

Network Working Group  
Request for Comments: 4638  
Category: Informational

P. Arberg  
D. Kourkouzelis  
Redback Networks  
M. Duckett  
T. Anschutz  
BellSouth  
J. Moisand  
Juniper Networks  
September 2006

Accommodating a Maximum Transit Unit/Maximum Receive Unit (MTU/MRU)  
Greater Than 1492 in the  
Point-to-Point Protocol over Ethernet (PPPoE)

Status of This Memo

This memo provides information for the Internet community. It does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (2006).

IESG Note

As of this writing, no current IEEE standard supports the use of "jumbo frames" (MTU greater than 1500). Although this document contains recommended mechanisms to detect problems in the path, interoperability and reliability of non-standard extensions cannot be assured. Both implementors and users of the protocol described here should exercise caution in its use.

Abstract

The Point-to-Point Protocol over Ethernet (PPPoE), as described in RFC 2516, mandates a maximum negotiated Maximum Receive Unit (MRU) of 1492. This document outlines a solution that relaxes this restriction and allows a maximum negotiated MRU greater than 1492 to minimize fragmentation in next-generation broadband networks.

## Table of Contents

1. Introduction .....	2
2. Terminology .....	4
3. Proposed Solution .....	4
4. PPPoE Discovery Stage .....	5
5. LCP Considerations .....	5
5.1. MRU Negotiations .....	5
5.2. MRU Test and Troubleshooting .....	6
6. Security Considerations .....	7
7. IANA Considerations .....	7
8. Acknowledgements .....	7
9. Normative References .....	7
10. Informative References .....	8

## 1. Introduction

As broadband network designs are changing from PC-initiated PPPoE [1] sessions in a combined Ethernet/Asynchronous Transfer Mode (ATM) setup, as shown in Figure 1, to more intelligent PPPoE-capable Residential Gateway (RG) and Gigabit Ethernet/ATM broadband network designs, as shown in Figures 2 and 3, the need to increase the maximum transmit and receive unit in the PPPoE protocol is becoming more important in order to reduce fragmentation in the network.

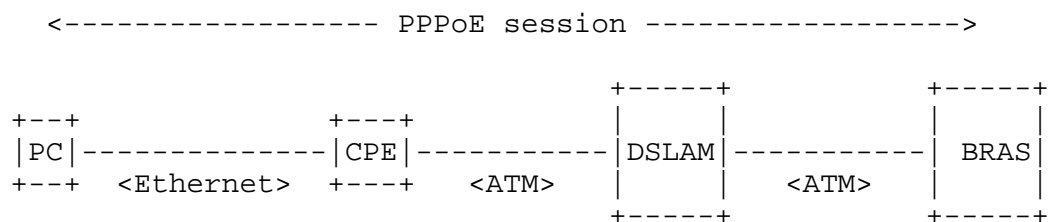


Figure 1. Initial broadband network designs with PPPoE

In the network design shown in Figure 1, fragmentation is typically not a problem, since the subscriber session is PPPoE end to end from the PC to the BRAS. Therefore, a PPP-negotiated MRU of 1492 octets is fully acceptable, as it makes the largest PPPoE frame adhere to the standard Ethernet MTU of 1500 octets.

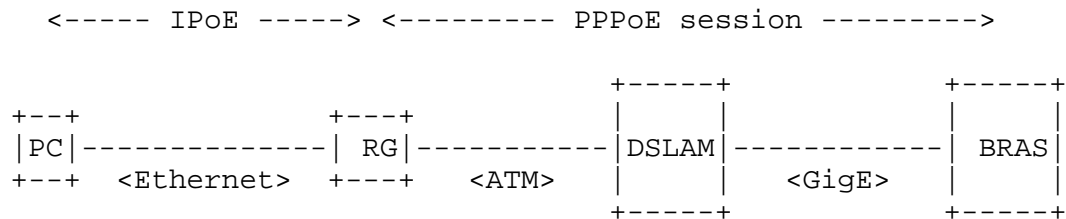


Figure 2. Next-generation broadband network designs with PPPoE

In the network design shown in Figure 2, fragmentation becomes a major problem, since the subscriber session is a combination of IPoE and PPPoE. The IPoE typically uses a Maximum Transit Unit (MTU) of 1500 octets. However, when the Residential Gateway and the Broadband Remote Access Server (BRAS) are the PPPoE session endpoints and therefore negotiate an MTU/MRU of 1492 octets, the result is a large number of fragmented packets in the network.

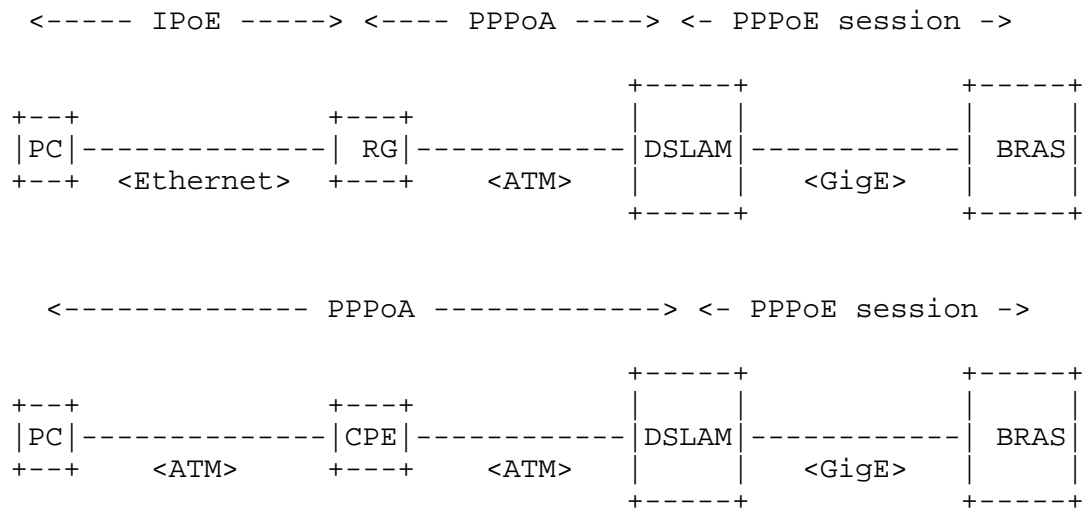


Figure 3. Broadband network designs with PPPoA-to-PPPoE conversion

In the network design shown in Figure 3, which is studied by the DSL-Forum in the context of the migration to Ethernet for broadband aggregation networks, fragmentation is not the only problem when MRU differences exist in Point-to-Point Protocol over AAL5 (PPPoA) and PPPoE sessions.

The subscriber session is a PPP session running over a combination of PPPoA and PPPoE. The PPP/PPPoA host typically negotiates a 1500-octet MRU. Widely deployed PPP/PPPoA hosts in Customer Premises Equipment (CPE) do not support a 1492-octet MRU, which creates an issue in turn for the BRAS (PPPoE server) if strict compliance to RFC

2516 [1] is mandated. For PPP/PPPoA hosts capable of negotiating a 1492-octet MRU size, then we are back to a fragmentation issue.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [3].

- ATM - Asynchronous Transfer Mode
- PPP - Point-to-Point Protocol
- PPPoA - PPP over AAL5
- PPPoE - PPP over Ethernet
- MTU - Maximum Transmit Unit
- MRU - Maximum Receive Unit
- PC - Personal Computer
- CPE - Customer Premises Equipment
- RG - Residential Gateway
- BRAS - Broadband Remote Access Server
- DSLAM - Digital Subscriber Line Access Multiplexer
- PPPoE - client PC, RG, or CPE that initiates a PPPoE session
- PPPoE - server BRAS terminating PPPoE sessions initiated by client
- PADI - PPPoE Active Discovery Initiation
- PADO - PPPoE Active Discovery Offer
- PADR - PPPoE Active Discovery Request
- PADS - PPPoE Active Discovery Session-confirmation

## 3. Proposed Solution

The procedure described in this document does not strictly conform to IEEE standards for Ethernet packet size but relies on a widely deployed behavior of supporting frames with Ethernet packet format, but exceeding the maximum packet lengths defined by [4].

Since next-generation broadband networks are built around Ethernet systems supporting baby-giants and jumbo frames with payload sizes larger than the normal Ethernet MTU of 1500 octets, a BRAS acting as a PPPoE server MUST support PPPoE MRU negotiations larger than 1492 octets in order to limit the amount of fragmented packets in networks similar to those described in Section 1.

By default, the Maximum-Receive-Unit (MRU) option MUST follow the rules set forward in RFC 1661 [2] but MUST NOT be negotiated to a size larger than 1492 to guarantee compatibility with Ethernet network segments limited to 1500-octet frames. In such a case, as the PPPoE header is 6 octets and the PPP Protocol ID is 2 octets, the PPP MRU MUST NOT be greater than 1492.

An optional PPPoE tag, "PPP-Max-Payload", allows a PPPoE client to override this default behavior by providing a maximum size for the PPP payload it can support in both the sending and receiving directions. When such a tag is received by the PPPoE server, the server MAY allow the negotiation of an MRU larger than 1492 and the use of an MTU larger than 1492, subject to limitations of its local configuration and according to the rules set forward in RFC 1661 [2], within the limits of the maximum payload size indicated by the PPPoE client.

#### 4. PPPoE Discovery Stage

If a PPPoE client wants to use an MTU/MRU higher than 1492 octets, then it MUST include an optional PPP-Max-Payload Tag in the PADI and PADR packets. If the PPPoE server can support an MTU/MRU higher than 1492 octets, it MUST respond with an echo of the clients tag in the PADO and PADS packets when the PPP-Max-Payload tag is received from the client.

Tag-name: PPP-Max-Payload  
Tag-value: 0x0120  
Tag-length: 2 octets  
Tag-value: binary encoded value (max PPP payload in octets)

Tag-description:

This TAG indicates that the client and server are capable of supporting a given maximum PPP payload greater than 1492 octets for both the sending and receiving directions. Note that this value represents the PPP payload; therefore it is directly comparable with the value used in the PPP MRU negotiation.

#### 5. LCP Considerations

##### 5.1. MRU Negotiations

Since Ethernet (without jumbo frames) has a maximum payload size of 1500 octets, the PPPoE header is 6 octets, and the PPP Protocol ID is 2 octets, the Maximum-Receive-Unit (MRU) option MUST NOT be negotiated to a size larger than 1492, unless both the PPPoE client and server have indicated the ability to support a larger MRU in the PPPoE Discovery Stage.

The initial MRU negotiation for the PPP/PPPoE server MUST follow a flow as shown below:

```
If PPPoE {  
  PPP_MRU_Max = 1492  
  If (PPP-Max-Payload-Tag) AND (PPP-Max-Payload-Tag > 1492)  
  Then PPP_MRU_Max = min (PPP-Max-Payload-Tag, Interface MTU-8)  
}  
"Normal" PPP_MRU_Negotiation (PPP_MRU_Max)
```

If the PPP-Max-Payload tag is present and greater than 1492, it MUST be considered along with the server's interface MTU settings when the maximum value is selected for the normal RFC 1661 [2] MRU negotiation which decides the actual MRU to use.

If the PPP-Max-Payload tag isn't present or is present but below 1492, then the existing MRU constraint of 1492 octets MUST stay applicable, thus preserving backward compatibility.

This, in summary, indicates the following behavior:

1. When a "PPP-Max-Payload" tag is received,
  - a. the value in this tag will indicate the maximum MRU allowed to be accepted or suggested in an MRU negotiation; and
  - b. if MRU is not negotiated, then RFC 1661 [2] will set the default MRU at 1500. This will say that the "PPP-Max-Payload" tag can have a value greater than 1500, but in this case RFC 1661 [2] sets the default MRU to 1500, and only if MRU is negotiated higher (up to maximum payload) will the "PPP-Max-Payload" tag value be used.
2. When a "PPP-Max-Payload" tag is not received by either end, then RFC 2516 [1] sets the rule.

## 5.2. MRU Test and Troubleshooting

If the MRU is negotiated to a value larger than 1492 octets, the sending side SHOULD have the option of sending one or more MRU-sized Echo-Request packets once the session is opened. This allows it to test that the receiving side and any intermediate Ethernet segments and equipment can handle such a packet size.

If no Echo-Replies are received, the sending side MAY choose to repeat the test with 1492 octets Echo-Request packets. If these packets receive replies, the sending side MUST not send packets bigger than 1492 octets for this session.

This capability SHOULD be enabled by default. It SHOULD be configurable and MAY be disabled on networks where there is some prior knowledge indicating that the test is not necessary.

## 6. Security Considerations

This document does not introduce new security issues. The security considerations pertaining to the original PPPoE protocol [1] remain relevant.

## 7. IANA Considerations

This document defines a new value in a space that currently has no IANA registry. There is work in progress to define a registry [5] and that document already contains the value assigned here. No IANA action is required for this document.

## 8. Acknowledgements

The authors would like to thank Prakash Jayaraman, Amit Cohen, Jim Ellis, David Thorne, John Reid, Oliver Thorp, Wojciech Dec, Jim Wilks, Mark Townsley, Bart Salaets, Tom Mistretta, Paul Howard, Dave Bernard, and Darren Nobel for their contributions and comments to this document.

## 9. Normative References

- [1] Mamakos, L., Lidl, K., Evarts, J., Carrel, D., Simone, D., and R. Wheeler, "A Method for Transmitting PPP Over Ethernet (PPPoE)", RFC 2516, February 1999.
- [2] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.
- [3] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.
- [4] Institute of Electrical and Electronic Engineers, IEEE Std 802.3-2005, "IEEE Standard for Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications - Draft amendment to - Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 3: Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications - Media Access Control Parameters, Physical Layers and Management Parameters", December 2005.

## 10. Informative References

- [5] Arberg, P. and V. Mammoliti, "IANA Considerations for PPP over Ethernet (PPPoE), Work in Progress, June 2006.

## Authors' Addresses

Peter Arberg  
Redback Networks, Inc.  
300 Holger Way  
San Jose, CA 95134

Email: [parberg@redback.com](mailto:parberg@redback.com)

Diamantis Kourkouzelis  
Redback Networks, Inc.  
300 Holger Way  
San Jose, CA 95134

Email: [diamondk@redback.com](mailto:diamondk@redback.com)

Mike Duckett  
BellSouth Telecommunications, Inc.  
575 Morosgo Drive  
Atlanta, GA 30324

Email: [mike.duckett@bellsouth.com](mailto:mike.duckett@bellsouth.com)

Tom Anschutz  
BellSouth Science and Technology  
725 W. Peachtree St.  
Atlanta, GA 30308

Email: [tom.anschutz@bellsouth.com](mailto:tom.anschutz@bellsouth.com)

Jerome Moisand  
Juniper Networks, Inc.  
10 Technology Park Drive  
Westford, MA 01886

Email: [jmoisand@juniper.net](mailto:jmoisand@juniper.net)



## 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).

