Internet Engineering Task Force
Internet-Draft
Intended status: Standards Track
Expires: June 18, 2016

Y. Fu
CNNIC
S. Jiang
Huawei Technologies Co., Ltd
J. Dong
Y. Chen
Tsinghua University
December 16, 2015

# DS-Lite Management Information Base (MIB) for AFTRs
# draft-ietf-softwire-dslite-mib-13

Abstract

This memo defines a portion of the Management Information Base (MIB) for using with network management protocols in the Internet community. In particular, it defines managed objects for Address Family Transition Routers (AFTRs) of Dual-Stack Lite (DS-Lite).

Table of Contents



## 1. Introduction

Dual-Stack Lite [RFC6333] is a solution to offer both IPv4 and IPv6 connectivity to customers crossing an IPv6 only infrastructure. One of its key components is an IPv4-over-IPv6 tunnel, which is used to provide IPv4 connectivity across a service provider's IPv6 network. Another key component is a carrier-grade IPv4-IPv4 Network Address Translation (NAT) to share service provider IPv4 addresses among customers.

This document defines a portion of the Management Information Base (MIB) for using with network management protocols in the Internet community. This MIB module may be used for configuration and monitoring Address Family Transition Routers (AFTRs) in a Dual-Stack Lite scenario.

## 2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119] when they appear in ALL CAPS. When these words are not in ALL CAPS (such as "should" or "Should"), they have their usual English meanings, and are not to be interpreted as [RFC2119] key words.

## 3. 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 [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 [RFC2578], [RFC2579] and [RFC2580].

## 4. Relationship to the IF-MIB

The Interfaces MIB [RFC2863] defines generic managed objects for managing interfaces. Each logical interface (physical or virtual) has an ifEntry. Tunnels are handled by creating a logical interface (ifEntry) for each tunnel. Each DS-Lite tunnel endpoint also acts as a virtual interface, which has a corresponding entry in the IP Tunnel MIB and Interface MIB. Those corresponding entries are indexed by ifIndex.

The ifOperStatus in ifTable is used to represent whether the DS-Lite tunnel function has been triggered. The ifInUcastPkts defined in ifTable will represent the number of IPv4 packets that have been encapsulated into IPv6 packets sent to a B4. The ifOutUcastPkts defined in ifTable contains the number of IPv6 packets that can be decapsulated to IPv4 in the virtual interface. Also, the IF-MIB defines ifMtu for the MTU of this tunnel interface, so DS-Lite MIB does not need to define the MTU for the tunnel.

## 5. Difference from the IP tunnel MIB and NATV2-MIB

The key technologies for DS-Lite are IP in IP (IPv4-in-IPv6) tunnels and NAT (IPv4 to IPv4 translation).

Notes: According to section 5.2 of [RFC6333], DS-Lite only defines IPv4 in IPv6 tunnels at this moment, but other types of encapsulation could be defined in the future. So this DS-Lite MIB only supports IP in IP encapsulation. If another RFC defines other tunnel types in the future, this DS-Lite MIB will be updated then.

The NATV2-MIB [RFC7659] is designed to carry translation from any address family to any address family, therefore it supports IPv4 to IPv4 translation.

The IP Tunnel MIB [RFC4087] is designed for managing tunnels of any type over IPv4 and IPv6 networks, therefore it has already supports IP in IP tunnels. But in a DS-Lite scenario, the tunnel type is point-to-multipoint IP in IP tunnels. The direct(2) defined in IP Tunnel MIB only supports point-to-point tunnel. So it needs to define a new tunnel type for DS-Lite.

However, the NATV2-MIB and IP Tunnel MIB together are not sufficient to support DS-Lite. This document describes the specific features for DS-Lite MIB, as below.

In the DS-Lite scenario, the Address Family Transition Router (AFTR) is not only the tunnel end concentrator, but also an IPv4-to-IPv4 NAT. So as defined in [RFC6333], when the IPv4 packets come back from the Internet to the AFTR, it knows how to reconstruct the IPv6 encapsulation by doing a reverse lookup in the extended IPv4 NAT binding table (section 6.6 of [RFC6333]). The NAT binding table in the AFTR is extended to include the IPv6 address of the tunnel initiator. However, the NAT binding information defined in NATV2-MIB as natv2PortMapTable is indexed by the NAT instance, protocol, and external realm and address. Because the tunnelIfTable defined in the TUNNEL-MIB [RFC4087] is indexed by the ifIndex, the DS-Lite-MIB needs to define the tunnel objects to extend the NAT binding entry by interface. Therefore, a combined MIB is necessary.

An implementation of the IP Tunnel MIB is required for DS-Lite. As the tunnel is not point-to-point in DS-Lite, it needs to define a new tunnel type for DS-Lite. And the tunnelIfEncapsMethod in the tunnelIfEntry should be set to dsLite ("xx"), and a corresponding entry in the DS-Lite module will exist for every tunnelIfEntry with this tunnelIfEncapsMethod. The tunnelIfRemoteInetAddress must be set to "::".

## 6. Structure of the MIB Module

The DS-Lite MIB provides a way to monitor and manage the devices (AFTRs) in a DS-Lite scenario through SNMP.

The DS-Lite MIB is configurable on a per-interface basis. It depends on several parts of the IF-MIB [RFC2863], IP Tunnel MIB [RFC4087], and NATV2-MIB [RFC7659].

### 6.1. The Object Group

This Group defines objects that are needed for DS-Lite MIB.

#### 6.1.1. The dsliteTunnel Subtree

The dsliteTunnel subtree describes managed objects used for managing tunnels in the DS-Lite scenario. Because the tunnelInetConfigLocalAddress and tunnelInetConfigRemoteAddress defined in the IP Tunnel MIB are not readable, a few new objects are defined in DS-Lite MIB.

#### 6.1.2. The dsliteNAT Subtree

The dsliteNAT subtree describes managed objects used for configuration as well as monitoring of an AFTR which is capable of a NAT function. Because the NATV2-MIB supports the NAT management function in DS-Lite, we may reuse it in DS-Lite MIB. The dsliteNAT subtree also provides the mapping information between the tunnel entry (dsliteTunnelEntry) and the NAT entry (dsliteNATBindEntry) by adding the IPv6 address of the B4 to the natv2PortMapEntry in the NATV2-MIB.

#### 6.1.3. The dsliteInfo Subtree

The dsliteInfo subtree provides statistical information for DS-Lite.

### 6.2. The Notification Group

This group defines some notification objects for DS-Lite.

#### 6.2.1. The dsliteTrap Subtree

The dsliteTrap subtree provides trap information in a DS-Lite scenario.

### 6.3. The Conformance Group

The dsliteConformance subtree provides conformance information of MIB objects.

## 7. MIB modules required for IMPORTS

This MIB module IMPORTs objects from [RFC2578], [RFC2580], [RFC2863], [RFC3411], [RFC4001] and [RFC7659].

## 8. Definitions

```
 DSLite-MIB DEFINITIONS  ::=  BEGIN

      IMPORTS
         MODULE-IDENTITY, OBJECT-TYPE, mib-2,
         NOTIFICATION-TYPE, Gauge32, TimeTicks,
         Integer32, Counter64, Unsigned32
            FROM SNMPv2-SMI

         OBJECT-GROUP, MODULE-COMPLIANCE,
         NOTIFICATION-GROUP
            FROM SNMPv2-CONF

         DisplayString
            FROM SNMPv2-TC

         SnmpAdminString
            FROM SNMP-FRAMEWORK-MIB

         ifIndex
            FROM IF-MIB

        InetAddress, InetAddressType, InetAddressPrefixLength,
        InetPortNumber
              FROM INET-ADDRESS-MIB

        ProtocolNumber, Natv2InstanceIndex, Natv2SubscriberIndex
             FROM NATV2-MIB;

      dsliteMIB MODULE-IDENTITY
      LAST-UPDATED "201512160000Z"          -- December 16, 2015
      ORGANIZATION "IETF Softwire Working Group"
      CONTACT-INFO
         "Yu Fu
          CNNIC
          No.4 South 4th Street, Zhongguancun, Hai-Dian District
          Beijing, P.R. China 100095
          EMail: fuyu@cnnic.cn

          Sheng Jiang
          Huawei Technologies Co., Ltd
          Huawei Building, 156 Beiqing Rd., Hai-Dian District
```

Beijing, P.R. China 100095
EMail: jiangsheng@huawei.com

Jiang Dong
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China
Email: knight.dongjiang@gmail.com

Yuchi Chen
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China
Email: flashfoxmx@gmail.com "

```
   DESCRIPTION
      "The MIB module is defined for management of objects in the
      DS-Lite scenario.
      Copyright (C) The Internet Society (2015).  This version
      of this MIB module is part of RFC yyyy; see the RFC itself
      for full legal notices. "
   REVISION    "201512160000Z"
   DESCRIPTION
      "Initial version. Published as RFC xxxx."
--RFC Ed.: RFC-edtitor pls fill in xxxx
      ::=  {  mib-2 xxx  }
--RFC Ed.: assigned by IANA, see section 10 for details



--Top level components of this MIB module

   dsliteMIBObjects OBJECT IDENTIFIER
      ::=  { dsliteMIB 1 }
   dsliteTunnel   OBJECT IDENTIFIER
      ::=  { dsliteMIBObjects 1 }

   dsliteNAT   OBJECT IDENTIFIER
      ::=  { dsliteMIBObjects 2 }

   dsliteInfo   OBJECT IDENTIFIER
      ::=  { dsliteMIBObjects 3 }

--Notifications section

   dsliteNotifications  OBJECT IDENTIFIER
```

```
        ::=  { dsliteMIB 0 }

      dsliteTraps  OBJECT IDENTIFIER
         ::=  { dsliteNotifications 1  }

   --dsliteTunnel

   --dsliteTunnelTable

      dsliteTunnelTable OBJECT-TYPE
         SYNTAX     SEQUENCE OF DsliteTunnelEntry
         MAX-ACCESS  not-accessible
         STATUS     current
         DESCRIPTION
            "The (conceptual) table containing information on
             configured tunnels. This table can be used to map
             a B4 address to the associated AFTR address. It can
             also be used for row creation."
         REFERENCE
            "B4, AFTR: RFC 6333."
         ::=  { dsliteTunnel 1 }

      dsliteTunnelEntry OBJECT-TYPE
         SYNTAX     DsliteTunnelEntry
         MAX-ACCESS  not-accessible
         STATUS     current
         DESCRIPTION
            "Each entry in this table contains the information on a
             particular configured tunnel."
             INDEX   { dsliteTunnelAddressType,
                       dsliteTunnelStartAddress,
                       dsliteTunnelEndAddress,
                       ifIndex }
         ::=  { dsliteTunnelTable 1 }

      DsliteTunnelEntry  ::=
         SEQUENCE {
          dsliteTunnelAddressType         InetAddressType,
          dsliteTunnelStartAddress        InetAddress,
          dsliteTunnelEndAddress          InetAddress,
          dsliteTunnelStartAddPreLen      InetAddressPrefixLength
      }

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

```
            "This object MUST be set to the value of ipv6(2).
             It describes the address type of the IPv4-in-IPv6
             tunnel initiator and endpoint."
        ::= { dsliteTunnelEntry 1 }

    dsliteTunnelStartAddress OBJECT-TYPE
        SYNTAX     InetAddress
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
           "The IPv6 address of the initiator of the tunnel
            The address type is given by dsliteTunnelAddressType."
        ::= { dsliteTunnelEntry 2 }

    dsliteTunnelEndAddress OBJECT-TYPE
        SYNTAX     InetAddress
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
           "The IPv6 address of the endpoint of the tunnel
            The address type is given by dsliteTunnelAddressType."
        ::= { dsliteTunnelEntry 3 }

    dsliteTunnelStartAddPreLen OBJECT-TYPE
        SYNTAX InetAddressPrefixLength
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
           "The IPv6 prefix length of the IP address for the
            initiator of the tunnel(dsliteTunnelStartAddress)."
        ::= { dsliteTunnelEntry 4 }


  --dsliteNATBindTable(according to the NAPT scheme)

     dsliteNATBindTable OBJECT-TYPE
       SYNTAX     SEQUENCE OF DsliteNATBindEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
          "This table contains information about currently
           active NAT binds in the NAT of the AFTR. This table
           adds the IPv6 address of a B4 to the natv2PortMapTable
           defined in NATV2-MIB (RFC7659)."
       REFERENCE
            "NATV2-MIB: section 4 of RFC7659."
       ::=  { dsliteNAT 1 }
```

```
      dsliteNATBindEntry OBJECT-TYPE
         SYNTAX     DsliteNATBindEntry
         MAX-ACCESS not-accessible
         STATUS     current
         DESCRIPTION
            "The entry in this table holds the mapping relationship
             between tunnel information and NAT bind information.
             Each entry in this table not only need to match a
             corresponding entry in the natv2PortMapTable but
             also a corresponding entry in the dsliteTunnelTable.
             So the INDEX of the entry needs to match a corresponding
             value in the natv2PortMapTable INDEX and a correspongding
             value in the dsliteTunnelTable INDEX. These entries are
             lost upon agent restart."
         REFERENCE
              "natv2PortMapTable: section 4 of RFC7659."
         INDEX   { dsliteNATBindMappingInstanceIndex,
                   dsliteNATBindMappingProto,
                   dsliteNATBindMappingExtRealm,
                   dsliteNATBindMappingExtAddressType,
                   dsliteNATBindMappingExtAddress,
                   dsliteNATBindMappingExtPort,
                   ifIndex,
                   dsliteTunnelStartAddress }
         ::=  {  dsliteNATBindTable 1   }

     DsliteNATBindEntry  ::=
         SEQUENCE {
         dsliteNATBindMappingInstanceIndex  Natv2InstanceIndex,
         dsliteNATBindMappingProto          ProtocolNumber,
         dsliteNATBindMappingExtRealm       SnmpAdminString,
         dsliteNATBindMappingExtAddressType InetAddressType,
         dsliteNATBindMappingExtAddress     InetAddress,
         dsliteNATBindMappingExtPort        InetPortNumber,
         dsliteNATBindMappingIntRealm       SnmpAdminString,
         dsliteNATBindMappingIntAddressType InetAddressType,
         dsliteNATBindMappingIntAddress     InetAddress,
         dsliteNATBindMappingIntPort        InetPortNumber,
         dsliteNATBindMappingPool           Unsigned32,
         dsliteNATBindMappingMapBehavior    INTEGER,
         dsliteNATBindMappingFilterBehavior INTEGER,
         dsliteNATBindMappingAddressPooling INTEGER
         }

      dsliteNATBindMappingInstanceIndex OBJECT-TYPE
          SYNTAX Natv2InstanceIndex
          MAX-ACCESS not-accessible
          STATUS current
```

```
          DESCRIPTION
             "Index of the NAT instance that created this port
              map entry."
        ::= { dsliteNATBindEntry 1 }

       dsliteNATBindMappingProto OBJECT-TYPE
          SYNTAX      ProtocolNumber
          MAX-ACCESS  not-accessible
          STATUS      current
          DESCRIPTION
             "This object specifies the mapping's transport protocol
             number."
          ::= { dsliteNATBindEntry 2 }

      dsliteNATBindMappingExtRealm OBJECT-TYPE
          SYNTAX     SnmpAdminString (SIZE(0..32))
          MAX-ACCESS not-accessible
          STATUS     current
          DESCRIPTION
             "The realm to which dsliteNATBindMappingExtAddress
             belongs."
          ::= { dsliteNATBindEntry 3 }

       dsliteNATBindMappingExtAddressType OBJECT-TYPE
          SYNTAX InetAddressType
          MAX-ACCESS not-accessible
          STATUS current
          DESCRIPTION
            "Address type for the mapping's external address.
             A value other than IPv4(1) would be unexpected."
          ::= { dsliteNATBindEntry 4 }

      dsliteNATBindMappingExtAddress OBJECT-TYPE
          SYNTAX InetAddress (SIZE (0..16))
          MAX-ACCESS not-accessible
          STATUS current
          DESCRIPTION
            "The mapping's external address. This is the source
             address for translated outgoing packets. The address
             type is given by dsliteNATBindMappingExtAddressType."
          ::= { dsliteNATBindEntry 5 }

       dsliteNATBindMappingExtPort OBJECT-TYPE
          SYNTAX InetPortNumber
          MAX-ACCESS not-accessible
          STATUS current
          DESCRIPTION
            "The mapping's assigned external port number.
```

```
            This is the source port for translated outgoing
            packets. This MUST be a non-zero value."
         ::= { dsliteNATBindEntry 6 }

      dsliteNATBindMappingIntRealm OBJECT-TYPE
         SYNTAX SnmpAdminString (SIZE(0..32))
         MAX-ACCESS read-only
         STATUS current
         DESCRIPTION
           "The realm to which natMappingIntAddress belongs. This
            realm defines the IPv6 address space from which the
            tunnel source address is taken. The realm of the
            encapsulated IPv4 address is restricted in scope to
            the tunnel, so there is no point in identifying it
            separately."
         ::= { dsliteNATBindEntry 7 }

   dsliteNATBindMappingIntAddressType OBJECT-TYPE
         SYNTAX InetAddressType
         MAX-ACCESS read-only
         STATUS current
         DESCRIPTION
            "Address type of the mapping's internal address.
             A value other than ipv4z(3) would be unexpected."
         ::= { dsliteNATBindEntry 8 }

     dsliteNATBindMappingIntAddress OBJECT-TYPE
         SYNTAX InetAddress
         MAX-ACCESS read-only
         STATUS current
         DESCRIPTION
           "The mapping's internal address. It is the IPv6 tunnel
            source address. The address type is given by
            dsliteNATBindMappingIntAddressType."
         ::= { dsliteNATBindEntry 9 }

     dsliteNATBindMappingIntPort OBJECT-TYPE
         SYNTAX InetPortNumber
         MAX-ACCESS read-only
         STATUS current
         DESCRIPTION
          "The mapping's internal port number. This MUST be a non-zero
           value."
          ::= { dsliteNATBindEntry 10 }

     dsliteNATBindMappingPool OBJECT-TYPE
         SYNTAX Unsigned32 (0|1..4294967295)
         MAX-ACCESS read-only
```

```
        STATUS current
        DESCRIPTION
          "Index of the pool that contains this mapping's external
           address and port. If zero, no pool is associated with this
           mapping."
        ::= { dsliteNATBindEntry 11 }

    dsliteNATBindMappingMapBehavior OBJECT-TYPE
        SYNTAX INTEGER{
        endpointIndependent (0),
        addressDependent(1),
        addressAndPortDependent (2)
        }
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
          "Mapping behavior as described in [RFC4787] section 4.1.

           endpointIndependent(0), the behavior REQUIRED by
           RFC 4787, REQ-1, maps the source address and port to
           the same external address and port for all destination
           address and port combinations reached through the same
           external realm and using the given protocol.

           addressDependent(1) maps to the same external address
           and port for all destination ports at the same
           destination address reached through the same external
           realm and using the given protocol.

           addressAndPortDependent(2) maps to a separate external
           address and port combination for each different
           destination address and port combination reached
           through the same external realm.

           For the DS-Lite scenario, it must be
           addressAndPortDependent(2)."
        REFERENCE
          "Mapping behavior: section 4.1 of RFC 4787.
           DS-Lite: RFC 6333."
        ::= { dsliteNATBindEntry 12 }

    dsliteNATBindMappingFilterBehavior OBJECT-TYPE
        SYNTAX INTEGER{
        endpointIndependent (0),
        addressDependent(1),
        addressAndPortDependent (2)
        }
        MAX-ACCESS read-only
```

```
        STATUS current
        DESCRIPTION
          "Filtering behavior as described in [RFC4787] section 5.

           endpointIndependent(0) accepts for translation packets
           from all combinations of remote address and port
           destined to the mapped external address and port via
           the given external realm and using the given protocol.

           addressDependent(1) accepts for translation packets from
           all remote ports from the same remote source address
           destined to the mapped external address and port via the
           given external realm and using the given protocol.

           addressAndPortDependent(2) accepts for translation only
           those packets with the same remote source address, port,
           and protocol incoming from the same external realm as
           identified when the applicable port map entry was
           created.

           RFC 4787, REQ-8 recommends either endpointIndependent(0)
           or addressDependent(1) filtering behavior depending on
           whether application friendliness or security takes
           priority.

           For the DS-Lite scenario, it must be
           addressAndPortDependent(2)."
        REFERENCE
          "Filtering behavior: section 5 of RFC 4787.
           DS-Lite: RFC 6333."
        ::= { dsliteNATBindEntry 13 }

    dsliteNATBindMappingAddressPooling OBJECT-TYPE
        SYNTAX INTEGER{
        arbitrary (0),
        paired (1)
        }
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
          "Type of address pooling behavior that was used to create
           this mapping.

           arbitrary(0) pooling behavior means that the NAT instance
           may create the new port mapping using any address in the
           pool that has a free port for the protocol concerned.

           paired(1) pooling behavior, the behavior RECOMMENDED by RFC
```

```
            4787, REQ-2, means that once a given internal address has
            been mapped to a particular address in a particular pool,
            further mappings of the same internal address to that pool
            will reuse the previously assigned pool member address."
          REFERENCE
            "Pooling behavior: section 4.1 of RFC 4787."
          ::= { dsliteNATBindEntry 14 }


      --dsliteInfo

      dsliteAFTRAlarmScalar OBJECT IDENTIFIER ::= { dsliteInfo 1 }

      dsliteAFTRAlarmB4AddrType OBJECT-TYPE
         SYNTAX  InetAddressType
         MAX-ACCESS accessible-for-notify
         STATUS current
         DESCRIPTION
            "This object indicates the address type of
             the B4 which will send an alarm."
         ::= { dsliteAFTRAlarmScalar 1 }

      dsliteAFTRAlarmB4Addr OBJECT-TYPE
         SYNTAX  InetAddress
         MAX-ACCESS accessible-for-notify
         STATUS current
         DESCRIPTION
            "This object indicates the IP address of
             B4 which will send an alarm. The address type is
             given by dsliteAFTRAlarmB4AddrType."
         ::= { dsliteAFTRAlarmScalar 2 }

      dsliteAFTRAlarmProtocolType OBJECT-TYPE
         SYNTAX INTEGER{
         tcp (0),
         udp (1),
         icmp (2),
         total (3)
         }
         MAX-ACCESS accessible-for-notify
         STATUS current
         DESCRIPTION
            "This object indicates the transport protocol type
             of alarm.

             tcp (0) means that the transport protocoal type of
             alarm is tcp.
```

```
            udp (1) means that the transport protocoal type of
            alarm is udp.

            icmp (2) means that the transport protocoal type of
            alarm is icmp.

            total (3) means that the transport protocoal type of
            alarm is total."
        ::= { dsliteAFTRAlarmScalar 3 }

    dsliteAFTRAlarmSpecificIPAddrType OBJECT-TYPE
        SYNTAX InetAddressType
        MAX-ACCESS accessible-for-notify
        STATUS current
        DESCRIPTION
           "This object indicates the address type of the IP address
            whose port usage has reached the threshold."
        ::= { dsliteAFTRAlarmScalar 4 }

     dsliteAFTRAlarmSpecificIP OBJECT-TYPE
        SYNTAX InetAddress
        MAX-ACCESS accessible-for-notify
        STATUS current
        DESCRIPTION
           "This object indicates the IP address whose port usage
            has reached the threshold. The address type is given by
            dsliteAFTRAlarmSpecificIPAddrType."
        ::= { dsliteAFTRAlarmScalar 5 }

      dsliteAFTRAlarmConnectNumber OBJECT-TYPE
        SYNTAX Integer32 (60..90)
        MAX-ACCESS read-write
        STATUS current
        DESCRIPTION
           "This object indicates the notification threshold
            of the DS-Lite tunnels which is active in
            the AFTR device."
         REFERENCE
           "AFTR: section 6 of RFC 6333."
        DEFVAL
            { 60 }
        ::= { dsliteAFTRAlarmScalar 6 }

      dsliteAFTRAlarmSessionNumber OBJECT-TYPE
        SYNTAX Integer32
        MAX-ACCESS read-write
        STATUS current
        DESCRIPTION
```

```
            "This object indicates the notification threshold of
             the IPv4 session for the user."
          REFERENCE
            "AFTR: section 6 of RFC 6333
             B4: section 5 of RFC 6333."
          DEFVAL
              { -1 }
          ::= { dsliteAFTRAlarmScalar 7 }

      dsliteAFTRAlarmPortNumber OBJECT-TYPE
          SYNTAX Integer32
          MAX-ACCESS read-write
          STATUS current
          DESCRIPTION
             "This object indicates the notification threshold of the NAT
              ports which have been used by user."
          DEFVAL
              { -1 }
          ::= { dsliteAFTRAlarmScalar 8 }

       dsliteStatisticsTable OBJECT-TYPE
          SYNTAX SEQUENCE OF DsliteStatisticsEntry
          MAX-ACCESS not-accessible
          STATUS current
          DESCRIPTION
             "This table provides statistical information
              about DS-Lite."
          ::= { dsliteInfo 2 }

       dsliteStatisticsEntry OBJECT-TYPE
          SYNTAX DsliteStatisticsEntry
          MAX-ACCESS not-accessible
          STATUS current
          DESCRIPTION
             "Ench entry in this table provides statistical information
              about DS-Lite."
          INDEX { dsliteStatisticsSubscriberIndex }
          ::= { dsliteStatisticsTable 1 }

       DsliteStatisticsEntry ::=
          SEQUENCE {
        dsliteStatisticsSubscriberIndex        Natv2SubscriberIndex,
        dsliteStatisticsDiscards               Counter64,
        dsliteStatisticsSends                  Counter64,
        dsliteStatisticsReceives               Counter64,
        dsliteStatisticsIpv4Session            Counter64,
        dsliteStatisticsIpv6Session            Counter64
       }
```

```
    dsliteStatisticsSubscriberIndex OBJECT-TYPE
       SYNTAX Natv2SubscriberIndex
       MAX-ACCESS not-accessible
       STATUS current
       DESCRIPTION
          "Index of the subscriber or host. A unique value,
           greater than zero, for each subscriber in the
           managed system."
        ::= { dsliteStatisticsEntry 1 }

     dsliteStatisticsDiscards OBJECT-TYPE
        SYNTAX Counter64
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
           "This object indicate the number of packets
            discarded from this subscriber."
       ::= { dsliteStatisticsEntry 2 }

      dsliteStatisticsSends  OBJECT-TYPE
        SYNTAX Counter64
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
           "This object indicate the number of packets which is
             sent to this subscriber."
        ::= { dsliteStatisticsEntry 3 }

      dsliteStatisticsReceives  OBJECT-TYPE
        SYNTAX Counter64
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
           "This object indicate the number of packets which is
             received from this subscriber."
        ::= { dsliteStatisticsEntry 4 }

     dsliteStatisticsIpv4Session OBJECT-TYPE
        SYNTAX Counter64
        MAX-ACCESS read-only
        STATUS current
        DESCRIPTION
           "This object indicate the number of the
            current IPv4 Sessions."
        REFERENCE
            "Session: the paragraph 2 of RFC 6333 section 11.
            (The AFTR should have the capability to log the
             tunnel-id, protocol, ports/IP addresses, and
```

```
             the creation time of the NAT binding to uniquely
             identify the user sessions)."
        ::= { dsliteStatisticsEntry 5 }

    dsliteStatisticsIpv6Session OBJECT-TYPE
       SYNTAX Counter64
       MAX-ACCESS read-only
       STATUS current
       DESCRIPTION
          "This object indicates the number of the
           current IPv6 Session. Because the AFTR is
           also a dual-stack device, it will also
           forward normal IPv6 packets for the
           inbound and outbound direction."
       REFERENCE
           "Session: the paragraph 2 of RFC 6333 section 11.
           (The AFTR should have the capability to log the
            tunnel-id, protocol, ports/IP addresses, and
            the creation time of the NAT binding to uniquely
            identify the user sessions)."
      ::= { dsliteStatisticsEntry 6 }

  ---dslite trap

    dsliteTunnelNumAlarm NOTIFICATION-TYPE
       OBJECTS { dsliteAFTRAlarmProtocolType,
                 dsliteAFTRAlarmB4AddrType,
                dsliteAFTRAlarmB4Addr }
       STATUS current
       DESCRIPTION
          "This trap is triggered when the number of
           current dslite tunnels exceeds the value of
           dsliteAFTRAlarmConnectNumber."
       ::= { dsliteTraps 1 }

    dsliteAFTRUserSessionNumAlarm NOTIFICATION-TYPE
       OBJECTS { dsliteAFTRAlarmProtocolType,
                 dsliteAFTRAlarmB4AddrType,
                 dsliteAFTRAlarmB4Addr }
       STATUS current
       DESCRIPTION
          "This trap is triggered when user sessions
           reach the threshold. The threshold
           is specified by the dsliteAFTRAlarmSessionNumber."
       REFERENCE
           "Session: the paragraph 2 of RFC 6333 section 11.
           (The AFTR should have the capability to log the
            tunnel-id, protocol, ports/IP addresses, and
```

```
             the creation time of the NAT binding to uniquely
             identify the user sessions)."
         ::= { dsliteTraps 2 }

       dsliteAFTRPortUsageOfSpecificIpAlarm NOTIFICATION-TYPE
         OBJECTS { dsliteAFTRAlarmSpecificIPAddrType,
                   dsliteAFTRAlarmSpecificIP }
         STATUS current
         DESCRIPTION
            "This trap is triggered when the used NAT
             ports of map address reach the threshold.
             The threshold is specified by the
             dsliteAFTRAlarmPortNumber."
         ::= { dsliteTraps 3 }

   --Module Conformance statement

      dsliteConformance   OBJECT IDENTIFIER
         ::=  { dsliteMIB 2 }

      dsliteCompliances OBJECT IDENTIFIER ::= { dsliteConformance 1 }

      dsliteGroups OBJECT IDENTIFIER ::= { dsliteConformance 2 }

   -- compliance statements

      dsliteCompliance MODULE-COMPLIANCE
          STATUS current
          DESCRIPTION
              "Describes the minimal requirements for conformance
               to the DSLite-MIB."
          MODULE -- this module
              MANDATORY-GROUPS { dsliteNATBindGroup,
                  dsliteTunnelGroup,
                  dsliteStatisticsGroup,
                  dsliteTrapsGroup,
                  dsliteAFTRAlarmScalarGroup }
          ::= { dsliteCompliances 1 }

      dsliteNATBindGroup OBJECT-GROUP
         OBJECTS {
                  dsliteNATBindMappingIntRealm,
                  dsliteNATBindMappingIntAddressType,
                  dsliteNATBindMappingIntAddress,
                  dsliteNATBindMappingIntPort,
                  dsliteNATBindMappingPool,
                  dsliteNATBindMappingMapBehavior,
                  dsliteNATBindMappingFilterBehavior,
```

```
                dsliteNATBindMappingAddressPooling }
        STATUS current
        DESCRIPTION
            "A collection of objects to support basic
             management of NAT binds in the NAT of the AFTR."
         ::= { dsliteGroups 1 }

    dsliteTunnelGroup OBJECT-GROUP
        OBJECTS { dsliteTunnelStartAddPreLen }
        STATUS current
        DESCRIPTION
            "A collection of objects to support management
             of ds-lite tunnels."
        ::= { dsliteGroups 2 }

    dsliteStatisticsGroup OBJECT-GROUP
        OBJECTS { dsliteStatisticsDiscards,
                  dsliteStatisticsSends,
                  dsliteStatisticsReceives,
                  dsliteStatisticsIpv4Session,
                  dsliteStatisticsIpv6Session }
      STATUS current
      DESCRIPTION
         " A collection of objects to support management
           of statistical information for AFTR devices."
       ::= { dsliteGroups 3 }

    dsliteTrapsGroup NOTIFICATION-GROUP
        NOTIFICATIONS { dsliteTunnelNumAlarm,
                        dsliteAFTRUserSessionNumAlarm,
                        dsliteAFTRPortUsageOfSpecificIpAlarm }
        STATUS current
        DESCRIPTION
           "A collection of objects to support management
            of trap information for AFTR devices."
        ::= { dsliteGroups 4 }

     dsliteAFTRAlarmScalarGroup OBJECT-GROUP
       OBJECTS { dsliteAFTRAlarmB4AddrType,
            dsliteAFTRAlarmB4Addr,
            dsliteAFTRAlarmProtocolType,
            dsliteAFTRAlarmSpecificIPAddrType,
            dsliteAFTRAlarmSpecificIP,
            dsliteAFTRAlarmConnectNumber,
            dsliteAFTRAlarmSessionNumber,
            dsliteAFTRAlarmPortNumber}
       STATUS current
       DESCRIPTION
```

```
            "A collection of objects to surpport management of
            the information about AFTR alarming Scalar."
        ::= { dsliteGroups 5 }

            END
```

## 9. Security Considerations

There are three objects defined in this MIB module with a MAX-ACCESS clause of read-write. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection opens devices to attack. These are the tables and objects and their sensitivity/vulnerability:

Notification thresholds: An attacker setting an arbitrarily low threshold can cause many useless notifications to be generated. Setting an arbitrarily high threshold can effectively disable notifications, which could be used to hide another attack.

dsliteAFTRAlarmConnectNumber

dsliteAFTRAlarmSessionNumber

dsliteAFTRAlarmPortNumber

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:

Objects that reveal host identities: Various objects can reveal the identity of private hosts that are engaged in a session with external end nodes. A curious outsider could monitor these to assess the number of private hosts being supported by the AFTR device. Further, a disgruntled former employee of an enterprise could use the information to break into specific private hosts by intercepting the existing sessions or originating new sessions into the host. If nothing else, unauthorized monitoring of these objects will violate individual subscribers' privacy.

entries in dsliteTunnelTable

entries in dsliteNATBindTable

Unauthorized read access to the dsliteTunnelTable would reveal information about the tunnel topology.

SNMP versions prior to SNMPv3 did not include adequate security. Even if the network itself is secure (for example by using IPSec), 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.

Implementations SHOULD provide the security features described by the SNMPv3 framework (see [RFC3410]), and implementations claiming compliance to the SNMPv3 standard MUST include full support for authentication and privacy via the User-based Security Model (USM) [RFC3414] with the AES cipher algorithm [RFC3826]. Implementations MAY also provide support for the Transport Security Model (TSM) [RFC5591] in combination with a secure transport such as SSH [RFC5592] or TLS/DTLS [RFC6353].

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.

## 10. IANA Considerations

The MIB module in this document uses the following IANA-assigned OBJECT IDENTIFIER value recorded in the SMI Numbers registry, and the following IANA-assigned tunnelType value recorded in the IANAtunnelType-MIB registry:

```
      Descriptor        OBJECT IDENTIFIER value
      ----------        -----------------------
      DSLite-MIB         { mib-2 XXX }

      IANAtunnelType ::= TEXTUAL-CONVENTION

                 SYNTAX     INTEGER {

                            dsLite ("XX")        -- dslite tunnel

                            }
```

## 11. Acknowledgements

The authors would like to thanks the valuable comments made by Suresh Krishnan, Ian Farrer, Yiu Lee, Qi Sun, Yong Cui, David Harrington, Dave Thaler, Tassos Chatzithomaoglou, Tom Taylor, Hui Deng, Carlos Pignataro, Matt Miller, Terry Manderson and other members of The SOFTWIRE WG.

This document was produced using the xml2rfc tool [RFC2629].

## 12. References

### 12.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, DOI 10.17487/RFC2578, April 1999, <http://www.rfc-editor.org/info/rfc2578>.

[RFC2580] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Conformance Statements for SMIv2", STD 58, RFC 2580, DOI 10.17487/RFC2580, April 1999, <http://www.rfc-editor.org/info/rfc2580>.

[RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, DOI 10.17487/RFC2863, June 2000, <http://www.rfc-editor.org/info/rfc2863>.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, DOI 10.17487/RFC3411, December 2002, <http://www.rfc-editor.org/info/rfc3411>.

[RFC4001] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 4001, DOI 10.17487/RFC4001, February 2005, <http://www.rfc-editor.org/info/rfc4001>.

[RFC4087] Thaler, D., "IP Tunnel MIB", RFC 4087, DOI 10.17487/RFC4087, June 2005, <http://www.rfc-editor.org/info/rfc4087>.

[RFC4787] Audet, F., Ed. and C. Jennings, "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", BCP 127, RFC 4787, DOI 10.17487/RFC4787, January 2007, <http://www.rfc-editor.org/info/rfc4787>.

[RFC6333] Durand, A., Droms, R., Woodyatt, J., and Y. Lee, "Dual-Stack Lite Broadband Deployments Following IPv4 Exhaustion", RFC 6333, DOI 10.17487/RFC6333, August 2011, <http://www.rfc-editor.org/info/rfc6333>.

[RFC7659] Perreault, S., Tsou, T., Sivakumar, S., and T. Taylor, "Definitions of Managed Objects for Network Address Translators (NATs)", RFC 7659, DOI 10.17487/RFC7659, October 2015, <http://www.rfc-editor.org/info/rfc7659>.

## 12.2. Informative References

[RFC2579] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Textual Conventions for SMIv2", STD 58, RFC 2579, DOI 10.17487/RFC2579, April 1999, <http://www.rfc-editor.org/info/rfc2579>.

[RFC2629] Rose, M., "Writing I-Ds and RFCs using XML", RFC 2629, DOI 10.17487/RFC2629, June 1999, <http://www.rfc-editor.org/info/rfc2629>.

[RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, DOI 10.17487/RFC3410, December 2002, <http://www.rfc-editor.org/info/rfc3410>.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, DOI 10.17487/RFC3414, December 2002, <http://www.rfc-editor.org/info/rfc3414>.

[RFC3826] Blumenthal, U., Maino, F., and K. McCloghrie, "The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model", RFC 3826, DOI 10.17487/RFC3826, June 2004, <http://www.rfc-editor.org/info/rfc3826>.

[RFC5591] Harrington, D. and W. Hardaker, "Transport Security Model for the Simple Network Management Protocol (SNMP)", STD 78, RFC 5591, DOI 10.17487/RFC5591, June 2009, <http://www.rfc-editor.org/info/rfc5591>.

[RFC5592] Harrington, D., Salowey, J., and W. Hardaker, "Secure Shell Transport Model for the Simple Network Management Protocol (SNMP)", RFC 5592, DOI 10.17487/RFC5592, June 2009, <http://www.rfc-editor.org/info/rfc5592>.

[RFC6353] Hardaker, W., "Transport Layer Security (TLS) Transport Model for the Simple Network Management Protocol (SNMP)", STD 78, RFC 6353, DOI 10.17487/RFC6353, July 2011, <http://www.rfc-editor.org/info/rfc6353>.

## Authors' Addresses

Yu Fu
CNNIC
No.4 South 4th Street, Zhongguancun
Hai-Dian District, Beijing, 100190
P.R. China

Email: fuyu@cnnic.cn

Sheng Jiang
Huawei Technologies Co., Ltd
Q14, Huawei Campus, No.156 Beiqing Road
Hai-Dian District, Beijing, 100095
P.R. China

Email: jiangsheng@huawei.com

Jiang Dong
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China

Email: knight.dongjiang@gmail.com

Yuchi Chen
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R. China

Email: flashfoxmx@gmail.com