

Network Working Group  
Request for Comments: 4382  
Category: Standards Track

T. Nadeau, Ed.  
H. van der Linde, Ed.  
Cisco Systems, Inc.  
February 2006

MPLS/BGP Layer 3 Virtual Private Network (VPN)  
Management Information Base

Status of This Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2006).

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 describes managed objects to configure and/or monitor Multiprotocol Label Switching Layer-3 Virtual Private Networks on a Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) supporting this feature.

## Table of Contents

1. Introduction .....	2
2. Terminology .....	3
3. The Internet-Standard Management Framework .....	3
4. Assumptions and Prerequisites .....	3
5. Brief Description of MIB Objects .....	3
5.1. mplsL3VpnVrfTable .....	3
5.2. mplsL3VpnIfConfTable .....	4
5.3. mplsL3VpnVrfPerfTable .....	4
5.4. mplsL3VpnVrfRouteTable .....	4
5.5. MplsVpnVrfRTTable .....	4
6. Example of MPLS L3VPN Setup .....	4
7. MPLS-L3VPN-STD-MIB Module Definitions .....	5
8. Security Considerations .....	38
9. IANA Considerations .....	40
9.1. IANA Considerations for MPLS-L3VPN-STD-MIB .....	40
10. Dedication .....	40
11. Acknowledgements .....	40
12. References .....	40
12.1. Normative References .....	40
12.2. Informative References .....	41

## 1. Introduction

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes managed objects to configure and/or monitor Multiprotocol Label Switching Layer-3 Virtual Private Networks on a Multi-Protocol Label Switching (MPLS) Label Switching Router (LSR) supporting this feature.

This document adopts the definitions, acronyms, and mechanisms described in [RFC4364]. Unless otherwise stated, the mechanisms of [RFC4364] apply and will not be re-described here.

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 [RFC2119].

## 2. Terminology

This document uses terminology from the document describing the MPLS architecture [RFC3031] and from the document describing MPLS Layer-3 VPNs (L3VPN) [RFC4364], as well as the MPLS architecture [RFC3031].

Throughout this document, the use of the terms "Provider Edge (PE) and Customer Edge (CE)" or "PE/CE" will be replaced by "PE" in all cases except when a network device is a CE when used in the carrier's carrier model.

## 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 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].

## 4. Assumptions and Prerequisites

It is assumed that certain things are configured and operational in order for the tables and objects described in this MIB to function correctly. These things are outlined below:

- MPLS in general, must be configured and operational.
- Label Distribution Protocol (LDP) paths or traffic-engineered tunnels [RFC3812] should be configured between PEs and CEs.

## 5. Brief Description of MIB Objects

The following subsections describe the purpose of each of the objects contained in the MPLS-L3VPN-STD-MIB.

### 5.1. mplsL3VpnVrfTable

This table represents the MPLS L3VPNs that are configured. A Network Management System (NMS) or SNMP agent creates an entry in this table for every MPLS L3VPN configured on the LSR being examined. The Virtual Routing and Forwarding (VRF) that is

configured at a particular device represents an instance of some VPN, but not the entire VPN (unless it is the only VRF, of course). The collective set of VRF instances comprises the actual VPN. This information is typically only known in its entirety at the NMS. That is, specific devices generally only know of their local VRF information, but not that of other LSRs' VRFs.

#### 5.2. mplsL3VpnIfConfTable

This table represents the MPLS L3VPN-enabled interfaces that are associated with a specific VRF as represented in the aforementioned mplsL3VpnVrfTable. Each entry in this table corresponds to an entry in the Interfaces MIB. In addition, each entry extends its corresponding entry in the Interfaces MIB to contain specific MPLS L3VPN information. Due to this correspondence, certain objects such as traffic counters are not found in this MIB to avoid overlap, but instead are found in the Interfaces MIB [RFC2863].

#### 5.3. mplsL3VpnVrfPerfTable

This table contains objects to measure the performance of MPLS L3VPNs and augments the mplsL3VpnVrfTable. High capacity counters are provided for objects that are likely to wrap around quickly on objects such as high-speed interface counters.

#### 5.4. mplsL3VpnVrfRouteTable

The table contains the objects necessary to configure and monitor routes used by a particular VRF. This includes a cross-connect pointer into the MPLS-LSR-STD-MIB's mplsXCTable, which may be used to refer that entry to its label stack used to label switch that entry.

#### 5.5. MplsVpnVrfRTTable

The table contains the objects necessary to configure and monitor route targets for a particular VRF.

### 6. Example of MPLS L3VPN Setup

In this section, we provide a brief example of using the MIB objects described in the following section. While this example is not meant to illustrate every nuance of the MIB, it is intended as an aid to understanding some of the key concepts. It is our intent that it is read only after the reader has gone through the MIB itself.

This configuration is under the assumption that 1) MPLS has been pre-configured in the network, through enabling LDP or Resource Reservation Protocol - Traffic Engineering (RSVP-TE); 2) OSPF or Intermediate System to Intermediate System (IS-IS) has been pre-configured; and 3) BGP sessions have been established between PEs.

Defining the VRF, the route target, and route distinguisher:

In mplsL3VpnVrfTable:

```
{
    mplsL3VpnVrfName           = "RED",
    mplsL3VpnVrfDescription    = "Intranet of Company ABC",
    mplsL3VpnVrfRD             = "100:1", -- octet string
    mplsL3VpnVrfRowStatus      = createAndGo(4)
}
```

In mplsL3VpnVrfRouteTable:

```
{
    mplsL3VpnVrfRTRowStatus."Red"."100:1".import = createAndGo,
    mplsL3VpnVrfRTRowStatus."Red"."100:1".export = createAndGo
}
```

## 7. MPLS-L3VPN-STD-MIB Module Definitions

MPLS-L3VPN-STD-MIB DEFINITIONS ::= BEGIN

IMPORTS

```
MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
Integer32, Counter32, Unsigned32, Gauge32
    FROM SNMPv2-SMI -- [RFC2578]
MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
    FROM SNMPv2-CONF -- [RFC2580]
TEXTUAL-CONVENTION, TruthValue, RowStatus,
TimeStamp, StorageType
    FROM SNMPv2-TC -- [RFC2579]
InterfaceIndex, InterfaceIndexOrZero
    FROM IF-MIB -- [RFC2863]
VPNIdOrZero
    FROM VPN-TC-STD-MIB -- [RFC4265]
SnmpAdminString
    FROM SNMP-FRAMEWORK-MIB -- [RFC3411]
IANAipRouteProtocol
    FROM IANA-RTPROTO-MIB -- [RTPROTO]
InetAddress, InetAddressType,
InetAddressPrefixLength,
InetAddressAutonomousSystemNumber
    FROM INET-ADDRESS-MIB -- [RFC4001]
mplsStdMIB
    FROM MPLS-TC-STD-MIB -- [RFC3811]
```

```
MplsIndexType
  FROM MPLS-LSR-STD-MIB
;

-- [RFC3813]

mplsL3VpnMIB MODULE-IDENTITY
  LAST-UPDATED "200601230000Z" -- 23 January 2006
  ORGANIZATION "IETF Layer-3 Virtual Private
    Networks Working Group."
  CONTACT-INFO
    "
      Thomas D. Nadeau
      tnadeau@cisco.com

      Harmen van der Linde
      havander@cisco.com

      Comments and discussion to l3vpn@ietf.org"
  DESCRIPTION
    "This MIB contains managed object definitions for the
      Layer-3 Multiprotocol Label Switching Virtual
      Private Networks.

      Copyright (C) The Internet Society (2006). This
      version of this MIB module is part of RFC4382; see
      the RFC itself for full legal notices."
  -- Revision history.
  REVISION
    "200601230000Z" -- 23 January 2006
  DESCRIPTION
    "Initial version. Published as RFC 4382."
    ::= { mplsStdMIB 11 }

-- Textual Conventions.
MplsL3VpnName ::= TEXTUAL-CONVENTION
  STATUS      current
  DESCRIPTION
    "An identifier that is assigned to each MPLS/BGP VPN and
      is used to uniquely identify it. This is assigned by the
      system operator or NMS and SHOULD be unique throughout
      the MPLS domain. If this is the case, then this identifier
      can then be used at any LSR within a specific MPLS domain
      to identify this MPLS/BGP VPN. It may also be possible to
      preserve the uniqueness of this identifier across MPLS
      domain boundaries, in which case this identifier can then
      be used to uniquely identify MPLS/BGP VPNs on a more global
      basis. This object MAY be set to the VPN ID as defined in
      RFC 2685."
  REFERENCE
    "RFC 2685 Fox B., et al, 'Virtual Private
```

Networks Identifier', September 1999."  
 SYNTAX OCTET STRING (SIZE (0..31))

MplsL3VpnRouteDistinguisher ::= TEXTUAL-CONVENTION  
 STATUS current  
 DESCRIPTION  
 "Syntax for a route distinguisher and route target  
 as defined in [RFC4364]."  
 REFERENCE  
 "[RFC4364]"  
 SYNTAX OCTET STRING(SIZE (0..256))

MplsL3VpnRtType ::= TEXTUAL-CONVENTION  
 STATUS current  
 DESCRIPTION  
 "Used to define the type of a route target usage.  
 Route targets can be specified to be imported,  
 exported, or both. For a complete definition of a  
 route target, see [RFC4364]."  
 REFERENCE  
 "[RFC4364]"  
 SYNTAX INTEGER { import(1), export(2), both(3) }

-- Top level components of this MIB.  
 mplsL3VpnNotifications OBJECT IDENTIFIER ::= { mplsL3VpnMIB 0 }  
 mplsL3VpnObjects OBJECT IDENTIFIER ::= { mplsL3VpnMIB 1 }  
 mplsL3VpnScalars OBJECT IDENTIFIER ::= { mplsL3VpnObjects 1 }  
 mplsL3VpnConf OBJECT IDENTIFIER ::= { mplsL3VpnObjects 2 }  
 mplsL3VpnPerf OBJECT IDENTIFIER ::= { mplsL3VpnObjects 3 }  
 mplsL3VpnRoute OBJECT IDENTIFIER ::= { mplsL3VpnObjects 4 }  
 mplsL3VpnConformance OBJECT IDENTIFIER ::= { mplsL3VpnMIB 2 }

--  
 -- Scalar Objects  
 --

mplsL3VpnConfiguredVrfs OBJECT-TYPE  
 SYNTAX Unsigned32  
 MAX-ACCESS read-only  
 STATUS current  
 DESCRIPTION  
 "The number of VRFs that are configured on this node."  
 ::= { mplsL3VpnScalars 1 }

mplsL3VpnActiveVrfs OBJECT-TYPE  
 SYNTAX Gauge32  
 MAX-ACCESS read-only  
 STATUS current

## DESCRIPTION

"The number of VRFs that are active on this node.

That is, those VRFs whose corresponding mplsL3VpnVrfOperStatus object value is equal to operational (1)."

::= { mplsL3VpnScalars 2 }

## mplsL3VpnConnectedInterfaces OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"Total number of interfaces connected to a VRF."

::= { mplsL3VpnScalars 3 }

## mplsL3VpnNotificationEnable OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

## DESCRIPTION

"If this object is true, then it enables the generation of all notifications defined in this MIB. This object's value should be preserved across agent reboots."

## REFERENCE

"See also [RFC3413] for explanation that notifications are under the ultimate control of the MIB modules in this document."

DEFVAL { false }

::= { mplsL3VpnScalars 4 }

## mplsL3VpnVrfConfMaxPossRts OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"Denotes maximum number of routes that the device will allow all VRFs jointly to hold. If this value is set to 0, this indicates that the device is unable to determine the absolute maximum. In this case, the configured maximum MAY not actually be allowed by the device."

::= { mplsL3VpnScalars 5 }

## mplsL3VpnVrfConfRteMxThrshTime OBJECT-TYPE

SYNTAX Unsigned32

UNITS "seconds"

MAX-ACCESS read-only

STATUS current



## DESCRIPTION

"Denotes the interval in seconds, at which the route max threshold notification may be reissued after the maximum value has been exceeded (or has been reached if `mplsL3VpnVrfConfMaxRoutes` and `mplsL3VpnVrfConfHighRteThresh` are equal) and the initial notification has been issued. This value is intended to prevent continuous generation of notifications by an agent in the event that routes are continually added to a VRF after it has reached its maximum value. If this value is set to 0, the agent should only issue a single notification at the time that the maximum threshold has been reached, and should not issue any more notifications until the value of routes has fallen below the configured threshold value. This is the recommended default behavior."

DEFVAL { 0 }

::= { mplsL3VpnScalars 6 }

`mplsL3VpnI11lLb1RcvThrsh` OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-write

STATUS current

## DESCRIPTION

"The number of illegally received labels above which the `mplsNumVrfSecI11glLb1ThrshExcd` notification is issued. The persistence of this value mimics that of the device's configuration."

::= { mplsL3VpnScalars 7 }

## -- VPN Interface Configuration Table

`mplsL3VpnIfConfTable` OBJECT-TYPE

SYNTAX SEQUENCE OF `MplsL3VpnIfConfEntry`

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"This table specifies per-interface MPLS capability and associated information."

::= { mplsL3VpnConf 1 }

`mplsL3VpnIfConfEntry` OBJECT-TYPE

SYNTAX `MplsL3VpnIfConfEntry`

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"An entry in this table is created by an LSR for every interface capable of supporting MPLS L3VPN. Each entry in this table is meant to correspond to an entry in the Interfaces Table."

```

INDEX          { mplsL3VpnVrfName, mplsL3VpnIfConfIndex }
 ::= { mplsL3VpnIfConfTable 1 }

```

```

MplsL3VpnIfConfEntry ::= SEQUENCE {
    mplsL3VpnIfConfIndex      InterfaceIndex,
    mplsL3VpnIfVpnClassification  INTEGER,
    mplsL3VpnIfVpnRouteDistProtocol  BITS,
    mplsL3VpnIfConfStorageType      StorageType,
    mplsL3VpnIfConfRowStatus        RowStatus
}

```

mplsL3VpnIfConfIndex OBJECT-TYPE

SYNTAX InterfaceIndex

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This is a unique index for an entry in the mplsL3VpnIfConfTable. A non-zero index for an entry indicates the ifIndex for the corresponding interface entry in the MPLS-VPN-layer in the ifTable. Note that this table does not necessarily correspond one-to-one with all entries in the Interface MIB having an ifType of MPLS-layer; rather, only those that are enabled for MPLS L3VPN functionality."

REFERENCE

"RFC2863"

```
 ::= { mplsL3VpnIfConfEntry 1 }

```

mplsL3VpnIfVpnClassification OBJECT-TYPE

```

SYNTAX          INTEGER { carrierOfCarrier (1),
                           enterprise (2),
                           interProvider (3)
}

```

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"Denotes whether this link participates in a carrier's carrier, enterprise, or inter-provider scenario."

DEFVAL { enterprise }

```
 ::= { mplsL3VpnIfConfEntry 2 }

```

mplsL3VpnIfVpnRouteDistProtocol OBJECT-TYPE

```

SYNTAX          BITS { none (0),
                       bgp (1),
                       ospf (2),
                       rip(3),
                       isis(4),

```

```

        static(5),
        other (6)
    }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Denotes the route distribution protocol across the
    PE-CE link. Note that more than one routing protocol
    may be enabled at the same time; thus, this object is
    specified as a bitmask. For example, static(5) and
    ospf(2) are a typical configuration."
 ::= { mplsL3VpnIfConfEntry 3 }

mplsL3VpnIfConfStorageType OBJECT-TYPE
SYNTAX          StorageType
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The storage type for this VPN If entry.
    Conceptual rows having the value 'permanent'
    need not allow write access to any columnar
    objects in the row."
REFERENCE
    "See RFC2579."
DEFVAL { volatile }
 ::= { mplsL3VpnIfConfEntry 4 }

mplsL3VpnIfConfRowStatus OBJECT-TYPE
SYNTAX          RowStatus
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "This variable is used to create, modify, and/or
    delete a row in this table. Rows in this
    table signify that the specified interface is
    associated with this VRF. If the row creation
    operation succeeds, the interface will have been
    associated with the specified VRF, otherwise the
    agent MUST not allow the association. If the agent
    only allows read-only operations on this table, it
    MUST create entries in this table as they are created
    on the device. When a row in this table is in
    active(1) state, no objects in that row can be
    modified except mplsL3VpnIfConfStorageType and
    mplsL3VpnIfConfRowStatus."
 ::= { mplsL3VpnIfConfEntry 5 }

-- VRF Configuration Table

```

```

mplsL3VpnVrfTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF MplsL3VpnVrfEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "This table specifies per-interface MPLS L3VPN
        VRF Table capability and associated information.
        Entries in this table define VRF routing instances
        associated with MPLS/VPN interfaces. Note that
        multiple interfaces can belong to the same VRF
        instance. The collection of all VRF instances
        comprises an actual VPN."
    ::= { mplsL3VpnConf 2 }

mplsL3VpnVrfEntry OBJECT-TYPE
    SYNTAX          MplsL3VpnVrfEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "An entry in this table is created by an LSR for
        every VRF capable of supporting MPLS L3VPN. The
        indexing provides an ordering of VRFs per-VPN
        interface."
    INDEX          { mplsL3VpnVrfName }
    ::= { mplsL3VpnVrfTable 1 }
MplsL3VpnVrfEntry ::= SEQUENCE {
    mplsL3VpnVrfName                MplsL3VpnName,
    mplsL3VpnVrfVpnId              VPNIidOrZero,
    mplsL3VpnVrfDescription        SnmpAdminString,
    mplsL3VpnVrfRD                 MplsL3VpnRouteDistinguisher,
    mplsL3VpnVrfCreationTime       TimeStamp,
    mplsL3VpnVrfOperStatus         INTEGER,
    mplsL3VpnVrfActiveInterfaces   Gauge32,
    mplsL3VpnVrfAssociatedInterfaces Unsigned32,
    mplsL3VpnVrfConfMidRteThresh  Unsigned32,
    mplsL3VpnVrfConfHighRteThresh Unsigned32,
    mplsL3VpnVrfConfMaxRoutes     Unsigned32,
    mplsL3VpnVrfConfLastChanged   TimeStamp,
    mplsL3VpnVrfConfRowStatus      RowStatus,
    mplsL3VpnVrfConfAdminStatus   INTEGER,
    mplsL3VpnVrfConfStorageType   StorageType
}

mplsL3VpnVrfName OBJECT-TYPE
    SYNTAX          MplsL3VpnName
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION

```

"The human-readable name of this VPN. This MAY be equivalent to the [RFC2685] VPN-ID, but may also vary. If it is set to the VPN ID, it MUST be equivalent to the value of mplsL3VpnVrfVpnId. It is strongly recommended that all sites supporting VRFs that are part of the same VPN use the same naming convention for VRFs as well as the same VPN ID."

## REFERENCE

"[RFC2685]"

::= { mplsL3VpnVrfEntry 1 }

## mplsL3VpnVrfVpnId OBJECT-TYPE

SYNTAX VPNIIDOrZero

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The VPN ID as specified in [RFC2685]. If a VPN ID has not been specified for this VRF, then this variable SHOULD be set to a zero-length OCTET STRING."

::= { mplsL3VpnVrfEntry 2 }

## mplsL3VpnVrfDescription OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The human-readable description of this VRF."

DEFVAL { "" }

::= { mplsL3VpnVrfEntry 3 }

## mplsL3VpnVrfRD OBJECT-TYPE

SYNTAX MplsL3VpnRouteDistinguisher

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The route distinguisher for this VRF."

DEFVAL { "" }

::= { mplsL3VpnVrfEntry 4 }

## mplsL3VpnVrfCreationTime OBJECT-TYPE

SYNTAX TimeStamp

MAX-ACCESS read-only

STATUS current

## DESCRIPTION

"The time at which this VRF entry was created."

::= { mplsL3VpnVrfEntry 5 }

## mplsL3VpnVrfOperStatus OBJECT-TYPE

SYNTAX           INTEGER { up (1),  
                                down (2)  
                                }

MAX-ACCESS       read-only

STATUS           current

## DESCRIPTION

"Denotes whether or not a VRF is operational. A VRF is up(1) when there is at least one interface associated with the VRF whose ifOperStatus is up(1). A VRF is down(2) when:

a. There does not exist at least one interface whose ifOperStatus is up(1).

b. There are no interfaces associated with the VRF."

::= { mplsL3VpnVrfEntry 6 }

## mplsL3VpnVrfActiveInterfaces OBJECT-TYPE

SYNTAX           Gauge32

MAX-ACCESS       read-only

STATUS           current

## DESCRIPTION

"Total number of interfaces connected to this VRF with ifOperStatus = up(1).

This value should increase when an interface is associated with the corresponding VRF and its corresponding ifOperStatus is equal to up(1). If an interface is associated whose ifOperStatus is not up(1), then the value is not incremented until such time as it transitions to this state.

This value should be decremented when an interface is disassociated with a VRF or the corresponding ifOperStatus transitions out of the up(1) state to any other state.

"

::= { mplsL3VpnVrfEntry 7 }

## mplsL3VpnVrfAssociatedInterfaces OBJECT-TYPE

SYNTAX           Unsigned32

MAX-ACCESS       read-only

STATUS           current

## DESCRIPTION

"Total number of interfaces connected to this VRF (independent of ifOperStatus type)."

::= { mplsL3VpnVrfEntry 8 }

## mplsL3VpnVrfConfMidRteThresh OBJECT-TYPE

SYNTAX           Unsigned32

MAX-ACCESS       read-create

```
STATUS          current
DESCRIPTION
  "Denotes mid-level water marker for the number
  of routes that this VRF may hold."
DEFVAL { 0 }
 ::= { mplsL3VpnVrfEntry 9 }

mplsL3VpnVrfConfHighRteThresh  OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
  "Denotes high-level water marker for the number of
  routes that this VRF may hold."
DEFVAL { 0 }
 ::= { mplsL3VpnVrfEntry 10 }

mplsL3VpnVrfConfMaxRoutes  OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
  "Denotes maximum number of routes that this VRF is
  configured to hold.  This value MUST be less than or
  equal to mplsL3VpnVrfConfMaxPossRts unless it is set
  to 0."
DEFVAL { 0 }
 ::= { mplsL3VpnVrfEntry 11 }

mplsL3VpnVrfConfLastChanged  OBJECT-TYPE
SYNTAX          TimeStamp
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
  "The value of sysUpTime at the time of the last
  change of this table entry, which includes changes of
  VRF parameters defined in this table or addition or
  deletion of interfaces associated with this VRF."
 ::= { mplsL3VpnVrfEntry 12 }

mplsL3VpnVrfConfRowStatus  OBJECT-TYPE
SYNTAX          RowStatus
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
  "This variable is used to create, modify, and/or
  delete a row in this table."
```

When a row in this table is in active(1) state, no objects in that row can be modified except  
 mplsL3VpnVrfConfAdminStatus, mplsL3VpnVrfConfRowStatus,  
 and mplsL3VpnVrfConfStorageType."

```
::= { mplsL3VpnVrfEntry 13 }
```

```
mplsL3VpnVrfConfAdminStatus OBJECT-TYPE
```

```
SYNTAX      INTEGER {
                                up(1),      -- ready to pass packets
                                down(2),    -- can't pass packets
                                testing(3) -- in some test mode
                        }
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"Indicates the desired operational status of this
  VRF."
```

```
::= { mplsL3VpnVrfEntry 14 }
```

```
mplsL3VpnVrfConfStorageType OBJECT-TYPE
```

```
SYNTAX      StorageType
```

```
MAX-ACCESS  read-create
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"The storage type for this VPN VRF entry.
  Conceptual rows having the value 'permanent'
  need not allow write access to any columnar
  objects in the row."
```

```
REFERENCE
```

```
"See RFC2579."
```

```
DEFVAL { volatile }
```

```
::= { mplsL3VpnVrfEntry 15 }
```

```
-- MplsL3VpnVrfRTTable
```

```
mplsL3VpnVrfRTTable OBJECT-TYPE
```

```
SYNTAX      SEQUENCE OF MplsL3VpnVrfRTEntry
```

```
MAX-ACCESS  not-accessible
```

```
STATUS      current
```

```
DESCRIPTION
```

```
"This table specifies per-VRF route target association.
  Each entry identifies a connectivity policy supported
  as part of a VPN."
```

```
::= { mplsL3VpnConf 3 }
```

```
mplsL3VpnVrfRTEntry OBJECT-TYPE
```

```
SYNTAX      MplsL3VpnVrfRTEntry
```

```
MAX-ACCESS  not-accessible
```



STATUS current

DESCRIPTION

"An entry in this table is created by an LSR for each route target configured for a VRF supporting a MPLS L3VPN instance. The indexing provides an ordering per-VRF instance. See [RFC4364] for a complete definition of a route target."

INDEX { mplsL3VpnVrfName, mplsL3VpnVrfRTIndex,  
mplsL3VpnVrfRTType }  
::= { mplsL3VpnVrfRTTable 1 }

MplsL3VpnVrfRTEntry ::= SEQUENCE {  
mplsL3VpnVrfRTIndex Unsigned32,  
mplsL3VpnVrfRTType MplsL3VpnRtType,  
mplsL3VpnVrfRT MplsL3VpnRouteDistinguisher,  
mplsL3VpnVrfRTDescr SnmpAdminString,  
mplsL3VpnVrfRTRowStatus RowStatus,  
mplsL3VpnVrfRTStorageType StorageType  
}

mplsL3VpnVrfRTIndex OBJECT-TYPE

SYNTAX Unsigned32 (1..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Auxiliary index for route targets configured for a particular VRF."

::= { mplsL3VpnVrfRTEntry 2 }

mplsL3VpnVrfRTType OBJECT-TYPE

SYNTAX MplsL3VpnRtType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The route target distribution type."

::= { mplsL3VpnVrfRTEntry 3 }

mplsL3VpnVrfRT OBJECT-TYPE

SYNTAX MplsL3VpnRouteDistinguisher

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The route target distribution policy."

DEFVAL { "" }

::= { mplsL3VpnVrfRTEntry 4 }

mplsL3VpnVrfRTDescr OBJECT-TYPE

SYNTAX SnmpAdminString

```

MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Description of the route target."
DEFVAL { "" }
 ::= { mplsL3VpnVrfRTEntry 5 }

```

```

mplsL3VpnVrfRTRowStatus OBJECT-TYPE
SYNTAX          RowStatus
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "This variable is used to create, modify, and/or
    delete a row in this table.  When a row in this
    table is in active(1) state, no objects in that row
    can be modified except mplsL3VpnVrfRTRowStatus."
 ::= { mplsL3VpnVrfRTEntry 6 }

```

```

mplsL3VpnVrfRTStorageType OBJECT-TYPE
SYNTAX          StorageType
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The storage type for this VPN route target (RT) entry.
    Conceptual rows having the value 'permanent'
    need not allow write access to any columnar
    objects in the row."
REFERENCE
    "See RFC2579."
DEFVAL { volatile }
 ::= { mplsL3VpnVrfRTEntry 7 }

```

#### -- VRF Security Table

```

mplsL3VpnVrfSecTable OBJECT-TYPE
SYNTAX          SEQUENCE OF MplsL3VpnVrfSecEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table specifies per MPLS L3VPN VRF Table
    security-related counters."
 ::= { mplsL3VpnConf 6 }

```

```

mplsL3VpnVrfSecEntry OBJECT-TYPE
SYNTAX          MplsL3VpnVrfSecEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION

```

"An entry in this table is created by an LSR for every VRF capable of supporting MPLS L3VPN. Each entry in this table is used to indicate security-related information for each VRF entry."

```
AUGMENTS      { mplsL3VpnVrfEntry }
 ::= { mplsL3VpnVrfSecTable 1 }
```

```
MplsL3VpnVrfSecEntry ::= SEQUENCE {
    mplsL3VpnVrfSecIllegalLblVltns      Counter32,
    mplsL3VpnVrfSecDiscontinuityTime    TimeStamp
}
```

```
mplsL3VpnVrfSecIllegalLblVltns OBJECT-TYPE
```

```
SYNTAX          Counter32
```

```
MAX-ACCESS      read-only
```

```
STATUS          current
```

```
DESCRIPTION
```

"Indicates the number of illegally received labels on this VPN/VRF."

Discontinuities in the value of this counter can occur at re-initialization of the management system, and at other times as indicated by the value of mplsL3VpnVrfSecDiscontinuityTime."

```
::= { mplsL3VpnVrfSecEntry 1 }
```

```
mplsL3VpnVrfSecDiscontinuityTime OBJECT-TYPE
```

```
SYNTAX          TimeStamp
```

```
MAX-ACCESS      read-only
```

```
STATUS          current
```

```
DESCRIPTION
```

"The value of sysUpTime on the most recent occasion at which any one or more of this entry's counters suffered a discontinuity. If no such discontinuities have occurred since the last re-initialization of the local management subsystem, then this object contains a zero value."

```
::= { mplsL3VpnVrfSecEntry 2 }
```

```
-- VRF Performance Table
```

```
mplsL3VpnVrfPerfTable OBJECT-TYPE
```

```
SYNTAX          SEQUENCE OF MplsL3VpnVrfPerfEntry
```

```
MAX-ACCESS      not-accessible
```

```
STATUS          current
```

```
DESCRIPTION
```

"This table specifies per MPLS L3VPN VRF Table performance

```

        information."
 ::= { mplsL3VpnPerf 1 }

mplsL3VpnVrfPerfEntry OBJECT-TYPE
    SYNTAX      MplsL3VpnVrfPerfEntry
    MAX-ACCESS   not-accessible
    STATUS       current
    DESCRIPTION
        "An entry in this table is created by an LSR for
         every VRF capable of supporting MPLS L3VPN."
    AUGMENTS     { mplsL3VpnVrfEntry }
    ::= { mplsL3VpnVrfPerfTable 1 }

MplsL3VpnVrfPerfEntry ::= SEQUENCE {
    mplsL3VpnVrfPerfRoutesAdded      Counter32,
    mplsL3VpnVrfPerfRoutesDeleted    Counter32,
    mplsL3VpnVrfPerfCurrNumRoutes    Gauge32,
    mplsL3VpnVrfPerfRoutesDropped    Counter32,
    mplsL3VpnVrfPerfDiscTime         TimeStamp
}

mplsL3VpnVrfPerfRoutesAdded OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Indicates the number of routes added to this VPN/VRF
         since the last discontinuity. Discontinuities in
         the value of this counter can occur
         at re-initialization of the management system, and at
         other times as indicated by the value of
         mplsL3VpnVrfPerfDiscTime."
    ::= { mplsL3VpnVrfPerfEntry 1 }

mplsL3VpnVrfPerfRoutesDeleted OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION
        "Indicates the number of routes removed from this VPN/VRF.

         Discontinuities in the value of this counter can occur
         at re-initialization of the management system, and at
         other times as indicated by the value of
         mplsL3VpnVrfPerfDiscTime."
    ::= { mplsL3VpnVrfPerfEntry 2 }

mplsL3VpnVrfPerfCurrNumRoutes      OBJECT-TYPE

```

```

SYNTAX          Gauge32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Indicates the number of routes currently used by this
    VRF."
 ::= { mplsL3VpnVrfPerfEntry 3 }

```

```
mplsL3VpnVrfPerfRoutesDropped OBJECT-TYPE
```

```

SYNTAX          Counter32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "This counter should be incremented when the number of routes
    contained by the specified VRF exceeds or attempts to exceed
    the maximum allowed value as indicated by
    mplsL3VpnVrfMaxRouteThreshold.

    Discontinuities in the value of this counter can occur
    at re-initialization of the management system, and at
    other times as indicated by the value of
    mplsL3VpnVrfPerfDiscTime."
 ::= { mplsL3VpnVrfPerfEntry 4 }

```

```
mplsL3VpnVrfPerfDiscTime OBJECT-TYPE
```

```

SYNTAX          TimeStamp
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "The value of sysUpTime on the most recent occasion at
    which any one or more of this entry's counters suffered
    a discontinuity. If no such discontinuities have
    occurred since the last re-initialization of the local
    management subsystem, then this object contains a zero
    value."
 ::= { mplsL3VpnVrfPerfEntry 5 }

```

```
-- VRF Routing Table
```

```
mplsL3VpnVrfRteTable OBJECT-TYPE
```

```

SYNTAX          SEQUENCE OF MplsL3VpnVrfRteEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "This table specifies per-interface MPLS L3VPN VRF Table
    routing information. Entries in this table define VRF routing
    entries associated with the specified MPLS/VPN interfaces. Note

```

that this table contains both BGP and Interior Gateway Protocol IGP routes, as both may appear in the same VRF."

## REFERENCE

"[RFC2096]"

::= { mplsL3VpnRoute 1 }

## mplsL3VpnVrfRteEntry OBJECT-TYPE

SYNTAX MplsL3VpnVrfRteEntry

MAX-ACCESS not-accessible

STATUS current

## DESCRIPTION

"An entry in this table is created by an LSR for every route present configured (either dynamically or statically) within the context of a specific VRF capable of supporting MPLS/BGP VPN. The indexing provides an ordering of VRFs per-VPN interface.

Implementers need to be aware that there are quite a few index objects that together can exceed the size allowed for an Object Identifier (OID). So implementers must make sure that OIDs of column instances in this table will have no more than 128 sub-identifiers, otherwise they cannot be accessed using SNMPv1, SNMPv2c, or SNMPv3."

```
INDEX { mplsL3VpnVrfName,
        mplsL3VpnVrfRteInetCidrDestType,
        mplsL3VpnVrfRteInetCidrDest,
        mplsL3VpnVrfRteInetCidrPfxLen,
        mplsL3VpnVrfRteInetCidrPolicy,
        mplsL3VpnVrfRteInetCidrNHopType,
        mplsL3VpnVrfRteInetCidrNextHop
      }
 ::= { mplsL3VpnVrfRteTable 1 }
```

## MplsL3VpnVrfRteEntry ::= SEQUENCE {

mplsL3VpnVrfRteInetCidrDestType	InetAddressType,
mplsL3VpnVrfRteInetCidrDest	InetAddress,
mplsL3VpnVrfRteInetCidrPfxLen	InetAddressPrefixLength,
mplsL3VpnVrfRteInetCidrPolicy	OBJECT IDENTIFIER,
mplsL3VpnVrfRteInetCidrNHopType	InetAddressType,
mplsL3VpnVrfRteInetCidrNextHop	InetAddress,
mplsL3VpnVrfRteInetCidrIfIndex	InterfaceIndexOrZero,
mplsL3VpnVrfRteInetCidrType	INTEGER,
mplsL3VpnVrfRteInetCidrProto	IANAipRouteProtocol,
mplsL3VpnVrfRteInetCidrAge	Gauge32,
mplsL3VpnVrfRteInetCidrNextHopAS	InetAutonomousSystemNumber,
mplsL3VpnVrfRteInetCidrMetric1	Integer32,
mplsL3VpnVrfRteInetCidrMetric2	Integer32,

```

mplsL3VpnVrfRteInetCidrMetric3      Integer32,
mplsL3VpnVrfRteInetCidrMetric4      Integer32,
mplsL3VpnVrfRteInetCidrMetric5      Integer32,
mplsL3VpnVrfRteXCPointer             MplsIndexType,
mplsL3VpnVrfRteInetCidrStatus        RowStatus
}

```

#### mplsL3VpnVrfRteInetCidrDestType OBJECT-TYPE

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

##### DESCRIPTION

"The type of the mplsL3VpnVrfRteInetCidrDest address, as defined in the InetAddress MIB.

Only those address types that may appear in an actual routing table are allowed as values of this object."

REFERENCE "RFC4001"

::= { mplsL3VpnVrfRteEntry 1 }

#### mplsL3VpnVrfRteInetCidrDest OBJECT-TYPE

SYNTAX InetAddress

MAX-ACCESS not-accessible

STATUS current

##### DESCRIPTION

"The destination IP address of this route.

The type of this address is determined by the value of the mplsL3VpnVrfRteInetCidrDestType object.

The values for the index objects mplsL3VpnVrfRteInetCidrDest and mplsL3VpnVrfRteInetCidrPfxLen must be consistent. When the value of mplsL3VpnVrfRteInetCidrDest is x, then the bitwise logical-AND of x with the value of the mask formed from the corresponding index object mplsL3VpnVrfRteInetCidrPfxLen MUST be equal to x. If not, then the index pair is not consistent and an inconsistentName error must be returned on SET or CREATE requests."

::= { mplsL3VpnVrfRteEntry 2 }

#### mplsL3VpnVrfRteInetCidrPfxLen OBJECT-TYPE

SYNTAX InetAddressPrefixLength (0..128)

MAX-ACCESS not-accessible

STATUS current

##### DESCRIPTION

"Indicates the number of leading one bits that form the

mask to be logical-ANDed with the destination address before being compared to the value in the mplsL3VpnVrfRteInetCidrDest field.

The values for the index objects mplsL3VpnVrfRteInetCidrDest and mplsL3VpnVrfRteInetCidrPfxLen must be consistent. When the value of mplsL3VpnVrfRteInetCidrDest is x, then the bitwise logical-AND of x with the value of the mask formed from the corresponding index object mplsL3VpnVrfRteInetCidrPfxLen MUST be equal to x. If not, then the index pair is not consistent and an inconsistentName error must be returned on SET or CREATE requests."

::= { mplsL3VpnVrfRteEntry 3 }

mplsL3VpnVrfRteInetCidrPolicy OBJECT-TYPE

SYNTAX OBJECT IDENTIFIER

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This object is an opaque object without any defined semantics. Its purpose is to serve as an additional index that may delineate between multiple entries to the same destination. The value { 0 0 } shall be used as the default value for this object."

::= { mplsL3VpnVrfRteEntry 4 }

mplsL3VpnVrfRteInetCidrNHopType OBJECT-TYPE

SYNTAX InetAddressType

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The type of the mplsL3VpnVrfRteInetCidrNextHop address, as defined in the InetAddress MIB.

Value should be set to unknown(0) for non-remote routes.

Only those address types that may appear in an actual routing table are allowed as values of this object."

REFERENCE "RFC4001"

::= { mplsL3VpnVrfRteEntry 5 }

mplsL3VpnVrfRteInetCidrNextHop OBJECT-TYPE

SYNTAX InetAddress

MAX-ACCESS not-accessible

STATUS current



## DESCRIPTION

"On remote routes, the address of the next system en route. For non-remote routes, a zero-length string. The type of this address is determined by the value of the mplsl3VpnVrfRteInetCidrNHopType object."

::= { mplsl3VpnVrfRteEntry 6 }

## mplsl3VpnVrfRteInetCidrIfIndex OBJECT-TYPE

SYNTAX InterfaceIndexOrZero

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The ifIndex value that identifies the local interface through which the next hop of this route should be reached. A value of 0 is valid and represents the scenario where no interface is specified."

DEFVAL { 0 }

::= { mplsl3VpnVrfRteEntry 7 }

## mplsl3VpnVrfRteInetCidrType OBJECT-TYPE

SYNTAX INTEGER {

other (1), -- not specified by this MIB  
 reject (2), -- route which discards traffic and  
           -- returns ICMP notification  
 local (3), -- local interface  
 remote (4), -- remote destination  
 blackhole(5) -- route which discards traffic  
               -- silently

}

MAX-ACCESS read-create

STATUS current

## DESCRIPTION

"The type of route. Note that local(3) refers to a route for which the next hop is the final destination; remote(4) refers to a route for which the next hop is not the final destination.

Routes that do not result in traffic forwarding or rejection should not be displayed even if the implementation keeps them stored internally.

reject(2) refers to a route that, if matched, discards the message as unreachable and returns a notification (e.g., ICMP error) to the message sender. This is used in some protocols as a means of correctly aggregating routes.

blackhole(5) refers to a route that, if matched,

```
        discards the message silently."
DEFVAL { other }
 ::= { mplsL3VpnVrfRteEntry 8 }

mplsL3VpnVrfRteInetCidrProto OBJECT-TYPE
    SYNTAX      IANAipRouteProtocol
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The routing mechanism via which this route was learned.
        Inclusion of values for gateway routing protocols is
        not intended to imply that hosts should support those
        protocols."
    ::= { mplsL3VpnVrfRteEntry 9 }

mplsL3VpnVrfRteInetCidrAge OBJECT-TYPE
    SYNTAX      Gauge32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The number of seconds since this route was last updated
        or otherwise determined to be correct. Note that no
        semantics of 'too old' can be implied except through
        knowledge of the routing protocol by which the route
        was learned."
    ::= { mplsL3VpnVrfRteEntry 10 }

mplsL3VpnVrfRteInetCidrNextHopAS OBJECT-TYPE
    SYNTAX      InetAutonomousSystemNumber
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The Autonomous System Number of the next hop. The
        semantics of this object are determined by the
        routing protocol specified in the route's
        mplsL3VpnVrfRteInetCidrProto value. When this
        object is unknown or not relevant, its value should
        be set to zero."
    DEFVAL { 0 }
    ::= { mplsL3VpnVrfRteEntry 11 }

mplsL3VpnVrfRteInetCidrMetric1 OBJECT-TYPE
    SYNTAX      Integer32 (-1 | 0..2147483647)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The primary routing metric for this route. The
        semantics of this metric are determined by the
```

```
        routing protocol specified in the route's
        mplsL3VpnVrfRteInetCidrProto value.  If this
        metric is not used, its value should be set to
        -1."
    DEFVAL { -1 }
    ::= { mplsL3VpnVrfRteEntry 12 }

mplsL3VpnVrfRteInetCidrMetric2 OBJECT-TYPE
    SYNTAX      Integer32 (-1 | 0..2147483647)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "An alternate routing metric for this route.  The
        semantics of this metric are determined by the routing
        protocol specified in the route's
        mplsL3VpnVrfRteInetCidrProto
        value.  If this metric is not used, its value should be
        set to -1."
    DEFVAL { -1 }
    ::= { mplsL3VpnVrfRteEntry 13 }

mplsL3VpnVrfRteInetCidrMetric3 OBJECT-TYPE
    SYNTAX      Integer32 (-1 | 0..2147483647)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "An alternate routing metric for this route.  The
        semantics of this metric are determined by the routing
        protocol specified in the route's
        mplsL3VpnVrfRteInetCidrProto
        value.  If this metric is not used, its value should be
        set to -1."
    DEFVAL { -1 }
    ::= { mplsL3VpnVrfRteEntry 14 }

mplsL3VpnVrfRteInetCidrMetric4 OBJECT-TYPE
    SYNTAX      Integer32 (-1 | 0..2147483647)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "An alternate routing metric for this route.  The
        semantics of this metric are determined by the routing
        protocol specified in the route's
        mplsL3VpnVrfRteInetCidrProto value.  If this metric
        is not used, its value should be set to -1."
    DEFVAL { -1 }
    ::= { mplsL3VpnVrfRteEntry 15 }
```

```

mplsL3VpnVrfRteInetCidrMetric5 OBJECT-TYPE
    SYNTAX      Integer32 (-1 | 0..2147483647)
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "An alternate routing metric for this route.  The
        semantics of this metric are determined by the routing
        protocol specified in the route's
        mplsL3VpnVrfRteInetCidrProto value.  If this metric is
        not used, its value should be set to -1."
    DEFVAL { -1 }
    ::= { mplsL3VpnVrfRteEntry 16 }

```

```

mplsL3VpnVrfRteXCPointer OBJECT-TYPE
    SYNTAX      MplsIndexType
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Index into mplsXCTable that identifies which cross-
        connect entry is associated with this VRF route entry
        by containing the mplsXCIndex of that cross-connect entry.
        The string containing the single-octet 0x00 indicates that
        a label stack is not associated with this route entry.  This
        can be the case because the label bindings have not yet
        been established, or because some change in the agent has
        removed them.

```

When the label stack associated with this VRF route is created, it MUST establish the associated cross-connect entry in the mplsXCTable and then set that index to the value of this object. Changes to the cross-connect object in the mplsXCTable MUST automatically be reflected in the value of this object. If this object represents a static routing entry, then the manager must ensure that this entry is maintained consistently in the corresponding mplsXCTable as well."

#### REFERENCE

"RFC 3813 - Multiprotocol Label Switching (MPLS) Label Switching Router (LSR) Management Information base (MIB), C. Srinivasan, A. Vishwanathan, and T. Nadeau, June 2004"

```

::= { mplsL3VpnVrfRteEntry 17 }

```

```

mplsL3VpnVrfRteInetCidrStatus OBJECT-TYPE
    SYNTAX      RowStatus
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The row status variable, used according to row
        installation and removal conventions.

```

A row entry cannot be modified when the status is marked as active(1)."  
 ::= { mplsL3VpnVrfRteEntry 18 }

-- MPLS L3VPN Notifications

mplsL3VpnVrfUp NOTIFICATION-TYPE

OBJECTS { mplsL3VpnIfConfRowStatus,  
 mplsL3VpnVrfOperStatus  
 }

STATUS current

DESCRIPTION

"This notification is generated when:

a. No interface is associated with this VRF, and the first (and only first) interface associated with it has its ifOperStatus change to up(1).

b. One interface is associated with this VRF, and the ifOperStatus of this interface changes to up(1).

c. Multiple interfaces are associated with this VRF, and the ifOperStatus of all interfaces is down(2), and the first of those interfaces has its ifOperStatus change to up(1)."

::= { mplsL3VpnNotifications 1 }

mplsL3VpnVrfDown NOTIFICATION-TYPE

OBJECTS { mplsL3VpnIfConfRowStatus,  
 mplsL3VpnVrfOperStatus  
 }

STATUS current

DESCRIPTION

"This notification is generated when:

a. One interface is associated with this VRF, and the ifOperStatus of this interface changes from up(1) to down(2).

b. Multiple interfaces are associated with this VRF, and the ifOperStatus of all except one of these interfaces is equal to up(1), and the ifOperStatus of that interface changes from up(1) to down(2).

c. The last interface with ifOperStatus equal to up(1) is disassociated from a VRF."

::= { mplsL3VpnNotifications 2 }

mplsL3VpnVrfRouteMidThreshExceeded NOTIFICATION-TYPE

OBJECTS { mplsL3VpnVrfPerfCurrNumRoutes,  
 mplsL3VpnVrfConfMidRteThresh

```

    }
STATUS      current
DESCRIPTION

```

"This notification is generated when the number of routes contained by the specified VRF exceeds the value indicated by mplsL3VpnVrfMidRouteThreshold. A single notification MUST be generated when this threshold is exceeded, and no other notifications of this type should be issued until the value of mplsL3VpnVrfPerfCurrNumRoutes has fallen below that of mplsL3VpnVrfConfMidRteThresh."

```
 ::= { mplsL3VpnNotifications 3 }
```

mplsL3VpnVrfNumVrfRouteMaxThreshExceeded NOTIFICATION-TYPE

```

OBJECTS      { mplsL3VpnVrfPerfCurrNumRoutes,
                mplsL3VpnVrfConfHighRteThresh
              }
STATUS      current
DESCRIPTION

```

"This notification is generated when the number of routes contained by the specified VRF exceeds or attempts to exceed the maximum allowed value as indicated by mplsL3VpnVrfMaxRouteThreshold. In cases where mplsL3VpnVrfConfHighRteThresh is set to the same value as mplsL3VpnVrfConfMaxRoutes, mplsL3VpnVrfConfHighRteThresh need not be exceeded; rather, just reached for this notification to be issued.

Note that mplsL3VpnVrfConfRteMxThrshTime denotes the interval at which the this notification will be reissued after the maximum value has been exceeded (or reached if mplsL3VpnVrfConfMaxRoutes and mplsL3VpnVrfConfHighRteThresh are equal) and the initial notification has been issued. This value is intended to prevent continuous generation of notifications by an agent in the event that routes are continually added to a VRF after it has reached its maximum value. The default value is 0 minutes. If this value is set to 0, the agent should only issue a single notification at the time that the maximum threshold has been reached, and should not issue any more notifications until the value of routes has fallen below the configured threshold value."

```
 ::= { mplsL3VpnNotifications 4 }
```

mplsL3VpnNumVrfSecIlleglLblThrshExcd NOTIFICATION-TYPE

```

OBJECTS      { mplsL3VpnVrfSecIllegalLblVltns }
STATUS      current
DESCRIPTION

```

"This notification is generated when the number of illegal label violations on a VRF as indicated by

```

mplsL3VpnVrfSecIllegalLblVltns has exceeded
mplsL3VpnIl1lLblRcvThrsh. The threshold is not
included in the varbind here because the value of
mplsL3VpnVrfSecIllegalLblVltns should be one greater than
the threshold at the time this notification is issued."
 ::= { mplsL3VpnNotifications 5 }

```

#### mplsL3VpnNumVrfRouteMaxThreshCleared NOTIFICATION-TYPE

```

OBJECTS      { mplsL3VpnVrfPerfCurrNumRoutes,
                mplsL3VpnVrfConfHighRteThresh
              }

```

```

STATUS      current

```

#### DESCRIPTION

"This notification is generated only after the number of routes contained by the specified VRF exceeds or attempts to exceed the maximum allowed value as indicated by mplsVrfMaxRouteThreshold, and then falls below this value. The emission of this notification informs the operator that the error condition has been cleared without the operator having to query the device.

Note that mplsL3VpnVrfConfRteMxThrshTime denotes the interval at which the mplsNumVrfRouteMaxThreshExceeded notification will be reissued after the maximum value has been exceeded (or reached if mplsL3VpnVrfConfMaxRoutes and mplsL3VpnVrfConfHighRteThresh are equal) and the initial notification has been issued. Therefore, the generation of this notification should also be emitted with the same frequency (assuming that the error condition is cleared). Specifically, if the error condition is reached and cleared several times during the period of time specified in mplsL3VpnVrfConfRteMxThrshTime, only a single notification will be issued to indicate the first instance of the error condition as well as the first time the error condition is cleared. This behavior is intended to prevent continuous generation of notifications by an agent in the event that routes are continually added and removed to/from a VRF after it has reached its maximum value. The default value is 0. If this value is set to 0, the agent should issue a notification whenever the maximum threshold has been cleared."

```

 ::= { mplsL3VpnNotifications 6 }

```

#### -- Conformance Statement

##### mplsL3VpnGroups

```

    OBJECT IDENTIFIER ::= { mplsL3VpnConformance 1 }

```

##### mplsL3VpnCompliances

OBJECT IDENTIFIER ::= { mplsL3VpnConformance 2 }

-- Module Compliance

mplsL3VpnModuleFullCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"Compliance statement for agents that provide full support  
for the MPLS-L3VPN-STD-MIB"

MODULE -- this module

MANDATORY-GROUPS { mplsL3VpnScalarGroup,  
mplsL3VpnVrfGroup,  
mplsL3VpnIfGroup,  
mplsL3VpnPerfGroup,  
mplsL3VpnVrfRteGroup,  
mplsL3VpnVrfRTGroup,  
mplsL3VpnSecGroup,  
mplsL3VpnNotificationGroup  
}

GROUP mplsL3VpnPerfRouteGroup

DESCRIPTION "This group is only mandatory for LSRs that  
support tracking the number of routes attempted  
to be added to VRFs."

OBJECT mplsL3VpnIfConfRowStatus

SYNTAX RowStatus { active(1), notInService(2) }

WRITE-SYNTAX RowStatus { active(1), notInService(2),  
createAndGo(4), destroy(6)  
}

DESCRIPTION "Support for createAndWait and notReady is  
not required."

OBJECT mplsL3VpnVrfConfRowStatus

SYNTAX RowStatus { active(1), notInService(2) }

WRITE-SYNTAX RowStatus { active(1), notInService(2),  
createAndGo(4), destroy(6)  
}

DESCRIPTION "Support for createAndWait and notReady is  
not required."

OBJECT mplsL3VpnVrfRTRowStatus

SYNTAX RowStatus { active(1), notInService(2) }

WRITE-SYNTAX RowStatus { active(1), notInService(2),  
createAndGo(4), destroy(6)  
}

DESCRIPTION "Support for createAndWait and notReady is  
not required."



```
::= { mplsL3VpnCompliances 1 }
```

```
--
```

```
-- ReadOnly Compliance
```

```
--
```

```
mplsL3VpnModuleReadOnlyCompliance MODULE-COMPLIANCE
```

```
    STATUS current
```

```
    DESCRIPTION "Compliance requirement for implementations that only
        provide read-only support for MPLS-L3VPN-STD-MIB.
        Such devices can then be monitored but cannot be
        configured using this MIB module."
```

```
    MODULE -- this module
```

```
        MANDATORY-GROUPS { mplsL3VpnScalarGroup,
                            mplsL3VpnVrfGroup,
                            mplsL3VpnIfGroup,
                            mplsL3VpnPerfGroup,
                            mplsL3VpnVrfRteGroup,
                            mplsL3VpnVrfRTGroup,
                            mplsL3VpnSecGroup,
                            mplsL3VpnNotificationGroup
                            }
```

```
GROUP      mplsL3VpnPerfRouteGroup
```

```
DESCRIPTION "This group is only mandatory for LSRs that
        support tracking the number of routes attempted to
        be added to VRFs."
```

```
OBJECT      mplsL3VpnIfConfRowStatus
```

```
SYNTAX      RowStatus { active(1) }
```

```
MIN-ACCESS  read-only
```

```
DESCRIPTION "Write access is not required."
```

```
OBJECT      mplsL3VpnVrfConfRowStatus
```

```
SYNTAX      RowStatus { active(1) }
```

```
MIN-ACCESS  read-only
```

```
DESCRIPTION "Write access is not required."
```

```
OBJECT      mplsL3VpnVrfRTRowStatus
```

```
SYNTAX      RowStatus { active(1) }
```

```
MIN-ACCESS  read-only
```

```
DESCRIPTION "Write access is not required."
```

```
OBJECT      mplsL3VpnIfVpnClassification
```

```
MIN-ACCESS  read-only
```

```
DESCRIPTION "Write access is not required."
```

OBJECT	mplsL3VpnIfVpnRouteDistProtocol
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnIfConfStorageType
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfVpnId
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfDescription
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfRD
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfConfMidRteThresh
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfConfHighRteThresh
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfConfMaxRoutes
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfConfStorageType
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfRT
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfRTDescr
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."
OBJECT	mplsL3VpnVrfRTStorageType
MIN-ACCESS	read-only
DESCRIPTION	"Write access is not required."

```

OBJECT      mplsL3VpnVrfRteInetCidrIfIndex
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrType
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrNextHopAS
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrMetric1
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrMetric2
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrMetric3
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrMetric4
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrMetric5
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteXCPointer
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

OBJECT      mplsL3VpnVrfRteInetCidrStatus
SYNTAX      RowStatus { active(1) }
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."
 ::= { mplsL3VpnCompliances 2 }

```

```
-- Units of conformance.
```

```

mplsL3VpnScalarGroup OBJECT-GROUP
    OBJECTS { mplsL3VpnConfiguredVrfs,
              mplsL3VpnActiveVrfs,
              mplsL3VpnConnectedInterfaces,

```

```

        mplsL3VpnNotificationEnable,
        mplsL3VpnVrfConfMaxPossRts,
        mplsL3VpnVrfConfRteMxThrshTime,
        mplsL3VpnI11L1RcvThrsh
    }
    STATUS current
    DESCRIPTION
        "Collection of scalar objects required for MPLS VPN
        management."
    ::= { mplsL3VpnGroups 1 }

mplsL3VpnVrfGroup OBJECT-GROUP
    OBJECTS { mplsL3VpnVrfVpnId,
        mplsL3VpnVrfDescription,
        mplsL3VpnVrfRD,
        mplsL3VpnVrfCreationTime,
        mplsL3VpnVrfOperStatus,
        mplsL3VpnVrfActiveInterfaces,
        mplsL3VpnVrfAssociatedInterfaces,
        mplsL3VpnVrfConfMidRteThresh,
        mplsL3VpnVrfConfHighRteThresh,
        mplsL3VpnVrfConfMaxRoutes,
        mplsL3VpnVrfConfLastChanged,
        mplsL3VpnVrfConfRowStatus,
        mplsL3VpnVrfConfAdminStatus,
        mplsL3VpnVrfConfStorageType
    }
    STATUS current
    DESCRIPTION
        "Collection of objects needed for MPLS VPN VRF
        management."
    ::= { mplsL3VpnGroups 2 }

mplsL3VpnIfGroup OBJECT-GROUP
    OBJECTS { mplsL3VpnIfVpnClassification,
        mplsL3VpnIfVpnRouteDistProtocol,
        mplsL3VpnIfConfStorageType,
        mplsL3VpnIfConfRowStatus
    }
    STATUS current
    DESCRIPTION
        "Collection of objects needed for MPLS VPN interface
        management."
    ::= { mplsL3VpnGroups 3 }

mplsL3VpnPerfGroup OBJECT-GROUP
    OBJECTS { mplsL3VpnVrfPerfRoutesAdded,
        mplsL3VpnVrfPerfRoutesDeleted,

```

```
        mplsL3VpnVrfPerfCurrNumRoutes
    }
    STATUS current
    DESCRIPTION
        "Collection of objects needed for MPLS VPN
        performance information."
    ::= { mplsL3VpnGroups 4 }

mplsL3VpnPerfRouteGroup OBJECT-GROUP
    OBJECTS { mplsL3VpnVrfPerfRoutesDropped,
              mplsL3VpnVrfPerfDiscTime
            }
    STATUS current
    DESCRIPTION
        "Collection of objects needed to track MPLS VPN
        routing table dropped routes."
    ::= { mplsL3VpnGroups 5 }

mplsL3VpnSecGroup OBJECT-GROUP
    OBJECTS { mplsL3VpnVrfSecIllegalLblVltns,
              mplsL3VpnVrfSecDiscontinuityTime
            }
    STATUS current
    DESCRIPTION
        "Collection of objects needed for MPLS VPN
        security-related information."
    ::= { mplsL3VpnGroups 7 }

mplsL3VpnVrfRteGroup OBJECT-GROUP
    OBJECTS {
        mplsL3VpnVrfRteInetCidrIfIndex,
        mplsL3VpnVrfRteInetCidrType,
        mplsL3VpnVrfRteInetCidrProto,
        mplsL3VpnVrfRteInetCidrAge,
        mplsL3VpnVrfRteInetCidrNextHopAS,
        mplsL3VpnVrfRteInetCidrMetric1,
        mplsL3VpnVrfRteInetCidrMetric2,
        mplsL3VpnVrfRteInetCidrMetric3,
        mplsL3VpnVrfRteInetCidrMetric4,
        mplsL3VpnVrfRteInetCidrMetric5,
        mplsL3VpnVrfRteXCPointer,
        mplsL3VpnVrfRteInetCidrStatus
    }
    STATUS current
    DESCRIPTION
        "Objects required for VRF route table management."
    ::= { mplsL3VpnGroups 8 }

mplsL3VpnVrfRTGroup OBJECT-GROUP
```

```

OBJECTS { mplsL3VpnVrfRTDescr,
           mplsL3VpnVrfRT,
           mplsL3VpnVrfRTRowStatus,
           mplsL3VpnVrfRTStorageType
         }
STATUS current
DESCRIPTION
    "Objects required for VRF route target management."
 ::= { mplsL3VpnGroups 9 }

mplsL3VpnNotificationGroup NOTIFICATION-GROUP
    NOTIFICATIONS { mplsL3VpnVrfUp,
                    mplsL3VpnVrfDown,
                    mplsL3VpnVrfRouteMidThreshExceeded,
                    mplsL3VpnVrfNumVrfRouteMaxThreshExceeded,
                    mplsL3VpnNumVrfSecIllglLblThrshExcd,
                    mplsL3VpnNumVrfRouteMaxThreshCleared
                  }
    STATUS current
    DESCRIPTION
        "Objects required for MPLS VPN notifications."
 ::= { mplsL3VpnGroups 10 }
END
-- End of MPLS-VPN-MIB

```

## 8. Security Considerations

It is clear that these MIB modules are potentially useful for monitoring of MPLS LSRs supporting L3 MPLS VPN. This MIB module can also be used for configuration of certain objects, and anything that can be configured can be incorrectly configured, with potentially disastrous results.

There are a number of management objects defined in this MIB module with a MAX-ACCESS clause of read-write and/or read-create. 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 can have a negative effect on network operations. These are the tables and objects and their sensitivity/vulnerability:

- o the mplsL3VpnVrfRouteTable, mplsL3VpnIfConfTable, and mplsL3VpnVrfTable tables collectively contain objects that may be used to provision MPLS VRF interfaces and configuration. Unauthorized access to objects in these tables could result in disruption of traffic on the network. This is especially true if these VRFs have been previously provisioned and are in use.

The use of stronger mechanisms such as SNMPv3 security should be considered where possible. Specifically, SNMPv3 VACM and USM MUST be used with any v3 agent that implements this MIB module. Administrators should consider whether read access to these objects should be allowed, since read access may be undesirable under certain circumstances.

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:

- o the mplsL3VpnVrfTable, mplsL3VpnIfConfTable tables collectively show the VRF interfaces and associated VRF configurations as well as their linkages to other MPLS-related configuration and/or performance statistics. Administrators not wishing to reveal this information should consider these objects sensitive/vulnerable and take precautions so they are not revealed.

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.

## 9. IANA Considerations

As described in MPLS-TC-STD-MIB [RFC3811], MPLS related standards track MIB modules should be rooted under the mplsStdMIB subtree. There is one MPLS-related MIB module contained in this document. The following subsection requests IANA for a new assignment under the mplsStdMIB subtree. New assignments can only be made via a Standards Action as specified in [RFC2434].

### 9.1. IANA Considerations for MPLS-L3VPN-STD-MIB

The IANA has assigned { mplsStdMIB 11 } to the MPLS-L3VPN-STD-MIB module specified in this document.

## 10. Dedication

Steve Brannon passed away suddenly on January 30, 2001. We would like to dedicate our efforts in this area and this document to his memory.

## 11. Acknowledgements

This document has benefited from discussions and input from Bill Fenner, Gerald Ash, Sumit Mukhopadhyay, Mike Piecuch, and Joan Weiss.

## 12. References

### 12.1. Normative References

- [RFC2119] Bradner, S., "Key Words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC3811] Nadeau, T. and J. Cucchiara, "Definition of Textual Conventions and for Multiprotocol Label Switching (MPLS) Management", RFC 3811, June 2004.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.
- [RFC4364] Rosen, E. and Y. Rekhter, "BGP/MPLS IP Virtual Private Networks (VPNs)", RFC 4364, February 2006.
- [RFC2685] Fox B., et al, "Virtual Private Networks Identifier", RFC 2685, September 1999.



- [RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.
- [RFC3813] Srinivasan, C., Viswanathan, A., and T. Nadeau, "MPLS Multiprotocol Label Switching (MPLS) Label Switch Router Management Information Base ", RFC 3813, June 2004
- [RFC3812] Srinivasan, C., Viswanathan, A., and T. Nadeau, "Multiprotocol Label Switching (MPLS) Traffic Engineering (TE) Management Information Base (MIB)", RFC 3812, June 2004.
- [RFC2096] Baker, F., "IP Forwarding Table MIB", RFC 2096, January 1997.
- [RFC4265] Schliesser, B. and T. Nadeau, "Definition of Textual Conventions for Virtual Private Network (VPN) Management", RFC 4265, November 2005.
- [RFC4001] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 4001, February 2005.
- [RFC2863] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.
- [RTPROTO] IANA, "IP Route Protocol MIB", <http://www.iana.org/assignments/ianaiprouteprotocol-mib>, September 2000.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

## 12.2. 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.
- [RFC3413] Levi, D., Meyer, P., and B. Stewart, "Simple Network  
Management Protocol (SNMP) Applications", STD 62, RFC  
3413, December 2002.
- [RFC2434] Narten, T. and H. Alvestrand., "Guidelines for Writing an  
IANA Considerations Section in RFCs", BCP 26, RFC 2434,  
October 1998.

## 13. Contributors' Addresses

Luyuan Fang  
AT&T  
200 Laurel Ave  
Middletown, NJ 07748

Phone: +1-732-420-1921  
EMail: luyuanfang@att.com

Martin Tatham  
British Telecom  
BT Adastal Park,  
Martlesham Heath,  
Ipswich, IP5 3RE  
UK

Phone: +44 1473 606349  
Fax: +44 1473 606727  
EMail: martin.tatham@bt.com

Fabio M. Chiussi  
Bell Laboratories,  
Lucent Technologies  
101 Crawfords Corner Road  
Room 4D-521  
Holmdel, NJ 07733

Phone: +1-732-949-2407  
EMail: fabio@bell-labs.com

Joseph Dube  
Avici Systems, Inc.  
101 Billerica Avenue

North Billerica, MA 01862

#### Editors' Addresses

Thomas D. Nadeau  
Cisco Systems, Inc.  
1414 Massachusetts Ave.  
Boxborough, MA 01719

Phone: +1-978-936-1470  
EMail: tnadeau@cisco.com

Harmen van der Linde  
Cisco Systems, Inc.  
1414 Massachusetts Ave.  
Boxborough, MA 01719

Phone: +1-732-420-1916  
EMail: havander@cisco.com

## Full Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

## Intellectual Property

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at <http://www.ietf.org/ipr>.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

## Acknowledgement

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).

