

Internet Draft
Document: [draft-ietf-pwe3-fragmentation-05.txt](#)
Expires: August 2004

Andrew G. Malis
Tellabs
W. Mark Townsley
Cisco Systems
February 2004

PWE3 Fragmentation and Reassembly

Status of this Memo

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

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

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

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

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

Abstract

This document defines a generalized method of performing fragmentation for use by PWE3 protocols and services.

Table of Contents

1.	Overview.....	2
2.	Alternatives to PWE3 Fragmentation/Reassembly.....	3
3.	PWE3 Fragmentation With MPLS.....	4
3.1	Fragment Bit Locations For MPLS.....	4
3.2	Other Considerations.....	5
4.	PWE3 Fragmentation With L2TP.....	5
4.1	PW-specific Fragmentation vs. IP fragmentation.....	6
4.2	Advertising Reassembly Support in L2TP.....	6
4.3	L2TP Maximum Receive Unit (MRU) AVP.....	7
4.4	L2TP Maximum Reassembled Receive Unit (MRRU) AVP.....	7
4.5	Fragment Bit Locations For L2TPv3 Encapsulation.....	8
4.6	Fragment Bit Locations for L2TPv2 Encapsulation.....	8
5.	Security Considerations.....	9
6.	IANA Considerations.....	9

February 2004

7. Acknowledgements.....	9
8. References.....	10
9. Authors' Addresses.....	11
10. Appendix A : Relationship Between This Document and RFC 1990	11

A Pseudo Wire (PW) payload is normally relayed across the PW as a single PSN (IP or MPLS) PDU. However, there are cases where the combined size of the payload and its associated PWE3 and PSN headers may exceed the PSN path Maximum Transmission Unit (MTU). When a packet exceeds the MTU of a given network, fragmentation and reassembly will allow the packet to traverse the network and reach its intended destination.

Fragmentation is also useful for real-time applications when the payload to be transmitted in a PW, such as a low-speed TDM multiframe structure, takes too much time to be encapsulated even

PWE3 Fragmentation and Reassembly

February 2004

though it may fit within the PW MTU. In this case, the payload may be fragmented for lower-latency transmission.

The purpose of this document is to define a generalized method of performing fragmentation for use with all PWE3 protocols and services. This method should be utilized only in cases where MTU-management methods fail. Due to the increased processing overhead, fragmentation and reassembly in core network devices should always be considered something to avoid whenever possible.

The PWE3 fragmentation and reassembly domain is shown in Figure 2:

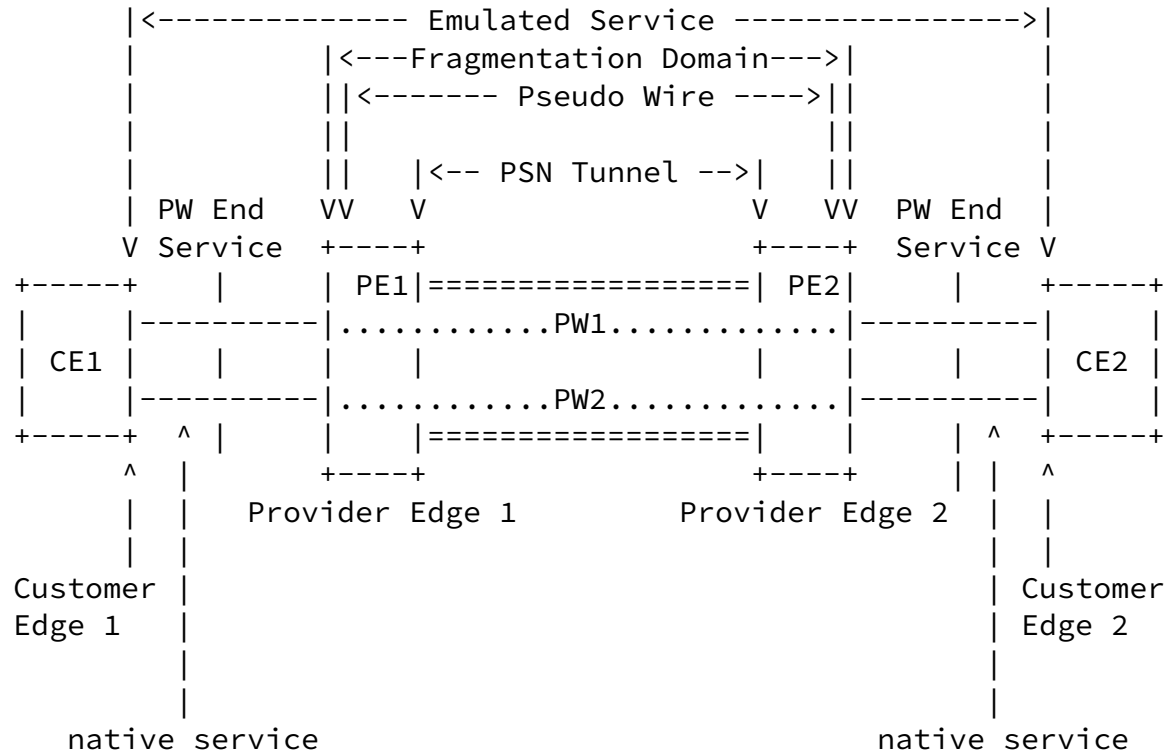


Figure 2: PWE3 Fragmentation/Reassembly Domain

Fragmentation takes place in the transmitting PE immediately prior

to PW insertion, and reassembly takes place in the receiving PE immediately after PW extraction.

2. Alternatives to PWE3 Fragmentation/Reassembly

Fragmentation and reassembly in network equipment generally requires significantly greater resources than sending a packet as a

single unit. As such, fragmentation and reassembly should be avoided whenever possible. Ideal solutions for avoiding fragmentation include proper configuration and management of MTU sizes between the CE, PE and across the PSN, as well as adaptive measures which operate with the originating host [e.g. [\[PATHMTU\]](#), [\[PATHMTUv6\]](#)] to reduce the packet sizes at the source.

A PE MAY choose to fragment a packet before allowing it to enter a PW. For example, if an IP packet arrives from a CE with an MTU which will yield a PW packet which is greater than the PW MTU, the PE may perform IP fragmentation on the packet. This effectively creates two (or more) packets, each carrying an IP fragment, for transport individually across the PW. The receiving PE is unaware that the originating host did not perform the IP fragmentation, and as such does not treat the PW packets in any special way. This ultimately has the affect of placing the burden of fragmentation on the PE, and reassembly on the IP destination host.

3. PWE3 Fragmentation With MPLS

When using the signaling procedures in [\[MPLS-TRANS\]](#), there is a Virtual Circuit FEC element parameter ID used to signal the use of fragmentation when advertising a VC label:

Parameter	ID	Length	Description
0x09		2	Fragmentation indicator

The presence of this parameter ID in the VC FEC element indicates that the receiver is able to reassemble fragments when the control word is in use for the VC label being advertised. It does not oblige the sender to use fragmentation; it is simply an indication that the sender MAY use fragmentation. The sender MUST NOT use fragmentation if this parameter ID is not present in the VC

FEC element.

If [\[MPLS-TRANS\]](#) signaling is not in use, then whether or not to use fragmentation MUST be provisioned in the sender.

[3.1](#) Fragment Bit Locations For MPLS

MPLS-based PWE3 [\[MPLS-ATM\]](#), [\[MPLS-Ethernet\]](#), [\[MPLS-FR\]](#), [\[MPLS-SATOP\]](#) uses the following control word format, with the B and E fragmentation bits identified in position 8 and 9:

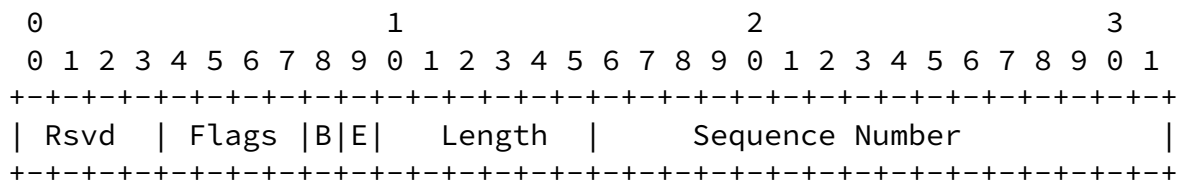


Figure 3: MPLS PWE3 Control Word

The B and E bits are defined as follows:

BE

--

00 indicates that the entire (un-fragmented) payload is carried in a single packet

01 indicates the packet carrying the first fragment

10 indicates the packet carrying the last fragment

11 indicates a packet carrying an intermediate fragment

See [Appendix A](#) for a discussion of the derivation of these values for the B and E bits.

The Sequence Number field is used as already specified in the above encapsulation specifications. Specifically, depending on the specific encapsulation in use, the value 0 may indicate that the sequence number is not in use, in which case its use for fragmentation must follow this same rule - as the sequence number

is incremented, it skips zero and wraps from 65535 to 1. Conversely, if the value 0 is part of the sequence space (as in [[MPLS-SATOP](#)]), then the same sequence space is also used for fragmentation and reassembly. Since a sequence number is necessary for the fragmentation and reassembly procedures, using the Sequence Number field on fragmented packets is REQUIRED.

[3.2](#) Other Considerations

Path MTU [[PATHMTU](#)] [[PATHMTUv6](#)] may be used to dynamically determine the maximum size for fragments. The application of path MTU to MPLS is discussed in [[LABELSTACK](#)]. The maximum size of the fragments may also be provisioned. The signaled Interface MTU parameter in [MPLS-TRANS] SHOULD be used to set the maximum size of the reassembly buffer for received packets to make optimal use of reassembly buffer resources.

[4.](#) PWE3 Fragmentation With L2TP

This section defines the location of the B and E bits for L2TPv3 [[L2TPv3](#)] and L2TPv2 [[L2TPv2](#)] headers, as well as the signaling

mechanism for advertising MRU (Maximum Receive Unit) values and support for fragmentation on a given PW. As IP is the most common PSN used with L2TP, IP fragmentation and reassembly is discussed as well.

[4.1](#) PW-specific Fragmentation vs. IP fragmentation

L2TPv3 recognizes that when it is used over IP networks, it may be subject to IP fragmentation. The following is quoted from [[L2TPv3](#)]:

IP fragmentation may occur as the L2TP packet travels over the IP substrate. L2TP makes no special efforts defined in this document to optimize this.

When proper MTU management across a network fails, IP fragmentation and reassembly may be used to accommodate MTU mismatches between tunnel endpoints. If the overall traffic requiring fragmentation and reassembly is very light, or there are sufficient optimized mechanisms for IP fragmentation and reassembly available, IP

fragmentation and reassembly may be sufficient and is allowed, particularly if PW-specific fragmentation is unavailable.

When facing a large number of PW packets requiring fragmentation and reassembly, a PW-specific method has properties that allow for more resource-friendly implementations. Specifically, the ability to assign buffer usage on a per-PW basis and per-PW sequencing may be utilized to significant advantage over a general mechanism applying to all IP packets equally. Further, PW fragmentation may be easily enabled in a selective manner for some or all PWs, rather than enabling reassembly for all IP traffic arriving at a given node.

Deployments MUST avoid a situation which relies upon a combination of IP and PW fragmentation and reassembly on the same node. Such operation clearly defeats the purpose behind the mechanism defined in this document. Care must be taken to ensure that the MTU/MRU values are set and advertised properly at each tunnel endpoint to avoid this. When fragmentation is enabled within a given PW, the DF bit MUST be set on all L2TP over IP packets for that PW. L2TPv3 nodes SHOULD participate in Path MTU [[PATHMTU](#)], [[PATHMTUv6](#)] for automatic adjustment of the PW MTU.

[4.2](#) Advertising Reassembly Support in L2TP

The constructs defined in this section for advertising fragmentation support in L2TP are applicable to L2TPv3 and L2TPv2.

This document defines two new AVPs to advertise maximum receive unit values and reassembly support. These AVPs MAY be present in the ICRQ, ICRP, ICCN, OCRQ, OCRP, OCCN, or SLI messages. The most recent value received always takes precedence over a previous value, and MUST be dynamic over the life of the session if received via the SLI message. One of the two new AVPs (MRRU) is used to advertise that PWE3 reassembly is supported by the sender of the AVP. Reassembly support MAY be unidirectional.

[4.3](#) L2TP Maximum Receive Unit (MRU) AVP

0	1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5	

```

+---+---+---+---+---+---+---+---+---+---+
|               MRU               |
+---+---+---+---+---+---+---+---+---+---+

```

MRU (Maximum Receive Unit), attribute number TBD1, is the maximum size in octets of a fragmented or complete PW frame, including L2TP encapsulation, receivable by the side of the PW advertising this value. The advertised MRU does NOT include the PSN header (i.e. the IP and/or UDP header). This AVP does not imply that PWE3 fragmentation or reassembly is supported. If reassembly is not enabled or unavailable, this AVP may be used alone to advertise the MRU for a complete frame.

All L2TP AVPs have an M (Mandatory) bit, H (Hidden) bit, Length, and Vendor ID. This AVP may be hidden (the H bit may be 0 or 1). The M bit for this AVP SHOULD be set to 0. The Length (before hiding) is 8. The Vendor ID is the IETF Vendor ID of 0.

[4.4](#) L2TP Maximum Reassembled Receive Unit (MRRU) AVP

```

      0                               1
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+---+---+---+---+---+---+---+---+
|               MRRU               |
+---+---+---+---+---+---+---+---+---+---+

```

MRRU (Maximum Reassembled Receive Unit AVP), attribute number TBD2, is the maximum size in octets of a reassembled frame, including any PW framing, but not including the L2TP encapsulation or L2-specific sublayer. Presence of this AVP signifies the ability to receive PW fragments and reassemble them. Packet fragments MUST NOT be sent to an implementation which has not received this value from its peer in a control message. If the MRRU is present in a message, the MRU AVP MUST be present as well.

All L2TP AVPs have an M (Mandatory) bit, H (Hidden) bit, Length, and Vendor ID. This AVP may be hidden (the H bit may be 0 or 1). The M bit for this AVP SHOULD be set to 0. The Length (before hiding) is 8. The Vendor ID is the IETF Vendor ID of 0.

[4.5](#) Fragment Bit Locations For L2TPv3 Encapsulation

The B and E bits are defined as bits 2 and 3 in the L2TPv3 default L2-specific sublayer as depicted below, using the values defined in [section 3.1](#):

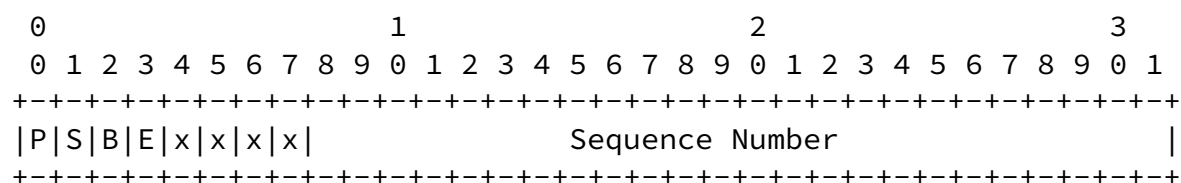


Figure 4: L2TPv3 over IP Header

Location of the B and E bits for PW-Types which use a variant L2-specific sublayer are outside the scope of this document.

Inclusion of the MRRU AVP in a control message suggests the need for a control sublayer which includes sequence numbers and the B and E bit fields. Thus, if reassembly support has been advertised, and packet fragments are to be sent, then presence of this sublayer and associated sequencing for all packet fragments MUST be enabled as defined for the given PW-type.

4.6 Fragment Bit Locations for L2TPv2 Encapsulation

The B and E bits are defined as bits 8 and 9 for the L2TPv2 header as depicted below (subject to IANA action), using the values defined in [section 3.1](#):

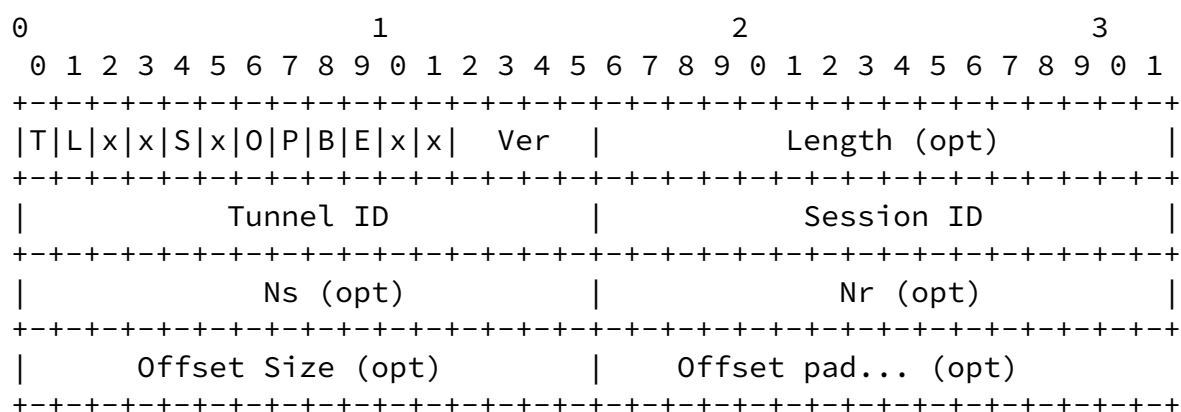


Figure 5: L2TPv2 over UDP Header

[5. Security Considerations](#)

As with any additional protocol construct, each level of complexity adds the potential to exploit protocol and implementation errors. Implementers should be especially careful of not tying up an abundance of resources, even for the most pathological combination of packet fragments that could be received. Beyond these issues of general implementation quality, there are no known notable security issues with using the mechanism defined in this document. It should be pointed out that [RFC 1990](#), on which this document is based, and its derivatives have been widely implemented and extensively used in the Internet and elsewhere.

[IPFRAG-SEC] and [[TINYFRAG](#)] describe potential network attacks associated with IP fragmentation and reassembly. The issues described in these documents attempt to bypass IP access controls by sending various carefully formed "tiny fragments", or by exploiting the IP offset field to cause fragments to overlap and rewrite interesting portions of an IP packet after access checks have been performed. The latter is not an issue with the PW-specific fragmentation method described in this document as there is no offset field; However, implementations MUST be sure to not allow more than one whole fragment to overwrite another in a reconstructed frame. The former may be a concern if packet filtering and access controls are being placed on tunneled frames within the PW encapsulation. To circumvent any possible attacks in either case, all filtering and access controls should be applied to the resulting reconstructed frame rather than any PW fragments.

[6. IANA Considerations](#)

This document does not define any new values for IANA to maintain.

This document requires definition of two reserved bits in the L2TPv2 [[L2TPv2](#)] header. Recommended locations are noted by the "B" and "E" bits in [section 5.6](#).

This document requires IANA to assign two new L2TP "Control Message Attribute Value Pairs" (TBD1 and TBD2 in this document).

[7. Acknowledgements](#)

Thanks to Eric Rosen for his review of this document.

8. References

- [Architecture] Bryant, S. et al, "PWE3 Architecture", [draft-ietf-pwe3-arch-06.txt](#), October 2003, work in progress
- [FAST] ATM Forum, "Frame Based ATM over SONET/SDH Transport (FAST)", af-fbatm-0151.000, July 2000
- [FRF.12] Frame Relay Forum, "Frame Relay Fragmentation Implementation Agreement", FRF.12, December 1997
- [LABELSTACK] Rosen, E. et al, "MPLS Label Stack Encoding", [RFC 3032](#), January 2001
- [L2TPv2] Townsley, Valencia, Rubens, Pall, Zorn, Palter, "Layer Two Tunneling Protocol 'L2TP'", [RFC 2661](#), June 1999
- [L2TPv3] Lau, J. et al, "Layer Two Tunneling Protocol (Version 3) 'L2TPv3'", [draft-ietf-l2tpext-l2tp-base-11.txt](#), October 2003, work in progress
- [MLPPP] Sklower, K. et al, "The PPP Multilink Protocol (MP)", [RFC 1990](#), August 1996
- [MPLS-ATM] Martini, L. et al, "Encapsulation Methods for Transport of ATM Cells/Frame Over IP and MPLS Networks", [draft-ietf-pwe3-atm-encap-04.txt](#), December 2003, work in progress
- [MPLS-Ethernet] Martini, L. et al, "Encapsulation Methods for Transport of Ethernet Frames Over IP and MPLS Networks", [draft-ietf-pwe3-ethernet-encap-05.txt](#), December 2003, work in progress
- [MPLS-FR] Martini, L. et al, "Frame Relay Encapsulation over Pseudo-Wires", [draft-ietf-pwe3-frame-relay-02.txt](#), February 2004, work in progress
- [MPLS-SATOP] Vainshtein, A. et al, "Structure-Agnostic TDM over Packet (SAToP)", [draft-ietf-pwe3-satop-01.txt](#), December 2003, work in progress
- [MPLS-TRANS] Martini, L. et al, "Transport of Layer 2 Frames Over

MPLS", [draft-ietf-pwe3-control-protocol-05.txt](#), December 2003, work in progress

[PATHMTU] Mogul, J. C. et al, "Path MTU Discovery", [RFC 1191](#), November 1990

Malis, Townsley

Expires June 2004

[Page 10]

PWE3 Fragmentation and Reassembly

February 2004

[PATHMTUv6] McCann, J. et al, "Path MTU Discovery for IP version 6", [RFC 1981](#), August 1996

[IPFRAG-SEC] Ziemba, G., Reed, D., Traina, P., "Security Considerations for IP Fragment Filtering", [RFC 1858](#), October 1995

[TINYFRAG] Miller, I., "Protection Against a Variant of the Tiny Fragment Attack", [RFC 3128](#), June 2001

[9.](#) Authors' Addresses

Andrew G. Malis
Tellabs
90 Rio Robles Drive
San Jose, CA 95134
Email: Andy.Malis@tellabs.com

W. Mark Townsley
Cisco Systems
7025 Kit Creek Road
PO Box 14987
Research Triangle Park, NC 27709
Email: mark@townsley.net

[10.](#) [Appendix A](#): Relationship Between This Document and [RFC 1990](#)

The fragmentation of large packets into smaller units for transmission is not new. One fragmentation and reassembly method was defined in [RFC 1990](#), Multi-Link PPP [[MLPPP](#)]. This method was also adopted for both Frame Relay [[FRF.12](#)] and ATM [[FAST](#)] network technology. This document adopts the [RFC 1990](#) fragmentation and reassembly procedures as well, with some distinct modifications described in this appendix. Familiarity with [RFC 1990](#) is assumed.

[RFC 1990](#) was designed for use in environments where packet fragments may arrive out of order due to their transmission on multiple parallel links, specifying that buffering be used to place the fragments in correct order. For PWE3, the ability to reorder fragments prior to reassembly is OPTIONAL; receivers MAY choose to drop frames when a lost fragment is detected. Thus, when the sequence number on received fragments shows that a fragment has been skipped, the partially reassembled packet MAY be dropped, or the receiver MAY wish to wait for the fragment to arrive out of order. In the latter case, a reassembly timer MUST be used to avoid locking up buffer resources for too long a period.

Dropping out-of-order fragments on a given PW can provide a considerable scalability advantage for network equipment performing reassembly. If out-of-order fragments are a relatively rare event on a given PW, throughput should not be adversely affected by this. Note, however, if there are cases where fragments of a given frame are received out-of-order in a consistent manner (e.g. a short fragment is always switched ahead of a larger fragment) then dropping out-of-order fragments will cause the fragmented frame to never be received. This condition may result in an effective denial of service to a higher-level application. As such, implementations fragmenting a PW frame MUST at the very least ensure that all fragments are sent in order from their own egress point.

An implementation may also choose to allow reassembly of a limited number of fragmented frames on a given PW, or across a set of PWs with reassembly enabled. This allows for a more even distribution of reassembly resources, reducing the chance of a single or small set of PWs exhausting all reassembly resources for a node. As with dropping out-of-order fragments, there are perceivable cases where this may also provide an effective denial of service. For example, if fragments of multiple frames are consistently received before each frame can be reconstructed in a set of limited PW reassembly buffers, then a set of these fragmented frames will never be delivered.

[RFC 1990](#) headers use two bits which indicate the first and last fragments in a frame, and a sequence number. The sequence number may be either 12 or 24 bits in length (from [\[MLPPP\]](#)):

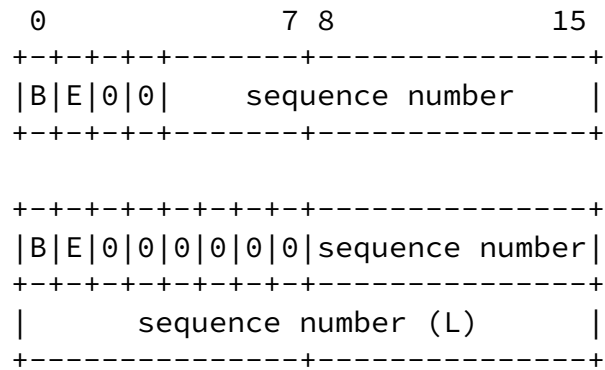


Figure 6: [RFC 1990](#) Header Formats

PWE3 fragmentation takes advantage of existing PW sequence numbers and control bit fields wherever possible, rather than defining a separate header exclusively for the use of fragmentation. Thus, it uses neither of the [RFC 1990](#) sequence number formats described above, relying instead on the sequence number that already exists in the PWE3 header.

[RFC 1990](#) defines a two one-bit fields, a (B)eginning fragment bit and an (E)nding fragment bit. The B bit is set to 1 on the first fragment derived from a PPP packet and set to 0 for all other fragments from the same PPP packet. The E bit is set to 1 on the last fragment and set to 0 for all other fragments. A complete unfragmented frame has both the B and E bits set to 1.

PWE3 fragmentation inverts the value of the B and E bits, while retaining the operational concept of marking the beginning and ending of a fragmented frame. Thus, for PW the B bit is set to 0 on the first fragment derived from a PW frame and set to 1 for all other fragments derived from the same frame. The E bit is set to 0 on the last fragment and set to 1 for all other fragments. A complete unfragmented frame has both the B and E bits set to 0. The motivation behind this value inversion for the B and E bits is to allow complete frames (and particularly, implementations that only support complete frames) to simply leave the B and E bits in the header set 0.

In order to support fragmentation, the B and E bits **MUST** be defined or identified for all PWE3 tunneling protocols. Sections [3](#) and [4](#)

define these locations for PWE3 MPLS [[MPLS-TRANS](#)], L2TPv2 [[L2TPv2](#)], and L2TPv3 [[L2TPv3](#)] tunneling protocols.