Management Information Base for IP Version 6
<draft-haskin-onishi-ipv6-mib-00.txt>

Wed Oct 9 14:39:52 EDT 1996

Dimitry Haskin
Steve Onishi

Bay Networks, Inc.

Status of this Memo

Abstract

This paper is intended to serve as a framework for definition of IPv6 MIB objects.

This memo defines an experimental portion of the Management Information Base (MIB) for use with network management protocols in the IPv6-based internets.

This document specifies a MIB module in a manner that is both compliant to the SNMPv2 SMI, and semantically identical to the peer SNMPv1 definitions.

Table of Contents

1. The SNMPv2 Network Management Framework 3
1.1 Object Definitions 3
2. Overview 4
3. IPv6 Address Representation 5
4. Definition of Textual Conventions 7
5. IPv6 MIB Definitions 9
5.1. The IPv6 General Group 9
5.2. The ICMPv6 Group 44
5.3. The UDP Group 61
5.4. The TCP Group 65
6. Acknowledgements 74
7. References 74
8. Security Considerations 75
9. Authors' Addresses.................................. 75

1. The SNMPv2 Network Management Framework

The SNMPv2 Network Management Framework presently consists of three major components. They are:

- the SMI, described in RFC 1902 [1] - the mechanisms used for describing and naming objects for the purpose of management.

- the MIB-II, described in RFC 1213/STD 17 [3] - the core set of managed objects for the Internet suite of protocols.

- RFC 1157 [4] and RFC 1905 [5] which define two versions of the protocol used for network access to managed objects.

The Framework permits new objects to be defined for the purpose of experimentation and evaluation.

1.1. Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, each object object type is named by an OBJECT IDENTIFIER, an administratively assigned name. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to refer to the object type.

2. Overview

The objects in this MIB are arranged into the following groups:

- IPv6 General
- ICMPv6
- UDP
- TCP

These groups are the basic unit of conformance: if the semantics of a group is applicable to an implementation, then it must implement all objects in that group. For example, an implementation must implement the TCP group if and only if it implements the TCP over IPv6 protocol. At minimum, implementations must implement IPv6 General and ICMPv6 groups.

The IPv6 General group consists of 6 tables:

- ipv6IfTable

  The IPv6 Interfaces table contains information on the entity's IPv6 interfaces.

- ipv6IfStatsTable

  This table contains information on the traffic statistics of the entity's IPv6 interfaces.

- ipv6AddrPrefixTable

  The IPv6 Address Prefix table contains information on Address Prefixes that are associated with the entity's IPv6 interfaces.

- ipv6AddrTable

  This table contains the addressing information relevant to the entity's IPv6 interfaces.

- ipv6RouteTable

  The IPv6 routing table contains an entry for each valid IPv6 unicast route that can be used for packet forwarding determination.

- ipv6NetToMediaEntryTable

  The IPv6 address translation table contain the IPv6 Address to `physical' address equivalences.

The ICMPv6 group consists of 1 table:

- ipv6IfIcmpTable

  This table contains information on the ICMPv6 statistics of the entity's IPv6 interfaces.

The UDP group consists of 1 table:

- ipv6UdpTable

  This table contains information on the entity's UDP end-points on which a local application is currently accepting datagrams.

The TCP group consists of 1 table:

- ipv6TcpConnTable

  This table contains information about the entity's existing TCP connections.

In addition to the object definitions, this document defines a number of textual convention to represent object types that are specific to IPv6.

3. IPv6 Address Representation

The IPv6 MIB defined in this memo uses an OCTET STRING of length 16 to represent 128-bit IPv6 address in network byte-order. This approach allows to implement IPv6 MIB without requiring any changes to the SNMPv2 SMI and compliant SNMP implementations. A drawback of this approach is that, if an IPv6 address is used for the object instance identification, **16 sub-identifiers are used to form the instance-identifier.**

An alternative approach would be to introduce a new object

type that represents an IPv6 address as a string of four 32-bit unsigned integers. Only 4 sub-identifiers would be used to form the instance-identifier from an IPv6 address. The disadvantage of this solution is that it would require changes to the SMI and SNMPv2 implementations to support IPv6 MIB.

4. Definition of Textual Conventions

```
   IPV6-TC DEFINITIONS ::= BEGIN

   IMPORTS
        Integer32                FROM SNMPv2-SMI
        TEXTUAL-CONVENTION       FROM SNMPv2-TC;

   -- IPv6 MIB sub-tree.
   -- It is not the best but a convenient temporary place
   -- for this identifier.
   ipv6 OBJECT IDENTIFIER ::= {experimental XX}

   -- definition of textual conventions
   Ipv6Address ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "2x:"
        STATUS       current
        DESCRIPTION
          "This data type is used to model IPv6 addresses.
           This is a binary string of 16 octets in network
           byte-order."
        SYNTAX       OCTET STRING (SIZE (16))

   Ipv6AddressPrefix ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "2x:"
        STATUS       current
        DESCRIPTION
          "This data type is used to model IPv6 address
          prefixes. This is a binary string of up to 16
          octets in network byte-order."
        SYNTAX       OCTET STRING (SIZE (0..16))

   Ipv6AddressToken ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "2x:"
        STATUS       current
        DESCRIPTION
          "This data type is used to model IPv6 address
          tokens. This is a binary string of up to 6
          octets in network byte-order."
        SYNTAX      OCTET STRING (SIZE (0..6))

   Ipv6IfIndex ::= TEXTUAL-CONVENTION
        DISPLAY-HINT "d"
        STATUS       current
```

```
        DESCRIPTION
          "A unique value, greater than zero, for each
          interface or interface sub-layer in the managed
          system.  It is recommended that values are assigned
          contiguously starting from 1.  The value for each
          interface sub-layer must remain constant at least
          from one re-initialization of the entity's network
          management system to the next re-initialization."
        SYNTAX       Integer32 (1..2147483647)

   END
```

5. IPv6 MIB Definitions

5.1. The IPv6 General Group

```
    IPV6-MIB DEFINITIONS ::= BEGIN

    IMPORTS
        MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
        Counter32, Integer32, Unsigned32      FROM SNMPv2-SMI
        DisplayString, PhysAddress, TruthValue, TimeStamp,
        VariablePointer, RowPointer           FROM SNMPv2-TC
        MODULE-COMPLIANCE, OBJECT-GROUP       FROM SNMPv2-CONF
        Ipv6IfIndex, Ipv6Address, Ipv6AddressPrefix,
        ipv6, Ipv6AddressToken                FROM IPV6-TC;


    ipv6MIB MODULE-IDENTITY
        LAST-UPDATED "9610042155Z"
        ORGANIZATION "IETF IPv6 MIB Working Group"
        CONTACT-INFO
          "           Dimitry Haskin

              Postal: Bay Networks, Inc.
                      2 Federal St.
                      Billerica, MA  01821
                      US

                 Tel: +1-508-916-8124
              E-mail: dhaskin@baynetworks.com

                      Steve Onishi

              Postal: Bay Networks, Inc.
                      3 Federal Street
                      Billerica, MA 01821
                      US

                 Tel: +1-508-916-3816
              E-mail: sonishi@baynetworks.com"
        DESCRIPTION
          "The MIB module for entities implementing the IPv6
           protocol."
        ::= { ipv6  1 }
```

```
    -- the IPv6 general group

    ipv6MIBObjects       OBJECT IDENTIFIER   ::= { ipv6MIB 1 }


    ipv6Forwarding OBJECT-TYPE
        SYNTAX     INTEGER {
                     forwarding(1),    -- acting as a router

                                       -- NOT acting as
                     notForwarding(2)  -- a router
                   }
         MAX-ACCESS read-write
         STATUS     current
         DESCRIPTION
           "The indication of whether this entity is acting
           as an IPv6 router in respect to the forwarding of
           datagrams received by, but not addressed to, this
           entity.  IPv6 routers forward datagrams.  IPv6
           hosts do not (except those source-routed via the
           host).

           Note that for some managed nodes, this object may
           take on only a subset of the values possible.
           Accordingly, it is appropriate for an agent to
           return a `badValue' response if a management
           station attempts to change this object to an
           inappropriate value."
         ::= { ipv6MIBObjects 1 }

    ipv6DefaultHopLimit OBJECT-TYPE
        SYNTAX  INTEGER(0..255)
        MAX-ACCESS  read-write
        STATUS  current
        DESCRIPTION
           "The default value inserted into the Hop Limit
           field of the IPv6 header of datagrams originated at
           this entity, whenever a Hop Limit value is not
           supplied by the transport layer protocol."
        DEFVAL  { 60 }
        ::= { ipv6MIBObjects 2 }

    ipv6InReceives OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
```

```
        STATUS  current
        DESCRIPTION
           "The total number of input datagrams received from
           interfaces, including those received in error."
        ::= { ipv6MIBObjects 3 }

    ipv6InHdrErrors OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS  current
        DESCRIPTION
          "The total number of input datagrams discarded due
          to errors in their IPv6 headers, including version
          number mismatch, other format errors, hop count
          exceeded, errors discovered in processing their
          IPv6 options, etc."
        ::= { ipv6MIBObjects 4 }

    ipv6TooBigErrors OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The total number of datagrams that could not be
          forwarded because their size exceeded the Path MTU"
        ::= { ipv6MIBObjects 5 }

    ipv6InAddrErrors OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The total number of input datagrams discarded
          because the IPv6 address in their IPv6 header's
          destination field was not a valid address to be
          received at this entity.  This count includes
          invalid addresses (e.g., ::0) and unsupported
          addresses (e.g., addresses with unallocated
          prefixes).  For entities which are not IPv6 routers
          and therefore do not forward datagrams, this
          counter includes datagrams discarded because the
          destination address was not a local address."
        ::= { ipv6MIBObjects 6 }

    ipv6TruncatedPkts OBJECT-TYPE
```

```
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of input datagrams discarded
          because datagram frame didn't carry enough data"
        ::= { ipv6MIBObjects 7 }

    ipv6ForwDatagrams OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The total number of input datagrams for which this
          entity was not their final IPv6 destination, as a
          result of which an attempt was made to find a
          route to forward them to that final destination.
          In entities which do not act as IPv6 routers, this
          counter will include only those packets which were
          Source-Routed via this entity, and the Source-
          Route option processing was successful."
        ::= { ipv6MIBObjects 8 }

    ipv6InUnknownProtos OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of locally-addressed datagrams
          received successfully but discarded because of an
          unknown or unsupported protocol."
        ::= { ipv6MIBObjects 9 }

    ipv6InDiscards OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The  total number of input IPv6 datagrams for which
          no problems were encountered to prevent their
          continued processing, but which were discarded
          (e.g., for lack of buffer space).  Note that this
          counter does not include any datagrams discarded
          while awaiting re-assembly."
        ::= { ipv6MIBObjects 10 }
```

```
    ipv6InDelivers OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of input datagrams successfully
          delivered to IPv6 user-protocols (including
          ICMPv6)."
        ::= { ipv6MIBObjects 11 }

    ipv6OutRequests OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of IPv6 datagrams which local IPv6
          user-protocols (including ICMPv6) supplied to IPv6
          requests for transmission.  Note that this counter
          in does not include any datagrams counted in
          ipv6ForwDatagrams."
        ::= { ipv6MIBObjects 12 }

    ipv6OutDiscards OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of output IPv6 datagrams for
          which no problem was encountered to prevent their
          transmission to their destination, but which were
          discarded (e.g., for lack of buffer space).  Note
          that this counter would include datagrams counted
          in ipv6ForwDatagrams if any such packets met this
          discretionary) discard criterion."
        ::= { ipv6MIBObjects 13 }

    ipv6OutNoRoutes OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The number of IPv6 datagrams discarded because
          no route could be found to transmit them to their
          destination.  Note that this counter includes any
          packets counted in ipv6ForwDatagrams which
```

```
          meet this `no-route' criterion.  Note that this
          includes any datagrams which a host cannot route
          because all of its default routers are down."
        ::= { ipv6MIBObjects 14 }

    ipv6ReasmTimeout OBJECT-TYPE
        SYNTAX     Unsigned32
        UNITS      "seconds"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The maximum number of seconds which received
          fragments are held while they are awaiting
          reassembly at this entity."
        ::= { ipv6MIBObjects 15 }

    ipv6ReasmReqds OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The number of IPv6 fragments received which needed
          to be reassembled at this entity."
        ::= { ipv6MIBObjects 16 }

    ipv6ReasmOKs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The number of IPv6 datagrams successfully
           reassembled."
        ::= { ipv6MIBObjects 17 }

    ipv6ReasmFails OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The number of failures detected by the IPv6 re-
          assembly algorithm (for whatever reason: timed
          out, errors, etc).  Note that this is not
          necessarily a count of discarded IPv6 fragments
          since some algorithms can lose track of the number
          of fragments by combining them as they are
```

```
          received."
        ::= { ipv6MIBObjects 18 }

    ipv6FragOKs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The number of IPv6 datagrams that have been
          successfully fragmented at this entity."
        ::= { ipv6MIBObjects 19 }

    ipv6FragFails OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The number of IPv6 datagrams that have been
          discarded because they needed to be fragmented
          at this entity but could not be."
        ::= { ipv6MIBObjects 20 }

    ipv6FragCreates OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of IPv6 datagram fragments that
          have been generated as a result of fragmentation
          at this entity."
        ::= { ipv6MIBObjects 21 }

    ipv6InMcastPkts OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The total number of multicast packets received
          by this entity"
        ::= { ipv6MIBObjects 22 }

    ipv6OutMcastPkts OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
```

```
        DESCRIPTION
          "The total number of multicast packets transmitted
          by this entity"
        ::= { ipv6MIBObjects 23 }

   ipv6Interfaces  OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of IPv6 interfaces (regardless of
           their current state) present on this system."
        ::= { ipv6MIBObjects 24 }

   ipv6IfTableLastChange  OBJECT-TYPE
        SYNTAX      TimeStamp
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The value of sysUpTime at the time of the last
          insertion or removal of an entry in the
          ipv6IfTable. If the number of entries has been
          unchanged since the last re-initialization of
          the local network management subsystem, then this
          object contains a zero value."
        ::= { ipv6MIBObjects 25 }


   -- the IPv6 Interfaces table

   ipv6IfTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF Ipv6IfEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "The IPv6 Interfaces table contains information
          on the entity's internetwork-layer interfaces.
          An IPv6 interface constitutes a logical network
          layer attachment to the layer immediately below
          IPv6 including internet layer 'tunnels', such as
          tunnels over IPv4 or IPv6 itself."
        ::= { ipv6MIBObjects 26 }

    ipv6IfEntry OBJECT-TYPE
        SYNTAX     Ipv6IfEntry
```

```
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "An interface entry containing objects
           about a particular IPv6 interface."
        INDEX   { ipv6IfIndex }
        ::= { ipv6IfTable 1 }

    Ipv6IfEntry ::= SEQUENCE {
            ipv6IfIndex             Ipv6IfIndex,
            ipv6IfDescr             DisplayString,
            ipv6IfLowerLayer        VariablePointer,
            ipv6IfEffectiveMtu      Unsigned32,
            ipv6IfReasmMaxSize      Unsigned32,
            ipv6IfToken             Ipv6AddressToken,
            ipv6IfTokenLength       INTEGER,
            ipv6IfPhysicalAddress   PhysAddress,
            ipv6IfAdminStatus       INTEGER,
            ipv6IfOperStatus        INTEGER,
            ipv6IfLastChange        TimeStamp
        }

    ipv6IfIndex OBJECT-TYPE
        SYNTAX     Ipv6IfIndex
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "A unique non-zero value identifying
           the particular IPv6 interface."
        ::= { ipv6IfEntry 1 }

    ipv6IfDescr OBJECT-TYPE
        SYNTAX     DisplayString
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
          "A textual string containing information about the
          interface.  This string may be set by the network
          management system."
        ::= { ipv6IfEntry 2 }

    ipv6IfLowerLayer OBJECT-TYPE
       SYNTAX      VariablePointer
       MAX-ACCESS  read-only
       STATUS      current
```

```
       DESCRIPTION
         "This object identifies the protocol layer over
          which this network interface operates.  If this
          network interface operates over the data-link
          layer, then the value of this object refers to an
          instance of ifIndex [6]. If this network interface
          operates over an IPv4 interface, the value of this
          object refers to an instance of ipAdEntAddr [3].
          If this network interface operates over another
          IPv6 interface, the value of this object refers to
          an instance of ipv6IfIndex.  If this network
          interface is not currently operating over an active
          protocol layer, then the value of this object
          should be set to the OBJECT ID { 0 0 }."
       ::= { ipv6IfEntry 3 }

    ipv6IfEffectiveMtu OBJECT-TYPE
       SYNTAX      Unsigned32
       UNITS       "octets"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
         "The size of the largest IPv6 packet which can be
         sent/received on the interface, specified in
         octets."
    ::= { ipv6IfEntry 4 }

    ipv6IfReasmMaxSize OBJECT-TYPE
       SYNTAX      Unsigned32 (0..65535)
       UNITS       "octets"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
         "The size of the largest IPv6 datagram which this
         entity can re-assemble from incoming IPv6 fragmented
         datagrams received on this interface."
    ::= { ipv6IfEntry 5 }

    ipv6IfToken OBJECT-TYPE
        SYNTAX      Ipv6AddressToken
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
           "The address token for this interface that is
           (at least) unique on the link this interface is
```

```
           attached to. The address token is combined with
           an address prefix to form an interface address.

           By default, the interface token is autoconfigured
           according to the rules of the link type this
           interface is attached to."
        ::= { ipv6IfEntry 6 }

    ipv6IfTokenLength OBJECT-TYPE
        SYNTAX      INTEGER (0..64)
        UNITS       "bits"
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
          "The length of the address token in bits."
        ::= { ipv6IfEntry 7 }

    ipv6IfPhysicalAddress OBJECT-TYPE
        SYNTAX      PhysAddress
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The interface's physical address. For example, for
          an IPv6 interface attached to an 802.x link, this
          object normally contains a MAC address. Note that
          in some cases this address may differ from the
          address of the interface's protocol sub-layer.  The
          interface's media-specific MIB must define the bit
          and byte ordering and the format of the value of
          this object. For interfaces which do not have such
          an address (e.g., a serial line), this object should
          contain an octet string of zero length."
        ::= { ipv6IfEntry 8 }

   ipv6IfAdminStatus OBJECT-TYPE
       SYNTAX  INTEGER {
                up(1),       -- ready to pass packets
                down(2)
            }
       MAX-ACCESS  read-write
       STATUS      current
       DESCRIPTION
         "The desired state of the interface.  When a managed
         system initializes,  all IPv6 interfaces start with
         ipv6IfAdminStatus in the down(2) state.  As a result
```

```
         of either explicit management action or per
         configuration information retained by the managed
         system,  ipv6IfAdminStatus is then changed to
         the up(1) state (or remains in the down(2) state)."
       ::= { ipv6IfEntry 9 }

   ipv6IfOperStatus OBJECT-TYPE
       SYNTAX  INTEGER {
                up(1),        -- ready to pass packets

                down(2),

                tokenless(3), -- no interface token

                              -- status can not be determined
                unknown(4),   -- for some reason

                notPresent(5) -- some component is missing
               }
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
         "The current operational state of the interface.
         The tokenless(3) state indicates that no valid
         address token is assigned to the interface. This
         state usually indicates that the link-local interface
         address failed Duplicate Address Detection. If
         ipv6IfAdminStatus is down(2) then ipv6IfOperStatus
         should be down(2).  If ipv6IfAdminStatus is changed
         to up(1) then ipv6IfOperStatus should change to up(1)
         if the interface is ready to transmit and receive
         network traffic; it should remain in the down(2) or
         tokenless(3) state if and only if there is a fault
         that prevents it from going to the up(1) state; it
         should remain in the notPresent(5) state if
         the interface has missing (typically, lower layer)
         components."
       ::= { ipv6IfEntry 10 }

   ipv6IfLastChange OBJECT-TYPE
       SYNTAX      TimeStamp
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
           "The value of sysUpTime at the time the interface
```

```
           entered its current operational state.  If the
           current state was entered prior to the last
           re-initialization of the local network management
           subsystem, then this object contains a zero
           value."
       ::= { ipv6IfEntry 11 }


    --  IPv6 Interface Statistics table

    ipv6IfStatsTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF Ipv6IfStatsEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "IPv6 interface traffic statistics"
        ::= { ipv6MIBObjects 27 }

    ipv6IfStatsEntry OBJECT-TYPE
        SYNTAX     Ipv6IfStatsEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "An interface statistics entry containing objects
            at a particular IPv6 interface."
        AUGMENTS { ipv6IfEntry }
        ::= { ipv6IfStatsTable 1 }

    Ipv6IfStatsEntry ::= SEQUENCE {
            ipv6IfStatsInReceives
                Counter32,
            ipv6IfStatsInHdrErrors
                Counter32,
            ipv6IfStatsTooBigErrors
                Counter32,
            ipv6IfStatsInAddrErrors
                Counter32,
            ipv6IfStatsForwDatagrams
                Counter32,
            ipv6IfStatsInUnknownProtos
                Counter32,
            ipv6IfStatsInDiscards
                Counter32,
            ipv6IfStatsInDelivers
                Counter32,
```

```
            ipv6IfStatsOutRequests
                Counter32,
            ipv6IfStatsOutDiscards
                Counter32,
            ipv6IfStatsOutNoRoutes
                Counter32,
            ipv6IfStatsReasmReqds
                Counter32,
            ipv6IfStatsReasmOKs
                Counter32,
            ipv6IfStatsReasmFails
                Counter32,
            ipv6IfStatsFragOKs
                Counter32,
            ipv6IfStatsFragFails
                Counter32,
            ipv6IfStatsFragCreates
                Counter32,
            ipv6IfStatsInMcastPkts
                Counter32,
            ipv6IfStatsOutMcastPkts
                Counter32,
            ipv6IfStatsInTruncatedPkts
                Counter32
        }

    ipv6IfStatsInReceives OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The total number of input datagrams received from
           interfaces, including those received in error."
        ::= { ipv6IfStatsEntry 1 }

    ipv6IfStatsInHdrErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The number of input datagrams discarded due to
           errors in their IPv6 headers, including version
           number mismatch, other format errors, hop count
           exceeded, errors discovered in processing their
           IPv6 options, etc."
```

```
        ::= { ipv6IfStatsEntry 2 }

    ipv6IfStatsTooBigErrors OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of datagrams that could not be forwarded
           because their size exceeded the Path MTU"
        ::= { ipv6IfStatsEntry 3 }

    ipv6IfStatsInAddrErrors OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of input datagrams discarded because
           the IPv6 address in their IPv6 header's destination
           field was not a valid address to be received at
           this entity.  This count includes invalid
           addresses (e.g., ::0) and unsupported addresses
           (e.g., addresses with unallocated prefixes).  For
           entities which are not IPv6 routers and therefore
           do not forward datagrams, this counter includes
           datagrams discarded because the destination address
           was not a local address."
        ::= { ipv6IfStatsEntry 4 }

    ipv6IfStatsForwDatagrams OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of input datagrams for which this
           entity was not their final IPv6 destination, as a
           result of which an attempt was made to find a
           route to forward them to that final destination.
           In entities which do not act as IPv6 routers, this
           counter will include only those packets which were
           Source-Routed via this entity, and the Source-
           Route option processing was successful."
        ::= { ipv6IfStatsEntry 5 }

    ipv6IfStatsInUnknownProtos OBJECT-TYPE
        SYNTAX      Counter32
```

```
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of locally-addressed datagrams
           received successfully but discarded because of an
           unknown or unsupported protocol."
        ::= { ipv6IfStatsEntry 6 }

    ipv6IfStatsInDiscards OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of input IPv6 datagrams for which no
           problems were encountered to prevent their
           continued processing, but which were discarded
           (e.g., for lack of buffer space).  Note that this
           counter does not include any datagrams discarded
           while awaiting re-assembly."
        ::= { ipv6IfStatsEntry 7 }

    ipv6IfStatsInDelivers OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
         "The total number of input datagrams successfully
         delivered to IPv6 user-protocols (including ICMP)."
        ::= { ipv6IfStatsEntry 8 }

    ipv6IfStatsOutRequests OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
         "The total number of IPv6 datagrams which local IPv6
         user-protocols (including ICMP) supplied to IPv6 in
         requests for transmission.  Note that this counter
         does not include any datagrams counted in
         ipv6IfStatsForwDatagrams."
        ::= { ipv6IfStatsEntry 9 }

    ipv6IfStatsOutDiscards OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
```

```
        STATUS      current
        DESCRIPTION
            "The number of output IPv6 datagrams for which no
            problem was encountered to prevent their
            transmission to their destination, but which were
            discarded (e.g., for lack of buffer space).  Note
            that this counter would include datagrams counted
            in ipv6IfStatsForwDatagrams if any such packets
            met this (discretionary) discard criterion."
        ::= { ipv6IfStatsEntry 10 }

    ipv6IfStatsOutNoRoutes OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of IPv6 datagrams discarded because no
            route could be found to transmit them to their
            destination.  Note that this counter includes any
            packets counted in ipv6IfStatsForwDatagrams which
            meet this `no-route' criterion.  Note that this
            includes any datagarms which a host cannot route
            because all of its default routers are down."
        ::= { ipv6IfStatsEntry 11 }

    ipv6IfStatsReasmReqds OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of IPv6 fragments received which needed
            to be reassembled at this entity."
        ::= { ipv6IfStatsEntry 12 }

    ipv6IfStatsReasmOKs OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of IPv6 datagrams successfully
          reassembled."
        ::= { ipv6IfStatsEntry 13 }

    ipv6IfStatsReasmFails OBJECT-TYPE
        SYNTAX      Counter32
```

```
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of failures detected by the IPv6 re-
           assembly algorithm (for whatever reason: timed
           out, errors, etc).  Note that this is not
           necessarily a count of discarded IPv6 fragments
           since some algorithms (notably the algorithm in
           RFC 815) can lose track of the number of fragments
           by combining them as they are received."
        ::= { ipv6IfStatsEntry 14 }

    ipv6IfStatsFragOKs OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of IPv6 datagrams that have been
            successfully fragmented at this entity."
        ::= { ipv6IfStatsEntry 15 }

    ipv6IfStatsFragFails OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of IPv6 datagrams that have been
            discarded because they needed to be fragmented
            at this entity but could not be."
        ::= { ipv6IfStatsEntry 16 }

    ipv6IfStatsFragCreates OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of IPv6 datagram fragments that have
            been generated as a result of fragmentation at
            this entity."
        ::= { ipv6IfStatsEntry 17 }

    ipv6IfStatsInMcastPkts OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS     current
```

```
        DESCRIPTION
           "The number of multicast packets received
            by the interface"
        ::= { ipv6IfStatsEntry 18 }

    ipv6IfStatsOutMcastPkts OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of multicast packets transmitted
            by the interface"
        ::= { ipv6IfStatsEntry 19 }

    ipv6IfStatsInTruncatedPkts OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "The number of input datagrams discarded because
            datagram frame didn't carry enough data"
        ::= { ipv6IfStatsEntry 20 }


    -- Address Prefix table

    -- The IPv6 Address Prefix table contains information on
    -- the entity's IPv6 Address Prefixes that are associated
    -- with IPv6 interfaces.

    ipv6AddrPrefixTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF Ipv6AddrPrefixEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "The list of IPv6 address prefixes of
            IPv6 interfaces."
        ::= { ipv6MIBObjects 28 }

    ipv6AddrPrefixEntry OBJECT-TYPE
        SYNTAX  Ipv6AddrPrefixEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
            "An interface entry containing objects of
```

```
            a particular IPv6 address prefix."
        INDEX   { ipv6IfIndex,
                  ipv6AddrPrefix,
                  ipv6AddrPrefixLength }
        ::= { ipv6AddrPrefixTable 1 }

    Ipv6AddrPrefixEntry ::= SEQUENCE {
         ipv6AddrPrefix                      Ipv6AddressPrefix,
         ipv6AddrPrefixLength                INTEGER (3..128),
         ipv6AddrPrefixOnLinkFlag            TruthValue,
         ipv6AddrPrefixAutonomousFlag        TruthValue,
         ipv6AddrPrefixAdvPreferredLifetime Unsigned32,
         ipv6AddrPrefixAdvValidLifetime      Unsigned32
        }

    ipv6AddrPrefix OBJECT-TYPE
        SYNTAX      Ipv6AddressPrefix
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "The prefix associated with the this interface."
        ::= { ipv6AddrPrefixEntry 1 }

    ipv6AddrPrefixLength OBJECT-TYPE
        SYNTAX      INTEGER (3..128)
        UNITS       "bits"
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "The length of the prefix (in bits). A prefix can
          not be shorter than 3 bits."
        ::= { ipv6AddrPrefixEntry 2 }

    ipv6AddrPrefixOnLinkFlag OBJECT-TYPE
        SYNTAX      TruthValue
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "This object has the value 'true(1)', if that this
          prefix can be used for on-link determination and
          the value 'false(2)' otherwise."
        ::= { ipv6AddrPrefixEntry 3 }

    ipv6AddrPrefixAutonomousFlag OBJECT-TYPE
        SYNTAX      TruthValue
```

```
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "Autonomous address configuration flag. When
          true(1), indicates that this prefix can be used
          for autonomous address configuration (i.e. can
          be used to form a local interface address).
          If false(2), it is not used to autoconfigure
          a local interface address."
        ::= { ipv6AddrPrefixEntry 4 }

    ipv6AddrPrefixAdvPreferredLifetime OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
           "It is the length of time in seconds that this
           prefix will remain preferred, i.e. time until
           deprecation.  A value of 4,294,967,295 represents
           infinity.

           The address generated from a deprecated prefix
           should no longer be used as a source address in
           new communications, but packets received on such
           an interface are processed as expected."
        ::= { ipv6AddrPrefixEntry 5 }

    ipv6AddrPrefixAdvValidLifetime OBJECT-TYPE
        SYNTAX      Unsigned32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "It is the length of time in seconds that this
          prefix will remain valid, i.e. time until
          invalidation.  A value of 4,294,967,295 represents
          infinity.

          The address generated from an invalidated prefix
          should not appear as the destination or source
          address of a packet."
        ::= { ipv6AddrPrefixEntry 6 }


    -- the IPv6 Address table
```

```
    -- The IPv6 address table contains this node's IPv6
    -- addressing information.

    ipv6AddrTable OBJECT-TYPE
       SYNTAX      SEQUENCE OF Ipv6AddrEntry
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
         "The table of addressing information relevant to
         this node's interface addresses."
       ::= { ipv6MIBObjects 29 }

    ipv6AddrEntry OBJECT-TYPE
       SYNTAX      Ipv6AddrEntry
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
           "The addressing information for one of this
           node's interface addresses."
       INDEX   { ipv6IfIndex, ipv6AddrAddress }
       ::= { ipv6AddrTable 1 }

    Ipv6AddrEntry ::=
       SEQUENCE {
            ipv6AddrAddress        Ipv6Address,
            ipv6AddrPfxLength      INTEGER,
            ipv6AddrType           INTEGER,
            ipv6AddrStatus         INTEGER
           }

    ipv6AddrAddress OBJECT-TYPE
       SYNTAX      Ipv6Address
       MAX-ACCESS  not-accessible
       STATUS      current
       DESCRIPTION
         "The IPv6 address to which this entry's addressing
         information pertains."
       ::= { ipv6AddrEntry 1 }

    ipv6AddrPfxLength OBJECT-TYPE
       SYNTAX      INTEGER(0..128)
       UNITS       "bits"
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
```

```
         "The length of the prefix (in bits) associated with
         the IPv6 address of this entry."
       ::= { ipv6AddrEntry 2 }

    ipv6AddrType OBJECT-TYPE
       SYNTAX      INTEGER {
                           -- address has been form
                           -- using stateless
            stateless(1),  -- autoconfiguration

                           -- address has been acquired
                           -- by stateful means
                           -- (e.g. DHCPv6, manual
            stateful(2),   -- configuration)

                           -- type can not be determined
            unknown(3)     -- for some reason.
          }
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
          "The type of address. Note that 'stateless(1)'
          refers to an address that was statelessly
          autoconfigured; 'stateful(2)' refers to a address
          which was acquired by via a stateful protocol
          (e.g. DHCPv6, manual configuration)."
       ::= { ipv6AddrEntry 3 }

    ipv6AddrStatus OBJECT-TYPE
       SYNTAX      INTEGER {
                preferred(1),

                deprecated(2),

                invalid(3),

                inaccessible(4),

                unknown(5)   -- status can not be determined
                             -- for some reason.
               }
       MAX-ACCESS  read-only
       STATUS      current
       DESCRIPTION
         "Address status.  The preferred(1) state indicates
```

```
         that this is a valid address that can appear as
         the destination or source address of a packet.
         The deprecated(2) this is a valid but deprecated
         address that should no longer be used as a source
         address in new communications, but packets addressed
         to such an address are processed as expected. The
         invalid(3) state indicates that this is not valid
         address which should not appear as the destination
         or source address of a packet. The inaccessible(4)
         state indicates the address is not accessible because
         the interface to which this address is assigned is
         not operational."
       ::= { ipv6AddrEntry 4 }


    -- IPv6 Routing objects

    ipv6InstalledRoutes OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of current ipv6RouteTable entries
           that are not invalid."
        ::= { ipv6MIBObjects 30 }

    ipv6DiscardedRoutes OBJECT-TYPE
        SYNTAX      Counter32
        MAX-ACCESS  read-only
        STATUS      current
        DESCRIPTION
          "The number of routing entries which were chosen
           to be discarded even though they are valid.  One
           possible reason for discarding such an entry could
           be to free-up buffer space for other routing
           entries."
        ::= { ipv6MIBObjects 31 }


    -- IPv6 Routing table

    -- The IPv6 routing table contains an entry for each
    -- valid IPv6 unicast route that can be used for
    -- packet forwarding determination.
```

```
    ipv6RouteTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF Ipv6RouteEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
                "IPv6 Routing table"
        ::= { ipv6MIBObjects 32 }

    ipv6RouteEntry OBJECT-TYPE
        SYNTAX     Ipv6RouteEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
                "A routing entry"
        INDEX   { ipv6RouteDest,
                  ipv6RoutePfxLength,
                  ipv6RouteIfIndex,
                  ipv6RouteIndex }
        ::= { ipv6RouteTable 1 }

    Ipv6RouteEntry ::= SEQUENCE {
            ipv6RouteDest           Ipv6Address,
            ipv6RoutePfxLength      INTEGER,
            ipv6RouteIfIndex        Ipv6IfIndex,
            ipv6RouteIndex          Integer32,
            ipv6RouteNextHop        Ipv6Address,
            ipv6RouteType           INTEGER,
            ipv6RouteProtocol       INTEGER,
            ipv6RoutePolicy         Unsigned32,
            ipv6RouteAge            Unsigned32,
            ipv6RouteNextHopRDI     OCTET STRING,
            ipv6RouteMetric         Unsigned32,
            ipv6RouteWeight         Unsigned32,
            ipv6RouteInfo           RowPointer,
            ipv6RouteValid          TruthValue
        }

    ipv6RouteDest OBJECT-TYPE
        SYNTAX     Ipv6Address
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The destination IPv6 address of this route.
             This object may not take a Multicast address
             value."
```

```
        ::= { ipv6RouteEntry 1 }

    ipv6RoutePfxLength OBJECT-TYPE
        SYNTAX     INTEGER(0..128)
        UNITS      "bits"
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
           "Indicates the prefix length of the destination
           address."
        ::= { ipv6RouteEntry 2 }

    ipv6RouteIfIndex OBJECT-TYPE
        SYNTAX     Ipv6IfIndex
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
         "The index value which uniquely identifies the local
          interface through which the next hop of this
          route should be reached.  The interface identified
          by a particular value of this index is the same
          interface as identified by the same value of
          ipv6IfIndex."
        ::= { ipv6RouteEntry 3 }

    ipv6RouteIndex OBJECT-TYPE
        SYNTAX     Integer32
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "The value which uniquely identifies the route
          among the routes to the same network layer
          destination.  The way this value is chosen is
          implementation specific but it must be unique for
          ipv6RouteDest/ipv6RoutePfxLength/ipv6RouteIfIndex
          tuple and remain constant for the life of the route."
        ::= { ipv6RouteEntry 4 }

    ipv6RouteNextHop OBJECT-TYPE
        SYNTAX     Ipv6Address
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "On remote routes, the address of the next system
          en route;  Otherwise, ::0"
```

```
        ::= { ipv6RouteEntry 5 }

    ipv6RouteType OBJECT-TYPE
        SYNTAX     INTEGER {
           other(1),     -- none of the following

           invalid(2),   -- an invalidated route

                         -- route to directly
           direct(3),    -- connected (sub-)network

                         -- route to a non-local
           indirect(4),  -- destination

                         -- an route indicating that
                         -- packets to destinations
                         -- matching this route are
           discard(5)    -- to be discarded
        }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The type of route. Note that 'direct(3)' refers
           to  a route for which the next hop is the final
           destination; 'indirect(4)' refers to a route for
           which  the  next  hop is not the final
           destination; 'discard(5)' refers to a route
           indicating that packets to destinations matching
           this route are to be discarded (sometimes called
           black-hole route)."
        ::= { ipv6RouteEntry 6 }

    ipv6RouteProtocol OBJECT-TYPE
        SYNTAX     INTEGER {
          other(1),   -- none of the following

                      -- non-protocol information,
                      -- e.g., manually configured
          local(2),   -- entries

          netmgmt(3), -- static route

                      -- obtained via Neighbor
                      -- Discovery protocol,
          ndisc(4),   -- e.g., Redirect
```

```
                      -- the following are all
                      -- dynamic routing protocols
            rip(5),       -- RIPng
            ospf(6),      -- Open Shortest Path First
            idrp(7)       -- InterDomain Routing Protocol
          }
          MAX-ACCESS read-only
          STATUS     current
          DESCRIPTION
            "The routing mechanism via which this route was
            learned."
          ::= { ipv6RouteEntry 7 }

      ipv6RoutePolicy OBJECT-TYPE
          SYNTAX     Unsigned32
          MAX-ACCESS read-only
          STATUS     current
          DESCRIPTION
           "The general set of conditions that would cause the
           selection of one multipath route (set of next hops
           for a given destination) is referred to as 'policy'.
           Unless the mechanism indicated by ipv6RouteProtocol
           specified otherwise, the policy specifier is the
           Priority field of the IPv6 packet header.  The
           encoding of IPv6 Priority is specified by
           the following convention:

             0 - uncharacterized traffic
             1 - 'filler' traffic (e.g., netnews)
             2 - unattended data transfer (e.g., email)
             3 - reserved
             4 - attended bulk transfer (e.g., FTP, NFS)
             5 - reserved
             6 - interactive traffic (e.g., telnet, X)
             7 - internet control traffic (e.g., routing
                 protocols, SNMP)

           Protocols defining 'policy' otherwise must either
           define a set of values which are valid for
           this  object  or  must  implement  an  integer-
           instanced  policy table for which this object's
           value acts as an index."
          ::= { ipv6RouteEntry 8 }

      ipv6RouteAge OBJECT-TYPE
```

```
        SYNTAX     Unsigned32
        UNITS      "seconds"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The number of seconds since this route was last
           updated or otherwise determined to be correct.
           Note that no semantics of `too old' can be implied
           except through knowledge of the routing protocol
           by which the route was learned."
        ::= { ipv6RouteEntry 9 }

    ipv6RouteNextHopRDI OBJECT-TYPE
        SYNTAX     OCTET STRING (SIZE (0 | 16))
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The Routing Domain ID of the Next Hop.
           When  this  is  unknown  or not relevant to the
           protocol indicated by ipv6RouteProtocol, it is
           an octet string of zero size."
        ::= { ipv6RouteEntry 10 }

    ipv6RouteMetric OBJECT-TYPE
        SYNTAX     Unsigned32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The routing metric for this route. The
           semantics of this metric are determined by the
           routing protocol specified in the route's
           ipv6RouteProtocol value.  When this is unknown
           or not relevant to the protocol indicated by
           ipv6RouteProtocol, the object value should be
           set to its maximum value (4,294,967,295)."
        ::= { ipv6RouteEntry 11 }

    ipv6RouteWeight OBJECT-TYPE
        SYNTAX     Unsigned32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
           "The system internal weight value for this route.
           The semantics of this value are determined by
           the implementation specific rules. Generally,
```

```
              within routes with the same ipv6RoutePolicy value,
              the lower the weight value the more preferred is
              the route."
           ::= { ipv6RouteEntry 12 }

       ipv6RouteInfo OBJECT-TYPE
           SYNTAX     RowPointer
           MAX-ACCESS read-only
           STATUS     current
           DESCRIPTION
              "A reference to MIB definitions specific to the
              particular routing protocol which is responsible
              for this route, as determined by the  value
              specified  in the route's ipv6RouteProto value.
              If this information is not present,  its  value
              should be set to the OBJECT ID { 0 0 },
              which is a syntactically valid object  identifier,
              and any implementation conforming to ASN.1
              and the Basic Encoding Rules must  be  able  to
              generate and recognize this value."
           ::= { ipv6RouteEntry 13 }

       ipv6RouteValid OBJECT-TYPE
           SYNTAX     TruthValue
           MAX-ACCESS read-write
           STATUS     current
           DESCRIPTION
              "Setting this object to the value 'false(2)' has
              the effect of invalidating the corresponding entry
              in the ipv6RouteTable object.  That is, it
              effectively disassociates the destination
              identified with said entry from the route
              identified with said entry.  It is an
              implementation-specific matter as to whether the
              agent removes an invalidated entry from the table.
              Accordingly, management stations must be prepared
              to receive tabular information from agents that
              corresponds to entries not currently in use.
              Proper interpretation of such entries requires
              examination of the relevant ipv6RouteInvalid
              object."
           DEFVAL  { true }
           ::= { ipv6RouteEntry 14 }
```

```
    -- IPv6 Address Translation table

    -- The IPv6 address translation table contain the
    -- Ipv6Address to `physical' address equivalences.
    -- Some interfaces do not use translation tables
    -- for determining address equivalences; if all
    -- interfaces are of this type, then the Address
    -- Translation table is empty, i.e., has zero
    -- entries.

    ipv6NetToMediaEntryTable OBJECT-TYPE
        SYNTAX      SEQUENCE OF Ipv6NetToMediaEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
          "The IPv6 Address Translation table used for
          mapping from IPv6 addresses to physical addresses."
        ::= { ipv6MIBObjects 33 }

    ipv6NetToMediaEntry OBJECT-TYPE
        SYNTAX     Ipv6NetToMediaEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "Each entry contains one IPv6 address to `physical'
          address equivalence."
        INDEX   { ipv6IfIndex,
                  ipv6NetToMediaNetAddress }
        ::= { ipv6NetToMediaEntryTable 1 }

    Ipv6NetToMediaEntry ::= SEQUENCE {
            ipv6NetToMediaNetAddress
                Ipv6Address,
            ipv6NetToMediaPhysAddress
                PhysAddress,
            ipv6NetToMediaType
                INTEGER,
            ipv6NetToMediaValid
                TruthValue
        }

    ipv6NetToMediaNetAddress OBJECT-TYPE
        SYNTAX     Ipv6Address
        MAX-ACCESS not-accessible
        STATUS     current
```

```
        DESCRIPTION
           "The IPv6 Address corresponding to
           the media-dependent `physical' address."
        ::= { ipv6NetToMediaEntry 1 }

    ipv6NetToMediaPhysAddress OBJECT-TYPE
        SYNTAX     PhysAddress
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The media-dependent `physical' address."
        ::= { ipv6NetToMediaEntry 2 }

    ipv6NetToMediaType OBJECT-TYPE
        SYNTAX     INTEGER {
                    other(1),    -- none of the following
                    dynamic(2),  -- dynamically resolved
                    static(3)    -- statically configured
                   }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
                "The type of mapping."
        ::= { ipv6NetToMediaEntry 3 }

    ipv6NetToMediaValid OBJECT-TYPE
        SYNTAX     TruthValue
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
         "Setting this object to the value 'false(2)' has
         the effect of invalidating the corresponding entry
         in the ipv6NetToMediaTable.  That is, it effectively
         disassociates the interface identified with said
         entry from the mapping identified with said entry.
         It is an implementation-specific matter as to
         whether the agent removes an invalidated entry
         from the table.  Accordingly, management stations
         must be prepared to receive tabular information
         from agents that corresponds to entries not
         currently in use.  Proper interpretation of such
         entries requires examination of the relevant
         ipv6NetToMediaInvalid object."
        DEFVAL  { true }
        ::= { ipv6NetToMediaEntry 4 }
```

```
   -- definition of IPv6-related traps.

    ipv6Traps            OBJECT IDENTIFIER   ::= { ipv6MIB 2 }

    ipv6IfStateChange NOTIFICATION-TYPE
        OBJECTS {
                   ipv6IfDescr,
                   ipv6IfOperStatus -- the new state of the If.
                }
        STATUS             current
        DESCRIPTION
           "An ipv6IfStateChange trap signifies that there
           has been a change in the state of an ipv6 interface.
           This trap should be generated when  the interface's
           operational status transitions to or from
           the down(2) state."
    ::= { ipv6Traps 1 }


   -- conformance information

   ipv6Conformance OBJECT IDENTIFIER ::= { ipv6MIB 3 }

   ipv6Compliances OBJECT IDENTIFIER ::= { ipv6Conformance 1 }
   ipv6Groups      OBJECT IDENTIFIER ::= { ipv6Conformance 2 }

   -- compliance statements

   ipMIBCompliance MODULE-COMPLIANCE
       STATUS  current
       DESCRIPTION
         "The compliance statement for SNMPv2 entities which
         implement ipv6 MIB."
       MODULE  -- this module
           MANDATORY-GROUPS { ipv6GeneralGroup }
       ::= { ipv6Compliances 1 }

   ipv6GeneralGroup OBJECT-GROUP
       OBJECTS { ipv6Forwarding,
                 ipv6DefaultHopLimit,
                 ipv6InReceives,
                 ipv6InHdrErrors,
                 ipv6TooBigErrors,
                 ipv6InAddrErrors,
                 ipv6TruncatedPkts,
```

```
                 ipv6ForwDatagrams,
                 ipv6InUnknownProtos,
                 ipv6InDiscards,
                 ipv6InDelivers,
                 ipv6OutRequests,
                 ipv6OutDiscards,
                 ipv6OutNoRoutes,
                 ipv6ReasmTimeout,
                 ipv6ReasmReqds,
                 ipv6ReasmOKs,
                 ipv6ReasmFails,
                 ipv6FragOKs,
                 ipv6FragFails,
                 ipv6FragCreates,
                 ipv6InMcastPkts,
                 ipv6OutMcastPkts,
                 ipv6Interfaces,
                 ipv6IfTableLastChange,
                 ipv6IfDescr,
                 ipv6IfLowerLayer,
                 ipv6IfEffectiveMtu,
                 ipv6IfReasmMaxSize,
                 ipv6IfToken,
                 ipv6IfTokenLength,
                 ipv6IfPhysicalAddress,
                 ipv6IfAdminStatus,
                 ipv6IfOperStatus,
                 ipv6IfLastChange,
                 ipv6IfStatsInReceives,
                 ipv6IfStatsInHdrErrors,
                 ipv6IfStatsTooBigErrors,
                 ipv6IfStatsInAddrErrors,
                 ipv6IfStatsForwDatagrams,
                 ipv6IfStatsInUnknownProtos,
                 ipv6IfStatsInDiscards,
                 ipv6IfStatsInDelivers,
                 ipv6IfStatsOutRequests,
                 ipv6IfStatsOutDiscards,
                 ipv6IfStatsOutNoRoutes,
                 ipv6IfStatsReasmReqds,
                 ipv6IfStatsReasmOKs,
                 ipv6IfStatsReasmFails,
                 ipv6IfStatsFragOKs,
                 ipv6IfStatsFragFails,
                 ipv6IfStatsFragCreates,
```

```
                 ipv6IfStatsInMcastPkts,
                 ipv6IfStatsOutMcastPkts,
                 ipv6IfStatsInTruncatedPkts,
                 ipv6AddrPrefixOnLinkFlag,
                 ipv6AddrPrefixAutonomousFlag,
                 ipv6AddrPrefixAdvPreferredLifetime,
                 ipv6AddrPrefixAdvValidLifetime,
                 ipv6AddrPfxLength,
                 ipv6AddrType,
                 ipv6AddrStatus,
                 ipv6InstalledRoutes,
                 ipv6DiscardedRoutes,
                 ipv6RouteNextHop,
                 ipv6RouteType,
                 ipv6RouteProtocol,
                 ipv6RoutePolicy,
                 ipv6RouteAge,
                 ipv6RouteNextHopRDI,
                 ipv6RouteMetric,
                 ipv6RouteWeight,
                 ipv6RouteInfo,
                 ipv6RouteValid,
                 ipv6NetToMediaPhysAddress,
                 ipv6NetToMediaType,
                 ipv6NetToMediaValid }
       STATUS    current
       DESCRIPTION
            "The IPv6 group of objects providing for basic
             management of IPv6 entities"
       ::= { ipv6Groups 1 }

    END
```

5.2. The ICMPv6 Group

```
    IPV6-ICMP-MIB DEFINITIONS ::= BEGIN

    IMPORTS
        MODULE-IDENTITY, OBJECT-TYPE,
        Counter32                        FROM SNMPv2-SMI
        MODULE-COMPLIANCE, OBJECT-GROUP  FROM SNMPv2-CONF
        ipv6                             FROM IPV6-TC
        ipv6IfEntry                      FROM IPV6-MIB;

    ipv6IcmpMIB MODULE-IDENTITY
        LAST-UPDATED "9610042155Z"
        ORGANIZATION "IETF IPv6 MIB Working Group"
        CONTACT-INFO
          "           Dimitry Haskin

              Postal: Bay Networks, Inc.
                      2 Federal St.
                      Billerica, MA  01821
                      US

                 Tel: +1-508-916-8124
              E-mail: dhaskin@baynetworks.com

                      Steve Onishi

              Postal: Bay Networks, Inc.
                      3 Federal Street
                      Billerica, MA 01821
                      US

                 Tel: +1-508-916-3816
              E-mail: sonishi@baynetworks.com"
        DESCRIPTION
          "The MIB module for entities implementing
           the ICMPv6."
        ::= { ipv6  2 }


    -- the ICMPv6 group

    ipv6IcmpMIBObjects OBJECT IDENTIFIER ::= { ipv6IcmpMIB  1 }
```

```
    ipv6IcmpInMsgs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The total number of ICMP messages which
            the entity received.  Note that this counter
            includes all those counted by ipv6IcmpInErrors."
        ::= { ipv6IcmpMIBObjects 1 }

    ipv6IcmpInErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP messages which the entity
         received but determined as having ICMP-specific
         errors (bad ICMP checksums, bad length, etc.)."
        ::= { ipv6IcmpMIBObjects 2 }

    ipv6IcmpInDestUnreachs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Destination Unreachable
         messages received."
        ::= { ipv6IcmpMIBObjects 3 }

    ipv6IcmpInTimeExcds OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Time Exceeded messages
          received."
        ::= { ipv6IcmpMIBObjects 4 }

    ipv6IcmpInParmProbs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Parameter Problem messages
          received."
```

```
        ::= { ipv6IcmpMIBObjects 5 }

    ipv6IcmpInPktTooBigs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Packet Too Big messages
         received."
        ::= { ipv6IcmpMIBObjects 6 }

    ipv6IcmpInEchos OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo (request) messages
          received."
        ::= { ipv6IcmpMIBObjects 7 }

    ipv6IcmpInEchoReps OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo Reply messages received."
        ::= { ipv6IcmpMIBObjects 8 }

    ipv6IcmpInRouterSolicits OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Router Solicit messages
          received."
        ::= { ipv6IcmpMIBObjects 9 }

    ipv6IcmpInRouterAdvertisements OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Router Advertisement messages
         received."
        ::= { ipv6IcmpMIBObjects 10 }
```

```
    ipv6IcmpInNeighborSolicits OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Neighbor Solicit messages
          received."
        ::= { ipv6IcmpMIBObjects 11 }

    ipv6IcmpInNeighborAdvertisements OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Neighbor Advertisement
         messages received."
        ::= { ipv6IcmpMIBObjects 12 }

    ipv6IcmpInRedirects OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of Redirect messages received."
        ::= { ipv6IcmpMIBObjects 13 }

    ipv6IcmpInAdminProhib OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "Number of ICMP destination unreachable/communication
         administratively prohibited messages received."
        ::= { ipv6IcmpMIBObjects 14 }

    ipv6IcmpOutMsgs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The total number of ICMP messages which this
         entity attempted to send.  Note that this counter
         includes all those counted by icmpOutErrors."
        ::= { ipv6IcmpMIBObjects 15 }
```

```
    ipv6IcmpOutErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP messages which this entity did
         not send due to problems discovered within ICMP
         such as a lack of buffers.  This value should not
         include errors discovered outside the ICMP layer
         such as the inability of IPv6 to route the resultant
         datagram.  In some implementations there may be no
         types of error which contribute to this counter's
         value."
        ::= { ipv6IcmpMIBObjects 16 }

    ipv6IcmpOutDestUnreachs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Destination Unreachable
         messages sent."
        ::= { ipv6IcmpMIBObjects 17 }

    ipv6IcmpOutTimeExcds OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Time Exceeded messages sent."
        ::= { ipv6IcmpMIBObjects 18 }

    ipv6IcmpOutParmProbs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Parameter Problem messages
         sent."
        ::= { ipv6IcmpMIBObjects 19 }

    ipv6IcmpOutPktTooBigs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
```

```
        DESCRIPTION
         "The number of ICMP Packet Too Big messages sent."
        ::= { ipv6IcmpMIBObjects 20 }

    ipv6IcmpOutEchos OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo (request) messages sent."
        ::= { ipv6IcmpMIBObjects 21 }

    ipv6IcmpOutEchoReps OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo Reply messages sent."
        ::= { ipv6IcmpMIBObjects 22 }

    ipv6IcmpOutRouterSolicits OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Router Solicitation messages
          sent."
        ::= { ipv6IcmpMIBObjects 23 }

    ipv6IcmpOutRouterAdvertisements OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Router Advertisement messages
         received."
        ::= { ipv6IcmpMIBObjects 24 }

    ipv6IcmpOutNeighborSolicits OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Neighbor Solicitation
          messages sent."
```

```
        ::= { ipv6IcmpMIBObjects 25 }

    ipv6IcmpOutNeighborAdvertisements OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Neighbor Advertisement
         messages
         received."
        ::= { ipv6IcmpMIBObjects 26 }

    ipv6IcmpOutRedirects OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of Redirect messages sent. For
         a host, this object will always be zero,
         since hosts do not send redirects."
        ::= { ipv6IcmpMIBObjects 27 }

    ipv6IcmpOutAdminProhib OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "Number of ICMP destination unreachable/
          communication administratively prohibited
          messages sent."
        ::= { ipv6IcmpMIBObjects 28 }

    -- Per-interface ICMPv6 statistics table

    ipv6IfIcmpTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF Ipv6IfIcmpEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
                "IPv6 ICMP statistics"
        ::= { ipv6IcmpMIBObjects 29 }

    ipv6IfIcmpEntry OBJECT-TYPE
        SYNTAX      Ipv6IfIcmpEntry
        MAX-ACCESS  not-accessible
```

```
        STATUS      current
        DESCRIPTION
            "An ICMPv6 statistics entry containing
            objects at a particular IPv6 interface."
        AUGMENTS { ipv6IfEntry }
        ::= { ipv6IfIcmpTable 1 }

    Ipv6IfIcmpEntry ::= SEQUENCE {
            ipv6IfIcmpInMsgs
                  Counter32      ,
            ipv6IfIcmpInErrors
                  Counter32      ,
            ipv6IfIcmpInDestUnreachs
                  Counter32      ,
            ipv6IfIcmpInTimeExcds
                  Counter32      ,
            ipv6IfIcmpInParmProbs
                  Counter32      ,
            ipv6IfIcmpInPktTooBigs
                  Counter32      ,
            ipv6IfIcmpInEchos
                  Counter32      ,
            ipv6IfIcmpInEchoReps
                  Counter32      ,
            ipv6IfIcmpInRouterSolicits
                  Counter32      ,
            ipv6IfIcmpInRouterAdvertisements
                  Counter32      ,
            ipv6IfIcmpInNeighborSolicits
                  Counter32      ,
            ipv6IfIcmpInNeighborAdvertisements
                  Counter32      ,
            ipv6IfIcmpInRedirects
                  Counter32      ,
            ipv6IfIcmpOutMsgs
                  Counter32      ,
            ipv6IfIcmpOutErrors
                  Counter32      ,
            ipv6IfIcmpOutDestUnreachs
                  Counter32      ,
            ipv6IfIcmpOutTimeExcds
                  Counter32      ,
            ipv6IfIcmpOutParmProbs
                  Counter32      ,
            ipv6IfIcmpOutPktTooBigs
```

```
                  Counter32      ,
            ipv6IfIcmpOutEchos
                  Counter32      ,
            ipv6IfIcmpOutEchoReps
                  Counter32      ,
            ipv6IfIcmpOutRouterSolicits
                  Counter32      ,
            ipv6IfIcmpOutRouterAdvertisements
                  Counter32      ,
            ipv6IfIcmpOutNeighborSolicits
                  Counter32      ,
            ipv6IfIcmpOutNeighborAdvertisements
                  Counter32      ,
            ipv6IfIcmpOutRedirects
                  Counter32      ,
            ipv6IfIcmpInAdminProhib
                  Counter32      ,
            ipv6IfIcmpOutAdminProhib
                  Counter32
        }

    ipv6IfIcmpInMsgs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The total number of ICMP messages which includes
         all those counted by ipv6IfIcmpInErrors."
        ::= { ipv6IfIcmpEntry 1 }

    ipv6IfIcmpInErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP messages which the entity
         received but determined as having ICMP-specific
         errors (bad ICMP checksums, bad length, etc.)."
        ::= { ipv6IfIcmpEntry 2 }

    ipv6IfIcmpInDestUnreachs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
```

```
         "The number of ICMP Destination Unreachable
         messages received."
        ::= { ipv6IfIcmpEntry 3 }

    ipv6IfIcmpInTimeExcds OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Time Exceeded messages
          received."
        ::= { ipv6IfIcmpEntry 4 }

    ipv6IfIcmpInParmProbs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Parameter Problem messages
          received."
        ::= { ipv6IfIcmpEntry 5 }

    ipv6IfIcmpInPktTooBigs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Packet Too Big messages
         received."
        ::= { ipv6IfIcmpEntry 6 }

    ipv6IfIcmpInEchos OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo (request) messages
          received."
        ::= { ipv6IfIcmpEntry 7 }

    ipv6IfIcmpInEchoReps OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
```

```
          "The number of ICMP Echo Reply messages received."
         ::= { ipv6IfIcmpEntry 8 }

     ipv6IfIcmpInRouterSolicits OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
          "The number of ICMP Router Solicit messages
           received."
         ::= { ipv6IfIcmpEntry 9 }

     ipv6IfIcmpInRouterAdvertisements OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
          "The number of ICMP Router Advertisement messages
          received."
         ::= { ipv6IfIcmpEntry 10 }

     ipv6IfIcmpInNeighborSolicits OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
          "The number of ICMP Neighbor Solicit messages
           received."
         ::= { ipv6IfIcmpEntry 11 }

     ipv6IfIcmpInNeighborAdvertisements OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
          "The number of ICMP Neighbor Advertisement
          messages received."
         ::= { ipv6IfIcmpEntry 12 }

     ipv6IfIcmpInRedirects OBJECT-TYPE
         SYNTAX     Counter32
         MAX-ACCESS read-only
         STATUS     current
         DESCRIPTION
          "The number of Redirect messages received."
```

```
        ::= { ipv6IfIcmpEntry 13 }

    ipv6IfIcmpInAdminProhib OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "Number of ICMP dest unreachable/communication
         administratively prohibited messages received."
        ::= { ipv6IfIcmpEntry 14 }

    ipv6IfIcmpOutMsgs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The total number of ICMP messages which this
         entity attempted to send.  Note that this counter
         includes all those counted by icmpOutErrors."
        ::= { ipv6IfIcmpEntry 15 }

    ipv6IfIcmpOutErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP messages which this entity did
         not send due to problems discovered within ICMP
         such as a lack of buffers.  This value should not
         include errors discovered outside the ICMP layer
         such as the inability of IPv6 to route the resultant
         datagram.  In some implementations there may be no
         types of error which contribute to this counter's
         value."
        ::= { ipv6IfIcmpEntry 16 }

    ipv6IfIcmpOutDestUnreachs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Destination Unreachable
         messages sent."
        ::= { ipv6IfIcmpEntry 17 }
```

```
    ipv6IfIcmpOutTimeExcds OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Time Exceeded messages sent."
        ::= { ipv6IfIcmpEntry 18 }

    ipv6IfIcmpOutParmProbs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Parameter Problem messages
         sent."
        ::= { ipv6IfIcmpEntry 19 }

    ipv6IfIcmpOutPktTooBigs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Packet Too Big messages sent."
        ::= { ipv6IfIcmpEntry 20 }

    ipv6IfIcmpOutEchos OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo (request) messages sent."
        ::= { ipv6IfIcmpEntry 21 }

    ipv6IfIcmpOutEchoReps OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Echo Reply messages sent."
        ::= { ipv6IfIcmpEntry 22 }

    ipv6IfIcmpOutRouterSolicits OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
```

```
        DESCRIPTION
         "The number of ICMP Router Solicitation messages
          sent."
        ::= { ipv6IfIcmpEntry 23 }

    ipv6IfIcmpOutRouterAdvertisements OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Router Advertisement messages
         received."
        ::= { ipv6IfIcmpEntry 24 }

    ipv6IfIcmpOutNeighborSolicits OBJECT-TYPE
        SYNTAX    Counter32
        MAX-ACCESS  read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Neighbor Solicitation
          messages sent."
        ::= { ipv6IfIcmpEntry 25 }

    ipv6IfIcmpOutNeighborAdvertisements OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of ICMP Neighbor Advertisement
         messages
         received."
        ::= { ipv6IfIcmpEntry 26 }

    ipv6IfIcmpOutRedirects OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of Redirect messages sent. For
         a host, this object will always be zero,
         since hosts do not send redirects."
        ::= { ipv6IfIcmpEntry 27 }

    ipv6IfIcmpOutAdminProhib OBJECT-TYPE
        SYNTAX     Counter32
```

```
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "Number of ICMP dest unreachable/communication
          administratively prohibited messages sent."
        ::= { ipv6IfIcmpEntry 28 }


   -- conformance information

   ipv6IcmpConformance OBJECT IDENTIFIER ::= { ipv6IcmpMIB 2 }

   ipv6IcmpCompliances
           OBJECT IDENTIFIER ::= { ipv6IcmpConformance 1 }
   ipv6IcmpGroups
           OBJECT IDENTIFIER ::= { ipv6IcmpConformance 2 }

   -- compliance statements

   ipv6IcmpCompliance MODULE-COMPLIANCE
       STATUS  current
       DESCRIPTION
         "The compliance statement for SNMPv2 entities which
         implement ICMPv6."
       MODULE  -- this module
           MANDATORY-GROUPS { ipv6IcmpGroup }
       ::= { ipv6IcmpCompliances 1 }

   ipv6IcmpGroup OBJECT-GROUP
       OBJECTS   {
                   ipv6IcmpInMsgs,
                   ipv6IcmpInErrors,
                   ipv6IcmpInDestUnreachs,
                   ipv6IcmpInTimeExcds,
                   ipv6IcmpInParmProbs ,
                   ipv6IcmpInEchoReps ,
                   ipv6IcmpInRouterSolicits,
                   ipv6IcmpInRouterAdvertisements,
                   ipv6IcmpInNeighborSolicits,
                   ipv6IcmpInNeighborAdvertisements,
                   ipv6IcmpInRedirects,
                   ipv6IcmpInPktTooBigs,
                   ipv6IcmpInEchos,
                   ipv6IcmpOutMsgs,
                   ipv6IcmpOutErrors,
```

```
                   ipv6IcmpOutDestUnreachs,
                   ipv6IcmpOutTimeExcds,
                   ipv6IcmpOutParmProbs,
                   ipv6IcmpOutPktTooBigs ,
                   ipv6IcmpOutEchos,
                   ipv6IcmpOutEchoReps,
                   ipv6IcmpOutRouterSolicits,
                   ipv6IcmpOutRouterAdvertisements,
                   ipv6IcmpOutNeighborSolicits,
                   ipv6IcmpOutNeighborAdvertisements,
                   ipv6IcmpOutRedirects,
                   ipv6IcmpInAdminProhib,
                   ipv6IcmpOutAdminProhib,
                   ipv6IfIcmpInMsgs,
                   ipv6IfIcmpInErrors,
                   ipv6IfIcmpInDestUnreachs,
                   ipv6IfIcmpInTimeExcds,
                   ipv6IfIcmpInParmProbs ,
                   ipv6IfIcmpInEchoReps ,
                   ipv6IfIcmpInRouterSolicits,
                   ipv6IfIcmpInRouterAdvertisements,
                   ipv6IfIcmpInNeighborSolicits,
                   ipv6IfIcmpInNeighborAdvertisements,
                   ipv6IfIcmpInRedirects,
                   ipv6IfIcmpOutMsgs,
                   ipv6IfIcmpOutErrors,
                   ipv6IfIcmpOutDestUnreachs,
                   ipv6IfIcmpOutTimeExcds,
                   ipv6IfIcmpOutParmProbs,
                   ipv6IfIcmpOutPktTooBigs ,
                   ipv6IfIcmpOutEchos,
                   ipv6IfIcmpOutEchoReps,
                   ipv6IfIcmpOutRouterSolicits,
                   ipv6IfIcmpOutRouterAdvertisements,
                   ipv6IfIcmpOutNeighborSolicits,
                   ipv6IfIcmpOutNeighborAdvertisements,
                   ipv6IfIcmpOutRedirects,
                   ipv6IfIcmpInAdminProhib,
                   ipv6IfIcmpOutAdminProhib,
                   ipv6IfIcmpInPktTooBigs,
                   ipv6IfIcmpInEchos
                 }
       STATUS    current
       DESCRIPTION
            "The ICMPv6 group of objects providing information
```

```
                 specific to ICMPv6."
           ::= { ipv6IcmpGroups 1 }

       END
```

5.3. The UDP Group

```
    IPV6-UDP-MIB DEFINITIONS ::= BEGIN

    IMPORTS
        MODULE-IDENTITY, OBJECT-TYPE,
        Counter32, Unsigned32            FROM SNMPv2-SMI
        ipv6, Ipv6Address                FROM IPV6-TC
        MODULE-COMPLIANCE, OBJECT-GROUP  FROM SNMPv2-CONF
        ipv6IfIndex                      FROM IPV6-MIB;


    ipv6UdpMIB MODULE-IDENTITY
        LAST-UPDATED "9610042155Z"
        ORGANIZATION "IETF IPv6 MIB Working Group"
        CONTACT-INFO
          "           Dimitry Haskin
              Postal: Bay Networks, Inc.
                      2 Federal St.
                      Billerica, MA  01821
                      US

                 Tel: +1-508-916-8124
              E-mail: dhaskin@baynetworks.com

                      Steve Onishi
              Postal: Bay Networks, Inc.
                      3 Federal Street
                      Billerica, MA 01821
                      US

                 Tel: +1-508-916-3816
              E-mail: sonishi@baynetworks.com"
        DESCRIPTION
          "The MIB module for entities implementing UDP
           over IPv6."
        ::= { ipv6  3}


     -- the UDP group

    ipv6UdpMIBObjects    OBJECT IDENTIFIER ::= { ipv6UdpMIB 1 }
```

```
    ipv6UdpInDatagrams OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of UDP datagrams delivered to
           UDP users."
        ::= { ipv6UdpMIBObjects 1 }

    ipv6UdpNoPorts OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of received UDP datagrams
           for which there was no application at
           the destination port."
        ::= { ipv6UdpMIBObjects 2 }

    ipv6UdpInErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The number of received UDP datagrams that
           could not be delivered for reasons other
           than the lack of an application at
           the destination port."
        ::= { ipv6UdpMIBObjects 3 }

    ipv6UdpOutDatagrams OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The total number of UDP datagrams sent
           from this entity."
        ::= { ipv6UdpMIBObjects 4 }

    ipv6UdpTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF Ipv6UdpEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "The UDP listener table contains information
```

```
           about this entity's UDP end-points on which
           a local application is currently accepting
           datagrams."
        ::= { ipv6UdpMIBObjects 5 }

    ipv6UdpEntry OBJECT-TYPE
        SYNTAX     Ipv6UdpEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "Information about a particular current UDP
           listener."
        INDEX   { ipv6IfIndex,
                  ipv6UdpLocalAddress,
                  ipv6UdpLocalPort }
        ::= { ipv6UdpTable 1 }

    Ipv6UdpEntry ::= SEQUENCE {
          ipv6UdpLocalAddress    Ipv6Address,
          ipv6UdpLocalPort       Unsigned32
        }

    ipv6UdpLocalAddress OBJECT-TYPE
        SYNTAX     Ipv6Address
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
          "The local IPv6 address for this UDP listener.
           In the case of a UDP listener which is willing
           to accept datagrams for any IPv6 address
           associated with the interface, the value ::0
           is used."
        ::= { ipv6UdpEntry 2 }

    ipv6UdpLocalPort OBJECT-TYPE
        SYNTAX     Unsigned32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The local port number for this UDP listener."
        ::= { ipv6UdpEntry 3 }


   -- conformance information
```

```
   ipv6UdpConformance OBJECT IDENTIFIER ::= { ipv6UdpMIB 2 }

   ipv6UdpCompliances OBJECT IDENTIFIER ::= { ipv6UdpConformance 1 }
   ipv6UdpGroups      OBJECT IDENTIFIER ::= { ipv6UdpConformance 2 }

   -- compliance statements

   ipv6UdpCompliance MODULE-COMPLIANCE
       STATUS  current
       DESCRIPTION
         "The compliance statement for SNMPv2 entities which
         implement UDP over IPv6."
       MODULE  -- this module
           GROUP       ipv6UdpGroup
           DESCRIPTION
               "This group is mandatory for all entities
               which implement UDP over IPv6."
       ::= { ipv6UdpCompliances 1 }

   ipv6UdpGroup OBJECT-GROUP
       OBJECTS   {
                   ipv6UdpInDatagrams,
                   ipv6UdpNoPorts,
                   ipv6UdpInErrors,
                   ipv6UdpOutDatagrams,
                   ipv6UdpLocalPort
                 }
       STATUS    current
       DESCRIPTION
            "The UDP group of objects providing information
             specific to UDP over IPv6."
       ::= { ipv6UdpGroups 1 }

   END
```

5.4. The TCP Group

```
   IPV6-TCP-MIB DEFINITIONS ::= BEGIN

   IMPORTS
        MODULE-IDENTITY, OBJECT-TYPE,
        Unsigned32, Gauge32, Counter32,
        Integer32                         FROM SNMPv2-SMI
        ipv6, Ipv6Address                 FROM IPV6-TC
        MODULE-COMPLIANCE, OBJECT-GROUP   FROM SNMPv2-CONF
        ipv6IfIndex                       FROM IPV6-MIB;


   ipv6TcpMIB MODULE-IDENTITY
        LAST-UPDATED "9610042155Z"
        ORGANIZATION "IETF IPv6 MIB Working Group"
        CONTACT-INFO
          "           Dimitry Haskin
              Postal: Bay Networks, Inc.
                      2 Federal St.
                      Billerica, MA  01821
                      US

                 Tel: +1-508-916-8124
              E-mail: dhaskin@baynetworks.com

                      Steve Onishi
              Postal: Bay Networks, Inc.
                      3 Federal Street
                      Billerica, MA 01821
                      US

                 Tel: +1-508-916-3816
              E-mail: sonishi@baynetworks.com"
        DESCRIPTION
          "The MIB module for entities implementing TCP
           over IPv6."
        ::= { ipv6  4}


     -- the TCPv6 group

     ipv6TcpMIBObjects OBJECT IDENTIFIER ::= { ipv6TcpMIB 1 }
```

```
     -- Note that instances of object types that represent
     -- information about a particular TCP connection are
     -- transient; they persist only as long as the connection
     -- in question.

     ipv6TcpRtoAlgorithm OBJECT-TYPE
        SYNTAX  INTEGER {
            other(1),    -- none of the following

            constant(2), -- a constant rto
            rsre(3),     -- MIL-STD-1778, Appendix B
            vanj(4)      -- Van Jacobson's algorithm [8]
                      }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
          "The algorithm used to determine the timeout value
           used for retransmitting unacknowledged octets."
        ::= { ipv6TcpMIBObjects 1 }

     ipv6TcpRtoMin OBJECT-TYPE
        SYNTAX     Unsigned32
        UNITS      "milliseconds"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The minimum value permitted by a TCP
         implementation for the retransmission timeout,
         measured in milliseconds.  More refined semantics
         for objects of this type depend upon the algorithm
         used to determine the retransmission timeout.  In
         particular, when the timeout algorithm is rsre(3),
         an object of this type has the semantics of the
         LBOUND quantity described in RFC 793."
        ::= { ipv6TcpMIBObjects 2 }

     ipv6TcpRtoMax OBJECT-TYPE
        SYNTAX     Unsigned32
        UNITS      "milliseconds"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The maximum value permitted by a TCP
         implementation for the retransmission timeout,
         measured in milliseconds.  More refined semantics
```

```
                for objects of this type depend upon the algorithm
                used to determine the retransmission timeout.  In
                particular, when the timeout algorithm is rsre(3),
                an object of this type has the semantics of the
                UBOUND quantity described in RFC 793."
               ::= { ipv6TcpMIBObjects 3 }

           ipv6TcpMaxConn OBJECT-TYPE
               SYNTAX     Integer32
               MAX-ACCESS read-only
               STATUS     current
               DESCRIPTION
                "The limit on the total number of TCP connections
                the entity can support.  In entities where the
                maximum number of connections is dynamic, this
                object should contain the value -1."
               ::= { ipv6TcpMIBObjects 4 }

           ipv6TcpActiveOpens OBJECT-TYPE
               SYNTAX     Counter32
               MAX-ACCESS read-only
               STATUS     current
               DESCRIPTION
                "The number of times TCP connections have made a
                direct transition to the SYN-SENT state from the
                CLOSED state."
               ::= { ipv6TcpMIBObjects 5 }

           ipv6TcpPassiveOpens OBJECT-TYPE
               SYNTAX     Counter32
               MAX-ACCESS read-only
               STATUS     current
               DESCRIPTION
                "The number of times TCP connections have made a
                direct transition to the SYN-RCVD state from the
                LISTEN state."
               ::= { ipv6TcpMIBObjects 6 }

           ipv6TcpAttemptFails OBJECT-TYPE
               SYNTAX     Counter32
               MAX-ACCESS read-only
               STATUS     current
               DESCRIPTION
                "The number of times TCP connections have made a
                direct transition to the CLOSED state from either
```

```
         the SYN-SENT state or the SYN-RCVD state, plus the
         number of times TCP connections have made a direct
         transition to the LISTEN state from the SYN-RCVD
         state."
        ::= { ipv6TcpMIBObjects 7 }

     ipv6TcpEstabResets OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of times TCP connections have made a
         direct transition to the CLOSED state from either
         the ESTABLISHED state or the CLOSE-WAIT state."
        ::= { ipv6TcpMIBObjects 8 }

     ipv6TcpCurrEstab OBJECT-TYPE
        SYNTAX  Gauge32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The number of TCP connections for which the
         current state is either ESTABLISHED or CLOSE-
         WAIT."
        ::= { ipv6TcpMIBObjects 9 }

     ipv6TcpInSegs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The total number of segments received, including
         those received in error.  This count includes
         segments received on currently established
         connections."
        ::= { ipv6TcpMIBObjects 10 }

     ipv6TcpOutSegs OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
         "The total number of segments sent, including
         those on current connections but excluding those
         containing only retransmitted octets."
```

```
             ::= { ipv6TcpMIBObjects 11 }

          ipv6TcpRetransSegs OBJECT-TYPE
             SYNTAX     Counter32
             MAX-ACCESS read-only
             STATUS     current
             DESCRIPTION
              "The total number of segments retransmitted - that
              is, the number of TCP segments transmitted
              containing one or more previously transmitted
              octets."
             ::= { ipv6TcpMIBObjects 12 }

          ipv6TcpInErrs OBJECT-TYPE
             SYNTAX     Counter32
             MAX-ACCESS read-only
             STATUS     current
             DESCRIPTION
              "The total number of segments received in error
              (e.g., bad TCP checksums)."
             ::= { ipv6TcpMIBObjects 13 }

          ipv6TcpOutRsts OBJECT-TYPE
             SYNTAX     Counter32
             MAX-ACCESS read-only
             STATUS     current
             DESCRIPTION
              "The number of TCP segments sent containing the
              RST flag."
             ::= { ipv6TcpMIBObjects 14 }


          -- the TCPv6 Connection table

          -- The TCPv6 connection table contains information
          -- about this entity's existing TCPv6 connections.

          ipv6TcpConnTable OBJECT-TYPE
             SYNTAX      SEQUENCE OF Ipv6TcpConnEntry
             MAX-ACCESS  not-accessible
             STATUS      current
             DESCRIPTION
              "A table containing TCP connection-specific
              information."
             ::= { ipv6TcpMIBObjects 15 }
```

```
     ipv6TcpConnEntry OBJECT-TYPE
        SYNTAX      Ipv6TcpConnEntry
        MAX-ACCESS  not-accessible
        STATUS      current
        DESCRIPTION
         "Information about a particular current TCP
         connection.  An object of this type is transient,
         in that it ceases to exist when (or soon after)
         the connection makes the transition to the CLOSED
         state."
        INDEX   { ipv6IfIndex,
                  ipv6TcpConnLocalAddress,
                  ipv6TcpConnLocalPort,
                  ipv6TcpConnRemAddress,
                  ipv6TcpConnRemPort }
        ::= { ipv6TcpConnTable 1 }

     Ipv6TcpConnEntry ::=
        SEQUENCE {
            ipv6TcpConnLocalAddress    Ipv6Address,
            ipv6TcpConnLocalPort       INTEGER (0..65535),
            ipv6TcpConnRemAddress      Ipv6Address,
            ipv6TcpConnRemPort         INTEGER (0..65535),
            ipv6TcpConnState           INTEGER
         }

     ipv6TcpConnLocalAddress OBJECT-TYPE
        SYNTAX     Ipv6Address
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
         "The local IPv6 address for this TCP connection. In
         the case of a connection in the listen state which
         is willing to accept connections for any IPv6
         address associated with the interface, the value
         ::0 is used."
        ::= { ipv6TcpConnEntry 1 }

     ipv6TcpConnLocalPort OBJECT-TYPE
        SYNTAX  INTEGER (0..65535)
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
         "The local port number for this TCP connection."
        ::= { ipv6TcpConnEntry 2 }
```

```
     ipv6TcpConnRemAddress OBJECT-TYPE
        SYNTAX     Ipv6Address
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
         "The remote IPv6 address for this TCP connection."
        ::= { ipv6TcpConnEntry 3 }

     ipv6TcpConnRemPort OBJECT-TYPE
        SYNTAX  INTEGER (0..65535)
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
         "The remote port number for this TCP connection."
        ::= { ipv6TcpConnEntry 4 }

     ipv6TcpConnState OBJECT-TYPE
        SYNTAX  INTEGER {
             closed(1),
             listen(2),
             synSent(3),
             synReceived(4),
             established(5),
             finWait1(6),
             finWait2(7),
             closeWait(8),
             lastAck(9),
             closing(10),
             timeWait(11),
             deleteTCB(12)
         }
        MAX-ACCESS  read-write
        STATUS      current
        DESCRIPTION
         "The state of this TCP connection.

         The only value which may be set by a management
         station is deleteTCB(12).  Accordingly, it is
         appropriate for an agent to return a `badValue'
         response if a management station attempts to set
         this object to any other value.

         If a management station sets this object to the
         value deleteTCB(12), then this has the effect of
         deleting the TCB (as defined in RFC 793) of the
```

```
         corresponding connection on the managed node,
         resulting in immediate termination of the
         connection.

         As an implementation-specific option, a RST
         segment may be sent from the managed node to the
         other TCP endpoint (note however that RST segments
         are not sent reliably)."
        ::= { ipv6TcpConnEntry 5 }

   -- conformance information

   ipv6TcpConformance OBJECT IDENTIFIER ::= { ipv6TcpMIB 2 }

   ipv6TcpCompliances OBJECT IDENTIFIER ::= { ipv6TcpConformance 1 }
   ipv6TcpGroups      OBJECT IDENTIFIER ::= { ipv6TcpConformance 2 }

   -- compliance statements

   ipv6TcpCompliance MODULE-COMPLIANCE
       STATUS  current
       DESCRIPTION
         "The compliance statement for SNMPv2 entities which
         implement TCP over IPv6."
       MODULE  -- this module
           GROUP       ipv6TcpGroup
           DESCRIPTION
               "This group is mandatory for all entities
               which implement TCP over IPv6."
       ::= { ipv6TcpCompliances 1 }

   ipv6TcpGroup OBJECT-GROUP
       OBJECTS   {
                   ipv6TcpRtoAlgorithm,
                   ipv6TcpRtoMin,
                   ipv6TcpRtoMax,
                   ipv6TcpMaxConn,
                   ipv6TcpActiveOpens,
                   ipv6TcpPassiveOpens,
                   ipv6TcpAttemptFails,
                   ipv6TcpEstabResets,
                   ipv6TcpCurrEstab,
                   ipv6TcpInSegs,
                   ipv6TcpOutSegs,
                   ipv6TcpRetransSegs,
```

```
                   ipv6TcpInErrs,
                   ipv6TcpOutRsts,
                   ipv6TcpConnState
                 }
       STATUS    current
       DESCRIPTION
            "The TCP group of objects providing information
             specific to TCP over IPv6."
       ::= { ipv6TcpGroups 1 }

    END
```

6. Acknowledgements

This document borrows from MIB works produced by IETF for IPv4-based internets.

7. References

[1] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1902, January 1996.

[2] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Textual Conventions for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1903, January 1996.

[3] McCloghrie, K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, RFC 1213, Hughes LAN Systems, Performance Systems International, March 1991.

[4] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "A Simple Network Management Protocol (SNMP)", STD 15, RFC 1157, SNMP Research, Performance Systems International, MIT Lab for Computer Science, May 1990.

[5] SNMPv2 Working Group, Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.

[6] McCloghrie, K. and F. Kastenholz, "Evolution of the Interfaces Group of MIB-II" , RFC 1573, January 1994.

[7] Deering, S., and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 1883, December 1995.

[8] Jacobson, V., "Congestion Avoidance and Control", SIGCOMM 1988, Stanford, California.

8. Security Considerations

Security issues are not discussed in this memo.

9. Authors' Address

Dimitry Haskin
Bay Networks, Inc.
2 Federal Street
Billerica, MA 01821
email: dhaskin@baynetworks.com

Steve Onishi
Bay Networks, Inc.
3 Federal Street
Billerica, MA 01821
email: sonishi@baynetworks.com