IPv4 multicast and broadcast over InfiniBand networks

Status of this memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC 2026.

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

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

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

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

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2001). All Rights Reserved.

Abstract

This document specifies a method for the transmission of IP version 4 multicast and broadcast datagrams over InfiniBand subnets.

Table of Contents

1.0 Introduction
2.0 InfiniBand addresses
2.1 Unicast GIDs
2.2 Multicast GIDs
2.3 InfiniBand Multicast group management
3.0 IPv4 on IB multicast fabrics
3.1 Scope bits
3.1.1 Options for implementing IPv4 subnets spanning multiple IB subnets
3.2 Flag bits
3.3 Mapping of IPv4 multicast address to IB address
3.3.1 IPv4 multicast addresses spanning multiple IB subnets
4.0 IPv4 multicast address to LID mapping
5.0 Guidelines on setup of IB multicast groups
6.0 Outline description of multicast on IPoIB subnets
6.1 IPv4 broadcast addresses
6.2 Receiving/forwarding multicast packets
6.3 Sending IPv4 multicast datagrams
6.4 Leaving/Deleteing a multicast group
7.0 Security considerations
8.0 Acknowledgement
9.0 References
10.0 Author’s Address
11.0 Full Copyright statement

1.0 Introduction

IPv4 multicasting provides a means of transmitting IPv4 datagrams to a group of interfaces. A group IPv4 address is used as the destination address in the IPv4 datagram as documented in STD 5, RFC 1112 [1]. Standard mappings are defined for various media types e.g. ethernet [1], fddi RFC1188 [2], and token ring RFC 1469[3] etc. IPv4 broadcast address is used to send packets to all the IPv4 nodes in the specific IPv4 network.

The address range of the multicast addresses is 224.0.0.0 to 239.255.255.255. The limited broadcast address is 255.255.255.255. The net broadcast address is <net_id><-1> or <net_id><subnet_id><-1>.

This document defines the mappings for IPv4 multicast and broadcast addresses to the InfiniBand multicast group addresses.

This document addresses the issues wrt IPv4. It further assumes unreliable datagram and raw datagram services of InfiniBand Architecture(IBA). These services are described in InfiniBand architecture specification [4]. For a concise overview of the InfiniBand architecture refer to draft-kashyap-ipoib-requirements-00.txt [5].
IPv6 multicast over datagram service of IBA will be described in a subsequent document.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

This document utilizes the text representations described in RFC2373[6] for both the IPv6 and InfiniBand (IB) addresses.

2.0 InfiniBand addresses

The InfiniBand architecture borrows heavily from the IPv6 architecture in terms of the InfiniBand subnet structure and global identifiers (GIDs).

The InfiniBand architecture defines the global identifier associated with a port as follows:

GID (Global Identifier): A 128-bit unicast or multicast identifier used to identify a port on a channel adapter, a port on a router, a switch, or a multicast group. A GID is a valid 128-bit IPv6 address (per RFC 2373) with additional properties/restrictions defined within IBA to facilitate efficient discovery, communication, and routing. Note: These rules apply only to IBA operation and do not apply to raw IPv6 operation unless specifically called out.

The raw IPv6 operation referred to in the note in the definition above is the IPv6 mode of InfiniBand’s raw datagram service. It does not mean IPv6 itself. The routers and switches referred to in the above definition are the InfiniBand routers and switches.

The InfiniBand(IB) specification defines two types of GIDs: unicast and multicast.

2.1 Unicast GIDs

The unicast GIDs are defined, as in IPv6, with three scopes. The IB specification states:

a. link local: This is defined to be FE80/10. The IB routers will not forward packets with a link local address in source or destination beyond the IB subnet.
b. site local: FEC0/10
   A unicast GID used within a collection of subnets which is unique within that collection (e.g. a data center or campus) but is not necessarily globally unique. IB routers must not forward any packets with either a site-local Source GID or a site-local Destination GID outside of the site.

c. global: A unicast GID with a global prefix, i.e. an IB router may use this GID to route packets throughout an enterprise or internet.

2.2 Multicast GIDs

The multicast GIDs also parallel the IPv6 multicast addresses. The IB specification defines the multicast GIDs as follows:

FFxy:<112 bits>

Flag bits:

The nibble, denoted by x above, are the 4 flag bits: 000T. The first three bits are reserved and are set to zero. The last bit is defined as follows:

T=0: denotes a permanently assigned i.e. well known GID
T=1: denotes a transient group

Scope bits:

The 4 bits, denoted by y in the GID above, are the scope bits. These are defined as:

<table>
<thead>
<tr>
<th>scope value</th>
<th>Address value</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Reserved</td>
</tr>
<tr>
<td>1</td>
<td>Unassigned</td>
</tr>
<tr>
<td>2</td>
<td>Link-local</td>
</tr>
<tr>
<td>3</td>
<td>Unassigned</td>
</tr>
<tr>
<td>4</td>
<td>Unassigned</td>
</tr>
<tr>
<td>5</td>
<td>Site-local</td>
</tr>
<tr>
<td>6</td>
<td>Unassigned</td>
</tr>
<tr>
<td>7</td>
<td>Unassigned</td>
</tr>
<tr>
<td>8</td>
<td>Organization-local</td>
</tr>
<tr>
<td>9</td>
<td>Unassigned</td>
</tr>
<tr>
<td>0xA</td>
<td>Unassigned</td>
</tr>
</tbody>
</table>
The IB specification further refers to RFC2373 [6] and RFC2375[7] while defining the well known multicast addresses. However, it then states that the well known addresses apply to IB raw IPv6 datagrams only. The IB unreliable datagram (UD) service recognises only one well known multicast address. This is the ALL_CHANNEL_ADAPTERS multicast address defined to be FF02::1. The scope of this address is limited to a single IB subnet.

2.3 InfiniBand Multicast group management

IB multicast groups (multicast GIDs) are managed by the subnet manager(SM). The SM explicitly programs the IB switches in the fabric to ensure that the packets are received by all the members of the multicast group.

When the group is created a create request is sent to the SM. The subnet manager records the group GIDs and the associated characteristics. The group characteristics are defined by the group path MTU, whether the group will be used for raw datagrams or unreliable datagrams, the service level, the partition key associated with the group, the LID (local identifier) associated with the group etc. These characteristics are defined at the time of the group creation.

The LID is a 16-bit value, valid only within an IB subnet, that is associated with the multicast group by the subnet manager(SM) at the time of the multicast group creation. An IB node may to request a specific LID be associated with a group. The SM determines the multicast tree based on all the group members and programs the relevant switches. The LID is used by the switches to route the packets.

Any member IB node wanting to participate in the group must join the group. As part of the join operation the node is returned the group characteristics. At the same time the subnet manager ensures that the requester can indeed participate in the group by verifying that it can support the group MTU, and accessibility to the rest of the group members. Other group
characteristics may need verification too.

The SM, for groups that span IB subnet boundaries, must interact with IB routers to determine the presence of this group in other IB subnets. If present the MTU must match across the IB subnets.

P_Key is another characteristic that must match across IB subnets since the P_Key inserted into a packet is not modified by the IB switches or IB routers. Thus if the P_Keys didn’t match the IB router(s) itself might drop the packets or destinations on other subnets might drop the packets.

These characteristics are returned to the IB endnode that joins the multicast group. A join operation may cause the SM to reprogram the fabric so that the new member can participate in the multicast group.

3.0 IPv4 on IB multicast fabrics

The InfiniBand architecture defines multiple transport methods to communicate between the IB endnodes. However, only two of these methods support multicast. These are the IB unreliable datagram (UD) service and the IB raw datagram service. Of the two the raw IB datagram service is optional. The UD service is the only service that all IB end nodes must support.

The IPv4 on InfiniBand multicast implementation are RECOMMENDED to use the UD service of IB.

The IB specification does not make multicast support mandatory though. Thus, in some IP subnets the multicast service MAY have to be implemented using a multicast server or some other method. It is RECOMMENDED that the IPv4 implementations on IB are implemented on fabrics that support multicast.

Note that the mappings defined in this document however are not effected by the above choices. The IPv4 broadcast and multicast to IB multicast GIDs are applicable to any IPv4 over InfiniBand network.

3.1 Scope bits

The IB multicast GID scope is as defined in table 1. The use of local scope will confine the IB multicast packets to an IB subnet.

The local scope at IB level conflicts with the requirement of an IP broadcast address if IP subnets can span across IB
subnets. The IP broadcast address will need to be mapped to an
IB GID that has a greater scope than the IB subnet.

Extending the IB router to bridge such packets (using
local-scope GIDs) across IB subnets suggests an extension to
the IB specification. It also makes an IP limited broadcast
address extend across all of the connected IB subnets if any
of them has an IPv4 subnet in them since all IPv4 subnets
support the broadcast address (multicast is optional).

The alternative of using the global scope has the same result
of extending the group across all the connected IB subnets
when IPv4 subnets are created in the IB subnets.

This brings in the associated administrative difficulties of
ensuring common MTUs and P_Keys across IB subnets for
implementing the IP multicast groups. The IB multicast groups
also take other values such as the TClass, HopLimit, FlowId
e tc. that will also cause administrative hiccups in ensuring
that they are consistent across IB subnets participating in
the group.

The use of any other scope is not well defined in IB
specification.

Therefore, in the interest of simplicity, it is RECOMMENDED
that the IPv4 multicast and broadcast addresses be mapped to
link-local scope IB multicast GIDs. It is further RECOMMENDED
that the IPv4 subnets implementations do not span multiple IB
subnets. IPv4 subnetting can be used to span a particular IPv4
subnet with a shorter mask across multiple IB subnets.

Note that the IB multicast group takes a hop limit. However,
setting a hop limit in the SM doesn’t limit the span of the
multicast group. The hop limit only specifies the hop limit
that the packets sent out by end nodes must use. Secondly, the
IB subnets may be only 1 subnet away but have multiple of them
linked by multiple or the same IB routers.

3.1.1 Options for implementing IPv4 subnets spanning multiple IB subnets

There are two alternatives for implementations that do need to
span multiple IB subnets and cannot use IPv4 subnetting for
this. Based on the discussions in the working group one of these
methods will be chosen.
1. Use of a ‘spans multiple IB subnets’ option

Some implementations may however wish to implement IP subnets across IB subnets. It is a MUST for all IPv4 over IB implementations to define a configuration parameter associated with an IPv4 subnet that defines the scope bit to be used for the multicast translations. The default is to use the link-local scope. This parameter MUST be IP subnet wide to ensure that all the IP end nodes map the addresses the same way.

It is however NOT RECOMMENDED that such a parameter be set to anything but the default value. The implication is that the IP subnets are by default assumed to be within an IB subnet.

2. Use of an IP subnet number

Each IP subnet can be implemented to be associated with a specific 16 bit number. This number MUST be kept unique by the fabric administrator. This number MUST be made part of the multicast mapping thereby creating unique IPv4 subnet wide mappings. In this case the IB multicast GIDs MUST use the global scope.

3.2 Flag bits

IPv4 multicast/broadcast addresses have no well defined IPv6 or IB subnet mappings. The flag bits will therefore always be set to 0001 in IPv4 multicast/broadcast mappings to the IB multicast GID.

3.3 Mapping of IPv4 multicast address to IB address

The IPv4 broadcast to IB multicast GID mapping is defined to be:

FF1y::255.255.255.255

This mapping applies to all broadcast addresses i.e. any directed broadcast addresses and the limited broadcast address.

The IPv4 multicast to IB multicast GID mapping is correspondingly defined as:

FF1y::<IPv4 multicast address>
The default value of 'y' is 0x2. Thus the default translation of IPv4 broadcast address is FF12::255.255.255.255.

3.3.1 IPv4 multicast addresses spanning multiple IB subnets

If the IPv4 subnet spans multiple IB subnets the scope value will be set according to parameters defined in section 3.1.1.

If the IP subnet number is used the mapping will be defined as FF1E::IP_subnet_number:<IPv4 multicast/broadcast address>.

4.0 IPv4 multicast address to LID mapping

In a generic LAN setup the IPv4 multicast addresses are mapped to the destination link layer address directly. In the case of InfiniBand this is only partly true. This document describes the IPv4 multicast address to the IB multicast address (GID) mapping but not the mapping to the LID. The IPoIB driver on the host must determine the LID that needs to be used when sending to the particular multicast group.

A mapping from the IPv4 multicast address or the corresponding IB multicast group to a LID is not defined because of the following reasons:

1) Sending to an IPv4 multicast address

An IB node cannot be assured of its packets reaching all the multicast members without itself joining the IB multicast group. This is because the relevant switches are programmed by the IB subnet manager only on receiving a join request.

Thus the sender will always have to join the IB multicast groups and keep track of the groups it has already joined. Mapping directly to the LID doesn’t help if the the group has not been joined.

Thus the implementation is required to keep track of the IB groups joined. It can therefore also record the corresponding LID removing the need to map the IPv4 multicast address to the LID.

2) Joining an IPv4 multicast group

At the time of joining an IPv4 multicast group the IP host either will create or join the corresponding IB multicast group. Thus the
situations are no different as in the above case in terms of receiving the LID value.

3) Reduction of LID conflicts

The LIDs in the range 0xC000 to 0xFFFE are designated as the multicast LIDs by IBA. This limits the range to $2^{14} - 1$ entries (16382 entries). This implies that $2^{18}$ or 256K IPv4 multicast groups could map to a single LID. It is better to let the SM decide on a more efficient usage of the multicast LID space.

4) SM and IB architecture should stay unaffected.

A mapping of the LIDs can conflict with the SM implementations. The SM is under no restrictions to choose a particular LID for any multicast group. Thus it could end up utilising a LID that maps from an IPv4 multicast address for some other multicast group since not everything on IB subnets is governed by the IPoIB rules.

5) No need to plan for LID conflicts

Allowing the SM decide on the LIDs also avoids having to come up with a solution to handle LID conflicts with other multicast groups.

Thus it is best to avoid such a mapping and leave it to the individual implementations to determine the LID from the SM. There is no extra work involved in this determination since the SM has to be contacted anyway for the IB multicast group join/create operations.

5.0 Guidelines on setup of IB multicast groups

In an IB subnet, to communicate with one another, the endpoints must have compatible P_Keys. Thus the administrator when setting up an IP subnet over an IB subnet must ensure that all the members have compatible P_Keys. An endpoint may however have multiple P_Keys [4].

It is a MUST that the administrator setup the IB multicast group corresponding to IPv4 broadcast address (henceforth called ‘broadcast group’) when the IPv4 subnet is setup. The administrator therefore chooses the parameters that are valid for the multicast group: P_Key, Q_Key, Hop Limit, Flow ID,
TClass and the Path MTU.

This has the additional benefit of distributing these values to all the members of the subnet. Since all members must join the broadcast group the parameters are conveyed without having to define another method for this dispersal.

An IPv4 host on joining a new IPv4 multicast group might find that the corresponding IB multicast group doesn’t exist. The host’s interface driver will therefore create the IB multicast group by contacting the subnet manager (SM). It is RECOMMENDED that the parameters used in the creation of the group are the same as those returned for the broadcast group.

It is also RECOMMENDED that the administrator set up the IB groups corresponding to all-systems (224.0.0.1) and all-routers(224.0.0.2) multicast groups.

6.0 Outline description of multicast on IPoIB subnets

IPv4 multicast on InfiniBand subnets follows the same concepts and rules as on any other media. However, unlike most other media multicast over InfiniBand requires interaction with another entity, the IB subnet manager. This section describes the outline of the process and also suggests some guidelines.

IB architecture specifies the following format for IB multicast packets when used over unreliable datagram (UD) mode [4][5]:

```
+--------+-------+---------+---------+-------+---------+---------+
|Local   |Global |Base     |Datagram |Packet |Invariant| Variant |
|Routing |Routing|Transport|Extended |Payload| CRC     |  CRC    |
|Header  |Header |Header   |Transport|Header  |         |         |
+--------+-------+---------+---------+-------+---------+---------+
```

For details about the various headers please refer to InfiniBand Architecture Specification [1].

The Global routing header (GRH) includes the IB multicast group GID. The Local routing header (LRH) includes the local identifier (LID). The IB switches in the fabric route the packet based on the LID.

The GID is made available to the receiving IB user (the IPoIB interface driver for example). The driver can therefore determine the IB group the packet belongs to.
6.1 IPv4 broadcast addresses

All nodes join the broadcast group whenever an interface is configured. The broadcast group parameters (P_Key, Q_Key, LID etc.) are recorded by the interface module to be used while creating other groups.

6.2 Receiving/forwarding multicast packets

An IP host sends an IGMP report to the router(s) when it wants to receive packets on a multicast group. The router could then create the IB group. However to receive the packets the IP host must join the corresponding IB multicast group. Therefore, it is simpler for the IB interface module on the IP host to first create the IB group and then send the IGMP message to the router. The router will then join the specified IB group.

The router MAY choose to create IB groups corresponding to the IPv4 groups it expects to forward.

Thus the creation of IB groups is done by receivers or routers only and not by senders thereby keeping things simple. The host must first try to join the group and only on failure attempt to create it.

6.3 Sending IPv4 multicast datagrams

An IP host may send a multicast packet at any time to any multicast address. The IP layer translates the address to the IB multicast group address as per the mapping described in this document.

The IP layer then conveys the packet to the IB interface driver/module. This module attempts to join the relevant IB multicast group. This is required since otherwise there is no guarantee that the packet will reach its destinations.

The IB join could fail if the group has not been created. This could imply that there are no listeners on the subnet and the router doesn’t expect to forward packets received on this group. In such a case the module would be justified in dropping the packet.

However, this may not be the case. The IB group may not exist because the SM ran out of resources or the SM policy allows only a limited set of multicast groups to be created. Additionally it is not reasonable to expect the router to
create IB groups for all the IPv4 multicast addresses that it may be called upon to forward.

Therefore, the multicast module of IPv4 interface, when sending a multicast packet MUST do one the following:

1) join the IB multicast group corresponding to the IPv4 multicast address. This is the RECOMMENDED option for multicast if the sender is itself a member of the group.

   As noted earlier, a particular IB multicast group may not exist for some reason. In such a case the implementation MUST fall back to one of the following methods.

2) Send the multicast packet out with the corresponding IB GID but with the LID associated with the all-systems IPv4 multicast address (224.0.0.1). This is the RECOMMENDED option.

   An implementation implementing 1) described above must fall back to this condition or the condition given below on failure to join the IB group corresponding to the IPv4 multicast address being sent to.

3) Send the multicast packet out with the corresponding IB GID but with the LID corresponding to the IPv4 limited broadcast address (255.255.255.255).

   An implementation MUST fall back to this option if both the options 1) and 2) fail.

6.4 Leaving/Deleteing a multicast group

An IPv4 sender joins the IB multicast group only because that is the only way to guarantee reception of the packets by all the group recipients. The sender must however leave the IB group at some time. It is RECOMMENDED that a sender, when not a receiver on the group, start a timer per multicast group sent to. The sender leaves the IB group when the timer goes off. It restarts the timer if another message is sent. It is RECOMMENDED that the duration of the timer be 120 seconds.

This recommendation doesn’t apply to the IB broadcast group. It also doesn’t apply to the IB group corresponding to the
all-hosts multicast group. A host MUST always remain a member of the broadcast group. It MAY choose to remain a member of all-hosts group.

Thus a sender that chooses to always send to the broadcast group and not to the specific multicast group does not need to implement a timer.

An IPv4 multicast receiver MUST leave the corresponding IB multicast group when it leaves the IPv4 multicast group. If it continues to be a sender then it MAY choose to not leave the IB group but start a timer as explained above.

A router is RECOMMENDED to leave the IB multicast group when there are no members of the IPv4 multicast address in the subnet and it has no explicit knowledge of any need to forward such packets.

The router and the IPv4 hosts MUST NOT delete the IB multicast group when they leave the group. It is possible for the same IB multicast group be used by a non-IPv4 protocol.

The IB specification mentions an IB specific protocol that will delete the IB groups when it determines that there are no IB members of the group.

7.0 Security considerations

Any multicast/broadcast communication is inherently insecure since anyone can receive the data. The applications must implement appropriate authentication/encryption methods for data security.

The IPv4 subnet communication can be disrupted by creating the IB broadcast/multicast groups with incompatible parameters. The implementations must leverage IB specific methods to protect against such situations.

8.0 Acknowledgement

The author thanks David L. Stevens for his useful suggestions and comments.
9.0 References


[5] draft-kashyap-ipoib_requirements-00.txt


10.0 Author’s Address

Vivek Kashyap

IBM

15450, SW Koll Parkway

Beaverton, OR 97006

Work: 503 578 3422

Email: vivk@us.ibm.com

11.0 Full Copyright Statement

Copyright (C) The Internet Society (2001). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.
The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
Vivek Kashyap
IBM
viv@sequent.com
vivk@us.ibm.com
503 578 3422 (o)