Internet-Draft
Expires August 1997

Dheeraj Sanghi
Sameer Shah
IIT, Kanpur
March 1997

Extended Path MTU Discovery for IP version 6

draft-sanghi-pmtudisc-ext-00.txt

Status of this Memo

Abstract

This draft discusses extensions to the present Path MTU Discovery mechanism [PMTUDISC]. It provides applications finer control over the delay and losses incurred during the PMTU Discovery process. The document proposes two extension header options that allow PMTU Discovery with minimal overheads.

1. Introduction

In the existing mechanism [PMTUDISC], a node starts with an initial estimate of PMTU equal to the next hop link mtu, receives Packet Too Big (PTB) messages until it discovers the correct PMTU; or decides to stop the process and use a minimum MTU value. Several iterations of the packet-sent/PTB message cycle may occur before actual data transfer begins.

This method has two disadvantages. First there is an initial variable delay before actual data transfer. Second network resources are

wasted due to loss of packets in the discovery process. The latter

effect is offset by using a better MTU estimate for subsequent packets, but the exact measure directly depends upon the amount of data sent subsequently.

Clearly this imposes a significant overhead in the case where hosts communicate infrequently (such as request-response kind of transfers) as it is likely that knowledge of Path MTU will not exist when data tranfer starts and will have to be discovered. With the variety of link technologies that can interoperate in IPv6 such as wireless links (very small MTU) to IP over SMDS (MTU of 9180 bytes), this overhead can be large.

The tolerance to such overheads can be defined in the context of an application. Many applications have restrictions on the tolerable delay. Additionally many applications can determine the total amount of data to be transferred before actual tranfer begins. If an application has stringent delay requirements and/or has very less data to send, it can as well do without the PMTU Discovery.

We recommend providing applications a finer control from applications over the PMTU discovery process. Based on tolerable loss and delays, applications should be able to decide on an optimal MTU value. We also propose two extension header options to discover the PMTU value. The PMTU Detection Option is present in the Hop-by-Hop extension header and records the PMTU value on packets sent from source to destination. The PMTU Indication Option is present in the Destination extension header and is used by the destination of a PMTU Detection Option to indicate the PMTU value to the source.

2. Finer Control from Applications

[PMTUDISC] suggests that an implementation provide a way to specify whether Path MTU Discovery not be done on a given path. This should be extended to the level of individual applications.

Applications should be able to specify a 'desired MTU' value to the transport layer. This means that the MTU to be used for packets from that application should not exceed this value and if the network layer notifies a Path MTU greater than this value, the MTU should be clamped to the desired value for the particular application.

Selection of a desired MTU depends upon the application and this value determines the amount of overhead due to PMTU Discovery. In the extreme case, this can be equal to 576 bytes meaning the least delay and no loss.

2.1. Upper Layer Issues

For applications on top of TCP, the desired value can be indicated to the TCP layer using a socket option. The TCP layer uses this value when packetizing data from the application.

Applications on top of UDP are responsible for the packetization. They indicate the desired value to the UDP layer. This value can be used to decide if an application should be notified of a PMTU change. For applications which do not respond to PMTU change notifications, this value should be used by the source IP level fragmentation.

3. Extension Header Options

The aim of PMTU Discovery is to use the largest possible MTU estimate so that the bandwidth utilized to transfer data is relatively larger than that used to transfer headers and other overheads.

But as previously mentioned this benefit is offset due to loss of packets during PMTU Discovery. We propose two extension headers that allow PMTU Discovery with minimal losses and delay. Note that this mechanism should be used alongwith the mechanism in [PMTUDISC]. The proposed method is applicable only in the case of unicast destination addresses.

We propose a hop-by-hop option that records the minimum mtu on a path from source to destination. A source sets this option and fills the next hop link mtu in an 'Affirmative MTU' field. Each router compares the Affirmative MTU value in this option with the link mtu for the next hop. If the link mtu for the next hop is smaller, it replaces the Affirmative MTU with the link MTU for the next hop.

In the existing MTU discovery algorithm, loss of packets can occur at two instances:

- Initial detection of PMTU

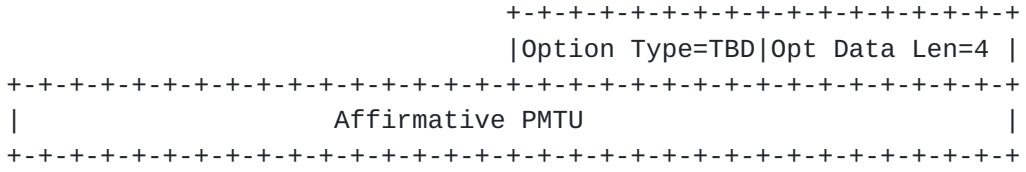
- Attempting to discover increases in PMTU

This option can be used at both these instances. When a node starts transmission to a destination for which a pmtu estimate is not available, it starts with a PMTU estimate of 576 bytes and sets this option in the Hop-by-Hop extension header for the first few packets. The destination node stores the received Affirmative PMTU value in the local representation of a path to the sender. It indicates the received Affirmative MTU value to the source using a new Destination Option.

Similarly this option can be set to find increases in the PMTU.

3.1. PMTU Detection Option

This option is present in the Hop-by-Hop extension header.

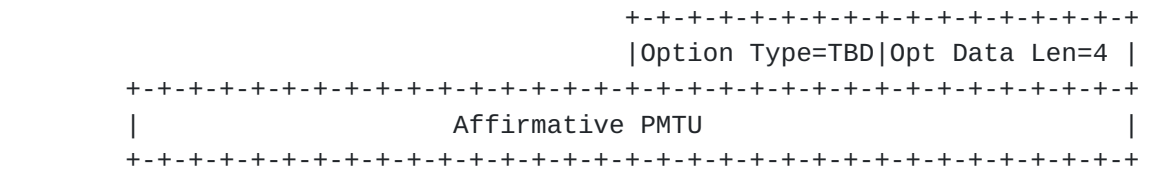


"TBD"	The Hop-by-Hop Option Type number allocated by IANA for this option.
Affirmative PMTU	This field is set by the sender node to be the next hop MTU. Each router sets this value to the minimum of the current value and the next hop MTU

Routers that do not recognize this option should discard the packet and send an ICMP Parameter Problem message to the packet's Source Address. This option can be changed en-route.

3.2. PMTU Indication Option

This option is present in the Destination extension header.



"TBD"	The Destination Option Type number allocated by IANA for this option.
Affirmative PMTU	The affirmative PMTU value received in the PMTU Indication Option on a packet from the destination node

The option data does not change en-route.

3.3. Discussion

A likely usage scenario is described here.

Applications indicate their willingness to set these options to the transport layer.

When a willing application sends data to a destination with a unicast address, it uses the available PMTU only if it is Affirmative i.e. it has been previously discovered using the PMTU Detection and PMTU Indication Options. Otherwise it uses a PMTU of 576 bytes. The PMTU Detection Option is set in the sent packets. Related state information such as the time when probe started and the number of

packets sent with the option set, can be stored in the local representation of a path to the destination e.g. the destination cache. When a PMTU Indication Option is received from the destination within a maximum timeout period or the round trip time (if available), the Affirmative MTU is indicated to the transport layers. Additionally the PMTU to the destination can be set equal to the received Affirmative MTU in the local representation of the destination. If a PMTU Indication Option is not received within the threshold time interval or a ICMP Parameter Problem message is received for a packet sent with the PMTU Detection Option, then PMTU Discovery proceeds using the method in [PMTUDISC].

When an increase in PMTU is to be discovered, the PMTU estimate is not changed for willing applications initially, but the PMTU Detection Option is set in sent packets. If the received PMTU Indication Option indicates a change in the Affirmative PMTU value, the new PMTU for the destination is notified to the packetization layers. Again if a timeout occurs and a PMTU Indication Option is not received or a ICMP Parameter Problem message is received, then it reverts to the default PMTU Discovery algorithm.

A node receiving the PMTU Detection Option sets the PMTU Indication Option in packets sent from willing applications to the initial sender. [PMTUDISC] recommends a timeout of 10 minutes before trying to discover an increased PMTU, but we expect the proposed extensions can be used with a higher frequency based on actual link conditions and previous feedbacks. If this option is used only once in 5 - 10 minutes, then the overhead is minimal.

4. Security Considerations

The PMTU Detection Option is vulnerable to similar denial-of-service attacks as described in [PMTUDISC].

The PMTU Detection Option is zeroed for AH calculations as it can change along the path. The PMTU Indication Option is included in the IPv6 Authentication Header and is not zeroed for AH calculations.

5. References

[PMTUDISC] J. McCann, S. Deering & J. Mogul, "Path MTU Discovery for IP version 6", RFC-1981, Internet Engineering Task Force, August 1996.

Authors' Addresses:

Dheeraj Sanghi
Department of CSE
Indian Institute of Technology
Kanpur, India

Phone: +91 (512) 25-7077
Email: dheeraj@iitk.ernet.in

Sameer Shah
Department of CSE
Indian Institute of Technology
Kanpur, India

Phone: +91 (512) 25-7653
Email: ocrds@iitk.ernet.in