IPv6 MIB Revision Design Team   Bill Fenner
INTERNET-DRAFT   AT&T Research
Expires: May 2002   Brian Haberman
                                            Nortel Networks
                                          Keith McCloghrie
                                             Cisco Systems
                                   Juergen Schoenwalder
                                          TU Braunschweig
                                             Dave Thaler
                                               Microsoft
                                           November 2001

Management Information Base
for the Transmission Control Protocol (TCP)
draft-ietf-ipngwg-rfc2012-update-01.txt

Status of this Document

This document is an Internet-Draft and is in full conformance with all
provisions of Section 10 of RFC2026.

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

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

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

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

This document is a product of the IPv6 MIB Revision Design Team.
Comments should be addressed to the authors, or the mailing list at
ipng@sunroof.eng.sun.com.

Copyright Notice

                                Abstract

This memo defines a portion of the Management Information Base (MIB) for
use with network management protocols in the Internet community. In
particular, it describes managed objects used for implementations of the
Transmission Control Protocol (TCP) [5] in an IP version independent
manner.

Table of Contents

1.   **The SNMP Management Framework**

The SNMP Management Framework presently consists of five major
components:

o An overall architecture, described in RFC 2571 [7].

o Mechanisms for describing and naming objects and events for the
  purpose of management. The first version of this Structure of
  Management Information (SMI) is called SMIv1 and described in STD 16,
  RFC 1155 [8], STD 16, RFC 1212 [9] and RFC 1215 [10]. The second
  version, called SMIv2, is described in STD 58, RFC 2578 [11], STD 58,
  RFC 2579 [12] and STD 58, RFC 2580 [13].

o Message protocols for transferring management information. The first
  version of the SNMP message protocol is called SNMPv1 and described in
  STD 15, RFC 1157 [14]. A second version of the SNMP message protocol,
  which is not an Internet standards track protocol, is called SNMPv2c
  and described in RFC 1901 [15] and RFC 1906 [16]. The third version of
  the message protocol is called SNMPv3 and described in RFC 1906 [16],
  RFC 2572 [17] and RFC 2574 [18].

o Protocol operations for accessing management information. The first
  set of protocol operations and associated PDU formats is described in
  STD 15, RFC 1157 [14]. A second set of protocol operations and
  associated PDU formats is described in RFC 1905 [19].

o A set of fundamental applications described in RFC 2573 [20] and the
  view-based access control mechanism described in RFC 2575 [21].

A more detailed introduction to the current SNMP Management Framework
can be found in RFC 2570 [22].

Managed objects are accessed via a virtual information store, termed the
Management Information Base or MIB. Objects in the MIB are defined
using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIv2. A MIB
conforming to the SMIv1 can be produced through the appropriate
translations. The resulting translated MIB must be semantically
equivalent, except where objects or events are omitted because no
translation is possible (use of Counter64). Some machine readable
information in SMIv2 will be converted into textual descriptions in
SMIv1 during the translation process. However, this loss of machine
readable information is not considered to change the semantics of the
MIB.

[2]. Revision History

Changes from draft-ietf-ipngwg-rfc2012-update-00.txt:

14 November 2001

    Added HC versions of connection counters.

    Added Listener table, with counters for accepted and timed out
    connection attempts.

    Added tcp{Connection,Listener}ProcessID to index into SYSAPPL-MIB
    or HOST-RESOURCES-MIB.

    Removed tcpConnectionRemAddrType, it must be the same as
    tcpConnectionLocalAddrType.

Changes from draft-ops-rfc2012-update-00.txt

12 Jul 2001

    Turned into IPNG WG document

    Added tcpCountersGroup for per-connection counters

Changes from first draft posted to v6mib mailing list:

23 Feb 2001

    Made threshold for HC packet counters 1Mpps

    Added copyright statements and table of contents

[21](#) Feb 2001 -- Juergen's changes

    Renamed tcpInetConn* to tcpConnection*

    Updated Conformance info

    Added missing tcpConnectionState and tcpConnState objects to
    SEQUENCEs

[6](#) Feb 2001

    Removed v6-only objects.

    Renamed inetTcp* to tcpInet*

    Added SIZE restriction to InetAddress index objects. (36 = 32-byte
    addresses plus 4-byte scope, but it's just a strawman)

    Used InetPortNumber TC from updated INET-ADDRESS-MIB

    Updated compliance statements.

    Added Keith to authors

    Added open issues section.

Changes from [RFC 2012](#)

    Deprecated tcpConnTable

    Added tcpConnectionTable

[3](#). MIB Structure

(Obviously this section needs a lot of work)

Addresses are as seen on the wire, not necessarily as the socket sees
them (e.g. IPv4 address, not IPv6-mapped IPv4)

Listener on in6addr_any without IPV6_V6ONLY socket option set (i.e.
willing to accept v4 or v6) is indicated by remote AF = unknown.

[4](#). Definitions

TCP-MIB DEFINITIONS ::= BEGIN

```
IMPORTS
     MODULE-IDENTITY, OBJECT-TYPE, Integer32, Unsigned32,
     Gauge32, Counter32, Counter64, IpAddress, mib-2
                                       FROM SNMPv2-SMI
     TimeStamp                         FROM SNMPv2-TC
     MODULE-COMPLIANCE, OBJECT-GROUP    FROM SNMPv2-CONF
     InetAddress, InetAddressType,
     InetPortNumber                    FROM INET-ADDRESS-MIB;

tcpMIB MODULE-IDENTITY
     LAST-UPDATED "200111140000Z"
     ORGANIZATION "IETF IPv6 MIB Revision Team"
     CONTACT-INFO
            "Bill Fenner (editor)

             AT&T Labs -- Research
             75 Willow Rd.
             Menlo Park, CA 94025

             Phone: +1 650 330-7893
             Email: <fenner@research.att.com>"
     DESCRIPTION
            "The MIB module for managing TCP implementations."
     REVISION      "200111140000Z"
     DESCRIPTION
            "IP version neutral revision, published as RFC XXXX."
     REVISION      "9411010000Z"
     DESCRIPTION
            "Initial SMIv2 version, published as RFC 2012."
     REVISION      "9103310000Z"
     DESCRIPTION
            "The initial revision of this MIB module was part of MIB-II."
     ::= { mib-2 49 }


-- the TCP base variables group

tcp     OBJECT IDENTIFIER ::= { mib-2 6 }

-- Scalars

tcpRtoAlgorithm 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 [1]
                 }
```

        MAX-ACCESS read-only
        STATUS   current
        DESCRIPTION
               "The algorithm used to determine the timeout value used for
                retransmitting unacknowledged octets."
        ::= { tcp 1 }

tcpRtoMin OBJECT-TYPE
        SYNTAX   Integer32
        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."
        ::= { tcp 2 }

tcpRtoMax OBJECT-TYPE
        SYNTAX   Integer32
        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."
        ::= { tcp 3 }

tcpMaxConn 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."
        ::= { tcp 4 }

```
tcpActiveOpens 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."
    ::= { tcp 5 }

tcpPassiveOpens 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."
    ::= { tcp 6 }

tcpAttemptFails 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."
    ::= { tcp 7 }

tcpEstabResets 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."
    ::= { tcp 8 }

tcpCurrEstab 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."
    ::= { tcp 9 }
```

```
tcpInSegs 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."
    ::= { tcp 10 }

tcpOutSegs 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."
    ::= { tcp 11 }

tcpRetransSegs 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."
    ::= { tcp 12 }

tcpInErrs OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
          "The total number of segments received in error (e.g., bad
           TCP checksums)."
    ::= { tcp 14 }

tcpOutRsts OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
          "The number of TCP segments sent containing the RST flag."
    ::= { tcp 15 }

tcpHCInSegs OBJECT-TYPE
```

```
    SYNTAX      Counter64
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
            "The total number of segments received, including those
             received in error, on systems that can receive more than 1
             million TCP packets per second.  This count includes
             segments received on currently established connections."
    ::= { tcp 17 }


tcpHCOutSegs OBJECT-TYPE
    SYNTAX      Counter64
    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, on systems that can transmit more than
             1 million TCP packets per second."
    ::= { tcp 18 }


-- The TCP Connection table

tcpConnectionTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TcpConnectionEntry
    MAX-ACCESS not-accessible
    STATUS      current
    DESCRIPTION
            "A table containing information about existing TCP
             connections.  Note that unlike earlier TCP MIBs, there is a
             seperate table for connections in the LISTEN state."
    ::= { tcp 19 }


tcpConnectionEntry OBJECT-TYPE
    SYNTAX      TcpConnectionEntry
    MAX-ACCESS not-accessible
    STATUS      current
    DESCRIPTION
            "A conceptual row of the tcpConnectionTable containing
             information about a particular current TCP connection.  Each
             row of this table is transient, in that it ceases to exist
             when (or soon after) the connection makes the transition to
             the CLOSED state."
    INDEX   { tcpConnectionLocalAddressType,
               tcpConnectionLocalAddress,
               tcpConnectionLocalPort,
               tcpConnectionRemAddress,
               tcpConnectionRemPort }
```

```
     ::= { tcpConnectionTable 1 }

TcpConnectionEntry ::= SEQUENCE {
        tcpConnectionLocalAddressType  InetAddressType,
        tcpConnectionLocalAddress      InetAddress,
        tcpConnectionLocalPort         InetPortNumber,
        tcpConnectionRemAddress        InetAddress,
        tcpConnectionRemPort           InetPortNumber,
        tcpConnectionState             INTEGER,
        tcpConnectionInPackets         Counter32,
        tcpConnectionOutPackets        Counter32,
        tcpConnectionInOctets          Counter32,
        tcpConnectionOutOctets         Counter32,
        tcpConnectionHCInPackets       Counter64,
        tcpConnectionHCOutPackets      Counter64,
        tcpConnectionHCInOctets        Counter64,
        tcpConnectionHCOutOctets       Counter64,
        tcpConnectionStartTime         TimeStamp,
        tcpConnectionProcessID         Unsigned32
    }

tcpConnectionLocalAddressType OBJECT-TYPE
    SYNTAX     InetAddressType
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
           "The address type of tcpConnectionLocalAddress.  Only IPv4
            and IPv6 addresses are expected."
    ::= { tcpConnectionEntry 1 }

tcpConnectionLocalAddress OBJECT-TYPE
    SYNTAX     InetAddress (SIZE(0..36))
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
           "The local IP address for this TCP connection.  In the case
            of a connection in the listen state which is willing to
            accept connections for any IP interface associated with the
            node, a value of all zeroes is used."
    ::= { tcpConnectionEntry 2 }

tcpConnectionLocalPort OBJECT-TYPE
    SYNTAX     InetPortNumber
    MAX-ACCESS not-accessible
    STATUS     current
    DESCRIPTION
           "The local port number for this TCP connection."
    ::= { tcpConnectionEntry 3 }
```

tcpConnectionRemAddress OBJECT-TYPE
    SYNTAX   InetAddress (SIZE(0..36))
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
            "The remote IP address for this TCP connection."
    ::= { tcpConnectionEntry 4 }

tcpConnectionRemPort OBJECT-TYPE
    SYNTAX   InetPortNumber
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
            "The remote port number for this TCP connection."
    ::= { tcpConnectionEntry 5 }

tcpConnectionState 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 value listen(2) is included only for parallelism to the
            old tcpConnTable, and should not be used. A connection in
            LISTEN state should be present in the tcpListenerTable.

            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)."
     ::= { tcpConnectionEntry 6 }

tcpConnectionInPackets OBJECT-TYPE
     SYNTAX   Counter32
     MAX-ACCESS read-only
     STATUS   current
     DESCRIPTION
             "The number of packets received on this connection. This
              count includes retransmitted data."
     ::= { tcpConnectionEntry 7 }

tcpConnectionOutPackets OBJECT-TYPE
     SYNTAX   Counter32
     MAX-ACCESS read-only
     STATUS   current
     DESCRIPTION
             "The number of packets transmitted on this connection. This
              count includes retransmitted data."
     ::= { tcpConnectionEntry 8 }

tcpConnectionInOctets OBJECT-TYPE
     SYNTAX   Counter32
     MAX-ACCESS read-only
     STATUS   current
     DESCRIPTION
             "The number of octets received on this connection. This
              count includes retransmitted data."
     ::= { tcpConnectionEntry 9 }

tcpConnectionOutOctets OBJECT-TYPE
     SYNTAX   Counter32
     MAX-ACCESS read-only
     STATUS   current
     DESCRIPTION
             "The number of octets transmitted on this connection. This
              count includes retransmitted data."
     ::= { tcpConnectionEntry 10 }

tcpConnectionHCInPackets OBJECT-TYPE
     SYNTAX   Counter64
     MAX-ACCESS read-only
     STATUS   current

```
    DESCRIPTION
            "The number of packets received on this connection.  This
             count includes retransmitted data."
    ::= { tcpConnectionEntry 11 }

tcpConnectionHCOutPackets OBJECT-TYPE
    SYNTAX     Counter64
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The number of packets transmitted on this connection.  This
             count includes retransmitted data."
    ::= { tcpConnectionEntry 12 }

tcpConnectionHCInOctets OBJECT-TYPE
    SYNTAX     Counter64
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The number of octets received on this connection.  This
             count includes retransmitted data."
    ::= { tcpConnectionEntry 13 }

tcpConnectionHCOutOctets OBJECT-TYPE
    SYNTAX     Counter64
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The number of octets transmitted on this connection.  This
             count includes retransmitted data."
    ::= { tcpConnectionEntry 14 }

tcpConnectionStartTime OBJECT-TYPE
    SYNTAX     TimeStamp
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The value of sysUpTime at the time this connection was
             established"
    ::= { tcpConnectionEntry 15 }

tcpConnectionProcessID OBJECT-TYPE
    SYNTAX     Unsigned32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The system's process ID for the process associated with this
             connection, or zero if there is no such process.  This value
```

```
            is expected to be the same as HOST-RESOURCES-
            MIB::hrSWRunIndex or SYSAPPL-MIB::sysApplElmtRunIndex for
            some row in the appropriate tables."
     ::= { tcpConnectionEntry 16 }



-- The TCP Listener table

tcpListenerTable OBJECT-TYPE
     SYNTAX     SEQUENCE OF TcpListenerEntry
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
           "A table containing information about TCP listeners."
     ::= { tcp 20 }

tcpListenerEntry OBJECT-TYPE
     SYNTAX     TcpListenerEntry
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
           "A conceptual row of the tcpListenerTable containing
            information about a particular TCP listener."
     INDEX   { tcpListenerLocalAddressType,
              tcpListenerLocalAddress,
              tcpListenerLocalPort,
              tcpListenerRemAddressType }
     ::= { tcpListenerTable 1 }

TcpListenerEntry ::= SEQUENCE {
        tcpListenerLocalAddressType      InetAddressType,
        tcpListenerLocalAddress          InetAddress,
        tcpListenerLocalPort             InetPortNumber,
        tcpListenerRemAddressType        InetAddressType,
        tcpListenerConnectionsTimedOut   Counter32,
        tcpListenerHCConnectionsTimedOut Counter64,
        tcpListenerConnectionsAccepted   Counter32,
        tcpListenerHCConnectionsAccepted Counter64,
        tcpListenerStartTime             TimeStamp,
        tcpListenerProcessID             Unsigned32
     }

tcpListenerLocalAddressType OBJECT-TYPE
     SYNTAX     InetAddressType
     MAX-ACCESS not-accessible
     STATUS     current
     DESCRIPTION
```

            "The address type of tcpListenerLocalAddress. Only IPv4 and
             IPv6 addresses are expected."
     ::= { tcpListenerEntry 1 }

tcpListenerLocalAddress OBJECT-TYPE
    SYNTAX   InetAddress (SIZE(0..36))
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
            "The local IP address for this TCP connection. In the case
             of a connection in the listen state which is willing to
             accept connections for any IP interface associated with the
             node, a value of all zeroes is used."
     ::= { tcpListenerEntry 2 }

tcpListenerLocalPort OBJECT-TYPE
    SYNTAX   InetPortNumber
    MAX-ACCESS not-accessible
    STATUS   current
    DESCRIPTION
            "The local port number for this TCP connection."
     ::= { tcpListenerEntry 3 }

tcpListenerRemAddressType OBJECT-TYPE
    SYNTAX   InetAddressType
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
            "The address type of connections that will be accepted by
             this listener. Only IPv4 and IPv6 addresses are expected,
             or unknown to indicate an endpoint willing to accept both
             IPv4 and IPv6 connections."
     ::= { tcpListenerEntry 4 }

tcpListenerConnectionsTimedOut OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
            "The number of connection attempts to this endpoint which
             have failed due to timeout of the three-way handshake, i.e.
             the row was removed from the tcpConnectionTable but
             tcpConnectionState never moved from synReceived to
             established."
     ::= { tcpListenerEntry 5 }

tcpListenerHCConnectionsTimedOut OBJECT-TYPE
    SYNTAX   Counter64

```
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The number of connection attempts to this endpoint which
             have failed due to timeout of the three-way handshake, i.e.
             the row was removed from the tcpConnectionTable but
             tcpConnectionState never moved from synReceived to
             established."
    ::= { tcpListenerEntry 6 }


tcpListenerConnectionsAccepted OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The number of connections which have been established to
             this endpoint."
    ::= { tcpListenerEntry 7 }


tcpListenerHCConnectionsAccepted OBJECT-TYPE
    SYNTAX     Counter64
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The number of connections which have been established to
             this endpoint."
    ::= { tcpListenerEntry 8 }


tcpListenerStartTime OBJECT-TYPE
    SYNTAX     TimeStamp
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The value of sysUpTime at the time this listener was
             established."
    ::= { tcpListenerEntry 9 }


tcpListenerProcessID OBJECT-TYPE
    SYNTAX     Unsigned32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
            "The system's process ID for the process associated with this
             listener, or zero if there is no such process.  This value
             is expected to be the same as HOST-RESOURCES-
             MIB::hrSWRunIndex or SYSAPPL-MIB::sysApplElmtRunIndex for
             some row in the appropriate tables."
    ::= { tcpListenerEntry 10 }
```

-- The deprecated TCP Connection table

```
tcpConnTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF TcpConnEntry
    MAX-ACCESS not-accessible
    STATUS      deprecated
    DESCRIPTION
            "A table containing information about existing IPv4-specific
             TCP connections or listeners.  This table has been
             deprecated in favor of the version neutral
             tcpConnectionTable."
    ::= { tcp 13 }

tcpConnEntry OBJECT-TYPE
    SYNTAX      TcpConnEntry
    MAX-ACCESS not-accessible
    STATUS      deprecated
    DESCRIPTION
            "A conceptual row of the tcpConnTable containing information
             about a particular current IPv4 TCP connection.  Each row of
             this table is transient, in that it ceases to exist when (or
             soon after) the connection makes the transition to the
             CLOSED state."
    INDEX   { tcpConnLocalAddress,
              tcpConnLocalPort,
              tcpConnRemAddress,
              tcpConnRemPort }
    ::= { tcpConnTable 1 }

TcpConnEntry ::= SEQUENCE {
        tcpConnState         INTEGER,
        tcpConnLocalAddress  IpAddress,
        tcpConnLocalPort     INTEGER,
        tcpConnRemAddress    IpAddress,
        tcpConnRemPort       INTEGER
    }

tcpConnState 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     deprecated
    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)."
    ::= { tcpConnEntry 1 }

tcpConnLocalAddress OBJECT-TYPE
    SYNTAX     IpAddress
    MAX-ACCESS read-only
    STATUS     deprecated
    DESCRIPTION
            "The local IP address for this TCP connection.  In the case
             of a connection in the listen state which is willing to
             accept connections for any IP interface associated with the
             node, the value 0.0.0.0 is used."
    ::= { tcpConnEntry 2 }

tcpConnLocalPort OBJECT-TYPE
    SYNTAX     INTEGER (0..65535)
    MAX-ACCESS read-only
    STATUS     deprecated
    DESCRIPTION
            "The local port number for this TCP connection."
    ::= { tcpConnEntry 3 }

tcpConnRemAddress OBJECT-TYPE
    SYNTAX     IpAddress
    MAX-ACCESS read-only
    STATUS     deprecated
```

```
    DESCRIPTION
            "The remote IP address for this TCP connection."
    ::= { tcpConnEntry 4 }

tcpConnRemPort OBJECT-TYPE
    SYNTAX     INTEGER (0..65535)
    MAX-ACCESS read-only
    STATUS     deprecated
    DESCRIPTION
            "The remote port number for this TCP connection."
    ::= { tcpConnEntry 5 }




-- conformance information

tcpMIBConformance OBJECT IDENTIFIER ::= { tcpMIB 2 }

tcpMIBCompliances OBJECT IDENTIFIER ::= { tcpMIBConformance 1 }
tcpMIBGroups      OBJECT IDENTIFIER ::= { tcpMIBConformance 2 }


-- compliance statements

tcpMIBCompliance2 MODULE-COMPLIANCE
    STATUS     current
    DESCRIPTION
            "The compliance statement for systems which implement TCP."
    MODULE  -- this module
        MANDATORY-GROUPS { tcpBaseGroup, tcpConnectionGroup, tcpListenerGroup }
        GROUP       tcpHCGroup
        DESCRIPTION
           "This group is mandatory for those systems which are capable
                of receiving or transmitting more than 1 million TCP
                packets per second.  1 million packets per second will
                cause a Counter32 to wrap in just over an hour."
        GROUP       tcpStatisticsGroup
        DESCRIPTION
           "This group is optional.  It provides visibility for counters
                that some systems already implement."
        GROUP       tcpHCStatisticsGroup
        DESCRIPTION
           "This group is mandatory for those systems which implement
                the tcpStatisticsGroup and are capable of receiving or
                transmitting more than 1 million TCP packets per second.
                1 million packets per second will cause a Counter32 to
                wrap in just over an hour."
```

```
        OBJECT       tcpConnectionState
        MIN-ACCESS   read-only
        DESCRIPTION
            "Write access is not required."
     ::= { tcpMIBCompliances 2 }

tcpMIBCompliance MODULE-COMPLIANCE
    STATUS       deprecated
    DESCRIPTION
            "The compliance statement for IPv4-only systems which
             implement TCP.  In order to be IP version independent, this
             compliance statement is deprecated in favor of
             tcpMIBCompliance2.  However, agents are still encouraged to
             implement these objects in order to interoperate with the
             deployed base of managers."
    MODULE   -- this module
        MANDATORY-GROUPS { tcpGroup }
        OBJECT       tcpConnState
        MIN-ACCESS   read-only
        DESCRIPTION
            "Write access is not required."
     ::= { tcpMIBCompliances 1 }

-- units of conformance

tcpGroup OBJECT-GROUP
    OBJECTS   { tcpRtoAlgorithm, tcpRtoMin, tcpRtoMax,
                tcpMaxConn, tcpActiveOpens,
                tcpPassiveOpens, tcpAttemptFails,
                tcpEstabResets, tcpCurrEstab, tcpInSegs,
                tcpOutSegs, tcpRetransSegs, tcpConnState,
                tcpConnLocalAddress, tcpConnLocalPort,
                tcpConnRemAddress, tcpConnRemPort,
                tcpInErrs, tcpOutRsts }
    STATUS      deprecated
    DESCRIPTION
            "The tcp group of objects providing for management of TCP
             entities."
     ::= { tcpMIBGroups 1 }

tcpBaseGroup OBJECT-GROUP
    OBJECTS   { tcpRtoAlgorithm, tcpRtoMin, tcpRtoMax,
                tcpMaxConn, tcpActiveOpens,
                tcpPassiveOpens, tcpAttemptFails,
                tcpEstabResets, tcpCurrEstab, tcpInSegs,
                tcpOutSegs, tcpRetransSegs,
                tcpInErrs, tcpOutRsts }
    STATUS      current
```

         DESCRIPTION
                "The group of counters common to TCP entities."
         ::= { tcpMIBGroups 2 }

    tcpHCGroup OBJECT-GROUP
         OBJECTS   { tcpHCInSegs, tcpHCOutSegs }
         STATUS   current
         DESCRIPTION
                "The group of objects providing for counters of high speed
                 TCP implementations."
         ::= { tcpMIBGroups 3 }

    tcpConnectionGroup OBJECT-GROUP
         OBJECTS   { tcpConnectionState }
         STATUS   current
         DESCRIPTION
                "The table of TCP connections."
         ::= { tcpMIBGroups 4 }

    tcpListenerGroup OBJECT-GROUP
         OBJECTS   { tcpListenerRemAddressType }
         STATUS   current
         DESCRIPTION
                "The table of TCP listeners."
         ::= { tcpMIBGroups 5 }

    tcpStatisticsGroup OBJECT-GROUP
         OBJECTS   { tcpConnectionInPackets, tcpConnectionOutPackets,
                      tcpConnectionInOctets, tcpConnectionOutOctets,
                              tcpConnectionStartTime, tcpConnectionProcessID,
                      tcpListenerConnectionsTimedOut,
                      tcpListenerConnectionsAccepted,
                      tcpListenerStartTime, tcpListenerProcessID }
         STATUS   current
         DESCRIPTION
                "The packet and octet counters and other statistics specific
                 to a TCP connection or listener."
         ::= { tcpMIBGroups 6 }

    tcpHCStatisticsGroup OBJECT-GROUP
         OBJECTS   { tcpConnectionHCInPackets, tcpConnectionHCOutPackets,
                      tcpConnectionHCInOctets, tcpConnectionHCOutOctets,
                      tcpListenerHCConnectionsTimedOut,
                      tcpListenerHCConnectionsAccepted }
         STATUS   current
         DESCRIPTION
                "The group of objects providing for statistics for listeners
                 or connections on high speed TCP implementations."

```
    ::= { tcpMIBGroups 7 }
```


END

5. Open Issues

Why is tcpListenerRemoteAddressType = unknown better than
tcpListenerLocalAddressType = unknown and tcpListenerLocalAddress = ''h?
Then we could get rid of tcpListenerRemoteAddressType.

Is tcp*ProcessID OK? Should there be an OID pointer into a row of some
Run table?

Are the current per-connection byte/segment counters appropriate? Other
stats? [in optional conformance group] ConnSRTT?

More HC counters?

6. Acknowledgements

This document contains a modified subset of RFC 1213 and updates RFC
2012 and RFC 2452.

7. References

[2] Rose, M. and K. McCloghrie, "Management Information Base for Network
     Management of TCP/IP-based internets", RFC 1213, March 1991.

[3] K. McCloghrie, "SNMPv2 Management Information Base for the
     Transmission Control Protocol using SMIv2", RFC 2012, November
     1996.

[4] Haskin, D. and S. Onishi, "IP Version 6 Management Information Base
     for the Transmission Control Protocol", RFC 2452, December 1998.

[5] Postel, J., "Transmission Control Protocol - DARPA Internet Program
     Protocol Specification", STD 7, RFC 793, DARPA, September 1981.

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

[7] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for
     Describing SNMP Management Frameworks", RFC 2571, April 1999.

[8] Rose, M., and K. McCloghrie, "Structure and Identification of
     Management Information for TCP/IP-based Internets", STD 16, RFC
     1155, May 1990.

[9] Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD 16, RFC
     1212, March 1991.

[10] Rose, M., "A Convention for Defining Traps for use with the SNMP",
     RFC 1215, March 1991.

[11] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M.,
     and S. Waldbusser, "Structure of Management Information Version 2
     (SMIv2)", STD 58, RFC 2578, April 1999.

[12] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M.,
     and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, RFC
     2579, April 1999.

[13] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M.,
     and S. Waldbusser, "Conformance Statements for SMIv2", STD 58, RFC
     2580, April 1999.

[14] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network
     Management Protocol", STD 15, RFC 1157, May 1990.

[15] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser,
     "Introduction to Community-based SNMPv2", RFC 1901, January 1996.

[16] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Transport
     Mappings for Version 2 of the Simple Network Management Protocol
     (SNMPv2)", RFC 1906, January 1996.

[17] Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message
     Processing and Dispatching for the Simple Network Management
     Protocol (SNMP)", RFC 2572, April 1999.

[18] Blumenthal, U., and B. Wijnen, "User-based Security Model (USM) for
     version 3 of the Simple Network Management Protocol (SNMPv3)", RFC
     2574, April 1999.

[19] 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.

[20] Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications", RFC
     2573, April 1999.

[21] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access
     Control Model (VACM) for the Simple Network Management Protocol
     (SNMP)", RFC 2575, April 1999.

[22] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction to
     Version 3 of the Internet-standard Network Management Framework",
     RFC 2570, April 1999.

8. Security Considerations

There are a number of management objects defined in this MIB that have a
MAX-ACCESS clause of read-write and/or read-create. Such objects may be
considered sensitive or vulnerable in some network environments. The
support for SET operations in a non-secure environment without proper
protection can have a negative effect on network operations.

There are a number of managed objects in this MIB that may contain
sensitive information. These are:

o The tcpConnectionLocalPort and tcpConnLocalPort objects can be used to
  identify what ports are open on the machine and can thus what attacks
  are likely to succeed, without the attacker having to run a port
  scanner.

o The tcpConnectionState and tcpConnState objects have a MAX-ACCESS
  clause of read-write, which allows termination of an arbitrary
  connection. Unauthorized access could cause a denial of service.

It is thus important to control even GET access to these objects and
possibly to even encrypt the values of these object when sending them
over the network via SNMP. Not all versions of SNMP provide features
for such a secure environment.

SNMPv1 by itself is not a secure environment. Even if the network
itself is secure (for example by using IPSec), even then, there is no
control as to who on the secure network is allowed to access and GET/SET
(read/change/create/delete) the objects in this MIB.

It is recommended that the implementers consider the security features
as provided by the SNMPv3 framework. Specifically, the use of the User-
based Security Model RFC 2574 [18] and the View-based Access Control
Model RFC 2575 [21] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity
giving access to an instance of this MIB, is properly configured to give

access to the objects only to those principals (users) that have
legitimate rights to indeed GET or SET (change/create/delete) them.

9. **Editor's Address**

Bill Fenner
AT&T Labs -- Research
75 **Willow Rd**
Menlo Park, CA 94025
USA

Email: fenner@research.att.com

10. **Full Copyright Statement**