Internet Control Message Protocol (ICMPv6)
for the Internet Protocol Version 6 (IPv6)
Specification

Status of this Memo

   This document specifies an Internet standards track protocol for the
   Internet community, and requests discussion and suggestions for
   improvements. Please refer to the current edition of the "Internet
   Official Protocol Standards" (STD 1) for the standardization state
   and status of this protocol. Distribution of this memo is unlimited.

Abstract

   This document specifies a set of Internet Control Message Protocol
   (ICMP) messages for use with version 6 of the Internet Protocol
   (IPv6). The Internet Group Management Protocol (IGMP) messages
   specified in STD 5, RFC 1112 have been merged into ICMP, for IPv6,
   and are included in this document.

Table of Contents

1. Introduction

   The Internet Protocol, version 6 (IPv6) is a new version of IP. IPv6
   uses the Internet Control Message Protocol (ICMP) as defined for IPv4
   [RFC-792], with a number of changes. The Internet Group Membership
   Protocol (IGMP) specified for IPv4 [RFC-1112] has also been revised
   and has been absorbed into ICMP for IPv6. The resulting protocol is
   called ICMPv6, and has an IPv6 Next Header value of 58.

   This document describes the format of a set of control messages used
   in ICMPv6. It does not describe the procedures for using these
   messages to achieve functions like Path MTU discovery or multicast
   group membership maintenance; such procedures are described in other
   documents (e.g., [RFC-1112, RFC-1191]). Other documents may also
   introduce additional ICMPv6 message types, such as Neighbor Discovery
   messages [IPv6-DISC], subject to the general rules for ICMPv6
   messages given in section 2 of this document.

   Terminology defined in the IPv6 specification [IPv6] and the IPv6
   Routing and Addressing specification [IPv6-ADDR] applies to this
   document as well.

2. ICMPv6 (ICMP for IPv6)

   ICMPv6 is used by IPv6 nodes to report errors encountered in
   processing packets, and to perform other internet-layer functions,
   such as diagnostics (ICMPv6 "ping") and multicast membership
   reporting. ICMPv6 is an integral part of IPv6 and MUST be fully
   implemented by every IPv6 node.

2.1 Message General Format

   ICMPv6 messages are grouped into two classes: error messages and
   informational messages. Error messages are identified as such by
   having a zero in the high-order bit of their message Type field
   values. Thus, error messages have message Types from 0 to 127;
   informational messages have message Types from 128 to 255.

   This document defines the message formats for the following ICMPv6
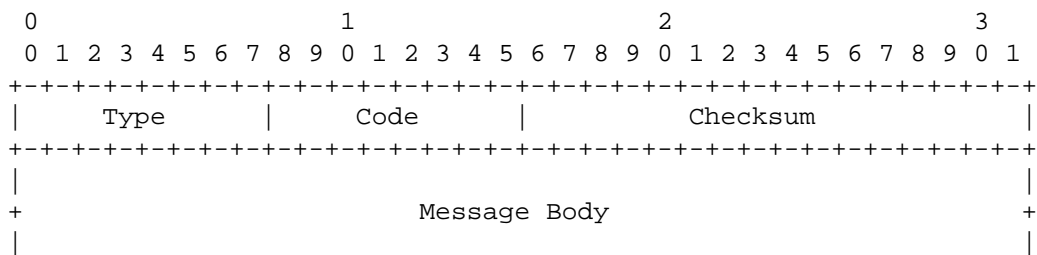   messages:

ICMPv6 error messages:

         1   Destination Unreachable   (see section 3.1)
         2   Packet Too Big   (see section 3.2)
         3   Time Exceeded   (see section 3.3)
         4   Parameter Problem   (see section 3.4)

ICMPv6 informational messages:

         128 Echo Request   (see section 4.1)
         129 Echo Reply   (see section 4.2)
         130 Group Membership Query   (see section 4.3)
         131 Group Membership Report   (see section 4.3)
         132 Group Membership Reduction (see section 4.3)

Every ICMPv6 message is preceded by an IPv6 header and zero or more
IPv6 extension headers. The ICMPv6 header is identified by a Next
Header value of 58 in the immediately preceding header. (NOTE: this
is different than the value used to identify ICMP for IPv4.)

The ICMPv6 messages have the following general format:

     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 |
    +-+
    | |
    +   Message Body   +
    | |

The type field indicates the type of the message. Its value
determines the format of the remaining data.

The code field depends on the message type. It is used to create an
additional level of message granularity.

The checksum field is used to detect data corruption in the ICMPv6
message and parts of the IPv6 header.

2.2 Message Source Address Determination

A node that sends an ICMPv6 message has to determine both the Source
and Destination IPv6 Addresses in the IPv6 header before calculating
the checksum. If the node has more than one unicast address, it must
choose the Source Address of the message as follows:

    (a) If the message is a response to a message sent to one of the
        node's unicast addresses, the Source Address of the reply must
        be that same address.

    (b) If the message is a response to a message sent to a multicast or
        anycast group in which the node is a member, the Source Address
        of the reply must be a unicast address belonging to the
        interface on which the multicast or anycast packet was received.

    (c) If the message is a response to a message sent to an address
        that does not belong to the node, the Source Address should be
        that unicast address belonging to the node that will be most
        helpful in diagnosing the error. For example, if the message is
        a response to a packet forwarding action that cannot complete
        successfully, the Source Address should be a unicast address
        belonging to the interface on which the packet forwarding
        failed.

    (d) Otherwise, the node's routing table must be examined to
        determine which interface will be used to transmit the message
        to its destination, and a unicast address belonging to that
        interface must be used as the Source Address of the message.

2.3 Message Checksum Calculation

   The checksum is the 16-bit one's complement of the one's complement
   sum of the entire ICMPv6 message starting with the ICMPv6 message
   type field, prepended with a "pseudo-header" of IPv6 header fields,
   as specified in [IPv6, section 8.1]. The Next Header value used in
   the pseudo-header is 58. (NOTE: the inclusion of a pseudo-header in
   the ICMPv6 checksum is a change from IPv4; see [IPv6] for the
   rationale for this change.)

   For computing the checksum, the checksum field is set to zero.

2.4 Message Processing Rules

   Implementations MUST observe the following rules when processing
   ICMPv6 messages (from [RFC-1122]):

   (a) If an ICMPv6 error message of unknown type is received, it MUST
       be passed to the upper layer.

   (b) If an ICMPv6 informational message of unknown type is received,
       it MUST be silently discarded.

(c) Every ICMPv6 error message (type < 128) includes as much of the
    IPv6 offending (invoking) packet (the packet that caused the
    error) as will fit without making the error message packet
    exceed 576 octets.

(d) In those cases where the internet-layer protocol is required to
    pass an ICMPv6 error message to the upper-layer protocol, the
    upper-layer protocol type is extracted from the original packet
    (contained in the body of the ICMPv6 error message) and used to
    select the appropriate upper-layer protocol entity to handle the
    error.

    If the original packet had an unusually large amount of
    extension headers, it is possible that the upper-layer protocol
    type may not be present in the ICMPv6 message, due to truncation
    of the original packet to meet the 576-octet limit. In that
    case, the error message is silently dropped after any IPv6-layer
    processing.

(e) An ICMPv6 error message MUST NOT be sent as a result of
    receiving:

    (e.1) an ICMPv6 error message, or

    (e.2) a packet destined to an IPv6 multicast address (there are
          two exceptions to this rule: (1) the Packet Too Big
          Message - Section 3.2 - to allow Path MTU discovery to
          work for IPv6 multicast, and (2) the Parameter Problem
          Message, Code 2 - Section 3.4 - reporting an unrecognized
          IPv6 option that has the Option Type highest-order two
          bits set to 10), or

    (e.3) a packet sent as a link-layer multicast, (the exception
          from e.2 applies to this case too), or

    (e.4) a packet sent as a link-layer broadcast, (the exception
          from e.2 applies to this case too), or

    (e.5) a packet whose source address does not uniquely identify
          a single node -- e.g., the IPv6 Unspecified Address, an
          IPv6 multicast address, or an address known by the ICMP
          message sender to be an IPv6 anycast address.

(f) Finally, to each sender of an erroneous data packet, an IPv6
    node MUST limit the rate of ICMPv6 error messages sent, in order
    to limit the bandwidth and forwarding costs incurred by the
    error messages when a generator of erroneous packets does not
    respond to those error messages by ceasing its transmissions.

There are a variety of ways of implementing the rate-limiting
function, for example:

   (f.1) Timer-based - for example, limiting the rate of
         transmission of error messages to a given source, or to
         any source, to at most once every T milliseconds.

   (f.2) Bandwidth-based - for example, limiting the rate at
         which error messages are sent from a particular interface
         to some fraction F of the attached link's bandwidth.

The limit parameters (e.g., T or F in the above examples) MUST
be configurable for the node, with a conservative default value
(e.g., T = 1 second, NOT 0 seconds, or F = 2 percent, NOT 100
percent).

The following sections describe the message formats for the above
ICMPv6 messages.

3. ICMPv6 Error Messages

3.1 Destination Unreachable Message
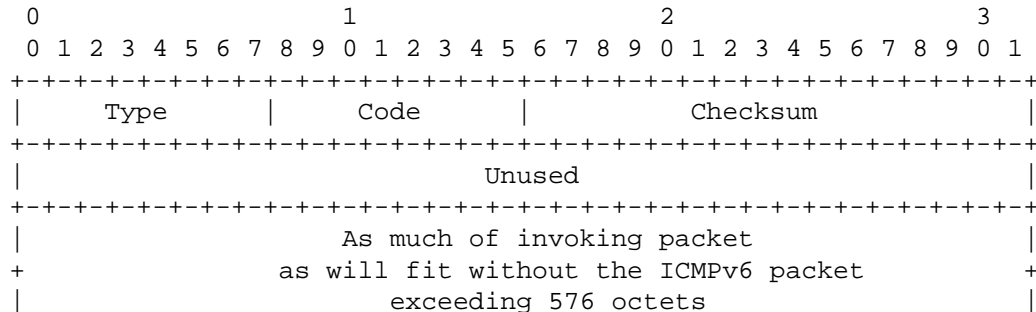
```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |     Code      |          Checksum             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Unused                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                   As much of invoking packet                  |
   +                as will fit without the ICMPv6 packet          +
   |                      exceeding 576 octets                     |
```

   IPv6 Fields:

   Destination Address

                 Copied from the Source Address field of the invoking
                 packet.

   ICMPv6 Fields:

   Type   1

   Code   0 - no route to destination
                 1 - communication with destination
                     administratively prohibited
                 2 - not a neighbor
                 3 - address unreachable
                 4 - port unreachable

   Unused   This field is unused for all code values.
                 It must be initialized to zero by the sender
                 and ignored by the receiver.
   Description

   A Destination Unreachable message SHOULD be generated by a router, or
   by the IPv6 layer in the originating node, in response to a packet
   that cannot be delivered to its destination address for reasons other
   than congestion. (An ICMPv6 message MUST NOT be generated if a
   packet is dropped due to congestion.)

   If the reason for the failure to deliver is lack of a matching entry
   in the forwarding node's routing table, the Code field is set to 0
   (NOTE: this error can occur only in nodes that do not hold a "default
   route" in their routing tables).

   If the reason for the failure to deliver is administrative
   prohibition, e.g., a "firewall filter", the Code field is set to 1.

   If the reason for the failure to deliver is that the next destination
   address in the Routing header is not a neighbor of the processing
   node but the "strict" bit is set for that address, then the Code
   field is set to 2.

   If there is any other reason for the failure to deliver, e.g.,
   inability to resolve the IPv6 destination address into a
   corresponding link address, or a link-specific problem of some sort,
   then the Code field is set to 3.

   A destination node SHOULD send a Destination Unreachable message with
   Code 4 in response to a packet for which the transport protocol
   (e.g., UDP) has no listener, if that transport protocol has no
   alternative means to inform the sender.

   Upper layer notification

   A node receiving the ICMPv6 Destination Unreachable message MUST
   notify the upper-layer protocol.

3.2 Packet Too Big Message
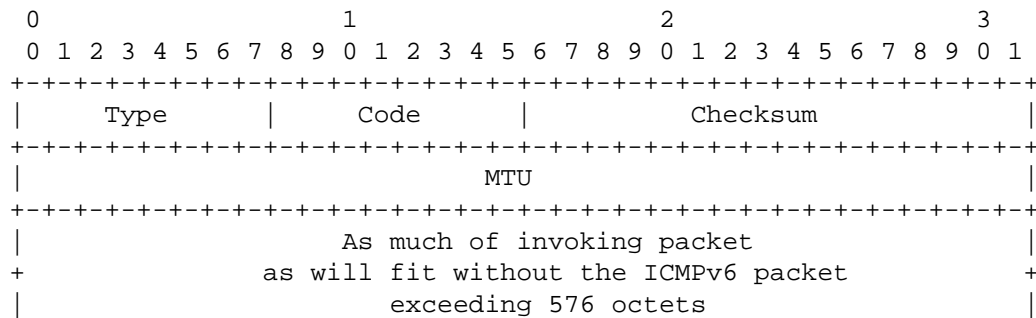
```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |     Code      |          Checksum             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                             MTU                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                  As much of invoking packet                   |
   +              as will fit without the ICMPv6 packet            +
   |                    exceeding 576 octets                       |
```

IPv6 Fields:

Destination Address

          Copied from the Source Address field of the invoking
          packet.

ICMPv6 Fields:

Type   2

Code   0

MTU   The Maximum Transmission Unit of the next-hop link.

Description

A Packet Too Big MUST be sent by a router in response to a packet
that it cannot forward because the packet is larger than the MTU of
the outgoing link. The information in this message is used as part
of the Path MTU Discovery process [RFC-1191].

Sending a Packet Too Big Message makes an exception to one of the
rules of when to send an ICMPv6 error message, in that unlike other
messages, it is sent in response to a packet received with an IPv6
multicast destination address, or a link-layer multicast or link-
layer broadcast address.

Upper layer notification

An incoming Packet Too Big message MUST be passed to the upper-layer
protocol.

3.3 Time Exceeded Message
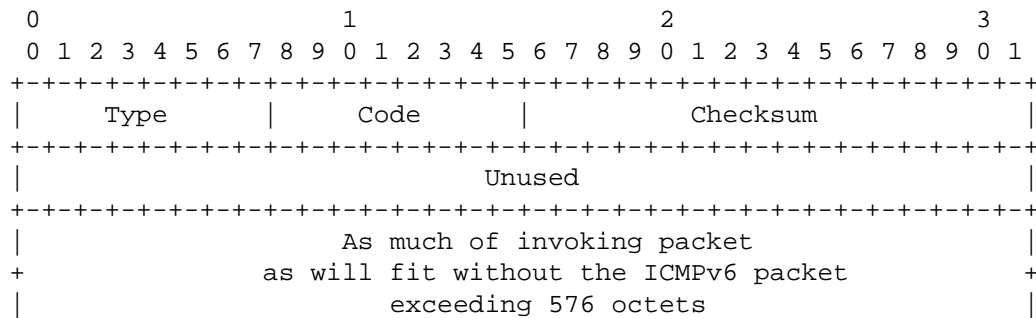
```
   0                   1                   2                   3
   0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |     Type      |     Code      |          Checksum             |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |                           Unused                              |
  +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
  |                 As much of invoking packet                    |
  +                as will fit without the ICMPv6 packet          +
  |                    exceeding 576 octets                       |
```

IPv6 Fields:

Destination Address

               Copied from the Source Address field of the invoking
               packet.

ICMPv6 Fields:

Type   3

Code   0 - hop limit exceeded in transit

               1 - fragment reassembly time exceeded

Unused   This field is unused for all code values.
               It must be initialized to zero by the sender
               and ignored by the receiver.

Description

If a router receives a packet with a Hop Limit of zero, or a router
decrements a packet's Hop Limit to zero, it MUST discard the packet
and send an ICMPv6 Time Exceeded message with Code 0 to the source of
the packet. This indicates either a routing loop or too small an
initial Hop Limit value.

The router sending an ICMPv6 Time Exceeded message with Code 0 SHOULD
consider the receiving interface of the packet as the interface on
which the packet forwarding failed in following rule (d) for
selecting the Source Address of the message.

Upper layer notification

An incoming Time Exceeded message MUST be passed to the upper-layer
protocol.

3.4 Parameter Problem Message
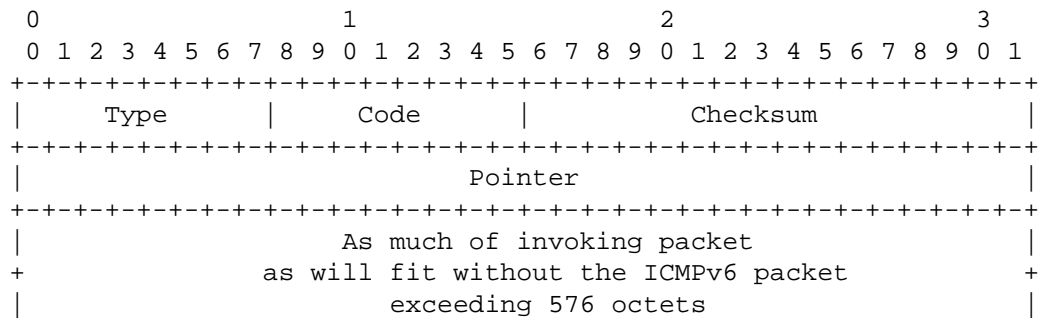
```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Type      |     Code      |          Checksum             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                            Pointer                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                    As much of invoking packet                 |
   +                 as will fit without the ICMPv6 packet        +
   |                       exceeding 576 octets                    |
```

   IPv6 Fields:

   Destination Address

                 Copied from the Source Address field of the invoking
                 packet.

   ICMPv6 Fields:

   Type   4

   Code   0 - erroneous header field encountered

                 1 - unrecognized Next Header type encountered

                 2 - unrecognized IPv6 option encountered

   Pointer   Identifies the octet offset within the
                 invoking packet where the error was detected.

                 The pointer will point beyond the end of the ICMPv6
                 packet if the field in error is beyond what can fit
                 in the 576-byte limit of an ICMPv6 error message.

   Description

   If an IPv6 node processing a packet finds a problem with a field in
   the IPv6 header or extension headers such that it cannot complete
   processing the packet, it MUST discard the packet and SHOULD send an
   ICMPv6 Parameter Problem message to the packet's source, indicating
   the type and location of the problem.

   The pointer identifies the octet of the original packet's header
   where the error was detected. For example, an ICMPv6 message with
   Type field = 4, Code field = 1, and Pointer field = 40 would indicate
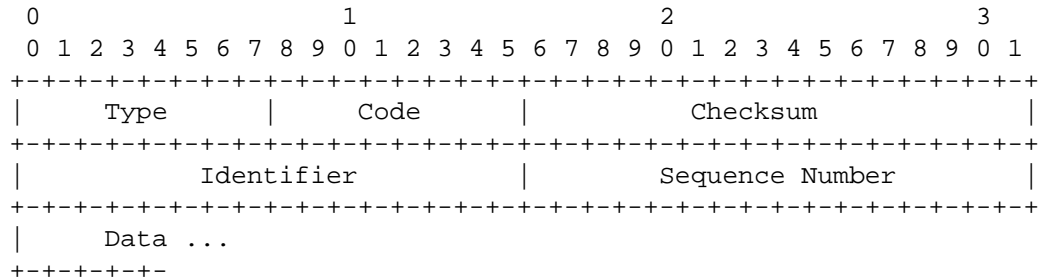
that the IPv6 extension header following the IPv6 header of the
original packet holds an unrecognized Next Header field value.

Upper layer notification

A node receiving this ICMPv6 message MUST notify the upper-layer
protocol.

4. ICMPv6 Informational Messages

4.1 Echo Request 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        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Data ...
    +-+-+-+-+-
```

   IPv6 Fields:

   Destination Address

               Any legal IPv6 address.

   ICMPv6 Fields:

   Type   128

   Code   0

   Identifier   An identifier to aid in matching Echo Replies
                 to this Echo Request. May be zero.

   Sequence Number

               A sequence number to aid in matching Echo Replies
               to this Echo Request. May be zero.

   Data   Zero or more octets of arbitrary data.

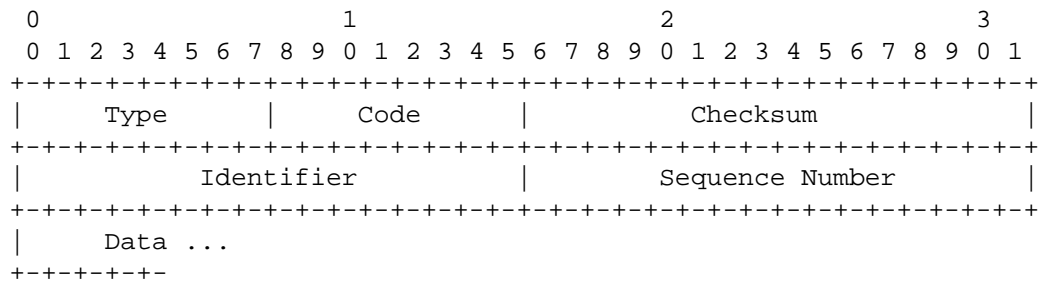   Description

   Every node MUST implement an ICMPv6 Echo responder function that
   receives Echo Requests and sends corresponding Echo Replies. A node
   SHOULD also implement an application-layer interface for sending Echo
   Requests and receiving Echo Replies, for diagnostic purposes.

   Upper layer notification

   A node receiving this ICMPv6 message MAY notify the upper-layer
   protocol.

4.2 Echo 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        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Data ...
    +-+-+-+-+-
```

   IPv6 Fields:

   Destination Address

                    Copied from the Source Address field of the invoking
                    Echo Request packet.

   ICMPv6 Fields:

   Type   129

   Code   0

   Identifier   The identifier from the invoking Echo Request message.

   Sequence   The sequence number from the invoking Echo Request
   Number   message.

   Data   The data from the invoking Echo Request message.

   Description

   Every node MUST implement an ICMPv6 Echo responder function that
   receives Echo Requests and sends corresponding Echo Replies. A node
   SHOULD also implement an application-layer interface for sending Echo
   Requests and receiving Echo Replies, for diagnostic purposes.

   The source address of an Echo Reply sent in response to a unicast
   Echo Request message MUST be the same as the destination address of
   that Echo Request message.

   An Echo Reply SHOULD be sent in response to an Echo Request message
   sent to an IPv6 multicast address. The source address of the reply
   MUST be a unicast address belonging to the interface on which the
   multicast Echo Request message was received.

The data received in the ICMPv6 Echo Request message MUST be returned
entirely and unmodified in the ICMPv6 Echo Reply message, unless the
Echo Reply would exceed the MTU of the path back to the Echo
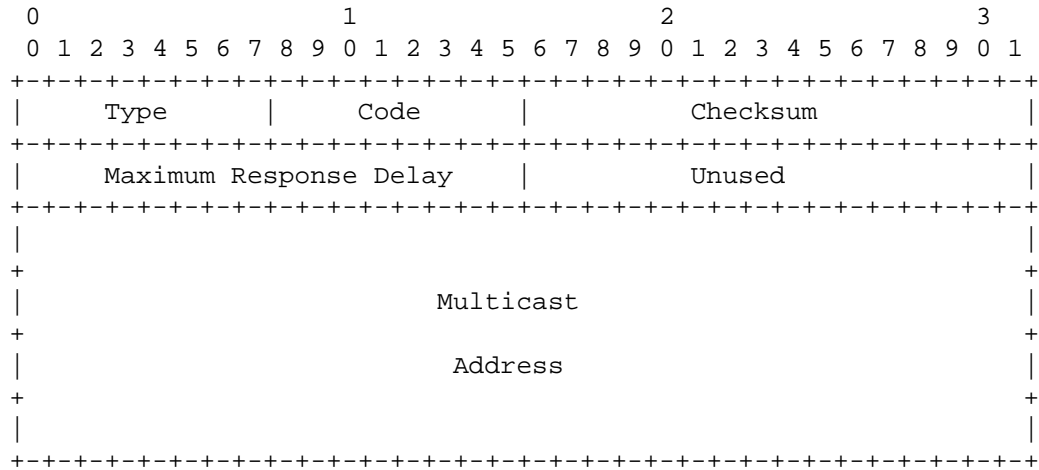requester, in which case the data is truncated to fit that path MTU.

Upper layer notification

Echo Reply messages MUST be passed to the ICMPv6 user interface,
unless the corresponding Echo Request originated in the IP layer.

4.3 Group Membership Messages

   The ICMPv6 Group Membership Messages have the following format:

```
     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             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Maximum Response Delay     |           Unused             |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                                                               |
    +                                                               +
    |                          Multicast                            |
    +                                                               +
    |                           Address                             |
    +                                                               +
    |                                                               |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   IPv6 Fields:

   Destination Address

              In a Group Membership Query message, the multicast
              address of the group being queried, or the Link-Local
              All-Nodes multicast address.

              In a Group Membership Report or a Group Membership
              Reduction message, the multicast address of the
              group being reported or terminated.

   Hop Limit   1

   ICMPv6 Fields:

   Type   130 - Group Membership Query
                  131 - Group Membership Report
                  132 - Group Membership Reduction

   Code   0

   Maximum Response Delay

              In Query messages, the maximum time that responding
              Report messages may be delayed, in milliseconds.

                    In Report and Reduction messages, this field is
                    is initialized to zero by the sender and ignored by
                    receivers.

        Unused   Initialized to zero by the sender; ignored by receivers.

        Multicast Address

                    The address of the multicast group about which the
                    message is being sent. In Query messages, the Multicast
                    Address field may be zero, implying a query for all
                    groups.

        Description

        The ICMPv6 Group Membership messages are used to convey information
        about multicast group membership from nodes to their neighboring
        routers. The details of their usage is given in [RFC-1112].

5. References

    [IPv6]   Deering, S., and R. Hinden, "Internet Protocol, Version
                  6, Specification", RFC 1883, Xerox PARC, Ipsilon
                  Networks, December 1995.

    [IPv6-ADDR]   Hinden, R., and S. Deering, Editors, "IP Version 6
                  Addressing Architecture", RFC 1884, Ipsilon Networks,
                  Xerox PARC, December 1995.

    [IPv6-DISC]   Narten, T., Nordmark, E., and W. Simpson, "Neighbor
                  Discovery for IP Version 6 (IPv6)", Work in Progress.

    [RFC-792]   Postel, J., "Internet Control Message Protocol", STD 5,
                  RFC 792, USC/Information Sciences Institute, September
                  1981.

    [RFC-1112]   Deering, S., "Host Extensions for IP Multicasting", STD
                  5, RFC 1112, Stanford University, August 1989.

    [RFC-1122]   Braden, R., "Requirements for Internet Hosts -
                  Communication Layers", STD 3, RFC 1122, USC/Information
                  Sciences Institute, October 1989.

    [RFC-1191]   Mogul, J., and S. Deering, "Path MTU Discovery", RFC
                  1191, DECWRL, Stanford University, November 1990.

6. Acknowledgements

    The document is derived from previous ICMP drafts of the SIPP and
    IPng working group.

    The IPng working group and particularly Robert Elz, Jim Bound, Bill
    Simpson, Thomas Narten, Charlie Lynn, Bill Fink, and Scott Bradner
    (in chronological order) provided extensive review information and
    feedback.

7. Security Considerations

    Security issues are not discussed in this memo.

Authors' Addresses:

   Alex Conta   Stephen Deering
   Digital Equipment Corporation   Xerox Palo Alto Research Center
   110 Spitbrook Rd   3333 Coyote Hill Road
   Nashua, NH 03062   Palo Alto, CA 94304

   Phone: +1-603-881-0744   Phone: +1-415-812-4839
   EMail: conta@zk3.dec.com   EMail: deering@parc.xerox.com