Pseudo-Wire Edge-to-Edge (PWE3) Working Group
Internet Draft
Document: <draft-ietf-pwe3-arch-00.txt>
Expires: April 2003

Stewart Bryant
Cisco Systems

Prayson Pate
Overture Networks, Inc.

Editors

October 2002

PWE3 Architecture

Status of this Memo

Abstract

This document describes an architecture for Pseudo Wire Emulation Edge-to-Edge (PWE3). It discusses the emulation of services (such as Frame Relay, ATM, Ethernet TDM and SONET/SDH) over packet switched networks (PSNs) using IP or MPLS. It presents the architectural framework for pseudo wires (PWs), defines terminology, specifies the various protocol elements and their functions.

Co-Authors

The following are co-authors of this document:

Thomas K. Johnson	Litchfield Communications
Kireeti Kompella	Juniper Networks, Inc.
Andrew G. Malis	Vivace Networks
Danny McPherson	TCB
Thomas D. Nadeau	Cisco Systems
Tricci So	Caspian Networks
W. Mark Townsley	Cisco Systems
Craig White	Level 3 Communications, LLC.
Lloyd Wood	Cisco Systems
XiPeng Xiao	Redback Networks

Table of Contents

1. Introduction

This document describes an architecture for Pseudo Wire Emulation Edge-to-Edge (PWE3) in support of [XIAO]. It discusses the emulation of services (such as Frame Relay, ATM, Ethernet TDM and SONET/SDH) over packet switched networks (PSNs) using IP or MPLS. It presents the architectural framework for pseudo wires (PWs), defines terminology, specifies the various protocol elements and their functions.

1.1 Pseudo Wire Definition

PWE3 is a mechanism that emulates the essential attributes of a service (such as a T1 leased line or Frame Relay) over a PSN. PWE3 is intended to provide only the minimum necessary functionality to emulate the wire with the required degree of faithfulness for the given service definition. Any required switching functionality is the responsibility of a forwarder function (FWRD). Any translation or other operation needing knowledge of the payload semantics is carried out by native service processing (NSP) elements. The functional definition of any FWRD or NSP elements is outside the scope of PWE3.

The required functions of PWs include encapsulating service-specific bit-streams, cells or PDUs arriving at an ingress port, and carrying them across a path or tunnel. In some cases it is necessary to perform other operation such as managing their timing and order, to emulate the behavior and characteristics of the service to the required degree of faithfulness.

From the perspective of a Customer Edge Equipment (CE), the PW is characterised as an unshared link or circuit of the chosen service. In some cases, there may be deficiencies in the PW emulation that impact the traffic carried over a PW, and hence limit the applicability of this technology. These limitations must be fully described in the appropriate service-specific documentation. It is possible that these limitations may lead to the definition of more than one PW emulation method, each providing a different degree of faithfulness.

1.2 PW Service Functionality

PWs provide the following functions in order to emulate the behavior and characteristics of the native service.

- Encapsulation of service-specific PDUs or circuit data arriving at the ingress port (logical or physical).
- Carriage of the encapsulated data across a PSN tunnel.
- Establishment of the PW including the exchange and/or distribution of the PW identifiers used by the PSN tunnel endpoints.
- Managing the signaling, timing, order or other aspects of the service at the boundaries of the PW.
- Service-specific status and alarm management.

1.3 Non-Goals of this document

The following are non-goals for this document:

- The on-the-wire specification of services encapsulation.
- The detailed definition of the protocols involved in PW setup and maintenance.

The following are outside the scope of PWE3:

- Any multicast service not native to the emulated medium. Thus, Ethernet transmission to a "multicast" IEEE-48 address is in scope, while multicast services like MARS [RFC2022] that are implemented on top of the medium are out of scope.
- Methods to signal or control the underlying PSN.

1.4 Terminology

Editor's note: Although it was decided at IETF-54 that the PWE3 common terminology should be published in a separate document, there is case for it remaining in the architecture document. If the PWE3-WG confirms the desire to have a separate document, we will remove this section in the next revision.

This document uses the following definitions of terms. These terms are illustrated in context in Figure 2.

Attachment Circuit (AC)	The circuit or virtual circuit attaching a CE to a PE.
Applicability Statement (AS)	Each PW service will have an Applicability Statement (AS) that describes the applicability of PWs for that service.

CE-bound	The traffic direction where PW-PDUs are received on a PW via the PSN, processed and then sent to the destination CE.
CE Signaling	Messages sent and received by the CEs control plane. It may be desirable or even necessary for the PE to participate in or monitor this signaling in order

	to effectively emulate the service.
Customer Edge (CE)	A device where one end of a service originates and/or terminates. The CE is not aware that it is using an emulated service rather than a native service.
Forwarder (FWRD)	A PE subsystem that selects the PW to use to transmit a payload received on an AC.
Fragmentation	The action of dividing a single PDU into multiple PDUs before transmission with the intent of the original PDU being reassembled elsewhere in the network. Fragmentation may be performed in order to allow sending of packets of a larger size than the network MTU which they will traverse.
Maximum transmission unit (MTU)	The packet size (excluding data link header) that an interface can be transmit without needing to fragment.
Native Service Processing (NSP)	Processing of the data received by the PE from the CE before presentation to the PW for transmission across the core.
Packet Switched Network (PSN)	Within the context of PWE3, this is a network using IP or MPLS as the mechanism for packet forwarding.
Protocol Data Unit (PDU)	The unit of data output to, or received from, the network by a protocol layer.
Provider Edge (PE)	A device that provides PWE3 to a CE.
PE-bound	The traffic direction where information from a CE is adapted to a PW, and PW-PDUs are sent into the PSN.
PE/PW Maintenance	Used by the PEs to set up, maintain and

	tear down the PW. It may be coupled with

	CE Signaling in order to effectively manage the PW.
Pseudo Wire (PW)	A mechanism that carries the essential elements of an emulated service from one PE to one or more other PEs over a PSN.
PW End Service (PWES)	The interface between a PE and a CE. This can be a physical interface like a T1 or Ethernet, or a virtual interface like a VC or VLAN.
Pseudo Wire Emulation Edge to Edge (PWE3)	A mechanism that emulates the essential attributes of service (such as a T1 leased line or frame relay) over a PSN.
Pseudo Wire PDU (PW-PDU)	A PDU sent on the PW that contains all of the data and control information necessary to emulate the desired service.
PSN Tunnel	A tunnel across a PSN inside which one or more PWs can be carried.
PSN Tunnel Signaling	Used to set up, maintain and tear down the underlying PSN tunnel.
PW Demultiplexer	Data-plane method of identifying a PW terminating at a PE.
Time Domain Multiplexing (TDM)	Synchronous bit-streams at rates defined by G.702.
Tunnel	A method of transparently carrying information over a network.

2. PWE3 Applicability

The PSN carrying a PW will subject payload packets to delay, jitter, network transients and re-ordering. The applicability of PWE3 to a particular service depends on the sensitivity of that service to these effects, and the ability of the adaptation layer to mask them. Some services, such as IP over FR over PWE3, may prove quite resilient to IP and MPLS PSN characteristics. Other services, such as the interconnection of PBX systems via PWE3, will require more careful consideration of the PSN and adaptation layer characteristics. In some instances, traffic engineering of the underlying PSN will be required, and in some cases the constraints may be such that it is not possible to provide the required service guarantees.

3. Protocol Layering Model

The PWE3 protocol-layering model is intended to minimise the differences between PWs operating over different PSN types. The design of the protocol-layering model has the goals of making each PW definition independent of the underlying PSN, and maximizing the reuse of IETF protocol definitions and their implementations.

3.1 Protocol Layers

The logical protocol-layering model required to support a PW is shown in Figure 1.

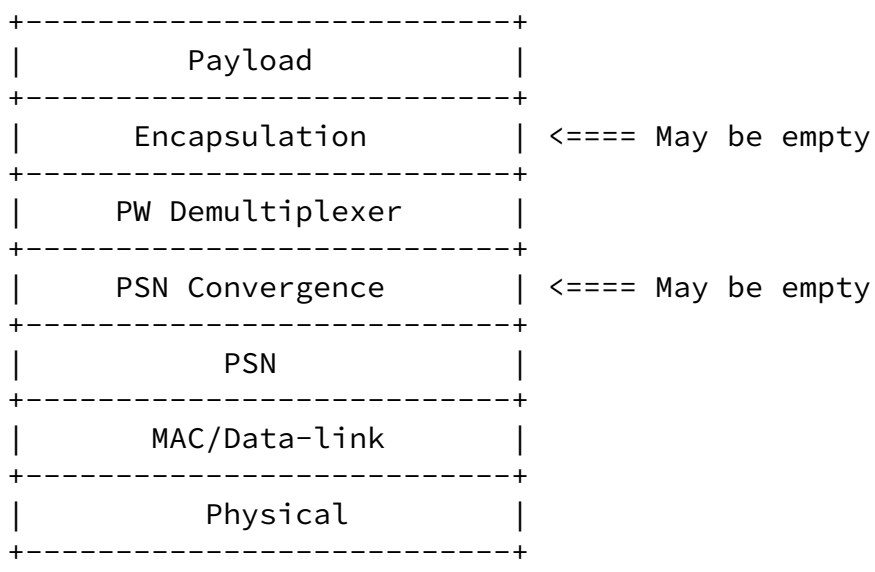


Figure 1: Logical Protocol Layering Model

The payload is transported over the Encapsulation Layer. The Encapsulation Layer carries any information, not already present within the payload itself, that is needed by the PW CE-bound PE interface to send the payload to the CE via the physical interface. If no information is needed beyond that in the payload itself, this layer is empty.

This layer also provides support for real-time processing, and sequencing, if needed.

The PW Demultiplexer Layer provides the ability to deliver multiple PWs over a single PSN tunnel. The PW demultiplexer value used to identify the PW in the data-plane may be unique per PE, but this is not a PWE3 requirement. It must, however, be unique per tunnel endpoint. If it is necessary to identify a particular tunnel, then that is the responsibility of the PSN layer.

The PSN Convergence Layer provides the enhancements needed to make the PSN conform to the assumed PSN service requirement. This layer therefore provides a consistent interface to the PW, making the PW independent of the PSN type. If the PSN already meets the service requirements, this layer is empty.

The PSN header, MAC/Data-link and Physical Layer definitions are outside the scope of this document. The PSN can be IPv4, IPv6 or MPLS.

3.2 Domain of PWE3

PWE3 defines the Encapsulation Layer, the method of carrying various payload types, and the interface to the PW Demultiplexer Layer. It is expected that the other layers will be provided by tunneling methods such as L2TP or MPLS over the PSN.

3.3 Payload Types

The payload is classified into the following generic types of native data unit:

- o Packet
- o Cell
- o Bit-stream
- o Structured bit-stream

Within these generic types there are specific service types. For example:

Generic Payload Type	PW Service
Packet	Ethernet (all types), HDLC, frame-relay, ATM AAL5 PDU.
Cell	ATM.
Bit-stream	SONET, TDM (e.g. DS1, DS3, E1).
Structured bit-stream	SONET, TDM.

3.3.1. Packet Payload

A packet payload is a variable-size data unit presented to the PE on the AC. A packet payload may be large compared to the PSN MTU. The delineation of the packet boundaries is encapsulation-specific. HDLC or Ethernet PDUs can be considered as examples of packet payloads. Typically a packet will be stripped of transmission overhead such as HDLC flags and stuffing bits before transmission over the PW.

A packet payload would normally be relayed across the PW as a single unit. However, there will be cases where the combined size of the packet payload and its associated PWE3 and PSN headers exceeds the PSN path MTU. In these cases, some fragmentation methodology needs to be applied. This may, for example, be the case when a user is providing the service and attaching to the service provider via an Ethernet, or where nested pseudo-wires are involved. Fragmentation is discussed in more detail in Section 5.

A packet payload may need sequencing and real-time support.

In some situations, the packet payload may be selected from the

packets presented on the emulated wire on the basis of some sub-multiplexing technique. For example, one or more frame-relay PDUs may be selected for transport over a particular pseudo-wire based on the frame-relay Data-Link Connection Identifier (DLCI), or, in the case of Ethernet payloads, on the basis of the VLAN identifier. This is an FWRD function, and this selection would therefore be made before the packet was presented to the PW Encapsulation Layer.

3.3.2. Cell Payload

A cell payload is created by capturing, transporting and replaying groups of bits presented on the wire in a fixed-size format. The delineation of the group of bits that comprise the cell is specific to the encapsulation type. Two common examples of cell payloads are 53-octet cells carrying ATM AAL2, and the larger 188-octet MPEG Transport Stream packets [ETSI].

To reduce per-PSN packet overhead, multiple cells may be concatenated into a single payload. The Encapsulation Layer may consider the payload complete on the expiry of a timer, or after a fixed number of cells have been received. The benefit of concatenating multiple PDUs should be weighed against the resulting increase in jitter and the larger penalty incurred by packet loss. In some cases, it may be appropriate for the Encapsulation Layer to perform a silence suppression or a similar compression.

The generic cell payload service will normally need sequence number support, and may also need real-time support. The generic cell payload service would not normally require fragmentation.

The Encapsulation Layer may apply some form of compression to some of these sub-types.

In some instances, the cells to be incorporated in the payload may be selected by filtering them from the stream of cells presented on the wire. For example, an ATM PWE3 service may select cells based on their VCI or VPI fields. This is an NSP function, and the selection would therefore be made before the packet was presented to the PW Encapsulation Layer.

3.3.3. Bit-stream

A bit-stream payload is created by capturing, transporting and replaying the bit pattern on the emulated wire, without taking advantage of any structure that, on inspection, may be visible within the relayed traffic (i.e. the internal structure has no effect on the fragmentation into packets). The Encapsulation Layer submits an identical number of bits for transport in each PW-PDU.

This service will require sequencing and real-time support.

3.3.4. Structured bit-stream

A bit-stream payload is created by using some knowledge of the underlying structure of the bit-stream to capture, transport and replay the bit pattern on the emulated wire (i.e. the internal structure directly effects the fragmentation into packets).

Two important points distinguish structured and unstructured bit-streams:

- Some part of the original (unstructured) bit-stream are stripped by, for example, the PSN-bound direction of the NSP block. For example, in Structured SONET the section and line overhead (and, possibly, more) may be stripped.

- The PW must preserve the structure across the PSN so that the CE-bound NSP block can insert it correctly into the reconstructed unstructured bit-stream.

The Encapsulation Layer may also perform silence/idle suppression or similar compression on a structured bit-stream.

Structured bit-streams are distinguished from cells in that the structures may be too long to be carried in a single packet (i.e. structured SONET). Note that "short" structures are indistinguishable from cells and may benefit from the use of cell encapsulations.

This service will require sequencing and real-time support.

3.3.5. Principle of Minimum Intervention

To minimise the scope of information, and to improve the efficiency

of data flow through the Encapsulation Layer, the payload should be transported as received with as few modifications as possible [RFC1958].

This minimum intervention approach decouples payload development from PW development and requires fewer translations at the NSP in a system with similar CE interfaces at each end. It also prevents any unwanted side-effects due to subtle mis-representation of the payload in the intermediate format.

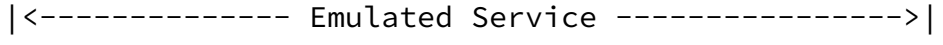
An intervention approach can be more wire-efficient in some cases and may result in fewer translations at the NSP where the CE interfaces are of different types. Any intermediate format effectively becomes a new framing type, requiring documentation and assured interoperability. This increases the amount of work for handling the protocol the intermediate format carries, and is undesirable.

4. Architecture of Pseudo-wires

This section describes the PWE3 architectural model.

4.1 Network Reference Model

Figure 2 illustrates the network reference model for point-to-point PWs.



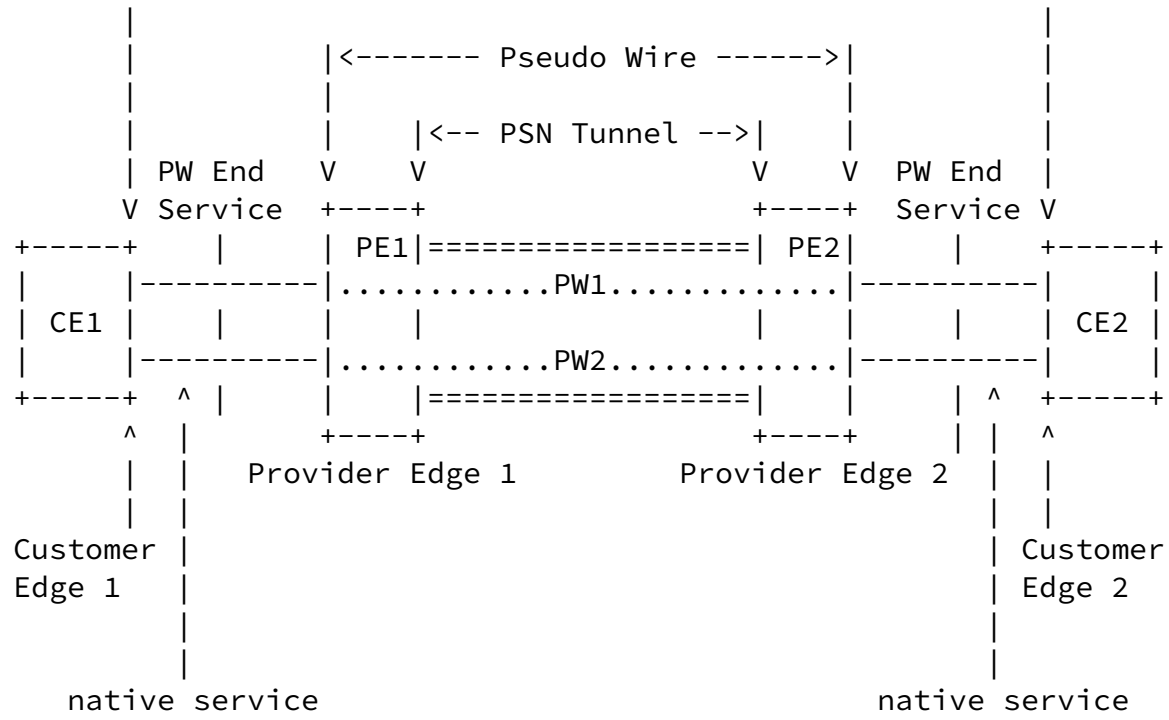


Figure 2: PWE3 Network Reference Model

The two PEs (PE1 and PE2) need to provide one or more PWs on behalf of their client CEs (CE1 and CE2) to enable the client CEs to communicate over the PSN. A PSN tunnel is established to provide a data path for the PW. The PW traffic is invisible to the core network, and the core network is transparent to the CEs. Native data units (bits, cells or packets) presented at the PW End Service (PWES) are encapsulated in a PW-PDU and carried across the underlying network via the PSN tunnel. The PEs perform the necesssary encapsulation and decapsulation of PW-PDUs, as well as handling any other functions required by the PW service, such as sequencing or timing.

There are situations in which a particular packet payload needs to be multicast so that it is received by a number of CEs. This is useful when using PWs as part of a "virtual LAN" service (see, e.g.,

[VPLS]). This can be achieved by replicating the payload and transmitting the replicas on PWs, but it may also be useful to have a type of PW which is inherently point-to-multipoint. In that case, the PW would need to be carried through a point-to-multipoint PSN

tunnel, employing a multicast mechanism provided by the PSN.

4.2 PWE3 Pre-processing

In some applications, there is a need to perform operations on the native data units received from the CE (including both payload and signaling traffic) before they are transmitted across the PW by the PE. Examples include Ethernet bridging, SONET cross-connect, translation of locally-significant identifiers such as VCI/VPI, or translation to another service type. These operations could be carried out in external equipment, and the processed data sent to the PE over one or more physical interfaces. In most cases, there are cost and operational benefits in undertaking these operations within the PE. This processed data is then presented to the PW via a virtual interface within the PE.

These pre-processing operations are included in the PWE3 reference model to provide a common reference point, but the detailed description of these operations is outside the scope of the PW definition given here.

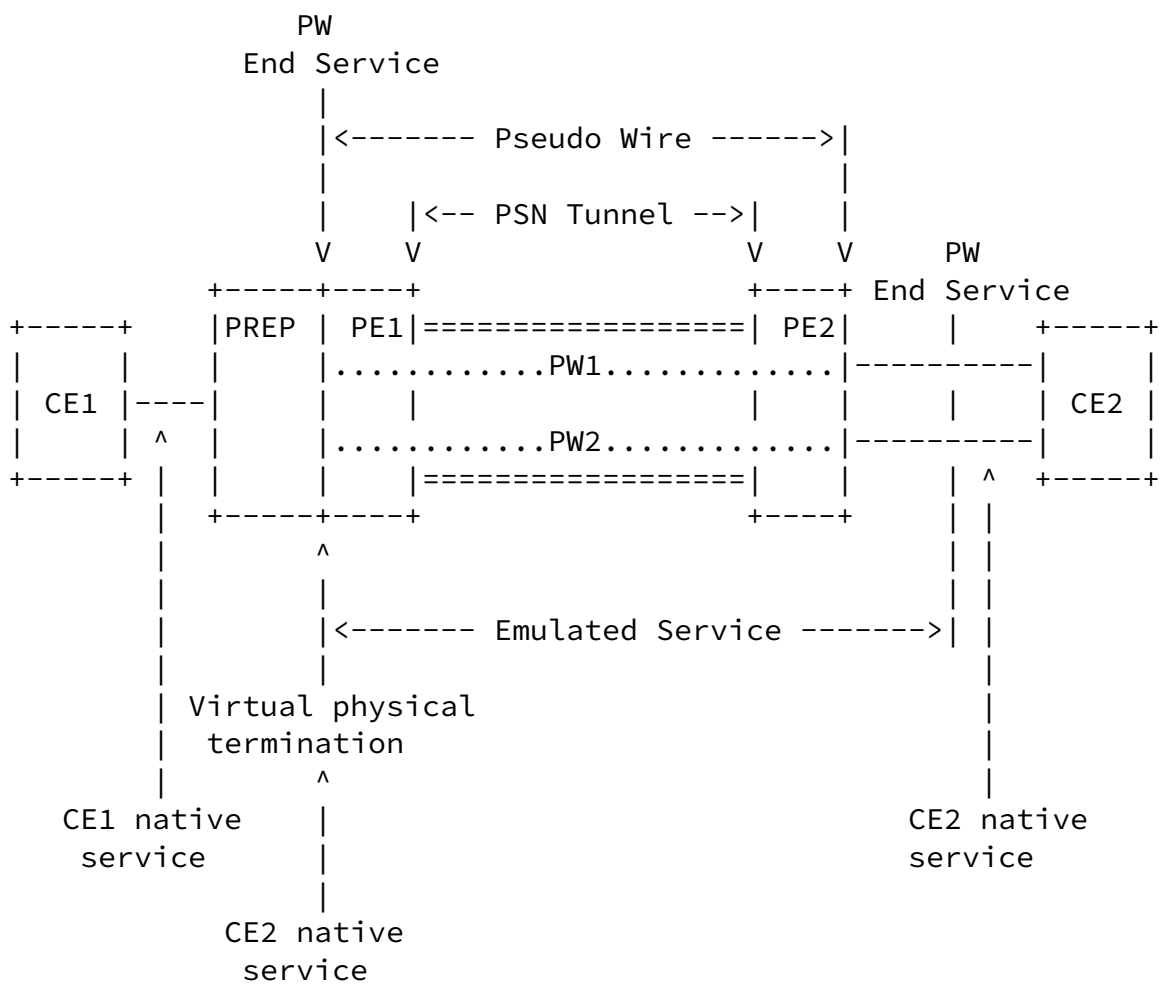


Figure 3: Pre-processing within the PWE3 Network Reference Model

Figure 3 shows the inter-working of one PE with pre-processing (PREP), and a second without this functionality. This is a useful reference point because it emphasises that the functional interface between PREP and the PW is that represented by a physical interface carrying the service. This effectively defines the necessary inter-working specification.

The operation of a system in which both PEs include PREP functionality is also supported.

The required pre-processing can be divided into two components:

- o Forwarder (FWRD)
- o Native Service Processing (NSP)

4.2.1. Forwarders

In some applications there is the need to selectively forward payload

elements from one of more ACs to one or more PWs. In such cases there will also be the need to perform the inverse function on PWE3-PDUs received by a PE from the PSN. This is the function of the FWRD.

The FWRD selects the PW based on, for example: the incoming AC, the contents of the payload, or some statically- or dynamically-configured forwarding information.

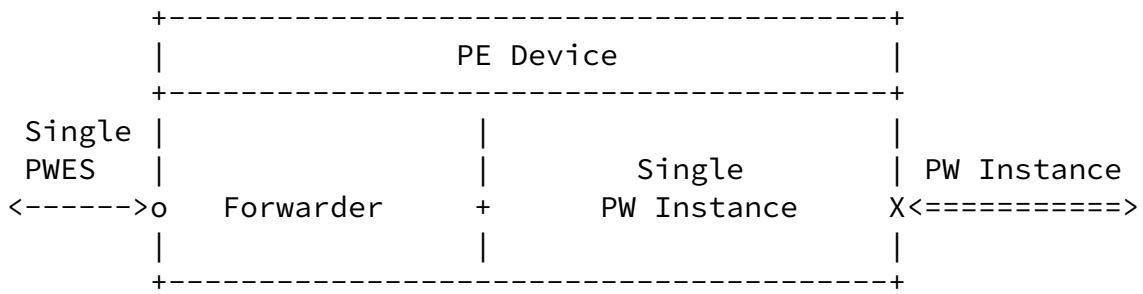


Figure 4a: Simple point-to-point service

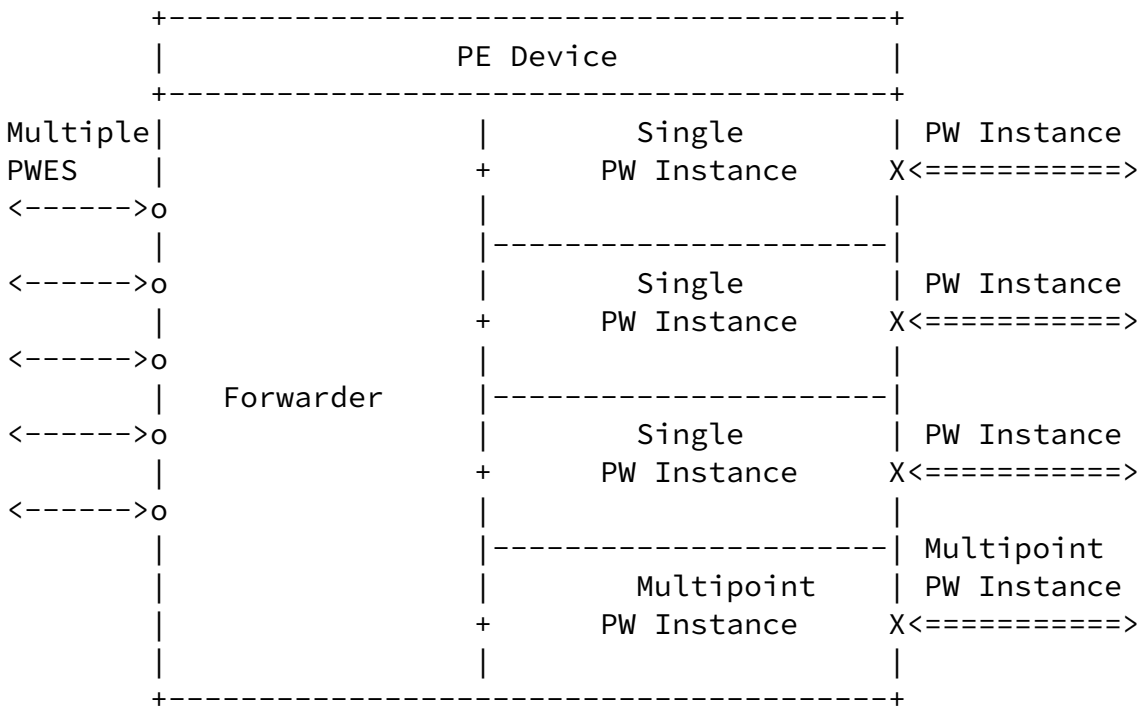


Figure 4b: Multiple PWEs to Multiple PW Forwarding

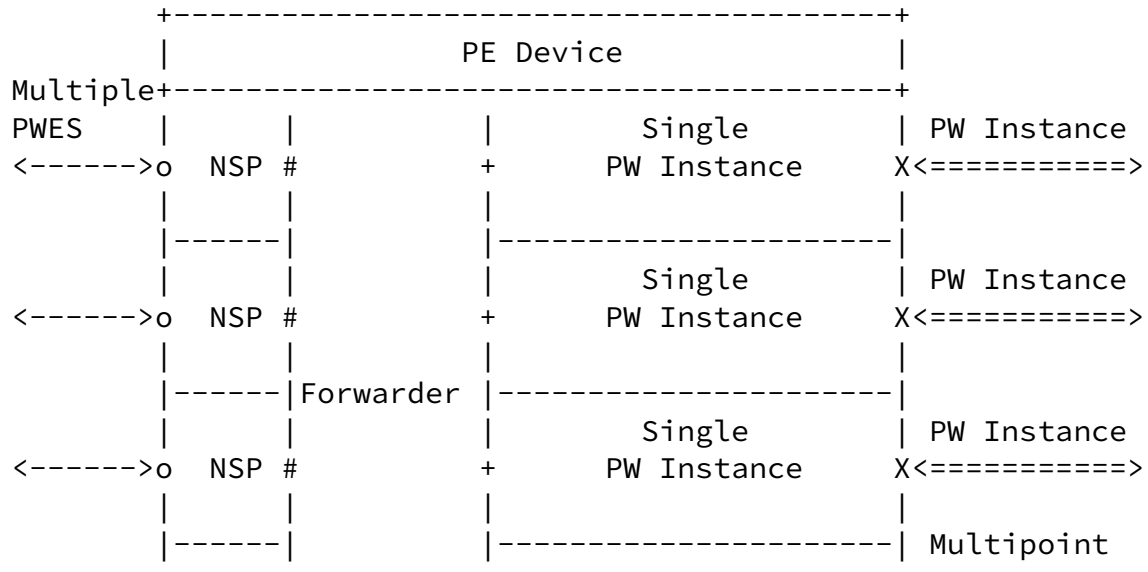
Figure 4a shows a simple FWRD that performs some type of filtering operation. Because the FWRD has a single input and a single output interface, filtering is the only type of forwarding operation that applies. Figure 4b shows a more general forwarding situation where payloads are extracted from one or more PWESs and directed to one or more PWs, including, in this instance, a multipoint PW. In this case

both filtering and direction operations may be performed on the payloads.

4.2.2. Native Service Processing

In some applications some form of data or address translation, or other operation requiring knowledge of the semantics of the payload, will be required. This is the function of the Native Service Processor (NSP).

The use of the NSP approach simplifies the design of the PW by restricting a PW to homogeneous operation. NSP is included in the reference model to provide a defined interface to this functionality. The specification of the various types of NSP is outside the scope of PWE3.



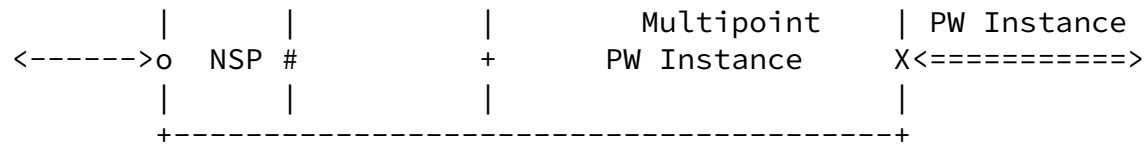


Figure 5: NSP in a Multiple PWEs to Multiple PW Forwarding PE

Figure 5 illustrates the relationship between NSP, FWRD and PWs in a PE. The NSP function may apply any transformation operation (modification, injection, etc.) on the payloads as they pass between the physical interface to the CE and the virtual interface to the FWRD. A PE device may contain more than one FWRD.

This model also supports the operation of a system in which the NSP functionality includes terminating the data-link and applying Network Layer processing to the payload is also supported.

4.3 Maintenance Reference Model

Figure 6 illustrates the maintenance reference model for PWs.

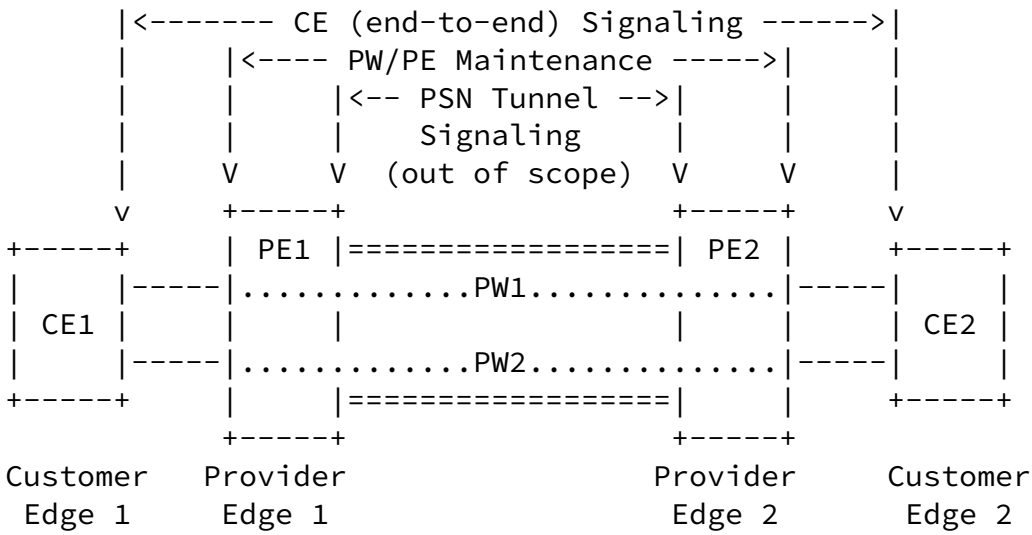


Figure 6: PWE3 Maintenance Reference Model

The following signaling mechanisms are required:

- The CE (end-to-end) signaling is between the CEs. This signaling could be frame relay PVC status signaling, ATM SVC signaling, etc.

- The PW/PE Maintenance is used between the PEs (or NSPs) to set up, maintain and tear down PWs, including any required coordination of parameters.

- The PSN Tunnel signaling controls the PW multiplexing and some elements of the underlying PSN. Examples are L2TP control protocol, MPLS LDP and RSVP-TE. The definition of the information that PWE3 needs to be signaled is within the scope of PWE3, but the signaling protocol itself is not.

4.4 Protocol Stack Reference Model

Figure 7 illustrates the protocol stack reference model for PWs.

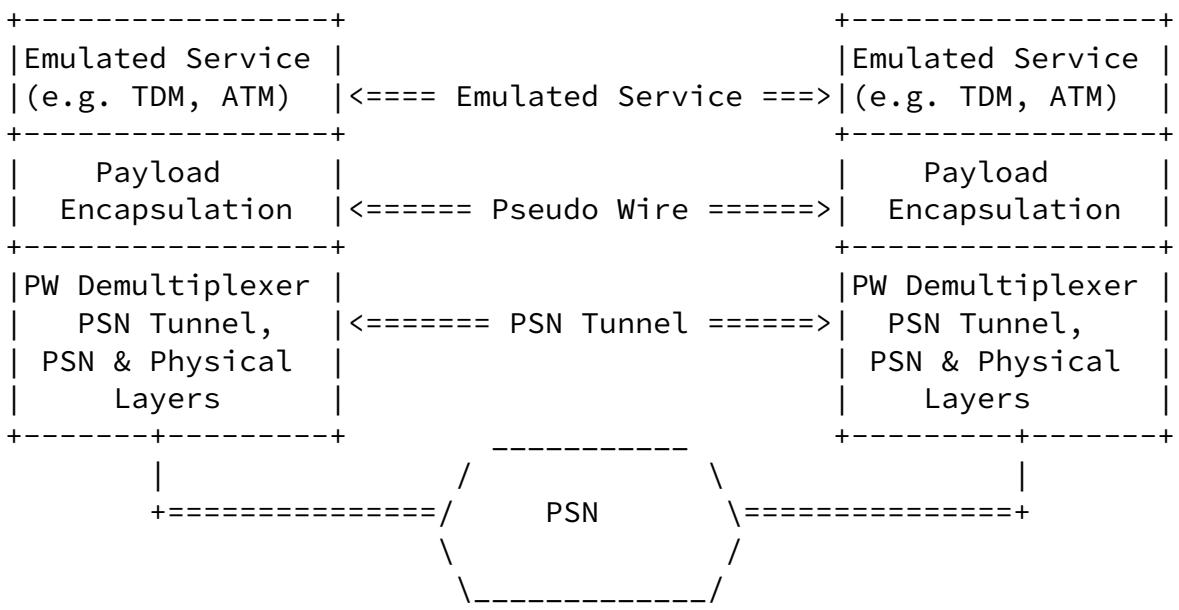
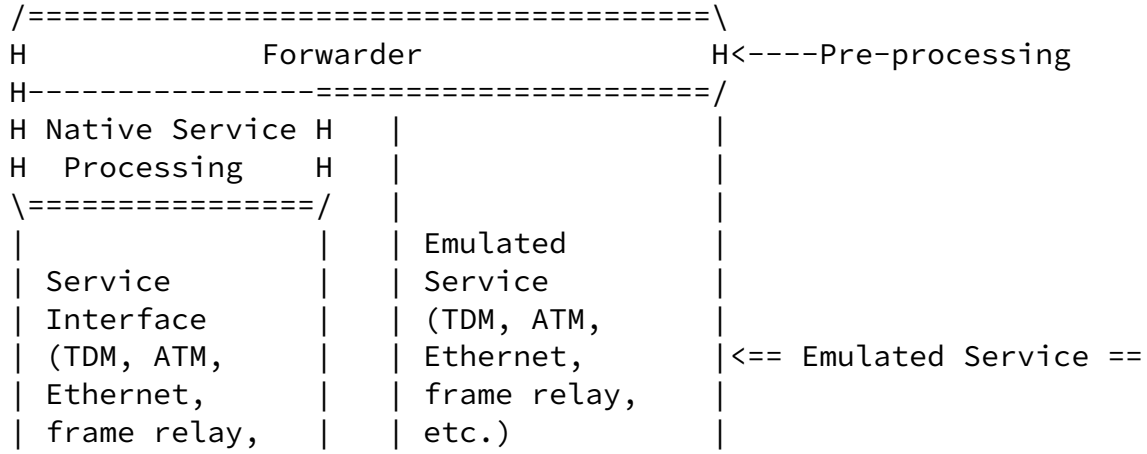


Figure 7: PWE3 Protocol Stack Reference Model

The PW provides the CE with an emulated physical or virtual connection to its peer at the far end. Native service PDUs from the CE are passed through an Encapsulation Layer at the sending PE, and then sent over the PSN. The receiving PE removes the encapsulation and restores the payload to its native format for transmission to the destination CE.

4.5 Pre-processing Extension to Protocol Stack Reference Model

Figure 8 illustrates how the protocol stack reference model is extended to include the provision of pre-processing (Forwarding and NSP). This shows the ideal placement of the physical interface relative to the CE.



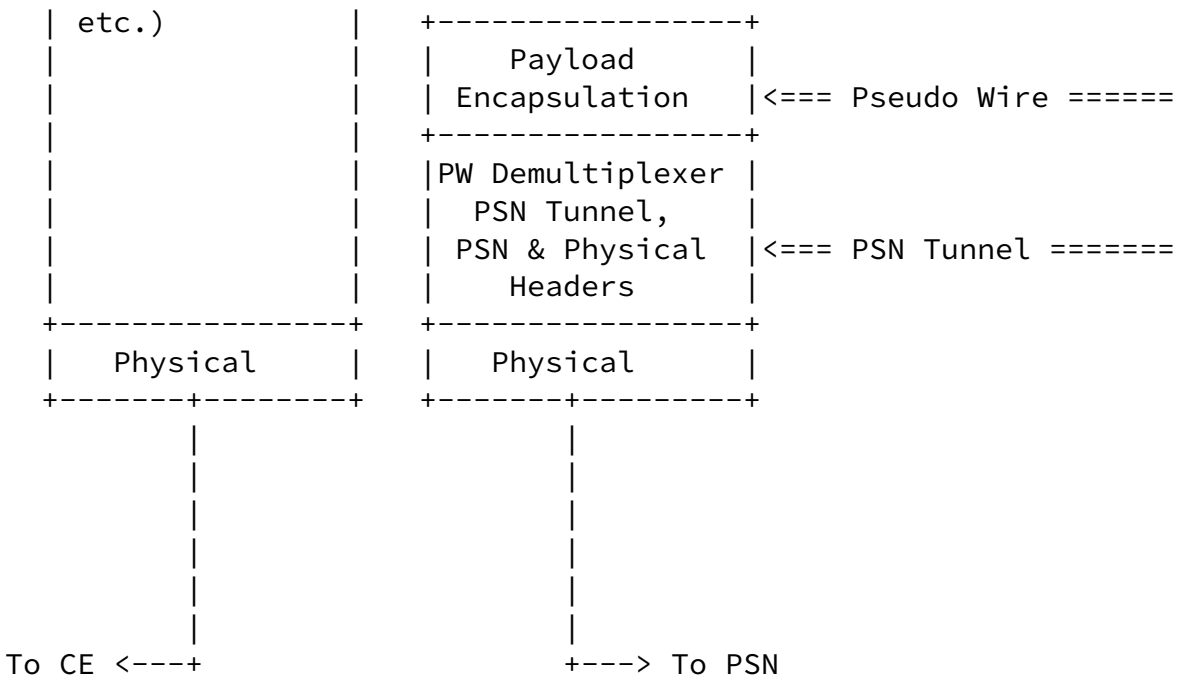


Figure 8: Protocol Stack Reference Model with Pre-processing

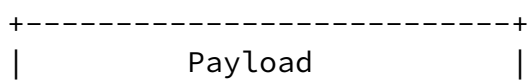
5. PW Encapsulation

The PW Encapsulation Layer provides the necessary infrastructure to adapt the specific payload type being transported over the PW to the PW Demultiplexer Layer that is used to carry the PW over the PSN.

The PW Encapsulation Layer consists of three sub-layers:

- o Payload Convergence
- o Timing
- o Sequencing

The PW Encapsulation sub-layering and its context with the protocol stack are shown, in Figure 9.



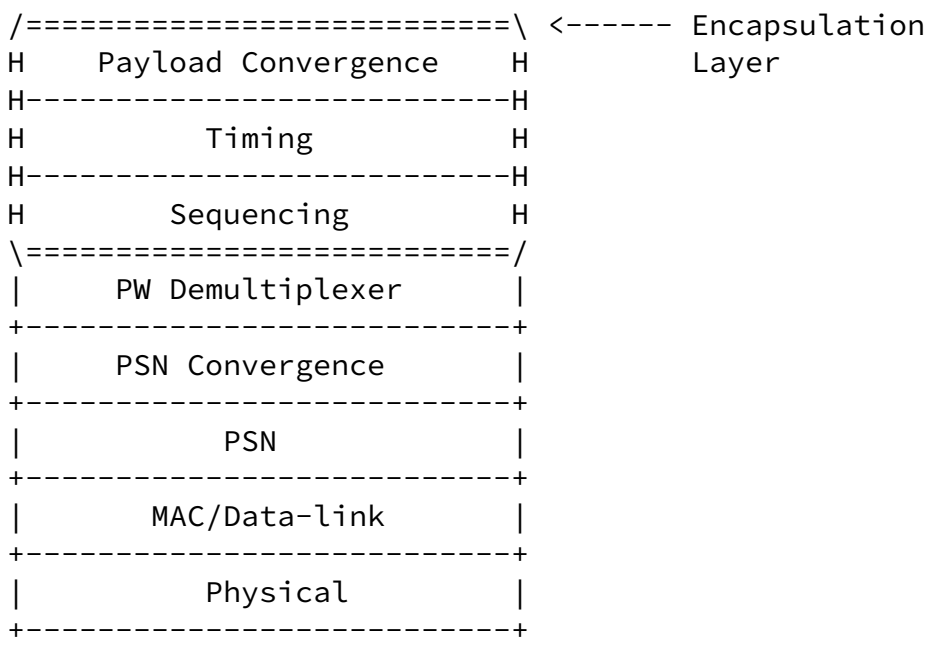


Figure 9: PWE3 Encapsulation Layer in Context

The Payload Convergence Sub-layer is highly tailored to the specific payload type, but, by grouping a number of target payload types into a generic class, and then providing a single convergence sub-layer type common to the group, we achieve a reduction in the number of payload convergence sub-layer types. This decreases implementation complexity. The provision of per-packet signaling and other out-of-band information (other than sequencing or timing) is undertaken by this layer.

The Timing Layer and the Sequencing Layer provide generic services to the Payload Convergence Layer for all payload types, when required.

5.1 Payload Convergence Layer

5.1.1. Encapsulation

The primary task of the Payload Convergence Layer is the encapsulation of the payload in PW-PDUs. The native data units to be encapsulated may or may not contain L2 or L1 header information. This is service specific. The Payload Convergence header carries the additional information needed to replay the native data units at the CE-bound physical interface. The PW Demultiplexer header is not considered as part of the PW header.

Not all the additional information needed to replay the native data units need to be carried in the PW header of the PW PDUs. Some information (e.g. service type of a PW) may be stored as state information at the destination PE during PW set-up.

5.1.2. PWE3 Channel Types

The PW Encapsulation Layer and its associated signaling require one or more of the following types of channels from its underlying PW Demultiplexer and PSN Layers:

1. A reliable control channel for signaling line events, status indications, and, in some exceptional cases, CE-CE events that must be translated and sent reliably between PEs.

   For example, this capability is needed in [PPPoL2TP] (PPP negotiation has to be split between the two ends of the tunnel). PWE3 may also need this type of control channel to provide faithful emulation of complex data-link protocols.

plus one or more data channels with the following characteristics:

2. A high-priority, unreliable, sequenced channel. A typical use is for CE-to-CE signaling. "High priority" may simply be indicated via DSCP/EXP bits for priority during transit. This channel type could also use a bit in the tunnel header itself to indicate that packets received at the PE should be processed with higher priority [RFC2474].

3. A sequenced channel for data traffic that is sensitive to packet reordering (one classification for use could be for any non-IP traffic).

4. An un-sequenced channel for data traffic insensitive to packet order.

The data channels (2, 3 and 4 above) should be carried "in band" with one another to as much of a degree as is reasonably possible on a PSN.

Where end-to-end connectivity may be disrupted by address translation [RFC3022], access-control lists, firewalls etc., there exists the possibility that the control channel may be able to pass traffic and set up the PW, but the PW data-path data traffic is blocked by one or more of these mechanisms. In these cases unless the control channel is also carried "in band" the signaling to set-up the PW will not confirm the existence of an end-to-end data path.

In some cases there is a need to synchronize some CE events with the data carried over a PW. This is especially the case with TDM circuits (e.g., on-hook/off-hook events in PSTN switches).

PWE3 channel types that are not needed by the supported PWs need not be included in such an implementation.

5.1.3. Quality of Service Considerations

Where possible, it is desirable to employ mechanisms to provide PW Quality of Service (QoS) support over PSNs.

5.2 Payload-independent PW Encapsulation Layers

Two PWE3 Encapsulation Sub-layers provide common services to all payload types: Sequencing and Timing. These services are optional and are only used if needed by a particular PW instance. If the service is not needed, the associated header may be omitted in order to conserve processing and network resources.

There will be instances where a specific payload type will be required to be transported with or without sequence and/or real-time support. For example, an invariant of frame relay transport is the preservation of packet order. Some frame-relay applications expect in-order delivery, and may not cope with reordering of the frames. However, where the frame relay service is itself only being used to carry IP, it may be desirable to relax that constraint in return for reduced per-packet processing cost.

The guiding principle is that, where possible, an existing IETF protocol should be used to provide these services. Where a suitable protocol is not available, the existing protocol should be extended or modified to meet the PWE3 requirements, thereby making that protocol available for other IETF uses. In the particular case of timing, more than one general method may be necessary to provide for the full scope of payload timing requirements.

5.2.1. Sequencing

The sequencing function provides three services: frame ordering, frame duplication detection and frame loss detection. These services allow the invariant properties of a physical wire to be emulated.

Support for sequencing depends on the payload type, and may be omitted if not needed.

The size of the sequence-number space depends on the speed of the emulated service, and the maximum time of the transient conditions in the PSN. A sequence number space greater than approximately 2^16 may

therefore be needed to prevent the sequence number space wrapping during the transient.

5.2.1.1 Frame Ordering

When packets carrying the PW-PDUs traverse a PSN, they may arrive out of order at the destination PE. For some services, the frames (control frames, data frames, or both control and data frames) must be delivered in order. For such services, some mechanism must be provided for ensuring in-order delivery. Providing a sequence number in the sequence sub-layer header for each packet is one possible approach to out-of-sequence detection. Alternatively it can be noted that sequencing is a subset of the problem of delivering timed packets, and that a single combined mechanism such as [RFC1889] may be employed.

There are two possible misordering strategies:

- Drop misordered PW PDUs.
- Try to sort PW PDUs into the correct order.

The choice of strategy will depend on:

- How critical the loss of packets is to the operation of the PW (e.g. the acceptable bit error rate).
- The speeds of the PW and PSN.
- The acceptable delay (since delay must be introduced to reorder)
- The incidence of expected misordering.

5.2.1.2 Frame Duplication Detection

In rare cases, packets traversing a PW may be duplicated by the underlying PSN. For some services, frame duplication is not acceptable. For such services, some mechanism must be provided to ensure that duplicated frames will not be delivered to the destination CE. The mechanism may or may not be the same as the mechanism used to ensure in-order frame delivery.

5.2.1.3 Frame Loss Detection

A destination PE can determine whether a frame has been lost by tracking the sequence numbers of the received PW PDUs.

In some instances, a destination PE will have to presume that a PW PDU is lost if it fails to arrive within a certain time. If a PW-PDU that has been processed as lost subsequently arrives, the destination PE must discard it.

5.2.2. Timing

A number of native services have timing expectations based on the characteristics of the networks that they were designed to travel over, and it can be necessary for the emulated service to duplicate these network characteristics as closely as possible, e.g. in delivering native traffic with the same jitter, bit-rate and timing characteristics as it was sent.

In such cases, it is necessary for the receiving PE to play out the native traffic as it was received at the sending PE. This relies on either timing information sent between the two PEs, or in some case timing information received from an external reference.

The Timing Sub-layer must therefore support two timing functions: clock recovery and timed payload delivery. A particular payload type may require either or both of these services.

5.2.2.1 Clock Recovery

Clock recovery is the extraction of output transmission bit timing information from the delivered packet stream, and requires a phase-locking mechanism. A physical wire provides this naturally, but it

is a relatively complex task to extract this from a highly jittered source such as packet stream. It is therefore desirable that an existing real-time protocol such as [RFC1889] be used for this purpose, unless it can be shown that this is unsuitable or unnecessary for a particular payload type.

5.2.2.2 Timed delivery

Timed delivery is the delivery of non-contiguous PW PDUs to the PW output interface with a constant phase relative to the input interface. The timing of the delivery may be relative to a clock derived from the packet stream via clock recovery, or via an external clock.

5.3 Fragmentation

A payload would normally be relayed across the PW as a single unit. However, there will be cases where the combined size of the payload and its associated PWE3 and PSN headers exceeds the PSN path MTU. When a packet exceeds the MTU of a given network, fragmentation and

reassembly may have to be performed in order for the packet to be delivered. Since fragmentation and reassembly generally consume a large amount of network resource as compared to simply switching a packet in its entirety, efforts should be made to reduce or eliminate the need for fragmentation and reassembly as much as possible throughout a network. Of particular concern for fragmentation and reassembly are aggregation points where large numbers of pseudowires are processed (e.g. at the PE).

Ideally, the equipment originating the traffic being sent over the PW will be configured to have adaptive measures (e.g. [RFC1191], [RFC1981]) in place such that it never sends a packet which must be fragmented. When this fails, the point closest to the sending host with fragmentation and reassembly capabilities should attempt to reduce the size of packets further into the network. Thus, in the reference model for PWE3 [Figure 3] fragmentation should first be performed at the CE if at all possible. If and only if the CE cannot adhere to an acceptable MTU size for the PW should the PE attempt its own fragmentation method.

In cases where MTU management fails to limit the payload to a size

suitable for transmission of the PW, the PE may fall back to either a generic PW fragmentation method, or, if available the fragmentation service of the underlying PSN.

It is acceptable for a PE implementation to not support fragmentation. A PE that does not support fragmentation will drop packets that exceed the PSN MTU, and the management plane of the encapsulating PE may be notified.

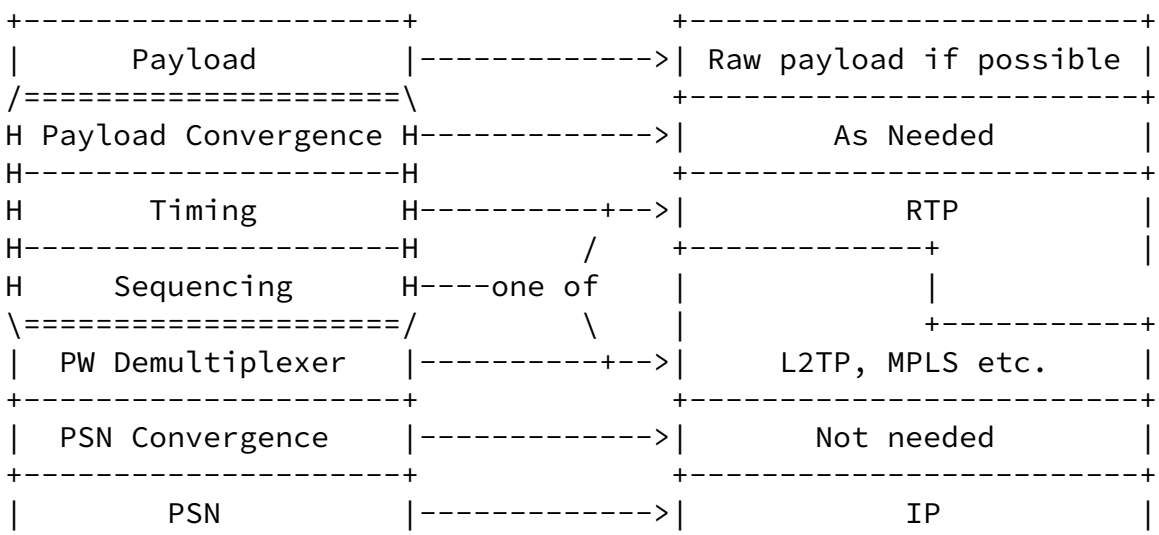
If the length of a L2/L1 frame, restored from a PW PDU, exceeds the MTU of the destination PWES, it must be dropped. In this case, the management plane of the destination PE may be notified.

5.4 Instantiation of the Protocol Layers

This document does not address the detailed mapping of the Protocol Layering model to existing or future IETF standards. The instantiation of the logical Protocol Layering model is shown in Figure 9.

5.4.1. PWE3 over an IP PSN

The protocol definition of PWE3 over an IP PSN should employ existing IETF protocols where possible.



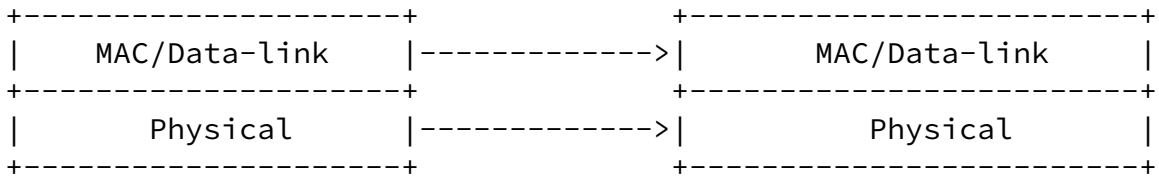


Figure 10: PWE3 over an IP PSN

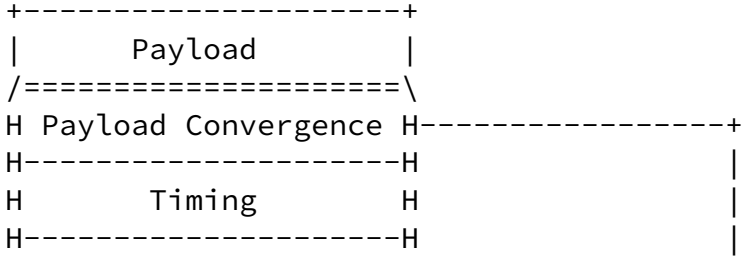
Figure 10 shows the protocol layering for PWE3 over an IP PSN. As a rule, the payload should be carried as received from the NSP, with the Payload Convergence Layer provided when needed. (It is accepted that there may sometimes be good reason not to follow this rule, but the exceptional circumstances need to be documented in the Encapsulation Layer definition for that payload type).

Where appropriate, timing is provided by RTP [RFC1889], which when used also provides a sequencing service. PW Demultiplexing may be provided by a number of existing IETF tunnel protocols. Some of these tunnel protocols provide an optional sequencing service. (Sequencing is provided either by RTP, or by the PW Demultiplexer Layer, but not both). A PSN Convergence Layer is not needed, because all the tunnel protocols shown above are designed to operate directly over an IP PSN.

As a special case, if the PW Demultiplexer is an MPLS label, the protocol architecture of section 5.4.2 can be used instead of the protocol architecture of this section.

5.4.2. PWE3 over an MPLS PSN

The MPLS ethos places importance on wire efficiency. By using a control word, some components of the PWE3 protocol layers can be compressed to increase this efficiency.



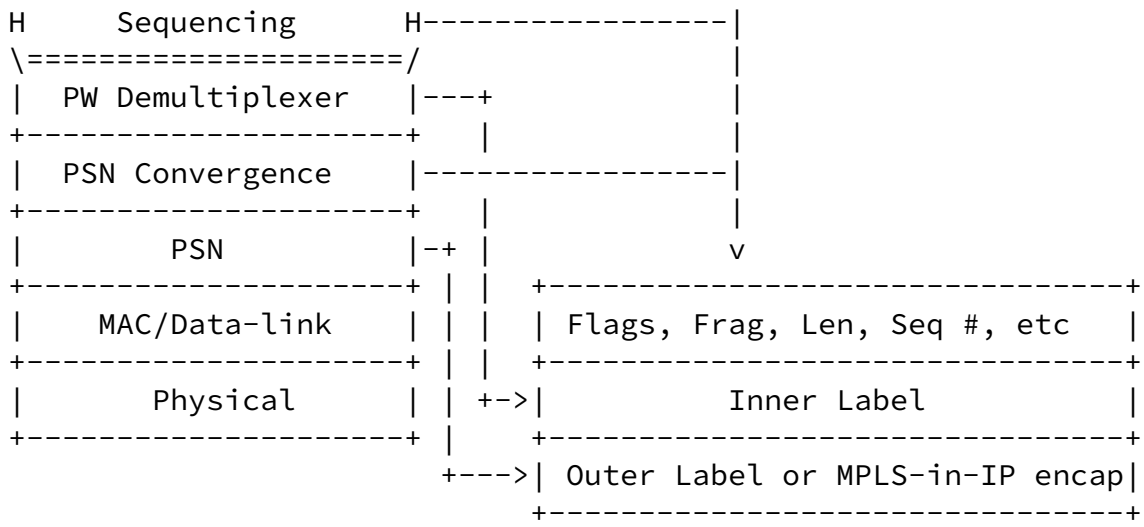


Figure 11: PWE3 over an MPLS PSN using a control word

Figure 11 shows the protocol layering for PWE3 over an MPLS PSN. An inner MPLS label is used to provide the PW demultiplexing function. A control word is used to carry most of the information needed by the PWE3 Encapsulation Layer and the PSN Convergence Layer in a compact format. The flags in the control word provide the necessary payload convergence. A sequence field provides support for both in-order payload delivery and (supported by a fragmentation control method) a PSN fragmentation service within the PSN Convergence Layer. Ethernet pads all frames to a minimum size of 64 bytes. The MPLS header does not include a length indicator. Therefore to allow PWE3 to be carried in MPLS to correctly pass over an Ethernet data-link, a length correction field is needed in the control word.

In some networks it may be necessary to carry PWE3 over MPLS over IP. In these circumstances, the PW is encapsulated for carriage over MPLS as described in this section, and then a standard method of carrying MPLS over an IP PSN (such as GRE [RFC2784], [RFC2890]) is applied to the resultant PW-PDU.

6. PW Demultiplexer Layer and PSN Requirements

PWE3 places three service requirements on the protocol layers used to carry it across the PSN:

- o Multiplexing
- o Fragmentation
- o Length and Delivery

6.1 Multiplexing

The purpose of the PW Demultiplexer Layer is to allow multiple PWs to be carried in a single tunnel. This minimizes complexity and conserves resources.

Some types of native service are capable of grouping multiple circuits into a "trunk", e.g. multiple ATM VCs in a VP, multiple Ethernet VLANs on a physical media, or multiple DS0 services within a T1 or E1. A PW may interconnect two end-trunks. That trunk would have a single multiplexing value.

6.2 Fragmentation

If the PSN provides a fragmentation and reassembly service of adequate performance, it MAY be used to obtain an effective MTU that is large enough to transport the PW PDUs. However, if the PSN does not offer an adequate service, and fragmentation at the PE cannot be avoided by any other means, then a PW-specific fragmentation method may be utilized here. See Section 5.3 for more details.

6.3 Length and Delivery

PDU delivery to the egress PE is the function of the PSN Layer.

If the underlying PSN does not provide all the information necessary to determine the length of a PW-PDU, the Encapsulation Layer will provide it.

6.4 PW-PDU Validation

It is a common practice to use a CRC or similar mechanism to assure end-to-end integrity of frames. The PW service-specific mechanisms MUST define whether the packet's checksum shall be preserved across the PW or be removed from PE bound PDUs and then be re-calculated for insertion in CE bound data.

The former approach saves work, while the latter saves bandwidth. For a given implementation the choice may be dictated by hardware

restrictions.

For protocols such as ATM and FR, the scope of the checksum is restriced to a single link. This is because the circuit identifiers (e.g. FR DLCI or ATM VPI/VCI) have only local significance and are changed on each hop or span. If the circuit identifier (and thus checksum) were going to change as a part of the PW emulation, it would be more efficient to strip and re-calculate the checksum.

The service specific document for each protocol must describe the validation scheme to be used.

6.5 Congestion Considerations

The PSN carrying the PW may be subject to congestion. The congestion characteristics will vary with the PSN type, the network architecture and configuration, and the loading of the PSN.

Each service specific document will have to specify whether it needs an appropriate mechanism for operating in the presence of this congestion, including methods of mapping between its native congestion reporting and avoidance mechanisms, and those provided by the PW.

7. Control Plane

This section describes PWE3 control plane services.

7.1 Set-up or Teardown of Pseudo-Wires

A PW must be set up before an emulated service can be established, and must be torn down when an emulated service is no longer needed.

Set up or teardown of a PW can be triggered by a CLI command, from the management plane of a PE, by signaling (i.e., set-up or teardown) of a PWES, e.g., an ATM SVC, or by an auto-discovery mechanism.

During the set-up process, the PEs need to exchange some information (e.g. learn each others' capabilities). The tunnel signalling protocol may be extended to provide mechanisms to enable the PEs to exchange all necessary information on behalf of the PW.

Manual configuration of PWs can be considered a special kind of signaling, and is allowed.

7.2 Status Monitoring

Some native services have mechanisms for status monitoring. For example, ATM supports OAM for this purpose. For such services, the corresponding emulated services must specify how to perform status monitoring.

7.3 Notification of Pseudo-wire Status Changes

7.3.1. Pseudo-wire Up/Down Notification

If a native service requires bi-directional connectivity, the corresponding emulated service can only be signaled up when the associated PWs, and PSN tunnels if any, are functional in both directions.

Because the two CEs of an emulated service are not adjacent, a failure may occur at a place such that one or both physical links between the CEs and PEs remain up. For example, in Figure 2, if the physical link between CE1 and PE1 fails, the physical link between CE2 and PE2 will not be affected and will remain up. Unless CE2 is notified about the remote failure, it will continue to send traffic over the emulated service to CE1. Such traffic will be discarded at PE1. Some native services have failure notification so that when the services fail, both CEs will be notified. For such native services, the corresponding PWE3 service must provide a failure notification mechanism.

Similarly, if a native service has notification mechanisms so that when a network failure is fixed, all the affected services will change status from "Down" to "Up", the corresponding emulated service must provide a similar mechanism for doing so.

These mechanisms may already be built into the tunneling protocol. For example, the L2TP control protocol [RFC2661] [L2TPv3] has this capability and LDP has the ability to withdraw the corresponding MPLS label.

7.3.2. Misconnection and Payload Type Mismatch

With PWE3, misconnection and payload type mismatch can occur. If a misconnection occurs it can breach the integrity of the system. If a payload mismatch occurs it can disrupt the customer network. In both instances, there are security and operational concerns.

The services of the underlying tunneling mechanism, and its associated control protocol, can be used to mitigate this. As part of the PW set-up a PW-TYPE identifier is exchanged. This is then used by the FWRD and NSP to verify the compatibility of the PWESs.

7.3.3. Packet Loss, Corruption, and Out-of-order Delivery

A PW can incur packet loss, corruption, and out-of-order delivery on the PSN path between the PEs. This can impact the working condition of an emulated service. For some payload types, packet loss, corruption, and out-of-order delivery can be mapped to either a bit error burst, or loss of carrier on the PW. If a native service has some mechanism to deal with bit error, the corresponding PWE3 service should provide a similar mechanism.

7.3.4. Other Status Notification

A PWE3 approach may provide a mechanism for other status notification, if any.

7.3.5. Collective Status Notification

Status of a group of emulated services may be affected identically by a single network incident. For example, when the physical link (or sub-network) between a CE and a PE fails, all the emulated services that go through that link (or sub-network) will fail. It is likely that there exists a group of emulated services that all terminate at a remote CE. There may also be multiple such CEs affected by the

failure. Therefore, it is desirable that a single notification message be used to notify failure of the whole group of emulated services.

A PWE3 approach may provide some mechanism for notifying status changes of a group of emulated circuits. One possible method is to associate each emulated service with a group ID when the PW for that emulated service is set up. Multiple emulated services can then be grouped by associating them with the same group ID. In status notification, that group ID can be used to refer all the emulated services in that group. The group ID mechanism should be a mechanism provided by the underlying tunnel signaling protocol.

7.4 Keep-alive

If a native service has a keep-alive mechanism, the corresponding emulated service needs to use a mechanism to propagate this across the PW. An approach following the principle of minimum intervention would be to transparently transport keep-alive messages over the PW. However, to accurately reproduce the semantics of the native

mechanism, some PWs may require an alternative approach, such as piggy-backing on the PW signaling mechanism.

7.5 Handling Control Messages of the Native Services

Some native services use control messages for maintaining the circuits. These control messages may be in-band, e.g. Ethernet flow control or ATM performance management, or out-of-band, e.g. the signaling VC of an ATM VP.

From the principle of minimum intervention, it is desirable that the PEs participate as little as possible in the signaling and maintenance of the native services. This principle should not, however, override the need to satisfactorily emulate the native service.

If control messages are passed through, it may be desirable to send them using either a higher priority or a reliable channel provided by the PW Demultiplexer layer. See PWE3 Channel Types.

8. Management and Monitoring

This section describes the management and monitoring architecture for PWE3.

8.1 Statistics

The PE can tabulate statistics that help monitor the state of the network, and to help with measurement of service level agreements (SLAs). Typical counters include:

- Counts of PW-PDUs sent and received, with and without errors.
- Counts of sequenced PW-PDUs lost.
- Counts of service PDUs sent and received, with and without errors (non-TDM).
- Service-specific interface counts.

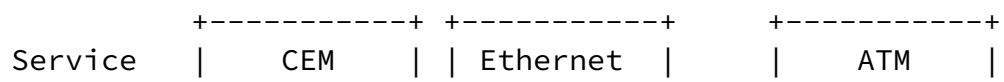
These counters would be contained in a PW-specific MIB, and they should not replicate existing MIB counters.

8.2 PW SNMP MIB Architecture

This section describes the general architecture for SNMP MIBs used to manage PW services and the underlying PSN. The intent here is to provide a clear picture of how all of the pertinent MIBs fit together to form a cohesive management framework for deploying PWE3 services.

8.2.1. MIB Layering

The SNMP MIBs created for PWE3 should fit the architecture shown in Figure 12.



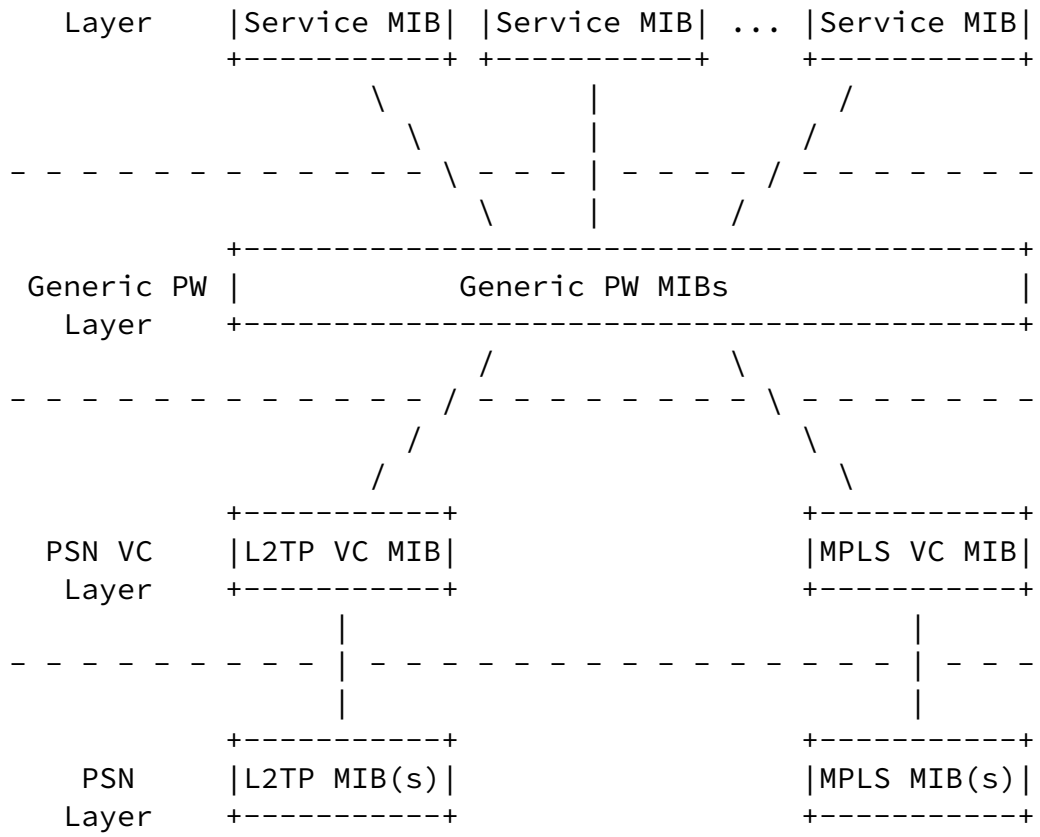
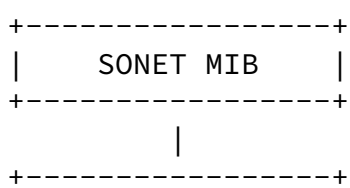


Figure 12: Relationship of SNMP MIBs

Figure 13 shows an example for a TDM PW carried over MPLS.



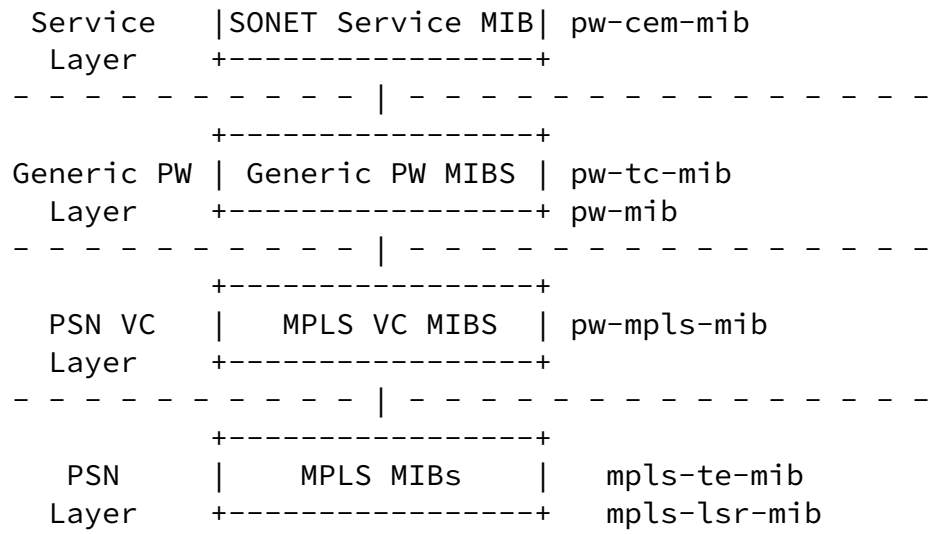


Figure 13: Service-specific Example for MIBs

Note that there is a separate MIB for each emulated service as well as one for each underlying PSN. These MIBs may be used in various combinations as needed.

8.2.2. Service Layer MIBs

The first layer is referred to as the Service Layer. It contains MIBs for PWE3 services such as Ethernet, ATM, circuits and Frame Relay. This layer contains those corresponding MIBs used to mate or adapt those emulated services to the underlying services. This working group should not produce any MIBs for managing the general service; rather, it should produce just those MIBs that are used to interface or adapt the emulated service onto the PWE3 management framework. For example, the standard SONET MIB [SONETMIB] is designed and maintained by another working group. Also, the SONET MIB is designed to manage the native service without PW emulation. Since the PWE3 working group is chartered to produce the corresponding adaptation MIB, in this case, it would produce the PW-CEM-MIB [PWMPLSMIB] that would be used to adapt SONET services to the underlying PSN that carries the PWE3 service.

8.2.3. Generic PW MIBs

The second layer is referred to as the Generic PW Layer. This layer is composed of two MIBs: the PWE-TC-MIB [PWTCMIB] and the PWE-MIB [PWMIB]. These MIBs are responsible for providing general PWE3 counters and service models used for monitoring and configuration of PWE3 services over any supported PSN service. That is, this MIB provides a general model of PWE3 abstraction for management purposes. This MIB is used to interconnect the Service Layer MIBs to the PSN VC Layer MIBs. The latter will be described in the next section. This layer also provides the PW-TC-MIB [PWTCMIB]. This MIB contains common SMI textual conventions [RFC1902] that may be used by any PW MIB.

8.2.4. PSN VC Layer MIBs

The third layer in the PWE3 management architecture is referred to as the PSN VC layer. This layer is comprised of MIBs that are specifically designed to interface general PWE3 services (VCs) onto those underlying PSN services. In general this means that the MIB provides a means with which an operator can map the PW service onto the native PSN service. For example, in the case of MPLS, it is required that the general VC service be layered onto MPLS LSPs or Traffic Engineered (TE) Tunnels [RFC3031]. In this case, the PW-MPLS-MIB [PWMPLSMIB] was created to adapt the general PWE3 circuit services onto MPLS. Like the Service Layer described above the PWE3 working group should produce these MIBs.

8.2.5. PSN Layer MIBs

The fourth and final layer in the PWE3 management architecture is referred to as the PSN layer. This layer is comprised of those MIBs that control the PSN service-specific services. For example, in the case of the MPLS [RFC3031] PSN service, the MPLS-LSR-MIB [LSRMIB] and the MPLS-TE-MIB [TEMIB] are used to interface the general PWE3 VC services onto native MPLS LSPs and/or TE tunnels to carry the emulated services. In addition, the MPLS-LDP-MIB [LDPMIB] may be used to reveal the MPLS labels that are distributed over the MPLS PSN in order to maintain the PW service. The MIBs in this layer are produced by other working groups that design and specify the native PSN services. These MIBs should contain the appropriate mechanisms for monitoring and configuring the PSN service such that the emulated PWE3 service will function correctly.

8.3 Connection Verification and Traceroute

A connection verification mechanism should be supported by PWs. Connection verification as well as other alarming mechanisms can alert the operator that a PW has lost its remote connection. The opaque nature of a PW means that it is not possible to specify a generic connection verification or traceroute mechanism that passes this status to the CEs over the PW. If connection verification status of the PW is needed by the CE, it must be mapped to the native connection status method.

For troubleshooting purposes, it is sometimes desirable to know the exact functional path of a PW between PEs, thus a traceroute function capable of reporting the path taken by data packets over the PW should be provided. The opaque nature of the PW means that this traceroute information is only available within the provider network e.g. at the PEs.

9. IANA considerations

There are no IANA considerations for this document.

10. Security Considerations

PWE3 provides no means of protecting the contents or delivery of the native data units. PWE3 may, however, leverage security mechanisms provided by the PW Demultiplexer or PSN Layers, such as IPSec [RFC2401]. This section addresses the PWE3 vulnerabilities, and the mechanisms available to protect the emulated native services.

The PW Tunnel End-Point, PW Demultiplexing mechanism, and the payloads of the native service are all vulnerable to attack.

10.1 PW Tunnel End-Point and PW Demultiplexer Security

Protection mechanisms must be considered for the PW Tunnel end-point and PW Demultiplexer mechanism in order to avoid denial-of-service

attacks upon the native service, and to prevent spoofing of the native data units. Exploitation of vulnerabilities from within the PSN may be directed to the PW Tunnel end-point so that PW Demultiplexer and PSN tunnel services are disrupted. Controlling PSN access to the PW Tunnel end-point may protect against this.

By restricting PW Tunnel end-point access to legitimate remote PE sources of traffic, the PE may reject traffic that would interfere with the PW Demultiplexing and PSN tunnel services.

10.2 Validation of PW Encapsulation

Protection mechanisms must address the spoofing of tunneled PW data. The validation of traffic addressed to the PW Demultiplexer end-point is paramount in ensuring integrity of PW encapsulation. Security protocols such as IPSec [RFC2401] may be used by the PW Demultiplexer Layer in order to maintain the integrity of the PW by authenticating data between the PW Demultiplexer End-points. IPSec may provide authentication, integrity, non-repudiation, and confidentiality of data transferred between two PEs. It cannot provide the equivalent services to the native service.

Based on the type of data being transferred, the PW may indicate to the PW Demultiplexer Layer that enhanced security services are required. The PW Demultiplexer Layer may define multiple protection profiles based on the requirements of the PW emulated service. CE-to-CE signaling and control events emulated by the PW and some data types may require additional protection mechanisms. Alternatively, the PW Demultiplexer Layer may use peer authentication for every PSN packet to prevent spoofed native data units from being sent to the destination CE.

Acknowledgments

We thank: Sasha Vainshtein for his work on Native Service Processing and advice on bit-stream over PW services. Thomas K. Johnson for his work on the background and motivation for PWs.

We also thank: Ron Bonica, Stephen Casner, Durai Chinnaiah, Jayakumar Jayakumar, Ghassem Koleyni, Eric Rosen, John Rutemiller, Scott Wainner and David Zelig for their comments and contributions.

References

Internet-drafts are works in progress available from <http://www.ietf.org/internet-drafts/>

[ETSI] EN 300 744 Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television (DVB-T), European Telecommunications Standards Institute (ETSI)

[LDP-MIB] Cucchiara, J., Sjostrand, H., and Luciani, J., "Definitions of Managed Objects for the Multiprotocol

Label Switching, Label Distribution Protocol (LDP)", <draft-ietf-mpls-ldp-mib-08.txt>, work in progress, August 2001.

[LSRMIB] Srinivasan et al, "MPLS Label Switch Router Management Information Base Using SMIv2", (draft-ietf-mpls-lsr-mib-08.txt, work in progress, January 2002.

[L2TPv3] Layer Two Tunneling Protocol (Version 3)'L2TPv3', J Lau, et. al. (work in progress).

[PPPoL2TP] PPP Tunneling Using Layer Two Tunneling Protocol, J Lau et al. <draft-ietf-l2tpext-l2tp-ppp-01.txt>, work in progress.

[PWMIB] Zelig et al, "Pseudo Wire (PW) Management Information Base Using SMIv2", (draft-ietf-pwe3-pw-mib-00.txt), work in progress, June 2002.

[PWTCMIB] Nadeau et al, "Definitions for Textual Conventions and OBJECT-IDENTITIES for Pseudo-Wires Management" (draft-ietf-pwe3-pw-tc-mib-00.txt), work in progress, June 2002.

[PWMPLSMIB] Danenberg et al, "SONET/SDH Circuit Emulation Service Over MPLS (CEM) Management Information Base Using SMIv2", (draft-ietf-pwe3-cep-mib-00.txt), work in progress, August 2001.

[RFC1191] RFC-1191: Path MTU discovery. J.C. Mogul, S.E. Deering.

[RFC1889] RFC-1889: RTP: A Transport Protocol for Real-Time Applications. H. Schulzrinne et. al.

[RFC1902] RFC-1902: Structure of Management Information for Version 2 of the Simple Network Management Protocol (SNMPv2), Case et al, January 1996.

[RFC1958] RFC-1958: Architectural Principles of the Internet, B. Carpenter et al.

[RFC1981] RFC-1981: Path MTU Discovery for IP version 6. J. McCann, S. Deering, J.Mogul.

[RFC2022] RFC-2022: Support for Multicast over UNI 3.0/3.1 based ATM Networks, G. Armitage.

[RFC2401] RFC-2401: Security Architecture for the Internet Protocol. S. Kent, R. Atkinson.

[RFC2474] RFC-2474: Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers, K. Nichols, et. al.

[RFC2661] RFC-2661: Layer Two Tunneling Protocol "L2TP". W. Townsley, et. al.

[RFC2784] RFC-2784: Generic Routing Encapsulation (GRE). D. Farinacci et al.

[RFC2890] RFC-2890: Key and Sequence Number Extensions to GRE. G. Dommety.

[RFC3022] RFC-3022: Traditional IP Network Address Translator (Traditional NAT). P Srisuresh et al.

[RFC3031] RFC3031: Multiprotocol Label Switching Architecture, E. Rosen, January 2001.

[SONETMIB] K. Tesink, "Definitions of Managed Objects for the SONET/SDH Interface Type", RFC2558, March 1999.

[TEMIB] Srinivasan et al, "Traffic Engineering Management Information Base Using SMIv2", (draft-ietf-mpls-te-mib-08.txt), work in progress, January 2002.

[VPLS] M. Lasserre, "Virtual Private LAN Services over MPLS", draft-lasserre-vkompella-ppvpn-vpls-02.txt, work in progress, June 2002.

[XIAO] Xiao et al, "Requirements for Pseudo-Wire Emulation Edge-to-Edge (PWE3)", (draft-ietf-pwe3-requirements-03.txt), X Xiao et al. work in progress, December 2002.

Editors' Addresses

Stewart Bryant
Cisco Systems,
4, The Square,
Stockley Park,
Uxbridge UB11 1BL,
United Kingdom. Email: stbryant@cisco.com

Prayson Pate
Overture Networks, Inc.
P. O. Box 14864
RTP, NC, USA 27709 Email: prayson.pate@overturenetworks.com