Network Working Group
Request For Comments: 1304

T. Cox
K. Tesink
Editors
Bell Communications Research
February 1992

Definitions of Managed Objects for the SIP Interface Type

Status of this Memo

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

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in TCP/IP-based internets. In particular, it defines objects for managing SIP (SMDS Interface Protocol) objects.

Table of Contents

1. The Network Management Framework

The Internet-standard Network Management Framework consists of three components. They are:

RFC 1155 [3] which defines the SMI, the mechanisms used for describing and naming objects for the purpose of management. RFC 1212 [9] defines a more concise description mechanism, which is wholly consistent with the SMI.

RFC 1156 [4] which defines MIB-I, the core set of managed objects for the Internet suite of protocols. RFC 1213 [6], defines MIB-II, an evolution of MIB-I based on implementation experience and new operational requirements.

RFC 1157 [5] which defines the SNMP, the protocol used for network access to managed objects.

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

2. Objects

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) International Standard 8824 [7] defined in the SMI. In particular, each object has a name, a syntax, and an encoding. The name is an object identifier, an administratively assigned name, which specifies an object type. 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 OBJECT DESCRIPTOR, to also refer to the object type.

The syntax of an object type defines the abstract data structure corresponding to that object type. The ASN.1 language is used for this purpose. However, the SMI RFC 1155 [3] purposely restricts the ASN.1 constructs which may be used. These restrictions are explicitly made for simplicity.

The encoding of an object type is simply how that object type is represented using the object type's syntax. Implicitly tied to the notion of an object type's syntax and encoding is how the object type is represented when being transmitted on the network. The SMI specifies the use of the basic encoding rules of ASN.1 International Standard 8825 [8], subject to the additional requirements imposed by the SNMP.

2.1. Format of Definitions

Section 4 contains contains the specification of all object types contained in this MIB module. The object types are defined using the conventions defined in the SMI, as amended by the extensions specified in RFC 1212 [9].

3. Overview

These objects are used when the particular media being used to realize an interface is a SIP interface. At present, this applies to these values of the ifType variable in the Internet-standard MIB:

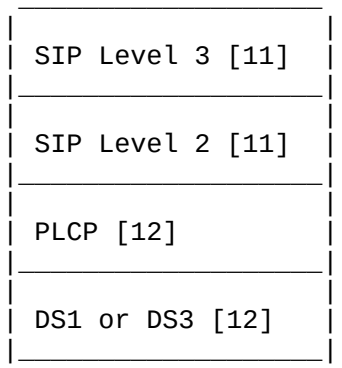
sip (31)

For these interfaces, the value of the ifSpecific variable in the MIB-II [6] has the OBJECT IDENTIFIER value:

```
sip    OBJECT IDENTIFIER ::= { transmission 31 }
```

The definitions contained herein are based on the SIP specifications in Bellcore TR-TSV-000772 and TR-TSV-000773 [11,12].

The SIP (SMDS Interface Protocol) protocol stack is defined as follows in TR-TSV-000772 [11]:



The PLCP (Physical Layer Convergence Procedure) adapts the capabilities of the transmission system (DS1 or DS3 formats) to the service expected by SIP Level 2. Managed objects for DS1 and DS3 Interface Types are defined in RFC 1232 [13] and RFC 1233 [14] respectively (and amended in RFC 1239 [17]), and can be utilized for management of SIP interfaces. This document defines managed objects for the remaining protocol levels of the SIP Interface Type. This document does not specify objects for the management of subscription

or configuration of Subscriber-Network Interfaces (SNIs). Those objects are defined in Definitions of Managed Objects for SMDS Subscription [18]. Bellcore requirements on these objects are specified in TA-TSV-001062 [16].

4. Object Definitions

```
RFC1304-MIB DEFINITIONS ::= BEGIN

IMPORTS
        Counter, TimeTicks, IpAddress
                FROM RFC1155-SMI
        transmission
                FROM RFC1213-MIB
        OBJECT-TYPE
                FROM RFC-1212;

--  This MIB module uses the extended OBJECT-TYPE macro
--  as defined in RFC-1212.

--  This is the MIB module for the SIP objects.


sip     OBJECT IDENTIFIER ::= { transmission 31 }

-- All representations of SMDS addresses in this MIB
-- module use, as a textual convention (i.e., this
-- convention does not affect their encoding), the
-- data type:

SMDSAddress ::= OCTET STRING (SIZE (8))
-- the 60-bit SMDS address, preceded by 4 bits with the
-- following values:
-- "1100" when representing an individual address
-- "1110" when representing a group address


-- The SIP Level 3 group
-- Implementation of the SIP Level 3 group is mandatory
-- for all systems implementing SIP Level 3.

sipL3Table  OBJECT-TYPE
    SYNTAX  SEQUENCE OF SipL3Entry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
            "This table contains SIP L3 parameters and
            state variables, one entry per SIP port."
```

```
        ::= { sip 1 }

    sipL3Entry  OBJECT-TYPE
        SYNTAX  SipL3Entry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "This list contains SIP L3 parameters and
                state variables."
       INDEX   { sipL3Index }
       ::= { sipL3Table 1 }

   SipL3Entry  ::= SEQUENCE {
       sipL3Index
           INTEGER,
       sipL3ReceivedIndividualDAs
           Counter,
       sipL3ReceivedGAs
           Counter,
       sipL3UnrecognizedIndividualDAs
           Counter,
       sipL3UnrecognizedGAs
           Counter,
       sipL3SentIndividualDAs
           Counter,
       sipL3SentGAs
           Counter,
       sipL3Errors
           Counter,
       sipL3InvalidSMDSAddressTypes
           Counter,
       sipL3VersionSupport
           INTEGER
       }

   sipL3Index  OBJECT-TYPE
       SYNTAX  INTEGER (1..65535)
       ACCESS  read-only
       STATUS  mandatory
       DESCRIPTION
               "The value of this object identifies the SIP
               port interface for which this entry contains
               management information.  The value of this
               object for a particular interface has the same
               value as the ifIndex object, defined in RFC
               1156 and RFC 1213, for the same interface."
      ::= { sipL3Entry 1 }
```

```
        sipL3ReceivedIndividualDAs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The total number of individually addressed SIP
                    Level 3 PDUs received from the remote system
                    across the SNI.  The total includes only
                    unerrored L3PDUs."
            ::= { sipL3Entry 2 }

        sipL3ReceivedGAs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The total number of group addressed SIP Level 3
                    PDUs received from the remote system across the
                    SNI.  The total includes only unerrored L3PDUs."
            ::= { sipL3Entry 3 }

        sipL3UnrecognizedIndividualDAs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of SIP Level 3 PDUs received from the
                    remote system with invalid or unknown individual
                    destination addresses (Destination Address
                    Screening violations are not included).  See SMDS
                    Subscription MIB module."
            ::= { sipL3Entry 4 }

        sipL3UnrecognizedGAs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of SIP Level 3 PDUs received from the
                    remote system with invalid or unknown group
                    addresses.  (Destination Address Screening
                    violations are not included).  See SMDS
                    Subscription MIB module."
            ::= { sipL3Entry 5 }

        sipL3SentIndividualDAs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
```

```
            STATUS  mandatory
            DESCRIPTION
                    "The number of individually addressed SIP Level 3
                    PDUs that have been sent by this system across the
                    SNI."
            ::= { sipL3Entry 6 }

        sipL3SentGAs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of group addressed SIP L3PDUs that
                    have been sent by this system across the SNI."
            ::= { sipL3Entry 7 }

        -- The total number of SIP L3PDU errors can be calculated as
        -- (Syntactic errors + Semantic Service errors )
        -- Syntactic errors include:
        --    sipL3Errors
        -- Latest occurrences of syntactic error types are logged in
        --    sipL3PDUErrorTable.
        -- Semantic Service errors include:
        --    sipL3UnrecognizedIndividualDAs
        --    sipL3UnrecognizedGAs
        --    sipL3InvalidSMDSAddressTypes
        -- Note that public networks supporting SMDS may discard
        -- SIP L3PDUs due to subscription violations.  Related
        -- managed objects are defined in Definitions of Managed
        -- Objects for SMDS Subscription.


        sipL3Errors OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The total number of SIP Level 3 PDUs received
                    from the remote system that were discovered to
                    have errors (including protocol processing and bit
                    errors but excluding addressing-related errors)
                    and were discarded.  Includes both group addressed
                    L3PDUs and L3PDUs containing an individual
                    destination address."
            ::= { sipL3Entry 8 }
```

```
        sipL3InvalidSMDSAddressTypes OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of SIP Level 3 PDUs received from the
                    remote system that had the Source or Destination
                    Address_Type subfields, (the four most significant
                    bits of the 64 bit address field), not equal to
                    the value 1100 or 1110.  Also, an error is
                    considered to have occurred if the Address_Type
                    field for a Source Address, the four most
                    significant bits of the 64 bits, is equal to 1110
                    (a group address)."
            ::= { sipL3Entry 9 }

        sipL3VersionSupport  OBJECT-TYPE
            SYNTAX  INTEGER (1..65535)
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "A value which indicates the version(s) of SIP
                    that this interface supports.  The value is a sum.
                    This sum initially takes the value zero.  For each
                    version, V, that this interface supports, 2 raised
                    to (V - 1) is added to the sum. For example, a
                    port supporting versions 1 and 2 would have a
                    value of (2^(1-1)+2^(2-1))=3.  The
                    sipL3VersionSupport is effectively a bit mask with
                    Version 1 equal to the least significant bit
                    (LSB)."
            ::= { sipL3Entry 10 }


        -- The SIP Level 2 group
        -- Implementation of the SIP Level 2 group is mandatory
        -- for all systems implementing SIP Level 2.


        sipL2Table  OBJECT-TYPE
            SYNTAX  SEQUENCE OF SipL2Entry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "This table contains SIP L2PDU parameters and
                    state variables, one entry per SIP port."
            ::= { sip 2 }
```

```
        sipL2Entry  OBJECT-TYPE
            SYNTAX  SipL2Entry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "This list contains SIP L2 parameters and state
                    variables."
            INDEX   { sipL2Index }
            ::= { sipL2Table 1 }

        SipL2Entry  ::= SEQUENCE {
            sipL2Index
                INTEGER,
            sipL2ReceivedCounts
                Counter,
            sipL2SentCounts
                Counter,
            sipL2HcsOrCRCErrors
                Counter,
            sipL2PayloadLengthErrors
                Counter,
            sipL2SequenceNumberErrors
                Counter,
            sipL2MidCurrentlyActiveErrors
                Counter,
            sipL2BomOrSSMsMIDErrors
                Counter,
            sipL2EomsMIDErrors
                Counter
            }

        sipL2Index  OBJECT-TYPE
            SYNTAX  INTEGER (1..65535)
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The value of this object identifies the SIP port
                    interface for which this entry contains management
                    information.  The value of this object for a
                    particular interface has the same value as the
                    ifIndex object, defined in RFC 1156 and RFC 1213,
                    for the same interface."
            ::= { sipL2Entry 1 }

        sipL2ReceivedCounts OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
```

```
            DESCRIPTION
                    "The number of SIP Level 2 PDUs received from the
                    remote system across the SNI. The total includes
                    only unerrored L2PDUs."
            ::= { sipL2Entry 2 }

        sipL2SentCounts OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of SIP Level 2 PDUs that have been
                    sent by this system across the SNI."
            ::= { sipL2Entry 3 }

        -- The total number of SIP L2PDU errors can be calculated as
        -- the sum of:
        --    sipL2HcsOrCRCErrors
        --    sipL2PayloadLengthErrors
        --    sipL2SequenceNumberErrors
        --    sipL2MidCurrentlyActiveErrors
        --    sipL2BomOrSSMsMIDErrors
        --    sipL2EomsMIDErrors

        sipL2HcsOrCRCErrors  OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of received SIP Level 2 PDUs that were
                    discovered to have either a Header Check Sequence
                    error or a Payload CRC violation."
            ::= { sipL2Entry 4 }

        sipL2PayloadLengthErrors  OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The number of received SIP Level 2 PDUs that had
                    Payload Length errors that fall in the following
                    specifications:
                    - SSM L2_PDU payload length field value less
                    - than 28 octets or greater than 44 octets,

                    - BOM or COM L2_PDU payload length field not
                    - equal to 44 octets,
```

```
                - EOM L2_PDU payload length field value less
                - than 4 octets or greater than 44 octets."
        ::= { sipL2Entry 5 }

    sipL2SequenceNumberErrors  OBJECT-TYPE
        SYNTAX  Counter
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
                "The number of received SIP Level 2 PDUs that had
                a sequence number within the L2PDU not equal to
                the expected sequence number of the SMDS SS
                receive process."
        ::= { sipL2Entry 6 }

    sipL2MidCurrentlyActiveErrors  OBJECT-TYPE
        SYNTAX Counter
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
                "The number of received SIP Level 2 PDUs that are
                BOMs for which an active receive process is
                already started."
        ::= { sipL2Entry 7 }

    sipL2BomOrSSMsMIDErrors  OBJECT-TYPE
        SYNTAX  Counter
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
                "The number of received SIP Level 2 PDUs that are
                SSMs with a MID not equal to zero or are BOMs with
                MIDs equal to zero."
        ::= { sipL2Entry 8 }

    sipL2EomsMIDErrors  OBJECT-TYPE
        SYNTAX  Counter
        ACCESS  read-only
        STATUS  mandatory
        DESCRIPTION
                "The number of received SIP Level 2 PDUs that are
                EOMs for which there is no active receive process
                for the MID (i.e., the receipt of an EOM which
                does not correspond to a BOM) OR the EOM has a MID
                equal to zero."
        ::= { sipL2Entry 9 }
```

```
        -- The SIP PLCP group
        -- Implementation of one of these groups is mandatory
        -- if the PLCP is implemented.

        sipPLCP OBJECT IDENTIFIER ::= { sip 3 }


        -- The SIP DS1 PLCP group
        -- Implementation of this group is mandatory
        -- if the DS1 PLCP is implemented.

        sipDS1PLCPTable  OBJECT-TYPE
            SYNTAX  SEQUENCE OF SipDS1PLCPEntry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "This table contains SIP DS1 PLCP parameters and
                    state variables, one entry per SIP port."
            ::= { sipPLCP 1 }

        sipDS1PLCPEntry  OBJECT-TYPE
            SYNTAX  SipDS1PLCPEntry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "This list contains SIP DS1 PLCP parameters and
                    state variables."
            INDEX   { sipDS1PLCPIndex }
            ::= { sipDS1PLCPTable 1 }

        SipDS1PLCPEntry  ::= SEQUENCE {
            sipDS1PLCPIndex
                INTEGER,
            sipDS1PLCPSEFSs
                Counter,
            sipDS1PLCPAlarmState
                INTEGER,
            sipDS1PLCPUASs
                Counter
            }


        sipDS1PLCPIndex  OBJECT-TYPE
            SYNTAX  INTEGER (1..65535)
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The value of this object identifies the SIP port
```

```
                    interface for which this entry contains management
                    information.  The value of this object for a
                    particular interface has the same value as the
                    ifIndex object, defined in RFC 1156 and RFC 1213,
                    for the same interface."
                ::= { sipDS1PLCPEntry 1 }


            sipDS1PLCPSEFSs OBJECT-TYPE
                SYNTAX  Counter
                ACCESS  read-only
                STATUS  mandatory
                DESCRIPTION
                        "A DS1 Severely Errored Framing Second (SEFS) is a
                        count of one-second intervals containing one or
                        more SEF events.  A Severely Errored Framing (SEF)
                        event is declared when an error in the A1 octet
                        and an error in the A2 octet of a framing octet
                        pair (i.e., errors in both framing octets), or two
                        consecutive invalid and/or nonsequential Path
                        Overhead Identifier octets are detected."
                ::= { sipDS1PLCPEntry 2 }

            sipDS1PLCPAlarmState OBJECT-TYPE
                SYNTAX  INTEGER {
                            noAlarm (1),
                            receivedFarEndAlarm (2),
                            incomingLOF (3)
                        }
                ACCESS  read-only
                STATUS  mandatory
                DESCRIPTION
                        "This variable indicates if there is an alarm
                        present for the DS1 PLCP.  The value
                        receivedFarEndAlarm means that the DS1 PLCP has
                        received an incoming Yellow Signal, the value
                        incomingLOF means that the DS1 PLCP has declared a
                        loss of frame (LOF) failure condition, and the
                        value noAlarm means that there are no alarms
                        present.  See TR-TSV-000773 for a description of
                        alarm states."
                ::= { sipDS1PLCPEntry 3 }


            sipDS1PLCPUASs OBJECT-TYPE
                SYNTAX  Counter
                ACCESS  read-only
                STATUS  mandatory
```

```
            DESCRIPTION
                    "The counter associated with the number of
                    Unavailable Seconds, as defined by TR-TSV-000773,
                    encountered by the PLCP."
            ::= { sipDS1PLCPEntry 4 }


        -- The SIP DS3 PLCP group
        -- Implementation of this group is mandatory
        -- if the DS3 PLCP is implemented.

        sipDS3PLCPTable  OBJECT-TYPE
            SYNTAX  SEQUENCE OF SipDS3PLCPEntry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "This table contains SIP DS3 PLCP parameters and
                    state variables, one entry per SIP port."
            ::= { sipPLCP 2 }

        sipDS3PLCPEntry  OBJECT-TYPE
            SYNTAX  SipDS3PLCPEntry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "This list contains SIP DS3 PLCP parameters and
                    state variables."
            INDEX   { sipDS3PLCPIndex }
            ::= { sipDS3PLCPTable 1 }

        SipDS3PLCPEntry  ::= SEQUENCE {
            sipDS3PLCPIndex
                INTEGER,
            sipDS3PLCPSEFSs
                Counter,
            sipDS3PLCPAlarmState
                INTEGER,
            sipDS3PLCPUASs
                Counter
            }


        sipDS3PLCPIndex  OBJECT-TYPE
            SYNTAX  INTEGER (1..65535)
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The value of this object identifies the SIP port
```

```
                    interface for which this entry contains management
                    information.  The value of this object for a
                    particular interface has the same value as the
                    ifIndex object, defined in RFC 1156 and RFC 1213,
                    for the same interface."
            ::= { sipDS3PLCPEntry 1 }

        sipDS3PLCPSEFSs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "A DS3 Severely Errored Framing Second (SEFS) is a
                    count of one-second intervals containing one or
                    more SEF events.  A Severely Errored Framing (SEF)
                    event is declared when an error in the A1 octet
                    and an error in the A2 octet of a framing octet
                    pair (i.e., errors in both framing octets), or two
                    consecutive invalid and/or nonsequential Path
                    Overhead Identifier octets are detected."
            ::= { sipDS3PLCPEntry 2 }

        sipDS3PLCPAlarmState OBJECT-TYPE
            SYNTAX  INTEGER {
                        noAlarm (1),
                        receivedFarEndAlarm (2),
                        incomingLOF (3)
                    }
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "This variable indicates if there is an alarm
                    present for the DS3 PLCP.  The value
                    receivedFarEndAlarm means that the DS3 PLCP has
                    received an incoming Yellow Signal, the value
                    incomingLOF means that the DS3 PLCP has declared a
                    loss of frame (LOF) failure condition, and the
                    value noAlarm means that there are no alarms
                    present.  See TR-TSV-000773 for a description of
                    alarm states."
            ::= { sipDS3PLCPEntry 3 }


        sipDS3PLCPUASs OBJECT-TYPE
            SYNTAX  Counter
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
```

```
                "The counter associated with the number of
                Unavailable Seconds, as defined by TR-TSV-000773,
                encountered by the PLCP."
        ::= { sipDS3PLCPEntry 4 }


    -- The SMDS Applications group
    -- Applications that have been identified for this group are:
    --          * IP-over-SMDS (details are specified in RFC 1209)
    -- Implementation of this group is mandatory for systems
    -- that implement IP-over-SMDS Interface Protocol.

    smdsApplications OBJECT IDENTIFIER ::= { sip 4 }

    ipOverSMDS OBJECT IDENTIFIER ::= { smdsApplications 1 }

    -- Although the objects in this group are read-only, at the
    -- agent's discretion they may be made read-write so that the
    -- management station, when appropriately authorized, may
    -- change the addressing information related to the
    -- configuration of a logical IP subnetwork implemented on
    -- top of SMDS.

    -- This table is necessary to support RFC1209 (IP-over-SMDS)
    -- and gives information on the Group Addresses and ARP
    -- Addresses used in the Logical IP subnetwork.
    -- One SMDS address may be associated with multiple IP
    -- addresses.  One SNI may be associated with multiple LISs.

    ipOverSMDSTable OBJECT-TYPE
        SYNTAX  SEQUENCE OF IpOverSMDSEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "The table of addressing information relevant to
                this entity's IP addresses."
        ::= { ipOverSMDS 1 }

    ipOverSMDSEntry OBJECT-TYPE
        SYNTAX  IpOverSMDSEntry
        ACCESS  not-accessible
        STATUS  mandatory
        DESCRIPTION
                "The addressing information for one of this
                entity's IP addresses."
        INDEX   { ipOverSMDSIndex, ipOverSMDSAddress }
        ::= { ipOverSMDSTable 1 }
```

```
          IpOverSMDSEntry ::=
              SEQUENCE {
                 ipOverSMDSIndex
                     INTEGER,
                 ipOverSMDSAddress
                     IpAddress,
                 ipOverSMDSHA
                     SMDSAddress,
                 ipOverSMDSLISGA
                     SMDSAddress,
                 ipOverSMDSARPReq
                     SMDSAddress
                 }

          ipOverSMDSIndex OBJECT-TYPE
              SYNTAX  INTEGER (1..65535)
              ACCESS  read-only
              STATUS  mandatory
              DESCRIPTION
                      "The value of this object identifies the SIP port
                      interface for which this entry contains management
                      information.  The value of this object for a
                      particular interface has the same value as the
                      ifIndex object, defined in RFC 1156 and RFC 1213,
                      for the same interface."
              ::= { ipOverSMDSEntry 1 }

          ipOverSMDSAddress OBJECT-TYPE
               SYNTAX  IpAddress
               ACCESS  read-only
               STATUS  mandatory
               DESCRIPTION
                      "The IP address to which this entry's addressing
                      information pertains."
              ::= { ipOverSMDSEntry 2 }

          ipOverSMDSHA OBJECT-TYPE
              SYNTAX  SMDSAddress
              ACCESS  read-only
              STATUS  mandatory
              DESCRIPTION
                      "The SMDS Individual address of the IP station."
              ::= { ipOverSMDSEntry 3 }

          ipOverSMDSLISGA OBJECT-TYPE
              SYNTAX  SMDSAddress
              ACCESS  read-only
              STATUS  mandatory
```

```
            DESCRIPTION
                    "The SMDS Group Address that has been configured
                    to identify the SMDS Subscriber-Network Interfaces
                    (SNIs) of all members of the Logical IP Subnetwork
                    (LIS) connected to the network supporting SMDS."
            ::= { ipOverSMDSEntry 4 }

        ipOverSMDSARPReq OBJECT-TYPE
            SYNTAX  SMDSAddress
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The SMDS address (individual or group) to which
                    ARP Requests are to be sent."
            ::= { ipOverSMDSEntry 5 }


        -- The SMDS Carrier Selection group
        -- This group is used as a place holder
        -- for carrier selection objects.

        smdsCarrierSelection OBJECT IDENTIFIER ::= { sip 5}


        -- The SIP Error Log
        -- Implementation of this group is mandatory
        -- for all systems that implement SIP Level 3.

        sipErrorLog OBJECT IDENTIFIER ::= { sip 6 }

        sipL3PDUErrorTable OBJECT-TYPE
            SYNTAX  SEQUENCE OF SipL3PDUErrorEntry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "A table that contains the latest occurrence of
                    the following syntactical SIP L3PDU errors:

                    - Destination Address Field Format Error,

                    The following pertains to the 60 least significant
                    bits of the 64 bit address field.  The 60 bits
                    contained in the address subfield can be used to
                    represent addresses up to 15 decimal digits.  Each
                    decimal digit shall be encoded into four bits
                    using Binary Coded Decimal (BCD), with the most
                    significant digit occurring left-most.  If not all
                    15 digits are required, then the remainder of this
```

field shall be padded on the right with bits set to one. An error is considered to have occurred: a). if the first four bits of the address subfield are not BCD, OR b). if the first four bits of the address subfield are populated with the country code value 0001, AND the 40 bits which follow are not Binary Coded Decimal (BCD) encoded values of the 10 digit addresses, OR the remaining 16 least significant bits are not populated with 1's, OR c). if the address subfield is not correct according to another numbering plan which is dependent upon the carrier assigning the numbers and offering SMDS.

- Source Address Field Format Error,

The description of this parameter is the same as the description of the Destination Address Field Format Error.

- Invalid BAsize Field Value,

An error is considered to have occurred when the BAsize field of an SIP L3PDU contains a value less that 32, greater than 9220 octets without the CRC32 field present, greater than 9224 octets with the CRC32 field present, or not equal to a multiple of 4 octets,

- Invalid Header Extension Length Field Value,

An error is considered to have occurred when the Header Extension Length field value is not equal 3.

- Invalid Header Extension - Element Length,

An error is considered to have occurred when the Header Extension - Element Length is greater than 12.

- Invalid Header Extension - Version Element Position, Length, or Value,

An error is considered to have occurred when a Version element with Length=3, Type=0, and Value=1 does not appear first within the Header Extension, or an element Type=0 appears somewhere other than

within the first three octets in the Header Extension.

- Invalid Header Extension - Carrier Selection Element Position, Length, Value or Format,

An error is considered to have occurred when a Carrier Selection element does not appear second within the Header Extension, if the Element Type does not equal 1, the Element Length does not equal 4, 6, or 8, the Element Value field is not four BCD encoded decimal digits used in specifying the Carrier Identification Code (CIC), or the identified CIC code is invalid.

- Header Extension PAD Error

An error is considered to have occurred when the Header Extension PAD is 9 octets in length, or if the Header Extension PAD is greater than zero octets in length and the Header Extension PAD does not follow all Header Extension elements or does not begin with at least one octet of all zeros.

- BEtag Mismatch Error,

An error is considered to have occurred when the Beginning-End Tags in the SIP L3PDU header and trailer are not equal.

- BAsize Field not equal to Length Field Error,

An error is considered to have occurred when the value of the BAsize Field does not equal the value of the Length Field.

- Incorrect Length Error, and

An error is considered to have occurred when the the Length field value is not equal to the portion of the SIP L3PDU which extends from the Destination Address field up to and including the CRC32 field (if present) or up to and including the PAD field (if the CRC32 field is not present). As an optional check, an error is considered to have occurred when the length of a partially received SIP L3PDU exceeds the BAsize value.

```
                    - MRI Timeout Error.

                    An error is considered to have occurred when the
                    elapsed time between receipt of BOM and
                    corresponding EOM exceeds the value of the MRI
                    (Message Receive Interval) for a particular
                    transport signal format.

                    An entry is indexed by interface number and error
                    type, and contains Source Address, Destination
                    Address and a timestamp. All these errors are
                    counted in the sipL3Errors counter.  When
                    sipL3PDUErrorTimeStamp is equal to zero, the
                    SipL3PDUErrorEntry does not contain any valid
                    information."
            ::= { sipErrorLog 1 }

        sipL3PDUErrorEntry  OBJECT-TYPE
            SYNTAX SipL3PDUErrorEntry
            ACCESS  not-accessible
            STATUS  mandatory
            DESCRIPTION
                    "An entry in the service disagreement table."
            INDEX   { sipL3PDUErrorIndex, sipL3PDUErrorType }
            ::= { sipL3PDUErrorTable 1 }

        SipL3PDUErrorEntry ::= SEQUENCE {
            sipL3PDUErrorIndex
                INTEGER,
            sipL3PDUErrorType
                INTEGER,
            sipL3PDUErrorSA
                SMDSAddress,
            sipL3PDUErrorDA
                SMDSAddress,
            sipL3PDUErrorTimeStamp
                TimeTicks
            }

        sipL3PDUErrorIndex OBJECT-TYPE
            SYNTAX  INTEGER (1..65535)
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The value of this object identifies the SIP port
                    interface for which this entry contains management
                    information.   The value of this object for a
                    particular interface has the same value as the
```

```
                    ifIndex object, defined in RFC 1156 and RFC 1213,
                    for the same interface."
            ::= { sipL3PDUErrorEntry 1 }

        sipL3PDUErrorType OBJECT-TYPE
            SYNTAX  INTEGER {
                 erroredDAFieldFormat (1),
                 erroredSAFieldFormat (2),
                 invalidBAsizeFieldValue (3),
                 invalidHdrExtLength (4),
                 invalidHdrExtElementLength (5),
                 invalidHdrExtVersionElementPositionLenthOrValue (6),
          invalidHdrExtCarSelectElementPositionLenghtValueOrFormat (7),
                 hePADError (8),
                 beTagMismatch (9),
                 baSizeFieldNotEqualToLengthField (10),
                 incorrectLength (11),
                 mriTimeout (12)
            }
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The type of error."
            ::= { sipL3PDUErrorEntry 2 }

        sipL3PDUErrorSA OBJECT-TYPE
            SYNTAX  SMDSAddress
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "A rejected SMDS source address."
            ::= { sipL3PDUErrorEntry 3 }

        sipL3PDUErrorDA OBJECT-TYPE
            SYNTAX  SMDSAddress
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "A rejected SMDS destination address."
            ::= { sipL3PDUErrorEntry 4 }

        sipL3PDUErrorTimeStamp OBJECT-TYPE
            SYNTAX  TimeTicks
            ACCESS  read-only
            STATUS  mandatory
            DESCRIPTION
                    "The timestamp for the service disagreement.  The
                    timestamp contains the value of sysUpTime at the
```

```
                    latest occurrence of this type of service
                    disagreement.  See textual description under
                    sipL3PDUErrorTable for boundary conditions."
            ::= { sipL3PDUErrorEntry 5 }

        END
```

5. Acknowledgments

This document was produced by the SNMP Working Group. In addition, the comments of the following individuals are also acknowledged: Ted Brunner, Jeff Case, Tracy Cox, Sherri Hiller, Steve Jaffe, Deirdre Kostick, Dave Piscitello, and Ron Reuss.

6. References

[1] Cerf, V., "IAB Recommendations for the Development of Internet Network Management Standards", RFC 1052, NRI, April 1988.

[2] Cerf, V., "Report of the Second Ad Hoc Network Management Review Group", RFC 1109, NRI, August 1989.

[3] Rose M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based internets", RFC 1155, Performance Systems International, Hughes LAN Systems, May 1990.

[4] McCloghrie K., and M. Rose, "Management Information Base for Network Management of TCP/IP-based internets", RFC 1156, Hughes LAN Systems, Performance Systems International, May 1990.

[5] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", RFC 1157, SNMP Research, Performance Systems International, Performance Systems International, MIT Laboratory for Computer Science, May 1990.

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

[7] Information processing systems - Open Systems Interconnection - Specification of Abstract Syntax Notation One (ASN.1), International Organization for Standardization, International Standard 8824, December 1987.

[8] Information processing systems - Open Systems Interconnection - Specification of Basic Encoding Rules for Abstract Notation One (ASN.1), International Organization for Standardization, International Standard 8825, December 1987.

[9] Rose, M., and K. McCloghrie, Editors, "Concise MIB Definitions", RFC 1212, Performance Systems International, Hughes LAN Systems, March 1991.

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

[11] "Generic System Requirements in Support of Switched Multi-megabit Data Service", Bellcore Technical Reference, TR-TSV-000772, Issue 1, May 1991.

[12] "Local Access System Generic Requirements, Objectives, and Interfaces in Support of Switched Multi-megabit Data Service", Bellcore Technical Reference, TR-TSV-000773, Issue 1, June 1990.

[13] Baker F., and C. Kolb, Editors, "Definitions of Managed Objects for the DS1 Interface Type", RFC 1232, ACC, Performance Systems International, Inc., May 1991.

[14] Cox, T., and K. Tesink, Editors, "Definitions of Managed Objects for the DS3 Interface Type", RFC 1233, Bell Communications Research, May 1991.

[15] Piscitello, D., and J. Lawrence, Editors, The Transmission of IP Datagrams over the SMDS Service", RFC 1209, Bell Communications Research, March 1991.

[16] "Generic Requirements For SMDS Customer Network Management Service", TA-TSV-001062, Issue 1, February 1991, and Supplement 1, April 1991.

[17] Reynolds, J., "Reassignment of Experimental MIBs to Standard MIBs", RFC 1239, USC/Information Sciences Institute, June 1991.

[18] Tesink, K., "Definitions of Managed Objects for SMDS Subscription", Version 1.0, Bellcore, March 1991.

7. Security Considerations

Security issues are not discussed in this memo.

8. Authors' Addresses

Tracy A. Cox
Bell Communications Research
331 Newman Springs Road
Red Bank, NJ 07701

Phone: (908) 758-2107
EMail: tacox@sabre.bellcore.com

Kaj Tesink
Bell Communications Research
331 Newman Springs Road
Red Bank, NJ 07701

Phone: (908) 758-5254
EMail: kaj@nvuxr.cc.bellcore.com