIM Join/Prune messages, PIM DF election messages and IGMP or MLD information from local hosts. It essentially stores the state of all multicast distribution trees at that router.

MFIB
: Multicast Forwarding Information Base. The TIB holds all the state that is necessary to forward multicast packets at a router. However, although this specification defines forwarding in terms of the TIB, to actually forward packets using the TIB is very inefficient. Instead a real router implementation will normally build an efficient MFIB from the TIB state to perform forwarding. How this is done is implementation-specific, and is not discussed in this document.

2.2. Pseudocode Notation

We use set notation in several places in this specification.

`A (+) B`
: is the union of two sets A and B.

`A (-) B`
: is the elements of set A that are not in set B.

`NULL`
: is the empty set or list.

In addition we use C-like syntax:

= denotes assignment of a variable.

== denotes a comparison for equality.

!= denotes a comparison for inequality.

Braces { and } are used for grouping.

3. Protocol Specification

The specification of BIDIR-PIM is broken into several parts:

- Section 3.1 details the protocol state stored.
- Section 3.2 defines the BIDIR-PIM extensions to the PIM-SM [4] neighbour discovery mechanism.
- Section 3.3 specifies the data packet forwarding rules.
- Section 3.4 specifies the BIDIR-PIM Join/Prune generation and processing rules.
- Designated Forwarder (DF) election is specified in Section 3.5.
- PIM packet formats are specified in Section 3.7.
- A summary of BIDIR-PIM timers and their default values is given in Section 3.6.

3.1. BIDIR-PIM Protocol State

This section specifies all the protocol state that a BIDIR-PIM implementation should maintain in order to function correctly. We term this state the *Tree Information Base* or TIB, as it holds the state of all the multicast distribution trees at this router. In this specification we define PIM mechanisms in terms of the TIB. However, only a very simple implementation would actually implement packet forwarding operations in terms of this state. Most implementations will use this state to build a multicast forwarding table, which would then be updated when the relevant state in the TIB changes.

Although we specify precisely the state to be kept, this does not mean that an implementation of BIDIR-PIM needs to hold the state in this form. This is actually an abstract state definition, which is needed in order to specify the router's behavior. A BIDIR-PIM implementation is free to hold whatever internal state it requires, and will still be conformant with this specification so long as it results in the same externally visible protocol behavior as an abstract router that holds the following state.

We divide TIB state into two sections:

RPA state
: State that maintains the DF election information for each RPA.

Group state
State that maintains a group-specific tree for groups that map to a given RPA.

The state that should be kept is described below. Of course, implementations will only maintain state when it is relevant to forwarding operations - for example, the "NoInfo" state might be assumed from the lack of other state information, rather than being held explicitly.

3.1.1. General Purpose State

A router holds the following state that is not specific to a RPA or group:

Neighbor State:

For each neighbor:

- Neighbor's Gen ID.
- Neighbor liveness timer (NLT)
- Other information from neighbor's Hello

For more information on Hello information look at section 3.2 as well as the PIM-SM specification in [4].

3.1.2. RPA State

A router maintains a multicast-group to RPA mapping which is built through static configuration or by using an automatic RP discovery mechanism like BSR or AUTO-RP (see section 4). For each BIDIR-PIM RPA a router holds the following state:

- RPA (actual address)

Designated Forwarder (DF) State:

For each router interface:

Acting DF information:

- DF IP Address
- DF metric

Election information:

- Election State
- DF election-Timer (DFT)
- Message-Count (MC)

Current best offer:

- IP address of best offering router
- Best offering router metric

Designated Forwarder state is described in section 3.5.

3.1.3. Group State

For every group G a router keeps the following state:

Group state:

For each interface:

Local Membership:

- State: One of {"NoInfo", "Include"}

PIM Join/Prune State:

- State: One of {"NoInfo" (NI), "Join" (J), "PrunePending" (PP)}
- Prune Pending Timer (PPT)
- Join/Prune Expiry Timer (ET)

Not interface specific:

- Upstream Join/Prune Timer (JT)
- Last RPA Used

Local membership is the result of the local membership mechanism (such as IGMP [2]) running on that interface. This information is used by the `pim_include(*,G)` macro described in section 3.1.4.

PIM Join/Prune state is the result of receiving PIM (*,G) Join/Prune messages on this interface, and is specified in section 3.4.1. The state is used by the macros that calculate the outgoing interface list in section 3.1.4, and in the `JoinDesired(G)` macro (defined in section 3.4.2) that is used in deciding whether a Join(*,G) should be sent upstream.

The upstream Join/Prune timer is used to send out periodic Join(*,G) messages, and to override Prune(*,G) messages from peers on an upstream LAN interface.

The last RPA used must be stored because if the group to RPA mapping changes (see RP Set changes in [4]) then state must be torn down and rebuilt for groups whose RPA changes.

3.1.4. State Summarization Macros

Using this state, we define the following "macro" definitions which we will use in the descriptions of the state machines and pseudocode in the following sections.

```
olist(G) =
    RPF_interface(RPA(G)) (+) joins(G) (+) pim_include(G)
```

`RPF_interface(RPA)` is the interface the MRIB indicates would be used to route packets to RPA. The olist(G) is the list of interfaces on which packets to group G must be forwarded.

The macro pim_include(G) indicates the interfaces to which traffic might be forwarded because of hosts that are local members on that interface.

```
pim_include(G) =
    { all interfaces I such that:
      I_am_DF(RPA(G),I) AND  local_receiver_include(G,I) }
```

The clause "I_am_DF(RPA,I)" is TRUE if the router is in the `Win` or `Backoff` states in the DF election state machine (described in section 3.5) for the given RPA on interface I. Otherwise it is FALSE.

The clause "local_receiver_include(G,I)" is true if the IGMP module, MLD module or other local membership mechanism has determined that there are local members on interface I that desire to receive traffic sent to group G.

The set "joins(G)" is the set of all interfaces on which the router has received (*,G) Joins:

```
joins(G) =
    { all interfaces I such that
      I_am_DF(RPA(G),I) AND
      DownstreamJPState(G,I) is either Joined or PrunePending }
```

DownstreamJPState(G,I) is the state of the finite state machine in section 3.4.1.

`RPF_DF(RPA)` is the neighbor that Join messages must be sent to in order to build the group shared tree rooted at the RPL for the given RPA. This is the Designated-Forwarder on the `RPF_interface(RPA)`.

3.2. PIM Neighbor Discovery

PIM routers exchange PIM-Hello messages with their neighboring PIM routers. These messages are used to update the `Neighbor State` described in section 3.1. The procedures for generating and processing Hello messages as well as maintaining `Neighbor State` are specified in the PIM-SM [4] documentation.

Bidir PIM introduces the Bidir_Capable PIM-Hello option that MUST be included in all Hello messages from a Bidir-PIM capable router. The Bidir_Capable option advertises the router's ability to participate in the Bidir-PIM protocol. The format of the Bidir_Capable option is described in section 3.7.

If a Bidir PIM router receives a PIM-Hello message that does not contain the Bidir_Capable option from one of its neighbours, the error must be logged to the router administrator in a rate-limited manner.

3.3. Data Packet Forwarding Rules

For groups mapping to a given RPA, the following responsibilities are uniquely assigned to the DF for that RPA on each link:

- The DF is the only router that forwards packets traveling downstream onto the link.
- The DF is the only router that picks-up upstream traveling packets off the link to forward towards the RPL.

Non-DF routers on a link, that use that link as their RPF interface to reach the RPA, may perform the following forwarding actions for bidirectional groups:

- Forward packets from the link towards downstream receivers.
- Forward packets from downstream sources onto the link (provided they are the DF for the downstream link from which the packet was picked-up).

The BIDIR-PIM packet forwarding rules are defined below in pseudocode.

`iif` is the incoming interface of the packet.
`G` is the destination address of the packet (group address).
`RPA` is the Rendezvous Point Address for this group.

First we check to see whether the packet should be accepted based on TIB state and the interface that the packet arrived on. A packet is accepted if it arrives on the RPF_interface to reach the RPA (downstream traveling packet) or if the router is the DF on the interface the packet arrives (upstream traveling packet).

If the packet should be forwarded we build an outgoing interface list for the packet.

Finally we remove the incoming interface from the outgoing interface list we've created, and if the resulting outgoing interface list is not empty, we forward the packet out of those interfaces.

On receipt of data to G on interface iif:

```
if( iif == RPF_interface(RPA) || I_am_DF(RPA,iif) ) {
   oiflist = olist(G) (-) iif
   forward packet on all interfaces in oiflist
}
```

3.3.1. Upstream Forwarding at RP

When configuring a BIDIR-PIM domain it is possible to assign the Rendezvous Point Address (RPA) such that it does not belong to a physical box but instead is simply a routable address. Routers that have interfaces on the RPL that the RPA belongs to will upstream forward traffic onto the link. Joins from receivers in the domain will propagate hop-by-hop till they reach one of the routers connected to the RPL where they will terminate (as there will be no DF elected on the RPL).

If instead the administrator chooses to configure the RPA to be the address of a physical interface of a specific router then nothing changes. That router must still upstream forward traffic on to the RPL and behave no differently than any other router with an interface on the RPL.

To configure a BIDIR-PIM network to operate in a mode similar to that of PIM-SM where a single router (the RP) is acting as the root of the distribution tree, the RPA can be configured to be the loopback interface of a router.

3.3.2. Source-Only Branches

Source-only branches of the distribution tree for a group G are branches which do not lead to any receivers, but which are used to forward packets traveling upstream from sources towards the RPL. Routers along source-only branches only have the RPF_interface to the RPA in their olist for G and hence do not need to maintain any group specific state. Upstream forwarding can be performed using only RPA specific state. An implementation may decide to maintain group state for source-only branches for accounting or performance reasons. However, doing so requires data-driven events (to discover the groups with active sources) thus sacrificing one of the main benefits of Bidir PIM.

3.3.3. Directly Connected Sources

A major advantage of using a Designated Forwarder in BIDIR-PIM compared to PIM-SM is that special treatment is no longer required for sources that are directly connected to a router. Data from such sources does not need to be differentiated from other multicast traffic and will automatically be picked up by the DF and forwarded upstream. This removes the need for performing a directly-connected-source check for data to groups that do not have existing state.

3.4. PIM Join/Prune Messages

BIDIR-PIM Join/Prune messages are used to construct group specific distribution trees between receivers and the RPL. Joins are originated by last-hop routers that are elected as the DF on an interface with directly connected receivers. The Joins propagate hop-by-hop towards the RPA till they reach a router connected to the RPL.

A BIDIR-PIM Join/Prune message consists of a list of Joined and Pruned Groups. When processing a received Join/Prune message, each Joined or Pruned Group is effectively considered individually by applying the following state machines. When considering a Join/Prune message whose PIM Destination field addresses this router, (*,G) Joins and Prunes can affect the downstream state machine. When considering a Join/Prune message whose PIM Destination field addresses another router, most Join or Prune entries could affect the upstream state machine.

3.4.1. Receiving (*,G) Join/Prune Messages

When a router receives a Join(*,G) or Prune(*,G) it MUST first check to see whether the RP address in the message matches RPA(G) (the router's idea of what the Rendezvous Point Address is). If the RP address in the message does not match RPA(G) the Join or Prune MUST be silently dropped.

If a router has no RPA information for the group (e.g. has not recently received a BSR message) then it MAY choose to accept Join(*,G) or Prune(*,G) and treat the RP address in the message as RPA(G). If the newly discovered RPA did not previously exist for any other group, a DF election has to be initiated.

Note that a router will process a Join(*,G) targeted to itself even if it is not the DF for RP(G) on the interface on which the message was received. This is an optimisation to eliminate the Join delay of one Join period (t_periodic) in the case where a new DF processes the received Pass and Join messages in reverse order. The BIDIR-PIM forwarding logic will ensure that data packets are not forwarded on such an interface while the router is no the DF (unless it is the RPF_interface towards the RPA).

The per-interface state-machine for receiving (*,G) Join/Prune Messages is given below. There are three states:

NoInfo (NI)
: The interface has no (*,G) Join state and no timers running.

Join (J)
: The interface has (*,G) Join state. If the router is the DF on this interface (I_am_DF(RPA(G),I) is TRUE), the Join state will cause us to forward packets destined for G on this interface.

PrunePending (PP)
: The router has received a Prune(*,G) on this interface from a downstream neighbor and is waiting to see whether the prune will be overridden by another downstream router. For forwarding purposes, the PrunePending state functions exactly like the Join state.

In addition the state-machine uses two timers:

ExpiryTimer (ET)
: This timer is restarted when a valid Join(*,G) is received. Expiry of the ExpiryTimer causes the interface state to revert to NoInfo for this group.

PrunePendingTimer (PPT)
: This timer is set when a valid Prune(*,G) is received. Expiry of the PrunePendingTimer causes the interface state to revert to NoInfo for this group.

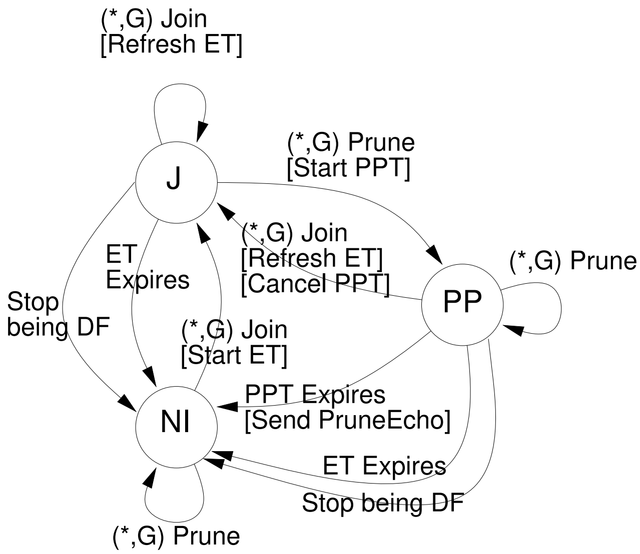

Figure 1: Downstream group per-interface state-machine

Downstream group per-interface state-machine in tabular form:

Event	Prev State		
	NoInfo (NI)	**Join (J)**	**Prune Pending (PP)**
Receive Join(*,G)	-> J state start Expiry Timer	-> J state restart Expiry Timer	-> J state restart Expiry Timer; stop Prune Pending Timer
Receive Prune(*,G)	-	-> PP state start Prune Pending Timer	-> PP state
Prune Pending Timer Expires	-	-	-> NI state Send Prune-Echo(*,G)
Expiry Timer Expires	-	-> NI state	-> NI state
Stop Being DF on I	-	-> NI state	-> NI state

The transition events "Receive Join(*,G)" and "Receive Prune(*,G)" imply receiving a Join or Prune targeted to this router's address on the received interface. If the destination address is not correct, these state transitions in this state machine must not occur, although seeing such a packet may cause state transitions in other state machines.

On unnumbered interfaces on point-to-point links, the router's address should be the same as the source address it chose for the Hello packet it sent over that interface. However, on point-to-point links we also RECOMMEND that PIM messages with a destination address of all zeros are also accepted.

The transition event "Stop being DF" implies a DF re-election taking place on this router interface for RPA(G) and the router changing status from being the active DF to being a non-DF router (the value of the `I_am_DF` macro changing to FALSE).

When ExpiryTimer is started or restarted, it is set to the HoldTime from the triggering received Join/Prune message.

When PrunePendingTimer is started, it is set to the J/P_Override_Interval if the router has more than one neighbor on that interface; otherwise it is set to zero causing it to expire immediately.

The action "Send PruneEcho(*,G)" is triggered when the router stops forwarding on an interface as a result of a prune. A PruneEcho(*,G) is simply a Prune(*,G) message sent by the upstream router to itself on a LAN. Its purpose is to add additional reliability so that if a Prune that should have been overridden by another router is lost locally on the LAN, then the PruneEcho may be received and cause the override to happen. A PruneEcho(*,G) need not be sent when the router has only one neighbour on the link.

3.4.2. Sending Join/Prune Messages

The downstream per-interface state-machines described above hold join state from downstream PIM routers. This state then determines whether a router needs to propagate a Join(*,G) upstream towards the RPA. Such Join(*,G) messages are sent on the RPF_interface towards the RPA and are targeted at the DF on that interface.

If a router wishes to propagate a Join(*,G) upstream, it must also watch for messages on its upstream interface from other routers on that subnet, and these may modify its behavior. If it sees a Join(*,G) to the correct upstream neighbor, it should suppress its own Join(*,G). If it sees a Prune(*,G) to the correct upstream neighbor, it should be prepared to override that prune by sending a Join(*,G) almost immediately. Finally, if it sees the Generation ID (see PIM-SM specification [4]) of the correct upstream neighbor change, it knows that the upstream neighbor has lost state, and it should be prepared to refresh the state by sending a Join(*,G) almost immediately.

In addition changes in the next hop towards the RPA trigger a prune off from the old next hop, and join towards the new next hop. Such a change can be caused by the following two events:

- The MRIB indicates that the RPF Interface towards the RPA has changed. In this case the DF on the new RPF_interface becomes the new RPF Neighbour.
- There is a DF re-election on the RPF_interface and a new router emerges as the DF.

The upstream (*,G) state-machine only contains two states:

Not Joined
The downstream state-machines indicate that the router does not need to join the RPA tree for this group.

Joined
The downstream state-machines indicate that the router would like to join the RPA tree for this group.

In addition, one timer JT(G) is kept which is used to trigger the sending of a Join(*,G) to the upstream next hop towards the RPA (the DF on the RPF_interface for RPA(G)).

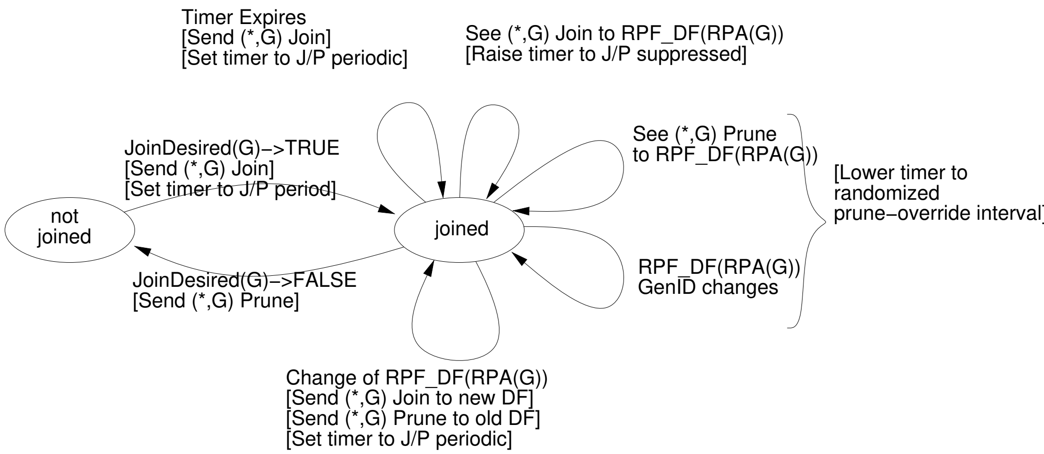

Figure 2: Upstream group state-machine

In tabular form, the state machine is:

Prev State	**Event**	
	JoinDesired(G) ->True	**JoinDesired(G) ->False**
NotJoined (NJ)	-> J state Send Join(*,G); Set Timer to t_periodic	-
Joined (J)	-	-> NJ state Send Prune(*,G)

In addition, we have the following transitions which occur within the Joined state:

In Joined (J) State			
Timer Expires	**See Join(*,G) to RPF_DF(RPA(G))**	**See Prune(*,G) to RPF_DF(RPA(G))**	**RPF_DF(RPA(G)) GenID changes**
Send Join(*,G); Set Timer to t_periodic	Increase Timer to t_suppressed	Decrease Timer to t_override	Decrease Timer to t_override

In Joined (J) State	
Change of RPF_DF(RPA(G))	**RPF_DF(RPA(G)) GenID changes**
Send Join(*,G) to new DF; Send Prune(*,G) to old DF; set Timer to t_periodic	Decrease Timer to t_override

This state machine uses the following macro:

```
bool JoinDesired(G) {
   if (olist(G) (-) RPF_interface(RPA(G))) != NULL
       return TRUE
   else
       return FALSE
}
```

3.5. Designated Forwarder (DF) Election

This section presents a fail-safe mechanism for electing a per-RPA designated router on each link in a BIDIR-PIM domain. We call this router the Designated Forwarder (DF). The DF election does not take place on the RPL for a RPA.

3.5.1. DF Requirements

The DF election chooses the best router on a link to assume the responsibility of forwarding traffic between the RPL and the link for the range of multicast groups served by the RPA. Different multicast groups that share a common RPA share the same upstream direction. Hence, the election of an upstream forwarder on each link does not have to be a group specific decision but instead can be RPA-specific. As the number of RPAs is typically small, the number of elections that have to be performed is significantly reduced by this observation.

To optimise tree creation, it is desirable that the winner of the election process should be the router on the link with the "best" unicast routing metric (as reported by the MRIB) to reach the RPA. When comparing metrics from different unicast routing protocols, we use the same comparison rules used by the PIM-SM assert process [4].

The election process needs to take place when information on a new RPA initially becomes available. The result can be re-used as new bidir groups that map to the same RPA are encountered. There are however some conditions under which an update to the election is required:

- There is a change in unicast metric to reach the RPA for any of the routers on the link.
- The interface on which the RPA is reachable (RPF Interface) changes to an interface for which the router was previously the DF.
- A new PIM neighbor starts up on a link that must participate in the elections and be informed of current outcome.
- The elected DF fails (detected through neighbor information timeout or MRIB RPF change at downstream router).

The election process has to be robust enough to ensure with very high probability that all routers on the link have a consistent view of the DF. This is because with the forwarding rules described in section 3.3 if multiple routers end-up thinking that they should be responsible for forwarding, loops may result. To reduce the possibility of this occurrence to a minimum, the election algorithm has been biased towards discarding DF information and suspending forwarding during periods of ambiguity.

3.5.2. DF Election description

This section gives an outline of the DF election process. It does not provide the definitive specification for the DF election. If any discrepancy exists between section 3.5.3 and this section, the specification in section 3.5.3 is to be assumed correct.

To perform the election of the DF for a particular RPA, routers on a link need to exchange their unicast routing metric information for reaching the RPA. Routers advertise their own metrics in Offer, Winner, Backoff and Pass messages. The advertised metric is calculated using the RPF Interface and metric to reach the RPA available through the MRIB. When a router is participating in a DF election for an RPA on the interface that its MRIB indicates as the RPF Interface then that router MUST always advertise an infinite metric in its election messages. When a router is participating in a DF election on an interface other than the MRIB indicated RPF Interface then it MUST advertise the MRIB provided metrics in its election messages.

In the election protocol described below, many message exchanges are repeated `Election_Robustness` times for reliability. In all those cases the message retransmissions are spaced in time by a small random interval. All of the following description is specific to the election on a single link for a single RPA.

3.5.2.1. Bootstrap Election

Initially when no DF has been elected, routers finding out about a new RPA start participating in the election by sending Offer messages. Offer messages include the router's metric to reach the RPA. Offers are periodically retransmitted with a period of `Offer_Interval`.

If a router hears a better offer than its own from a neighbor, it stops participating in the election for a period of `Election_Robustness * Offer_Interval` thus giving a chance to the neighbour with the better metric to be elected DF. If during this period no winner is elected, the router restarts the election from the beginning. If at any point during the initial election a router receives an out of order offer with worse metrics than its own, then it restarts the election from the beginning.

The result should be that all routers except the best candidate stop advertising their offers.

A router assumes the role of the DF after having advertised its metrics `Election_Robustness` times without receiving any offer from any other neighbor. At that point it transmits a Winner message which declares to every other router on the link the identity of the winner and the metrics it is using.

Routers receiving a winner message stop participating in the election and record the identity and metrics of the winner. If the local metrics are better than those of the winner then the router records the identity of the winner (accepting it as the acting DF) but re-initiates the election to try and take over.

3.5.2.2. Loser Metric Changes

Whenever the unicast metric to a RPA changes at a non-DF router to a value that is better than that previously advertised by the acting DF, the router with the new better metric should take action to eventually assume forwarding responsibility. When the metric change is detected, the non-DF router with the now better metric restarts the DF election process by sending Offer messages with this new metric. Note that at any point during an election if no response is received after `Election_Robustness` retransmissions of an offer, a router assumes the role of the DF following the usual Winner announcement procedure.

Upon receipt of an offer that is worse than its current metric, the DF will respond with a Winner message declaring its status and advertising its better metric. Upon receiving the Winner message, the originator of the Offer records the identity of the DF and aborts the election.

Upon receipt of an offer that is better than its current metric, the DF records the identity and metrics of the offering router and responds with a Backoff message. This instructs the offering router to hold off for a short period of time while the unicast routing stabilises and other routers get a chance to put in their offers. The Backoff message includes the offering router's new metric and address. All routers on the link that have pending offers with metrics worse than those in the backoff message (including the original offering router) will hold further offers for a period of time defined in the Backoff message.

If during the `Backoff_Period`, a third router sends a new better offer, the Backoff message is repeated for the new offer and the `Backoff_Period` restarted.

Before the `Backoff_Period` expires, the acting DF nominates the router having made the best offer as the new DF using a Pass message. This message includes the IDs and metrics of both the old and new DFs. The old DF stops performing its tasks at the time the Pass message transmission is made. The new DF assumes the role of the DF as soon as it receives the Pass message. All other routers on the link take note of the new DF and its metric. Note that this event constitutes an RPF Neighbour change which may trigger Join messages to the new DF (see section 3.4).

3.5.2.3. Winner Metric Changes

If the DF's routing metric to reach the RPA changes to a worse value, it sends a set of `Election_Robustness` randomly spaced Winner messages on the link, advertising the new metric. Routers that receive this announcement but have a better metric may respond with an Offer message which results in the same handoff procedure described above. All routers assume the DF has not changed until they see a Pass or Winner message indicating the change.

There is no pressure to make this handoff quickly if the acting DF still has a path to the RPL. The old path may now be suboptimal but it will still work while the re-election is in progress.

3.5.2.4. Winner Loses Path

If a router's RPF Interface to the RPA switches to be on a link for which it is acting as the DF, then it can no longer provide forwarding services for that link. It therefore immediately stops being the DF and restarts the election. As its path to the RPA is through the link, an infinite metric is used in the Offer message it sends.

3.5.2.5. Late Router Starting Up

A late router starting up after the DF election process has completed will have no immediate knowledge of the election outcome. As a result, it will start advertising its metric in Offer messages. As soon as this happens, the currently elected DF will respond with a Winner message if its metric is better than the metric in the Offer message, or with a Backoff message if its metric is worse than the metric in the Offer message.

3.5.2.6. Winner Dies

Whenever the DF dies, a new DF has to be elected. The speed at which this can be achieved depends on whether there are any downstream routers on the link.

If there are downstream routers, typically their MRIB reported next-hop before the DF dies will be the DF itself. They will therefore notice either a change in the metric for the route to the RPA or a change in next-hop away from the DF and can restart the election by transmitting Offer messages. If according to the MRIB the RPA is now reachable through the same link via another upstream router, an infinite metric will be used in the Offer.

If no downstream routers are present, the only way for other upstream routers to detect a DF failure is by the timeout of the PIM neighbor information, which will take significantly longer.

3.5.3. Election Protocol Specification

This section provides the definitive specification for the DF election process. If any discrepancy exists between section 3.5.2 and this section, the specification in this section is to be assumed correct.

3.5.3.1. Election State

The DF election state is maintained per RPA for each multicast enabled interface I on the router as introduced in section 3.1.

The state machine has the following four states:

`Offer`
: Initial election state. When in the Offer state a router thinks it can eventually become the winner and periodically generates Offer messages.

`Lose`
: In this state the router knows that there either is a different election winner or that no router on the link has a path to the RPA.

`Winner`
: The router is the acting DF without any contest.

`Backoff`
: The router is the acting DF but another router has made a bid to take over.

In the state machine a router is considered to be an acting DF if it is in the `Win` or `Backoff` states.

The operation of the election protocol makes use of the variables and timers described below:

`Acting DF information`
: Used to store the identity and advertised metrics of the election winner that is the currently acting DF.

`DF election-Timer (DFT)`
: Used to schedule transmission of Offer, Winner and Pass messages.

`Message-Count (MC)`
: Used to maintain the number of times an Offer or Winner message has been transmitted.

`Best-Offer`
: Used by the DF to record the identity and advertised metrics of the router that has made the last offer, for use when sending the Path message.

3.5.3.2. Election Messages

The election process uses the following PIM control messages, the packet format of which is described in section 3.7:

`Offer (OfferingID, Metric)`
: Sent by routers that believe they have a better metric to the RPA than the metric that has been on offer so far.

`Winner (DF-ID, DF-Metric)`
: Sent by a router when assuming the role of the DF or when re-asserting in response to worse offers.

`Backoff (DF-ID, DF-Metric, OfferingID, OfferMetric, BackoffInterval)`

Used by the DF to acknowledge better offers. It instructs other routers with equal or worse offers to wait till the DF passes responsibility to the sender of the offer.

`Pass (Old-DF-ID, Old-DF-Metric, New-DF-ID, New-DF-Metric)`

Used by the old DF to pass forwarding responsibility to a router that has previously made an offer. The Old-DF-Metric is the current metric of the DF at the time the pass is sent.

Note that when a router is participating in a DF election for an RPA on the interface that its MRIB indicates as the RPF Interface then that router MUST always advertise an infinite metric in its election messages. When a router is participating in a DF election on an interface other than the MRIB indicated RPF Interface then it MUST advertise the MRIB provided metrics in its election messages.

3.5.3.3. Election Events

During protocol operation the following events can take place:

`Control message reception`

Reception of one of the four control DF election messages (Offer, Winner, Backoff and Pass). When a control message is received and actions are specified on a condition that metrics are Better or Worse the comparison must be performed as follows:

- On receipt of an Offer or Winner message compare our current metrics for the RPA with the metrics advertised for the sender of the message.
- On receipt of a Backoff or Pass message compare our current metrics for the RPA with the metrics advertised for the target of the message.

`Path to RPA lost`

Losing the path to the RPA can happen in two ways. The first happens when the route learned through the MRIB is withdrawn and the MRIB no longer reports an available route to reach the RPA. The second case happens when the next-hop information reported by the MRIB changes to indicate a next-hop that is reachable through the router interface under consideration. Clearly as the router is using the interface as its RPF Interface it cannot offer forwarding services towards the RPL to other routers on that link.

`Metric reported by the MRIB to reach the RPA changes`

This event is triggered when the MRIB supplied information for the RPA changes and the new information provides a path to the RPA. If the new MRIB information either reports no route or reports a next-hop interface through the interface for which the DF election is taking place then the "Path to RPA lost" event triggers instead. In specific states the event may be further filtered by specifying whether it is expected of the metric to become better or worse and which stored metric the new MRIB information must be compared against. The new information must be compared with either the router's old metric, the stored DF metric or the stored Best Offer metric.

`Election-Timer (DFT) Expiration`

Expiration of the DFT election timer can cause message transmission and state transitions. The event might be further qualified by specifying the value of the Message Count (MC) as well as the current existence of a path to the RPA (as defined above).

`Detection of DF failure`

Detection of DF failure can occur through the timeout of PIM neighbor state.

3.5.3.4. Election Actions

The DF election state machine action descriptions use the following notation in addition to the pseudocode notation described earlier in this spec.

?= denotes the operation of lowering a timer to a new value. If the timer is not running then it is started using the new value. If the timer is running with an expiration lower than the new value, then the timer is not altered.

When an action of "set DF to Sender or Target" is encountered during receipt of a Winner, Pass or Backoff message it means the following:

- On receipt of a Winner message set the DF to be the originator of the message and record its metrics.
- On receipt of a Pass message set the DF to be the target of the message and record its metrics.
- On receipt of a Backoff message set the DF to be the originator of the message and record its metrics.

3.5.3.5. Election State Transitions

When a Designated Forwarder election is initiated the starting state is the Offer state, the message counter (MC) is set to zero and the DF election Timer (DFT) is set to OPlow (see section 3.6 for a definition of timer values).

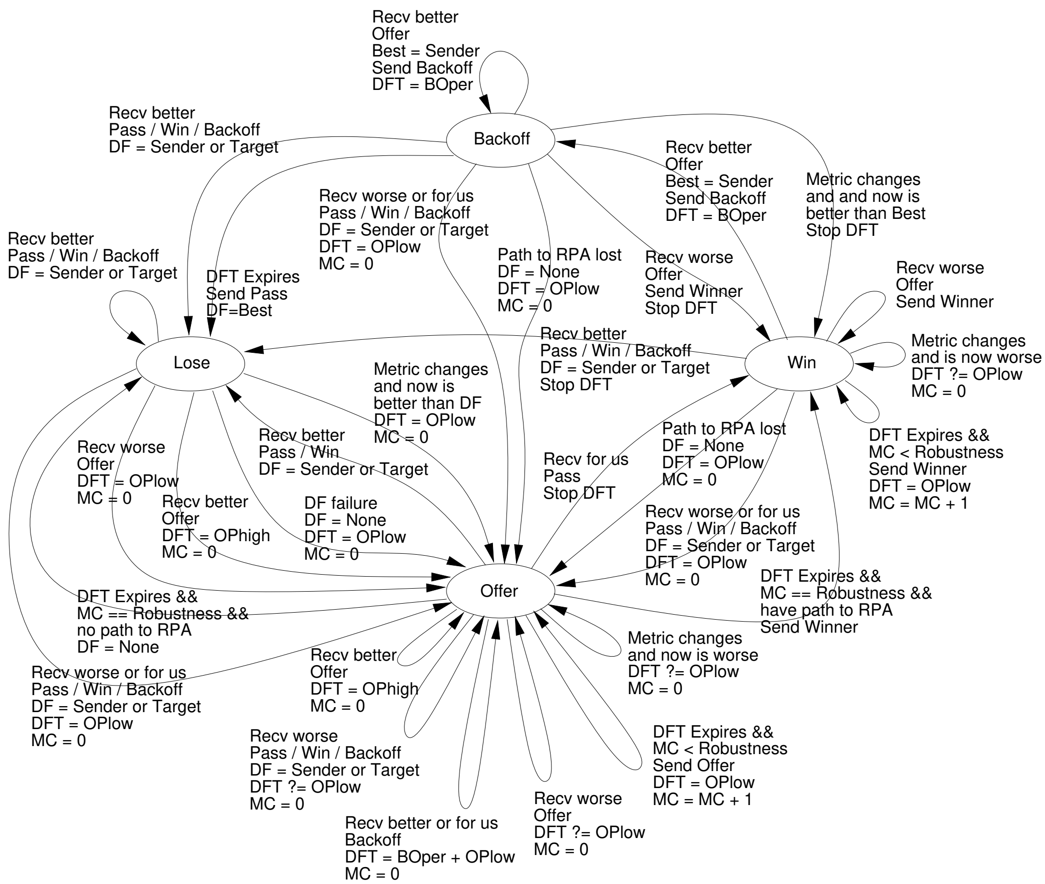

Figure 3: Designated Forwarder election state-machine

Designated Forwarder election state-machine in tabular form:

Prev State	Event: Recv better Pass / Win	Event: Recv better Backoff	Event: Recv better Offer
Offer	-> Lose DF = Sender or Target; Stop DFT	- DFT = BOperiod + OPlow; MC = 0	- DFT = OPhigh; MC = 0
Lose	- DF = Sender or Target	- DF = Sender	-> Offer DFT = OPhigh; MC = 0
Win	-> Lose DF = Sender or Target; Stop DFT	-> Lose DF = Sender; Stop DFT	-> Backoff Set Best to Sender; Send Backoff; DFT = BOperiod
Backoff	-> Lose DF = Sender or Target; Stop DFT	-> Lose DF = Sender; Stop DFT	- Set Best to Sender; Send Backoff; DFT = BOperiod

Prev State	Event: Recv Backoff for us	Event: Recv Pass for us	Event: Recv Worse Pass / Win / Backoff	Event: Recv worse Offer
Offer	- DFT = BOperiod + OPlow; MC = 0	-> Win Stop DFT	- Set DF to Sender or Target; DFT ?= OPlow; MC = 0	- DFT ?= OPlow; MC = 0
Lose	-> Offer DF = Sender; DFT = OPlow; MC = 0	-> Offer DF = Sender; DFT = OPlow; MC = 0	-> Offer DF = Sender or Target; DFT = OPlow; MC = 0	-> Offer DFT = OPlow; MC = 0
Win	-> Offer DF = Sender; DFT = OPlow; MC = 0	-> Offer DF = Sender; DFT = OPlow; MC = 0	-> Offer DF = Sender or Target; DFT = OPlow; MC = 0	- Send Winner
Backoff	-> Offer DF = Sender; DFT = OPlow; MC = 0	-> Offer DF = Sender; DFT = OPlow; MC = 0	-> Offer DF = Sender or Target; DFT = OPlow; MC = 0	-> Win Send Winner; Stop DFT

In Offer State		
DFT Expires and MC is less than Robustness	**DFT Expires and MC is equal to Robustness and we have path to RPA**	**DFT Expires and MC is equal to Robustness and there is no path to RPA**
- Send Offer; DFT = OPlow; MC = MC + 1	-> Win Send Winner	-> Lose Set DF to None

In Offer State
Metric changes and is now worse
DFT ?= OPlow MC = 0

In Lose State	
Detect DF Failure	**Metric changes and now is better than DF**
-> Offer DF = None; DFT = OPlow_int; MC = 0	-> Offer DFT = OPlow_int; MC = 0

In Win State		
Metric changes and is now worse	**Timer Expires and MC is less than Robustness**	**Path to RPA lost**
- DFT = OPlow; MC = 0	- Send Winner; DFT = OPlow; MC = MC + 1	-> Offer Set DF to None; DFT = OPlow; MC = 0

In Backoff State		
Metric changes and is now better than Best	**Timer Expires**	**Path to RPA lost**
-> Win Stop Timer	-> Lose Send Pass; Set DF to stored Best	-> Offer Set DF to None; DFT = OPlow; MC = 0

3.5.4. Election Reliability Enhancements

For the correct operation of BIDIR-PIM it is very important to avoid situations where two routers consider themselves to be Designated Forwarders for the same link. The two precautions below are not required for correct operation but can help diagnose anomalies and correct them.

3.5.5. Missing Pass

After a DF has been elected, a router whose metrics change to become better than the DF will attempt to take over. If during the re-election the acting DF has a condition that causes it to lose all of the election messages (like a CPU overload), the new candidate will transmit three offers and assume the role of the forwarder resulting in two DFs on the link. This situation is pathological and should be corrected by fixing the overloaded router. It is desirable that such an event can be detected by a network administrator.

When a router becomes the DF for a link without receiving a Pass message from the known old DF, the PIM neighbor information for the old DF can be marked to this effect. Upon receiving the next PIM Hello message from the old DF, the router can retransmit Winner messages for all the RPAs for which it is acting as the DF. The anomaly may also be logged by the router in a rate-limited manner to alert the operator.

3.5.6. Periodic Winner Announcement

An additional degree of safety can be achieved by having the DF for each RPA periodically announce its status in a Winner message. Transmission of the periodic Winner message can be restricted to occur only for RPAs which have active groups, thus avoiding the periodic control traffic in areas of the network without senders or receivers for a particular RPA.

3.6. Timers, Counters and Constants

BIDIR-PIM maintains the following timers, as discussed in section 3.1. All timers are countdown timers - they are set to a value and count down to zero, at which point they typically trigger an action. Of course they can just as easily be implemented as count-up timers, where the absolute expiry time is stored and compared against a real-time clock, but the language in this specification assumes that they count downwards to zero.

Per Rendezvous-Point Address (RPA):

- Per interface (I):
  - **DF Election Timer: DFT(RPA,I)**

Per Group (G):

- **Upstream Join Timer: JT(G)**
- Per interface (I):
  - **Join Expiry Timer: ET(G,I)**
  - **PrunePending Timer: PPT(G,I)**

When timers are started or restarted, they are set to default values. This section summarizes those default values.

Timer Name: DF Election Timer (DFT)

Value Name	Value	Explanation
Offer_Period	100 ms	Interval to wait between repeated Offer and Winner messages.
Backoff_Period	1 sec	Period that acting DF waits between receiving a better Offer and sending the Pass message to transfer DF responsibility.
OPlow	rand(0.5, 1) * Offer_Period	Range of actual randomised value used between repeated messages.
OPhigh	Election_Robustness * Offer_Period	Interval to wait in order to give a chance to a router with a better Offer to become the DF.

Timer Names: Join Expiry Timer (ET(G,I))

Value Name	Value	Explanation
J/P HoldTime	from message	Hold Time from Join/Prune Message

Timer Names: Prune Pending Timer (PPT(G,I))

Value Name	Value	Explanation
J/P Override Interval	Default: 3 secs	Short period after a join or prune to allow other routers on the LAN to override the join or prune

Note that the value of the J/P Override Interval is interface specific and depends on both the Propagation_Delay and the Override_Interval values that may change when Hello messages are received [4].

Timer Names: Upstream Join Timer (JT(G))

Value Name	Value	Explanation
t_periodic	Default: 60 secs	Period between Join/Prune Messages
t_suppressed	rand(1.1 * t_periodic, 1.4 * t_periodic)	Suppression period when someone else sends a J/P message so we don't need to do so.
t_override	rand(0, 0.9 * J/P Override Interval)	Randomized delay to prevent response implosion when sending a join message to override someone else's prune message.

For more information about these values refer to the PIM-SM [4] documentation.

Constant Name: DF Election Robustness

Constant Name	Value	Explanation
Election_Robustness	Default: 3	Minimum number of election messages that must be lost in order for election to fail.

3.7. BIDIR-PIM Packet Formats

This section describes the details of the packet formats for BIDIR-PIM control messages. BIDIR-PIM shares a number of control messages in common with PIM-SM [4]. These include the Hello and Join/Prune messages as well as the format for the Encoded-Unicast address. For details on the format of these packets please refer to the PIM-SM documentation. Here we will only define the additional packets that are introduced by BIDIR-PIM. These are the packets used in the DF election process as well as the Bidir_Capable PIM-Hello option.

3.7.1. DF Election Packet Formats

All PIM control messages have IP protocol number 103.

BIDIR-PIM messages are multicast with TTL 1 to the 'ALL-PIM-ROUTERS' group The IPv4 'ALL-PIM-ROUTERS' group is '224.0.0.13'. The IPv6 'ALL-PIM-ROUTERS' group is 'ff02::d'.

All DF election BIDIR-PIM control messages share the common header below:

```
 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  |Subtype| Rsvd  |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Encoded-Unicast-RP-Address
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-...
|                   Sender Metric Preference                    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sender Metric                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PIM Ver
PIM Version number is 2.

Type
All DF-Election PIM control messages share the PIM message Type of 10.

Subtype
Subtypes for DF election messages are:

1 = Offer
2 = Winner
3 = Backoff
4 = Pass

Rsvd
Set to zero on transmission. Ignored upon receipt.

Checksum
The checksum is standard IP checksum, i.e. the 16-bit one's complement of the one's complement sum of the entire PIM message. For computing the checksum, the checksum field is zeroed.

RP-Address
The bidir RPA for which the election is taking place (note that the length of this field will be different than 32 bits depending on the family and encoding of the address).

Sender Metric Preference
Preference value assigned to the unicast routing protocol that the message sender used to obtain the route to the RPA.

Sender Metric
The unicast routing table metric used by the message sender to reach the RPA. The metric is in units applicable to the unicast routing protocol used.

In addition to the fields defined above the Backoff and Pass messages have the extra fields described below.

3.7.2. Backoff Message

The Backoff message uses the following fields in addition to the common election message format described above.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Encoded-Unicast-Offering-Address
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...
|                   Offering Metric Preference                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Offering Metric                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Interval            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Offering Address
The address of the router that made the last (best) Offer (note that the length of this field will be different than 32 bits depending on the family and encoding of the address).

Offering Metric Preference
Preference value assigned to the unicast routing protocol that the offering router used to obtain the route to the RPA.

Offering Metric
The unicast routing table metric used by the offering router to reach the RPA. The metric is in units applicable to the unicast routing protocol used.

Interval
The backoff interval in milliseconds to be used by routers with worse metrics than the offering router.

3.7.3. Pass Message

The Pass message uses the following fields in addition to the common election fields described above.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Encoded-Unicast-New-Winner-Address
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+...
|                  New Winner Metric Preference                 |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       New Winner Metric                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

New Winner Address
The address of the router that made the last (best) Offer (note that the length of this field will be different than 32 bits depending on the family and encoding of the address).

New Winner Metric Preference
Preference value assigned to the unicast routing protocol that the offering router used to obtain the route to the RPA.

New Winner Metric
The unicast routing table metric used by the offering router to reach the RPA. The metric is in units applicable to the unicast routing protocol used.

3.7.4. Bidir Capable PIM-Hello Option

BIDIR-PIM introduces one new PIM-Hello option.

- OptionType 22: Bidir 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 = 22           |          Length = 0           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4. RP Discovery

Routers discover that a range of multicast group addresses operates in bi-directional mode and the address of the Rendezvous-Point address (RPA) serving the group range either through static configuration or using an automatic RP discovery mechanism like the PIM Bootstrap mechanism (BSR) [9] or Auto-RP.

5. Security Considerations

The IPsec [5] authentication header MAY be used to provide data integrity protection and group-wise data origin authentication of BIDIR-PIM protocol messages. Authentication of BIDIR-PIM messages can protect against unwanted behaviour caused by unauthorized or altered BIDIR-PIM messages.

5.1. Attacks Based on Forged Messages

As in PIM Sparse-Mode, the extent of possible damage depends on the type of counterfeit messages accepted. BIDIR-PIM only uses link-local multicast messages sent to the ALL_PIM_ROUTERS address, hence attacks can only be carried out by directly connected nodes, or with the complicity of directly connected routers.

Some of the BIDIR-PIM protocol messages (Join/Prune and Hello) are identical, both in format and functionality, to the respective messages used in PIM-SM. Security considerations for these messages are to be found in [4]. Other messages (DF-election messages) are specific to BIDIR-PIM and will be discussed in the following paragraphs.

By forging DF-election messages an attacker can disrupt the election of the Designated Forwarder on a link in two different ways:

5.1.1. Election of an Incorrect DF

An attacker can force its election as DF by participating in a regular election and advertising the best metric to reach the RPA. An attacker can also try to force the election of another router as DF by sending an Offer, Winner or Pass message and impersonating another router. In some cases (e.g. the Offer) multiple messages might be needed to carry out an attack.

In the case of Offer or Winner messages the attacker will have to impersonate the node that it wants to have become the DF. In the case of the Pass it will have to impersonate the current DF. This type of attack causes the wrong DF to be recorded in all nodes apart from the one that is being impersonated. This node typically will be able to detect the anomaly and, possibly, restart a new election.

A more sophisticated attacker might carry out a concurrent DoS attack on the node being impersonated, so that it will not be able to detect the forged packets and/or take countermeasures.

All attacks based on impersonation can be detected by all routers and avoided if the source of DF-election messages can be authenticated. When authentication is available, spoofed messages MUST be discarded and a rate-limited warning message SHOULD be logged.

A more subtle attacker could use MAC-level addresses to partition the set of recipients of DF-election messages and create an inconsistent DF view on the link. For example the attacker could use unicast MAC addresses for its forged DF-election messages. To prevent this type of attack, BIDIR-PIM routers SHOULD check the destination MAC address of received DF-election messages. This however is ineffective on links that do not support layer-2 multicast delivery.

Source authentication is also sufficient to prevent this kind of attack.

5.1.2. Preventing Election Convergence

By forging DF election messages, an attacker can prevent the election from converging thus disrupting the establishment of multicast forwarding trees. There are many ways to achieve this. The simplest is by sending an infinite sequence of Offer messages (the metric used in the messages is not important).

5.2. Non-cryptographic Authentication Mechanisms

A BIDIR-PIM router SHOULD provide an option to limit the set of neighbors from which it will accept Join/Prune, Assert, and DF-election messages. Either static configuration of IP addresses or an IPsec security association may be used. Furthermore, a PIM router SHOULD NOT accept protocol messages from a router from which it has not yet received a valid Hello message.

5.2.1. Basic Access Control

In a PIM-SM domain, when all routers are trusted, it is possible to implement a basic form of access control for both sources and receivers: Receivers can be validated by the last-hop DR and sources can be validated by the first-hop DR and/or the RP.

In BIDIR-PIM this is generally feasible only for receivers, as sources can send to the multicast group without the need for routers to detect their activity and create source-specific state. However it is possible to modify the standard BIDIR-PIM behaviour, in a backward compatible way, to allow per-source access control. The tradeoff would be protocol simplicity, memory and processing requirements.

5.3. Authentication Using IPsec

Just like for PIM-SM, the IPsec [5] transport mode using the Authentication Header (AH) is the recommended method to prevent the above attacks against BIDIR-PIM.

It is recommended that IPsec authentication be applied to all BIDIR-PIM protocol messages. The specification on how this is done is to be found in [4]. specifically the authentication of PIM-SM link-local messages, described in [4] applies to all BIDIR-PIM messages as well.

5.4. Denial of Service Attacks

The denial of service attack based on forged Join described in [4] also apply to BIDIR-PIM.

6. Change history

From 06 to 07: Minor corrections in response to WG last call comments.

From 05 to 06:

Minor editorial corrections.

From 03 to 05:

RP concept replaced by RP Address (RPA) and RP Link (RPL). No DF election on RPL. RP forwards upstream on RPL. Accept joins even if not DF but do not forward. Added event description for DF election state machine. Security considerations by Lorenzo.Removed comparison with Dino's draft.

From 02 to 03:

Consistency fixes in DF election tables to match state transition diagram pointed out by Apoorva.

From 00 to 01:

The differences between this version (-01) of the BIDIR-PIM specification and draft-ietf-pim-bidir-new-00.txt are mostly in the format of the information presented. As BIDIR-PIM has many similarities in operation to Sparse-Mode PIM, the earlier version of this spec relied heavily on the now obsolete PIM-SM [8] specification. This revision removes this dependency and instead references the new Sparse-Mode documentation [4] where necessary. In addition the method in which the protocol specification is presented has been updated to follow the format of [4].

7. Acknowledgments

The bidir proposal in this draft is heavily based on the ideas and text presented by Estrin and Farinacci in [7]. The main difference between the two proposals is in the method chosen for upstream forwarding.

We would also like to thank John Zwiebel at Procket, Deborah Estrin at ISI/USC as well as Nidhi Bhaskar, Yiqun Cai, Toerless Eckert, Apoorva Karan, Rajitha Sumanasekera and Beau Williamson at cisco for their contributions and comments to this draft.

8. Authors' Addresses

Mark Handley
Computer Science Department
University College London
M.Handley@cs.ucl.ac.uk

Isidor Kouvelas
Cisco Systems
kouvelas@cisco.com

Tony Speakman
Cisco Systems
speakman@cisco.com

Lorenzo Vicisano
Cisco Systems
lorenzo@cisco.com

9. Normative References

[1] S.E. Deering, "Host extensions for IP multicasting", RFC 1112, Aug 1989.

[2] B. Cain, S Deering, W. Fenner, I Kouvelas, A. Thyagarajan, "Internet Group Management Protocol, Version 3", RFC 3376.

[3] S. Deering, W. Fenner, B. Haberman, "Multicast Listener Discovery (MLD) for IPv6", RFC 2710.

[4] B. Fenner, M. Handley, H. Holbrook, I. Kouvelas "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", Work In Progress, <draft-ietf-pim-sm-v2-new-09.txt>, 2004.

[5] S. Kent, R. Atkinson, "Security Architecture for the Internet Protocol.", RFC 2401.

10. Informative References

[6] T. Bates , R. Chandra , D. Katz , Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 2283

[7] D. Estrin, D. Farinacci, "Bi-directional Shared Trees in PIM-SM", <draft-farinacci-bidir-pim-01.txt>, May 1999.

[8] D. Estrin et al, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", RFC 2362, Nov 1999.

[9] W. Fenner, M. Handley, R. Kermode and D. Thaler, "Bootstrap Router (BSR) Mechanism for PIM Sparse Mode", Work in progress <draft-ietf-pim-sm-bsr-03.txt>, 2003.

11. Index