

Network Working Group
Request for Comments: 3812
Category: Standards Track

C. Srinivasan
Bloomberg L.P.
A. Viswanathan
Forcel0 Networks, Inc.
T. Nadeau
Cisco Systems, Inc.
June 2004

Multiprotocol Label Switching (MPLS) Traffic Engineering (TE)
Management Information Base (MIB)

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 (2004).

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 for Multiprotocol Label Switching (MPLS) based traffic engineering (TE).

Table of Contents

1.	Introduction	2
2.	Terminology.	2
3.	The Internet-Standard Management Framework	3
4.	Feature List	3
5.	Outline.	3
5.1.	Summary of Traffic Engineering MIB Module.	4
6.	Brief Description of MIB Objects	4
6.1.	mplsTunnelTable.	4
6.2.	mplsTunnelResourceTable.	5
6.3.	mplsTunnelHopTable	5
6.4.	mplsTunnelARHopTable	5
6.5.	mplsTunnelCHopTable.	5
6.6.	mplsTunnelPerfTable.	6
6.7.	mplsTunnelCRLDPResTable.	6
7.	Use of 32-bit and 64-bit Counters.	6

8.	Application of the Interface Group to MPLS Tunnels	6
8.1.	Support of the MPLS Tunnel Interface by ifTable.	7
9.	Example of Tunnel Setup.	8
10.	The Use of RowPointer.	11
11.	MPLS Traffic Engineering MIB Definitions	11
12.	Security Considerations.	63
13.	Acknowledgments.	64
14.	IANA Considerations.	64
14.1.	IANA Considerations for MPLS-TE-STD-MIB.	65
15.	References	65
15.1.	Normative References	65
15.2.	Informative References	66
16.	Authors' Addresses	67
17.	Full Copyright Statement	68

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 for modeling a Multiprotocol Label Switching (MPLS) [RFC3031] based traffic engineering. This MIB module should be used in conjunction with the companion document [RFC3813] for MPLS based traffic engineering configuration and management.

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 BCP 14, RFC 2119, reference [RFC2119].

2. Terminology

This document uses terminology from the MPLS architecture document [RFC3031] and MPLS Label Switch Router MIB [RFC3813]. Some frequently used terms are described next.

An explicitly routed LSP (ERLSP) is referred to as an MPLS tunnel. It consists of in-segment(s) and/or out-segment(s) at the egress/ingress LSRs, each segment being associated with one MPLS interface. These are also referred to as tunnel segments. Additionally, at an intermediate LSR, we model a connection as consisting of one or more in-segments and/or one or more out-segments. The binding or interconnection between in-segments and out-segments is performed using a cross-connect. These objects are defined in the MPLS Label Switch Router MIB [RFC3813].

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. Feature List

The MPLS traffic engineering MIB module is designed to satisfy the following requirements and constraints:

- The MIB module supports configuration of point-to-point unidirectional tunnels.
- MPLS tunnels need not be interfaces, but it is possible to configure a tunnel as an interface.
- The MIB module supports tunnel establishment via an MPLS signalling protocol wherein the tunnel parameters are specified using this MIB module at the head end of the LSP, and end-to-end tunnel LSP establishment is accomplished via signalling. The MIB module also supports manually configured tunnels, i.e., those for which label associations at each hop of the tunnel LSP are provisioned by the administrator via the LSR MIB [RFC3813].
- The MIB module supports persistent, as well as non-persistent tunnels.

5. Outline

Traffic engineering support for MPLS tunnels requires the following configuration:

- Setting up MPLS tunnels along with appropriate configuration parameters.
- Configuring tunnel for loose and strict source routed hops.

These actions may need to be accompanied by corresponding actions using [RFC3813] to establish and configure tunnel segments, if this is done manually. Also, the in-segment and out-segment performance tables, `mplsInSegmentPerfTable`, and `mplsOutSegmentPerfTable` [RFC3813], should be used to determine performance of the tunnels and tunnel segments, in addition to `mplsTunnelPerfTable` in this MIB module.

5.1. Summary of Traffic Engineering MIB Module

The MIB module objects for performing these actions consist of the following tables:

- Tunnel table (`mplsTunnelTable`) for setting up MPLS tunnels.
- Resource table (`mplsTunnelResourceTable`) for setting up the tunnel resources.
- Tunnel specified, actual, and computed hop tables (`mplsTunnelHopTable`, `mplsTunnelARHopTable`, and `mplsTunnelCHopTable`) for strict and loose source routed MPLS tunnel hops.
- Tunnel performance table (`mplsTunnelPerfTable`) for measuring tunnel performance.
- CRLDP resource table (`mplsTunnelCRLDPResTable`) for specifying resource objects applicable to tunnels signaled using CRLDP.

These tables are described in the subsequent sections.

6. Brief Description of MIB Objects

The objects described in this section support the functionality described in documents [RFC3209] and [RFC3212]. The tables support both manually configured and signaled tunnels.

6.1. `mplsTunnelTable`

The `mplsTunnelTable` allows new MPLS tunnels to be created between an MPLS LSR and a remote endpoint, and existing tunnels to be reconfigured or removed. Note that we only support point-to-point tunnels, although multipoint-to-point and point-to-multipoint connections are supported by an LSR acting as a cross-connect. Each MPLS tunnel can thus have one out-segment originating at an LSR and/or one in-segment terminating at that LSR.

`mplsTunnelTable` does not define the in and out segments forming the tunnel. Instead, these are defined by creating rows in the in-segment and out-segment tables, defining relationships in the cross-connect table, and referring to these rows in the `mplsTunnelTable` using a cross-connect index, `mplsTunnelXCIndex`. These segment and cross-connect related objects are defined in [RFC3813].

6.2. `mplsTunnelResourceTable`

`mplsTunnelResourceTable` is used to indicate the resources required for a tunnel. Multiple tunnels may share the same resources by pointing to the same entry in this table. Tunnels that do not share resources must point to separate entries in this table.

6.3. `mplsTunnelHopTable`

`mplsTunnelHopTable` is used to indicate the hops, strict or loose, for an MPLS tunnel defined in `mplsTunnelTable`, when it is established via signalling. Multiple tunnels may share the same hops by pointing to the same entry in this table. Each row also has a secondary index, `mplsTunnelHopIndex`, corresponding to the next hop of this tunnel. The scalar `mplsTunnelMaxHops` indicates the maximum number of hops that can be specified on each tunnel supported by this LSR.

At transit LSRs, this table contains the hops, strict or loose, that apply to the downstream part of this tunnel only. This corresponds to the requested path received through the signaling protocol.

6.4. `mplsTunnelARHopTable`

`mplsTunnelARHopTable` is used to indicate the actual hops traversed by a tunnel as reported by the MPLS signalling protocol after the tunnel is setup. The support of this table is optional since not all MPLS signalling protocols may support this feature.

At transit LSRs, this table contains the actual hops traversed by the tunnel along its entire length if that information is available. This corresponds to the recorded path reported by the MPLS signalling protocol, possibly derived from multiple signaling messages.

6.5. `mplsTunnelCHoptable`

`mplsTunnelCHoptable` lists the actual hops computed by a constraint-based routing algorithm based on the `mplsTunnelHopTable` for the MPLS signalling protocol in use. The support of this table is optional since not all implementations may support computation of hop lists using a constraint-based routing protocol.

At transit LSRs, this table contains the hops computed to apply to the downstream part of this tunnel. This corresponds to the requested path signaled from this LSR through the signaling protocol.

6.6. mplsTunnelPerfTable

mplsTunnelPerfTable provides several counters to measure the performance of the MPLS tunnels. This table augments mplsTunnelTable.

6.7. mplsTunnelCRLDPResTable

mplsTunnelCRLDPResTable contains resource information for those tunnels that are signaled using CRLDP [RFC3212]. This is a sparse extension to mplsTunnelResourceTable and is also indexed by mplsTunnelResourceIndex. As with mplsTunnelResourceTable, multiple tunnels may share the same resources by pointing to the same entry in this table. Tunnels that do not share resources must point to separate entries in this table. The mplsTunnelCRLDPResTable may be supported only by implementations that support the CR-LDP signaling protocol.

7. Use of 32-bit and 64-bit Counters

64-bit counters are provided in this MIB module for high-speed interfaces where the use of 32-bit counters might be impractical. The requirements on the use of 32-bit and 64-bit counters (copied verbatim from [RFC2863]) are as follows:

For interfaces that operate at 20,000,000 (20 million) bits per second or less, 32-bit byte and packet counters MUST be supported. For interfaces that operate faster than 20,000,000 bits/second, and slower than 650,000,000 bits/second, 32-bit packet counters MUST be supported and 64-bit octet counters MUST be supported. For interfaces that operate at 650,000,000 bits/second or faster, 64-bit packet counters AND 64-bit octet counters MUST be supported.

8. Application of the Interface Group to MPLS Tunnels

The Interfaces Group of MIB II defines generic managed objects for managing interfaces. This memo contains the media-specific extensions to the Interfaces Group for managing MPLS Tunnels as logical interfaces.

This memo assumes the interpretation of the Interfaces Group to be in accordance with [RFC2863] which states that the interfaces table (ifTable) contains information on the managed resource's interfaces and that each sub-layer below the internetwork layer of a network

interface is considered an interface. Thus, the MPLS interface is represented as an entry in the ifTable. The inter-relation of entries in the ifTable is defined by the Interfaces Stack Group defined in [RFC2863].

When using MPLS Tunnels as interfaces, the interface stack table might appear as follows:

```

+-----+
| MPLS tunnel interface ifType = mplsTunnel(150) |
+-----+
|           MPLS interface ifType = mpls(166)           |
+-----+
|                               Underlying layer                               |
+-----+

```

In the above diagram, "Underlying Layer" refers to the ifIndex of any interface type for which MPLS internetworking has been defined. Examples include ATM, Frame Relay, and Ethernet.

8.1. Support of the MPLS Tunnel Interface by ifTable

Some specific interpretations of the ifTable for those MPLS tunnels represented as interfaces follow:

Object	Use for the MPLS tunnel.
ifIndex	Each MPLS tunnel is represented by an ifEntry.
ifDescr	Description of the MPLS tunnel.
ifType	The value that is allocated for the MPLS tunnel is 150.
ifSpeed	The total bandwidth in bits per second for use by the MPLS tunnel.
ifPhysAddress	Unused.
ifAdminStatus	See [RFC2863].
ifOperStatus	This value reflects the actual operational status of the MPLS tunnel. Assumes the value down(2) if the MPLS tunnel is down.
ifLastChange	See [RFC2863].

ifInOctets	The number of octets received over the MPLS tunnel.
ifOutOctets	The number of octets transmitted over the MPLS tunnel.
ifInErrors	The number of labeled packets dropped due to uncorrectable errors.
ifInUnknownProtos	The number of received packets discarded during packet header validation, including packets with unrecognized label values.
ifOutErrors	See [RFC2863].
ifName	Textual name (unique on this system) of the MPLS tunnel or an octet string of zero length.
ifLinkUpDownTrapEnable	Default is disabled (2).
ifConnectorPresent	Set to false (2).
ifHighSpeed	See [RFC2863].
ifHCInOctets	The 64-bit version of ifInOctets; supported if required by the compliance statements in [RFC2863].
ifHCOctets	The 64-bit version of ifOutOctets; supported if required by the compliance statements in [RFC2863].
ifAlias	The non-volatile 'alias' name for the MPLS tunnel as specified by a network manager.

9. Example of Tunnel Setup

This section contains an example of which MIB objects should be modified if one would like to create a best effort, loosely routed, unidirectional traffic engineered tunnel, which spans two hops of a simple network. Note that these objects should be created on the "head-end" LSR. Those objects relevant to illustrating the relationships amongst different tables are shown here. Other objects may be needed before conceptual row activation can happen.

The RowStatus values shown in this section are those to be used in the set request, typically createAndGo(4) which is used to create the conceptual row and have its status immediately set to active. A subsequent retrieval operation on the conceptual row will return a different value, such as active(1). Please see [RFC2579] for a detailed discussion on the use of RowStatus.

In mplsTunnelResourceTable:

```
{
  mplsTunnelResourceIndex          = 5,
  mplsTunnelResourceMaxRate        = 0,
  mplsTunnelResourceMeanRate       = 0,
  mplsTunnelResourceMaxBurstSize   = 0,
  mplsTunnelResourceMeanBurstSize  = 0,
  mplsTunnelResourceExBurstSize    = 0,
  mplsTunnelResourceExBurstSize    = unspecified (1),
  mplsTunnelResourceWeight         = 0,
  -- Mandatory parameters needed to activate the row go here
  mplsTunnelResourceRowStatus      = createAndGo (4)
}
```

The next two instances of mplsTunnelHopEntry are used to denote the hops this tunnel will take across the network.

The following denotes the beginning of the tunnel, or the first hop. We have used the fictitious LSR identified by "192.168.100.1" as our example head-end router.

In mplsTunnelHopTable:

```
{
  mplsTunnelHopListIndex          = 1,
  mplsTunnelPathOptionIndex       = 1,
  mplsTunnelHopIndex              = 1,
  mplsTunnelHopAddrType           = ipv4 (1),
  mplsTunnelHopIpAddr             = "192.168.100.1",
  mplsTunnelHopIpPrefixLen        = 32,
  mplsTunnelHopType                = strict (2),
  mplsTunnelHopInclude             = true (1),
  mplsTunnelHopPathOptionName     = "Here to there",
  mplsTunnelHopEntryPathComp      = explicit (2),
  -- Mandatory parameters needed to activate the row go here
  mplsTunnelHopRowStatus          = createAndGo (4)
}
```

The following denotes the end of the tunnel, or the last hop in our example. We have used the fictitious LSR identified by "192.168.101.1" as our end router.

In mplsTunnelHopTable:

```
{
  mplsTunnelHopListIndex      = 1,
  mplsTunnelPathOptionIndex   = 1,
  mplsTunnelHopIndex          = 2,
  mplsTunnelHopAddrType       = ipv4 (1),
  mplsTunnelHopIpAddr         = "192.168.101.1",
  mplsTunnelHopIpPrefixLen    = 32,
  mplsTunnelHopType           = loose (2),
  mplsTunnelHopInclude        = true (1),
  mplsTunnelHopPathOptionName = "Here to there",
  mplsTunnelHopEntryPathComp  = explicit (2),
  -- Mandatory parameters needed to activate the row go here
  mplsTunnelHopRowStatus      = createAndGo (4)
}
```

The following denotes the configured tunnel "head" entry:

In mplsTunnelTable:

```
{
  mplsTunnelIndex              = 1,
  mplsTunnelInstance           = 0,
  mplsTunnelIngressLSRId       = 192.168.100.1,
  mplsTunnelEgressLSRId        = 192.168.101.1,
  mplsTunnelName                = "My first tunnel",
  mplsTunnelDescr               = "Here to there",
  mplsTunnelIsIf                = true (1),
  -- RowPointer MUST point to the first accessible column
  mplsTunnelXCPointer           = 0.0,
  mplsTunnelSignallingProto     = none (1),
  mplsTunnelSetupPrio           = 0,
  mplsTunnelHoldingPrio         = 0,
  mplsTunnelSessionAttributes   = 0,
  mplsTunnelLocalProtectInUse   = false (0),
  -- RowPointer MUST point to the first accessible column
  mplsTunnelResourcePointer     = mplsTunnelResourceMaxRate.5,
  mplsTunnelInstancePriority    = 1,
  mplsTunnelHopTableIndex       = 1,
  mplsTunnelIncludeAnyAffinity  = 0,
  mplsTunnelIncludeAllAffinity  = 0,
  mplsTunnelExcludeAnyAffinity  = 0,
  mplsTunnelPathInUse           = 1,
}
```

```

    mplsTunnelRole          = head (1),
-- Mandatory parameters needed to activate the row go here
    mplsTunnelRowStatus     = createAndGo (4)
}

```

Note that any active or signaled instances of the above tunnel would appear with the same primary `mplsTunnelIndex`, but would have values greater than 0 for `mplsTunnelInstance`. They would also have other objects such as the `mplsTunnelXCPointer` set accordingly.

10. The Use of RowPointer

`RowPointer` is a textual convention used to identify a conceptual row in a conceptual table in a MIB by pointing to the first accessible object. In this MIB module, in `mplsTunnelTable`, the objects `mplsTunnelXCPointer` and `mplsTunnelResourcePointer` are of type `RowPointer`. The object `mplsTunnelXCPointer` points to a specific entry in the `mplsXCTable` [RFC3813]. This entry in the `mplsXCTable` is the associated LSP for the given MPLS tunnel entry. The object `mplsTunnelResourcePointer` points to a specific entry in a traffic parameter table. An example of such a traffic parameter table is `mplsTunnelResourceTable`. It indicates a specific instance of a traffic parameter entry that is associated with a given MPLS tunnel entry. These `RowPointer` objects MUST point to the first instance of the first accessible columnar object in the appropriate conceptual row in order to allow the manager to find the appropriate corresponding entry in either `MPLS-LSR-STD-MIB` [RFC3813] or `MPLS-TE-STD-MIB`. If object `mplsTunnelXCPointer` returns `zeroDotZero`, it implies that there is no LSP associated with that particular instance of tunnel entry. If object `mplsTunnelResourcePointer` returns `zeroDotZero`, it implies that there is no QoS resource associated with that particular instance of tunnel entry.

11. MPLS Traffic Engineering MIB Definitions

```
MPLS-TE-STD-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```

    MODULE-IDENTITY, OBJECT-TYPE, NOTIFICATION-TYPE,
    Integer32, Unsigned32, Counter32, Counter64, TimeTicks,
    zeroDotZero
    FROM SNMPv2-SMI          -- [RFC2578]
    MODULE-COMPLIANCE, OBJECT-GROUP, NOTIFICATION-GROUP
    FROM SNMPv2-CONF        -- [RFC2580]
    TruthValue, RowStatus, RowPointer, StorageType,
    TimeStamp
    FROM SNMPv2-TC          -- [RFC2579]
    InterfaceIndexOrZero, ifGeneralInformationGroup,

```

```

ifCounterDiscontinuityGroup
    FROM IF-MIB -- [RFC2863]
mplsStdMIB, MplsBitRate, MplsBurstSize, MplsLSPID,
MplsTunnelIndex, MplsTunnelInstanceIndex,
MplsTunnelAffinity, MplsExtendedTunnelId, MplsPathIndex,
MplsPathIndexOrZero, MplsOwner, TeHopAddressType,
TeHopAddress, TeHopAddressAS, TeHopAddressUnnum
    FROM MPLS-TC-STD-MIB -- [RFC3811]
SnmpAdminString
    FROM SNMP-FRAMEWORK-MIB -- [RFC3411]
IndexIntegerNextFree
    FROM DIFFSERV-MIB -- [RFC3289]
InetAddressPrefixLength
    FROM INET-ADDRESS-MIB -- [RFC3291]
;

```

mplsTeStdMIB MODULE-IDENTITY

```

LAST-UPDATED
    "200406030000Z" -- June 3, 2004
ORGANIZATION
    "Multiprotocol Label Switching (MPLS) Working Group"
CONTACT-INFO
    "
        Cheenu Srinivasan
        Bloomberg L.P.
        Email: cheenu@bloomberg.net

        Arun Viswanathan
        Force10 Networks, Inc.
        Email: arunv@force10networks.com

        Thomas D. Nadeau
        Cisco Systems, Inc.
        Email: tnadeau@cisco.com
    "

```

Comments about this document should be emailed directly to the MPLS working group mailing list at mpls@uu.net.

DESCRIPTION

"Copyright (C) The Internet Society (2004). The initial version of this MIB module was published in RFC 3812. For full legal notices see the RFC itself or see: <http://www.ietf.org/copyrights/ianamib.html>

This MIB module contains managed object definitions for MPLS Traffic Engineering (TE) as defined in:

1. Extensions to RSVP for LSP Tunnels, Awduche et al, RFC 3209, December 2001
2. Constraint-Based LSP Setup using LDP, Jamoussi

(Editor), RFC 3212, January 2002
 3. Requirements for Traffic Engineering Over MPLS,
 Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M.,
 and J. McManus, [RFC2702], September 1999"

-- Revision history.

REVISION

"200406030000Z" -- June 3, 2004

DESCRIPTION

"Initial version issued as part of RFC 3812."

::= { mplsStdMIB 3 }

-- Top level components of this MIB module.

-- traps

mplsTeNotifications OBJECT IDENTIFIER ::= { mplsTeStdMIB 0 }

-- tables, scalars

mplsTeScalars OBJECT IDENTIFIER ::= { mplsTeStdMIB 1 }

mplsTeObjects OBJECT IDENTIFIER ::= { mplsTeStdMIB 2 }

-- conformance

mplsTeConformance OBJECT IDENTIFIER ::= { mplsTeStdMIB 3 }

-- MPLS Tunnel scalars.

mplsTunnelConfigured OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of tunnels configured on this device. A
 tunnel is considered configured if the
 mplsTunnelRowStatus is active(1)."

::= { mplsTeScalars 1 }

mplsTunnelActive OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"The number of tunnels active on this device. A
 tunnel is considered active if the
 mplsTunnelOperStatus is up(1)."

::= { mplsTeScalars 2 }

mplsTunnelTEDistProto OBJECT-TYPE

```
SYNTAX      BITS {
    other (0),
    ospf (1),
    isis (2)
}
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The traffic engineering distribution protocol(s)
    used by this LSR. Note that an LSR may support more
    than one distribution protocol simultaneously."
 ::= { mplsTeScalars 3 }

mplsTunnelMaxHops OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The maximum number of hops that can be specified for
    a tunnel on this device."
 ::= { mplsTeScalars 4 }

mplsTunnelNotificationMaxRate OBJECT-TYPE
SYNTAX      Unsigned32
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
    "This variable indicates the maximum number of
    notifications issued per second. If events occur
    more rapidly, the implementation may simply fail to
    emit these notifications during that period, or may
    queue them until an appropriate time. A value of 0
    means no throttling is applied and events may be
    notified at the rate at which they occur."
DEFVAL      { 0 }
 ::= { mplsTeScalars 5 }

-- End of MPLS Tunnel scalars.

-- MPLS tunnel table.

mplsTunnelIndexNext OBJECT-TYPE
SYNTAX      IndexIntegerNextFree (0..65535)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "This object contains an unused value for
```

mplsTunnelIndex, or a zero to indicate that none exist. Negative values are not allowed, as they do not correspond to valid values of mplsTunnelIndex.

Note that this object offers an unused value for an mplsTunnelIndex value at the ingress side of a tunnel. At other LSRs the value of mplsTunnelIndex SHOULD be taken from the value signaled by the MPLS signaling protocol.

"

::= { mplsTeObjects 1 }

mplsTunnelTable OBJECT-TYPE

SYNTAX SEQUENCE OF MplsTunnelEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"The mplsTunnelTable allows new MPLS tunnels to be created between an LSR and a remote endpoint, and existing tunnels to be reconfigured or removed. Note that only point-to-point tunnel segments are supported, although multipoint-to-point and point-to-multipoint connections are supported by an LSR acting as a cross-connect. Each MPLS tunnel can thus have one out-segment originating at this LSR and/or one in-segment terminating at this LSR."

::= { mplsTeObjects 2 }

mplsTunnelEntry OBJECT-TYPE

SYNTAX MplsTunnelEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"An entry in this table represents an MPLS tunnel. An entry can be created by a network administrator or by an SNMP agent as instructed by an MPLS signalling protocol. Whenever a new entry is created with mplsTunnelIsIf set to true(1), then a corresponding entry is created in ifTable as well (see RFC 2863). The ifType of this entry is mplsTunnel(150).

A tunnel entry needs to be uniquely identified across a MPLS network. Indices mplsTunnelIndex and mplsTunnelInstance uniquely identify a tunnel on the LSR originating the tunnel. To uniquely identify a tunnel across an MPLS network requires

index mplsTunnelIngressLSRId. The last index mplsTunnelEgressLSRId is useful in identifying all instances of a tunnel that terminate on the same egress LSR."

REFERENCE

"1. RFC 2863 - The Interfaces Group MIB, McCloghrie, K., and F. Kastenholz, June 2000 "

```
INDEX { mplsTunnelIndex,
        mplsTunnelInstance,
        mplsTunnelIngressLSRId,
        mplsTunnelEgressLSRId
      }
 ::= { mplsTunnelTable 1 }
```

```
MplsTunnelEntry ::= SEQUENCE {
  mplsTunnelIndex           MplsTunnelIndex,
  mplsTunnelInstance       MplsTunnelInstanceIndex,
  mplsTunnelIngressLSRId   MplsExtendedTunnelId,
  mplsTunnelEgressLSRId   MplsExtendedTunnelId,
  mplsTunnelName           SnmpAdminString,
  mplsTunnelDescr         SnmpAdminString,
  mplsTunnelIsIf          TruthValue,
  mplsTunnelIfIndex       InterfaceIndexOrZero,
  mplsTunnelOwner         MplsOwner,
  mplsTunnelRole          INTEGER,
  mplsTunnelXCPointer     RowPointer,
  mplsTunnelSignallingProto INTEGER,
  mplsTunnelSetupPrio     Integer32,
  mplsTunnelHoldingPrio   Integer32,
  mplsTunnelSessionAttributes BITS,
  mplsTunnelLocalProtectInUse TruthValue,
  mplsTunnelResourcePointer RowPointer,
  mplsTunnelPrimaryInstance MplsTunnelInstanceIndex,
  mplsTunnelInstancePriority Unsigned32,
  mplsTunnelHopTableIndex MplsPathIndexOrZero,
  mplsTunnelPathInUse     MplsPathIndexOrZero,
  mplsTunnelARHopTableIndex MplsPathIndexOrZero,
  mplsTunnelCHopTableIndex MplsPathIndexOrZero,
  mplsTunnelIncludeAnyAffinity MplsTunnelAffinity,
  mplsTunnelIncludeAllAffinity MplsTunnelAffinity,
  mplsTunnelExcludeAnyAffinity MplsTunnelAffinity,
  mplsTunnelTotalUpTime   TimeTicks,
  mplsTunnelInstanceUpTime TimeTicks,
  mplsTunnelPrimaryUpTime TimeTicks,
  mplsTunnelPathChanges   Counter32,
  mplsTunnelLastPathChange TimeTicks,
  mplsTunnelCreationTime  TimeStamp,
  mplsTunnelStateTransitions Counter32,
```



```

mplsTunnelAdminStatus      INTEGER,
mplsTunnelOperStatus      INTEGER,
mplsTunnelRowStatus       RowStatus,
mplsTunnelStorageType     StorageType
}

mplsTunnelIndex OBJECT-TYPE
SYNTAX      MplsTunnelIndex
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "Uniquely identifies a set of tunnel instances
    between a pair of ingress and egress LSRs.
    Managers should obtain new values for row
    creation in this table by reading
    mplsTunnelIndexNext. When
    the MPLS signalling protocol is rsvp(2) this value
    SHOULD be equal to the value signaled in the
    Tunnel Id of the Session object. When the MPLS
    signalling protocol is crldp(3) this value
    SHOULD be equal to the value signaled in the
    LSP ID."
 ::= { mplsTunnelEntry 1 }

mplsTunnelInstance OBJECT-TYPE
SYNTAX      MplsTunnelInstanceIndex
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "Uniquely identifies a particular instance of a
    tunnel between a pair of ingress and egress LSRs.
    It is useful to identify multiple instances of
    tunnels for the purposes of backup and parallel
    tunnels. When the MPLS signaling protocol is
    rsvp(2) this value SHOULD be equal to the LSP Id
    of the Sender Template object. When the signaling
    protocol is crldp(3) there is no equivalent
    signaling object."
 ::= { mplsTunnelEntry 2 }

mplsTunnelIngressLSRId OBJECT-TYPE
SYNTAX      MplsExtendedTunnelId
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "Identity of the ingress LSR associated with this
    tunnel instance. When the MPLS signalling protocol
    is rsvp(2) this value SHOULD be equal to the Tunnel

```

Sender Address in the Sender Template object and MAY be equal to the Extended Tunnel Id field in the SESSION object. When the MPLS signalling protocol is crldp(3) this value SHOULD be equal to the Ingress LSR Router ID field in the LSPID TLV object."

REFERENCE

1. RSVP-TE: Extensions to RSVP for LSP Tunnels, Awduche et al, RFC 3209, December 2001
2. Constraint-Based LSP Setup using LDP, Jamoussi (Editor), RFC 3212, January 2002"

::= { mplsTunnelEntry 3 }

mplsTunnelEgressLSRId OBJECT-TYPE

SYNTAX MplsExtendedTunnelId

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Identity of the egress LSR associated with this tunnel instance."

::= { mplsTunnelEntry 4 }

mplsTunnelName OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"The canonical name assigned to the tunnel. This name can be used to refer to the tunnel on the LSR's console port. If mplsTunnelIsIf is set to true then the ifName of the interface corresponding to this tunnel should have a value equal to mplsTunnelName. Also see the description of ifName in RFC 2863."

REFERENCE

- "RFC 2863 - The Interfaces Group MIB, McCloghrie, K., and F. Kastenholz, June 2000"

DEFVAL { "" }

::= { mplsTunnelEntry 5 }

mplsTunnelDescr OBJECT-TYPE

SYNTAX SnmpAdminString

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A textual string containing information about the tunnel. If there is no description this object contains a zero length string. This object is may not be signaled by MPLS signaling protocols,

consequently the value of this object at transit and egress LSRs MAY be automatically generated or absent."

DEFVAL { "" }
 ::= { mplsTunnelEntry 6 }

mplsTunnelIsIf OBJECT-TYPE
SYNTAX TruthValue
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Denotes whether or not this tunnel corresponds to an interface represented in the interfaces group table. Note that if this variable is set to true then the ifName of the interface corresponding to this tunnel should have a value equal to mplsTunnelName. Also see the description of ifName in RFC 2863. This object is meaningful only at the ingress and egress LSRs."
REFERENCE
"RFC 2863 - The Interfaces Group MIB, McCloghrie, K., and F. Kastenholz, June 2000"
DEFVAL { false }
 ::= { mplsTunnelEntry 7 }

mplsTunnelIfIndex OBJECT-TYPE
SYNTAX InterfaceIndexOrZero
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"If mplsTunnelIsIf is set to true, then this value contains the LSR-assigned ifIndex which corresponds to an entry in the interfaces table. Otherwise this variable should contain the value of zero indicating that a valid ifIndex was not assigned to this tunnel interface."
REFERENCE
"RFC 2863 - The Interfaces Group MIB, McCloghrie, K., and F. Kastenholz, June 2000"
DEFVAL { 0 }
 ::= { mplsTunnelEntry 8 }

mplsTunnelOwner OBJECT-TYPE
SYNTAX MplsOwner
MAX-ACCESS read-only
STATUS current
DESCRIPTION
"Denotes the entity that created and is responsible

for managing this tunnel. This column is automatically filled by the agent on creation of a row."

```
 ::= { mplsTunnelEntry 9 }
```

mplsTunnelRole OBJECT-TYPE

```
SYNTAX          INTEGER { head(1),
                           transit(2),
                           tail(3),
                           headTail(4) }
```

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This value signifies the role that this tunnel entry/instance represents. This value MUST be set to head(1) at the originating point of the tunnel. This value MUST be set to transit(2) at transit points along the tunnel, if transit points are supported. This value MUST be set to tail(3) at the terminating point of the tunnel if tunnel tails are supported.

The value headTail(4) is provided for tunnels that begin and end on the same LSR."

```
DEFVAL { head }
```

```
 ::= { mplsTunnelEntry 10 }
```

mplsTunnelXCPointer OBJECT-TYPE

```
SYNTAX          RowPointer
```

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This variable points to a row in the mplsXCTable. This table identifies the segments that compose this tunnel, their characteristics, and relationships to each other. A value of zeroDotZero indicates that no LSP has been associated with this tunnel yet."

REFERENCE

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

```
DEFVAL          { zeroDotZero }
```

```
 ::= { mplsTunnelEntry 11 }
```

mplsTunnelSignallingProto OBJECT-TYPE

```
SYNTAX          INTEGER {
```

```

        none(1),
        rsvp(2),
        crldp(3),
        other(4)
    }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION     "The signalling protocol, if any, used to setup this
                tunnel."
DEFVAL         { none }
 ::= { mplsTunnelEntry 12 }

mplsTunnelSetupPrio OBJECT-TYPE
SYNTAX          Integer32 (0..7)
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION     "Indicates the setup priority of this tunnel."
REFERENCE      "1. RSVP-TE: Extensions to RSVP for LSP Tunnels,
                Awduche et al, RFC 3209, December 2001
                2. Constraint-Based LSP Setup using LDP, Jamoussi
                (Editor), RFC 3212, January 2002"
DEFVAL         { 0 }
 ::= { mplsTunnelEntry 13 }

mplsTunnelHoldingPrio OBJECT-TYPE
SYNTAX          Integer32 (0..7)
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION     "Indicates the holding priority for this tunnel."
REFERENCE      "1. RSVP-TE: Extensions to RSVP for LSP Tunnels,
                Awduche et al, RFC 3209, December 2001

                2. Constraint-Based LSP Setup using LDP, Jamoussi
                (Editor), RFC 3212, January 2002"
DEFVAL         { 0 }
 ::= { mplsTunnelEntry 14 }

mplsTunnelSessionAttributes OBJECT-TYPE
SYNTAX          BITS {
                fastReroute (0),
                mergingPermitted (1),
                isPersistent (2),
                isPinned (3),

```

```

        recordRoute(4)
    }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "This bit mask indicates optional session values for
    this tunnel. The following describes these bit
    fields:

    fastRerouteThis flag indicates that the any tunnel
    hop may choose to reroute this tunnel without
    tearing it down. This flag permits transit routers
    to use a local repair mechanism which may result in
    violation of the explicit routing of this tunnel.
    When a fault is detected on an adjacent downstream
    link or node, a transit router can re-route traffic
    for fast service restoration.

    mergingPermitted This flag permits transit routers
    to merge this session with other RSVP sessions for
    the purpose of reducing resource overhead on
    downstream transit routers, thereby providing
    better network scaling.

    isPersistent Indicates whether this tunnel should
    be restored automatically after a failure occurs.

    isPinned This flag indicates whether the loose-
    routed hops of this tunnel are to be pinned.

    recordRouteThis flag indicates whether or not the
    signalling protocol should remember the tunnel path
    after it has been signaled."
REFERENCE
    "1. RSVP-TE: Extensions to RSVP for LSP Tunnels,
    Awduche et al, RFC 3209, December 2001."
 ::= { mplsTunnelEntry 15 }

mplsTunnelLocalProtectInUse OBJECT-TYPE
SYNTAX          TruthValue
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "Indicates that the local repair mechanism is in use
    to maintain this tunnel (usually in the face of an
    outage of the link it was previously routed over)."
```

```

DEFVAL { false }
 ::= { mplsTunnelEntry 16 }
```

```

mplsTunnelResourcePointer OBJECT-TYPE
    SYNTAX          RowPointer
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "This variable represents a pointer to the traffic
        parameter specification for this tunnel.  This
        value may point at an entry in the
        mplsTunnelResourceEntry to indicate which
        mplsTunnelResourceEntry is to be assigned to this
        LSP instance.  This value may optionally point at
        an externally defined traffic parameter
        specification table.  A value of zeroDotZero
        indicates best-effort treatment.  By having the
        same value of this object, two or more LSPs can
        indicate resource sharing."
    DEFVAL          { zeroDotZero }
    ::= { mplsTunnelEntry 17 }

mplsTunnelPrimaryInstance OBJECT-TYPE
    SYNTAX          MplsTunnelInstanceIndex
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Specifies the instance index of the primary instance
        of this tunnel.  More details of the definition of
        tunnel instances and the primary tunnel instance
        can be found in the description of the TEXTUAL-CONVENTION
        MplsTunnelInstanceIndex."
    DEFVAL          { 0 }
    ::= { mplsTunnelEntry 18 }

mplsTunnelInstancePriority OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "This value indicates which priority, in descending
        order, with 0 indicating the lowest priority,
        within a group of tunnel instances.  A group of
        tunnel instances is defined as a set of LSPs with
        the same mplsTunnelIndex in this table, but with a
        different mplsTunnelInstance.  Tunnel instance
        priorities are used to denote the priority at which
        a particular tunnel instance will supercede
        another.  Instances of tunnels containing the same
        mplsTunnelInstancePriority will be used for load
        sharing."

```

```
DEFVAL          { 0 }
 ::= { mplsTunnelEntry 19 }

mplsTunnelHopTableIndex OBJECT-TYPE
SYNTAX          MplsPathIndexOrZero
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION
    "Index into the mplsTunnelHopTable entry that
     specifies the explicit route hops for this tunnel.
     This object is meaningful only at the head-end of
     the tunnel."
DEFVAL { 0 }
 ::= { mplsTunnelEntry 20 }

mplsTunnelPathInUse OBJECT-TYPE
SYNTAX          MplsPathIndexOrZero
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION
    "This value denotes the configured path that was
     chosen for this tunnel. This value reflects the
     secondary index into mplsTunnelHopTable. This path
     may not exactly match the one in
     mplsTunnelARHopTable due to the fact that some CSPF
     modification may have taken place. See
     mplsTunnelARHopTable for the actual path being
     taken by the tunnel. A value of zero denotes that
     no path is currently in use or available."
DEFVAL { 0 }
 ::= { mplsTunnelEntry 21 }

mplsTunnelARHopTableIndex OBJECT-TYPE
SYNTAX          MplsPathIndexOrZero
MAX-ACCESS     read-only
STATUS         current
DESCRIPTION
    "Index into the mplsTunnelARHopTable entry that
     specifies the actual hops traversed by the tunnel.
     This is automatically updated by the agent when the
     actual hops becomes available."
DEFVAL { 0 }
 ::= { mplsTunnelEntry 22 }

mplsTunnelCHopTableIndex OBJECT-TYPE
SYNTAX          MplsPathIndexOrZero
MAX-ACCESS     read-only
STATUS         current
```


DESCRIPTION

"Index into the mplsTunnelCHopTable entry that specifies the computed hops traversed by the tunnel. This is automatically updated by the agent when computed hops become available or when computed hops get modified."

DEFVAL { 0 }

::= { mplsTunnelEntry 23 }

mplsTunnelIncludeAnyAffinity OBJECT-TYPE

SYNTAX MplsTunnelAffinity

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A link satisfies the include-any constraint if and only if the constraint is zero, or the link and the constraint have a resource class in common."

REFERENCE

"1. RSVP-TE: Extensions to RSVP for LSP Tunnels, Awduche et al, RFC 3209, December 2001."

::= { mplsTunnelEntry 24 }

mplsTunnelIncludeAllAffinity OBJECT-TYPE

SYNTAX MplsTunnelAffinity

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A link satisfies the include-all constraint if and only if the link contains all of the administrative groups specified in the constraint."

REFERENCE

"1. RSVP-TE: Extensions to RSVP for LSP Tunnels, Awduche et al, RFC 3209, December 2001."

::= { mplsTunnelEntry 25 }

mplsTunnelExcludeAnyAffinity OBJECT-TYPE

SYNTAX MplsTunnelAffinity

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"A link satisfies the exclude-any constraint if and only if the link contains none of the administrative groups specified in the constraint."

REFERENCE

"1. RSVP-TE: Extensions to RSVP for LSP Tunnels, Awduche et al, RFC 3209, December 2001."

DEFVAL { 0 }

::= { mplsTunnelEntry 26 }

```
mplsTunnelTotalUpTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This value represents the aggregate up time for all
         instances of this tunnel, if available. If this
         value is unavailable, it MUST return a value of 0."
    ::= { mplsTunnelEntry 27 }

mplsTunnelInstanceUpTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "This value identifies the total time that this
         tunnel instance's operStatus has been Up(1)."
    ::= { mplsTunnelEntry 28 }

mplsTunnelPrimaryUpTime OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the total time the primary instance of
         this tunnel has been active. The primary instance
         of this tunnel is defined in
         mplsTunnelPrimaryInstance."
    ::= { mplsTunnelEntry 29 }

mplsTunnelPathChanges OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the number of times the actual path for
         this tunnel instance has changed."
    ::= { mplsTunnelEntry 30 }

mplsTunnelLastPathChange OBJECT-TYPE
    SYNTAX      TimeTicks
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the time since the last change to the
         actual path for this tunnel instance."
    ::= { mplsTunnelEntry 31 }
```

```
mplsTunnelCreationTime OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the value of SysUpTime when the first
         instance of this tunnel came into existence.
         That is, when the value of mplsTunnelOperStatus
         was first set to up(1)."
```

```
 ::= { mplsTunnelEntry 32 }
```

```
mplsTunnelStateTransitions OBJECT-TYPE
    SYNTAX      Counter32
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Specifies the number of times the state
         (mplsTunnelOperStatus) of this tunnel instance has
         changed."
```

```
 ::= { mplsTunnelEntry 33 }
```

```
mplsTunnelAdminStatus OBJECT-TYPE
    SYNTAX      INTEGER {
        -- ready to pass packets
        up(1),
        down(2),
        -- in some test mode
        testing(3)
    }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Indicates the desired operational status of this
         tunnel."
```

```
 ::= { mplsTunnelEntry 34 }
```

```
mplsTunnelOperStatus OBJECT-TYPE
    SYNTAX      INTEGER {
        -- ready to pass packets
        up(1),
        down(2),
        -- in some test mode
        testing(3),
        -- status cannot be determined
        unknown(4),
        dormant(5),
        -- some component is missing
        notPresent(6),
```

```

        -- down due to the state of
        -- lower layer interfaces
        lowerLayerDown(7)
    }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "Indicates the actual operational status of this
    tunnel, which is typically but not limited to, a
    function of the state of individual segments of
    this tunnel."
 ::= { mplsTunnelEntry 35 }

mplsTunnelRowStatus 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 by the agent except
    mplsTunnelAdminStatus, mplsTunnelRowStatus and
    mplsTunnelStorageType."
 ::= { mplsTunnelEntry 36 }

mplsTunnelStorageType OBJECT-TYPE
SYNTAX          StorageType
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION     "The storage type for this tunnel entry.
    Conceptual rows having the value 'permanent'
    need not allow write-access to any columnar
    objects in the row."
DEFVAL { volatile }
 ::= { mplsTunnelEntry 37 }

-- End of mplsTunnelTable

mplsTunnelHopListIndexNext OBJECT-TYPE
SYNTAX          MplsPathIndexOrZero
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION
    "This object contains an appropriate value to be used
    for mplsTunnelHopListIndex when creating entries in
    the mplsTunnelHopTable.  If the number of
    unassigned entries is exhausted, a retrieval

```

operation will return a value of 0. This object may also return a value of 0 when the LSR is unable to accept conceptual row creation, for example, if the `mplsTunnelHopTable` is implemented as read-only. To obtain the value of `mplsTunnelHopListIndex` for a new entry in the `mplsTunnelHopTable`, the manager issues a management protocol retrieval operation to obtain the current value of `mplsTunnelHopIndex`.

When the SET is performed to create a row in the `mplsTunnelHopTable`, the Command Responder (agent) must determine whether the value is indeed still unused; Two Network Management Applications may attempt to create a row (configuration entry) simultaneously and use the same value. If it is currently unused, the SET succeeds and the Command Responder (agent) changes the value of this object, according to an implementation-specific algorithm. If the value is in use, however, the SET fails. The Network Management Application must then re-read this variable to obtain a new usable value."

```
::= { mplsTeObjects 3 }
```

```
mplsTunnelHopTable OBJECT-TYPE
```

```
SYNTAX SEQUENCE OF MplsTunnelHopEntry
```

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

```
STATUS current
```

```
DESCRIPTION
```

```
"The mplsTunnelHopTable is used to indicate the hops,
strict or loose, for an instance of an MPLS tunnel
defined in mplsTunnelTable, when it is established
via signalling, for the outgoing direction of the
tunnel. Thus at a transit LSR, this table contains
the desired path of the tunnel from this LSR
onwards. Each row in this table is indexed by
mplsTunnelHopListIndex which corresponds to a group
of hop lists or path options. Each row also has a
secondary index mplsTunnelHopIndex, which indicates
a group of hops (also known as a path option).
Finally, the third index, mplsTunnelHopIndex
indicates the specific hop information for a path
option. In case we want to specify a particular
interface on the originating LSR of an outgoing
tunnel by which we want packets to exit the LSR,
we specify this as the first hop for this tunnel in
mplsTunnelHopTable."
```

```
::= { mplsTeObjects 4 }
```

```

mplsTunnelHopEntry OBJECT-TYPE
    SYNTAX      MplsTunnelHopEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in this table represents a tunnel hop.  An
         entry is created by a network administrator for
         signaled ERLSP set up by an MPLS signalling
         protocol."
    INDEX {
        mplsTunnelHopListIndex,
        mplsTunnelHopPathOptionIndex,
        mplsTunnelHopIndex
    }
    ::= { mplsTunnelHopTable 1 }

MplsTunnelHopEntry ::= SEQUENCE {
    mplsTunnelHopListIndex      MplsPathIndex,
    mplsTunnelHopPathOptionIndex MplsPathIndex,
    mplsTunnelHopIndex          MplsPathIndex,
    mplsTunnelHopAddrType       TeHopAddressType,
    mplsTunnelHopIpAddr         TeHopAddress,
    mplsTunnelHopIpPrefixLen    InetAddressPrefixLength,
    mplsTunnelHopAsNumber       TeHopAddressAS,
    mplsTunnelHopAddrUnnum      TeHopAddressUnnum,
    mplsTunnelHopLspId          MplsLSPID,
    mplsTunnelHopType           INTEGER,
    mplsTunnelHopInclude        TruthValue,
    mplsTunnelHopPathOptionName SnmpAdminString,
    mplsTunnelHopEntryPathComp  INTEGER,
    mplsTunnelHopRowStatus      RowStatus,
    mplsTunnelHopStorageType    StorageType
}

mplsTunnelHopListIndex OBJECT-TYPE
    SYNTAX      MplsPathIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Primary index into this table identifying a
         particular explicit route object."
    ::= { mplsTunnelHopEntry 1 }

mplsTunnelHopPathOptionIndex OBJECT-TYPE
    SYNTAX      MplsPathIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION

```

```

        "Secondary index into this table identifying a
        particular group of hops representing a particular
        configured path. This is otherwise known as a path
        option."
 ::= { mplsTunnelHopEntry 2 }

mplsTunnelHopIndex OBJECT-TYPE
SYNTAX      MplsPathIndex
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "Tertiary index into this table identifying a
    particular hop."
 ::= { mplsTunnelHopEntry 3 }

mplsTunnelHopAddrType OBJECT-TYPE
SYNTAX      TeHopAddressType
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "The Hop Address Type of this tunnel hop.

            The value of this object cannot be changed
            if the value of the corresponding
            mplsTunnelHopRowStatus object is 'active'.

            Note that lspid(5) is a valid option only
            for tunnels signaled via CRLDP.
            "
DEFVAL      { ipv4 }
 ::= { mplsTunnelHopEntry 4 }

mplsTunnelHopIpAddr OBJECT-TYPE
SYNTAX      TeHopAddress
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "The Tunnel Hop Address for this tunnel hop.

            The type of this address is determined by the
            value of the corresponding mplsTunnelHopAddrType.

            The value of this object cannot be changed
            if the value of the corresponding
            mplsTunnelHopRowStatus object is 'active'.
            "
DEFVAL      { '00000000'h } -- IPv4 address 0.0.0.0
 ::= { mplsTunnelHopEntry 5 }

mplsTunnelHopIpPrefixLen OBJECT-TYPE

```

```

SYNTAX      InetAddressPrefixLength
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "If mplsTunnelHopAddrType is set to ipv4(1) or
            ipv6(2), then this value will contain an
            appropriate prefix length for the IP address in
            object mplsTunnelHopIpAddr. Otherwise this value
            is irrelevant and should be ignored.
            "
DEFVAL      { 32 }
 ::= { mplsTunnelHopEntry 6 }

mplsTunnelHopAsNumber OBJECT-TYPE
SYNTAX      TeHopAddressAS
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "If mplsTunnelHopAddrType is set to asnumber(3), then
    this value will contain the AS number of this hop.
    Otherwise the agent should set this object to zero-
    length string and the manager should ignore this."
 ::= { mplsTunnelHopEntry 7 }

mplsTunnelHopAddrUnnum OBJECT-TYPE
SYNTAX      TeHopAddressUnnum
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "If mplsTunnelHopAddrType is set to unnum(4), then
    this value will contain the interface identifier of
    the unnumbered interface for this hop. This object
    should be used in conjunction with
    mplsTunnelHopIpAddress which would contain the LSR
    Router ID in this case. Otherwise the agent should
    set this object to zero-length string and the
    manager should ignore this."
 ::= { mplsTunnelHopEntry 8 }

mplsTunnelHopLspId OBJECT-TYPE
SYNTAX      MplsLSPID
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "If mplsTunnelHopAddrType is set to lspid(5), then
    this value will contain the LSPID of a tunnel of
    this hop. The present tunnel being configured is
    tunneled through this hop (using label stacking).
    This object is otherwise insignificant and should

```



```

        contain a value of 0 to indicate this fact."
 ::= { mplsTunnelHopEntry 9 }

mplsTunnelHopType OBJECT-TYPE
SYNTAX      INTEGER {
                strict(1),
                loose(2)
            }
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "Denotes whether this tunnel hop is routed in a
     strict or loose fashion. The value of this object
     has no meaning if the mplsTunnelHopInclude object
     is set to 'false'."
 ::= { mplsTunnelHopEntry 10 }

mplsTunnelHopInclude OBJECT-TYPE
SYNTAX      TruthValue
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "If this value is set to true, then this indicates
     that this hop must be included in the tunnel's
     path. If this value is set to 'false', then this hop
     must be avoided when calculating the path for this
     tunnel. The default value of this object is 'true',
     so that by default all indicated hops are included
     in the CSPF path computation. If this object is set
     to 'false' the value of mplsTunnelHopType should be
     ignored."
DEFVAL { true }
 ::= { mplsTunnelHopEntry 11 }

mplsTunnelHopPathOptionName OBJECT-TYPE
SYNTAX      SnmpAdminString
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The description of this series of hops as they
     relate to the specified path option. The
     value of this object SHOULD be the same for
     each hop in the series that comprises a
     path option."
 ::= { mplsTunnelHopEntry 12 }

mplsTunnelHopEntryPathComp OBJECT-TYPE
SYNTAX      INTEGER {

```

```

        dynamic(1),    -- CSPF computed
        explicit(2)   -- strict hop
    }
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "If this value is set to dynamic, then the user
     should only specify the source and destination of
     the path and expect that the CSPF will calculate
     the remainder of the path.  If this value is set to
     explicit, the user should specify the entire path
     for the tunnel to take.  This path may contain
     strict or loose hops.  Each hop along a specific
     path SHOULD have this object set to the same value"
 ::= { mplsTunnelHopEntry 13 }

mplsTunnelHopRowStatus 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 by the agent except
     mplsTunnelHopRowStatus and
     mplsTunnelHopStorageType."
 ::= { mplsTunnelHopEntry 14 }

mplsTunnelHopStorageType OBJECT-TYPE
SYNTAX          StorageType
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION
    "The storage type for this Hop entry.  Conceptual
     rows having the value 'permanent' need not
     allow write-access to any columnar objects
     in the row."
DEFVAL { volatile }
 ::= { mplsTunnelHopEntry 15 }

-- End of mplsTunnelHopTable

-- Begin of mplsTunnelResourceTable

mplsTunnelResourceIndexNext OBJECT-TYPE
SYNTAX          Unsigned32 (0.. 2147483647)
MAX-ACCESS      read-only

```

STATUS current
DESCRIPTION

"This object contains the next appropriate value to be used for mplsTunnelResourceIndex when creating entries in the mplsTunnelResourceTable. If the number of unassigned entries is exhausted, a retrieval operation will return a value of 0. This object may also return a value of 0 when the LSR is unable to accept conceptual row creation, for example, if the mplsTunnelTable is implemented as read-only. To obtain the mplsTunnelResourceIndex value for a new entry, the manager must first issue a management protocol retrieval operation to obtain the current value of this object.

When the SET is performed to create a row in the mplsTunnelResourceTable, the Command Responder (agent) must determine whether the value is indeed still unused; Two Network Management Applications may attempt to create a row (configuration entry) simultaneously and use the same value. If it is currently unused, the SET succeeds and the Command Responder (agent) changes the value of this object, according to an implementation-specific algorithm. If the value is in use, however, the SET fails. The Network Management Application must then re-read this variable to obtain a new usable value."

::= { mplsTeObjects 5 }

mplsTunnelResourceTable OBJECT-TYPE

SYNTAX SEQUENCE OF MplsTunnelResourceEntry
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"The mplsTunnelResourceTable allows a manager to specify which resources are desired for an MPLS tunnel. This table also allows several tunnels to point to a single entry in this table, implying that these tunnels should share resources."

::= { mplsTeObjects 6 }

mplsTunnelResourceEntry OBJECT-TYPE

SYNTAX MplsTunnelResourceEntry
MAX-ACCESS not-accessible
STATUS current

DESCRIPTION

"An entry in this table represents a set of resources for an MPLS tunnel. An entry can be created by a

network administrator or by an SNMP agent as instructed by any MPLS signalling protocol. An entry in this table referenced by a tunnel instance with zero `mplsTunnelInstance` value indicates a configured set of resource parameter. An entry referenced by a tunnel instance with a non-zero `mplsTunnelInstance` reflects the in-use resource parameters for the tunnel instance which may have been negotiated or modified by the MPLS signaling protocols."

```
INDEX          { mplsTunnelResourceIndex }
 ::= { mplsTunnelResourceTable 1 }
```

```
MplsTunnelResourceEntry ::= SEQUENCE {
    mplsTunnelResourceIndex          Unsigned32,
    mplsTunnelResourceMaxRate        MplsBitRate,
    mplsTunnelResourceMeanRate       MplsBitRate,
    mplsTunnelResourceMaxBurstSize   MplsBurstSize,
    mplsTunnelResourceMeanBurstSize  MplsBurstSize,
    mplsTunnelResourceExBurstSize    MplsBurstSize,
    mplsTunnelResourceFrequency      INTEGER,
    mplsTunnelResourceWeight         Unsigned32,
    mplsTunnelResourceRowStatus      RowStatus,
    mplsTunnelResourceStorageType    StorageType
}
```

```
mplsTunnelResourceIndex OBJECT-TYPE
SYNTAX          Unsigned32 (1..2147483647)
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION     "Uniquely identifies this row."
 ::= { mplsTunnelResourceEntry 1 }
```

```
mplsTunnelResourceMaxRate OBJECT-TYPE
SYNTAX          MplsBitRate
UNITS           "kilobits per second"
MAX-ACCESS      read-create
STATUS          current
DESCRIPTION     "The maximum rate in bits/second. Note that setting
                mplsTunnelResourceMaxRate,
                mplsTunnelResourceMeanRate, and
                mplsTunnelResourceMaxBurstSize to 0 indicates best-
                effort treatment."
 ::= { mplsTunnelResourceEntry 2 }
```

```
mplsTunnelResourceMeanRate OBJECT-TYPE
```

```

SYNTAX      MplsBitRate
UNITS       "kilobits per second"
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "This object is copied into an instance of
     mplsTrafficParamMeanRate in the
     mplsTrafficParamTable. The OID of this table entry
     is then copied into the corresponding
     mplsInSegmentTrafficParamPtr."
 ::= { mplsTunnelResourceEntry 3 }

mplsTunnelResourceMaxBurstSize OBJECT-TYPE
SYNTAX      MplsBurstSize
UNITS       "bytes"
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The maximum burst size in bytes."
 ::= { mplsTunnelResourceEntry 4 }

mplsTunnelResourceMeanBurstSize OBJECT-TYPE
SYNTAX      MplsBurstSize
UNITS       "bytes"
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The mean burst size in bytes. The implementations
     which do not implement this variable must return
     a noSuchObject exception for this object and must
     not allow a user to set this object."
 ::= { mplsTunnelResourceEntry 5 }

mplsTunnelResourceExBurstSize OBJECT-TYPE
SYNTAX      MplsBurstSize
UNITS       "bytes"
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The Excess burst size in bytes. The implementations
     which do not implement this variable must return
     noSuchObject exception for this object and must
     not allow a user to set this value."
REFERENCE
    "CR-LDP Specification, Section 4.3."
 ::= { mplsTunnelResourceEntry 6 }

mplsTunnelResourceFrequency OBJECT-TYPE

```

```

SYNTAX      INTEGER { unspecified(1),
                    frequent(2),
                    veryFrequent(3)
                    }
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The granularity of the availability of committed
    rate. The implementations which do not implement
    this variable must return unspecified(1) for this
    value and must not allow a user to set this value."
REFERENCE
    "CR-LDP Specification, Section 4.3."
 ::= { mplsTunnelResourceEntry 7 }

```

```

mplsTunnelResourceWeight OBJECT-TYPE
SYNTAX      Unsigned32(0..255)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The relative weight for using excess bandwidth above
    its committed rate. The value of 0 means that
    weight is not applicable for the CR-LSP."
REFERENCE
    "CR-LDP Specification, Section 4.3."
 ::= { mplsTunnelResourceEntry 8 }

```

```

mplsTunnelResourceRowStatus 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 by the agent except
    mplsTunnelResourceRowStatus and
    mplsTunnelResourceStorageType."
 ::= { mplsTunnelResourceEntry 9 }

```

```

mplsTunnelResourceStorageType OBJECT-TYPE
SYNTAX      StorageType
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "The storage type for this Hop entry. Conceptual
    rows having the value 'permanent' need not
    allow write-access to any columnar objects

```

```

        in the row."
    DEFVAL { volatile }

 ::= { mplsTunnelResourceEntry 10 }

-- End mplsTunnelResourceTable
-- Tunnel Actual Route Hop table.

mplsTunnelARHopTable OBJECT-TYPE
    SYNTAX          SEQUENCE OF MplsTunnelARHopEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION
        "The mplsTunnelARHopTable is used to indicate the
        hops for an MPLS tunnel defined in mplsTunnelTable,
        as reported by the MPLS signalling protocol. Thus at
        a transit LSR, this table (if the table is supported
        and if the signaling protocol is recording actual
        route information) contains the actual route of the
        whole tunnel. If the signaling protocol is not
        recording the actual route, this table MAY report
        the information from the mplsTunnelHopTable or the
        mplsTunnelCHopTable.

        Each row in this table is indexed by
        mplsTunnelARHopListIndex. Each row also has a
        secondary index mplsTunnelARHopIndex, corresponding
        to the next hop that this row corresponds to.

        Please note that since the information necessary to
        build entries within this table is not provided by
        some MPLS signalling protocols, implementation of
        this table is optional. Furthermore, since the
        information in this table is actually provided by
        the MPLS signalling protocol after the path has
        been set-up, the entries in this table are provided
        only for observation, and hence, all variables in
        this table are accessible exclusively as read-
        only.

        Note also that the contents of this table may change
        while it is being read because of re-routing
        activities. A network administrator may verify that
        the actual route read is consistent by reference to
        the mplsTunnelLastPathChange object."
 ::= { mplsTeObjects 7 }

```

```

mplsTunnelARHopEntry OBJECT-TYPE
    SYNTAX      MplsTunnelARHopEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in this table represents a tunnel hop. An
         entry is created by the agent for signaled ERLSP
         set up by an MPLS signalling protocol."
    INDEX { mplsTunnelARHopListIndex, mplsTunnelARHopIndex }
    ::= { mplsTunnelARHopTable 1 }

MplsTunnelARHopEntry ::= SEQUENCE {
    mplsTunnelARHopListIndex      MplsPathIndex,
    mplsTunnelARHopIndex          MplsPathIndex,
    mplsTunnelARHopAddrType       TeHopAddressType,
    mplsTunnelARHopIpAddr         TeHopAddress,
    mplsTunnelARHopAddrUnnum      TeHopAddressUnnum,
    mplsTunnelARHopLspId          MplsLSPID
}

mplsTunnelARHopListIndex OBJECT-TYPE
    SYNTAX      MplsPathIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Primary index into this table identifying a
         particular recorded hop list."
    ::= { mplsTunnelARHopEntry 1 }

mplsTunnelARHopIndex OBJECT-TYPE
    SYNTAX      MplsPathIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Secondary index into this table identifying the
         particular hop."
    ::= { mplsTunnelARHopEntry 2 }

mplsTunnelARHopAddrType OBJECT-TYPE
    SYNTAX      TeHopAddressType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The Hop Address Type of this tunnel hop.

         Note that lspid(5) is a valid option only
         for tunnels signaled via CRLDP."
    DEFVAL     { ipv4 }

```



```
 ::= { mplsTunnelARHopEntry 3 }

mplsTunnelARHopIpAddress OBJECT-TYPE
    SYNTAX      TeHopAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The Tunnel Hop Address for this tunnel hop.

        The type of this address is determined by the
        value of the corresponding mplsTunnelARHopAddrType.
        If mplsTunnelARHopAddrType is set to unnum(4),
        then this value contains the LSR Router ID of the
        unnumbered interface. Otherwise the agent SHOULD
        set this object to the zero-length string and the
        manager should ignore this object."
    DEFVAL      { '00000000'h } -- IPv4 address 0.0.0.0
 ::= { mplsTunnelARHopEntry 4 }

mplsTunnelARHopAddrUnnum OBJECT-TYPE
    SYNTAX      TeHopAddressUnnum
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "If mplsTunnelARHopAddrType is set to unnum(4), then
        this value will contain the interface identifier of
        the unnumbered interface for this hop. This object
        should be used in conjunction with
        mplsTunnelARHopIpAddress which would contain the LSR
        Router ID in this case. Otherwise the agent should
        set this object to zero-length string and the
        manager should ignore this."
 ::= { mplsTunnelARHopEntry 5 }

mplsTunnelARHopLspId OBJECT-TYPE
    SYNTAX      MplsLSPID
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "If mplsTunnelARHopAddrType is set to lspid(5), then
        this value will contain the LSP ID of this hop.
        This object is otherwise insignificant and should
        contain a value of 0 to indicate this fact."
 ::= { mplsTunnelARHopEntry 6 }

-- End of mplsTunnelARHopTable
```

-- Tunnel Computed Hop table.

```
mplsTunnelCHopTable OBJECT-TYPE
SYNTAX          SEQUENCE OF MplsTunnelCHopEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "The mplsTunnelCHopTable is used to indicate the
    hops, strict or loose, for an MPLS tunnel defined
    in mplsTunnelTable, as computed by a constraint-
    based routing protocol, based on the
    mplsTunnelHopTable for the outgoing direction of
    the tunnel. Thus at a transit LSR, this table (if
    the table is supported) MAY contain the path
    computed by the CSPF engine on (or on behalf of)
    this LSR. Each row in this table is indexed by
    mplsTunnelCHopListIndex. Each row also has a
    secondary index mplsTunnelCHopIndex, corresponding
    to the next hop that this row corresponds to. In
    case we want to specify a particular interface on
    the originating LSR of an outgoing tunnel by which
    we want packets to exit the LSR, we specify this as
    the first hop for this tunnel in
    mplsTunnelCHopTable.
```

Please note that since the information necessary to build entries within this table may not be supported by some LSRs, implementation of this table is optional. Furthermore, since the information in this table describes the path computed by the CSPF engine the entries in this table are read-only."

```
::= { mplsTeObjects 8 }
```

```
mplsTunnelCHopEntry OBJECT-TYPE
SYNTAX          MplsTunnelCHopEntry
MAX-ACCESS      not-accessible
STATUS          current
DESCRIPTION
    "An entry in this table represents a tunnel hop. An
    entry in this table is created by a path
    computation engine using CSPF techniques applied to
    the information collected by routing protocols and
    the hops specified in the corresponding
    mplsTunnelHopTable."
INDEX { mplsTunnelCHopListIndex, mplsTunnelCHopIndex }
 ::= { mplsTunnelCHopTable 1 }
```

```

MplsTunnelCHopEntry ::= SEQUENCE {
    mplsTunnelCHopListIndex      MplsPathIndex,
    mplsTunnelCHopIndex          MplsPathIndex,
    mplsTunnelCHopAddrType       TeHopAddressType,
    mplsTunnelCHopIpAddr         TeHopAddress,
    mplsTunnelCHopIpPrefixLen    InetAddressPrefixLength,
    mplsTunnelCHopAsNumber        TeHopAddressAS,
    mplsTunnelCHopAddrUnnum       TeHopAddressUnnum,
    mplsTunnelCHopLspId           MplsLSPID,
    mplsTunnelCHopType            INTEGER
}

```

```

mplsTunnelCHopListIndex OBJECT-TYPE
    SYNTAX      MplsPathIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Primary index into this table identifying a
         particular computed hop list."
    ::= { mplsTunnelCHopEntry 1 }

```

```

mplsTunnelCHopIndex OBJECT-TYPE
    SYNTAX      MplsPathIndex
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Secondary index into this table identifying the
         particular hop."
    ::= { mplsTunnelCHopEntry 2 }

```

```

mplsTunnelCHopAddrType OBJECT-TYPE
    SYNTAX      TeHopAddressType
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The Hop Address Type of this tunnel hop.

        Note that lspid(5) is a valid option only
        for tunnels signaled via CRLDP."
    DEFVAL      { ipv4 }
    ::= { mplsTunnelCHopEntry 3 }

```

```

mplsTunnelCHopIpAddr OBJECT-TYPE
    SYNTAX      TeHopAddress
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "The Tunnel Hop Address for this tunnel hop."

```

The type of this address is determined by the value of the corresponding `mplsTunnelCHopAddrType`.

If `mplsTunnelCHopAddrType` is set to `unnum(4)`, then this value will contain the LSR Router ID of the unnumbered interface. Otherwise the agent should set this object to the zero-length string and the manager SHOULD ignore this object."

```
DEFVAL      { '00000000'h } -- IPv4 address 0.0.0.0
 ::= { mplsTunnelCHopEntry 4 }
```

`mplsTunnelCHopIpPrefixLen` OBJECT-TYPE

SYNTAX InetAddressPrefixLength

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If `mplsTunnelCHopAddrType` is set to `ipv4(1)` or `ipv6(2)`, then this value will contain an appropriate prefix length for the IP address in object `mplsTunnelCHopIpAddr`. Otherwise this value is irrelevant and should be ignored.

"

```
DEFVAL      { 32 }
 ::= { mplsTunnelCHopEntry 5 }
```

`mplsTunnelCHopAsNumber` OBJECT-TYPE

SYNTAX TeHopAddressAS

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If `mplsTunnelCHopAddrType` is set to `asnumber(3)`, then this value will contain the AS number of this hop. Otherwise the agent should set this object to zero-length string and the manager should ignore this."

```
::= { mplsTunnelCHopEntry 6 }
```

`mplsTunnelCHopAddrUnnum` OBJECT-TYPE

SYNTAX TeHopAddressUnnum

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"If `mplsTunnelCHopAddrType` is set to `unnum(4)`, then this value will contain the unnumbered interface identifier of this hop. This object should be used in conjunction with `mplsTunnelCHopIpAddr` which would contain the LSR Router ID in this case.

```

        Otherwise the agent should set this object to zero-
        length string and the manager should ignore this."
 ::= { mplsTunnelCHopEntry 7 }

mplsTunnelCHopLspId OBJECT-TYPE
    SYNTAX      MplsLSPID
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "If mplsTunnelCHopAddrType is set to lspid(5), then
         this value will contain the LSP ID of this hop.
         This object is otherwise insignificant and should
         contain a value of 0 to indicate this fact."
 ::= { mplsTunnelCHopEntry 8 }

mplsTunnelCHopType OBJECT-TYPE
    SYNTAX      INTEGER { strict(1),
                          loose(2)
                        }
    MAX-ACCESS  read-only
    STATUS      current
    DESCRIPTION
        "Denotes whether this tunnel hop is routed in a
         strict or loose fashion."
 ::= { mplsTunnelCHopEntry 9 }

-- End of mplsTunnelCHopTable

-- MPLS Tunnel Performance Table.

mplsTunnelPerfTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF MplsTunnelPerfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "This table provides per-tunnel instance MPLS
         performance information."
 ::= { mplsTeObjects 9 }

mplsTunnelPerfEntry OBJECT-TYPE
    SYNTAX      MplsTunnelPerfEntry
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "An entry in this table is created by the LSR for
         every tunnel. Its is an extension to
         mplsTunnelEntry."

```

```
AUGMENTS { mplsTunnelEntry }
 ::= { mplsTunnelPerfTable 1 }

MplsTunnelPerfEntry ::= SEQUENCE {
    mplsTunnelPerfPackets          Counter32,
    mplsTunnelPerfHCPackets       Counter64,
    mplsTunnelPerfErrors          Counter32,
    mplsTunnelPerfBytes           Counter32,
    mplsTunnelPerfHCBytes         Counter64
}

mplsTunnelPerfPackets OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets forwarded by the tunnel.
        This object should represents the 32-bit
        value of the least significant part of the
        64-bit value if both mplsTunnelPerfHCPackets
        is returned."
    ::= { mplsTunnelPerfEntry 1 }

mplsTunnelPerfHCPackets OBJECT-TYPE
    SYNTAX          Counter64
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "High capacity counter for number of packets
        forwarded by the tunnel. "
    ::= { mplsTunnelPerfEntry 2 }

mplsTunnelPerfErrors OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of packets dropped because of errors or for
        other reasons."
    ::= { mplsTunnelPerfEntry 3 }

mplsTunnelPerfBytes OBJECT-TYPE
    SYNTAX          Counter32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION
        "Number of bytes forwarded by the tunnel.
        This object should represents the 32-bit
```

```

        value of the least significant part of the
        64-bit value if both mplsTunnelPerfHCBytes
        is returned."
 ::= { mplsTunnelPerfEntry 4 }

mplsTunnelPerfHCBytes OBJECT-TYPE
SYNTAX      Counter64
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "High capacity counter for number of bytes forwarded
    by the tunnel."
 ::= { mplsTunnelPerfEntry 5 }

-- End of mplsTunnelPerfTable

-- CR-LDP Tunnel Resource Table

mplsTunnelCRLDPResTable OBJECT-TYPE
SYNTAX      SEQUENCE OF MplsTunnelCRLDPResEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "The mplsTunnelCRLDPResTable allows a manager to
    specify which CR-LDP-specific resources are desired
    for an MPLS tunnel if that tunnel is signaled using
    CR-LDP. Note that these attributes are in addition
    to those specified in mplsTunnelResourceTable. This
    table also allows several tunnels to point to a
    single entry in this table, implying that these
    tunnels should share resources."
 ::= { mplsTeObjects 10 }

mplsTunnelCRLDPResEntry OBJECT-TYPE
SYNTAX      MplsTunnelCRLDPResEntry
MAX-ACCESS  not-accessible
STATUS      current
DESCRIPTION
    "An entry in this table represents a set of resources
    for an MPLS tunnel established using CRLDP
    (mplsTunnelSignallingProto equal to crldp (3)). An
    entry can be created by a network administrator or
    by an SNMP agent as instructed by any MPLS
    signalling protocol."
INDEX { mplsTunnelResourceIndex }
 ::= { mplsTunnelCRLDPResTable 1 }

```

```

MplsTunnelCRLDPResEntry ::= SEQUENCE {
    mplsTunnelCRLDPResMeanBurstSize  MplsBurstSize,
    mplsTunnelCRLDPResExBurstSize    MplsBurstSize,
    mplsTunnelCRLDPResFrequency      INTEGER,
    mplsTunnelCRLDPResWeight         Unsigned32,
    mplsTunnelCRLDPResFlags          Unsigned32,
    mplsTunnelCRLDPResRowStatus      RowStatus,
    mplsTunnelCRLDPResStorageType    StorageType
}

mplsTunnelCRLDPResMeanBurstSize OBJECT-TYPE
    SYNTAX      MplsBurstSize
    UNITS       "bytes"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The mean burst size in bytes."
    ::= { mplsTunnelCRLDPResEntry 1 }

mplsTunnelCRLDPResExBurstSize OBJECT-TYPE
    SYNTAX      MplsBurstSize
    UNITS       "bytes"
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The Excess burst size in bytes."
    REFERENCE
        "CR-LDP Specification, Section 4.3."
    ::= { mplsTunnelCRLDPResEntry 2 }

mplsTunnelCRLDPResFrequency OBJECT-TYPE
    SYNTAX  INTEGER {
        unspecified(1),
        frequent(2),
        veryFrequent(3)
    }
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "The granularity of the availability of committed
        rate."
    REFERENCE
        "CR-LDP Specification, Section 4.3."
    ::= { mplsTunnelCRLDPResEntry 3 }

mplsTunnelCRLDPResWeight OBJECT-TYPE
    SYNTAX      Unsigned32(0..255)
    MAX-ACCESS  read-create

```



```

STATUS          current
DESCRIPTION
    "The relative weight for using excess bandwidth above
    its committed rate.  The value of 0 means that
    weight is not applicable for the CR-LSP."
REFERENCE
    "CR-LDP Specification, Section 4.3."
DEFVAL { 0 }
 ::= { mplsTunnelCRLDPResEntry 4 }

mplsTunnelCRLDPResFlags OBJECT-TYPE
SYNTAX          Unsigned32 (0..63)
MAX-ACCESS     read-create
STATUS         current
DESCRIPTION
    "The value of the 1 byte Flags conveyed as part of
    the traffic parameters during the establishment of
    the CRLSP.  The bits in this object are to be
    interpreted as follows.

    +-----+
    | Res |F6|F5|F4|F3|F2|F1|
    +-----+

    Res - These bits are reserved.  Zero on transmission.
          Ignored on receipt.
    F1 - Corresponds to the PDR.
    F2 - Corresponds to the PBS.
    F3 - Corresponds to the CDR.
    F4 - Corresponds to the CBS.
    F5 - Corresponds to the EBS.
    F6 - Corresponds to the Weight.

    Each flag if is a Negotiable Flag corresponding to a
    Traffic Parameter.  The Negotiable Flag value zero
    denotes Not Negotiable and value one denotes
    Negotiable."
REFERENCE
    "1. Section 4.3, Constraint-Based LSP Setup using
    LDP, Jamoussi (Editor), RFC 3212, January 2002"
DEFVAL { 0 }
 ::= { mplsTunnelCRLDPResEntry 5 }

mplsTunnelCRLDPResRowStatus 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 by the agent except
        mplsTunnelCRLDPResRowStatus and
        mplsTunnelCRLDPResStorageType."
 ::= { mplsTunnelCRLDPResEntry 6 }

mplsTunnelCRLDPResStorageType OBJECT-TYPE
    SYNTAX          StorageType
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION
        "The storage type for this CR-LDP Resource entry.
        Conceptual rows having the value 'permanent'
        need not allow write-access to any columnar
        objects in the row."
    DEFVAL { volatile }
    ::= { mplsTunnelCRLDPResEntry 7 }

-- Notifications.

mplsTunnelNotificationEnable OBJECT-TYPE
    SYNTAX          TruthValue
    MAX-ACCESS      read-write
    STATUS          current
    DESCRIPTION
        "If this object is true, then it enables the
        generation of mplsTunnelUp and mplsTunnelDown
        traps, otherwise these traps are not emitted."
    DEFVAL { false }
    ::= { mplsTeObjects 11 }

mplsTunnelUp NOTIFICATION-TYPE
    OBJECTS          {
        mplsTunnelAdminStatus,
        mplsTunnelOperStatus
    }
    STATUS          current
    DESCRIPTION
        "This notification is generated when a
        mplsTunnelOperStatus object for one of the
        configured tunnels is about to leave the down state
        and transition into some other state (but not into
        the notPresent state).  This other state is
        indicated by the included value of
        mplsTunnelOperStatus."

```

```
 ::= { mplsTeNotifications 1 }

mplsTunnelDown NOTIFICATION-TYPE
OBJECTS      {
    mplsTunnelAdminStatus,
    mplsTunnelOperStatus
}
STATUS      current
DESCRIPTION
    "This notification is generated when a
    mplsTunnelOperStatus object for one of the
    configured tunnels is about to enter the down state
    from some other state (but not from the notPresent
    state). This other state is indicated by the
    included value of mplsTunnelOperStatus."
 ::= { mplsTeNotifications 2 }

mplsTunnelRerouted NOTIFICATION-TYPE
OBJECTS      {
    mplsTunnelAdminStatus,
    mplsTunnelOperStatus
}
STATUS      current
DESCRIPTION
    "This notification is generated when a tunnel is
    rerouted. If the mplsTunnelARHopTable is used, then
    this tunnel instance's entry in the
    mplsTunnelARHopTable MAY contain the new path for
    this tunnel some time after this trap is issued by
    the agent."
 ::= { mplsTeNotifications 3 }

mplsTunnelReoptimized NOTIFICATION-TYPE
OBJECTS      {
    mplsTunnelAdminStatus,
    mplsTunnelOperStatus
}
STATUS      current
DESCRIPTION
    "This notification is generated when a tunnel is
    reoptimized. If the mplsTunnelARHopTable is used,
    then this tunnel instance's entry in the
    mplsTunnelARHopTable MAY contain the new path for
    this tunnel some time after this trap is issued by
    the agent."
 ::= { mplsTeNotifications 4 }

-- End of notifications.
```

```
-- Module compliance.

mplsTeGroups
  OBJECT IDENTIFIER ::= { mplsTeConformance 1 }

mplsTeCompliances
  OBJECT IDENTIFIER ::= { mplsTeConformance 2 }

-- Compliance requirement for fully compliant implementations.

mplsTeModuleFullCompliance MODULE-COMPLIANCE
  STATUS current
  DESCRIPTION
    "Compliance statement for agents that provide full
    support the MPLS-TE-STD-MIB module."

  MODULE IF-MIB -- The Interfaces Group MIB, RFC 2863.
    MANDATORY-GROUPS {
      ifGeneralInformationGroup,
      ifCounterDiscontinuityGroup
    }

  MODULE -- this module

    -- The mandatory group has to be implemented by all
    -- LSRs that originate/terminate ESLSPs/tunnels.
    -- In addition, depending on the type of tunnels
    -- supported, other groups become mandatory as
    -- explained below.

    MANDATORY-GROUPS {
      mplsTunnelGroup,
      mplsTunnelScalarGroup
    }

    GROUP mplsTunnelManualGroup
    DESCRIPTION
      "This group is mandatory for devices which support
      manual configuration of tunnels."

    GROUP mplsTunnelSignaledGroup
    DESCRIPTION
      "This group is mandatory for devices which support
      signaled tunnel set up."

    GROUP mplsTunnelIsNotIntfcGroup
    DESCRIPTION
      "This group is mandatory for devices which support
```

tunnels that are not interfaces."

GROUP mplsTunnelIsIntfcGroup

DESCRIPTION

"This group is mandatory for devices which support tunnels that are interfaces."

GROUP mplsTunnelCRLDPResOptionalGroup

DESCRIPTION

"Objects in this group are required by implementations supporting the CR-LDP protocol for signalling of TE tunnels."

GROUP mplsTeNotificationGroup

DESCRIPTION "This group is mandatory for those implementations which can implement the notifications contained in this group."

OBJECT mplsTunnelRowStatus

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 mplsTunnelHopRowStatus

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 mplsTunnelCRLDPResRowStatus

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."

::= { mplsTeCompliances 1 }

-- Compliance requirement for read-only implementations.

mplsTeModuleReadOnlyCompliance MODULE-COMPLIANCE

STATUS current

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

MODULE -- this module

    -- mplsTunnelTable

    MANDATORY-GROUPS {
        mplsTunnelGroup,
        mplsTunnelScalarGroup
    }

    GROUP mplsTunnelManualGroup
    DESCRIPTION
        "This group is mandatory for devices which support
        manual configuration of tunnels."

    GROUP mplsTunnelSignaledGroup
    DESCRIPTION
        "This group is mandatory for devices which support
        signaled tunnel set up."

    GROUP mplsTunnelIsNotIntfcGroup
    DESCRIPTION
        "This group is mandatory for devices which support
        tunnels that are not interfaces."

    GROUP mplsTunnelIsIntfcGroup
    DESCRIPTION
        "This group is mandatory for devices which support
        tunnels that are interfaces."

    GROUP mplsTunnelCRLDPResOptionalGroup
    DESCRIPTION
        "Objects in this group are required by
        implementations supporting the CR-LDP protocol for
        signalling of TE tunnels."

    GROUP mplsTeNotificationGroup
    DESCRIPTION "This group is mandatory for those implementations
        which can implement the notifications
        contained in this group."

    -- mplsTunnelTable
```

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

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

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

OBJECT mplsTunnelIfIndex
DESCRIPTION
"Write access is not required."

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

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

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

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

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

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

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

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

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

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

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

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

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

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

OBJECT mplsTunnelAdminStatus
SYNTAX INTEGER { up (1), down (2) }
MIN-ACCESS read-only
DESCRIPTION
"Only up and down states must be supported. Write
access is not required."

OBJECT mplsTunnelRowStatus


```
SYNTAX      RowStatus { active(1) }
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

-- mplsTunnelHopTable

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

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

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

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

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

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

OBJECT      mplsTunnelHopType
SYNTAX      INTEGER { strict(1) }
MIN-ACCESS  read-only
DESCRIPTION "loose(2) need not be supported. Write access is
            not required."

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

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

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

OBJECT      mplsTunnelHopRowStatus
SYNTAX      RowStatus { active(1) }
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

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

-- mplsTunnelResourceTable

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

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

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

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

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

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

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

OBJECT      mplsTunnelResourceRowStatus
SYNTAX      RowStatus { active(1) }
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."
```

```

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

-- mplsTunnelCRLDPResTable

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

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

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

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

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

OBJECT      mplsTunnelCRLDPResRowStatus
SYNTAX      RowStatus { active(1) }
MIN-ACCESS  read-only
DESCRIPTION "Write access is not required."

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

 ::= { mplsTeCompliances 2 }

-- Units of conformance.

mplsTunnelGroup OBJECT-GROUP
OBJECTS {
    mplsTunnelIndexNext,
    mplsTunnelName,
    mplsTunnelDescr,
    mplsTunnelOwner,
    mplsTunnelXCPointer,
    mplsTunnelIfIndex,

```

mplsTunnelHopTableIndex,
mplsTunnelARHopTableIndex,
mplsTunnelCHopTableIndex,
mplsTunnelAdminStatus,
mplsTunnelOperStatus,
mplsTunnelRowStatus,
mplsTunnelNotificationEnable,
mplsTunnelStorageType,
mplsTunnelConfigured,
mplsTunnelActive,
mplsTunnelPrimaryInstance,
mplsTunnelPrimaryUpTime,
mplsTunnelPathChanges,
mplsTunnelLastPathChange,
mplsTunnelCreationTime,
mplsTunnelStateTransitions,
mplsTunnelIncludeAnyAffinity,
mplsTunnelIncludeAllAffinity,
mplsTunnelExcludeAnyAffinity,
mplsTunnelPerfPackets,
mplsTunnelPerfHCPackets,
mplsTunnelPerfErrors,
mplsTunnelPerfBytes,
mplsTunnelPerfHCBytes,
mplsTunnelResourcePointer,
mplsTunnelInstancePriority,
mplsTunnelPathInUse,
mplsTunnelRole,
mplsTunnelTotalUpTime,
mplsTunnelInstanceUpTime,
mplsTunnelResourceIndexNext,
mplsTunnelResourceMaxRate,
mplsTunnelResourceMeanRate,
mplsTunnelResourceMaxBurstSize,
mplsTunnelResourceMeanBurstSize,
mplsTunnelResourceExBurstSize,
mplsTunnelResourceFrequency,
mplsTunnelResourceWeight,
mplsTunnelResourceRowStatus,
mplsTunnelResourceStorageType,
mplsTunnelARHopAddrType,
mplsTunnelARHopIpAddr,
mplsTunnelARHopAddrUnnum,
mplsTunnelARHopLspId,
mplsTunnelCHopAddrType,
mplsTunnelCHopIpAddr,
mplsTunnelCHopIpPrefixLen,
mplsTunnelCHopAsNumber,

```

    mplsTunnelCHopAddrUnnum,
    mplsTunnelCHopLspId,
    mplsTunnelCHopType
}
STATUS current
DESCRIPTION
    "Necessary, but not sufficient, set of objects to
    implement tunnels. In addition, depending on the
    type of the tunnels supported (for example,
    manually configured or signaled, persistent or non-
    persistent, etc.), the following other groups
    defined below are mandatory: mplsTunnelManualGroup
    and/or mplsTunnelSignaledGroup,
    mplsTunnelIsNotIntfcGroup and/or
    mplsTunnelIsIntfcGroup."
 ::= { mplsTeGroups 1 }

mplsTunnelManualGroup OBJECT-GROUP
OBJECTS { mplsTunnelSignallingProto }
STATUS current
DESCRIPTION
    "Object(s) needed to implement manually configured
    tunnels."
 ::= { mplsTeGroups 2 }

mplsTunnelSignaledGroup OBJECT-GROUP
OBJECTS {
    mplsTunnelSetupPrio,
    mplsTunnelHoldingPrio,
    mplsTunnelSignallingProto,
    mplsTunnelLocalProtectInUse,
    mplsTunnelSessionAttributes,
    mplsTunnelHopListIndexNext,
    mplsTunnelHopAddrType,
    mplsTunnelHopIpAddr,
    mplsTunnelHopIpPrefixLen,
    mplsTunnelHopAddrUnnum,
    mplsTunnelHopAsNumber,
    mplsTunnelHopLspId,
    mplsTunnelHopType,
    mplsTunnelHopInclude,
    mplsTunnelHopPathOptionName,
    mplsTunnelHopEntryPathComp,
    mplsTunnelHopRowStatus,
    mplsTunnelHopStorageType
}
STATUS current
DESCRIPTION

```

```
        "Objects needed to implement signaled tunnels."
 ::= { mplsTeGroups 3 }

mplsTunnelScalarGroup OBJECT-GROUP
OBJECTS {
    mplsTunnelConfigured,
    mplsTunnelActive,
    mplsTunnelTEDistProto,
    mplsTunnelMaxHops,
    mplsTunnelNotificationMaxRate
}
STATUS current
DESCRIPTION
    "Scalar object needed to implement MPLS tunnels."
 ::= { mplsTeGroups 4 }

mplsTunnelIsIntfcGroup OBJECT-GROUP
OBJECTS { mplsTunnelIsIf }
STATUS current
DESCRIPTION
    "Objects needed to implement tunnels that are
     interfaces."
 ::= { mplsTeGroups 5 }

mplsTunnelIsNotIntfcGroup OBJECT-GROUP
OBJECTS { mplsTunnelIsIf }
STATUS current
DESCRIPTION
    "Objects needed to implement tunnels that are not
     interfaces."
 ::= { mplsTeGroups 6 }

mplsTunnelCRLDPResOptionalGroup OBJECT-GROUP
OBJECTS {
    mplsTunnelCRLDPResMeanBurstSize,
    mplsTunnelCRLDPResExBurstSize,
    mplsTunnelCRLDPResFrequency,
    mplsTunnelCRLDPResWeight,
    mplsTunnelCRLDPResFlags,
    mplsTunnelCRLDPResRowStatus,
    mplsTunnelCRLDPResStorageType
}
STATUS current
DESCRIPTION
    "Set of objects implemented for resources applicable
     for tunnels signaled using CR-LDP."
 ::= { mplsTeGroups 7 }
```

```

mplsTeNotificationGroup NOTIFICATION-GROUP
  NOTIFICATIONS {
    mplsTunnelUp,
    mplsTunnelDown,
    mplsTunnelRerouted,
    mplsTunnelReoptimized
  }
  STATUS current
  DESCRIPTION
    "Set of notifications implemented in this module.
     None is mandatory."
 ::= { mplsTeGroups 8 }

END

```

12. Security Considerations

It is clear that this MIB module is potentially useful for the monitoring of MPLS TE tunnels. This MIB module can also be used for the 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:

- the mplsTunnelTable, mplsTunnelHopTable, mplsTunnelResourceTable, and mplsTunnelCRLDPResTable collectively contain objects to provision MPLS tunnels, tunnel hops, and tunnel resources. Unauthorized access to objects in these tables, could result in disruption of traffic on the network. This is especially true if a tunnel has been established. 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 which implements this MIB. 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:

- the `mplsTunnelTable`, `mplsTunnelHopTable`, `mplsTunnelResourceTable`, `mplsTunnelARHopTable`, `mplsTunnelCHopTable`, `mplsTunnelPerfTable`, and `mplsTunnelCRLDPResTable` collectively show the MPLS-TE tunnel network topology and its performance characteristics. If an Administrator does not want to reveal this information, then these tables should be considered sensitive/vulnerable.

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 that SNMPv3 be deployed and cryptographic security enabled. 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 only those principals (users) that have legitimate rights to those objects.

13. Acknowledgments

We wish to thank Adrian Farrel, Bert Wijnen, Eric Gray, Joan Cucchiara, Patrick Kerharo, Paul Langille, Marcus Brunner, Mike MacFaden, and Mike Piecuch for their comments on this document.

Comments should be made directly to the MPLS mailing list at `mpls@uu.net`.

14. IANA Considerations

As described in [MPLSMGMT] and as requested in the MPLS-TC-STD-MIB [RFC3811], MPLS related standards track MIB modules should be rooted under the `mplsStdMIB` subtree. There are 4 MPLS MIB Modules contained in this document, each of the following "IANA Considerations" subsections requests IANA for a new assignment under the `mplsStdMIB` subtree. New assignments can only be made via a Standards Action as specified in [RFC2434].

14.1. IANA Considerations for MPLS-TE-STD-MIB

The IANA has assigned { mplsStdMIB 3 } to the MPLS-TE-STD-MIB module specified in this document.

15. References

15.1. Normative References

- [RFC2119] Bradner, S., "Key Words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [RFC2578] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [RFC2579] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [RFC2702] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, September 1999.
- [RFC2863] McCloghrie, K. and F. Kastenholtz, "The Interfaces Group MIB ", RFC 2863, June 2000.
- [RFC3031] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", RFC 3031, January 2001.
- [RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.
- [RFC3212] Jamoussi, B., Ed., Andersson, L., Callon, R, Dantu, R., Wu, L., Doolan, P., Worster, T., Feldman, N., Fredette, A., Girish, M., Gray, E., Heinanen, J., Kilty, T., and A. Malis, "Constraint-Based LSP Setup using LDP", RFC 3212, January 2002.
- [RFC3289] Baker, F., Chan, K., and A. Smith, "Management Information Base for the Differentiated Services Architecture", RFC 3289, May 2002.

- [RFC3291] Daniele, M., Haberman, B., Routhier, S., and J. Schoenwaelder, "Textual Conventions for Internet Network Addresses", RFC 3291, May 2002.
- [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.
- [RFC3811] Nadeau, T. and J. Cucchiara, "Definition of Textual Conventions and for Multiprotocol Label Switching (MPLS) Management", RFC 3811, June 2004.
- [RFC3813] Srinivasan, C., Viswanathan, A., and T. Nadeau, "Multiprotocol Label Switching (MPLS) Label Switching (LSR) Router Management Information Base (MIB)", RFC 3813, June 2004.

15.2. Informative References

- [MPLSMGMT] Nadeau, T., Srinivasan, C., and A. Farrel, "Multiprotocol Label Switching (MPLS) Management Overview", Work in Progress, September 2003.
- [RFC2434] Narten, T. and H. Alvestrand., "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.
- [RFC3410] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statement for Internet Standard Management Framework", RFC 3410, December 2002.

16. Authors' Addresses

Cheenu Srinivasan
Bloomberg L.P.
499 Park Ave., New York, NY 10022

Phone: +1-212-893-3682
EMail: cheenu@bloomberg.net

Arun Viswanathan
Force10 Networks, Inc.
1440 McCarthy Blvd
Milpitas, CA 95035

Phone: +1-408-571-3516
EMail: arunv@force10networks.com

Thomas D. Nadeau
Cisco Systems, Inc.
300 Apollo Drive
Chelmsford, MA 01824

Phone: +1-978-244-3051
EMail: tnadeau@cisco.com

17. Full Copyright Statement

Copyright (C) The Internet Society (2004). 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.

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.