

Software
Internet-Draft
Intended status: Standards Track
Expires: June 10, 2016

Y. Cui
J. Dong
P. Wu
M. Xu
Tsinghua University
A. Yla-Jaaski
Aalto University
December 8, 2015

Software Mesh Management Information Base (MIB)
draft-ietf-software-mesh-mib-13

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 a software mesh.

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 June 10, 2016.

Copyright Notice

Copyright (c) 2015 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in [Section 4](#).e of

the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1.	Introduction	2
2.	The Internet-Standard Management Framework	2
3.	Terminology	3
4.	Structure of the MIB Module	3
4.1.	The swmSupportedTunnelTable Subtree	3
4.2.	The swmEncapsTable Subtree	3
4.3.	The swmBGPNeighborTable Subtree	4
4.4.	The swmConformance Subtree	4
5.	Relationship to Other MIB Modules	4
5.1.	Relationship to the IF-MIB	4
5.2.	Relationship to the IP Tunnel MIB	5
5.3.	MIB modules required for IMPORTS	5
6.	Definitions	5
7.	Security Considerations	13
8.	IANA Considerations	14
9.	Acknowledgements	14
10.	References	14
10.1.	Normative References	14
10.2.	Informative References	16
	Authors' Addresses	16

[1.](#) Introduction

The Softwire mesh framework [RFC 5565](#) [[RFC5565](#)] is a tunneling mechanism that enables connectivity between islands of IPv4 networks across a single IPv6 backbone and vice versa. In a softwire mesh, extended multiprotocol-BGP (MP-BGP) is used to set up tunnels and advertise prefixes among address family border routers (AFBRs).

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 a softwire mesh [[RFC5565](#)].

[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). They

are defined using the mechanisms stated in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIV2 (Structure of Management Information Version 2), which is described in STD 58, [RFC 2578](#) [[RFC2578](#)], STD 58, [RFC 2579](#) [[RFC2579](#)] and STD 58, [RFC 2580](#) [[RFC2580](#)].

3. Terminology

This document uses terminology from the software problem statement [RFC 4925](#) [[RFC4925](#)], the BGP encapsulation subsequent address family identifier (SAFI) and the BGP tunnel encapsulation attribute [RFC 5512](#) [[RFC5512](#)], the software mesh framework [RFC 5565](#) [[RFC5565](#)] and the BGP IPsec tunnel encapsulation attribute and [RFC 5566](#) [[RFC5566](#)].

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 [RFC 2119](#) [[RFC2119](#)].

4. Structure of the MIB Module

The software mesh MIB provides a method to monitor the software mesh objects through SNMP.

4.1. The swmSupportedTunnelTable Subtree

The swmSupportedTunnelTable subtree provides the information about what types of tunnels can be used for software mesh scenarios in the AFBR. The software mesh framework [RFC 5565](#) [[RFC5565](#)] does not mandate the use of any particular tunneling technology. Based on the BGP tunnel encapsulation attribute tunnel types introduced by [RFC 5512](#) [[RFC5512](#)] and [RFC 5566](#) [[RFC5566](#)], the software mesh tunnel types include at least L2TPv3 (Layer Two Tunneling Protocol-Version 3) over IP, GRE (Generic Routing Encapsulation), Transmit tunnel endpoint, IPsec in Tunnel-mode, IP in IP tunnel with IPsec Transport Mode, MPLS-in-IP tunnel with IPsec Transport Mode and IP in IP. The detailed encapsulation information of different tunnel types (e.g., L2TPv3 Session ID, GRE Key, etc.) is not managed in the swmMIB.

4.2. The swmEncapsTable Subtree

The swmEncapsTable subtree provides software mesh NLRI-NH information (Network Layer Reachability Information-Next Hop) about the AFBR. It keeps the mapping between the External-IP (E-IP) prefix and the Internal-IP (I-IP) address of the next hop. The mappings determine which I-IP destination address will be used to encapsulate the received packet according to its E-IP destination address. The definitions of E-IP and I-IP are explained in [section 4.1](#) of RFC

5565[RFC5565]. The number of entries in `swmEncapsTable` shows how many software mesh tunnels are maintained in this AFBR.

4.3. The `swmBGPNeighborTable` Subtree

The subtree provides the software mesh BGP neighbor information of an AFBR. It includes the address of the software mesh BGP peer, and the kind of tunnel that the AFBR would use to communicate with this BGP peer.

4.4. The `swmConformance` Subtree

The subtree provides the conformance information of MIB objects.

5. Relationship to Other MIB Modules

5.1. 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 logical interfaces (`ifEntry`). Being a tunnel, software mesh interface has an entry in the Interface MIB, as well as an entry in IP Tunnel MIB. Those corresponding entries are indexed by `ifIndex`.

The `ifOperStatus` in the `ifTable` represents whether the mesh function of the AFBR has been triggered. If the software mesh capability is negotiated during the BGP OPEN phase, the mesh function is considered to be started, and the `ifOperStatus` is "up". Otherwise the `ifOperStatus` is "down".

In the case of an IPv4-over-IPv6 software mesh tunnel, `ifInUcastPkts` counts the number of IPv6 packets which are sent to the virtual interface for decapsulation into IPv4. The `ifOutUcastPkts` counts the number of IPv6 packets which are generated by encapsulating IPv4 packets sent to the virtual interface. Particularly, if these IPv4 packets need fragmentation, `ifOutUcastPkts` counts the number of packets after fragmentation.

In the case of an IPv6-over-IPv4 software mesh tunnel, `ifInUcastPkts` counts the number of IPv4 packets, which are delivered up to the virtual interface for decapsulation into IPv6. The `ifOutUcastPkts` counts the number of IPv4 packets, which are generated by encapsulating IPv6 packets sent down to the virtual interface. Particularly, if these IPv6 packets need to be fragmented, `ifOutUcastPkts` counts the number of packets after fragmentation. Similar definitions apply to other counter objects in the `ifTable`.

5.2. Relationship to the IP Tunnel MIB

The IP Tunnel MIB [[RFC4087](#)] contains objects applicable to all IP tunnels, including software mesh tunnels. Meanwhile, the Software Mesh MIB extends the IP Tunnel MIB to further describe encapsulation-specific information.

When running a point to multi-point tunnel, it is necessary for a software mesh AFBF to maintain an encapsulation table in order to perform correct "forwarding" among AFBFs. This forwarding function on an AFBF is performed by using the E-IP destination address to look up in the encapsulation table for the I-IP encapsulation destination address. An AFBF also needs to know the BGP peer information of the other AFBFs, so that it can negotiate the NLRI-NH information and the tunnel parameters with them.

The Software mesh MIB requires the implementation of the IP Tunnel MIB. The tunnelIfEncapsMethod in the tunnelIfEntry MUST be set to softwareMesh("xx"), and a corresponding entry in the software mesh MIB module will be presented for the tunnelIfEntry. The tunnelIfRemoteInetAddress MUST be set to "0.0.0.0" for IPv4 or "::" for IPv6 because it is a point to multi-point tunnel.

-- RFC Ed.: Please replace "xx" with IANA assigned number here.

The tunnelIfAddressType in the tunnelIfTable represents the type of address in the corresponding tunnelIfLocalInetAddress and tunnelIfRemoteInetAddress objects. The tunnelIfAddressType is identical to swmEncapsIIPDstType in software mesh, which can support either IPv4-over-IPv6 or IPv6-over-IPv4. When the swmEncapsEIPDstType is IPv6 and the swmEncapsIIPDstType is IPv4, the tunnel type is IPv6-over-IPv4; When the swmEncapsEIPDstType is IPv4 and the swmEncapsIIPDstType is IPv6, the encapsulation mode would be IPv4-over-IPv6.

5.3. MIB modules required for IMPORTS

The following MIB module IMPORTS objects from SNMPv2-SMI [[RFC2578](#)], SNMPv2-CONF [[RFC2580](#)], IF-MIB [[RFC2863](#)] and INET-ADDRESS-MIB [[RFC4001](#)].

6. Definitions

SOFTWARE-MESH-MIB DEFINITIONS ::= BEGIN

IMPORTS

MODULE-IDENTITY, OBJECT-TYPE, smi-2 FROM SNMPv2-SMI

OBJECT-GROUP, MODULE-COMPLIANCE FROM SNMPv2-CONF

InetAddress, InetAddressType, InetAddressPrefixLength

FROM INET-ADDRESS-MIB

ifIndex FROM IF-MIB

IANA tunnelType FROM IANAifType-MIB;

swmMIB MODULE-IDENTITY

LAST-UPDATED "201512080000Z" -- December 08, 2015

ORGANIZATION "Software Working Group"

CONTACT-INFO "

Yong Cui

Email: yong@csnet1.cs.tsinghua.edu.cn

Jiang Dong

Email: knight.dongjiang@gmail.com

Peng Wu

Email: weapon9@gmail.com

Mingwei Xu

Email: xmw@cernet.edu.cn

Antti Yla-Jaaski

Email: antti.yla-jaaski@aalto.fi

Email comments directly to the software WG Mailing
List at softwires@ietf.org

"

DESCRIPTION

"This MIB module contains managed object definitions for
the software mesh framework.

Copyright (C) The Internet Society (2015). This
version of this MIB module is part of [RFC 5565](#);
see the RFC itself for full legal notices."

REVISION "201512080000Z"

DESCRIPTION

"The MIB module is defined for management of object in
the Software mesh framework."

::= { mib-2 XXX }


```
swmObjects OBJECT IDENTIFIER ::= { swmMIB 1 }

-- swmSupportedTunnelTable
swmSupportedTunnelTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SwmSupportedTunnelEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table of objects that shows what kind of tunnels
        can be supported by the AFBR."
    ::= { swmObjects 1 }

swmSupportedTunnelEntry OBJECT-TYPE
    SYNTAX      SwmSupportedTunnelEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A set of objects that show what kind of tunnels
        can be supported in the AFBR. If the AFBR supports
        multiple tunnel types, the swmSupportedTunnelTable
        would have several entries."
    INDEX { swmSupportedTunnelType }
    ::= { swmSupportedTunnelTable 1 }

SwmSupportedTunnelEntry ::= SEQUENCE {
    swmSupportedTunnelType      IANAtunnelType
}

swmSupportedTunnelType OBJECT-TYPE
    SYNTAX      IANAtunnelType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Represents the tunnel type that can be used for software
        mesh scenarios, such as L2TPv3 over IP, GRE, Transmit
        tunnel endpoint, IPsec in Tunnel-mode, IP in IP tunnel with
        IPsec Transport Mode, MPLS-in-IP tunnel with IPsec Transport
        Mode and IP in IP. There is no restriction of tunnel type
        the Software mesh can use."
    REFERENCE
        "L2TPv3 over IP, GRE, IP in IP in RFC5512.
        Transmit tunnel endpoint, IPsec in Tunnel-mode, IP in IP
        tunnel with IPsec Transport Mode, MPLS-in-IP tunnel with
        IPsec Transport Mode in RFC5566."
    ::= { swmSupportedTunnelEntry 1 }

-- end of swmSupportedTunnelTable
```



```
--swmEncapsTable
swmEncapsTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF SwmEncapsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table of objects that display the
        software mesh encapsulation information."
    ::= { swmObjects 2 }

swmEncapsEntry OBJECT-TYPE
    SYNTAX      SwmEncapsEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "A table of objects that manage the software mesh I-IP
        encapsulation destination based on the E-IP destination
        prefix."
    INDEX { ifIndex,
            swmEncapsEIPDstType,
            swmEncapsEIPDst,
            swmEncapsEIPPrefixLength
          }
    ::= { swmEncapsTable 1 }

SwmEncapsEntry ::= SEQUENCE {
    swmEncapsEIPDstType      InetAddressType,
    swmEncapsEIPDst          InetAddress,
    swmEncapsEIPPrefixLength InetAddressPrefixLength,
    swmEncapsIIPDstType      InetAddressType,
    swmEncapsIIPDst          InetAddress
}

swmEncapsEIPDstType OBJECT-TYPE
    SYNTAX      InetAddressType
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This object specifies the address type used for
        swmEncapsEIPDst. It is different from the tunnelIfAddressType
        in the tunnelIfTable. The swmEncapsEIPDstType is IPv6 (2)
        if it is IPv6-over-IPv4 tunneling. The swmEncapsEIPDstType is
        IPv4 (1) if it is IPv4-over-IPv6 tunneling."
    REFERENCE
        "IPv4 and IPv6 in RFC 4001."
    ::= { swmEncapsEntry 1 }

swmEncapsEIPDst OBJECT-TYPE
```


SYNTAX InetAddress
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The E-IP destination prefix, which is
 used for I-IP encapsulation destination looking up."
REFERENCE
 "E-IP and I-IP in [RFC 5565](#)."
::= { swmEncapsEntry 2 }

swmEncapsEIPPrefixLength OBJECT-TYPE

SYNTAX InetAddressPrefixLength
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The prefix length of the E-IP destination prefix."
::= { swmEncapsEntry 3 }

swmEncapsIIPDstType OBJECT-TYPE

SYNTAX InetAddressType
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "This object specifies the address type used for
 swmEncapsIIPDst. It is the same as the tunnelIfAddressType
 in the tunnelIfTable."
REFERENCE
 "IPv4 and IPv6 in [RFC 4001](#)."
::= { swmEncapsEntry 4 }

swmEncapsIIPDst OBJECT-TYPE

SYNTAX InetAddress
MAX-ACCESS read-only
STATUS current
DESCRIPTION
 "The I-IP destination address, which is used as the
 encapsulation destination for the corresponding E-IP
 prefix. Since the tunnelIfRemoteInetAddress in the
 tunnelIfTable should be 0.0.0.0 or ::, swmEncapsIIPDst
 should be the destination address used in the outer
 IP header."
REFERENCE
 "E-IP and I-IP in [RFC 5565](#)."
::= { swmEncapsEntry 5 }

-- End of swmEncapsTable

-- swmBGPNeighborTable

swmBGPNeighborTable OBJECT-TYPE

SYNTAX SEQUENCE OF SwmBGPNeighborEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "A table of objects that display the software mesh
 BGP neighbor information."
::= { swmObjects 3 }

swmBGPNeighborEntry OBJECT-TYPE
SYNTAX SwmBGPNeighborEntry
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "A set of objects that display the software mesh
 BGP neighbor information."
INDEX {
 ifIndex,
 swmBGPNeighborInetAddressType,
 swmBGPNeighborInetAddress
}
::= { swmBGPNeighborTable 1 }

SwmBGPNeighborEntry ::= SEQUENCE {
 swmBGPNeighborInetAddressType InetAddressType,
 swmBGPNeighborInetAddress InetAddress,
 swmBGPNeighborTunnelType IANA_tunnelType
}

swmBGPNeighborInetAddressType OBJECT-TYPE
SYNTAX InetAddressType
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "This object specifies the address type used for
 swmBGPNeighborInetAddress."
::= { swmBGPNeighborEntry 1 }

swmBGPNeighborInetAddress OBJECT-TYPE
SYNTAX InetAddress
MAX-ACCESS not-accessible
STATUS current
DESCRIPTION
 "The address of the AFBR's BGP neighbor. The
 address type is the same as the tunnelIfAddressType
 in the tunnelIfTable."
::= { swmBGPNeighborEntry 2 }

swmBGPNeighborTunnelType OBJECT-TYPE


```
SYNTAX      IANAtunnelType
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "Represents the type of tunnel that the AFBR
    chooses to transmit traffic with another AFBR/BGP
    neighbor."
    ::= { swmBGPNeighborEntry 3 }
-- End of swmBGPNeighborTable

-- conformance information
swmConformance
    OBJECT IDENTIFIER ::= { swmMIB 2 }

swmCompliances
    OBJECT IDENTIFIER ::= { swmConformance 1 }

swmGroups
    OBJECT IDENTIFIER ::= { swmConformance 2 }

-- compliance statements
swmCompliance MODULE-COMPLIANCE
    STATUS current
    DESCRIPTION
        "Describes the requirements for conformance to the software
        mesh MIB.

        The following index objects cannot be added as OBJECT
        clauses but nevertheless have compliance requirements:
        "

        -- OBJECT  swmEncapsEIPDstType
        -- SYNTAX  InetAddressType { ipv4(1), ipv6(2) }
        -- DESCRIPTION
        -- "An implementation is required to support
        -- global IPv4 and/or IPv6 addresses, depending
        -- on its support for IPv4 and IPv6."

        -- OBJECT  swmEncapsEIPDst
        -- SYNTAX  InetAddress (SIZE(4|16))
        -- DESCRIPTION
        -- "An implementation is required to support
        -- global IPv4 and/or IPv6 addresses, depending
        -- on its support for IPv4 and IPv6."

        -- OBJECT  swmEncapsEIPPrefixLength
        -- SYNTAX  InetAddressPrefixLength (Unsigned32 (0..128))
        -- DESCRIPTION
        -- "An implementation is required to support
        -- global IPv4 and/or IPv6 addresses, depending
```



```
-- on its support for IPv4 and IPv6."

-- OBJECT  swmBGPNeighborInetAddressType
-- SYNTAX  InetAddressType { ipv4(1), ipv6(2) }
-- DESCRIPTION
-- "An implementation is required to support
-- global IPv4 and/or IPv6 addresses, depending
-- on its support for IPv4 and IPv6."

-- OBJECT  swmBGPNeighborInetAddress
-- SYNTAX  InetAddress (SIZE(4|16))
-- DESCRIPTION
-- "An implementation is required to support
-- global IPv4 and/or IPv6 addresses, depending
-- on its support for IPv4 and IPv6."

MODULE -- this module
MANDATORY-GROUPS {
    swmSupportedTunnelGroup,
    swmEncapsGroup,
    swmBGPNeighborGroup
}
::= { swmCompliances 1 }

swmSupportedTunnelGroup    OBJECT-GROUP
OBJECTS {
    swmSupportedTunnelType
}
STATUS current
DESCRIPTION
    "The collection of objects which are used to show
    what kind of tunnel the AFBF supports."
::= { swmGroups 1 }

swmEncapsGroup            OBJECT-GROUP
OBJECTS {
    swmEncapsIIPDst,
    swmEncapsIIPDstType
}
STATUS current
DESCRIPTION
    "The collection of objects which are used to display
    software mesh encapsulation information."
::= { swmGroups 2 }

swmBGPNeighborGroup       OBJECT-GROUP
OBJECTS {
    swmBGPNeighborTunnelType
```



```
}
STATUS current
DESCRIPTION
    "The collection of objects which are used to display
    software mesh BGP neighbor information."
 ::= { swmGroups 3 }

END
```

7. Security Considerations

Because this MIB module reuses the IP tunnel MIB, the security considerations of the IP tunnel MIB is also applicable to the Software mesh MIB.

There are no management objects defined in this MIB module that have a MAX-ACCESS clause of read-write and/or read-create. So, if this MIB module is implemented correctly, then there is no risk that an intruder can alter or create any management objects of this MIB module via direct SNMP SET operations.

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 objects and their sensitivity/vulnerability.

Particularly, `swmSupportedTunnelType`, `swmEncapsIIPDstType`, `swmEncapsIIPDst` and `swmBGPNeighborTunnelType` can expose the types of tunnels used within the internal network, and potentially reveal the topology of the internal network.

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.

8. IANA Considerations

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

Descriptor	OBJECT IDENTIFIER value
-----	-----
swmMIB	{ mib-2 XXX }

IANAtunnelType ::= TEXTUAL-CONVENTION	
SYNTAX	INTEGER {
	softwareMesh ("xx") -- software Mesh tunnel
	}

9. Acknowledgements

The authors would like to thank Dave Thaler, Jean-Philippe Dionne, Qi Sun, Sheng Jiang, Yu Fu for their valuable comments.

10. References

10.1. 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>>.

- [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>>.
- [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>>.
- [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>>.
- [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>>.
- [RFC5512] Mohapatra, P. and E. Rosen, "The BGP Encapsulation Subsequent Address Family Identifier (SAFI) and the BGP Tunnel Encapsulation Attribute", [RFC 5512](#), DOI 10.17487/RFC5512, April 2009, <<http://www.rfc-editor.org/info/rfc5512>>.
- [RFC5565] Wu, J., Cui, Y., Metz, C., and E. Rosen, "Softwire Mesh Framework", [RFC 5565](#), DOI 10.17487/RFC5565, June 2009, <<http://www.rfc-editor.org/info/rfc5565>>.
- [RFC5566] Berger, L., White, R., and E. Rosen, "BGP IPsec Tunnel Encapsulation Attribute", [RFC 5566](#), DOI 10.17487/RFC5566, June 2009, <<http://www.rfc-editor.org/info/rfc5566>>.
- [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>>.

10.2. Informative References

- [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>>.
- [RFC4925] Li, X., Ed., Dawkins, S., Ed., Ward, D., Ed., and A. Durand, Ed., "Software Problem Statement", [RFC 4925](#), DOI 10.17487/RFC4925, July 2007, <<http://www.rfc-editor.org/info/rfc4925>>.
- [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>>.
- [RFC4087] Thaler, D., "IP Tunnel MIB", [RFC 4087](#), DOI 10.17487/RFC4087, June 2005, <<http://www.rfc-editor.org/info/rfc4087>>.

Authors' Addresses

Yong Cui
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6260-3059
EMail: yong@csnet1.cs.tsinghua.edu.cn

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

Phone: +86-10-6278-5822
EMail: knight.dongjiang@gmail.com

Peng Wu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
EMail: weapon9@gmail.com

Mingwei Xu
Tsinghua University
Department of Computer Science, Tsinghua University
Beijing 100084
P.R.China

Phone: +86-10-6278-5822
EMail: xmw@cernet.edu.cn

Antti Yla-Jaaski
Aalto University
Konemiehentie 2
Espoo 02150
Finland

Phone: +358-40-5954222
EMail: antti.yla-jaaski@aalto.fi

