TCP Fails To Respect IPV6_USE_MINMTU
draft-andrews-tcp-and-ipv6-use-minmtu-00

Abstract

   The IPV6_USE_MINMTU socket option is used to set the maximum IPv6
   packet size to be used on a socket. Many implementations of TCP
   running over IPv6 neglect to check the IPV6_USE_MINMTU value when
   performing MSS negotiation and when contructing a TCP segment. This
   lead to oversized IPv6 packets being sent resulting in unintended
   Path Maximum Transport Unit Discovery (PMTUD) being performed and to
   fragmented IPv6 packets being sent.

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 April 17, 2016.

Copyright Notice

Table of Contents

1. Introduction

The IPV6_USE_MINMTU [RFC3542] socket option is used to set the
maximum IPv6 packet size to be used on a socket. Many
implementations of TCP running over IPv6 neglect to check this value
when performing MSS negotiation and when contructing a TCP segment.
This lead to oversized IPv6 packets being sent resulting unintended
PMTUD [RFC1191] being performed and to fragmented IPv6 packets being
sent.

TCP when running over IPv6 SHOULD check the state of the
IPV6_USE_MINMTU when performing MSS negotiation. TCP implementions
already use learnt PMTU and interface MTU when performing MSS
negotiation.

TCP, when running over IPv6, SHOULD check the state of the
IPV6_USE_MINMTU when calculating the segment size to send. TCP
implementions already use learnt PMTU and interface MTU when
performing calculating the segment size to send.

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 [RFC2119].

2. MSS Negotiation

TCP when running over IPv6 SHOULD check the state of the
IPV6_USE_MINMTU when performing MSS negotiation. If the value of
IPV6_USE_MINMTU is one (1) then the application has requested that
PMTUD not be performed on the is socket and that IPv6 packets be sent
at a size no greater then the network minumum MTU of 1280 bytes.
This means that the TCP MSS negotiation size SHOULD be no bigger than

1220 (1280 - 40 - 20) to account for the IPv6 header and the TCP
header and MAY be smaller.

If this negotiation is properly performed then PMTUD of reply traffic
should not normally occur.

3. Segment Size Calculation

TCP when running over IPv6 SHOULD check the state of the
IPV6_USE_MINMTU when calculation the next segment to send. If the
value of IPV6_USE_MINMTU is one (1) them the maximum segment size
SHOULD be 1220.

If the TCP layer neglects to check the value of IPV6_USE_MINMTU and
it is one (1), the packet, when passed to the IPv6 layer, will be
fragmented if the resulting packet is bigger that 1280 octets. This
can result in communications failures due too itermediate nodes not
passing fragmented packets.

4. Current Usage

The IPV6_USE_MINMTU and TCP are used together in DNS nameservers as
TCP message streams are normally no more than a couple of IPv6
packets, there are lots of servers / clients and there are external
time contraints where recovery from lost ICMPv6 PTB will fall outside
of the time constraint window.

5. Normative References

[RFC1191] Mogul, J. and S. Deering, "Path MTU discovery", RFC 1191,
           DOI 10.17487/RFC1191, November 1990,
           <http://www.rfc-editor.org/info/rfc1191>.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
           Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/
           RFC2119, March 1997,
           <http://www.rfc-editor.org/info/rfc2119>.

[RFC3542] Stevens, W., Thomas, M., Nordmark, E., and T. Jinmei,
           "Advanced Sockets Application Program Interface (API) for
           IPv6", RFC 3542, DOI 10.17487/RFC3542, May 2003,
           <http://www.rfc-editor.org/info/rfc3542>.

Author's Address

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

   Email: marka@isc.org