Network Working Group
Internet-Draft
Updates: RFC 3542 (if approved)
Intended status: Informational
Expires: July 25, 2012

M. Andrews
ISC

January 22, 2012

Forcing Fragmentation of IPv6 Packets
draft-andrews-6man-force-fragmentation-01

Abstract

Extend The Advanced Sockets API for IPv6 (RFC 3542) to provide a mechanism to force a Fragment header to be added to a packet.

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 July 25, 2012.

Copyright Notice

Table of Contents

1. Background

In order to avoid Path MTU Discover (MTUD) [RFC 1191] in IPv6 an application must not only force packets to be fragmented at the minimum IPv6 MTU it must also force the addition of a Fragment header to unfragmented packets otherwise PTB ICMPv6 may be sent if the packet goes through a IPv6 to IPv4 translating router [RFC 2460].

The Advanced Sockets API for IPv6 [RFC 3542] provides mechanisms to force fragmentation of a packet greater than the minimum IPv6 MTU (IPV6_USE_MIN_MTU). It also provides mechanisms to prevent fragmentation of a packet (IPV6_DONTFRAG). It however does not provide a mechanism to force a fragmentation header to be added. This document intends to add such a mechanism.

There appears to be 3 viable alternatives. 1) extend IPV6_USE_MIN_MTU to force the inclusion of a Fragment header. 2) extend IPV6_DONTFRAG to signal that a Fragment header needs to be included. 3) add a new socket option like IPV6_DOFRAG.

While there are multiple options available we should select exactly one.

1.1. Reserved Words

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].

2. Background

Path MTU Discover assumes that the application / network stack will be in the position to retransmit the packet apon reception of a PTB ICMPv6 message. While this is true of TCP, there is class of protocols for which this is not true, like DNS/UDP. DNS/UDP is a stateless single packet exchange protocol. The client asks a query of the server and the server responds, usually with a bigger response, then immediately forgets the request. This works well for IPv4 where routers fragment the packets. For IPv6, however, it requires the client to timeout as the PTB message is usually being sent to the server which has no memory of the packet that triggered the PTB message. Setting IPV6_USE_MIN_MTU to 1 on the UDP socket reduces but does not remove the problem as the "minimum IPv6 MTU" of 1280 is not a true minimum.

[RFC 2460] permits PTB messages with values smaller than 1280 and requires the sending node to add a fragmentation header to subsequent

packets. To avoid timeouts protocols, like DNS/UDP, need a way to force the addition of a fragment header to packets at or below the minimum IPv6 MTU.

3. Extend IPV6_USE_MIN_MTU

Currently IPV6_USE_MIN_MTU only requires a Fragmentation header to be added when the packet size is greater that 1280 byte, packets less that 1280 bytes are not required to have a Fragment header added. This change would result in a Fragmentation header also being added to packets that are 1280 bytes or less.

If IPV6_USE_MIN_MTU is set to 1 then a Fragment header MUST be added to both multicast and unicast packets.

If IPV6_USE_MIN_MTU is set to -1 then a Fragment header MUST be added to multicast packets.

Alternatively, one could use 2 and -2 to indicate the forced addition of a Fragment header in addition to limiting the MTU.

4. Extend IPV6_DONTFRAG

If IPV6_DONTFRAG is set to -1 (new value) then a Fragment header MUST be added to the packet.

5. Add IPV6_DOFRAG

This is similar to IPV6_USE_MIN_MTU in that it is a tri-state flag. When set to -1 a Fragment header MUST be added to multicast packets. When set to 1 a Fragment header MUST be added to both multicast and unicast packets. When set to 0 a Fragment header is only added if fragmentation would otherwise happen.

```
int on = 1;
setsockopt(fd, IPPROTO_IPV6, IPV6_DOFRAG, &on, sizeof(on));
```

Setting IPV6_DONTFRAG to 1 will force IPV6_DOFRAG to 0.

Setting IPV6_DOFRAG to 1 or -1 will force IPV6_DONTFRAG to 0.

6. IANA Considerations

No IANA Considerations.

7. Security Considerations

TBA

8. Normative References

[RFC 1191]
Mogul, J. and S. Deering, "Path MTU Discovery", RFC 1191, November 1990.

[RFC 2119]
Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

[RFC 3542]
Stevens, W., Thomas, M., Normark, E., and T. Jinmei, "Advanced Sockets Application Program Interface (API) for IPv6", RFC 2003, May 2003.

Author's Address

Mark Andrews
Internet Systems Consortium
950 Charter Street
Redwood City, CA 94063
US

Email: marka@isc.org