Network Working Group
Request for Comments: 2012
Updates: 1213
Category: Standards Track

K. McCloghrie, Editor
Cisco Systems
November 1996

SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2

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.

IESG Note:

The IP, UDP, and TCP MIB modules currently support only IPv4. These three modules use the IpAddress type defined as an OCTET STRING of length 4 to represent the IPv4 32-bit internet addresses. (See RFC 1902, SMI for SNMPv2.) They do not support the new 128-bit IPv6 internet addresses.

Table of Contents

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].

This document is the MIB module which defines managed objects for managing implementations of the Transmission Control Protocol (TCP) [3].

The managed objects in this MIB module were originally defined using the SNMPv1 framework as a part of MIB-II [4]. This document defines the same objects for TCP using the SNMPv2 framework.

2. Definitions

```
TCP-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, Integer32, Gauge32,
    Counter32, IpAddress, mib-2        FROM SNMPv2-SMI
    MODULE-COMPLIANCE, OBJECT-GROUP    FROM SNMPv2-CONF;

tcpMIB MODULE-IDENTITY
    LAST-UPDATED "9411010000Z"
    ORGANIZATION "IETF SNMPv2 Working Group"
    CONTACT-INFO
            "        Keith McCloghrie

             Postal: Cisco Systems, Inc.
                     170 West Tasman Drive
                     San Jose, CA  95134-1706
                     US

             Phone:  +1 408 526 5260
             Email:  kzm@cisco.com"
```

```
    DESCRIPTION
            "The MIB module for managing TCP implementations."
    REVISION      "9103310000Z"
    DESCRIPTION
            "The initial revision of this MIB module was part of MIB-
            II."
    ::= { mib-2 49 }

-- the TCP group

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

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 [5]
                }
    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."
```

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.