

IP Version 6 Management Information Base  
for the Transmission Control Protocol

Status of this Memo

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

Copyright Notice

Copyright (C) The Internet Society (1998). All Rights Reserved.

Abstract

This document is one in the series of documents that define various MIB objects for IPv6. Specifically, this document is the MIB module which defines managed objects for implementations of the Transmission Control Protocol (TCP) over IP Version 6 (IPv6).

This document also recommends a specific policy with respect to the applicability of RFC 2012 for implementations of IPv6. Namely, that most of managed objects defined in RFC 2012 are independent of which IP versions underlie TCP, and only the TCP connection information is IP version-specific.

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

1. Introduction

A management system contains: several (potentially many) nodes, each with a processing entity, termed an agent, which has access to management instrumentation; at least one management station; and, a management protocol, used to convey management information between the agents and management stations. Operations of the protocol are carried out under an administrative framework which defines authentication, authorization, access control, and privacy policies.

Management stations execute management applications which monitor and control managed elements. Managed elements are devices such as hosts, routers, terminal servers, etc., which are monitored and controlled via access to their management information.

Management information is viewed as a collection of managed objects, residing in a virtual information store, termed the Management Information Base (MIB). Collections of related objects are defined in MIB modules. These modules are written using a subset of OSI's Abstract Syntax Notation One (ASN.1) [1], termed the Structure of Management Information (SMI) [2].

## 2. Overview

This document is one in the series of documents that define various MIB objects, and statements of conformance, for IPv6. This document defines the required instrumentation for implementations of TCP over IPv6.

## 3. Transparency of IP versions to TCP

The fact that a particular TCP connection uses IPv6 as opposed to IPv4, is largely invisible to a TCP implementation. A "TCPng" did not need to be defined, implementations simply need to support IPv6 addresses.

As such, the managed objects already defined in [TCP MIB] are sufficient for managing TCP in the presence of IPv6. These objects are equally applicable whether the managed node supports IPv4 only, IPv6 only, or both IPv4 and IPv6.

For example, tcpActiveOpens counts "The number of times TCP connections have made a direct transition to the SYN-SENT state from the CLOSED state", regardless of which version of IP is used between the connection endpoints.

Stated differently, TCP implementations don't need separate counters for IPv4 and for IPv6.

## 4. Representing TCP Connections

The exception to the statements in section 3 is the tcpConnTable. Since IPv6 addresses cannot be represented with the IpAddress syntax, not all TCP connections can be represented in the tcpConnTable defined in [TCP MIB].

This memo defines a new, separate table to represent only those TCP connections between IPv6 endpoints. TCP connections between IPv4 endpoints continue to be represented in tcpConnTable [TCP MIB]. (It is not possible to establish a TCP connection between an IPv4 endpoint and an IPv6 endpoint.)

A different approach would have been to define a new table to represent all TCP connections regardless of IP version. This would require changes to [TCP MIB] and hence to existing (IPv4-only) TCP implementations. The approach suggested in this memo has the advantage of leaving IPv4-only implementations intact.

It is assumed that the objects defined in this memo will eventually be defined in an update to [TCP MIB]. For this reason, the module identity is assigned under the experimental portion of the MIB.

## 5. Conformance

This memo contains conformance statements to define conformance to this MIB for TCP over IPv6 implementations.

## 6. Definitions

IPV6-TCP-MIB DEFINITIONS ::= BEGIN

### IMPORTS

```
MODULE-COMPLIANCE, OBJECT-GROUP          FROM SNMPv2-CONF
MODULE-IDENTITY, OBJECT-TYPE,
mib-2, experimental                      FROM SNMPv2-SMI
Ipv6Address, Ipv6IfIndexOrZero           FROM IPV6-TC;
```

### ipv6TcpMIB MODULE-IDENTITY

```
LAST-UPDATED "9801290000Z"
ORGANIZATION "IETF IPv6 MIB Working Group"
CONTACT-INFO
    "Mike Daniele
```

```
Postal: Compaq Computer Corporation
        110 Spitbrook Rd
        Nashua, NH 03062.
        US
```

```
Phone: +1 603 884 1423
Email: daniele@zk3.dec.com"
```

### DESCRIPTION

```
"The MIB module for entities implementing TCP over IPv6."
::= { experimental 86 }
```

-- objects specific to TCP for IPv6

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

-- the TCP over IPv6 Connection table

-- This connection table contains information about this  
 -- entity's existing TCP connections between IPv6 endpoints.  
 -- Only connections between IPv6 addresses are contained in  
 -- this table. This entity's connections between IPv4  
 -- endpoints are contained in tcpConnTable.

ipv6TcpConnTable OBJECT-TYPE

SYNTAX SEQUENCE OF Ipv6TcpConnEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A table containing TCP connection-specific information,  
 for only those connections whose endpoints are IPv6 addresses."

::= { tcp 16 }

ipv6TcpConnEntry OBJECT-TYPE

SYNTAX Ipv6TcpConnEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A conceptual row of the ipv6TcpConnTable 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.

Note that conceptual rows in this table require an additional  
 index object compared to tcpConnTable, since IPv6 addresses  
 are not guaranteed to be unique on the managed node."

INDEX { ipv6TcpConnLocalAddress,  
 ipv6TcpConnLocalPort,  
 ipv6TcpConnRemAddress,  
 ipv6TcpConnRemPort,  
 ipv6TcpConnIfIndex }

::= { ipv6TcpConnTable 1 }

Ipv6TcpConnEntry ::=

SEQUENCE {	ipv6TcpConnLocalAddress	Ipv6Address,
	ipv6TcpConnLocalPort	INTEGER (0..65535),
	ipv6TcpConnRemAddress	Ipv6Address,
	ipv6TcpConnRemPort	INTEGER (0..65535),
	ipv6TcpConnIfIndex	Ipv6IfIndexOrZero,

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 managed node, 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 }

ipv6TcpConnIfIndex OBJECT-TYPE

SYNTAX        Ipv6IfIndexOrZero

MAX-ACCESS not-accessible

STATUS        current

DESCRIPTION

"An index object used to disambiguate conceptual rows in the table, since the connection 4-tuple may not be unique."

If the connection's remote address (ipv6TcpConnRemAddress) is a link-local address and the connection's local address

(ipv6TcpConnLocalAddress) is not a link-local address, this object identifies a local interface on the same link as the connection's remote link-local address.

Otherwise, this object identifies the local interface that is associated with the ipv6TcpConnLocalAddress for this TCP connection. If such a local interface cannot be determined, this object should take on the value 0. (A possible example of this would be if the value of ipv6TcpConnLocalAddress is ::0.)

The interface identified by a particular non-0 value of this index is the same interface as identified by the same value of ipv6IfIndex.

The value of this object must remain constant during the life of the TCP connection."

::= { ipv6TcpConnEntry 5 }

#### 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 an error response ('badValue' for SNMPv1, 'wrongValue' for SNMPv2) 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 6 }

--
-- 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
    MANDATORY-GROUPS { ipv6TcpGroup }
    ::= { ipv6TcpCompliances 1 }

ipv6TcpGroup OBJECT-GROUP
    OBJECTS      { -- these are defined in this module
        -- ipv6TcpConnLocalAddress (not-accessible)
        -- ipv6TcpConnLocalPort (not-accessible)
        -- ipv6TcpConnRemAddress (not-accessible)
        -- ipv6TcpConnRemPort (not-accessible)
        -- ipv6TcpConnIfIndex (not-accessible)
        ipv6TcpConnState }
    STATUS      current
    DESCRIPTION
        "The group of objects providing management of
        TCP over IPv6."
    ::= { ipv6TcpGroups 1 }

END
```

## 7. Acknowledgments

This memo is a product of the IPng work group, and benefited especially from the contributions of the following working group members:

Dimitry Haskin	Bay Networks
Margaret Forsythe	Epilogue
Tim Hartrick	Mentat
Frank Solensky	FTP
Jack McCann	DEC

## 8. References

- [1] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization. International Standard 8824, (December, 1987).
- [2] McCloghrie, K., Editor, "Structure of Management Information for version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1902, January 1996.
- [TCP MIB] SNMPv2 Working Group, McCloghrie, K., Editor, "SNMPv2 Management Information Base for the Transmission Control Protocol using SMIV2", RFC 2012, November 1996.
- [IPV6 MIB TC] Haskin, D., and S. Onishi, "Management Information Base for IP Version 6: Textual Conventions and General Group", RFC 2465, December 1998.
- [IPV6] Deering, S., and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.
- [RFC2274] Blumenthal, U., and B. Wijnen, "The User-Based Security Model for Version 3 of the Simple Network Management Protocol (SNMPv3)", RFC 2274, January 1998.
- [RFC2275] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model for the Simple Network Management Protocol (SNMP)", RFC 2275, January 1998.

## 9. Security Considerations

This MIB contains a management object that has a MAX-ACCESS clause of read-write and/or read-create. In particular, it is possible to delete individual TCP control blocks (i.e., connections).



Consequently, anyone having the ability to issue a SET on this object can impact the operation of the node.

There are a number of managed objects in this MIB that may be considered to contain sensitive information in some environments. For example, the MIB identifies the active TCP connections on the node. Although this information might be considered sensitive in some environments (i.e., to identify ports on which to launch denial-of-service or other attacks), there are already other ways of obtaining similar information. For example, sending a random TCP packet to an unused port prompts the generation of a TCP reset message.

Therefore, it may be important in some environments to control read and/or write 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 does not provide encryption or strong authentication.

It is recommended that the implementors consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model [RFC2274] and the View-based Access Control Model [RFC2275] 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 those objects only to those principals (users) that have legitimate rights to access them.

#### 10. Author's Address

Mike Daniele  
Compaq Computer Corporation  
110 Spit Brook Rd  
Nashua, NH 03062

Phone: +1-603-884-1423  
EMail: danielle@zk3.dec.com

## 11. Full Copyright Statement

Copyright (C) The Internet Society (1998). All Rights Reserved.

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

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

