

Network Working Group  
Request for Comments: 4447  
Category: Standards Track

L. Martini, Ed.  
E. Rosen  
Cisco Systems, Inc.  
N. El-Aawar  
Level 3 Communications, LLC.  
T. Smith  
Network Appliance, Inc.  
G. Heron  
Tellabs  
April 2006

## **Pseudowire Setup and Maintenance Using the Label Distribution Protocol (LDP)**

### Status of This Memo

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

### Copyright Notice

Copyright (C) The Internet Society (2006).

### Abstract

Layer 2 services (such as Frame Relay, Asynchronous Transfer Mode, and Ethernet) can be "emulated" over an MPLS backbone by encapsulating the Layer 2 Protocol Data Units (PDU) and transmitting them over "pseudowires". It is also possible to use pseudowires to provide low-rate Time Division Multiplexed and a Synchronous Optical NETworking circuit emulation over an MPLS-enabled network. This document specifies a protocol for establishing and maintaining the pseudowires, using extensions to Label Distribution Protocol (LDP). Procedures for encapsulating Layer 2 PDUs are specified in a set of companion documents.

## Table of Contents

<a href="#">1. Introduction</a>	<a href="#">3</a>
<a href="#">2. Specification of Requirements</a>	<a href="#">5</a>
<a href="#">3. The Pseudowire Label</a>	<a href="#">5</a>
<a href="#">4. Details Specific to Particular Emulated Services</a>	<a href="#">7</a>
<a href="#">4.1. IP Layer 2 Transport</a>	<a href="#">7</a>
<a href="#">5. LDP</a>	<a href="#">7</a>
<a href="#">5.1. LDP Extensions</a>	<a href="#">8</a>
<a href="#">5.2. The PWid FEC Element</a>	<a href="#">8</a>
<a href="#">5.3. The Generalized PWid FEC Element</a>	<a href="#">10</a>
<a href="#">5.3.1. Attachment Identifiers</a>	<a href="#">11</a>
<a href="#">5.3.2. Encoding the Generalized ID FEC Element</a>	<a href="#">13</a>
<a href="#">5.3.2.1. Interface Parameters TLV</a>	<a href="#">14</a>
<a href="#">5.3.2.2. PW Grouping TLV</a>	<a href="#">14</a>
<a href="#">5.3.3. Signaling Procedures</a>	<a href="#">15</a>
<a href="#">5.4. Signaling of Pseudowire Status</a>	<a href="#">16</a>
<a href="#">5.4.1. Use of Label Mappings Messages</a>	<a href="#">16</a>
<a href="#">5.4.2. Signaling PW Status</a>	<a href="#">17</a>
<a href="#">5.4.3. Pseudowire Status Negotiation Procedures</a>	<a href="#">18</a>
<a href="#">5.5. Interface Parameters Sub-TLV</a>	<a href="#">19</a>
<a href="#">6. Control Word</a>	<a href="#">20</a>
<a href="#">6.1. PW Types for Which the Control Word is REQUIRED</a>	<a href="#">20</a>
<a href="#">6.2. PW Types for Which the Control Word is NOT Mandatory</a>	<a href="#">21</a>
<a href="#">6.3. LDP Label Withdrawal Procedures</a>	<a href="#">22</a>
<a href="#">6.4. Sequencing Considerations</a>	<a href="#">23</a>
<a href="#">6.4.1. Label Advertisements</a>	<a href="#">23</a>
<a href="#">6.4.2. Label Release</a>	<a href="#">24</a>
<a href="#">7. IANA Considerations</a>	<a href="#">24</a>
<a href="#">7.1. LDP TLV TYPE</a>	<a href="#">24</a>
<a href="#">7.2. LDP Status Codes</a>	<a href="#">24</a>
<a href="#">7.3. FEC Type Name Space</a>	<a href="#">25</a>
<a href="#">8. Security Considerations</a>	<a href="#">25</a>
<a href="#">8.1. Data-Plane Security</a>	<a href="#">25</a>
<a href="#">8.2. Control-Plane Security</a>	<a href="#">26</a>
<a href="#">9. Acknowledgements</a>	<a href="#">27</a>
<a href="#">10. Normative References</a>	<a href="#">27</a>
<a href="#">11. Informative References</a>	<a href="#">27</a>
<a href="#">12. Additional Contributing Authors</a>	<a href="#">28</a>
<a href="#">Appendix A. C-bit Handling Procedures Diagram</a>	<a href="#">31</a>



## 1. Introduction

In [FRAME], [ATM], [PPPHDLC], and [ETH], it is explained how to encapsulate a Layer 2 Protocol Data Unit (PDU) for transmission over an MPLS-enabled network. Those documents specify that a "pseudowire header", consisting of a demultiplexor field, will be prepended to the encapsulated PDU. The pseudowire demultiplexor field is prepended before transmitting a packet on a pseudowire. When the packet arrives at the remote endpoint of the pseudowire, the demultiplexor is what enables the receiver to identify the particular pseudowire on which the packet has arrived. To transmit the packet from one pseudowire endpoint to another, the packet may need to travel through a "Packet Switched Network (PSN) tunnel"; this will require that an additional header be prepended to the packet.

Accompanying documents [CEP, SAToP] specify methods for transporting time-division multiplexing (TDM) digital signals (TDM circuit emulation) over a packet-oriented MPLS-enabled network. The transmission system for circuit-oriented TDM signals is the Synchronous Optical Network (SONET)[SDH]/Synchronous Digital Hierarchy (SDH) [ITUg]. To support TDM traffic, which includes voice, data, and private leased-line service, the pseudowires must emulate the circuit characteristics of SONET/SDH payloads. The TDM signals and payloads are encapsulated for transmission over pseudowires. A pseudowire demultiplexor and a PSN tunnel header is prepended to this encapsulation.

[SAToP] describes methods for transporting low-rate time-division multiplexing (TDM) digital signals (TDM circuit emulation) over PSNs, while [CEP] similarly describes transport of high-rate TDM (SONET/SDH). To support TDM traffic, the pseudowires must emulate the circuit characteristics of the original T1, E1, T3, E3, SONET, or SDH signals. [SAToP] does this by encapsulating an arbitrary but constant amount of the TDM data in each packet, and the other methods encapsulate TDM structures.

In this document, we specify the use of the MPLS Label Distribution Protocol, LDP [RFC3036], as a protocol for setting up and maintaining the pseudowires. In particular, we define new TLVs, FEC elements, parameters, and codes for LDP, which enable LDP to identify pseudowires and to signal attributes of pseudowires. We specify how a pseudowire endpoint uses these TLVs in LDP to bind a demultiplexor field value to a pseudowire, and how it informs the remote endpoint of the binding. We also specify procedures for reporting pseudowire status changes, for passing additional information about the pseudowire as needed, and for releasing the bindings.



In the protocol specified herein, the pseudowire demultiplexor field is an MPLS label. Thus, the packets that are transmitted from one end of the pseudowire to the other are MPLS packets, which must be transmitted through an MPLS tunnel. However, if the pseudowire endpoints are immediately adjacent and penultimate hop popping behavior is in use, the MPLS tunnel may not be necessary. Any sort of PSN tunnel can be used, as long as it is possible to transmit MPLS packets through it. The PSN tunnel can itself be an MPLS LSP, or any other sort of tunnel that can carry MPLS packets. Procedures for setting up and maintaining the MPLS tunnels are outside the scope of this document.

This document deals only with the setup and maintenance of point-to-point pseudowires. Neither point-to-multipoint nor multipoint-to-point pseudowires are discussed.

QoS-related issues are not discussed in this document. The following two figures describe the reference models that are derived from [RFC3985] to support the PW emulated services.

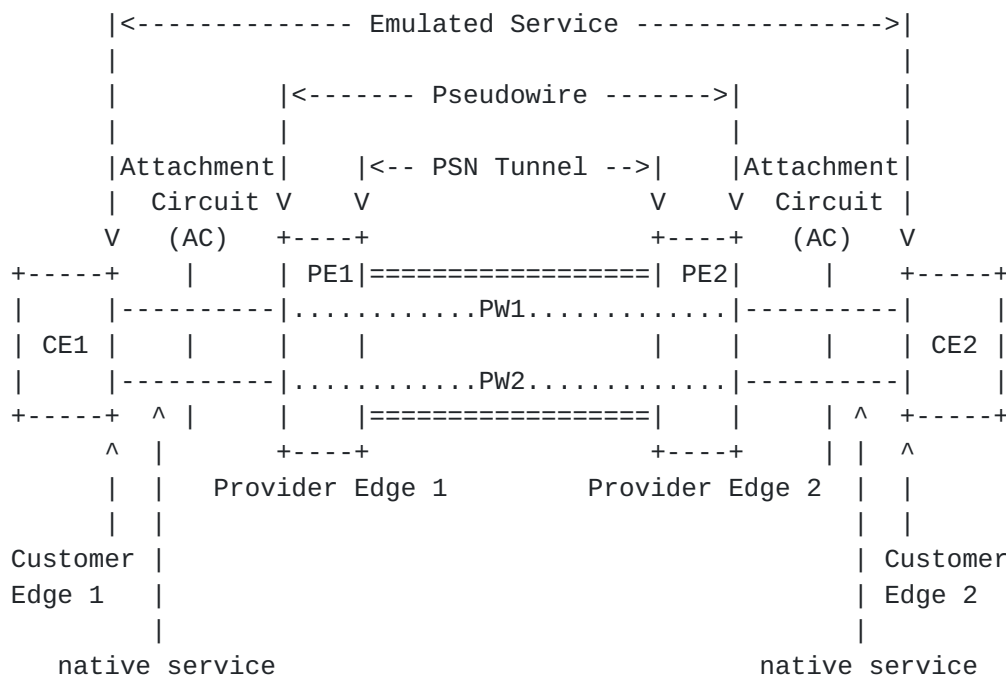


Figure 1: PWE3 Reference Model



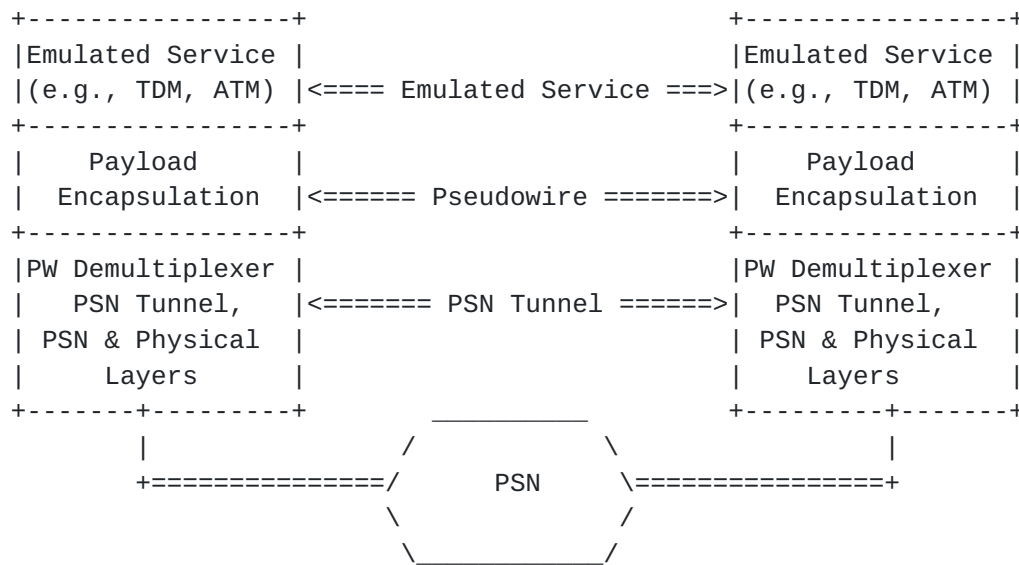


Figure 2: PWE3 Protocol Stack Reference Model

For the purpose of this document, PE1 will be defined as the ingress router, and PE2 as the egress router. A layer 2 PDU will be received at PE1, encapsulated at PE1, transported and decapsulated at PE2, and transmitted out of PE2.

## 2. 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 [\[RFC2119\]](#).

### 3. The Pseudowire Label

Suppose that it is desired to transport Layer 2 PDUs from ingress LSR PE1 to egress LSR PE2, across an intervening MPLS-enabled network. We assume that there is an MPLS tunnel from PE1 to PE2. That is, we assume that PE1 can cause a packet to be delivered to PE2 by encapsulating the packet in an "MPLS tunnel header" and sending the result to one of its adjacencies. The MPLS tunnel is an MPLS Label Switched Path (LSP); thus, putting on an MPLS tunnel encapsulation is a matter of pushing on an MPLS label.

We presuppose that a large number of pseudowires can be carried through a single MPLS tunnel. Thus, it is never necessary to maintain state in the network core for individual pseudowires. We do not presuppose that the MPLS tunnels are point to point; although the pseudowires are point to point, the MPLS tunnels may be multipoint to point. We do not presuppose that PE2 will even be able to determine the MPLS tunnel through which a received packet was transmitted.





(For example, if the MPLS tunnel is an LSP and penultimate hop popping is used, when the packet arrives at PE2, it will contain no information identifying the tunnel.)

When PE2 receives a packet over a pseudowire, it must be able to determine that the packet was in fact received over a pseudowire, and it must be able to associate that packet with a particular pseudowire. PE2 is able to do this by examining the MPLS label that serves as the pseudowire demultiplexor field shown in Figure 2. Call this label the "PW label".

When PE1 sends a Layer 2 PDU to PE2, it creates an MPLS packet by adding the PW label to the packet, thus creating the first entry of the label stack. If the PSN tunnel is an MPLS LSP, the PE1 pushes another label (the tunnel label) onto the packet as the second entry of the label stack. The PW label is not visible again until the MPLS packet reaches PE2. PE2's disposition of the packet is based on the PW label.

If the payload of the MPLS packet is, for example, an ATM AAL5 PDU, the PW label will generally correspond to a particular ATM VC at PE2. That is, PE2 needs to be able to infer from the PW label the outgoing interface and the VPI/VCI value for the AAL5 PDU. If the payload is a Frame Relay PDU, then PE2 needs to be able to infer from the PW label the outgoing interface and the DLCI value. If the payload is an Ethernet frame, then PE2 needs to be able to infer from the PW label the outgoing interface, and perhaps the VLAN identifier. This process is uni-directional and will be repeated independently for bi-directional operation. It is REQUIRED that the same PW ID and PW type be assigned for a given circuit in both directions. The group ID (see below) MUST NOT be required to match in both directions. The transported frame MAY be modified when it reaches the egress router. If the header of the transported Layer 2 frame is modified, this MUST be done at the egress LSR only. Note that the PW label must always be at the bottom of the packet's label stack, and labels MUST be allocated from the per-platform label space.

This document does not specify a method for distributing the MPLS tunnel label or any other labels that may appear above the PW label on the stack. Any acceptable method of MPLS label distribution will do. This document specifies a protocol for assigning and distributing the PW label. This protocol is LDP, extended as specified in the remainder of this document. An LDP session must be set up between the pseudowire endpoints. LDP MUST be used in its "downstream unsolicited" mode. LDP's "liberal label retention" mode SHOULD be used.



In addition to the protocol specified herein, static assignment of PW labels may be used, and implementations of this protocol SHOULD provide support for static assignment.

This document specifies all the procedures necessary to set up and maintain the pseudowires needed to support "unswitched" point-to-point services, where each endpoint of the pseudowire is provisioned with the identify of the other endpoint. There are also protocol mechanisms specified herein that can be used to support switched services and other provisioning models. However, the use of the protocol mechanisms to support those other models and services is not described in this document.

#### **4. Details Specific to Particular Emulated Services**

##### **4.1. IP Layer 2 Transport**

This mode carries IP packets over a pseudowire. The encapsulation used is according to [RFC3032]. The PW control word MAY be inserted between the MPLS label stack and the IP payload. The encapsulation of the IP packets for forwarding on the attachment circuit is implementation specific, is part of the native service processing (NSP) function [RFC3985], and is outside the scope of this document.

#### **5. LDP**

The PW label bindings are distributed using the LDP downstream unsolicited mode described in [RFC3036]. The PEs will establish an LDP session using the Extended Discovery mechanism described in [LDP, sections 2.4.2 and 2.5].

An LDP Label Mapping message contains an FEC TLV, a Label TLV, and zero or more optional parameter TLVs.

The FEC TLV is used to indicate the meaning of the label. In the current context, the FEC TLV would be used to identify the particular pseudowire that a particular label is bound to. In this specification, we define two new FEC TLVs to be used for identifying pseudowires. When setting up a particular pseudowire, only one of these FEC TLVs is used. The one to be used will depend on the particular service being emulated and on the particular provisioning model being supported.

LDP allows each FEC TLV to consist of a set of FEC elements. For setting up and maintaining pseudowires, however, each FEC TLV MUST contain exactly one FEC element.



The LDP base specification has several kinds of label TLVs, including the Generic Label TLV, as specified in [\[RFC3036\]](#), [section 3.4.2.1](#). For setting up and maintaining pseudowires, the Generic Label TLV MUST be used.

### 5.1. LDP Extensions

This document specifies no new LDP messages.

This document specifies the following new TLVs to be used with LDP:

TLV	Specified in Section	Defined for Message
=====		
PW Status TLV	5.4.2	Notification
PW Interface Parameters TLV	5.3.2.1	FEC
PW Grouping ID TLV	5.3.2.2	FEC

Additionally, the following new FEC element types are defined:

FEC Element Type	Specified in Section	Defined for Message
=====		
0x80	5.2	FEC
0x81	5.3	FEC

The following new LDP error codes are also defined:

Status Code	Specified in Section
=====	
"Illegal C-Bit"	6.1
"Wrong C-Bit"	6.2
"Incompatible bit-rate"	<a href="#">[CEP]</a>
"CEP/TDM mis-configuration"	<a href="#">[CEP]</a>
"PW status"	5.4.2
"Unassigned/Unrecognized TAI"	5.3.3
"Generic Misconfiguration Error"	<a href="#">[SAToP]</a>
"Label Withdraw PW Status Method Not Supported"	5.4.1

### 5.2. The PWid FEC Element

The PWid FEC element may be used whenever both pseudowire endpoints have been provisioned with the same 32-bit identifier for the pseudowire.









The Group ID is very useful for sending wild card label withdrawals, or PW wild card status notification messages to remote PEs upon physical port failure.

- PW ID

A non-zero 32-bit connection ID that, together with the PW type, identifies a particular PW. Note that the PW ID and the PW type MUST be the same at both endpoints.

- Interface Parameter Sub-TLV

This variable-length TLV is used to provide interface-specific parameters, such as attachment circuit MTU.

Note that as the "interface parameter sub-TLV" is part of the FEC, the rules of LDP make it impossible to change the interface parameters once the pseudowire has been set up. Thus, the interface parameters field must not be used to pass information, such as status information, that may change during the life of the pseudowire. Optional parameter TLVs should be used for that purpose.

Using the Pwid FEC, each of the two pseudowire endpoints independently initiates the setup of a unidirectional LSP. An outgoing LSP and an incoming LSP are bound together into a single pseudowire if they have the same PW ID and PW type.

### **5.3. The Generalized Pwid FEC Element**

The Pwid FEC element can be used if a unique 32-bit value has been assigned to the PW, and if each endpoint has been provisioned with that value. The Generalized Pwid FEC element requires that the PW endpoints be uniquely identified; the PW itself is identified as a pair of endpoints. In addition, the endpoint identifiers are structured to support applications where the identity of the remote endpoints needs to be auto-discovered rather than statically configured.

The "Generalized Pwid FEC Element" is FEC type 0x81.

The Generalized Pwid FEC Element does not contain anything corresponding to the "Group ID" of the Pwid FEC element. The functionality of the "Group ID" is provided by a separate optional LDP TLV, the "PW Grouping TLV", described below. The Interface Parameters field of the Pwid FEC element is also absent; its functionality is replaced by the optional Interface Parameters TLV, described below.



### 5.3.1. Attachment Identifiers

As discussed in [RFC3985], a pseudowire can be thought of as connecting two "forwarders". The protocol used to set up a pseudowire must allow the forwarder at one end of a pseudowire to identify the forwarder at the other end. We use the term "attachment identifier", or "AI", to refer to the field that the protocol uses to identify the forwarders. In the PWid FEC, the PWid field serves as the AI. In this section, we specify a more general form of AI that is structured and of variable length.

Every Forwarder in a PE must be associated with an Attachment Identifier (AI), either through configuration or through some algorithm. The Attachment Identifier must be unique in the context of the PE router in which the Forwarder resides. The combination <PE router IP address, AI> must be globally unique.

It is frequently convenient to regard a set of Forwarders as being members of a particular "group", where PWs may only be set up among members of a group. In such cases, it is convenient to identify the Forwarders relative to the group, so that an Attachment Identifier would consist of an Attachment Group Identifier (AGI) plus an Attachment Individual Identifier (AII).

An Attachment Group Identifier may be thought of as a VPN-id, or a VLAN identifier, some attribute that is shared by all the Attachment PWs (or pools thereof) that are allowed to be connected.

The details of how to construct the AGI and AII fields identifying the pseudowire endpoints are outside the scope of this specification. Different pseudowire applications, and different provisioning models, will require different sorts of AGI and AII fields. The specification of each such application and/or model must include the rules for constructing the AGI and AII fields.

As previously discussed, a (bidirectional) pseudowire consists of a pair of unidirectional LSPs, one in each direction. If a particular pseudowire connects PE1 with PE2, the PW direction from PE1 to PE2 can be identified as:

<PE1, <AGI, AII1>, PE2, <AGI, AII2>>,

The PW direction from PE2 to PE1 can be identified by:

<PE2, <AGI, AII2>, PE1, <AGI, AII1>>.



Note that the AGI must be the same at both endpoints, but the AII will in general be different at each endpoint. Thus, from the perspective of a particular PE, each pseudowire has a local or "Source AII", and a remote or "Target AII". The pseudowire setup protocol can carry all three of these quantities:

- Attachment Group Identifier (AGI)
- Source Attachment Individual Identifier (SAII)
- Target Attachment Individual Identifier (TAII)

If the AGI is non-null, then the Source AI (SAI) consists of the AGI together with the SAII, and the Target AI (TAI) consists of the TAII together with the AGI. If the AGI is null, then the SAII and TAII are the SAI and TAI, respectively.

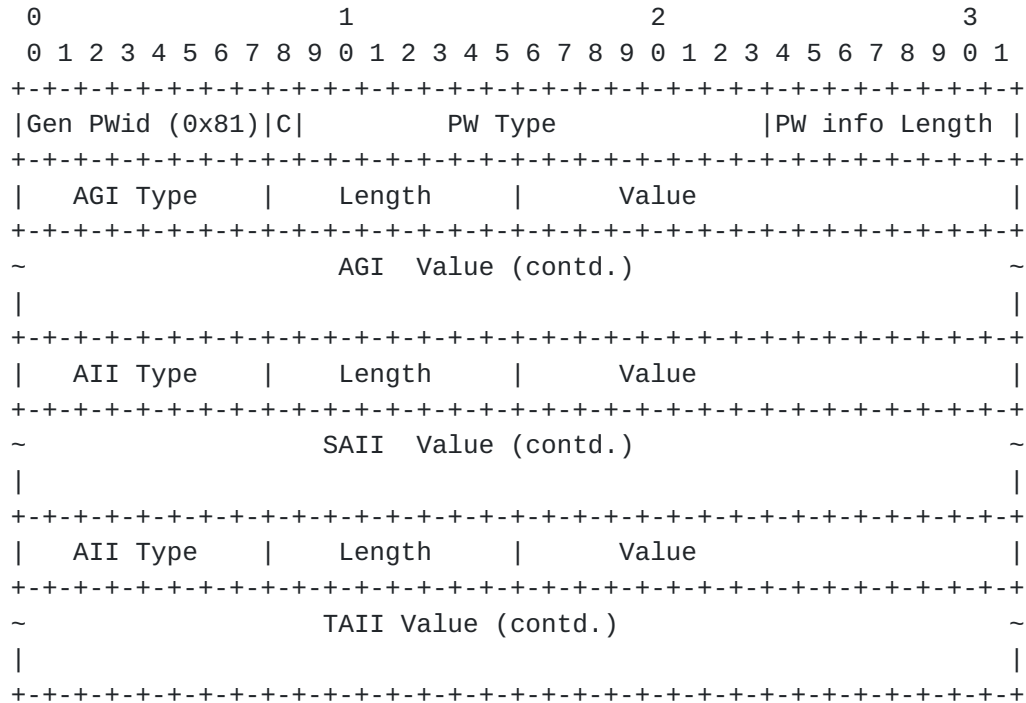
The interpretation of the SAI and TAI is a local matter at the respective endpoint.

The association of two unidirectional LSPs into a single bidirectional pseudowire depends on the SAI and the TAI. Each application and/or provisioning model that uses the Generalized ID FEC element must specify the rules for performing this association.



### 5.3.2. Encoding the Generalized ID FEC Element

FEC element type 0x81 is used. The FEC element is encoded as follows:



This document does not specify the AII and AGI type field values; specification of the type field values to be used for a particular application is part of the specification of that application. IANA has assigned these values using the method defined in the [\[IANA\]](#) document.

The SAII, TAI, and AGI are simply carried as octet strings. The length byte specifies the size of the Value field. The null string can be sent by setting the length byte to 0. If a particular application does not need all three of these sub-elements, it MUST send all the sub-elements but set the length to 0 for the unused sub-elements.

The PW information length field contains the length of the SAII, TAI, and AGI, combined in octets. If this value is 0, then it references all PWs using the specified grouping ID. In this case, there are no other FEC element fields (AGI, SAII, etc.) present, nor any interface parameters TLVs.

Note that the interpretation of a particular field as AGI, SAII, or TAI depends on the order of its occurrence. The type field identifies the type of the AGI, SAII, or TAI. When comparing two





occurrences of an AGI (or SAII or TAI), the two occurrences are considered identical if the type, length, and value fields of one are identical, respectively, to those of the other.

#### 5.3.2.1. Interface Parameters TLV

This TLV MUST only be used when sending the Generalized PW FEC. It specifies interface-specific parameters. Specific parameters, when applicable, MUST be used to validate that the PEs and the ingress and egress ports at the edges of the circuit have the necessary capabilities to interoperate with each other.

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|0|0| PW Intf P. TLV (0x096B) |                               Length |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Sub-TLV Type | Length | Variable Length Value |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Variable Length Value |
|                               " |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

A more detailed description of this field can be found in the section "Interface Parameters Sub-TLV", below.

#### 5.3.2.2. PW Grouping TLV

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|0|0|PW Grouping ID TLV (0x096C)|                               Length |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Value |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

The PW Grouping ID is an arbitrary 32-bit value that represents an arbitrary group of PWs. It is used to create group PWs; for example, a PW Grouping ID can be used as a port index and assigned to all PWs that lead to that port. Use of the PW Grouping ID enables one to send "wild card" label withdrawals, or "wild card" status notification messages, to remote PEs upon physical port failure.

Note Well: The PW Grouping ID is different from, and has no relation to, the Attachment Group Identifier.

The PW Grouping ID TLV is not part of the FEC and will not be advertised except in the PW FEC advertisement. The advertising PE



MAY use the wild card withdraw semantics, but the remote PEs MUST implement support for wild card messages. This TLV MUST only be used when sending the Generalized PW ID FEC.

To issue a wildcard command (status or withdraw):

- Set the PW Info Length to 0 in the Generalized ID FEC Element.
- Send only the PW Grouping ID TLV with the FEC (no AGI/SAII/TAII is sent).

### **5.3.3. Signaling Procedures**

In order for PE1 to begin signaling PE2, PE1 must know the address of the remote PE2, and a TAI. This information may have been configured at PE1, or it may have been learned dynamically via some autodiscovery procedure.

The egress PE (PE1), which has knowledge of the ingress PE, initiates the setup by sending a Label Mapping Message to the ingress PE (PE2). The Label Mapping message contains the FEC TLV, carrying the Generalized PWid FEC Element (type 0x81). The Generalized PWid FEC element contains the AGI, SAI, and TAI information.

Next, when PE2 receives such a Label Mapping message, PE2 interprets the message as a request to set up a PW whose endpoint (at PE2) is the Forwarder identified by the TAI. From the perspective of the signaling protocol, exactly how PE2 maps AIs to Forwarders is a local matter. In some Virtual Private Wire Services (VPWS) provisioning models, the TAI might, for example, be a string that identifies a particular Attachment Circuit, such as "ATM3VPI4VCI5", or it might, for example, be a string, such as "Fred", that is associated by configuration with a particular Attachment Circuit. In VPLS, the AGI could be a VPN-id, identifying a particular VPLS instance.

If PE2 cannot map the TAI to one of its Forwarders, then PE2 sends a Label Release message to PE1, with a Status Code of "Unassigned/Unrecognized TAI", and the processing of the Label Mapping message is complete.

The FEC TLV sent in a Label Release message is the same as the FEC TLV received in the Label Mapping being released (but without the interface parameter TLV). More generally, the FEC TLV is the same in all LDP messages relating to the same PW. In a Label Release, this means that the SAI is the remote peer's AI and the TAI is the sender's local AI.



If the Label Mapping Message has a valid TAI, PE2 must decide whether to accept it. The procedures for so deciding will depend on the particular type of Forwarder identified by the TAI. Of course, the Label Mapping message may be rejected due to standard LDP error conditions as detailed in [[RFC3036](#)].

If PE2 decides to accept the Label Mapping message, then it has to make sure that a PW LSP is set up in the opposite (PE1-->PE2) direction. If it has already signaled for the corresponding PW LSP in that direction, nothing more needs to be done. Otherwise, it must initiate such signaling by sending a Label Mapping message to PE1. This is very similar to the Label Mapping message PE2 received, but the SAI and TAI are reversed.

Thus, a bidirectional PW consists of two LSPs, where the FEC of one has the SAI and TAI reversed with respect to the FEC of the other.

#### **[5.4.](#) Signaling of Pseudowire Status**

##### **[5.4.1.](#) Use of Label Mappings Messages**

The PEs MUST send Label Mapping Messages to their peers as soon as the PW is configured and administratively enabled, regardless of the attachment circuit state. The PW label should not be withdrawn unless the operator administratively configures the pseudowire down (or the PW configuration is deleted entirely). Using the procedures outlined in this section, a simple label withdraw method MAY also be supported as a legacy means of signaling PW status and AC status. In any case, if the label-to-PW binding is not available, the PW MUST be considered in the down state.

Once the PW status negotiation procedures are completed, if they result in the use of the label withdraw method for PW status communication, and this method is not supported by one of the PEs, then that PE must send a Label Release Message to its peer with the following error:

"Label Withdraw PW Status Method Not Supported"

If the label withdraw method for PW status communication is selected for the PW, it will result in the Label Mapping Message being advertised only if the attachment circuit is active. The PW status signaling procedures described in this section MUST be fully implemented.



### 5.4.2. Signaling PW Status

The PE devices use an LDP TLV to indicate status to their remote peers. This PW Status TLV contains more information than the alternative simple Label Withdraw message.

The format of the PW Status TLV is:

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|1|0|   PW Status (0x096A)   |           Length           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Status Code                  |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

The status code is a 4-octet bit field as specified in the PW IANA Allocations document [[IANA](#)]. The length specifies the length of the Status Code field in octets (equal to 4).

Each bit in the status code field can be set individually to indicate more than a single failure at once. Each fault can be cleared by sending an appropriate Notification message in which the respective bit is cleared. The presence of the lowest bit (PW Not Forwarding) acts only as a generic failure indication when there is a link-down event for which none of the other bits apply.

The Status TLV is transported to the remote PW peer via the LDP Notification message. The general format of the Notification Message is:

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|0|   Notification (0x0001)   |           Message Length           |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Message ID                        |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               Status (TLV)                      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               PW Status TLV                     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|                               PWId FEC TLV or Generalized ID FEC TLV
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```





The Status TLV status code is set to 0x00000028, "PW status", to indicate that PW status follows. Since this notification does not refer to any particular message, the Message Id and Message Type fields are set to 0.

The PW FEC TLV SHOULD not include the interface parameter sub-TLVs, as they are ignored in the context of this message. When a PE's attachment circuit encounters an error, use of the PW Notification Message allows the PE to send a single "wild card" status message, using a PW FEC TLV with only the group ID set, to denote this change in status for all affected PW connections. This status message contains either the PW FEC TLV with only the group ID set, or else it contains the Generalized FEC TLV with only the PW Grouping ID TLV.

As mentioned above, the Group ID field of the PWid FEC element, or the PW Grouping ID TLV used with the Generalized ID FEC element, can be used to send a status notification for all arbitrary sets of PWs. This procedure is OPTIONAL, and if it is implemented, the LDP Notification message should be as follows: If the PWid FEC element is used, the PW information length field is set to 0, the PW ID field is not present, and the interface parameter sub-TLVs are not present. If the Generalized FEC element is used, the AGI, SAII, and TAI are not present, the PW information length field is set to 0, the PW Grouping ID TLV is included, and the Interface Parameters TLV is omitted. For the purpose of this document, this is called the "wild card PW status notification procedure", and all PEs implementing this design are REQUIRED to accept such a notification message but are not required to send it.

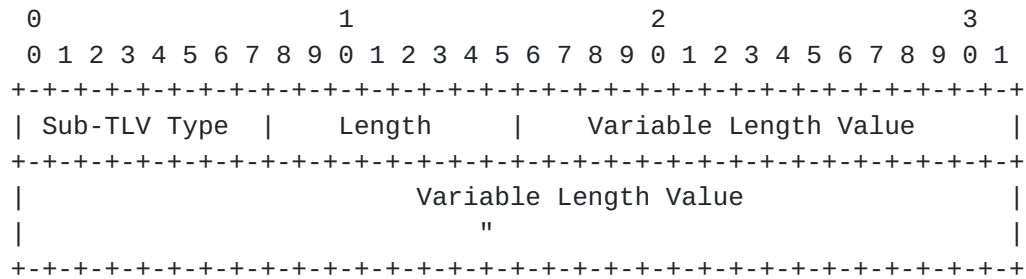
#### **5.4.3. Pseudowire Status Negotiation Procedures**

When a PW is first set up, the PEs MUST attempt to negotiate the usage of the PW status TLV. This is accomplished as follows: A PE that supports the PW Status TLV MUST include it in the initial Label Mapping message following the PW FEC and the interface parameter sub-TLVs. The PW Status TLV will then be used for the lifetime of the pseudowire. This is shown in the following diagram:









The interface parameter sub-TLV type values are specified in "IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)" [[IANA](#)].

The Length field is defined as the length of the interface parameter including the parameter id and length field itself. Processing of the interface parameters should continue when unknown interface parameters are encountered, and they MUST be silently ignored.

- Interface MTU sub-TLV type

A 2-octet value indicating the MTU in octets. This is the Maximum Transmission Unit, excluding encapsulation overhead, of the egress packet interface that will be transmitting the decapsulated PDU that is received from the MPLS-enabled network. This parameter is applicable only to PWs transporting packets and is REQUIRED for these PW types. If this parameter does not match in both directions of a specific PW, that PW MUST NOT be enabled.

- Optional Interface Description string sub-TLV type

This arbitrary, and OPTIONAL, interface description string is used to send a human-readable administrative string describing the interface to the remote. This parameter is OPTIONAL and is applicable to all PW types. The interface description parameter string length is variable and can be from 0 to 80 octets. Human-readable text MUST be provided in the UTF-8 charset using the Default Language [[RFC2277](#)].

## 6. Control Word

### 6.1. PW Types for Which the Control Word is REQUIRED

The Label Mapping messages that are sent in order to set up these PWs MUST have c=1. When a Label Mapping message for a PW of one of these types is received and c=0, a Label Release message MUST be sent, with an "Illegal C-bit" status code. In this case, the PW will not be enabled.



## 6.2. PW Types for Which the Control Word is NOT Mandatory

If a system is capable of sending and receiving the control word on PW types for which the control word is not mandatory, then each such PW endpoint MUST be configurable with a parameter that specifies whether the use of the control word is PREFERRED or NOT PREFERRED. For each PW, there MUST be a default value of this parameter. This specification does NOT state what the default value should be.

If a system is NOT capable of sending and receiving the control word on PW types for which the control word is not mandatory, then it behaves exactly as if it were configured for the use of the control word to be NOT PREFERRED.

If a Label Mapping message for the PW has already been received but no Label Mapping message for the PW has yet been sent, then the procedure is as follows:

- i. If the received Label Mapping message has c=0, send a Label Mapping message with c=0; the control word is not used.
- ii. If the received Label Mapping message has c=1 and the PW is locally configured such that the use of the control word is preferred, then send a Label Mapping message with c=1; the control word is used.
- iii. If the received Label Mapping message has c=1 and the PW is locally configured such that the use of the control word is not preferred or the control word is not supported, then act as if no Label Mapping message for the PW had been received (i.e., proceed to the next paragraph).

If a Label Mapping message for the PW has not already been received (or if the received Label Mapping message had c=1 and either local configuration says that the use of the control word is not preferred or the control word is not supported), then send a Label Mapping message in which the c bit is set to correspond to the locally configured preference for use of the control word. (That is, set c=1 if locally configured to prefer the control word, and set c=0 if locally configured to prefer not to use the control word or if the control word is not supported).





The next action depends on what control message is next received for that PW. The possibilities are as follows:

- i. A Label Mapping message with the same c bit value as specified in the Label Mapping message that was sent. PW setup is now complete, and the control word is used if c=1 but is not used if c=0.
- ii. A Label Mapping message with c=1, but the Label Mapping message that was sent has c=0. In this case, ignore the received Label Mapping message and continue to wait for the next control message for the PW.
- iii. A Label Mapping message with c=0, but the Label Mapping message that was sent has c=1. In this case, send a Label Withdraw message with a "Wrong C-bit" status code, followed by a Label Mapping message that has c=0. PW setup is now complete, and the control word is not used.
- iv. A Label Withdraw message with the "Wrong c-bit" status code. Treat as a normal Label Withdraw, but do not respond. Continue to wait for the next control message for the PW.

If at any time after a Label Mapping message has been received a corresponding Label Withdraw or Release is received, the action taken is the same as for any Label Withdraw or Release that might be received at any time.

If both endpoints prefer the use of the control word, this procedure will cause it to be used. If either endpoint prefers not to use the control word or does not support the control word, this procedure will cause it not to be used. If one endpoint prefers to use the control word but the other does not, the one that prefers not to use it has no extra protocol to execute; it just waits for a Label Mapping message that has c=0.

The diagram in [Appendix A](#) illustrates the above procedure.

### **6.3. LDP Label Withdrawal Procedures**

As mentioned above, the Group ID field of the PWid FEC element, or the PW Grouping ID TLV used with the Generalized ID FEC element, can be used to withdraw all PW labels associated with a particular PW group. This procedure is OPTIONAL, and if it is implemented, the LDP Label Withdraw message should be as follows: If the PWid FEC element is used, the PW information length field is set to 0, the PW ID field is not present, the interface parameter sub-TLVs are not present, and the Label TLV is not present.



If the Generalized FEC element is used, the AGI, SAII, and TAI are not present, the PW information length field is set to 0, the PW Grouping ID TLV is included, the Interface Parameters TLV is not present, and the Label TLV is not present. For the purpose of this document, this is called the "wild card withdraw procedure", and all PEs implementing this design are REQUIRED to accept such withdrawn message but are not required to send it. Note that the PW Grouping ID TLV only applies to PWs using the Generalized ID FEC element, while the Group ID only applies to PWid FEC element.

The interface parameter sub-TLVs, or TLV, MUST NOT be present in any LDP PW Label Withdraw or Label Release message. A wild card Label Release message MUST include only the group ID, or Grouping ID TLV. A Label Release message initiated by a PE router must always include the PW ID.

#### **6.4. Sequencing Considerations**

In the case where the router considers the sequence number field in the control word, it is important to note the following details when advertising labels.

##### **6.4.1. Label Advertisements**

After a label has been withdrawn by the output router and/or released by the input router, care must be taken not to advertise (re-use) the same released label until the output router can be reasonably certain that old packets containing the released label no longer persist in the MPLS-enabled network.

This precaution is required to prevent the imposition router from restarting packet forwarding with a sequence number of 1 when it receives a Label Mapping message that binds the same FEC to the same label if there are still older packets in the network with a sequence number between 1 and 32768. For example, if there is a packet with sequence number= $n$ , where  $n$  is in the interval  $[1, 32768]$  traveling through the network, it would be possible for the disposition router to receive that packet after it re-advertises the label. Since the label has been released by the imposition router, the disposition router SHOULD be expecting the next packet to arrive with a sequence number of 1. Receipt of a packet with a sequence number equal to  $n$  will result in  $n$  packets potentially being rejected by the disposition router until the imposition router imposes a sequence number of  $n+1$  into a packet. Possible methods to avoid this are for the disposition router always to advertise a different PW label, or for the disposition router to wait for a sufficient time before



attempting to re-advertise a recently released label. This is only an issue when sequence number processing is enabled at the disposition router.

#### 6.4.2. Label Release

In situations where the imposition router wants to restart forwarding of packets with sequence number 1, the router shall 1) send to the disposition router a Label Release Message, and 2) send to the disposition router a Label Request message. When sequencing is supported, advertisement of a PW label in response to a Label Request message MUST also consider the issues discussed in the section on Label Advertisements.

### 7. IANA Considerations

#### 7.1. LDP TLV TYPE

This document uses several new LDP TLV types; IANA already maintains a registry of name "TLV TYPE NAME SPACE" defined by [RFC 3036](#). The following values are suggested for assignment:

TLV type	Description
=====	
0x096A	PW Status TLV
0x096B	PW Interface Parameters TLV
0x096C	Group ID TLV

#### 7.2. LDP Status Codes

This document uses several new LDP status codes; IANA already maintains a registry of name "STATUS CODE NAME SPACE" defined by [RFC 3036](#). The following values are suggested for assignment:

Range/Value	E	Description	Reference
-----	----	-----	-----
0x00000024	0	Illegal C-Bit	<a href="#">[RFC4447]</a>
0x00000025	0	Wrong C-Bit	<a href="#">[RFC4447]</a>
0x00000026	0	Incompatible bit-rate	<a href="#">[RFC4447]</a>
0x00000027	0	CEP-TDM mis-configuration	<a href="#">[RFC4447]</a>
0x00000028	0	PW Status	<a href="#">[RFC4447]</a>
0x00000029	0	Unassigned/Unrecognized TAI	<a href="#">[RFC4447]</a>
0x0000002A	0	Generic Misconfiguration Error	<a href="#">[RFC4447]</a>
0x0000002B	0	Label Withdraw PW Status Method	<a href="#">[RFC4447]</a>



### **7.3. FEC Type Name Space**

This document uses two new FEC element types, 0x80 and 0x81, from the registry "FEC Type Name Space" for the Label Distribution Protocol (LDP [RFC 3036](#)).

## **8. Security Considerations**

This document specifies the LDP extensions that are needed for setting up and maintaining pseudowires. The purpose of setting up pseudowires is to enable Layer 2 frames to be encapsulated in MPLS and transmitted from one end of a pseudowire to the other. Therefore, we treat the security considerations for both the data plane and the control plane.

### **8.1. Data-Plane Security**

With regard to the security of the data plane, the following areas must be considered:

- MPLS PDU inspection
- MPLS PDU spoofing
- MPLS PDU alteration
- MPLS PSN protocol security
- Access Circuit security
- Denial-of-service prevention on the PE routers

When an MPLS PSN is used to provide pseudowire service, there is a perception that security MUST be at least equal to the currently deployed Layer 2 native protocol networks that the MPLS/PW network combination is emulating. This means that the MPLS-enabled network SHOULD be isolated from outside packet insertion in such a way that it SHOULD not be possible to insert an MPLS packet into the network directly. To prevent unwanted packet insertion, it is also important to prevent unauthorized physical access to the PSN, as well as unauthorized administrative access to individual network elements.

As mentioned above, as MPLS enabled network should not accept MPLS packets from its external interfaces (i.e., interfaces to CE devices or to other providers' networks) unless the top label of the packet was legitimately distributed to the system from which the packet is being received. If the packet's incoming interface leads to a different SP (rather than to a customer), an appropriate trust relationship must also be present, including the trust that the other SP also provides appropriate security measures.

The three main security problems faced when using an MPLS-enabled network to transport PWs are spoofing, alteration, and inspection.





First, there is a possibility that the PE receiving PW PDUs will get a PDU that appears to be from the PE transmitting the PW into the PSN, but that was not actually transmitted by the PE originating the PW. (That is, the specified encapsulations do not by themselves enable the decapsulator to authenticate the encapsulator.) A second problem is the possibility that the PW PDU will be altered between the time it enters the PSN and the time it leaves the PSN (i.e., the specified encapsulations do not by themselves assure the decapsulator of the packet's integrity.) A third problem is the possibility that the PDU's contents will be seen while the PDU is in transit through the PSN (i.e., the specification encapsulations do not ensure privacy.) How significant these issues are in practice depends on the security requirements of the applications whose traffic is being sent through the tunnel, and how secure the PSN itself is.

## 8.2. Control-Plane Security

General security considerations with regard to the use of LDP are specified in [section 5 of RFC 3036](#). Those considerations also apply to the case where LDP is used to set up pseudowires.

A pseudowire connects two attachment circuits. It is important to make sure that LDP connections are not arbitrarily accepted from anywhere, or else a local attachment circuit might get connected to an arbitrary remote attachment circuit. Therefore, an incoming LDP session request MUST NOT be accepted unless its IP source address is known to be the source of an "eligible" LDP peer. The set of eligible peers could be pre-configured (either as a list of IP addresses, or as a list of address/mask combinations), or it could be discovered dynamically via an auto-discovery protocol that is itself trusted. (Obviously, if the auto-discovery protocol were not trusted, the set of "eligible peers" it produces could not be trusted.)

Even if an LDP connection request appears to come from an eligible peer, its source address may have been spoofed. Therefore, some means of preventing source address spoofing must be in place. For example, if all the eligible peers are in the same network, source address filtering at the border routers of that network could eliminate the possibility of source address spoofing.

The LDP MD5 authentication key option, as described in [section 2.9 of RFC 3036](#), MUST be implemented, and for a greater degree of security, it must be used. This provides integrity and authentication for the LDP messages and eliminates the possibility of source address spoofing. Use of the MD5 option does not provide privacy, but privacy of the LDP control messages is not usually considered important. As the MD5 option relies on the configuration of pre-



shared keys, it does not provide much protection against replay attacks. In addition, its reliance on pre-shared keys may make it very difficult to deploy when the set of eligible neighbors is determined by an auto-configuration protocol.

When the Generalized ID FEC Element is used, it is possible that a particular LDP peer may be one of the eligible LDP peers but may not be the right one to connect to the particular attachment circuit identified by the particular instance of the Generalized ID FEC element. However, given that the peer is known to be one of the eligible peers (as discussed above), this would be the result of a configuration error, rather than a security problem. Nevertheless, it may be advisable for a PE to associate each of its local attachment circuits with a set of eligible peers rather than have just a single set of eligible peers associated with the PE as a whole.

## **9. Acknowledgements**

The authors wish to acknowledge the contributions of Vach Kompella, Vanson Lim, Wei Luo, Himanshu Shah, and Nick Weeds.

## **10. Normative References**

- [RFC2119] Bradner S., "Key words for use in RFCs to Indicate Requirement Levels", [RFC 2119](#), March 1997
- [RFC3036] Andersson, L., Doolan, P., Feldman, N., Fredette, A., and B. Thomas, "LDP Specification", [RFC 3036](#), January 2001.
- [RFC3032] Rosen, E., Tappan, D., Fedorkow, G., Rekhter, Y., Farinacci, D., Li, T., and A. Conta, "MPLS Label Stack Encoding", [RFC 3032](#), January 2001.
- [IANA] Martini, L., "IANA Allocations for Pseudowire Edge to Edge Emulation (PWE3)", [BCP 116](#), [RFC 4446](#), April 2006.

## **11. Informative References**

- [CEP] Malis, A., Pate, P., Cohen, R., Ed., and D. Zelig, "SONET/SDH Circuit Emulation Service Over Packet (CEP)", Work in Progress.
- [SAToP] Vainshtein, A., Ed. and Y. Stein, Ed., "Structure-Agnostic TDM over Packet (SAToP)", Work in Progress.



- [FRAME] Martini, L., Ed. and C. Kawa, Ed., "Encapsulation Methods for Transport of Frame Relay Over MPLS Networks", Work in Progress.
- [ATM] Martini, L., Ed., El-Aawar, N., and M. Bocci, Ed., "Encapsulation Methods for Transport of ATM Over MPLS Networks", Work in Progress.
- [PPPHDLC] Martini, L., Rosen, E., Heron, G., and A. Malis, "Encapsulation Methods for Transport of PPP/HDLC Frames Over IP and MPLS Networks", Work in Progress.
- [ETH] Martini, L., Rosen, E., El-Aawar, N., and G. Heron, "Encapsulation Methods for Transport of Ethernet Over MPLS Networks", [RFC 4448](#), April 2006.
- [SDH] American National Standards Institute, "Synchronous Optical Network Formats," ANSI T1.105-1995.
- [ITUG] ITU Recommendation G.707, "Network Node Interface For The Synchronous Digital Hierarchy", 1996.
- [RFC3985] Bryant, S. and P. Pate, "Pseudo Wire Emulation Edge-to-Edge (PWE3) Architecture", [RFC 3985](#), March 2005.
- [RFC2277] Alvestrand, H., "IETF Policy on Character Sets and Languages", [BCP 18](#), [RFC 2277](#), January 1998.

## **[12.](#) Additional Contributing Authors**

Dimitri Stratton Vlachos  
Mazu Networks, Inc.  
125 Cambridgepark Drive  
Cambridge, MA 02140

EMail: [d@mazunetworks.com](mailto:d@mazunetworks.com)

Jayakumar Jayakumar,  
Cisco Systems Inc.  
225, E.Tasman, MS-SJ3/3,  
San Jose, CA, 95134

EMail: [jjayakum@cisco.com](mailto:jjayakum@cisco.com)



Alex Hamilton,  
Cisco Systems Inc.  
285 W. Tasman, MS-SJCI/3/4,  
San Jose, CA, 95134

EMail: tahamilt@cisco.com

Steve Vogelsang  
ECI Telecom  
Omega Corporate Center  
1300 Omega Drive  
Pittsburgh, PA 15205

EMail: stephen.vogelsang@ecitele.com

John Shirron  
ECI Telecom  
Omega Corporate Center  
1300 Omega Drive  
Pittsburgh, PA 15205

EMail: john.shirron@ecitele.com

Andrew G. Malis  
Tellabs  
90 Rio Robles Dr.  
San Jose, CA 95134

EMail: Andy.Malis@tellabs.com

Vinai Sirkay  
Redback Networks  
300 Holger Way  
San Jose, CA 95134

EMail: vsirkay@redback.com

Vasile Radoaca  
Nortel Networks  
600 Technology Park  
Billerica MA 01821

EMail: vasile@nortelnetworks.com





Chris Liljenstolpe  
Alcatel  
11600 Sallie Mae Dr.  
9th Floor  
Reston, VA 20193

EMail: [chris.liljenstolpe@alcatel.com](mailto:chris.liljenstolpe@alcatel.com)

Dave Cooper  
Global Crossing  
960 Hamlin Court  
Sunnyvale, CA 94089

EMail: [dcooper@gblx.net](mailto:dcooper@gblx.net)

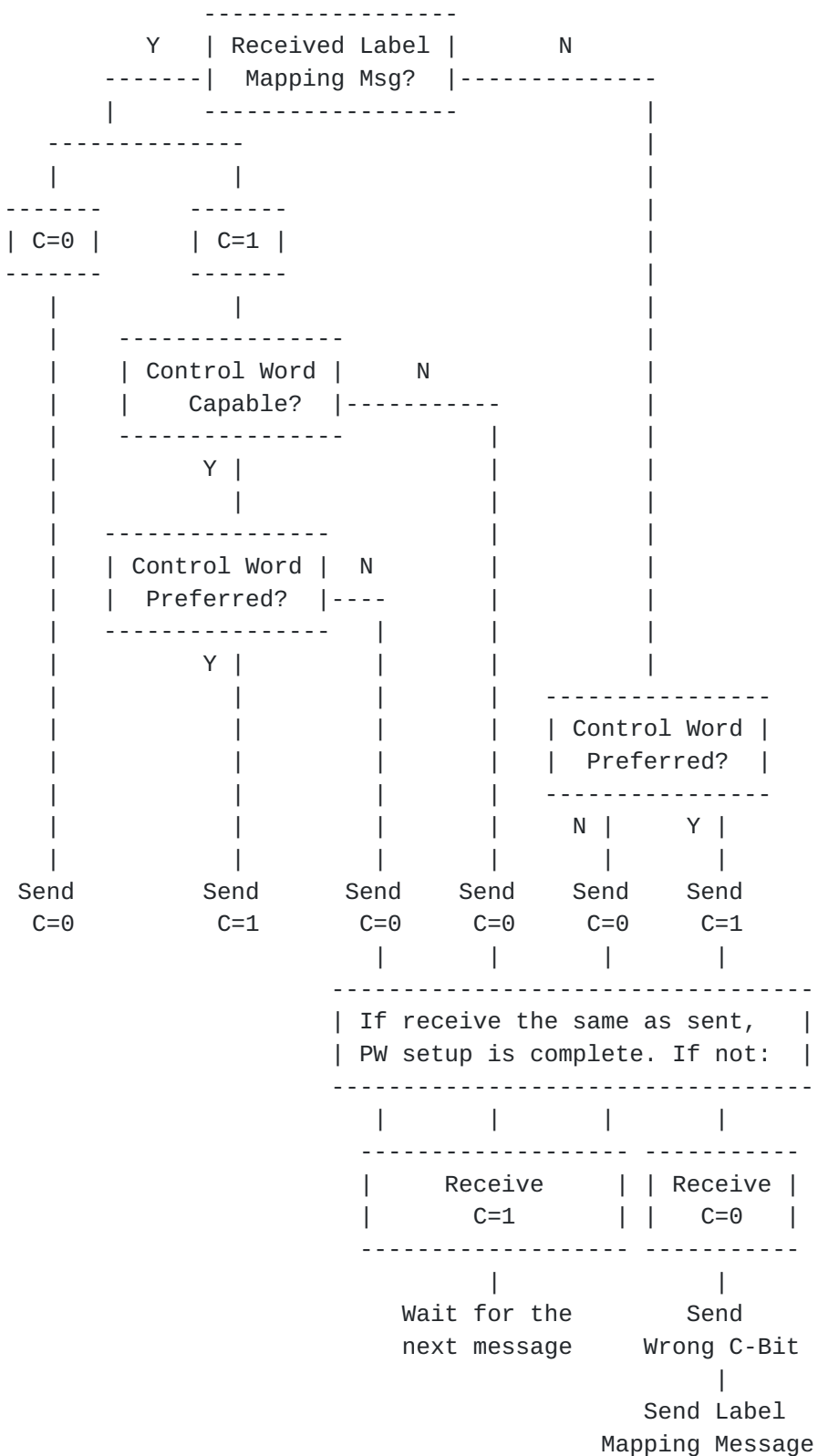
Kireeti Kompella  
Juniper Networks  
1194 N. Mathilda Ave  
Sunnyvale, CA 94089

EMail: [kireeti@juniper.net](mailto:kireeti@juniper.net)

Dan Tappan  
Cisco Systems, Inc.  
1414 Massachusetts Avenue  
Boxborough, MA 01719

EMail: [tappan@cisco.com](mailto:tappan@cisco.com)



**Appendix A. C-bit Handling Procedures Diagram**



Authors' Addresses

Luca Martini  
Cisco Systems, Inc.  
9155 East Nichols Avenue, Suite 400  
Englewood, CO, 80112

EMail: [lmartini@cisco.com](mailto:lmartini@cisco.com)

Nasser El-Aawar  
Level 3 Communications, LLC.  
1025 Eldorado Blvd.  
Broomfield, CO, 80021

EMail: [nna@level3.net](mailto:nna@level3.net)

Giles Heron  
Tellabs  
Abbey Place  
24-28 Easton Street  
High Wycombe  
Bucks  
HP11 1NT  
UK

EMail: [giles.heron@tellabs.com](mailto:giles.heron@tellabs.com)

Eric C. Rosen  
Cisco Systems, Inc.  
1414 Massachusetts Avenue  
Boxborough, MA 01719

EMail: [erosen@cisco.com](mailto:erosen@cisco.com)

Toby Smith  
Network Appliance, Inc.  
800 Cranberry Woods Drive  
Suite 300  
Cranberry Township, PA 16066

EMail: [tob@netapp.com](mailto:tob@netapp.com)



## Full Copyright Statement

Copyright (C) The Internet Society (2006).

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

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

## Intellectual Property

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

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

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

## Acknowledgement

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



