                              Host Groups:
              A Multicast Extension to the Internet Protocol

1. Status of this Memo

   This RFC defines a model of service for Internet multicasting and
   proposes an extension to the Internet Protocol (IP) to support such a
   multicast service. Discussion and suggestions for improvements are
   requested. Distribution of this memo is unlimited.

2. Acknowledgements

   This memo was adapted from a paper [7] presented at the Ninth Data
   Communications Symposium. This work was sponsored in part by the
   Defense Advanced Research Projects Agency under contract N00039-83-
   K-0431 and National Science Foundation Grant DCR-83-52048.

   The Internet task force on end-to-end protocols, headed by Bob
   Braden, has provided valuable input in the development of the host
   group model.

3. Introduction

   In this paper, we describe a model of multicast service we call host
   groups and propose this model as a way to support multicast in the
   DARPA Internet environment [14]. We argue that it is feasible to
   implement this facility as an extension of the existing "unicast" IP
   datagram model and mechanism.

   Multicast is the transmission of a datagram packet to a set of zero
   or more destination hosts in a network or internetwork, with a single
   address specifying the set of destination hosts. For example, hosts
   A, B, C and D may be associated with multicast address X. On
   transmission, a packet with destination address X is delivered with
   datagram reliability to hosts A, B, C and D.

   Multicast has two primary uses, namely distributed binding and
   multi-destination delivery. As a binding mechanism, multicast is a
   robust and often more efficient alternative to the use of name
   servers for finding a particular object or service when a particular
   host address is not known. For example, in a distributed file
   system, all the file servers may be associated with one well-known
   multicast address. To bind a file name to a particular server, a
   client sends a query packet containing the file name to the file
   server multicast address, for delivery to all the file servers. The

server that recognizes the file name then responds to the client,
allowing subsequent interaction directly with that server host. Even
when name servers are employed, multicast can be used as the first
step in the binding process, that is, finding a name server.

Multi-destination delivery is useful to several applications,
including:

- distributed, replicated databases [6,9].

- conferencing [11].

- distributed parallel computation, including distributed
  gaming [2].

Ideally, multicast transmission to a set of hosts is not more
complicated or expensive for the sender than transmission to a single
host. Similarly, multicast transmission should not be more expensive
for the networks and gateways than traversing the shortest path tree
that connects the sending host to the hosts identified by the
multicast address.

Multicast, transmission to a set of hosts, is properly distinguished
from broadcast, transmission to all hosts on a network or
internetwork. Broadcast is not a generally useful facility since
there are few reasons for communicating with all hosts.

A variety of local network applications and systems make use of
multicast. For instance, the V distributed system [8] uses
network-level multicast for implementing efficient operations on
groups of processes spanning multiple machines. Similar use is being
made for replicated databases [6] and other distributed applications
[4]. Providing multicast in the Internet environment would allow
porting such local network distributed applications to the Internet,
as well as making some existing Internet applications more robust and
portable (by, for example, removing "wired-in" lists of addresses,
such as gateway addresses).

At present, an Internet application logically requiring multicast
must send individually addressed packets to each recipient. There
are two problems with this approach. Firstly, requiring the sending
host to know the specific addresses of all the recipients defeats its
use as a binding mechanism. For example, a diskless workstation
needs on boot to determine the network address of a disk server and
it is undesirable to "wire in" specific network addresses. With a
multicast facility, the multicast address of the boot servers (or

   name servers that hold the addresses of the boot servers) can be
   well-known, allowing the workstation to transmit its initial queries
   to this address.

   Secondly, transmitting multiple copies of the same packet makes
   inefficient use of network bandwidth, gateway resources and sender
   resources. For instance, the same packet may repeatedly traverse the
   same network links and pass through the same gateways. Furthermore,
   the local network level cannot recognize multi-destination delivery
   to take advantage of multicast facilities that the underlying network
   technologies may provide. For example, local-area bus, ring, or
   radio networks, as well as satellite-based wide-area networks, can
   provide efficient multicast delivery directly. Besides using
   excessive communication resources, the use of multiple transmissions
   to effect multicast severely limits the amount of parallelism in
   transmission and processing that can be achieved compared to an
   integrated multicast facility.

   The next section describes the host group model of multicast service.
   Section 5 describes the extensions to IP to support the host group
   model. Section 6 discusses the implementation of multicast within
   the networks and gateways making up the Internet. Section 7 relates
   this model to other proposals. Finally, we conclude with remarks on
   our experimental prototype implementation of host groups and comments
   on future directions for investigation.

4. The Host Group Model

   The Internet architecture defines a name space of individual host
   addresses. The host group model extends that name space to include
   addresses of host groups. A host group is a set of zero or more
   Internet hosts [1].   When an IP packet is sent with a host group
   address as its destination, it is delivered with "best effort"
   datagram reliability to all members of that host group.

   The sender need not be a member of the destination group. We refer
   to such a group as open, in contrast to a closed group where only
   members are allowed to send to the group. We chose to provide open
   groups because they are more flexible and more consistent as an
   extension of conventional unicast models (even though they may harder
   to implement).

   Dynamic management of group membership provides flexible binding of
   Internet addresses to hosts. Hosts may join and leave groups over
   time. A host may also belong to more than one group at a time.
   Finally, a host may belong to no groups at times, during which that
   host is unreachable within the Internet architecture. In fact, a

host need not have an individual Internet address at all. Some hosts
may only be associated with multi-host group addresses. For
instance, there may be no reason to contact an individual time server
in the Internet, so time servers would not require individual
addresses.

Internet addresses are dynamically allocated for transient groups,
groups that often last only as long as the execution of a single
distributed program. In addition, a range of host group identifiers
is reserved for identifying permanent groups. One use of permanent
host groups identifiers is for host groups with standard logical
meanings such as "name server group", "boot server group", "Internet
monitor group", etc.

In the current Internet architecture, addresses are bound to single
hosts. The host group model generalizes the binding of Internet
addresses to hosts by allowing one address to bind to multiple hosts
on multiple networks, more than one address to be bound (in part) to
one host, and the binding of an address to host to be dynamic, i.e.
possible to be modified under application control. Within this more
general model, the current architecture is supported as a special
case, retaining its current semantics and implementation.

The following subsections provide further details of the model.

4.1. Host Group Management

   Dynamic binding of Internet addresses to hosts is managed by the
   following three operations which are made available to clients of
   the Internet Protocol <2>:

      CreateGroup (type) --> outcome, group-address, access-key

   requests the creation of a new transient host group with the
   invoking host as its only member. The type argument specifies
   whether the group is restricted or unrestricted. A restricted
   group restricts membership based on the access-key. Only hosts
   presenting a valid host access-key are allowed to join. All
   unrestricted host groups have a null access-key. outcome
   indicates whether the request is approved or denied. If it is
   approved, a new transient group address is returned in
   group-address. access-key is the protection key (or password)
   associated with the new group. This should fail only if there are
   no free transient group addresses.

      JoinGroup (group-address, access-key) --> outcome

requests that the invoking host become a member of the identified
host group (permanent or transient). outcome indicates whether
the request is approved or denied. A request is denied if the
access key is invalid.

    LeaveGroup (group-address) --> outcome

requests that the invoking host be dropped from membership in the
identified group (permanent or transient). outcome indicates
whether the request is approved or denied.

There is no operation to destroy a transient host group because a
transient host group is deemed to no longer exist when its
membership goes to zero.

Permanent host group addresses are allocated and published by
Internet administrators, in the same way as well-known TCP and UDP
port numbers. That is, they are published in future editions of
the "Assigned Numbers" document [17].

4.2. Packet Transmission

Transmission of a packet in the host group model is controlled by
two parameters of scope, one being the destination internetwork
address and the other being the "distance" to the destination
host(s). In particular,

    Send (dest-address, source-address, data, distance)

transmits the specified data in an internetwork datagram to the
host(s) identified by dest-address that are within the specified
distance. The destination address is thus similar to conventional
networks except that delivery may be to multiple hosts; the
distance parameter requires further discussion.

Distance may be measured in several ways, including number of
network hops, time to deliver and what might be called
administrative distance. Administrative distance refers to the
distance between the administrations of two different networks.
For example, in a company the networks of the research group and
advanced development group might be considered quite close to each
other, networks of the corporate management more distant, and
networks of other companies much more distant. One may wish to
restrict a query to members within one's own administrative domain
because servers outside that domain may not be trusted.
Similarly, error reporting outside of an administrative domain may
not be productive and may in fact be confusing.

Besides limiting the scope of transmission, the distance parameter
can be used to control the scope of multicast as a binding
mechanism and to implement an expanding scope of search for a
desired service. For instance, to locate a name server familiar
with a given name, one might check with nearby name servers and
expand the distance (by incrementing the distance on
retransmission) to include more distant name servers until the
name is found.

To reach all members of a group, a sender specifies the maximum
value for the distance parameter. This maximum must exceed the
"diameter" of the Internet.

Packet reception is the same as conventional architectures. That
is,

    Receive () --> dest-address, source-address, data

returns the next internetwork datagram that is, or has been,
received.

4.3. Delivery Requirements

We identify several requirements for the packet delivery mechanism
that are essential to host groups being a useful and used
facility.

Firstly, given the predominance of broadcast local-area networks
and the locality of communication to individual networks, the
delivery mechanism must be able to exploit the hardware's
capability for very efficient multicast within a single local-area
network.

Secondly, the delivery mechanism must scale in sophistication to
efficient delivery across the Internet as it acquires high-speed
wide-area communication links and higher performance gateways.
The former are being provided by the introduction of high-speed
satellite channels and long-haul fiber optic links. The latter
are made feasible by the falling cost of memory and processing
power plus the increasing importance in controlling access to
relatively unprotected local network environments. A host group
delivery mechanism must be able to take advantage of these trends
as they materialize.

Finally, the delivery mechanism must avoid "systematic errors" in
delivery to members of the host group. That is, a small number of
repeated transmissions must result in delivery to all group

        members within the specified distance, unless a member is
        disconnected or has failed. We refer to this property as
        coverage. In general, most reliable protocols make this basic
        assumption for unicast delivery. It is important to guarantee
        this assumption for multicast as well or else applications using
        multicast may fail in unexpected ways when coverage is not
        provided. For efficiency, the multicast delivery mechanism should
        also avoid regularly delivering multiple copies of a packet to
        individual hosts.

        Failure notification is not viewed as an essential requirement,
        given the datagram semantics of delivery. However, a host group
        extension to IP should provide "hint"-level failure notification
        as the natural extension of the failure notification for unicast.

5. Extensions to IP

    This section discusses the specific extensions to the DARPA Internet
    Protocol required to support the host group model. The extensions
    need be implemented only on those hosts that wish to join host groups
    or send to host groups; existing implementations are not affected by
    the proposed changes.

    5.1. Group Addresses

        A portion of the 32-bit IP address space is reserved for host
        group addresses. The range of group addresses is chosen to be
        easily recognized and to not conflict with existing individual
        addresses. Either Class A addresses with a distinguished
        (currently unused) network number or Class D addresses (those
        starting with 111) would be suitable. The range of group addresses
        is further subdivided into a set of permanent group addresses and
        a set of temporary group addresses.

        Host group addresses may be used in the same way as individual
        addresses in the source, destination, and options fields of IP
        datagrams. An IP implementation adds to the list of its own
        individual addresses, the addresses of all groups to which it
        belongs. The source addresses of locally originated datagrams are
        validated against the list, and incoming datagrams which are not
        destined to an address on the list are discarded. The addresses
        on the list change dynamically as IP users create, join and leave
        groups.

   5.2. Group Management

      To support the group management operations of CreateGroup,
      JoinGroup and LeaveGroup, an IP module must interact with one or
      more multicast agents which reside in neighbouring gateways or
      other special-purpose hosts. These interaction are handled by an
      Internet Group Management Protocol (IGMP) which, like ICMP [15],
      is an integral part of the IP implementation. A proposed
      specification for IGMP is given in Appendix I.

   5.3. Multicast Delivery

      In order to transmit a datagram destined to a host group, an IP
      module must map the destination group address into a local network
      address. As with individual IP addresses, the mapping algorithm
      is local-network- specific. On networks that directly support
      multicast, the IP host group address is mapped to a local network
      multicast address that includes all local members of the host
      group plus one or more multicast agents. For networks that do not
      directly support multicast, the mapping may be to a more general
      broadcast address, to a list of local unicast addresses, or
      perhaps to the address of a single machine that handles
      multi-destination relaying.

   5.4. Distance Control

      The existing Time to Live field in the IP header can be used for
      crude control over the delivery radius of multicast datagrams. To
      provide finer-grain control, a new IP option is defined to specify
      the maximum delivery distance in "administrative units", such as
      "this network", "this department", "this company", "this country",
      etc. The set of units and their encoding is to be determined.

6. Implementation

   In this section, we sketch a design for implementing the host group
   model within the Internet. This description of the design is given
   to further support the feasibility of the host group model as well as
   point out some of the problems yet to be addressed.

   Implementation of host groups involves implementing a binding
   mechanism (binding Internet addresses to zero or more hosts) and a
   packet delivery mechanism (delivering a packet to each host to which
   its destination address binds). This facility fits most naturally
   into the gateways of the Internet and the switching nodes of the
   constituent point-to-point networks (as opposed to separate machines)
   because multicast binding and delivery is a natural extension of the

unicast binding and delivery (i.e. routing plus store-and-forward).
That is, a multicast packet is routed and transmitted to multiple
destinations, rather than to a single destination.

In the following description, we start with a basic, simple
implementation that provides coverage and then refine this mechanism
with various optimizations to improve efficiency of delivery and
group management.

6.1. Basic Implementation

   A host group defines a network group, which is the set of networks
   containing current members of the host group. When a packet is
   sent to a host group, a copy is delivered to each network in the
   corresponding network group. Then, within each network, a copy is
   delivered to each host belonging to the group.

   To support such multicast delivery, every Internet gateway
   maintains the following data structures:

      - routing table: conventional Internet routing information,
        including the distance and direction to the nearest gateway
        on every network.

      - network membership table: A set of records, one for every
        currently existing host group. The network membership record
        for a group lists the network group, i.e. the networks that
        contain members of the group.

      - local host membership table: A set of records, one for each
        host group that has members on directly attached networks.
        Each local host membership record indicates the local hosts
        that are members of the associated host group. For networks
        that support multicast or broadcast, the record may contain
        only the local network-specific multicast address used by the
        group plus a count of local members. Otherwise, local group
        members may be identified by a list of unicast addresses to
        be used in the software implementation of multicast within
        the network.

   A host invokes the multicast delivery service by sending a
   group-destined IP datagram to an immediate neighbour gateway (i.e.
   a gateway that is directly attached to the same network as the
   sending host). Upon receiving a group-destined datagram from a
   directly attached network, a gateway looks up the network
   membership record corresponding to the destination address of the
   datagram. For each of the networks listed in the membership

record, the gateway consults its routing table. If, according to
the routing table, a member network is directly attached, the
gateway transmits a copy of the datagram on that network, using
the network-specific multicast address allocated for the group on
that network. For a member network that is not directly attached
the gateway creates a copy of the datagram with an additional
inter-gateway header identifying the destination network. This
inter-gateway datagram is forwarded to the nearest gateway on the
destination network, using conventional store-and-forward routing
techniques. At the gateway on the destination network, the
datagram is stripped of its inter-gateway header and transmitted
to the group's multicast address on that network. The datagram is
dropped by the relaying gateways whenever it exceeds its distance
limit.

The network membership records and the network-specific multicast
structures are updated in response to group management requests
from hosts. A host sends a request to create, join, or leave a
group to an immediate neighbour gateway. If the host requests
creation of a group, a new network membership record is created by
the serving gateway and distributed to all other gateways. If the
host is the first on its network to join a group, or if the host
is the last on its network to leave a group, the group's network
membership record is updated in all gateways. The updates need
not be performed atomically at all gateways, due to the datagram
delivery semantics; hosts can tolerate misrouted and lost packets
caused by temporary gateway inconsistencies, as long as the
inconsistencies are resolved within normal host retransmission
periods. In this respect, the network membership data is similar
to the network reachability data maintained by conventional
routing algorithms, and can be handled by similar mechanisms.

In many cases, a host joins a group that already has members on
the same network, or leaves a group that has remaining members on
the same network. This is then a local matter between the hosts
and gateways on a single network: only the local host membership
table needs to be updated to include or exclude the host.

This basic implementation strategy meets the delivery requirements
stated at the end of Section 4. However, it is far from optimal,
in terms of either delivery efficiency or group management
overhead. Below, we discuss some further refinements to the basic
implementation.

   6.2. Multicast Routing Between Networks

      Multicast routing among the Internet gateways is similar to
      store-and-forward routing in a point-to-point network. The main
      difference is that the links between the nodes (gateways) can be a
      mixture of broadcast and unicast-type networks with widely
      different throughput and delay characteristics. In addition,
      packets are addressed to networks rather than hosts (at the
      gateway level).

      We intend to use the extended reverse path forwarding algorithm of
      Dalal and Metcalfe [10]. Although originally designed for
      broadcast, it is a simple and efficient technique that can serve
      well for multicast delivery if network membership records in each
      gateway are augmented with information from neighbouring gateways.
      This algorithm uses the source network identifier, rather than a
      destination network identifier to make routing decisions. Since
      the source address of a datagram may be a group address, it cannot
      be used to identify the source network of the datagram; the first
      gateway must add a header specifying the source network. This
      approach minimizes redundant transmissions when multiple
      destination networks are reachable across a common intergateway
      link, a problem with the basic implementation described above.

      Note that we eliminate from consideration techniques that fail to
      deliver along the branches of the shortest delay tree rooted at
      the source, such as Wall's center-based forwarding [16] because
      this compromises the meaning of the multicast distance parameter
      and detracts from multicast performance in general. We also
      rejected the approach of having a multicast packet carry more than
      one network identifier in its inter-gateway header to indicate
      multiple destination networks because the resulting variable
      length headers would cause buffering and fragmentation problems in
      the gateways.

   6.3. Multicasting Within Networks

      A simple optimization within a network is to have the sender use
      the local multicast address of a host group for its initial
      transmission. This allows the local host group members to receive
      the transmission immediately along with the gateways (which must
      now "eavesdrop" on all multicast transmissions). A gateway only
      forwards the datagram if the destination host group includes
      members on other networks. This scheme reduces the cost to reach
      local group members to one packet transmission from two required

in the basic implementation <3> so transmission to local members
is basically as efficient as the local multicast support provided
by the network.

A similar opportunity for reducing packet traffic arises when a
datagram must traverse a network to get from one gateway to
another, and that network also holds members of the destination
group. Again, use of a network-specific multicast address which
includes member hosts plus gateways can achieve the desired
effect. However, in this case, hosts must be prepared to accept
datagrams that include an inter-gateway header or, alternatively,
every datagram must include a spare field in its header for use by
gateways in lieu of an additional inter-gateway header.

6.4. Distributing Membership Information

A refinement to host group membership maintenance is to store the
host group membership record for a group only in those gateways
that are directly connected to member networks. Information about
other groups is cached in the gateway only while it is required to
route to those other groups. When a gateway receives a datagram
to be forwarded to a group for which it has no network membership
record (which can only happen if the gateway is not directly
connected to a member network), it takes the following action.
The gateway assumes temporarily that the destination group has
members on every network in the internetwork, except those
directly attached to the sending gateway, and routes the datagram
accordingly. In the inter-gateway header of the outgoing packet,
the gateway sets a bit indicating that it wishes to receive a copy
of the network membership record for the destination host group.
When such a datagram reaches a gateway on a member network, that
gateway sends a copy of the membership record back to the
requesting gateway and clears the copy request bit in the
datagram.

Copies of network membership records sent to gateways outside of a
group's member networks are cached for use in subsequent
transmissions by those gateways. That raises the danger of a
stale cache entry leading to systematic delivery failures. To
counter that problem, the inter-gateway header contains a field
which is a hash value or checksum on the network membership record
used to route the datagram. Gateways on member networks compare
the checksum on incoming datagrams with their up-to-date records.
If the checksums don't match, an up-to-date copy of the record is
returned to the gateway with the bad record.

This caching strategy minimizes intergateway traffic for groups

that are only used within one network or within the set of
networks on which members reside, the expected common cases.
Partial replication with caching also reduces the overhead for
network traffic to disseminate updates and keep all copies
consistent. Finally, it also reduces the total space required in
all the gateways to support a large number of host groups.

We have not addressed here the problem of maintaining up-to-date,
consistent network membership records within the set of gateways
connected to members of a group. This can be viewed as a
distributed database problem which has been well studied in other
contexts. The loose consistency requirements on network
membership records suggest that the techniques used in Grapevine
[3] might be useful for this application.

7. Related Work

The use of unreliable multicast by higher-level protocols and the
implementation of multicast within various individual networks have
been well-studied (see [7] for references and discussion). However,
there is relatively little published work on the use or
implementation of internetwork multicasting.

Boggs, in his thesis [4], describes a number of distributed
applications that are impossible or very awkward to support without
the flexible binding nature of broadcast addressing. Although he
recognizes that almost all of his applications would be best served
by a multicast mechanism, he advocates the use of "directed
broadcast" because it is easy to implement within many kinds of
networks and can be extended across an internetwork without placing
any new burden on internetwork gateways. In RFC-919 [13], Mogul
proposes adopting directed broadcast for the DARPA Internet.

Broadcasting has the undesirable side effect of delivering packets to
more hosts than necessary, thus incurring overhead on uninvolved
parties and possibly creating security problems. As more and more
applications take advantage of broadcasting, the overhead on all
hosts continues to rise. Clearly, broadcast does not scale up to a
large internetwork. As an attempt to handle the scaling problem,
directed broadcast is less attractive than true multicast because the
set of hosts that can be reached by a single "send" operation is an
artifact of the internetwork topology, rather than a grouping that is
meaningful to the sender.

In RFC-947 [12], Lebowitz and Mankins propose the use of broadcast

repeaters that pick up broadcast datagrams from one network and relay
them to other networks for broadcast there. This technique is even
less selective of its targets than Bogg's directed broadcast method.

Aguilar [1] suggests allowing an IP datagram to carry multiple
destination addresses, which are used by the gateways to route the
datagram to each recipient. Such a facility would alleviate some of
the inefficiencies of sending individual datagrams to a group, but it
would not be able to take advantage of local network multicast
facilities. More seriously, Aguilar's scheme requires the sender to
know the individual IP addresses of all members of the destination
group and thus lacks the flexible binding nature of true multicast or
broadcast.

8. Concluding Remarks

We have described a model of multicast communication for the
Internet. As an extension of the existing Internet architecture, it
views unicast communication and time-to-live constraints as special
cases of the more general form of communication arising with
multicast. We have argued that this model is implementable in the
Internet and that it provides a powerful facility for a variety of
applications. In some cases, it provides a facility that is required
for certain applications to work in the Internet environment. In
other cases, it provides a more efficient, robust and possibly more
elegant way of implementing existing Internet applications.

We are currently implementing a prototype host group facility as an
extension of IP. For practical reasons, this prototype implements
all group management functions and multicast routing outside of the
Internet gateways, in special hosts called multicast agents, which
are similar to the broadcast repeaters of Lebowitz and Mankins. The
collection of multicast agents in effect provides a second gateway
system on top of the existing Internet, for multicast purposes. The
major costs of this separation are redundancy of routing tables
between gateways and multicast agents and the increased delay and
unreliability of extra hops in the delivery path. Much of the
routing information in the multicast agents must be "wired-in"
because they do not have access to the gateways' routing tables.
However, this rudimentary implementation provides an environment for
evaluating the interface to the multicast service and for
investigating group management and multicast routing protocols for
eventual use in the gateways. It also serves as a testbed for
porting multicast-based distributed applications to the Internet.

For now, we are restricting group membership to local networks that
already have a broadcast or multicast capability, such as the

Ethernet. We feel that, in the future, any network that is to support
hosts other than just gateways must have a multicast addressing mode.
Efficient implementation of multicast within point-to-point or
virtual circuit networks deserves investigation.

A significant issue raised by the host group model is authentication
and access control in the Internet. Gateways must control which
hosts can create and join host groups, presumably making their
decision based on the identity of the requestor (thus requiring
authentication) and permissions (access control lists). This issue
does not arise in conventional internetwork architectures because
host addresses are administratively assigned with no notion of
dynamic assignment and binding as provided by host groups. We
believe that access control should be recognized as a proper and
necessary function of gateways so as to protect the hosts of local
networks from general internetwork activity. Thus, group access
control can be subsumed as part of this more general mechanism,
although more investigation of the general issue is called for.

On a philosophical point, there has been considerable reluctance to
make open use of multicast on local networks because it was
network-specific and not provided across the Internet. We were
originally of that school. However, we recognized that our "hidden"
uses of multicast in the V distributed system were essential unless
we resorted to dramatically poorer solutions - wired-in addresses.
We also recognized, as described in this paper, that an adequate
multicast facility for the Internet was feasible. As a consequence,
we now argue that multicast is an important and basic facility to
provide in local networks and internetworks. Higher levels of
communication, including applications, should feel free to make use
of this powerful facility. Networks and internetworks lacking
multicast should be regarded as deficient relative to the future (and
present) requirements of sophisticated distributed applications and
communication systems.

Appendix I. Internet Group Management Protocol (IGMP)

   The Internet Group Management Protocol (IGMP) is used between IP
   hosts and their immediate neighbour multicast agents to support the
   allocation of temporary group addresses and the addition and deletion
   of members of a group.

   Like ICMP, IGMP is a required part of all IP implementations. IGMP
   messages are encapsulated in IP datagrams, with an IP protocol number
   of 2. IGMP messages are formatted similarly to ICMP messages and the
   different IGMP message types are given values distinct from ICMP
   message types, so that both protocols may share common implementation
   modules or, perhaps, be merged into a single protocol.

   IGMP interactions take the form of request-response transactions. A
   request message is sent by hosts to the permanent group of all
   immediate neighbour multicast agents. Multicast agents reply to the
   IP source address of a request. If no reply is received within a
   (currently unspecified) timeout interval, a host retransmits its
   request, up to some (currently unspecified) maximum number of times.
   IGMP transactions are considered idempotent, so that multicast agents
   need not recognize and filter out duplicate requests nor buffer
   replies <4>.

   The IGMP message formats and procedures are defined below, in the
   style used in the ICMP specification.

Create Group Request or Create Group Reply Message

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|           Identifier          |        Sequence Number        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Group Address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                          Access Key                           +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
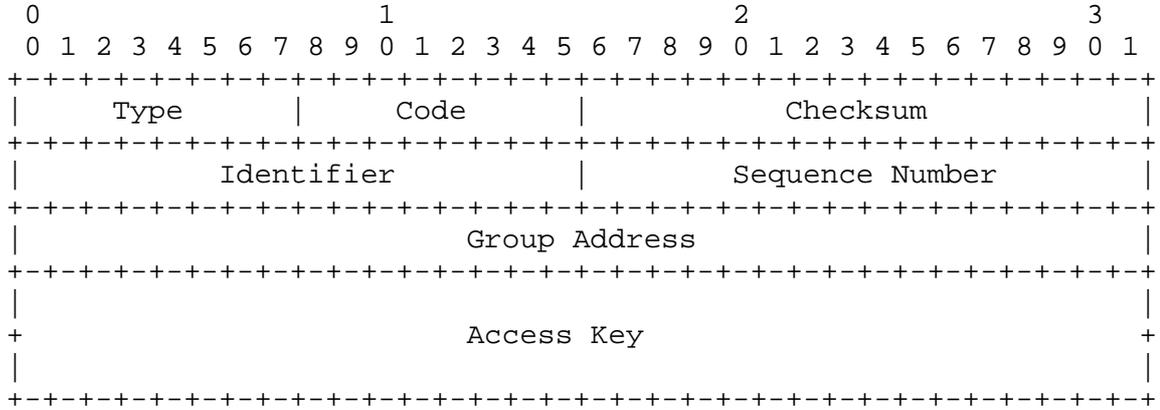
IP Fields:

Addresses

   A Create Group Request message is sent with an individual IP
   address of the sending host as its source, and the well-known
   group address of the multicast agents as its destination.

   The corresponding Create Group Reply is sent with those two
   addresses reversed.

IGMP Fields:

Type

   101 for Create Group Request
   102 for Create Group Reply

Code

   For a Create Group Request message, the Code field indicates if
   the group is to be restricted:

      0 = unrestricted
      1 = restricted

   For a Create Group Reply message, the Code field specifies the
   outcome of the request:

      0 = request approved
      1 = request denied, no resources

Checksum

   The checksum is the 16-bit one's complement of the one's
   complement sum of the IGMP message starting with the IGMP Type.
   For computing the checksum, the checksum field should be zero.
   This checksum may be replaced in the future.

Identifier

   An identifier to aid in matching Request and Reply messages.

Sequence Number

   A sequence number to aid in matching Request and Reply
   messages.

Group Address

   For a Create Group Request message, a value of 0.

   For a Create Group Reply message, either a newly allocated
   group address (if the request is approved) or a value of 0 (if
   denied).

Access Key

   For a Create Group Request message, a value of 0.

   For a Create Group Reply message, either a pseudo-random 64-bit
   number (if the request for a restricted group is approved) or
   0.

Description

   A Create Group Request message is sent to the the group of
   local multicast agents by a host wishing to allocate a new
   temporary group.

   If no Reply message is received within t seconds, the Request
   is retransmitted. If no Reply is received after n
   transmissions, the request is deemed to have failed.

   The first Reply message to arrive, if any, specifies the
   outcome of the request. The request may be denied because of
   lack of resources (e.g. no table space in gateways or all
   temporary addresses in use).

If the request is approved, the requesting host is considered
to be the first and only current member of the new host group.

The Identifier and Sequence Number fields are used to match the
Reply to the corresponding Request. The multicast agents may
choose to use these values to minimize the chance of allocating
more than one new group for a single request, for example when
a Reply is lost and a

Request is retransmitted. However, the multicast agents must
be prepared to recover temporary group addresses without
requiring explicit Leave Group Requests from all members; they
may choose simply to allocate a new address for every
retransmission and recover unused ones when needed <5>.

    Join Group Request or Join Group Reply Message

        0   1   2   3
        0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
       +-+
       | Type | Code | Checksum |
       +-+
       | Identifier | Sequence Number |
       +-+
       | Group Address |
       +-+
       | |
       +   Access Key   +
       | |
       +-+
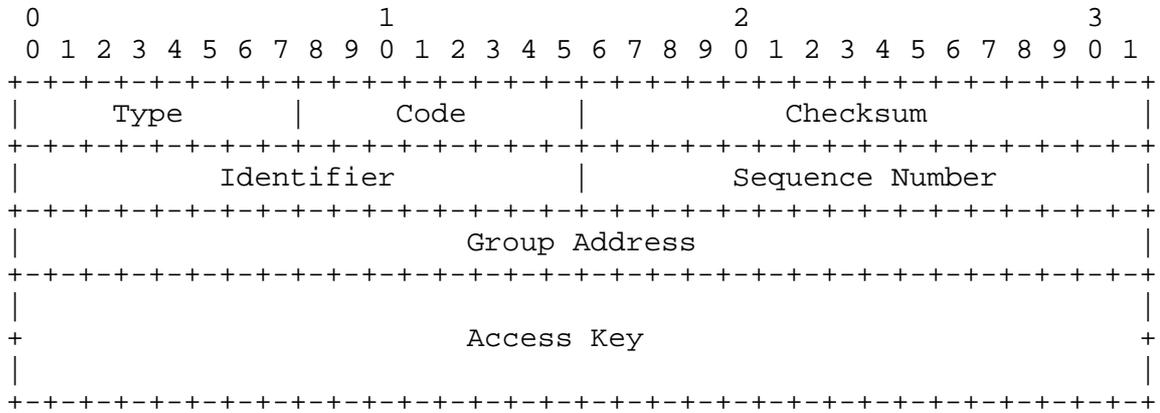
       IP Fields:

       Addresses

          A Join Group Request message is sent with an individual IP
          address of the sending host as its source, and the well-known
          group address of the multicast agents as its destination.

          The corresponding Join Group Reply is sent with those two
          addresses reversed.

       IGMP Fields:

       Type

          103 for Join Group Request
          104 for Join Group Reply

       Code

          For a Join Group Request message, the Code field contains 0.

          For a Join Group Reply message, the Code field specifies the
          outcome of the request:

             0 = request approved
             1 = request denied, no resources
             2 = request denied, invalid group address
             3 = request denied, invalid access key

Checksum

The checksum is the 16-bit one's complement of the one's
complement sum of the IGMP message starting with the IGMP Type.
For computing the checksum, the checksum field should be zero.
This checksum may be replaced in the future.

Identifier

An identifier to aid in matching Request and Reply messages.

Sequence Number

A sequence number to aid in matching Request and Reply
messages.

Group Address

For a Join Group Request message, a host group address.

For a Join Group Reply message, the same group address as in
the corresponding request.

Access Key

For a Join Group Request message, the access key allocated when
the group was created (0 for unrestricted groups).

For a Join Group Reply message, the same access key as in the
corresponding request.

Description

A Join Group Request message is sent to the the group of local
multicast agents by a host wishing to join a specified,
existing group. If no Reply message is received within t
seconds, the Request is retransmitted. If no reply is received
after n transmissions, the request is deemed to have failed.

The first Reply message to arrive, if any, specifies the
outcome of the request. The request may be denied because of
an invalid access key, an invalid specified group address (e.g.
non-existent group) or lack of resources (e.g. no table space
in gateways).

The Identifier and Sequence Number fields are used to match the
Reply to the corresponding Request. If a multicast agent

receives a request from a host to join a group to which it
already belongs, the agent approves the request, under the
assumption that the request was a retransmission for a lost
Reply.

Leave Group Request or Leave Group Reply Message

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Identifier         |        Sequence Number        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Group Address                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
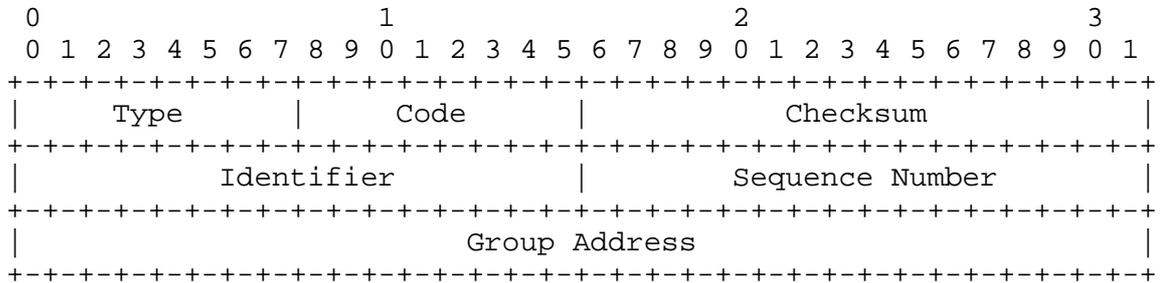
IP Fields:

Addresses

   A Leave Group Request message is sent with an individual IP
   address of the sending host as its source, and the well-known
   group address of the multicast agents as its destination.

   The corresponding Leave Group Reply is sent with those two
   addresses reversed.

IGMP Fields:

Type

   105 for Leave Group Request
   106 for Leave Group Reply

Code

   For a Leave Group Request message, the Code field contains 0.

   For Leave Group Reply message, the Code field specifies the
   outcome of the request:

      0 = request approved
      2 = request denied, invalid group address

Checksum

   The checksum is the 16-bit one's complement of the one's
   complement sum of the IGMP message starting with the IGMP Type.
   For computing the checksum, the checksum field should be zero.
   This checksum may be replaced in the future.

Identifier

   An identifier to aid in matching Request and Reply messages.

Sequence Number

   A sequence number to aid in matching Request and Reply
   messages.

Group Address

   For a Leave Group Request message, a host group address.

   For a Leave Group Reply message, the same group address as in
   the corresponding request.

Description

   A Leave Group Request message is sent to the the group of local
   multicast agents by a host wishing to leave a specified,
   existing group. If no Reply message is received within t
   seconds, the Request is retransmitted. If no reply is received
   after n transmissions, the request is deemed to have succeeded.

   The first Reply message to arrive, if any, specifies the
   outcome of the request. The request may be denied only if the
   specified group address is invalid (e.g. an individual rather
   than a group address.)

   The Identifier and Sequence Number fields are used to match the
   Reply to the corresponding Request, as with other ICMP
   transactions. If a multicast agent receives a request from a
   host to leave a group to which it does not belong, the agent
   approves the request, under the assumption that the request was
   a retransmission for a lost Reply.

Notes:

   <1>   In reality, Internet addresses (individual or group) are bound
         to network interfaces or network attachment points, not the host
         machines per se.

   <2>   In this procedure call notation, the arguments for an operation
         are listed in parentheses after the operation name, and the
         returned values, if any, are listed after a --> symbol.

   <3>   One unicast transmission from sender to gateway and one
         multicast transmission from gateway to local group members

   <4>   This protocol may eventually be replaced by a more general
         reliable transaction protocol designed for this type of
         client/server interaction, as suggested in RFC-955 [5].

   <5>   Multicast agents can use an ICMP Echo message to determine if a
         group has any current members. The Echo message should be
         transmitted several times before deciding the group address is
         no longer in use.

References

    [1]   L. Aguilar. Datagram Routing for Internet Multicasting. In ACM
           SIGCOMM '84 Communications Architectures and Protocols, pages
           58-63. ACM, June, 1984.

    [2]   E. J. Berglund and D. R. Cheriton. Amaze: A distributed
           multi-player game program using the distributed V kernel. In
           Proceedings of the Fourth International Conference on
           Distributed Systems. IEEE, May, 1984.

    [3]   A. D. Birrell et al. Grapevine: an exercise in distributed
           computing. Communications of the ACM 25(4):260-274, April,
           1982.

    [4]   D. R. Boggs. Internet Broadcasting. PhD thesis, Stanford
           University, January, 1982.

    [5]   R. Braden. Towards a Transport Service for Transaction
           Processing Applications. Technical Report RFC-919, SRI Network
           Information Center, September, 1985.

    [6]   J-M. Chang. Simplifying Distributed Database Design by Using a
           Broadcast Network. In SIGMOD '84. ACM, June, 1984.

    [7]   D. R. Cheriton and S. E. Deering. Host Groups: A Multicast
           Extension for Datagram Internetworks. In Proceedings of the
           Ninth Data Communications Symposium. ACM/IEEE, September, 1985.

    [8]   D. R. Cheriton and W. Zwaenepoel. Distributed Process Groups in
           the V Kernel. ACM Transactions on Computer Systems 3(3), May,
           1985.

    [9]   F. Cristian et al. Atomic Broadcast: from simple message
           diffusion to Byzantine agreement. In 15th International
           Conference on Fault Tolerant Computing. , Ann Arbor, Michigan,
           June, 1985.

    [10]   Y. K. Dalal and R. M. Metcalfe. Reverse Path Forwarding of
           Broadcast Packets. Communications of the ACM 21(2):1040-1047,
           December, 1978.

    [11]   H. Forsdick. MMCF: A Multi-Media Conferencing Facility.
           personal communication.

   [12] K. Lebowitz and D. Mankins. Multi-network Broadcasting within
         the Internet.Technical Report RFC-947, SRI Network Information
         Center, June, 1985.

   [13] J. Mogul. Broadcasting Internet Datagrams. Technical Report
         RFC-919, SRI Network Information Center, October, 1984.

   [14] J. Postel. Internet Protocol. Technical Report RFC-791, SRI
         Network Information Center, September, 1981.

   [15] J. Postel. Internet Control Message Protocol. Technical Report
         RFC-792, SRI Network Information Center, September, 1981.

   [16] D. W, Wall. Mechanisms for Broadcast and Selective Broadcast.
         Technical Report 190, Computer Systems Laboratory, Stanford
         University, June, 1980.

   [17] J. K. Reynolds and J. Postel. Assigned Numbers. Technical
         Report RFC-960, SRI Network Information Center, September,
         1981.