

Internet Draft
Expires: August 2002

David Zelig
Sharon Mantin
Corrigent Systems

Thomas D. Nadeau
Cisco Systems, Inc.

Dave Danenberg
Litchfield Communications, Inc.

A. Malis
Vivace Networks, Inc.
February 2002

Pseudo Wire (PW) over MPLS PSN Management Information Base

[draft-zelig-pw-mpls-mib-01.txt](#)

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of [Section 10 of RFC2026](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at <http://www.ietf.org/ietf/1id-abstracts.txt>

The list of Internet-Draft Shadow Directories can be accessed at <http://www.ietf.org/shadow.html>.

Copyright (C) The Internet Society (2001). All rights reserved.

1 Abstract

This memo defines an experimental portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. In particular, it describes MIB module for PW operation over Multi-Protocol Label Switching (MPLS) Label Switch Router (LSR).

Table of Contents

1	Abstract.....	1
2	Introduction.....	3
3	Terminology.....	4
4	The SNMP Management Framework.....	5
4.1	Object Definitions.....	6
5	Feature Checklist.....	6
6	MIB usage.....	7
6.1	PW-MPLS-MIB usage.....	7
6.2	Example of MIB usage.....	9
7	Object definitions.....	11
8	Security Considerations.....	29
9	References.....	30
10	Author's Addresses.....	32
11	Full Copyright Statement.....	33

2 Introduction

This document describes a model for managing pseudo wire services for transmission over different flavors of MPLS tunnels. The general PW MIB [[PW-MIB](#)] defines the parameters global to the VC regardless of underlying PSN and emulated service. Indicating PSN type of MPLS in PW-MIB references this module.

This document describes the MIB objects that define pseudo wire association to the MPLS PSN, in a way that is not specific to the carried service.

Together, [TEMIB and LSRMIB], describe the modeling of an MPLS Tunnel, and a Tunnel's underlying cross-connects. The defined MIB support MPLS-TE PSN, MPLS LSR PSN (an outer tunnel created by LDP or manually), and MPLS VC only (no outer tunnel).

Some flavors of MPLS, such as carrying PW in MPLS in IP and MPLS in GRE are not defined here, waiting to the WG documents that will describe the exact MPLS functionality. It still for further study whether this functionality will be treated in the same MIB modules as the other types of MPLS or not.

Conventions used in this document

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](#) [[BCP14](#)].

3 Terminology

This document uses terminology from the document describing the PW framework [[FRMWK](#)], from [[ENCAP](#)] and [[TRANS](#)].

"Adaptation" refers to the method of adapting a "foreign" communications protocol such that it can be carried by a packet switched net (the PSN). For example, in a CEP service the foreign protocol is SONET/SDH and the PSN is MPLS.

The terms "Outbound" and "Inbound" in this MIB module are based on the common practice in the MPLS standards, i.e. "outbound" are toward the PSN. However, where these terms are used in an object name, the object description clarifies the exact packet direction to prevent confusion with these terms in other documents.

"PSN Tunnel" is a general term indicating a virtual connection between the two PW edge devices. Each tunnel may potentially carry multiple VCs inside. In the scope of this document, it is MPLS tunnel.

"Maintenance protocol" is the protocol used to signal the PW VC labels and other parameters related to the PW establishment.

Since a PW service is bi-directional, PW services require two uni-directional tunnels in the case of MPLS.

PW will normally transmit into an originating "head" end of a PSN Tunnel, and receive from a terminating "tail" end of a Tunnel. While the transmit tunnel needs to be configured at the edge device, it is not always known a-priori which tunnel will be the inbound tunnel for specific service. This knowledge depends on the maintenance protocol used for PW set-up.

This document uses terminology from the document describing the MPLS architecture [[MPLSArch](#)] for MPLS PSN. A Label Switched Path (LSP) is modeled as described in [LSRMIB and TEMIB] via a series of cross-connects through 1 or more Label switch routers (LSR).

In MPLS PSN, a PW connection typically uses a VC (Virtual Connection) Label within a Tunnel Label [[TRANS](#)]. Multiple PW VCs each with a unique VC Label can share the same Tunnel. For PW transport over MPLS, the Tunnel Label is known as the "outer" Label, while the VC Label is known as the "inner" Label. An exception to this is with adjacent LSRs or the use of PHP. In this case, there is an option for PW VCs to connect directly without an outer Label.

VC level protection is for further study. Protection is currently assumed at the outer tunnel level only, on bulk of VCs. Future

revision of this document will control the behavior of such protection in more details.

4 The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- o An overall architecture, described in [RFC 2571](#) [[RFC2571](#)].
- o Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIV1 and described in STD 16, [RFC 1155](#) [[RFC1155](#)], STD 16, [RFC 1212](#) [[RFC1212](#)] and [RFC 1215](#) [[RFC1215](#)]. The second version, called SMIV2, is described in STD 58, [RFC 2578](#) [[RFC2578](#)], STD 58, [RFC 2579](#) [[RFC2579](#)] and STD 58, [RFC 2580](#) [[RFC2580](#)].
- o Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in STD 15, [RFC 1157](#) [[RFC1157](#)]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in [RFC 1901](#) [[RFC1901](#)] and [RFC 1906](#) [[RFC1906](#)]. The third version of the message protocol is called SNMPv3 and described in [RFC 1906](#) [[RFC1906](#)], [RFC 2572](#) [[RFC2572](#)] and [RFC 2574](#) [[RFC2574](#)].
- o Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in STD 15, [RFC 1157](#) [[RFC1157](#)]. A second set of protocol operations and associated PDU formats is described in [RFC 1905](#) [[RFC1905](#)].
- o A set of fundamental applications described in [RFC 2573](#) [[RFC2573](#)] and the view-based access control mechanism described in [RFC 2575](#) [[RFC2575](#)].

A more detailed introduction to the current SNMP Management Framework can be found in [RFC 2570](#) [[RFC2570](#)].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIV2. A MIB conforming to the SMIV1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine readable information in SMIV2 will be converted into textual descriptions in

SMIv1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

4.1 Object Definitions

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the subset of Abstract Syntax Notation One (ASN.1) defined in the SMI. In particular, an OBJECT IDENTIFIER, an administratively assigned name, names each object type. The object type together with an object instance serves to uniquely identify a specific instantiation of the object. For human convenience, we often use a textual string, termed the descriptor, to also refer to the object type.

5 Feature Checklist

The PW MPLS MIB (PW-MPLS-MIB) is designed to satisfy the following requirements and constraints:

- The MIB supports both manually configured and signaled VCs.
- The MIB supports point-to-point PW connections.
- The MIB enables the use of any emulated service.
- The MIB supports MPLS-TE outer tunnel, MPLS LSR outer tunnel (an outer tunnel signaled by LDP or set-up manually), and no outer tunnel (where the VC label is the only label in the incoming MPLS stack).
- The MIB enables both strict and loose incoming VC lookup. In strict mode, only VC carried inside explicitly configured or signaled tunnels are accepted.

6 MIB usage

The MIB structure for defining a PW service is composed from three types of modules.

The first type is the PW-MIB module [[PW-MIB](#)], which configures general parameters of the VC that are common to all types of emulated services and PSNs.

The second type of modules is per PSN module. There is a different module for each type of PSN. This document defines the MIB module for MPLS (PW-MPLS-MIB).

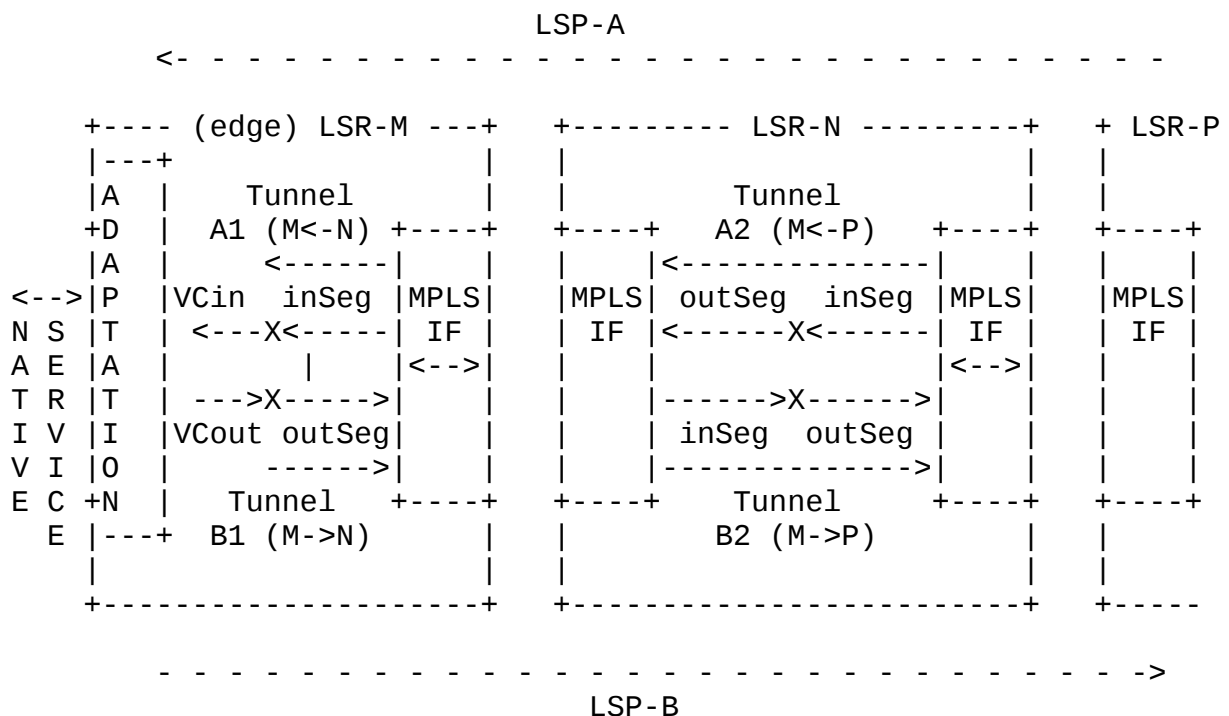
The third type of modules is service-specific module, which is emulated signal type dependent. These modules are defined in other documents; see for example [[CEPMIB](#)].

[PWTC] defines some of the object types used in these modules.

[6.1](#) PW-MPLS-MIB usage

- The VC table (pwVcTable) in [[PW-MIB](#)] is used for all VC types (ATM, FR, Ethernet, SONET, etc.). This table contains high level generic parameters related to the VC creation. A row is created by the operator for each PW service.
- If the PSN type in pwVcTable is MPLS, the agent create a row in the MPLS specific parameters table (pwMplsVcTable) in PW-MPLS-MIB, which contain MPLS specific parameters such as EXP bits handling and outer tunnel selection.
- A row is created (either by an operator or manually) in MPLS outbound tunnels table (pwVcMplsOutboundTable), which associates the VC to one or more (in a case of backup tunnels) MPLS tunnels. This table simply associates the VC with the entries at the relevant MPLS MIBs.
- The MPLS inbound tunnel table (pwVcMplsInboundTable) associates the VC to the incoming tunnel. This table is optional, as some maintenance protocols for VC setup do not include the association data. This table may be also used for restricting the packet reception for a specific PW from pre-defined tunnels, bringing better security and better miss-configuration error immunity.
- The MPLS tunnels mapping table (pwMplsMappingTable) associates the tunnel and the VC label to the VC index. This table is used for easy lookup process when searching VC information.

The relation to the MPLS network is by configuration of the edge LSR only - that is to say, the LSR providing the PW function. Since Tunnels are uni-directional, a pair of tunnels must exist (one for inbound, one for outbound). The following graphic depicts a VC that originates and terminates at LSR-M. It uses LSPs A and B formed by Tunnels Ax and Bx continuing through LSR-N to LSR-P. The concatenations of Tunnels create the LSPs. Note: 'X' denotes a Tunnel's cross-connect.



The PW-MPLS-MIB supports three options for MPLS network:

- In the MPLS-TE case, Tunnel A1 and B1 are created via the MPLS-TE MIB [[TEMIB](#)]. The tunnels are associated to the VC by the (4) indexes that uniquely identify the Tunnel at the TE-MIB.
- In the MPLS-LSP case, Tunnel A1 and B1 are either manually configured or set up with LDP. The tunnels are associated to the VC by the XC index in the MPLS-LSR MIB [[LSRMIB](#)], that uniquely identify the Tunnel at the LSR-MIB.
- In the VC only case, there is no outer Tunnel on top of the VC label. This case is useful in case of adjacent PE (see [[TRANS](#)]) or when LSR-N acts as PHP for the outer tunnel label. In this case, Association is done directly to the physical interfaces in the PW-MPLS-MIB tables.

Note that for some maintenance protocols used to set-up the VC it is not always possible to know the association between the VC and

the Tunnel at the inbound side (Tunnel A1). For MPLS PSN for example, it is not always possible to know the association between the VC and its inbound LSP (inSeg cross-connect).

A combination of MPLS-TE outer tunnel(s) and LDP outer tunnel for the same VC is allowed by creating the rows with the same VcIndex with different MPLS tunnel indexes types.

[6.2](#) Example of MIB usage

In this section we provide an example of using the MIB objects for setting up a VC over MPLS. While this example is not meant to illustrate every permutation of the MIB, it is intended as an aid to understanding some of the key concepts. It is meant to be read after going through the MIB itself.

In this example a PW service for CEP is configured over an MPLS-TE tunnel. It uses LDP as in [\[TRANS\]](#) for service set-up.

In PW-MIB:

In pwVcTable:

```
{
    pwVcIndex                5,

    pwVcType                  cep,
    pwVcOwner                  maintenanceProtocol,
    pwVcPsntype                mpls,
    pwVcPriority                0,
    pwVcinboundMode            loose,

    pwVcPeerAddrType           ipv4(2),
    pwVcPeerAddr               1.4.3.2, -- In this case equal to the
                                   -- peer tunnel IP address

    pwVcID                     10,
    pwVcLocalGroupID           12,

    pwVcControlWord            false, -- Control word not to be sent
    pwVcLocalIfMtu              0,     -- Do not send ifMtu parameter
    pwVcLocalIfString           false, -- Do not signal if string

    pwVcRemoteGroupID          0xFFFF, -- Will be received by
                                   -- maintenance protocol
    pwVcRemoteControlWord       notKnownYet,
    pwVcRemoteIfMtu             0,
    pwVcRemoteIfString          ""
```

```

pwVcOutboundVcLabel      0xFFFF, -- Will be received by
                           -- maintenance protocol
pwVcInboundVcLabel       0xFFFF, -- Will be set by signaling

pwVcName                  "Example of CEP VC",
pwVcDescr                  "",
..

pwVcAdminStatus           up,
..

```

The operator now create a row in pwVcMplsTable based on the VcIndex, that is configured with MPLS specific values:

In pwVcMplsTable:

```

{
    pwVcMplsMplsType        mplsTe,
    pwVcMplsExpBitsMode     outerTunnel,
    pwVcMplsExpBits         0,
    pwVcMplsTtl             2,
    pwVcMplsLocalLdpID      1.2.3.4.0.0 -- Global label space
    pwVcMplsLocalLdpEntityID 0,
    pwVcMplsPeerLdpID       0, -- Not known yet
    ...
}

```

The operator now associates the VC with an outgoing TE tunnel:

In pwVcMplsOutboundTable:

```

{
    pwVcMplsOutboundIndex      0, -- The first row
                                --for this VcIndex.
    pwVcMplsOutBoundLsrXcIndex 0, -- MPLS-TE
    pwVcMplsOutboundTunnelIndex 500,
    pwVcMplsOutboundTunnelInstance 0,
    pwVcMplsOutboundTunnelLclLSR 1.2.3.4, -- Always
                                -- the LSR ID of the current node.
    pwVcMplsOutboundTunnelPeerLSR 1.4.3.2
    pwVcMplsOutboundIfIndex     0, -- MPLS-TE
    ..
}

```

pwVcMplsInboundTable is not used because loose LDP set-up is used.

pwVcMplsMappingTable entry will be created by the agent once the LDP maintenance session will be finished and will enable easy lookup for the VcIndex from knowledge of VC label or the tunnel.

7 Object definitions

```
PW-MPLS-MIB DEFINITIONS ::= BEGIN
```

```
IMPORTS
```

```
    MODULE-IDENTITY, OBJECT-TYPE,  
    experimental, Unsigned32  
    FROM SNMPv2-SMI
```

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

```
    RowStatus, StorageType  
    FROM SNMPv2-TC
```

```
    InterfaceIndexOrZero  
    FROM IF-MIB
```

```
    MplsLabel, MplsTunnelIndex, MplsTunnelInstanceIndex,  
    MplsLdpIdentifier  
    FROM MPLS-TC-MIB
```

```
    PwVcIndexType  
    FROM PW-TC-MIB
```

```
    pwVcIndex  
    FROM PW-MIB
```

```
;
```

```
pwVcMplsMIB MODULE-IDENTITY
```

```
    LAST-UPDATED "200201291200Z" -- 29 January 2002 12:00:00 EST  
    ORGANIZATION "Pseudo Wire Edge to Edge Emulation (PWE3) Working  
    Group"
```

```
    CONTACT-INFO
```

```
        "  
        David Zelig  
        Postal: Corrigent Systems  
        126, Yigal Alon St.  
        Tel Aviv, ISRAEL  
        Phone: +972-3-6945273  
        E-mail: davidz@corrigent.com
```

```
        Thomas D. Nadeau
```

Postal: Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
Tel: +1-978-497-3051
Email: tnadeau@cisco.com

Dave Danenberg
Postal: Litchfield Communications, Inc.
76 Westbury Park Rd
Princeton Building East
Watertown, CT 06795
Tel: +1-860-945-1573 x3180
Email: dave_danenberg@litchfieldcomm.com

Sharon Mantin
Postal: Corrigent Systems
126, Yigal Alon St.
Tel Aviv, ISRAEL
Phone: +972-3-6948608
E-mail: sharonm@corrigent.com

The PWE3 Working Group (email distribution pwe3@ietf.org) "

DESCRIPTION

"This MIB complements the PW-MIB for PW operation over MPLS.

This MIB is dependant on the MIBs as defined by Nadeau,
T., et al, <[draft-ietf-mpls-lsr-mib.txt](#)>,
<[draft-ietf-mpls-te-mib.txt](#)>,
and <[draft-ietf-mpls-tc-mib.txt](#)>."

-- Revision history.

REVISION

"200201291200Z" -- 29 January 2002 12:00:00 EST

DESCRIPTION

"Changes from previous version:

- 1) Add LDP entity association.
- 2) Clarify inbound/outbound directions.
- 3) Simplify indexing of outbound and inbound tables
and providing get next variables. "

REVISION

"200107111200Z" -- 7 November 2001 12:00:00 EST

DESCRIPTION

"Changes from previous version:

- 1) Remove Vc instance from table indexing.
- 2) Update descriptions of indexing and protection.
- 3) Remove the need for MPLS-LSR in case of VC only.
- 4) Change pwVcMplsMplsType to BITS in order to enable
multiple types of outer tunnel.
- 5) Add ifindex to outer tunnel tables to support vcOnly

```
option.
6) change naming of outbound, inbound and mapping tables to
   reflect addition of VC only port ifindexes.
7) Adapt order of items in mapping table to SNMP convention.
"
REVISION
  "200107111200Z"  -- July 11 2001 12:00:00 EST
DESCRIPTION
  "draft-zelig-pw-mib-00.txt - initial version"

 ::= { experimental xxx }

-- Top-level components of this MIB.

-- Traps
pwVcMplsNotifications OBJECT IDENTIFIER
                        ::= { pwVcMplsMIB 0 }
pwVcMplsNotifyPrefix  OBJECT IDENTIFIER
                        ::= { pwVcMplsNotifications 0 }

-- Tables, Scalars
pwVcMplsObjects       OBJECT IDENTIFIER
                        ::= { pwVcMplsMIB 1 }

-- Conformance
pwVcMplsConformance   OBJECT IDENTIFIER
                        ::= { pwVcMplsMIB 2 }

-- PW VC MPLS table

pwVcMplsTable         OBJECT-TYPE
    SYNTAX             SEQUENCE OF PwVcMplsEntry
    MAX-ACCESS          not-accessible
    STATUS              current
    DESCRIPTION
        "This table specifies information for VC to be carried over
        MPLS PSN."
    ::= { pwVcMplsObjects 1 }

pwVcMplsEntry         OBJECT-TYPE
    SYNTAX             PwVcMplsEntry
    MAX-ACCESS          not-accessible
    STATUS              current
    DESCRIPTION
        "A row in this table represents parameters specific to MPLS
        PSN for a pseudo wire connection (VC). The row is created
        automatically by the local agent if the pwVcPsnType is
        MPLS. It is indexed by pwVcIndex, which uniquely
        identifying a singular connection.
        "

INDEX { pwVcIndex }
```

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

```
 ::= { pwVcMplsTable 1 }
```

```
PwVcMplsEntry ::= SEQUENCE {
    pwVcMplsMplsType          BITS,
    pwVcMplsExpBitsMode       INTEGER,
    pwVcMplsExpBits           Unsigned32,
    pwVcMplsTtl               Unsigned32,
    pwVcMplsLocalLdpID        MplsLdpIdentifier,
    pwVcMplsLocalLdpEntityID  Unsigned32,
    pwVcMplsPeerLdpID         MplsLdpIdentifier,
    pwVcMplsStorageType       StorageType
}
```

pwVcMplsMplsType OBJECT-TYPE

```
SYNTAX  BITS {
    mplsTe      (0),
    mplsLsp     (1),
    vcOnly      (2)
}
```

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Set by the operator to indicate the outer tunnel types, if exists. mplsTe is used if the outer tunnel was set-up by MPLS-TE, and mplsLsp is used the outer tunnel was set up by LDP or manually. Combination of mplsTe and mplsLsp may exist in case of outer tunnel protection. vcOnly is used if there is no outer tunnel label. vcOnly cannot be combined with mplsLsp or mplsTe."

```
 ::= { pwVcMplsEntry 1 }
```

pwVcMplsExpBitsMode OBJECT-TYPE

```
SYNTAX  INTEGER {
    outerTunnel      (1),
    specifiedValue   (2),
    serviceDependant (3)
}
```

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Set by the operator to indicate the way the VC shim label EXP bits are to be determined. The value of outerTunnel(1) is used where there is an outer tunnel (MPLS-TE or MPLS-LSP) - pwVcMplsMplsType is mplsTeOrLsp(1). Note that in this case there is no need to mark the VC label with the

EXP bits since it will may done by the outer tunnel termination node.
If there is no outer tunnel, specifiedValue(2) indicate

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

that the value is specified by pwVcMplsExpBits, and serviceDependant(3) indicate that the EXP bits are setup based on a rule specified in the emulated service specific tables, for example when the EXP bits are a function of 802.1p marking for Ethernet emulated service."

REFERENCE

"martini et al, <[draft-martini-l2circuit-encap-mpls.txt](#)> sections [3.3](#) and [4.3](#)."

DEFVAL { outerTunnel }

::= { pwVcMplsEntry 2 }

pwVcMplsExpBits OBJECT-TYPE

SYNTAX Unsigned32 (0..7)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Set by the operator to indicate the MPLS EXP bits to be used on the VC shim label if pwVcMplsExpBitsMode is specifiedValue(2), zero otherwise."

DEFVAL { 0 }

::= { pwVcMplsEntry 3 }

pwVcMplsTtl OBJECT-TYPE

SYNTAX Unsigned32 (0..255)

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"Set by the operator to indicate the VC TTL bits to be used on the VC shim label."

REFERENCE

"martini et al, <[draft-martini-l2circuit-encap-mpls](#)> "

DEFVAL { 2 }

::= { pwVcMplsEntry 4 }

pwVcMplsLocalLdpID OBJECT-TYPE

SYNTAX MplsLdpIdentifier

MAX-ACCESS read-write

STATUS current

DESCRIPTION

"The local LDP identifier of the LDP entity creating this VC in the local node. As the VC labels are always set from a global label space, the last two octets in the LDP ID MUST be always both zeros."

REFERENCE

```

    "<draft-ietf-ldp-mib>,
    "<draft-martini-l2circuit-encap-mpls>."
    ::= { pwVcMplsEntry 5 }

```

pwVcMplsLocalLdpEntityID OBJECT-TYPE

Zelig et al

Expires August 2002

[page 15]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

```

SYNTAX      Unsigned32
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
    "The local LDP Entity index of the LDP entity to be used
    for this VC on the local node. Should be set to all zeros
    if not used."
REFERENCE
    "<draft-ietf-ldp-mib>."
    ::= { pwVcMplsEntry 6 }

pwVcMplsPeerLdpID OBJECT-TYPE
SYNTAX      MplsLdpIdentifier
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION
    "The peer LDP identifier as identified from the LDP
    session. Should be zero if not relevant or not known yet."
REFERENCE
    "<draft-ietf-ldp-mib>,
    "<draft-martini-l2circuit-encap-mpls>."
    ::= { pwVcMplsEntry 7 }

pwVcMplsStorageType OBJECT-TYPE
SYNTAX      StorageType
MAX-ACCESS  read-write
STATUS      current
DESCRIPTION
    "This variable indicates the storage type for this object."
    ::= { pwVcMplsEntry 8 }

-- End of PW MPLS VC table

-- Pseudo Wire VC MPLS Outbound Tunnel table

pwVcMplsOutboundIndexNext OBJECT-TYPE
SYNTAX      Unsigned32 (0..4294967295)
MAX-ACCESS  read-only
STATUS      current
DESCRIPTION

```

"This object contains an appropriate value to be used for pwVcMplsOutboundIndex when creating entries in the pwVcMplsOutboundTable. The value 0 indicates that no unassigned entries are available. To obtain the pwVcMplsOutboundIndex value for a new entry, the manager issues a management protocol retrieval operation to obtain the current value of this object. After each retrieval, the agent should modify the value to

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

the next unassigned index, however the agent MUST NOT assume such a retrieval will be done for each row created."

::= { pwVcMplsObjects 2 }

pwVcMplsOutboundTable OBJECT-TYPE

SYNTAX SEQUENCE OF PwVcMplsOutboundEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table associates VCs using MPLS PSN with the outbound MPLS tunnels (i.e. toward the PSN) or the physical interface in case of VC only."

::= { pwVcMplsObjects 3 }

pwVcMplsOutboundEntry OBJECT-TYPE

SYNTAX PwVcMplsOutboundEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A row in this table represents a link between PW VC (that require MPLS tunnels) and MPLS tunnel toward the PSN.

In the case of VC only, it associate the VC with the interface that shall carry the VC.

This table is indexed by the pwVcIndex and an additional index enabling multiple rows for the same VC index.

At least one entry is created in this table by the operator for each PW VC that requires MPLS PSN. Note that the first entry for each VC can be indexed by pwVcMplsOutboundIndex equal zero without a need for retrieval of pwVcMplsOutboundIndexNext.

This table points to the appropriate MPLS MIB. In the case of MPLS-TE, the 4 variables relevant to the indexing of a TE MPLS tunnel are set as in Srinivasan, et al, <[draft-ietf-mpls-te-mib](#)>.

In case of MPLS LSP (an outer tunnel label assigned by LDP or manually) the table points to the XC entry in the LSR

MIB as in Srinivasan, et al, <[draft-ietf-mpls-lsr-mib](#)>. In case of VC only (no outer tunnel) the ifindex of the port to carry the VC is configured.

Each VC may have multiple rows in this tables if protection is available at the outer tunnel level, each row may be of different type except for VC only, on which only rows with ifIndex of the port are allowed.

INDEX { pwVcIndex, pwVcMplsOutboundIndex }

Zelig et al

Expires August 2002

[page 17]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

::= { pwVcMplsOutboundTable 1 }

PwVcMplsOutboundEntry ::= SEQUENCE {
 pwVcMplsOutboundIndex Unsigned32,
 pwVcMplsOutboundLsrXcIndex Unsigned32,
 pwVcMplsOutboundTunnelIndex MplsTunnelIndex,
 pwVcMplsOutboundTunnelInstance MplsTunnelInstanceIndex,
 pwVcMplsOutboundTunnelLclLSR Unsigned32,
 pwVcMplsOutboundTunnelPeerLSR Unsigned32,
 pwVcMplsOutboundIfIndex InterfaceIndexOrZero,
 pwVcMplsOutboundRowStatus RowStatus,
 pwVcMplsOutboundStorageType StorageType
}

pwVcMplsOutboundIndex OBJECT-TYPE

SYNTAX Unsigned32 (0..4294967295)

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"Arbitrary index for enabling multiple rows per VC in this table. Next available free index can be retrieved using pwVcMplsOutboundIndexNext.

"

::= { pwVcMplsOutboundEntry 1 }

pwVcMplsOutboundLsrXcIndex OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-create

STATUS current

DESCRIPTION

"This object will be set by the operator. If the outer label is defined in the MPL-LSR MIB, i.e. set by LDP or manually, this object points to the XC index of the outer tunnel. Otherwise, it is set to zero."

::= { pwVcMplsOutboundEntry 2 }

pwVcMplsOutboundTunnelIndex	OBJECT-TYPE
SYNTAX	MplsTunnelIndex
MAX-ACCESS	read-create
STATUS	current
DESCRIPTION	
"Part of set of indexes for outbound tunnel in the case of MPLS-TE outer tunnel, otherwise set to zero."	
::= { pwVcMplsOutboundEntry 3 }	

pwVcMplsOutboundTunnelInstance	OBJECT-TYPE
SYNTAX	MplsTunnelInstanceIndex
MAX-ACCESS	read-create
STATUS	current
DESCRIPTION	

Zelig et al

Expires August 2002

[page 18]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

"Part of set of indexes for outbound tunnel in the case of MPLS-TE outer tunnel, otherwise set to zero."

::= { pwVcMplsOutboundEntry 4 }

pwVcMplsOutboundTunnelLc1LSR	OBJECT-TYPE
SYNTAX	Unsigned32
MAX-ACCESS	read-create
STATUS	current
DESCRIPTION	
"Part of set of indexes for outbound tunnel in the case of MPLS-TE outer tunnel, otherwise set to zero."	
::= { pwVcMplsOutboundEntry 5 }	

pwVcMplsOutboundTunnelPeerLSR	OBJECT-TYPE
SYNTAX	Unsigned32
MAX-ACCESS	read-create
STATUS	current
DESCRIPTION	
"Part of set of indexes for outbound tunnel in the case of MPLS-TE outer tunnel, otherwise set to zero."	
::= { pwVcMplsOutboundEntry 6 }	

pwVcMplsOutboundIfIndex	OBJECT-TYPE
SYNTAX	InterfaceIndexOrZero
MAX-ACCESS	read-create
STATUS	current
DESCRIPTION	
"In case of VC only (no outer tunnel), this object holds the ifIndex of the outbound port, otherwise set to zero."	
::= { pwVcMplsOutboundEntry 7 }	

pwVcMplsOutboundRowStatus	OBJECT-TYPE
SYNTAX	RowStatus

MAX-ACCESS read-create
 STATUS current
 DESCRIPTION
 "For creating, modifying, and deleting this row."
 ::= { pwVcMplsOutboundEntry 8 }

pwVcMplsOutboundStorageType OBJECT-TYPE
 SYNTAX StorageType
 MAX-ACCESS read-create
 STATUS current
 DESCRIPTION
 "This variable indicates the storage type for this object."
 ::= { pwVcMplsOutboundEntry 9 }

-- End of Pseudo Wire VC MPLS Outbound Tunnel table

-- Pseudo Wire VC MPLS Inbound Tunnel table

Zelig et al Expires August 2002 [page 19]

Pseudo Wire(PW) Over MPLS Management Information Base

February 2002

pwVcMplsInboundIndexNext OBJECT-TYPE
 SYNTAX Unsigned32 (0..4294967295)
 MAX-ACCESS read-only
 STATUS current
 DESCRIPTION
 "This object contains an appropriate value to
 be used for pwVcMplsInboundIndex when creating
 entries in the pwVcMplsInboundTable. The value
 0 indicates that no unassigned entries are
 available. To obtain the pwVcMplsInboundIndex
 value for a new entry, the manager issues a
 management protocol retrieval operation to obtain
 the current value of this object. After each
 retrieval, the agent should modify the value to
 the next unassigned index, however the agent MUST
 NOT assume such a retrieval will be done for each
 row created."
 ::= { pwVcMplsObjects 4 }

pwVcMplsInboundTable OBJECT-TYPE
 SYNTAX SEQUENCE OF PwVcMplsInboundEntry
 MAX-ACCESS not-accessible
 STATUS current
 DESCRIPTION
 "This table associates VCs using MPLS PSN with the inbound
 MPLS tunnels (i.e. for packets coming from the PSN),
 for maintenance protocols that support association of the
 VC with the inbound MPLS tunnel or where desired for
 security reasons."

::= { pwVcMplsObjects 5 }

pwVcMplsInboundEntry OBJECT-TYPE

SYNTAX PwVcMplsInboundEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A row in this table represents a link between PW VCs (that require MPLS tunnels) and MPLS tunnel for packets arriving from the PSN.

This table is indexed by the set of indexes used to identify the VC - pwVcIndex and an additional index enabling multiple rows for the same VC index.

Note that the first entry for each VC can be indexed by pwVcMplsOutboundIndex equal zero without a need for retrieval of pwVcMplsInboundIndexNext.

An entry is created in this table either automatically by the local agent for each VC that was created by a maintenance protocol that enable such association, or

Zelig et al

Expires August 2002

[page 20]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

created manually by the operator in cases that strict mode is required.

Note that the control messages contain VC ID and VC type, which together with the remote IP address identify the pwVcIndex in the local node.

This table points to the appropriate MPLS MIB. In the case of MPLS-TE, the 4 variables relevant to the indexing of a TE MPLS tunnel are set as in Srinivasan, et al, <[draft-ietf-mpls-te-mib-06.txt](#)>.

In case of MPLS LSP (an outer tunnel label assigned by LDP or manually) the table points to the XC entry in the LSR MIB as in Srinivasan, et al, <[draft-ietf-mpls-lsr-mib-07.txt](#)>.

Each VC may have multiple rows in this tables if protection is available at the outer tunnel level, each row may be of different type except for VC only, on which only rows with ifindex of the port are allowed.

"

INDEX { pwVcIndex, pwVcMplsInboundIndex }

::= { pwVcMplsInboundTable 1 }

```

PwVcMplsInboundEntry ::= SEQUENCE {
    pwVcMplsInboundIndex          Unsigned32,
    pwVcMplsInboundLsrXcIndex     Unsigned32,
    pwVcMplsInboundTunnelIndex    MplsTunnelIndex,
    pwVcMplsInboundTunnelInstance MplsTunnelInstanceIndex,
    pwVcMplsInboundTunnelLclLSR   Unsigned32,
    pwVcMplsInboundTunnelPeerLSR  Unsigned32,
    pwVcMplsInboundIfIndex        InterfaceIndexOrZero,
    pwVcMplsInboundRowStatus      RowStatus,
    pwVcMplsInboundStorageType    StorageType
}

```

```

pwVcMplsInboundIndex OBJECT-TYPE
    SYNTAX      Unsigned32 (0..4294967295)
    MAX-ACCESS  not-accessible
    STATUS      current
    DESCRIPTION
        "Arbitrary index for enabling multiple rows per VC in
         this table. Next available free index can be retrieved
         using pwVcMplsInboundIndexNext.
        "
    ::= { pwVcMplsInboundEntry 1 }

```

```

pwVcMplsInboundLsrXcIndex OBJECT-TYPE

```

Zelig et al

Expires August 2002

[page 21]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

```

SYNTAX      Unsigned32
MAX-ACCESS  read-create
STATUS      current
DESCRIPTION
    "If the outer label is defined in the MPL-LSR MIB, i.e. set
     by LDP or manually, this object points to the XC index
     of the outer tunnel. Otherwise, it is set to zero."
    ::= { pwVcMplsInboundEntry 2 }

```

```

pwVcMplsInboundTunnelIndex          OBJECT-TYPE
    SYNTAX      MplsTunnelIndex
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION
        "Part of set of indexes for outbound tunnel in the case of
         MPLS-TE outer tunnel, otherwise set to zero."
    ::= { pwVcMplsInboundEntry 3 }

```

```

pwVcMplsInboundTunnelInstance      OBJECT-TYPE
    SYNTAX      MplsTunnelInstanceIndex
    MAX-ACCESS  read-create
    STATUS      current
    DESCRIPTION

```

"Part of set of indexes for outbound tunnel in the case of
MPLS-TE outer tunnel, otherwise set to zero."
::= { pwVcMplsInboundEntry 4 }

pwVcMplsInboundTunnelLclLSR OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Part of set of indexes for outbound tunnel in the case of
MPLS-TE outer tunnel, otherwise set to zero."
::= { pwVcMplsInboundEntry 5 }

pwVcMplsInboundTunnelPeerLSR OBJECT-TYPE
SYNTAX Unsigned32
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"Part of set of indexes for outbound tunnel in the case of
MPLS-TE outer tunnel, otherwise set to zero."
::= { pwVcMplsInboundEntry 6 }

pwVcMplsInboundIfIndex OBJECT-TYPE
SYNTAX InterfaceIndexOrZero
MAX-ACCESS read-create
STATUS current
DESCRIPTION

Zelig et al

Expires August 2002

[page 22]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

"In case of VC only (no outer tunnel), this object holds the
ifIndex of the inbound port, otherwise set to zero."
::= { pwVcMplsInboundEntry 7 }

pwVcMplsInboundRowStatus OBJECT-TYPE
SYNTAX RowStatus
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"For creating, modifying, and deleting this row."
::= { pwVcMplsInboundEntry 8 }

pwVcMplsInboundStorageType OBJECT-TYPE
SYNTAX StorageType
MAX-ACCESS read-create
STATUS current
DESCRIPTION
"This variable indicates the storage type for this object."
::= { pwVcMplsInboundEntry 9 }

-- End of Pseudo Wire VC MPLS Inbound Tunnel table

-- VC to MPLS Mapping Table.

pwVcMplsMappingTable OBJECT-TYPE

SYNTAX SEQUENCE OF PwVcMplsMappingEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"This table maps an inbound/outbound Tunnel/VcLabel to a VC."

::= { pwVcMplsObjects 6 }

pwVcMplsMappingEntry OBJECT-TYPE

SYNTAX PwVcMplsMappingEntry

MAX-ACCESS not-accessible

STATUS current

DESCRIPTION

"A row in this table represents the connection between a Tunnel/VcLabel and the VC, or the physical interface and the VC for VC only case. It is indexed by the same indexes that index the tunnel for MPLS TE tunnel, the XC index for MPLS LSP tunnel, or ifindex of the port in VC only case, then adds the VC Label as a 6th index. The same table is used in both inbound and outbound directions, but in a different row for each. Note that for some PW maintenance protocols, the inbound association is not known.

Rows are created by the local agent when all the

Zelig et al

Expires August 2002

[page 23]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

association data is available for display."

INDEX { pwVcMplsMappingXcTunnelIndex,
pwVcMplsMappingTunnelIndex,
pwVcMplsMappingTunnelInstance,
pwVcMplsMappingTunnelPeerLsrID,
pwVcMplsMappingTunnelLocalLsrID,
pwVcMplsMappingTifIndex,
pwVcMplsMappingVcLabel }

::= { pwVcMplsMappingTable 1 }

PwVcMplsMappingEntry ::= SEQUENCE {

pwVcMplsMappingXcTunnelIndex	Unsigned32,
pwVcMplsMappingTunnelIndex	MplsTunnelIndex,
pwVcMplsMappingTunnelInstance	MplsTunnelInstanceIndex,
pwVcMplsMappingTunnelPeerLsrID	Unsigned32,
pwVcMplsMappingTunnelLocalLsrID	Unsigned32,

```

        pwVcMplsMappingTifIndex      InterfaceIndexOrZero,
        pwVcMplsMappingVcLabel       MplsLabel,
        pwVcMplsMappingTunnelDirection INTEGER,
        pwVcMplsMappingVcIndex       PwVcIndexType
    }

```

pwVcMplsMappingXcTunnelIndex OBJECT-TYPE

SYNTAX Unsigned32 (0..4294967295)

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Index for the conceptual XC row identifying Tunnel/VcLabel to VC mappings when the outer tunnel is MPLS-LSP, Zero otherwise."

::= { pwVcMplsMappingEntry 1 }

pwVcMplsMappingTunnelIndex OBJECT-TYPE

SYNTAX MplsTunnelIndex

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Primary index for the conceptual row identifying Tunnel/VcLabel to VC mappings when the outer tunnel is MPLS-TE, Zero otherwise."

::= { pwVcMplsMappingEntry 2 }

pwVcMplsMappingTunnelInstance OBJECT-TYPE

SYNTAX MplsTunnelInstanceIndex

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Uniquely identifies an instance of a mapping when the outer tunnel is MPLS-TE, Zero otherwise"

Zelig et al

Expires August 2002

[page 24]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

::= { pwVcMplsMappingEntry 3 }

pwVcMplsMappingTunnelPeerLsrID OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Uniquely identifies an Peer LSR when the outer tunnel is MPLS-TE, Zero otherwise"

::= { pwVcMplsMappingEntry 4 }

pwVcMplsMappingTunnelLocalLsrID OBJECT-TYPE

SYNTAX Unsigned32

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Uniquely identifies the local LSR, when the outer tunnel is MPLS-TE, Zero otherwise"

::= { pwVcMplsMappingEntry 5 }

pwVcMplsMappingTifIndex OBJECT-TYPE

SYNTAX InterfaceIndexOrZero

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Identify the port on which the VC is carried for VC only case."

::= { pwVcMplsMappingEntry 6 }

pwVcMplsMappingVcLabel OBJECT-TYPE

SYNTAX MplsLabel

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Identifies the VC label on this tunnel"

::= { pwVcMplsMappingEntry 7 }

pwVcMplsMappingTunnelDirection OBJECT-TYPE

SYNTAX INTEGER {

outbound (1),

inbound (2)

}

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Identifies if the row represent an outbound or inbound mapping."

::= { pwVcMplsMappingEntry 8 }

pwVcMplsMappingVcIndex OBJECT-TYPE

SYNTAX PwVcIndexType

Zelig et al

Expires August 2002

[page 25]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

MAX-ACCESS read-only

STATUS current

DESCRIPTION

"Index for entry in the pwVcTable. This value is not the VC ID."

::= { pwVcMplsMappingEntry 9 }

-- End of MPLS Tunnel/VC Mapping Table

-- Notifications - PW over MPLS VCs

-- End of notifications.

-- conformance information

-- Note: Conformance at the object access and values level is
-- still FFS, therefore current conformance is defined at the
-- object existence level only.

pwVcMplsGroups OBJECT IDENTIFIER ::= { pwVcMplsConformance 1 }
pwVcMplsCompliances OBJECT IDENTIFIER ::= { pwVcMplsConformance 2 }

pwMplsModuleCompliance MODULE-COMPLIANCE

STATUS current

DESCRIPTION

"The compliance statement for agent that support PW
over MPLS PSN operation."

MODULE -- this module

MANDATORY-GROUPS { pwVcMplsGroup,
pwVcMplsOutboundGroup,
pwVcMplsMappingGroup
}

GROUP pwVcMplsInboundGroup

DESCRIPTION

"This group is mandatory for those PE that support 1+1
APS at the VC level."

::= { pwVcMplsCompliances 1 }

-- Units of conformance.

pwVcMplsGroup OBJECT-GROUP

OBJECTS {

pwVcMplsMplsType,
pwVcMplsExpBitsMode,
pwVcMplsExpBits,
pwVcMplsTtl,

Zelig et al

Expires August 2002

[page 26]

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

pwVcMplsLocalLdpID,
pwVcMplsLocalLdpEntityID,
pwVcMplsPeerLdpID,
pwVcMplsStorageType
}

STATUS current

DESCRIPTION

"Collection of objects needed for PW VC

```
        over MPLS PSN configuration."
 ::= { pwVcMplsGroups 1 }
```

```
pwVcMplsOutboundGroup OBJECT-GROUP
  OBJECTS {
    pwVcMplsOutboundIndexNext,
    pwVcMplsOutboundLsrXcIndex,
    pwVcMplsOutboundTunnelIndex,
    pwVcMplsOutboundTunnelInstance,
    pwVcMplsOutboundTunnelLclLSR,
    pwVcMplsOutboundTunnelPeerLSR,
    pwVcMplsOutboundIfIndex,
    pwVcMplsOutboundRowStatus,
    pwVcMplsOutboundStorageType
  }

  STATUS current
  DESCRIPTION
    "Collection of objects needed for PW VC
    performance."
  ::= { pwVcMplsGroups 2 }
```

```
pwVcMplsMappingGroup OBJECT-GROUP
  OBJECTS {
    pwVcMplsMappingXcTunnelIndex,
    pwVcMplsMappingTunnelIndex,
    pwVcMplsMappingTunnelInstance,
    pwVcMplsMappingTunnelPeerLsrID,
    pwVcMplsMappingTunnelLocalLsrID,
    pwVcMplsMappingTIfIndex,
    pwVcMplsMappingTunnelDirection,
    pwVcMplsMappingVcLabel,
    pwVcMplsMappingVcIndex
  }

  STATUS current
  DESCRIPTION
    "Collection of objects used for mapping of tunnels and VC
    labels to VC index and instances."
  ::= { pwVcMplsGroups 3 }
```

```
pwVcMplsInboundGroup OBJECT-GROUP
  OBJECTS {
    pwVcMplsInboundIndexNext,
    pwVcMplsInboundLsrXcIndex,
    pwVcMplsInboundTunnelIndex,
```

```

        pwVcMplsInboundTunnelInstance,
        pwVcMplsInboundTunnelLclLSR,
        pwVcMplsInboundTunnelPeerLSR,
        pwVcMplsInboundIfIndex,
        pwVcMplsInboundRowStatus,
        pwVcMplsInboundStorageType
    }

    STATUS    current
    DESCRIPTION
        "Collection of objects needed for inbound association of
        VC and MPLS tunnels. This group is mandatory for PE with
        PW signaling protocols that enable such association or
        in the case of active conservative mode."
    ::= { pwVcMplsGroups 4 }

END -- of PW-MPLS-MIB

```

8 Security Considerations

There are a number of management objects defined in this MIB that

have 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.

No managed objects in this MIB contain sensitive information.

SNMPv1 by itself is not a secure environment. 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.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model [RFC 2574](#) [[RFC2574](#)] and the View-based Access Control Model [RFC 2575](#) [[RFC2575](#)] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, 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.

The use of strict inbound VC lookup the security problems related to a global VC space in a node is greatly reduced, by limiting the accepted packets to a small set of controlled tunnels.

9 References

- [MPLSArch] Rosen, E., Viswanathan, A., and R. Callon, "Multiprotocol Label Switching Architecture", [RFC 3031](#), Jan 2001.
- [CEP] Malis, A., et al, "SONET/SDH Circuit Emulation over Packet (CEP)", Internet Draft <[draft-malis-pwe3-sonet-01](#)>, November 2001.
- [TRANS] Martini et al, "Transport of Layer 2 Frames Over MPLS" <[draft-martini-l2circuit-trans-mpls-08.txt](#)>, November 2001.
- [ENCAP] Martini et al, "Encapsulation Methods for Transport of Layer 2 Frames Over MPLS", <[draft-martini-l2circuit-encap-mpls-04.txt](#)>, November 2001.
- [FRMWK] Pate et al, "Framework for Pseudo Wire Emulation Edge-to-Edge (PWE3)", <[draft-ietf-pwe3-framework-00.txt](#)>, February 2002.
- [PWREQ] Xiao et al, "Requirements for Pseudo Wire Emulation Edge-to-Edge (PWE3)", <[draft-ietf-pwe3-requirements-02.txt](#)>, November 2001.
- [PW-MIB] Zelig et al, "Pseudo Wire (PW) Management Information Base", <[draft-zelig-pw-mib-02](#)>, February 2002.
- [CEPMIB] Danenberg et al, "SONET/SDH Circuit Emulation Service Over MPLS (CEP) Management Information Base Using SMIPv2", <[draft-danenberg-pw-cem-mib-02](#)>, February 2001.
- [LSRMIB] Srinivasan, C., Viswanathan, A., and Nadeau, T. "MPLS Label Switch Router Management Information Base Using SMIPv2", <[draft-ietf-mpls-lsr-mib-08.txt](#)>, January 2002.
- [PWTC] Nadeau, T., et al, "Definitions for Textual Conventions and OBJECT-IDENTITIES for Pseudo-Wires Management", <[draft-Nadeau-pw-tc-mib-02.txt](#)>, February 2002.
- [TEMIB] Srinivasan, C., Viswanathan, A., and Nadeau, T. "MPLS Traffic Engineering Management Information Base Using SMIPv2", <[draft-ietf-mpls-te-mib-08.txt](#)>, January 2002.
- [LDPMIB] J. Cucchiara et al, "Definitions of Managed Objects for the Multiprotocol Label Switching, Label Distribution Protocol (LDP)", <[draft-ietf-mpls-ldp-mib-08.txt](#)>, August 2001.

Zelig et al

Expires August 2002

[page 30]

- [LblStk] Rosen, E., Rekhter, Y., Tappan, D., Farinacci, D., Federokow, G., Li, T., and A. Conta, "MPLS Label Stack Encoding", [RFC 3032](#), January 2001.
- [Assigned] Reynolds, J., and J. Postel, "Assigned Numbers", [RFC 1700](#), October 1994. See also: <http://www.isi.edu/in-notes/iana/assignments/smi-numbers>
- [IANAFamily] Internet Assigned Numbers Authority (IANA), ADDRESS FAMILY NUMBERS, (<http://www.isi.edu/in-notes/iana/assignments/address-family-numbers>), for MIB see: <ftp://ftp.isi.edu/mib/ianaaddressfamilynumbers.mib>
- [IFMIB] McCloghrie, K., and F. Kastenholz, "The Interfaces Group MIB using SMIV2", [RFC 2233](#), Nov. 1997
- [BCP14] Bradner, S., "Key words for use in RFCs to Indicate requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [RFC2571] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", [RFC 2571](#), April 1999.
- [RFC1155] Rose, M., and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, [RFC 1155](#), May 1990.
- [RFC1212] Rose, M., and K. McCloghrie, "Concise MIB Definitions", STD 16, [RFC 1212](#), March 1991.
- [RFC1215] M. Rose, "A Convention for Defining Traps for use with the SNMP", [RFC 1215](#), March 1991.
- [RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Structure of Management Information Version 2 (SMIV2)", STD 58, [RFC 2578](#), April 1999.
- [RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Textual Conventions for SMIV2", STD 58, [RFC 2579](#), April 1999.
- [RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M., and S. Waldbusser, "Conformance Statements for SMIV2", STD 58, [RFC 2580](#), April 1999.
- [RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol", STD 15, [RFC 1157](#), May 1990.

- [RFC1901] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Introduction to Community-based SNMPv2", [RFC 1901](#), January 1996.
- [RFC1906] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1906](#), January 1996.
- [RFC2572] Case, J., Harrington D., Presuhn R., and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", [RFC 2572](#), April 1999.
- [RFC2574] Blumenthal, U., and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", [RFC 2574](#), April 1999.
- [RFC1905] Case, J., McCloghrie, K., Rose, M., and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", [RFC 1905](#), January 1996.
- [RFC2573] Levi, D., Meyer, P., and B. Stewart, "SNMPv3 Applications", [RFC 2573](#), April 1999.
- [RFC2575] Wijnen, B., Presuhn, R., and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", [RFC 2575](#), April 1999.
- [RFC2570] Case, J., Mundy, R., Partain, D., and B. Stewart, "Introduction to Version 3 of the Internet-standard Network Management Framework", [RFC 2570](#), April 1999.

10 Author's Addresses

David Zelig
Corrigent Systems
126, Yigal Alon st.
Tel Aviv, ISRAEL
Phone: +972-3-6945273
Email: davidz@corrigent.com

Thomas D. Nadeau
Cisco Systems, Inc.
250 Apollo Drive
Chelmsford, MA 01824
Email: tnadeau@cisco.com

Dave Danenberg

Pseudo Wire(PW) Over MPLS
Management Information Base

February 2002

Litchfield Communications, Inc.
76 Westbury Park Rd
Princeton Building East
Watertown, CT 06795
Email: dave_danenberglitchfieldcomm.com

Andrew G. Malis
Vivace Networks, Inc.
2730 Orchard Parkway
San Jose, CA 95134
Email: Andy.Malis@vivacenetworks.com

Sharon Mantin
Corrigent Systems
126, Yigal Alon st.
Tel Aviv, ISRAEL
Phone: +972-3-6948608
Email: sharonm@corrigent.com

11 Full Copyright Statement

Copyright (C) The Internet Society (2000). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns. This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS 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.

