Network Working Group
Internet-Draft
Intended status: Informational
Expires: January 10, 2014

F. Templin, Ed.
Boeing Research & Technology

July 9, 2013

Fragmentation Revisited
draft-generic-6man-tunfrag-08.txt

Abstract

IP fragmentation has long been subject for scrutiny since the publication of "Fragmentation Considered Harmful" in 1987. This work cast fragmentation in a negative light that has persisted to the present day. However, the tone of the work failed to honor two principles of creative thinking: never say "always" and never say "never". This document discusses uses for fragmentation that apply both to the present day and moving forward into the future.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on January 10, 2014.

Table of Contents

1. Introduction

IP fragmentation has long been subject for scrutiny since the publication of "Fragmentation Considered Harmful" in 1987 [FRAG]. This work cast fragmentation in a negative light that has persisted to the present day. However, the tone of the work failed to honor two principles of creative thinking: never say "always" and never say "never". This document discusses uses for fragmentation that apply both to the present day and moving forward into the future.

2. Problem Statement

The de facto "Internet cell size" is effectively 1500 bytes, i.e., the minimum maximum Transmission Unit (minMTU) configured by the vast majority of links in the Internet. IPv6 constrains this even further by specifying a minMTU of 1280 bytes and a minimum Maximum Reassembly Unit (minMRU) of 1500 bytes [RFC2460]. IPv4 specifies both minMTU/ minMRU as only 576 bytes [RFC0791][RFC1122], although it is widely assumed that the vast majority of nodes will configure an IPv4 minMRU of at least 1500 bytes.

The 1280 IPv6 minMTU originated from a November 14, 1997 mailing from Steve Deering to the IPng mailing list, which stated:

> "In the ipngwg meeting in Munich, I proposed increasing the IPv6 minimum MTU from 576 bytes to something closer to the Ethernet MTU of 1500 bytes, (i.e., 1500 minus room for a couple layers of encapsulating headers, so that min- MTU-size packets that are tunneled across 1500-byte-MTU paths won't be subject to fragmentation/reassembly on ingress/egress from the tunnels, in most cases).
>
> ...
>
> The number I propose for the new minimum MTU is 1280 bytes (1024 + 256, as compared to the classic 576 value which is 512 + 64). That would leave generous room for encapsulating/tunnel headers within the Ethernet MTU of 1500, e.g., enough for two layers of secure tunneling including both ESP and AUTH headers."

However, there was a fundamental flaw in this reasoning . In particular to avoid fragmentation for several nested layers of encapsulation, the first tunnel (T1) would have to set a 1280 MTU so that its tunneled packets would emerge as 1320 bytes (1280 bytes plus 40 bytes for the encapsulating IPv6 header). Then, the next tunnel (T2) would have to set a 1320 MTU so its tunneled packets would emerge as 1360. Then the next tunnel (T3) would have to set a 1360

MTU so that its tunneled packets would emerge as 1400, etc. until the available path MTU is exhausted. The question is, how can those nested tunnels be so carefully coordinated so that there would never be an MTU infraction? In a single administrative domain where an operator can lay hands on every tunnel ingress this may be possible, but in the general case it cannot be expected that the nested tunnel MTUs would be so well orchestrated. It is therefore necessary to consider as a limiting condition a tunnel that configures a 1280 MTU in which the tunnel crosses a link (perhaps another tunnel) that also configures a 1280 MTU. In that case, the tunnel ingress has two choices: 1) perform fragmentation that the tunnel egress needs to reassemble, or 2) shut down the tunnel due to failure to meet the IPv6 minMTU requirement.

In addition, it is becoming increasingly evident that Path MTU Discovery (PMTUD) [RFC1981] does not work properly in all cases. This is due to the fact that the Packet Too Big (PTB) messages required for PMTUD can be lost due to network filters that block ICMPv6 messages [RFC2923][WAND][SIGCOMM][RIPE]. It is therefore necessary to consider the case where IPv6 packets are dropped silently in the network due to a size restriction, but the IPv6 source host never receives the necesary indication from the network that the packet was lost. The source host must therefore support some form of IP fragmentation in order to ensure that isolated large packets are delivered, as well as a packet size probing capabilitiy (see: [RFC4821]) to ensure that large packets that are part of a coordinated stream are making it through to the destination.

Due to these considerations, there are at least two use cases for network layer fragmentation that must be satisfied now and for the long term. In the following sections, we discuss these considerations in more detail.

3. IPv6 Hosts Sending Large Isolated Packets

IPv6 hosts that send large isolated packets have no way of ensuring that the packets are delivered to the final destination if their size exceeds the path MTU. The host must therefore perform network layer fragmentation to a fragment size of no larger than 1280 bytes to ensure that the fragmented packets are delivered to the destination without loss due to a size restriction. However, the destination node need only configure a minMRU size of 1500 bytes per the IPv6 specs. Therefore, the source must either limit its packet sizes to 1500 bytes (i.e., before fragmentation) or somehow have a way of determining that the destination configures a larger minMRU. Two uses for this host-based fragmentation to support large isolated packets are OSPVFv3 and DNS.

4. IPv6 Tunnels

IPv6 tunnels are used for many purposes, including transition, security, mobility, routing control, etc. While it is assumed that transition mechanisms will eventually give way to native IPv6, it is clear that the use of tunnels for other purposes will continue and even expand. A long term strategy for dealing with tunnel MTUs is therefore required.

Tunnels may cross links (perhaps even other tunnels) that configue only the IPv6 minMTU of 1280 bytes while the tunnel ingress must be able to send packets that are at least 1280 bytes in length so that the IPv6 minMTU is extended to the source. However, these tunneled packets become (1280 + HLEN) bytes on the wire (where HLEN is the length of the encapsulating headers), meaning that they would be vulerable to loss at a link within the tunnel that configures a smaller MTU. Therefore, the only way to satisfy the IPv6 minMTU is through network layer fragmentation and reassembly between the tunnel ingress and egress, where the ingress fragments its tunneled packets that are larger than (1280 - HLEN) bytes.

Unfortunately, fragmentation and reassembly are a pain point for in-the-network routers - especially for those that are nearer the core of the network. It is therefore highly desirable for the tunnel ingress to discover whether this fragmentation and reassembly can be avoided. This can only be done by allowing the ingress to probe the path to the egress by sending whole 1500 byte probe packets to discover whether the probes can be delivered to the egress without fragmentation. These 1500 byte probes appear as (1500 + HLEN) bytes on the wire, therefore the path must support an MTU of at least this size in order for the probe to succeed.

The tunnel fragmentation and reassembly strategy is therefore as follows:

1. When the tunnel ingress receives a packet that is no larger than (1280-HLEN) bytes, it encapsulates the packet and sends it to the egress without fragmentation. The egress will receive the packet since it is small enough to fit within the IPv6 minMTU of 1280 bytes.

2. When the tunnel egress receives a packet that is larger than 1500 bytes, it encapsulates the packet and sends it to the egress without fragmentation. If the packet is lost in the network due to a size restriction, the ingress may or may not reeceive a PTB message which it can then forward to the original soruce. Whether or not a PTB message is received, however, it is the responsibility of the original source to ensure that its packets

larger than 1500 bytes are making it to the final destination by using a path probing technique such as specified by [RFC4821].

3. When the tunnel ingress receives a packet larger than (1280 - HLEN) but no larger than 1500 bytes, and it is not yet known whether packets of this size can reach the egress without fragmentation, the ingress encapsulates the packet and uses network layer fragmentation to fragment it into two pieces that are each signifiicantly smaller than (1280 - HLEN) bytes. At the same time, the tunnel ingress sends an unfragmented 1500 byte probe packet toward the egress (subject to rate limiting) which will appear as (1500 + HLEN) bytes on the wire. If the egress receives the probe, it informs the ingress that the probe succeeded. If the probe succeeds, the ingress can suspend the fragmentation process and send packets between (1280-HLEN) and 1500 bytes without using fragmentation. This probing process exactly parallels [RFC4821].

In this method, the tunnel egress must configure a slightly larger MRU than the minMRU specified for IPv6 in order to accommodate the HLEN bytes of tunnel encapsulation during reassembly. 2KB is recommended as the minMRU for this reason.

These procedures give way to the ability for the tunnel ingress to configure an unlimited MTU (theoretical limit is 2^16 bytes for IPv4 and 2^32 bytes for IPv6). They will therefore naturally lead to the Internet migrating to larger packet sizes with no dependence on traditional path MTU discovery. Operators will also soon discover that configuring larger MTUs on links between routers (e.g., 2KB or larger) will dampen the fragmentation and reassembly requirements until fragmentation and reassembly usage is gradually tuned out of the network.

These procedures are not supported by the existing IPv6 fragmentation procedures, however they are exactly those specified in the Subnetwork Encapsulation and Adaptation Layer (SEAL) [I-D.templin-intarea-seal]. Widespread adoption of SEAL will therefore naturally lead to an Internet which no longer places MTU restrictions on tunnels and therefore supports natural migration to unbounded packet sizes.

5. IANA Considerations

There are no IANA considerations for this document.

6. Security Considerations

The security considerations for [RFC2460] apply also to this document.

7. Acknowledgments

This method was inspired through discussion on various IETF mailing lists in the 2012-2013 timeframe.

8. References

8.1. Normative References

[RFC0791] Postel, J., "Internet Protocol", STD 5, RFC 791, September 1981.

[RFC1122] Braden, R., "Requirements for Internet Hosts - Communication Layers", STD 3, RFC 1122, October 1989.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

[RFC4443] Conta, A., Deering, S., and M. Gupta, "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, March 2006.

8.2. Informative References

[FRAG] Kent, C. and J. Mogul, "Fragmentation Considered Harmful", October 1987.

[I-D.templin-intarea-seal] Templin, F., "The Subnetwork Encapsulation and Adaptation Layer (SEAL)", draft-templin-intarea-seal-59 (work in progress), July 2013.

[RFC1981] McCann, J., Deering, S., and J. Mogul, "Path MTU Discovery for IP version 6", RFC 1981, August 1996.

[RFC2923] Lahey, K., "TCP Problems with Path MTU Discovery", RFC 2923, September 2000.

[RFC4821] Mathis, M. and J. Heffner, "Packetization Layer Path MTU Discovery", RFC 4821, March 2007.

[RIPE] De Boer, M. and J. Bosma, "Discovering Path MTU Black Holes on the Internet using RIPE Atlas", July 2012.

[SIGCOMM] Luckie, M. and B. Stasiewicz, "Measuring Path MTU Discovery Behavior", November 2010.

[WAND] Luckie, M., Cho, K., and B. Owens, "Inferring and Debugging Path MTU Discovery Failures", October 2005.

Author's Address

Fred L. Templin (editor)
Boeing Research & Technology
P.O. Box 3707
Seattle, WA 98124
USA

Email: fltemplin@acm.org