Internet Engineering Task Force   J. Schoenwaelder
Internet-Draft   A. Sehgal
Intended status: Standards Track   Jacobs University
Expires: August 10, 2013   T. Tsou
                                             Huawei Technologies (USA)
                                                              C. Zhou
                                                   Huawei Technologies
                                                     February 6, 2013

Definition of Managed Objects for IPv6 over Low-Power Wireless Personal
                       Area Networks (6LoWPANs)
                      draft-schoenw-6lowpan-mib-03

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 defines objects for managing IPv6 over Low-Power
   Wireless Personal Area Networks (6LoWPANs).

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   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."

   This Internet-Draft will expire on August 10, 2013.

Table of Contents

1. Introduction

   This memo defines a portion of the Management Information Base (MIB)
   for use with network management protocols. In particular it defines
   objects for managing IPv6 over Low-Power Wireless Personal Area
   Networks (6LoWPANs) [RFC4944].

2. The Internet-Standard Management Framework

   For a detailed overview of the documents that describe the current
   Internet-Standard Management Framework, please refer to section 7 of
   RFC 3410 [RFC3410].

   Managed objects are accessed via a virtual information store, termed
   the Management Information Base or MIB. MIB objects are generally
   accessed through the Simple Network Management Protocol (SNMP).
   Objects in the MIB are defined using the mechanisms defined in the
   Structure of Management Information (SMI). This memo specifies a MIB
   module that is compliant to the SMIv2, which is described in STD 58,
   RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580
   [RFC2580].

3. Conventions

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
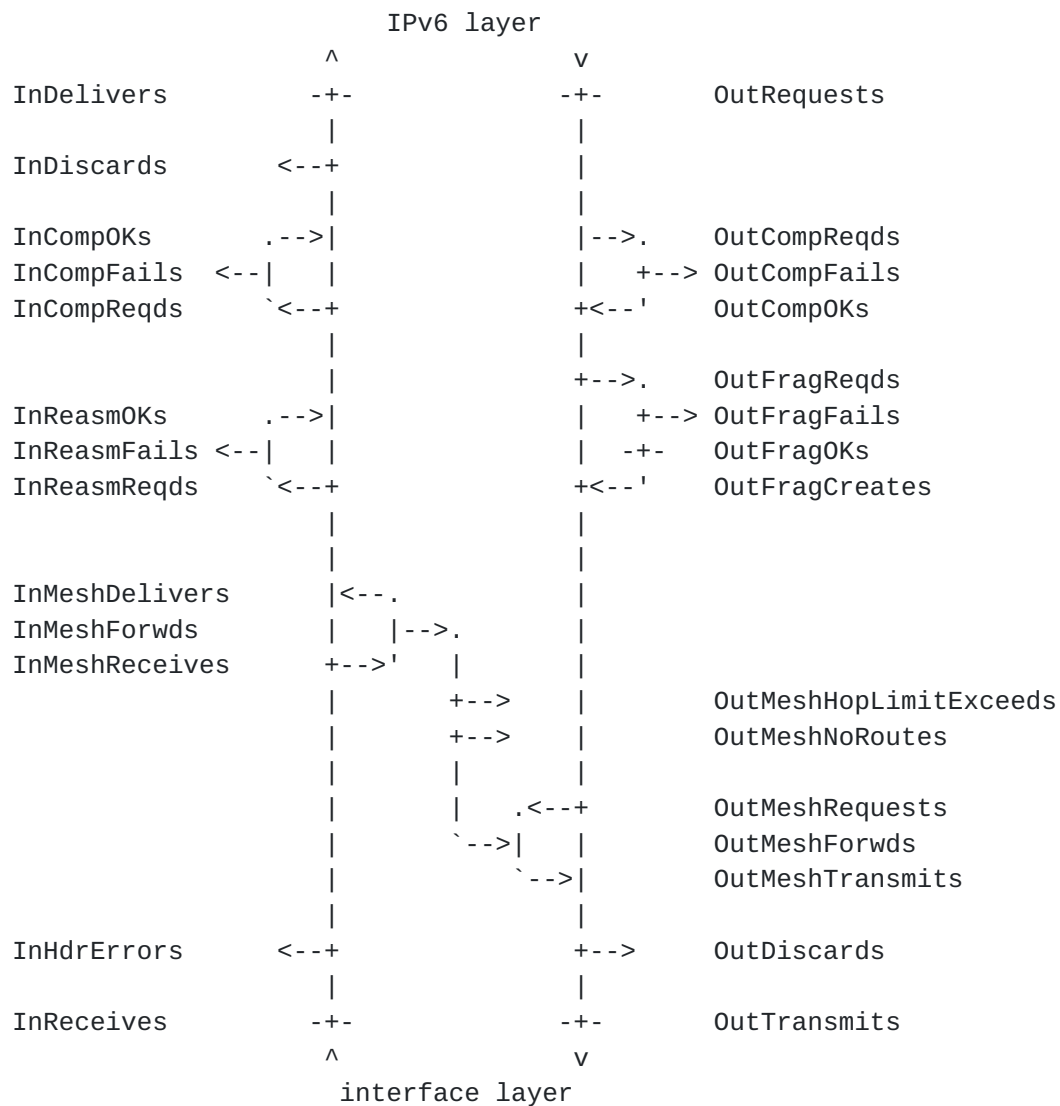   document are to be interpreted as described in RFC 2119 [RFC2119].

4. Overview

   The MIB module is organized into groups of scalars and tables.

```
# LOWPAN-MIB registration tree (generated by smidump 0.4.8)

--lowpanMIB(1.3.6.1.2.1.XXXX)
  +--lowpanNotifications(0)
  +--lowpanObjects(1)
  |  +-- r-n Unsigned32 lowpanReasmTimeout(1)
  |  +-- r-n Counter32  lowpanInReceives(2)
  |  +-- r-n Counter32  lowpanInHdrErrors(3)
  |  +-- r-n Counter32  lowpanInMeshReceives(4)
  |  +-- r-n Counter32  lowpanInMeshForwds(5)
  |  +-- r-n Counter32  lowpanInMeshDelivers(6)
  |  +-- r-n Counter32  lowpanInReasmReqds(7)
  |  +-- r-n Counter32  lowpanInReasmFails(8)
  |  +-- r-n Counter32  lowpanInReasmOKs(9)
  |  +-- r-n Counter32  lowpanInCompReqds(10)
  |  +-- r-n Counter32  lowpanInCompFails(11)
  |  +-- r-n Counter32  lowpanInCompOKs(12)
  |  +-- r-n Counter32  lowpanInDiscards(13)
  |  +-- r-n Counter32  lowpanInDelivers(14)
  |  +-- r-n Counter32  lowpanOutRequests(15)
  |  +-- r-n Counter32  lowpanOutCompReqds(16)
  |  +-- r-n Counter32  lowpanOutCompFails(17)
  |  +-- r-n Counter32  lowpanOutCompOKs(18)
  |  +-- r-n Counter32  lowpanOutFragReqds(19)
  |  +-- r-n Counter32  lowpanOutFragFails(20)
  |  +-- r-n Counter32  lowpanOutFragOKs(21)
  |  +-- r-n Counter32  lowpanOutFragCreates(22)
  |  +-- r-n Counter32  lowpanOutMeshHopLimitExceeds(23)
  |  +-- r-n Counter32  lowpanOutMeshNoRoutes(24)
  |  +-- r-n Counter32  lowpanOutMeshRequests(25)
  |  +-- r-n Counter32  lowpanOutMeshForwds(26)
  |  +-- r-n Counter32  lowpanOutMeshTransmits(27)
  |  +-- r-n Counter32  lowpanOutDiscards(28)
  |  +-- r-n Counter32  lowpanOutTransmits(29)
  +--lowpanConformance(2)
     +--lowpanGroups(1)
     |  +--lowpanCoreGroup(1)
     +--lowpanCompliances(2)
        +--lowpanCompliance(1)
```

The counters defined in the MIB module provide information about the
6LoWPAN datagrams received and transmitted and how they are processed
in the 6LoWPAN layer. For the purpose of this specification, a
6LoWPAN datagram is an IEEE 805.14.5 datagram with a dispatch byte
matching the bit patterns 01xxxxxx, 10xxxxxx, or 11xxxxxx. The
processing of IEEE 805.14.5 datagrams matching the bit pattern
00xxxxxx (NALP - not a LoWPAN frame) [RFC4944] is not considered by
this specification. Other radio technologies may use different

mechanisms to identify 6LoWPAN datagrams (e.g., the BLUETOOTH Low
Energy Logical Link Control and Adaptation Protocol uses Channel
Identifiers [I-D.ietf-6lowpan-btle]). The following Case diagram
illustrates the conceptual relationships of the counters.

```
                            IPv6 layer
                        ^                 v
    InDelivers         -+-               -+-        OutRequests
                        |                 |
    InDiscards     <--+                   |
                        |                 |
    InCompOKs       .-->|               |-->.    OutCompReqds
    InCompFails  <--|    |              |   +--> OutCompFails
    InCompReqds     `<--+               +<--'    OutCompOKs
                        |                 |
                        |               +-->.    OutFragReqds
    InReasmOKs      .-->|               |   +--> OutFragFails
    InReasmFails <--|    |              |  -+-    OutFragOKs
    InReasmReqds    `<--+               +<--'    OutFragCreates
                        |                 |
                        |                 |
    InMeshDelivers     |<--.             |
    InMeshForwds       |   |-->.         |
    InMeshReceives    +-->'   |          |
                        |       +-->    |        OutMeshHopLimitExceeds
                        |       +-->    |        OutMeshNoRoutes
                        |       |        |
                        |       |   .<--+        OutMeshRequests
                        |     `-->|   |          OutMeshForwds
                        |        `-->|           OutMeshTransmits
                        |                 |
    InHdrErrors    <--+               +-->        OutDiscards
                        |                 |
    InReceives         -+-               -+-        OutTransmits
                        ^                 v
                         interface layer
```

The fragmentation related counters have been modeled after the
fragmentation related counters of the IP-MIB [RFC4293]. The discard
counters have been placed at the end of the input and output chains
but they can be bumped any time if a datagram is discarded for a
reason not covered by the other counters.

The compression related counters provide insights into compression
requests and in particular also compression related failures. Note
that the diagram is conceptual in the sense that compression happens
after reassembly for incoming 6LoWPAN datagrams and compression
happens before fragmentation for outgoing 6LoWPAN datagrams.

Implementations may choose to implement things slightly differently.
For example, implementations may decompress FRAG1 fragments as soon
as they are received, not waiting for reassembly to complete.

The mesh header processing related counters do not have an explicit
discard counter. Implementations that do not support mesh forwarding
MUST count the number of received 6LoWPAN datagrams with a MESH
header (lowpanInMeshReceives) but they MUST NOT increment the
lowpanInMeshReceives and lowpanInMeshDelivers counters if these
6LoWPAN datagrams are dropped.

5. **Relationship to Other MIB Modules**

The MIB module IMPORTS definitions from SNMPv2-SMI [RFC2578] and
SNMPv2-CONF [RFC2580].

6. **Definitions**

LOWPAN-MIB DEFINITIONS ::= BEGIN

IMPORTS
    MODULE-IDENTITY, OBJECT-TYPE, Unsigned32, Counter32, mib-2
        FROM SNMPv2-SMI   -- RFC 2578
    OBJECT-GROUP, MODULE-COMPLIANCE
        FROM SNMPv2-CONF;   -- RFC 2580

lowpanMIB   MODULE-IDENTITY
    LAST-UPDATED   "201301090000Z"
    ORGANIZATION
        "Jacobs University Bremen"
    CONTACT-INFO
        "Juergen Schoenwaelder
         Jacobs University Bremen
         Email: j.schoenwaelder@jacobs-university.de

         Anuj Sehgal
         Jacobs University Bremen
         Email: s.anuj@jacobs-university.de

         Tina Tsou
         Huawei Technologies
         Email: tina.tsou.zouting@huawei.com

         Cathy Zhou
         Huawei Technologies
         Email: cathyzhou@huawei.com"
     DESCRIPTION

      REVISION "201301090000Z"
      DESCRIPTION
          "Initial version, published as RFC XXXX."
      -- RFC Ed.: replace XXXX with actual RFC number and remove this note

      ::= { mib-2 XXXX }

-- object definitions

lowpanNotifications   OBJECT IDENTIFIER ::= { lowpanMIB 0 }
lowpanObjects   OBJECT IDENTIFIER ::= { lowpanMIB 1 }
lowpanConformance   OBJECT IDENTIFIER ::= { lowpanMIB 2 }

lowpanReasmTimeout OBJECT-TYPE
    SYNTAX   Unsigned32
    UNITS   "seconds"
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
        "The maximum number of seconds that received fragments are
         held while they are awaiting reassembly at this entity."
    ::= { lowpanObjects 1 }

lowpanInReceives OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
        "The total number of 6LoWPAN datagrams received, including
         those received in error."
    ::= { lowpanObjects 2 }

lowpanInHdrErrors OBJECT-TYPE
    SYNTAX   Counter32

```
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of received 6LoWPAN datagrams discarded due to
         errors in their headers, including unknown dispatch values,
         errors discovered during any decompression attempts, etc."
    ::= { lowpanObjects 3 }

lowpanInMeshReceives OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of received 6LoWPAN datagrams with a MESH header."
    ::= { lowpanObjects 4 }

lowpanInMeshForwds OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of received 6LoWPAN datagrams requiring MESH
         forwarding."
    ::= { lowpanObjects 5 }

lowpanInMeshDelivers OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of received 6LoWPAN datagrams with a MESH header
         delivered to the local system."
    ::= { lowpanObjects 6 }

lowpanInReasmReqds OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of received 6LoWPAN fragments that needed to
         be reassembled. This includes both FRAG1 and FRAGN 6LoWPAN
         datagrams."
    ::= { lowpanObjects 7 }

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

    DESCRIPTION
        "The number of failures detected by the re-assembly algorithm
        (e.g., timeouts). Note that this is not necessarily a count of
        discarded 6LoWPAN fragments since implementations can lose
        track of the number of fragments by combining them as
        received."
    ::= { lowpanObjects 8 }

lowpanInReasmOKs OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
        "The number of IPv6 packets successfully reassembled."
    ::= { lowpanObjects 9 }

lowpanInCompReqds OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
        "The number of 6LoWPAN datagrams requiring header
         decompression."
    ::= { lowpanObjects 10 }

lowpanInCompFails OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
        "The number of 6LoWPAN datagrams where header decompression
         failed (e.g., because the necessary context information is
         not available)."
    ::= { lowpanObjects 11 }

lowpanInCompOKs OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current
    DESCRIPTION
        "The number of 6LoWPAN datagrams where header decompression
         was successful."
    ::= { lowpanObjects 12 }

lowpanInDiscards OBJECT-TYPE
    SYNTAX   Counter32
    MAX-ACCESS read-only
    STATUS   current

```
    DESCRIPTION
        "The number of received 6LoWPAN datagrams for which no
        problems were encountered to prevent their continued
        processing, but were discarded (e.g., for lack of buffer
        space).  Note that this counter does not include any
        datagrams discarded due to a reassembly failure."
    ::= { lowpanObjects 13 }

lowpanInDelivers OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The total number of IPv6 packets successfully delivered
         to the IPv6 layer."
    ::= { lowpanObjects 14 }

lowpanOutRequests OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The total number of IPv6 packets supplied by the IPv6 layer."
    ::= { lowpanObjects 15 }

lowpanOutCompReqds OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The total number of IPv6 packets for which header compression
         was attempted."
    ::= { lowpanObjects 16 }

lowpanOutCompFails OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The total number of IPv6 packets for which header compression
         failed (e.g., because the UDP checksum check failed while
         performing UDP header compression)."
    ::= { lowpanObjects 17 }

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

```
    DESCRIPTION
        "The total number of IPv6 packets for which header compression
         was successful."
    ::= { lowpanObjects 18 }


lowpanOutFragReqds OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The number of IPv6 packets that require fragmentation
         in order to be transmitted."
    ::= { lowpanObjects 19 }


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


lowpanOutFragOKs OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The number of IPv6 packets that have been successfully
         fragmented."
    ::= { lowpanObjects 21 }


lowpanOutFragCreates OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The number of 6LoWPAN fragments that have been
         generated as a result of fragmentation. This includes
         both FRAG1 and FRAGN 6LoWPAN datagrams."
    ::= { lowpanObjects 22 }


lowpanOutMeshHopLimitExceeds OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The number of 6LoWPAN datagrams with a MESH header that
```

```
        were dropped because the hop limit has been exceeded."
    ::= { lowpanObjects 23 }

lowpanOutMeshNoRoutes OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of 6LoWPAN datagrams with a MESH header that
         were dropped because there was no forwarding information
         available."
    ::= { lowpanObjects 24 }

lowpanOutMeshRequests OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of 6LoWPAN datagrams requiring MESH header
         encapsulation."
    ::= { lowpanObjects 25 }

lowpanOutMeshForwds OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of 6LoWPAN datagrams with a MESH header for
         which suitable forwarding information was available."
    ::= { lowpanObjects 26 }

lowpanOutMeshTransmits OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of 6LoWPAN datagrams with a MESH header
         created."
    ::= { lowpanObjects 27 }

lowpanOutDiscards OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS read-only
    STATUS      current
    DESCRIPTION
        "The number of IPv6 packets for which no problem was
         encountered to prevent their transmission to their
         destination, but were discarded (e.g., for lack of
```

```
        buffer space)."
    ::= { lowpanObjects 28 }

lowpanOutTransmits OBJECT-TYPE
    SYNTAX     Counter32
    MAX-ACCESS read-only
    STATUS     current
    DESCRIPTION
        "The total number of 6LoWPAN datagram that this entity
         supplied to the lower layers for transmission."
    ::= { lowpanObjects 29 }

-- conformance definitions

lowpanGroups      OBJECT IDENTIFIER ::= { lowpanConformance 1 }
lowpanCompliances OBJECT IDENTIFIER ::= { lowpanConformance 2 }

lowpanCompliance MODULE-COMPLIANCE
    STATUS     current
    DESCRIPTION
        "Compliance statement for systems that implement 6LoWPAN."
    MODULE      -- this module
    MANDATORY-GROUPS {
        lowpanCoreGroup
    }
    ::= { lowpanCompliances 1 }

lowpanCoreGroup OBJECT-GROUP
    OBJECTS {
        lowpanReasmTimeout,
        lowpanInReceives,
        lowpanInHdrErrors,
        lowpanInMeshReceives,
        lowpanInMeshForwds,
        lowpanInMeshDelivers,
        lowpanInReasmReqds,
        lowpanInReasmFails,
        lowpanInReasmOKs,
        lowpanInCompReqds,
        lowpanInCompFails,
        lowpanInCompOKs,
        lowpanInDiscards,
        lowpanInDelivers,
        lowpanOutRequests,
        lowpanOutCompReqds,
        lowpanOutCompFails,
        lowpanOutCompOKs,
        lowpanOutFragReqds,
```

```
        lowpanOutFragFails,
        lowpanOutFragOKs,
        lowpanOutFragCreates,
        lowpanOutMeshHopLimitExceeds,
        lowpanOutMeshNoRoutes,
        lowpanOutMeshRequests,
        lowpanOutMeshForwds,
        lowpanOutMeshTransmits,
        lowpanOutDiscards,
        lowpanOutTransmits
    }
    STATUS      current
    DESCRIPTION
        "A collection of objects providing information and
         statistics about the processing of 6LoWPAN datagrams."
    ::= { lowpanGroups 1 }

END
```

7. Security Considerations

   Some of the readable objects in this MIB module (i.e., objects with a
   MAX-ACCESS other than not-accessible) may be considered sensitive or
   vulnerable in some network environments. It is thus important to
   control even GET and/or NOTIFY access to these objects and possibly
   to even encrypt the values of these objects when sending them over
   the network via SNMP. These are the tables and objects and their
   sensitivity/vulnerability:

   The read-only counters provide insights into the amount of 6LoWPAN
   traffic a node is receiving or transmitting. This might provide
   information whether a device is regularly exchanging information with
   other devices or whether a device is mostly not participating in any
   communication (e.g., the device might be "easier" to take away
   unnoticed). The reassembly counters could be used to direct denial
   of service attacks on the reassembly mechanism.

   SNMP versions prior to SNMPv3 did not include adequate security.
   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 module.

   It is RECOMMENDED that implementers consider the security features as
   provided by the SNMPv3 framework (see [RFC3410], section 8),
   including full support for the SNMPv3 cryptographic mechanisms (for
   authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT
RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to
enable cryptographic security. It is then a customer/operator
responsibility to ensure that the SNMP entity giving access to an
instance of this MIB module 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.

8. IANA Considerations

IANA is requested to assign a value for "XXXX" under the 'mib-2'
subtree and to record the assignment in the SMI Numbers registry.
When the assignment has been made, the RFC Editor is asked to replace
"XXXX" (here and in the MIB module) with the assigned value and to
remove this note.

9. Acknowledgements

This specification borrows heavily from the IP-MIB defined in
[RFC4293].

Juergen Schoenwaelder and Anuj Sehgal were partly funded by Flamingo,
a Network of Excellence project (ICT-318488) supported by the
European Commission under its Seventh Framework Programme.

10. References

10.1. Normative References

[RFC2119]   Bradner, S., "Key words for use in
                              RFCs to Indicate Requirement Levels",
                              BCP 14, RFC 2119, March 1997.

[RFC2578]   McCloghrie, K., Ed., Perkins, D., Ed.,
                              and J. Schoenwaelder, Ed., "Structure
                              of Management Information Version 2
                              (SMIv2)", STD 58, RFC 2578,
                              April 1999.

[RFC2579]   McCloghrie, K., Ed., Perkins, D., Ed.,
                              and J. Schoenwaelder, Ed., "Textual
                              Conventions for SMIv2", STD 58,
                              RFC 2579, April 1999.

[RFC2580]   McCloghrie, K., Perkins, D., and J.
                              Schoenwaelder, "Conformance Statements
                              for SMIv2", STD 58, RFC 2580,
                              April 1999.

   [RFC4944]   Montenegro, G., Kushalnagar, N., Hui,
                                  J., and D. Culler, "Transmission of
                                  IPv6 Packets over IEEE 802.15.4
                                  Networks", RFC 4944, September 2007.

   [I-D.ietf-6lowpan-btle]   Nieminen, J., Savolainen, T., Isomaki,
                                  M., Patil, B., Shelby, Z., and C.
                                  Gomez, "Transmission of IPv6 Packets
                                  over BLUETOOTH Low Energy",
                                  draft-ietf-6lowpan-btle-11 (work in
                                  progress), October 2012.

10.2. Informative References

   [RFC3410]   Case, J., Mundy, R., Partain, D., and
                                  B. Stewart, "Introduction and
                                  Applicability Statements for Internet-
                                  Standard Management Framework",
                                  RFC 3410, December 2002.

   [RFC4293]   Routhier, S., "Management Information
                                  Base for the Internet Protocol (IP)",
                                  RFC 4293, April 2006.

   [RFC6643]   Schoenwaelder, J., "Translation of
                                  Structure of Management Information
                                  Version 2 (SMIv2) MIB Modules to YANG
                                  Modules", RFC 6643, July 2012.

   [I-D.lhotka-netmod-yang-json] Lhotka, L., "Modeling JSON Text with
                                  YANG",
                                  draft-lhotka-netmod-yang-json-00 (work
                                  in progress), October 2012.

Appendix A. JSON Representation

   Using the translation algorithm defined in [RFC6643], the SMIv2
   module can be translated to YANG. Using the JSON representation of
   data modeled in YANG defined in [I-D.lhotka-netmod-yang-json], the
   objects defined in the MIB module can be represented in JSON as shown
   below. The compact representation without any white space uses 468
   octets. (Of course, this number depends on the number of octets
   needed for the counter values.)

```
{
    "LOWPAN-MIB:LOWPAN-MIB": {
        "lowpanReasmTimeout": 20,
        "lowpanInReceives": 42,
        "lowpanInHdrErrors": 0,
        "lowpanInMeshReceives": 8,
        "lowpanInMeshForwds": 0,
        "lowpanInMeshDelivers": 0,
        "lowpanInReasmReqds": 22,
        "lowpanInReasmFails": 2,
        "lowpanInReasmOKs": 20,
        "lowpanInCompReqds": 16,
        "lowpanInCompFails": 2,
        "lowpanInCompOKs": 14,
        "lowpanInDiscards": 1,
        "lowpanInDelivers": 12,
        "lowpanOutRequests": 12,
        "lowpanOutCompReqds": 0,
        "lowpanOutCompFails": 0,
        "lowpanOutCompOKs": 0,
        "lowpanOutFragReqds": 5,
        "lowpanOutFragFails": 0,
        "lowpanOutFragOKs": 5,
        "lowpanOutFragCreates": 8,
        "lowpanOutMeshHopLimitExceeds": 0,
        "lowpanOutMeshNoRoutes": 0,
        "lowpanOutMeshRequests": 0,
        "lowpanOutMeshForwds": 0,
        "lowpanOutMeshTransmits": 0,
        "lowpanOutDiscards": 0,
        "lowpanOutTransmits": 15
    }
}
```

Authors' Addresses

Juergen Schoenwaelder
Jacobs University
Campus Ring 1
Bremen 28759
Germany

EMail: j.schoenwaelder@jacobs-university.de

Anuj Sehgal
Jacobs University
Campus Ring 1
Bremen 28759
Germany

EMail: s.anuj@jacobs-university.de

Tina Tsou
Huawei Technologies (USA)
2330 Central Expressway
Santa Clara CA 95050
USA

EMail: tina.tsou.zouting@huawei.com

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen 518129
P.R. China

EMail: cathyzhou@huawei.com