

A Traffic Engineering (TE) 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 (2005).

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 Traffic Engineered (TE) Tunnels; for example, Multi-Protocol Label Switched Paths.

Table of Contents

1.	Introduction	2
1.1.	Specification of Requirements.	2
2.	The Internet-Standard Management Framework	2
3.	Overview of the MIB Module	2
3.1.	Traffic Engineering Information.	3
3.2.	Traffic Tunnel Information	3
3.3.	Path Information	3
3.4.	Hop Information.	4
3.5.	Relationship with Other MIB Modules.	4
4.	Creating, Modifying, and Deleting a TE Tunnel.	4
5.	MIB Specification.	5
6.	References	40
6.1.	Normative References	40
6.2.	Informative References	40
7.	Security Considerations.	41
	Acknowledgments.	42
	Author's Address	43
	Full Copyright Statement	44

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 Traffic Engineered (TE) Tunnels; for example, Multi-Protocol Label Switched Paths ([7], [8]). The MIB module defined by this memo allows one to configure TE Tunnels, to assign one or more paths to a Tunnel, and to monitor operational aspects of the Tunnel, such as the number of octets and packets that have passed through the Tunnel.

As it stands, this MIB module can only be used to configure or monitor a TE Tunnel at its ingress. The ingress is then expected to use some protocol (such as RSVP-TE) to signal the other routers in the path the information they need to set up the tunnel. The extension of this module for use at other points of a Tunnel is for further study.

1.1. Specification of Requirements

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to Section 7 of RFC 3410 [8].

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 [2], STD 58, RFC 2579 [3] and STD 58, RFC 2580 [4].

3. Overview of the MIB Module

The Traffic Engineering MIB module consists of four parts:

- 1) Traffic Engineering information,
- 2) a table of Traffic Engineering Tunnels,
- 3) a table of Paths that tunnels take, and
- 4) a table of Hops that make up a tunnel path.

The MIB module also has statements for minimal and full compliance.

The following subsections give an overview of each part. All objects are mandatory. For minimal compliance, all objects MAY be implemented read-only; for full compliance, all objects must be implemented to their stated MAX-ACCESS capabilities. Notifications are optional.

3.1. Traffic Engineering Information

This part contains information about the Link State Protocols used to carry TE information, the signaling protocols used to set up Traffic Tunnels, the number of Traffic Tunnels that have been configured and that are operational, and a mapping of Administrative Group (called Resource Classes in [7]) numbers to names.

3.2. Traffic Tunnel Information

This part contains a table of Traffic Tunnels and information about each one. This information includes the Tunnel name, its configuration information, its operational information, and the active path(s) that the Tunnel takes.

Configuration information includes the end points of the Traffic Tunnel, and the number of configured paths for the Traffic Tunnel.

Operational information includes the current state (up/down), the count of octets and packets sent on the Traffic Tunnel, how long it has been up, and how many state transitions the Traffic Tunnel has had.

Operational path information includes the number of operational paths, the number of path changes, and when the last path change was.

3.3. Path Information

A Tunnel is a logical entity. An instantiation of a Tunnel is one or more Paths; each Path has a route (also called Explicit Route) or sequence of hops. A Path is indexed by a dual index: The primary index is that of the Tunnel to which the Path belongs; the secondary index is that of the Path itself.

The configured information for a Path consists of the constraints for the Path and a configured route.

The operational information consists of the Path status, the computed route (i.e., the route that was computed to satisfy the constraints), and the actual path as recorded by the signaling protocol.

3.4. Hop Information

A path consists of a sequence of hops. A hop can be loose (meaning that the path eventually traverses the specified node) or strict (meaning that the specified node and possibly the link must be the next node in the path). A hop can be specified as an IPv4 address, an IPv6 address, an Autonomous System number or an unnumbered interface index [5].

The Hop Table contains all hops for all paths on a given router. It is organized as follows. There is a primary index that identifies a list of hops and a secondary index that identifies individual hops. Thus, to get the sequence of recorded hops for a path, one looks up the path's `tePathRecordedRoute`, which is a primary index into the Hop Table. Then to get the list of actual hops in order for the recorded path, one uses a secondary index of 1, 2,

3.5. Relationship with Other MIB Modules

A TE Tunnel can extend objects from two other MIB modules; one is the Interfaces MIB [10], and the other is the IP Tunnel MIB [11]. The mechanism for doing so is to assign the TE Tunnel index (`teTunnelIndex`) with a valid `ifIndex` value in `ifTable`.

If a TE Tunnel is deemed an interface, a new interface object is created and assigned an `ifIndex` value in `ifTable`. Then a TE Tunnel object is created, setting `teTunnelIndex` to the same value as the interface index.

If (and only if) a TE Tunnel is considered an interface, it may also be considered an IP tunnel (if the encapsulation of the TE Tunnel is IP). In that case, the interface associated with the TE Tunnel should have its `ifType` set to `tunnel(131)`.

If a TE Tunnel is not considered an interface, then the TE Tunnel index (`teTunnelIndex`) SHOULD be set to a value at least 2^{24} , so that it is distinct from normal interfaces.

4. Creating, Modifying, and Deleting a TE Tunnel

To create a TE Tunnel, one first obtains a free Tunnel index by using the object `teNextTunnelIndex`. One then creates the Tunnel, including all parameters, either as `createAndGo` or `createAndWait`. Then, TE Paths for this Tunnel can be created by using the `teTunnelNextPathIndex` object, again as `createAndGo` or `createAndWait`. A particular Path is computed and signaled when both the Path and the enclosing Tunnel have `RowStatus` 'active'.

To build a Path's configured route, one first gets a free PathHop index by using `teNextPathHopIndex`, and then builds the route hop-by-hop using the secondary index, setting the `AddrType`, `Address`, and `HopType` for each Hop. Finally, one sets the `tePathConfiguredRoute` in the Path to the PathHop index obtained.

Modifying certain properties of a TE Tunnel or a TE Path may require setting the `RowStatus` of the Tunnel (or Path) to 'notInService' before making the changes and then setting the `RowStatus` of the Tunnel (or Path) back to 'active' to re-signal all Paths of the Tunnel (or the modified Path).

A TE Tunnel and all its Paths can be deleted by setting the Tunnel's `RowStatus` to 'destroy'. A specific Path within a Tunnel can be destroyed by setting that Path's `RowStatus` to 'destroy'.

5. MIB Specification

This MIB module IMPORTS objects from RFCs 2578 [2], 2579 [3], 2580 [3], 3411 [6], and 3811 [5] and it also has REFERENCE clauses to RFCs 3209 [8] and 3212 [12].

TE-MIB DEFINITIONS ::= BEGIN

IMPORTS

```

MODULE-IDENTITY, OBJECT-TYPE,
NOTIFICATION-TYPE, mib-2,
Integer32, Gauge32, Counter32,
Counter64, Unsigned32, TimeTicks          FROM SNMPv2-SMI

RowStatus, StorageType, TimeStamp,
TruthValue                                FROM SNMPv2-TC

SnmpAdminString                           FROM SNMP-FRAMEWORK-MIB

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

TeHopAddress, TeHopAddressType,
MplsBitRate                               FROM MPLS-TC-STD-MIB;
```

teMIB MODULE-IDENTITY

```

LAST-UPDATED "200501040000Z"              -- 01 January 2005
ORGANIZATION "IETF Traffic Engineering Working Group"
CONTACT-INFO "
    Editor:          Kireeti Kompella
                   Postal: Juniper Networks, Inc.
                   1194 Mathilda Ave
```

Sunnyvale, CA 94089
 Tel: +1 408 745 2000
 E-mail: kireeti@juniper.net

The IETF Traffic Engineering Working Group is chaired by Jim Boyle and Ed Kern.

WG Mailing List information:

General Discussion: te-wg@ops.ietf.org
 To Subscribe: te-wg-request@ops.ietf.org
 In Body: subscribe
 Archive: ftp://ops.ietf.org/pub/lists

Comments on the MIB module should be sent to the mailing list. The archives for this mailing list should be consulted for previous discussion on this MIB.

DESCRIPTION "The Traffic Engineering MIB module.

Copyright (C) The Internet Society (2005). This version of this MIB module is part of RFC 3970; see the RFC itself for full legal notices.

-- revision history

REVISION "200501040000Z" -- 01 January 2005
 DESCRIPTION "Initial version, published as RFC 3970."
 ::= { mib-2 122 }

-- Top level objects

teMIBNotifications OBJECT IDENTIFIER ::= { teMIB 0 }
 teMIBObjects OBJECT IDENTIFIER ::= { teMIB 1 }
 teMIBConformance OBJECT IDENTIFIER ::= { teMIB 2 }

-- *****

--

-- TE MIB Objects

--

-- TE Info

teInfo OBJECT IDENTIFIER ::= { teMIBObjects 1 }

teDistProtocol OBJECT-TYPE

```

SYNTAX          BITS {
                  other(0),
                  isis(1),
                  ospf(2)
                }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "IGP used to distribute Traffic Engineering
                information and topology to each device for the
                purpose of automatic path computation. More than
                one IGP may be used to distribute TE information.
                "
 ::= { teInfo 1 }

teSignalingProto OBJECT-TYPE
SYNTAX          BITS {
                  other(0),
                  rsvpte(1),
                  crldp(2),
                  static(3)      -- static configuration
                }
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "Traffic Engineering signaling protocols supported
                by this device. More than one protocol may be
                supported.
                "
REFERENCE      "For a description of RSVP-TE, see RFC 3209;
                for CR-LDP, see RFC 3212.
                "
 ::= { teInfo 2 }

teNotificationEnable OBJECT-TYPE
SYNTAX          TruthValue
MAX-ACCESS      read-write
STATUS          current
DESCRIPTION     "If this object is true, then it enables the
                generation of notifications from this MIB module.
                Otherwise notifications are not generated.
                "
DEFVAL { false }
 ::= { teInfo 3 }

teNextTunnelIndex OBJECT-TYPE
SYNTAX          Unsigned32
MAX-ACCESS      read-only
STATUS          current
DESCRIPTION     "An integer that may be used as a new Index in the

```

teTunnelTable.

The special value of 0 indicates that no more new entries can be created in that table.

When this MIB module is used for configuration, this object always contains a legal value (if non-zero) for an index that is not currently used in that table. The Command Generator (Network Management Application) reads this variable and uses the (non-zero) value read when creating a new row with an SNMP SET. When the SET is performed, 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.

"

::= { teInfo 4 }

teNextPathHopIndex OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "An integer that may be used as a new Index in the tePathHopTable.

The special value of 0 indicates that no more new entries can be created in that table.

When this MIB module is used for configuration, this object always contains a legal value (if non-zero) for an index that is not currently used in that table. The Command Generator (Network Management Application) reads this variable and uses the (non-zero) value read when creating a new row with an SNMP SET. When the SET is performed, 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.

"

::= { teInfo 5 }

teConfiguredTunnels OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "Number of currently configured Tunnels."

::= { teInfo 6 }

teActiveTunnels OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "Number of currently active Tunnels."

::= { teInfo 7 }

tePrimaryTunnels OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "Number of currently active Tunnels running on their primary paths."

"

::= { teInfo 8 }

teAdminGroupTable OBJECT-TYPE

SYNTAX SEQUENCE OF TeAdminGroupEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION "A mapping of configured administrative groups. Each entry represents an Administrative Group and provides a name and index for the group. Administrative groups are used to label links in the Traffic Engineering topology in order to place constraints (include and exclude) on Tunnel paths.

A groupName can only be linked to one group number. The groupNumber is the number assigned to the administrative group used in constraints, such as tePathIncludeAny or tePathIncludeAll.

"

```
::= { teInfo 9 }
```

```
teAdminGroupEntry OBJECT-TYPE
```

```
SYNTAX TeAdminGroupEntry
```

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

```
STATUS current
```

```
DESCRIPTION "A mapping between a configured group number and
its human-readable name. The group number should
be between 1 and 32, inclusive. Group number n
represents bit number (n-1) in the bit vector for
Include/Exclude constraints.
```

```
All entries in this table MUST be kept in stable
storage so that they will re-appear in case of a
restart/reboot.
```

```
"
```

```
INDEX { teAdminGroupNumber }
```

```
::= { teAdminGroupTable 1 }
```

```
TeAdminGroupEntry ::=
```

```
SEQUENCE {
```

```
teAdminGroupNumber Integer32,
```

```
teAdminGroupName SnmpAdminString,
```

```
teAdminGroupRowStatus RowStatus
```

```
}
```

```
teAdminGroupNumber OBJECT-TYPE
```

```
SYNTAX Integer32 (1..32)
```

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

```
STATUS current
```

```
DESCRIPTION "Index of the administrative group."
```

```
::= { teAdminGroupEntry 1 }
```

```
teAdminGroupName OBJECT-TYPE
```

```
SYNTAX SnmpAdminString (SIZE (1..32))
```

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

```
STATUS current
```

```
DESCRIPTION "Name of the administrative group."
```

```
::= { teAdminGroupEntry 2 }
```

```
teAdminGroupRowStatus OBJECT-TYPE
```

```
SYNTAX RowStatus
```

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

```
STATUS current
```

```
DESCRIPTION "The status of this conceptual row.
```

```
The value of this object has no effect on whether
other objects in this conceptual row can be
```

```

        modified.
    "
    ::= { teAdminGroupEntry 3 }

-- Tunnel Table

teTunnelTable      OBJECT-TYPE
    SYNTAX          SEQUENCE OF TeTunnelEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION     "Table of Configured Traffic Tunnels."
    ::= { teMIBObjects 2 }

teTunnelEntry      OBJECT-TYPE
    SYNTAX          TeTunnelEntry
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION     "Entry containing information about a particular
                    Traffic Tunnel.
                    "
    INDEX           { teTunnelIndex }
    ::= { teTunnelTable 1 }

TeTunnelEntry ::=
    SEQUENCE {
        teTunnelIndex                Unsigned32,
        teTunnelName                  SnmpAdminString,
        teTunnelNextPathIndex         Unsigned32,
        -- Conceptual row information:
        teTunnelRowStatus              RowStatus,
        teTunnelStorageType            StorageType,
        -- Address information:
        teTunnelSourceAddressType      TeHopAddressType,
        teTunnelSourceAddress          TeHopAddress,
        teTunnelDestinationAddressType TeHopAddressType,
        teTunnelDestinationAddress     TeHopAddress,
        -- State/performance information:
        teTunnelState                  INTEGER,
        teTunnelDiscontinuityTimer     TimeStamp,
        teTunnelOctets                 Counter64,
        teTunnelPackets                Counter64,
        teTunnelLPOctets               Counter32,
        teTunnelLPPackets              Counter32,
        teTunnelAge                    TimeTicks,
        teTunnelTimeUp                 TimeTicks,
        teTunnelPrimaryTimeUp          TimeTicks,
        teTunnelTransitions            Counter32,
        teTunnelLastTransition         TimeTicks,
    }

```

```

        teTunnelPathChanges          Counter32,
        teTunnelLastPathChange       TimeTicks,
        teTunnelConfiguredPaths      Gauge32,
        teTunnelStandbyPaths         Gauge32,
        teTunnelOperationalPaths     Gauge32
    }

teTunnelIndex      OBJECT-TYPE
    SYNTAX          Unsigned32 (1..4294967295)
    MAX-ACCESS      not-accessible
    STATUS          current
    DESCRIPTION     "A unique index that identifies a Tunnel.  If the TE
                    Tunnel is considered an interface, then this index
                    must match the interface index of the corresponding
                    interface.  Otherwise, this index must be at least
                    2^24, so that it does not overlap with any existing
                    interface index.
                    "
    ::= { teTunnelEntry 1 }

teTunnelName       OBJECT-TYPE
    SYNTAX          SnmpAdminString (SIZE (1..32))
    MAX-ACCESS      read-create
    STATUS          current
    DESCRIPTION     "Name of the Traffic Tunnel.

                    Note that the name of a Tunnel MUST be unique.
                    When a SET request contains a name that is already
                    in use for another entry, then the implementation
                    must return an inconsistentValue error.

                    The value of this object cannot be changed if the
                    if the value of the corresponding teTunnelRowStatus
                    object is 'active'.
                    "
    ::= { teTunnelEntry 2 }

teTunnelNextPathIndex OBJECT-TYPE
    SYNTAX          Unsigned32
    MAX-ACCESS      read-only
    STATUS          current
    DESCRIPTION     "An integer that may be used as a new Index for the
                    next Path in this Tunnel.

                    The special value of 0 indicates that no more Paths
                    can be created for this Tunnel, or that no more new
                    entries can be created in tePathTable.

```

When this MIB module is used for configuration, this object always contains a legal value (if non-zero) for an index that is not currently used in that table. The Command Generator (Network Management Application) reads this variable and uses the (non-zero) value read when creating a new row with an SNMP SET. When the SET is performed, 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.

"

```
::= { teTunnelEntry 3 }
```

teTunnelRowStatus OBJECT-TYPE

SYNTAX RowStatus

MAX-ACCESS read-create

STATUS current

DESCRIPTION "The status of this conceptual row.

When the value of this object is 'active', then the values for the corresponding objects teTunnelName, teTunnelSourceAddressType, teTunnelSourceAddress, teTunnelDestinationAddressType, and teTunnelDestinationAddress cannot be changed.

"

```
::= { teTunnelEntry 4 }
```

teTunnelStorageType OBJECT-TYPE

SYNTAX StorageType

MAX-ACCESS read-create

STATUS current

DESCRIPTION "The storage type for this conceptual row.

Conceptual rows having the value 'permanent' need not allow write-access to any columnar objects in the row.

"

```
::= { teTunnelEntry 5 }
```

`teTunnelSourceAddressType OBJECT-TYPE``SYNTAX TeHopAddressType``MAX-ACCESS read-create``STATUS current`

`DESCRIPTION` "The type of Traffic Engineered Tunnel hop address for the source of this Tunnel. Typically, this address type is IPv4 or IPv6, with a prefix length of 32 or 128, respectively. If the TE Tunnel path is being computed by a path computation server, however, it is possible to use more flexible source address types, such as AS numbers or prefix lengths less than host address lengths.

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

"

`::= { teTunnelEntry 6 }`

`teTunnelSourceAddress OBJECT-TYPE``SYNTAX TeHopAddress``MAX-ACCESS read-create``STATUS current`

`DESCRIPTION` "The Source Traffic Engineered Tunnel hop address of this Tunnel.

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

Note that the source and destination addresses of a Tunnel can be different address types.

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

"

`::= { teTunnelEntry 7 }`

`teTunnelDestinationAddressType OBJECT-TYPE``SYNTAX TeHopAddressType``MAX-ACCESS read-create``STATUS current`

`DESCRIPTION` "The type of Traffic Engineered Tunnel hop address for the destination of this Tunnel.

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

```

"
 ::= { teTunnelEntry 8 }

teTunnelDestinationAddress OBJECT-TYPE
    SYNTAX      TeHopAddress
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION  "The Destination Traffic Engineered Tunnel hop
                  address of this Tunnel.

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

                  Note that source and destination addresses of a
                  Tunnel can be different address types.

                  The value of this object cannot be changed
                  if the value of the corresponding teTunnelRowStatus
                  object is 'active'."
"
 ::= { teTunnelEntry 9 }

teTunnelState OBJECT-TYPE
    SYNTAX      INTEGER {
                    unknown(1),
                    up(2),
                    down(3),
                    testing(4)
                }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "The operational state of the Tunnel."
 ::= { teTunnelEntry 10 }

teTunnelDiscontinuityTimer OBJECT-TYPE
    SYNTAX      TimeStamp
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "The value of sysUpTime on the most recent occasion
                  at which any one or more of this tunnel's counters
                  suffered a discontinuity. The relevant counters
                  are teTunnelOctets, teTunnelPackets,
                  teTunnelLPOctets, and teTunnelLPPackets. If no such
                  discontinuities have occurred since the last
                  re-initialization of the local management subsystem
                  then this object contains a zero value."
"
 ::= { teTunnelEntry 11 }

```

```
teTunnelOctets    OBJECT-TYPE
    SYNTAX        Counter64
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION   "The number of octets that have been forwarded over
                    the Tunnel.

                    Discontinuities in the value of this counter can
                    occur at re-initialization of the management system,
                    and at other times, as indicated by the value of
                    teTunnelDiscontinuityTimer.
                    "
    ::= { teTunnelEntry 12 }

teTunnelPackets   OBJECT-TYPE
    SYNTAX        Counter64
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION   "The number of packets that have been forwarded over
                    the Tunnel.

                    Discontinuities in the value of this counter can
                    occur at re-initialization of the management system
                    and at other times, as indicated by the value of
                    teTunnelDiscontinuityTimer.
                    "
    ::= { teTunnelEntry 13 }

teTunnelLPOctets  OBJECT-TYPE
    SYNTAX        Counter32
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION   "The number of octets that have been forwarded over
                    the Tunnel.

                    Discontinuities in the value of this counter can
                    occur at re-initialization of the management system
                    and at other times, as indicated by the value of
                    teTunnelDiscontinuityTimer.
                    "
    ::= { teTunnelEntry 14 }

teTunnelLPPackets OBJECT-TYPE
    SYNTAX        Counter32
    MAX-ACCESS    read-only
    STATUS        current
    DESCRIPTION   "The number of packets that have been forwarded over
                    the Tunnel.
```


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

"
 ::= { teTunnelEntry 15 }

teTunnelAge OBJECT-TYPE
 SYNTAX TimeTicks
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION "The age (i.e., time from creation of this conceptual
 row till now) of this Tunnel in hundredths of a
 second. Note that because TimeTicks wrap in about
 16 months, this value is best used in interval
 measurements.

"
 ::= { teTunnelEntry 16 }

teTunnelTimeUp OBJECT-TYPE
 SYNTAX TimeTicks
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION "The total time in hundredths of a second that this
 Tunnel has been operational. Note that because
 TimeTicks wrap in about 16 months, this value is
 best used in interval measurements.

An example of usage of this object would be to compute the percentage up time over a period of time by obtaining values of teTunnelAge and teTunnelTimeUp at two points in time and computing the following ratio:

$$\frac{((\text{teTunnelTimeUp2} - \text{teTunnelTimeUp1}) / (\text{teTunnelAge2} - \text{teTunnelAge1})) * 100 \%}{}$$
 In doing so, the management station must account for wrapping of the values of teTunnelAge and teTunnelTimeUp between the two measurements.

"
 ::= { teTunnelEntry 17 }

teTunnelPrimaryTimeUp OBJECT-TYPE
 SYNTAX TimeTicks
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION "The total time in hundredths of a second that this
 Tunnel's primary path has been operational. Note
 that because TimeTicks wrap in about 16 months, this

value is best used in interval measurements.

An example of usage of this field would be to compute what percentage of time that a TE Tunnel was on the primary path over a period of time by computing

$((\text{teTunnelPrimaryTimeUp2} - \text{teTunnelPrimaryTimeUp1}) / (\text{teTunnelTimeUp2} - \text{teTunnelTimeUp1})) * 100 \%$. In doing so, the management station must account for wrapping of the values of `teTunnelPrimaryTimeUp` and `teTunnelTimeUp` between the two measurements.

"

::= { teTunnelEntry 18 }

teTunnelTransitions OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The number of operational state transitions (up -> down and down -> up) this Tunnel has undergone."

"

::= { teTunnelEntry 19 }

teTunnelLastTransition OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The time in hundredths of a second since the last operational state transition occurred on this Tunnel."

Note that if the last transition was over 16 months ago, this value will be inaccurate.

"

::= { teTunnelEntry 20 }

teTunnelPathChanges OBJECT-TYPE

SYNTAX Counter32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The number of path changes this Tunnel has had."

::= { teTunnelEntry 21 }

teTunnelLastPathChange OBJECT-TYPE

SYNTAX TimeTicks

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The time in hundredths of a second since the last path change occurred on this Tunnel.

Note that if the last transition was over 16 months ago, this value will be inaccurate.

Path changes may be caused by network events or by reconfiguration that affects the path.

"

::= { teTunnelEntry 22 }

teTunnelConfiguredPaths OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The number of paths configured for this Tunnel."

::= { teTunnelEntry 23 }

teTunnelStandbyPaths OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The number of standby paths configured for this Tunnel.

"

::= { teTunnelEntry 24 }

teTunnelOperationalPaths OBJECT-TYPE

SYNTAX Gauge32

MAX-ACCESS read-only

STATUS current

DESCRIPTION "The number of operational paths for this Tunnel. This includes the path currently active, as well as operational standby paths.

"

::= { teTunnelEntry 25 }

-- *****

--

-- Tunnel Path Table

--

tePathTable OBJECT-TYPE

SYNTAX SEQUENCE OF TePathEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION "Table of Configured Traffic Tunnels."

::= { teMIBObjects 3 }

```

tePathEntry      OBJECT-TYPE
    SYNTAX        TePathEntry
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION   "Entry containing information about a particular
                   Traffic Tunnel.  Each Traffic Tunnel can have zero
                   or more Traffic Paths.

```

As a Traffic Path can only exist over an existing Traffic Tunnel, all tePathEntries with a value of n for teTunnelIndex MUST be removed by the implementation when the corresponding teTunnelEntry with a value of n for teTunnelIndex is removed.

```

"
INDEX            { teTunnelIndex, tePathIndex }
 ::= { tePathTable 1 }

```

```

TePathEntry ::=
    SEQUENCE {
        tePathIndex                Unsigned32,
        tePathName                  SnmpAdminString,
        -- Conceptual row information
        tePathRowStatus             RowStatus,
        tePathStorageType           StorageType,
        -- Path properties
        tePathType                   INTEGER,
        tePathConfiguredRoute        Unsigned32,
        tePathBandwidth              MplsBitRate,
        tePathIncludeAny             Unsigned32,
        tePathIncludeAll             Unsigned32,
        tePathExclude                Unsigned32,
        tePathSetupPriority           Integer32,
        tePathHoldPriority            Integer32,
        tePathProperties             BITS,
        -- Path status
        tePathOperStatus             INTEGER,
        tePathAdminStatus            INTEGER,
        tePathComputedRoute          Unsigned32,
        tePathRecordedRoute          Unsigned32
    }

```

```

tePathIndex      OBJECT-TYPE
    SYNTAX        Unsigned32 (1..4294967295)
    MAX-ACCESS    not-accessible
    STATUS        current
    DESCRIPTION   "An index that uniquely identifies a path within
                   a Tunnel.

```

The combination of <teTunnelIndex, tePathIndex> thus uniquely identifies a path among all paths on this router.

"

::= { tePathEntry 1 }

tePathName OBJECT-TYPE
 SYNTAX SnmpAdminString (SIZE(0..32))
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION "The name of this path.

A pathName must be unique within the set of paths over a single tunnel. If a SET request is received with a duplicate name, then the implementation MUST return an inconsistentValue error.

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

"

::= { tePathEntry 2 }

tePathRowStatus OBJECT-TYPE
 SYNTAX RowStatus
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION "The status of this conceptual row.

When the value of this object is 'active', then the value of tePathName cannot be changed. All other writable objects may be changed; however, these changes may affect traffic going over the TE tunnel or require the path to be computed and/or re-sigaled.

"

::= { tePathEntry 3 }

tePathStorageType OBJECT-TYPE
 SYNTAX StorageType
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION "The storage type for this conceptual row.

Conceptual rows having the value 'permanent' need not allow write-access to any columnar objects in the row.

"

```
::= { tePathEntry 4 }
```

```
tePathType OBJECT-TYPE
```

```
SYNTAX          INTEGER {
                    other(1),
                    primary(2),
                    standby(3),
                    secondary(4)
                }
```

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

```
STATUS          current
```

```
DESCRIPTION     "The type for this PathEntry; i.e., whether this path
                  is a primary path, a standby path, or a secondary
                  path."
```

```
::= { tePathEntry 5 }
```

```
tePathConfiguredRoute OBJECT-TYPE
```

```
SYNTAX          Unsigned32
```

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

```
STATUS          current
```

```
DESCRIPTION     "The route that this TE path is configured to follow;
                  i.e., an ordered list of hops. The value of this
                  object gives the primary index into the Hop Table.
                  The secondary index is the hop count in the path, so
                  to get the route, one could get the first hop with
                  index <tePathConfiguredRoute, 1> in the Hop Table
                  and do a getnext to get subsequent hops."
```

```
::= { tePathEntry 6 }
```

```
tePathBandwidth OBJECT-TYPE
```

```
SYNTAX          MplsBitRate
```

```
UNITS           "Kilobits per second"
```

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

```
STATUS          current
```

```
DESCRIPTION     "The configured bandwidth for this Tunnel,
                  in units of thousands of bits per second (Kbps)."
```

```
DEFVAL          { 0 }
```

```
::= { tePathEntry 7 }
```

```
tePathIncludeAny OBJECT-TYPE
```

```
SYNTAX          Unsigned32
```

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

```
STATUS          current
```

```
DESCRIPTION     "This is a configured set of administrative groups
                  specified as a bit vector (i.e., bit n is 1 if group
```

n is in the set, where n = 0 is the LSB). For each link that this path goes through, the link must have at least one of the groups specified in IncludeAny to be acceptable. If IncludeAny is zero, all links are acceptable.

```
"
DEFVAL      { 0 }
::= { tePathEntry 8 }
```

tePathIncludeAll OBJECT-TYPE

```
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "This is a configured set of administrative groups
            specified as a bit vector (i.e., bit n is 1 if group
            n is in the set, where n = 0 is the LSB). For each
            link that this path goes through, the link must have
            all of the groups specified in IncludeAll to be
            acceptable. If IncludeAll is zero, all links are
            acceptable."
"
DEFVAL      { 0 }
::= { tePathEntry 9 }
```

tePathExclude OBJECT-TYPE

```
SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "This is a configured set of administrative groups
            specified as a bit vector (i.e., bit n is 1 if group
            n is in the set, where n = 0 is the LSB). For each
            link that this path goes through, the link MUST have
            groups associated with it, and the intersection of
            the link's groups and the 'exclude' set MUST be
            null."
"
DEFVAL      { 0 }
::= { tePathEntry 10 }
```

tePathSetupPriority OBJECT-TYPE

```
SYNTAX      Integer32 (0..7)
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION "The setup priority configured for this path, with 0
            as the highest priority and 7 as the lowest."
"
DEFVAL      { 7 }
```

```

 ::= { tePathEntry 11 }

tePathHoldPriority OBJECT-TYPE
    SYNTAX      Integer32 (0..7)
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION  "The hold priority configured for this path, with 0
                  as the highest priority and 7 as the lowest.
                  "
    DEFVAL       { 0 }
    ::= { tePathEntry 12 }

tePathProperties OBJECT-TYPE
    SYNTAX      BITS {
                    recordRoute(0),
                    cspf(1),
                    makeBeforeBreak(2),
                    mergeable(3),
                    fastReroute(4),
                    protected(5)
                }
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION  "The set of configured properties for this path,
                  expressed as a bit map.  For example, if the path
                  supports 'make before break', then bit 2 is set.
                  "
    ::= { tePathEntry 13 }

tePathOperStatus OBJECT-TYPE
    SYNTAX      INTEGER {
                    unknown(0),
                    down(1),
                    testing(2),
                    dormant(3),
                    ready(4),
                    operational(5)
                }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "The operational status of the path:
                  unknown:
                  down:      Signaling failed.
                  testing:   Administratively set aside for testing.
                  dormant:   Not signaled (for a backup tunnel).
                  ready:     Signaled but not yet carrying traffic.
                  operational: Signaled and carrying traffic.
                  "

```



```
::= { tePathEntry 14 }
```

```
tePathAdminStatus OBJECT-TYPE
```

```
SYNTAX          INTEGER {
                    normal(1),
                    testing(2)
                }
```

```
MAX-ACCESS      read-create
STATUS          current
```

```
DESCRIPTION     "The operational status of the path:
                    normal:      Used normally for forwarding.
                    testing:     Administratively set aside for testing.
                "
```

```
::= { tePathEntry 15 }
```

```
tePathComputedRoute OBJECT-TYPE
```

```
SYNTAX          Unsigned32
```

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

```
STATUS          current
```

```
DESCRIPTION     "The route computed for this path, perhaps using
                    some form of Constraint-based Routing. The
                    algorithm is implementation dependent.
```

This object returns the computed route as an ordered list of hops. The value of this object gives the primary index into the Hop Table. The secondary index is the hop count in the path, so to get the route, one could get the first hop with index <tePathComputedRoute, 1> in the Hop Table and do a getnext to get subsequent hops.

A value of zero (0) means there is no computedRoute.

```
::= { tePathEntry 16 }
```

```
tePathRecordedRoute OBJECT-TYPE
```

```
SYNTAX          Unsigned32
```

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

```
STATUS          current
```

```
DESCRIPTION     "The route actually used for this path, as recorded
                    by the signaling protocol. This is again an ordered
                    list of hops; each hop is expected to be strict.
```

The value of this object gives the primary index into the Hop Table. The secondary index is the hop count in the path, so to get the route, one can get the first hop with index <tePathRecordedRoute, 1> in the Hop Table and do a getnext to get subsequent

hops.

A value of zero (0) means there is no recordedRoute.

```

"
 ::= { tePathEntry 17 }

-- *****
--
-- Tunnel Path Hop Table
--

tePathHopTable    OBJECT-TYPE
    SYNTAX         SEQUENCE OF TePathHopEntry
    MAX-ACCESS     not-accessible
    STATUS         current
    DESCRIPTION    "Table of Tunnel Path Hops."
    ::= { teMIBObjects 4 }

tePathHopEntry    OBJECT-TYPE
    SYNTAX         TePathHopEntry
    MAX-ACCESS     not-accessible
    STATUS         current
    DESCRIPTION    "Entry containing information about a particular
                    hop."
    INDEX          { teHopListIndex, tePathHopIndex }
    ::= { tePathHopTable 1 }

TePathHopEntry ::=
    SEQUENCE {
        teHopListIndex          Unsigned32,
        tePathHopIndex          Unsigned32,
        -- Conceptual row information
        tePathHopRowStatus      RowStatus,
        tePathHopStorageType    StorageType,
        tePathHopAddrType       TeHopAddressType,
        tePathHopAddress        TeHopAddress,
        tePathHopType           INTEGER
    }

teHopListIndex    OBJECT-TYPE
    SYNTAX         Unsigned32 (1..4294967295)
    MAX-ACCESS     not-accessible
    STATUS         current
    DESCRIPTION    "An index that identifies a list of hops. This is
                    the primary index to access hops."
    ::= { tePathHopEntry 1 }

```

```

tePathHopIndex      OBJECT-TYPE
    SYNTAX            Unsigned32 (1..4294967295)
    MAX-ACCESS        not-accessible
    STATUS            current
    DESCRIPTION       "An index that identifies a particular hop among the
                        list of hops for a path.  An index of i identifies
                        the ith hop.  This is the secondary index for a hop
                        entry.
                        "
    ::= { tePathHopEntry 2 }

tePathHopRowStatus  OBJECT-TYPE
    SYNTAX            RowStatus
    MAX-ACCESS        read-create
    STATUS            current
    DESCRIPTION       "The status of this conceptual row.

                        Any field in this table can be changed, even if the
                        value of this object is 'active'.  However, such a
                        change may cause traffic to be rerouted or even
                        disrupted.
                        "
    ::= { tePathHopEntry 3 }

tePathHopStorageType OBJECT-TYPE
    SYNTAX            StorageType
    MAX-ACCESS        read-create
    STATUS            current
    DESCRIPTION       "The storage type for this conceptual row.

                        Conceptual rows having the value 'permanent' need
                        not allow write-access to any columnar objects
                        in the row.
                        "
    ::= { tePathHopEntry 4 }

tePathHopAddrType  OBJECT-TYPE
    SYNTAX            TeHopAddressType
    MAX-ACCESS        read-create
    STATUS            current
    DESCRIPTION       "The type of Traffic Engineered Tunnel hop Address
                        of this hop.

                        The value of this object cannot be changed
                        if the value of the corresponding tePathRowStatus
                        object is 'active'.
                        "
    ::= { tePathHopEntry 5 }

```

```

tePathHopAddress OBJECT-TYPE
    SYNTAX      TeHopAddress
    MAX-ACCESS   read-create
    STATUS       current
    DESCRIPTION  "The Traffic Engineered Tunnel hop Address of this
                  hop.

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

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

```

```

 ::= { tePathHopEntry 6 }

```

```

tePathHopType OBJECT-TYPE
    SYNTAX      INTEGER {
                    unknown(0),
                    loose(1),
                    strict(2)
                }
    MAX-ACCESS   read-only
    STATUS       current
    DESCRIPTION  "The type of hop:
                  unknown:
                  loose:   This hop is a LOOSE hop.
                  strict:  This hop is a STRICT hop."

```

```

 ::= { tePathHopEntry 7 }

```

```

-- *****
--
-- TE Notifications
--

```

```

teTunnelUp NOTIFICATION-TYPE
    OBJECTS   { teTunnelName,
                tePathName } -- TunnelPath
    STATUS     current
    DESCRIPTION "A teTunnelUp notification is generated when the
                  Tunnel indexed by teTunnelName transitions to the
                  'up' state."

```

A tunnel is up when at least one of its paths is up.
 The tePathName is the name of the path whose
 transition to up made the tunnel go up.

This notification MUST be limited to at most one every minute, in case the tunnel flaps up and down.

"

::= { teMIBNotifications 1 }

teTunnelDown NOTIFICATION-TYPE
 OBJECTS { teTunnelName,
 tePathName } -- TunnelPath
 STATUS current
 DESCRIPTION "A teTunnelDown notification is generated when the Tunnel indexed by teTunnelName transitions to the 'down' state.

A tunnel is up when at least one of its paths is up. The tePathName is the name of the path whose transition to down made the tunnel go down.

This notification MUST be limited to at most one every minute, in case the tunnel flaps up and down.

"

::= { teMIBNotifications 2 }

teTunnelChanged NOTIFICATION-TYPE
 OBJECTS { teTunnelName,
 tePathName } -- toTunnelPath
 STATUS current
 DESCRIPTION "A teTunnelChanged notification is generated when an active path on the Tunnel indexed by teTunnelName changes or a new path becomes active. The value of tePathName is the new active path.

This notification MUST be limited to at most one every minute, in case the tunnel changes quickly.

"

::= { teMIBNotifications 3 }

teTunnelRerouted NOTIFICATION-TYPE
 OBJECTS { teTunnelName,
 tePathName } -- toTunnelPath
 STATUS current
 DESCRIPTION "A teTunnelRerouted notification is generated when an active path for the Tunnel indexed by teTunnelName stays the same, but its route changes.

This notification MUST be limited to at most one every minute, in case the tunnel reroutes quickly.

"

::= { teMIBNotifications 4 }

```
-- End of TE-MIB objects

-- *****
--
-- TE Compliance Statements
--

teGroups
    OBJECT IDENTIFIER ::= { teMIBConformance 1 }

teModuleCompliance
    OBJECT IDENTIFIER ::= { teMIBConformance 2 }

-- *****
--
-- TE object groups
--

teTrafficEngineeringGroup OBJECT-GROUP
    OBJECTS {
        teTunnelName,
        teTunnelNextPathIndex,
        teTunnelRowStatus,
        teTunnelStorageType,
        teTunnelSourceAddressType,
        teTunnelSourceAddress,
        teTunnelDestinationAddressType,
        teTunnelDestinationAddress,
        teTunnelState,
        teTunnelDiscontinuityTimer,
        teTunnelOctets,
        teTunnelPackets,
        teTunnelLPOctets,
        teTunnelLPPackets,
        teTunnelAge,
        teTunnelTimeUp,
        teTunnelPrimaryTimeUp,
        teTunnelTransitions,
        teTunnelLastTransition,
        teTunnelPathChanges,
        teTunnelLastPathChange,
        teTunnelConfiguredPaths,
        teTunnelStandbyPaths,
        teTunnelOperationalPaths,
        tePathBandwidth,
        tePathIncludeAny,
        tePathIncludeAll,
        tePathExclude,
```

```

        tePathSetupPriority,
        tePathHoldPriority,
        tePathProperties,
        tePathOperStatus,
        tePathAdminStatus,
        tePathComputedRoute,
        tePathRecordedRoute,
        teDistProtocol,
        teSignalingProto,
        teNotificationEnable,
        teNextTunnelIndex,
        teNextPathHopIndex,
        teAdminGroupName,
        teAdminGroupRowStatus,
        teConfiguredTunnels,
        teActiveTunnels,
        tePrimaryTunnels,
        tePathName,
        tePathType,
        tePathRowStatus,
        tePathStorageType,
        tePathConfiguredRoute,
        tePathHopRowStatus,
        tePathHopStorageType,
        tePathHopAddrType,
        tePathHopAddress,
        tePathHopType
    }
    STATUS          current
    DESCRIPTION "Objects for Traffic Engineering in this MIB module."
    ::= { teGroups 1 }

teNotificationGroup NOTIFICATION-GROUP
    NOTIFICATIONS {
        teTunnelUp,
        teTunnelDown,
        teTunnelChanged,
        teTunnelRerouted
    }
    STATUS          current
    DESCRIPTION "Notifications specified in this MIB module."
    ::= { teGroups 2 }

-- *****
--
-- TE compliance statements
--
--     There are four compliance statements: read-only and full

```

```
-- compliance for regular TE devices, and read-only and full
-- compliance for path computation servers.
--
```

```
teModuleReadOnlyCompliance MODULE-COMPLIANCE
```

```
    STATUS      current
```

```
    DESCRIPTION "When this MIB module is implemented without support
                  for read-create (i.e., in read-only mode), then such
                  an implementation can claim read-only compliance.
                  Such a device can be monitored but cannot be
                  configured with this MIB module.
                  "
```

```
MODULE      -- enclosing module, i.e., TE-MIB
```

```
    MANDATORY-GROUPS {
        teTrafficEngineeringGroup
    }
```

```
    GROUP      teNotificationGroup
```

```
    DESCRIPTION "Implementation of this group is optional."
```

```
    OBJECT      teNotificationEnable
```

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

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

```
    OBJECT      teAdminGroupName
```

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

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

```
    OBJECT      teAdminGroupRowStatus
```

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

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

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

```
    OBJECT      teTunnelName
```

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

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

```
    OBJECT      teTunnelRowStatus
```

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

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

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

```
    OBJECT      teTunnelStorageType
```

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

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


OBJECT teTunnelSourceAddressType
SYNTAX TeHopAddressType { ipv4(1), ipv6(2) }
MIN-ACCESS read-only
DESCRIPTION "Write access is not required. An
 implementation is only required to support
 IPv4 and IPv6 host addresses."

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

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

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

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

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

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

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

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

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

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

```

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

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

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

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

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

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

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

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

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

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

```

```
 ::= { teModuleCompliance 1 }
```

```
teModuleFullCompliance MODULE-COMPLIANCE
```

```
STATUS      current
```

```
DESCRIPTION "When this MIB module is implemented with support for
read-create, then the implementation can claim
full compliance. Such devices can be both
```

monitored and configured with this MIB module.
"

MODULE -- enclosing module, i.e., TE-MIB

MANDATORY-GROUPS {
 teTrafficEngineeringGroup
}

GROUP teNotificationGroup
DESCRIPTION "Implementation of this group is optional."

OBJECT teAdminGroupRowStatus
SYNTAX RowStatus { active(1) }
WRITE-SYNTAX RowStatus { createAndGo(4), destroy(6) }
DESCRIPTION "Support for notInService, createAndWait and
notReady is not required."
"

OBJECT teTunnelRowStatus
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 teTunnelSourceAddressType
SYNTAX TeHopAddressType { ipv4(1), ipv6(2) }
DESCRIPTION "Write access is required. An implementation is
only required to support IPv4 and IPv6 host
addresses."
"

OBJECT tePathRowStatus
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 tePathHopRowStatus
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.
                "
 ::= { teModuleCompliance 2 }

teModuleServerReadOnlyCompliance MODULE-COMPLIANCE
    STATUS          current
    DESCRIPTION "When this MIB module is implemented by a path
                computation server without support for read-create
                (i.e., in read-only mode), then the implementation
                can claim read-only compliance. Such
                a device can be monitored but cannot be
                configured with this MIB module.
                "

MODULE              -- enclosing module, i.e., TE-MIB

    MANDATORY-GROUPS {
        teTrafficEngineeringGroup
    }

    GROUP            teNotificationGroup
    DESCRIPTION "Implementation of this group is optional."

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

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

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

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

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

```

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

OBJECT teTunnelSourceAddressType
MIN-ACCESS read-only
DESCRIPTION "Write access is not required. A path
 computation server SHOULD implement all types
 of tunnel source address types."
 "

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

```
::= { teModuleCompliance 3 }
```

```
teModuleServerFullCompliance MODULE-COMPLIANCE
```

```
STATUS          current
DESCRIPTION     "When this MIB module is implemented by a path
                computation server with support for read-create,
                then the implementation can claim full
                compliance.
                "

MODULE          -- enclosing module, i.e., TE-MIB
MANDATORY-GROUPS {
    teTrafficEngineeringGroup
}

GROUP          teNotificationGroup
DESCRIPTION     "Implementation of this group is optional."

OBJECT          teAdminGroupRowStatus
SYNTAX          RowStatus { active(1) }
WRITE-SYNTAX    RowStatus { createAndGo(4), destroy(6) }
DESCRIPTION     "Support for notInService, createAndWait, and
                notReady is not required.
                "

OBJECT          teTunnelRowStatus
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          teTunnelSourceAddressType
DESCRIPTION     "Write access is required. An implementation
                of a path computation server SHOULD support all
                types of tunnel source address types.
                "

OBJECT          tePathRowStatus
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          tePathHopRowStatus
```

```
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.
            "
```

```
::= { teModuleCompliance 4 }
```

END

6. References

6.1. Normative References

- [1] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [2] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.
- [3] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.
- [4] McCloghrie, K., Perkins, D., and J. Schoenwaelder, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.
- [5] Nadeau, T. and J. Cucchiara, "Definitions of Textual Conventions (TCs) for Multiprotocol Label Switching (MPLS) Management", RFC 3811, June 2004.
- [6] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.
- [7] Awduche, D., Malcolm, J., Agogbua, J., O'Dell, M., and J. McManus, "Requirements for Traffic Engineering Over MPLS", RFC 2702, September 1999.

6.2. Informative References

- [8] 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.

- [9] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction and Applicability Statements for Internet-Standard Management Framework", RFC 3410, December 2002.
- [10] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2863, June 2000.
- [11] Thaler, D., "IP Tunnel MIB", RFC 2667, August 1999.
- [12] Jamoussi, B., 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.

7. Security Considerations

This MIB module relates to the configuration and management of Traffic Engineering tunnels. The unauthorized manipulation of fields in the tables `teAdminGroupTable`, `teTunnelTable`, `tePathTable`, and `tePathHopTable` may lead to tunnel flapping, tunnel paths being changed, or traffic being disrupted. In addition, if these tables are read by unauthorized parties, the information can be used to trace traffic patterns, traffic volumes, and tunnel paths. This may be considered proprietary and confidential information by some providers.

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:

`teAdminGroupTable`: Changing this will affect the semantics of include and exclude constraints, and thus traffic takes unintended routes.

`teTunnelTable`: Changing this affects many properties of traffic tunnels.

`tePathTable`: Changing this affects the constraints (including bandwidth) of tunnel paths, as well as the status of the path.

`tePathHopTable`: Changing this affects the route followed by a traffic tunnel path.

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:

teTunnelTable: Describes tunnel endpoints and traffic volumes.
tePathTable: Describes path properties.
tePathHopTable: Describes path routes.

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 [9], section 8), including full support for the SNMPv3 cryptographic mechanisms (for authentication and privacy).

Further, deployment of SNMP versions prior to SNMPv3 is NOT RECOMMENDED. Instead, it is RECOMMENDED to deploy SNMPv3 and to enable cryptographic security. It is then a customer/operator responsibility to ensure that the SNMP entity giving access to an instance of this MIB module is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

Acknowledgments

It was Tony Li's suggestion that the author embark on this MIB. Many thanks to him and to Der-Hwa Gan for their input and help.

Many thanks, too, to Bert Wijnen for his incredible help, both with improving the correctness, structure, and readability of the MIB module, and with the text of the RFC. Thanks also to Adrian Farrel for his detailed review.

Author's Address

Kireeti Kompella
Juniper Networks, Inc.
1194 N. Mathilda Ave
Sunnyvale, CA 94089

EMail: kireeti@juniper.net

Full Copyright Statement

Copyright (C) The Internet Society (2005).

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 IETF's procedures with respect to rights in IETF 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.

