INTERNET-DRAFT
Intended Status: Standard
Expires: November 2017

T. Herbert
Quantonium

May 10, 2017

ICMPv6 errors for discarding packets due to processing limits
draft-herbert-6man-icmp-limits-01

Abstract

Network nodes may discard packets if they are unable to process protocol headers of packets due to processing constraints or limits. When such packets are dropped, the sender receives no indication so it cannot take action to address the cause of discarded packets. This document defines ICMP errors that can be sent by a node that discards packets because it is unable to process the protocol headers. A sender that receives such an ICMP error may be able to modify what it sends in future packets to avoid subsequent packet discards.

Table of Contents

1 Introduction

This document specifies ICMP Parameter Problem type errors that can sent when a node discards a packet due to it being unable to process the necessary protocol headers because of processing constraints and limits.

Four of the errors are specific to processing limits of extension headers; another error is used when the aggregate protocol headers in a packet exceed the processing limits of a node.

1.1 Extension header limits

With IPv6, optional internet-layer information is carried in one or more IPv6 Extension Headers [RFC2460]. Extension Headers are placed between the IPv6 header and the Upper-Layer Header in a packet. The term "Header Chain" refers collectively to the IPv6 header, Extension Headers, and Upper-Layer Header occurring in a packet. Individual extension headers may have a length of 2048 and must fit into one MTU. Destination Options and Hop by Hop Options contain a list options in Type-length-value (TLV) format. Each option includes a length of the data field in octets and the minimum size of a (non-pad) option is two bytes and the maximum length is 257 bytes. The number of options in an extension header is only limited by the length of the extension header and MTU. Options may also be skipped over by a receiver if they are unknown and the Option Type indicates to skip (first two bits are 00).

Per [RFC2460], except for Hop by Hop options, extension headers are not examined or processed by intermediate nodes. Many intermediate nodes, however, do examine extension header for various purposes. For instance, a node may examine all extension headers to locate the transport header of packet in order to implement transport layer filtering or to track connections to implement a stateful firewall.

Destination hosts are expected to process all extensions headers and options in Hop by Hop and Destination Options.

Due to the variable lengths and high limits of lengths of extension header and chains, many devices have operational limits of extension headers in packets they can process. [RFC7045] discusses the requirements of intermediate nodes that discard packets because of unrecognized extension headers. When a limit is exceeded, the typical behavior is to silently discard a packet. The limits are non-standard and may be configurable per implementation. Both intermediate nodes and end hosts may institute such limits on extension header processing.

This document defines three Parameter Problem codes and extends the applicably of an existing code that are sent by a node that discards a packet due to processing limits of extension headers being exceeded. A source host that receives an ICMP error can modify the use of extension headers in subsequent packets to the destination in order to avoid further occurrences of packets with extension headers being discarded.

1.2 Aggregate header limits

Many hardware devices implement a parsing buffer of a fixed sized to process packets. The parsing buffer is expected to contain all the headers (often up to a transport layer header for filtering) that a device needs to examine. Parsing buffers have been implemented with various sizes (512 is common, some devices have smaller sizes).

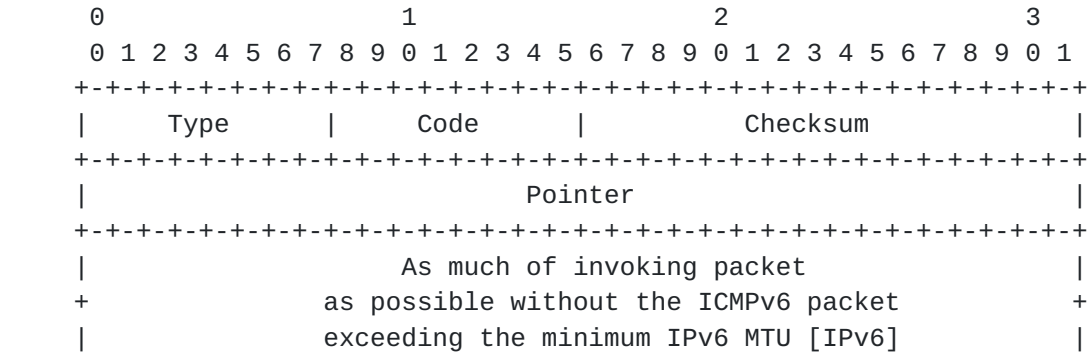
When the aggregate length of headers in a packet exceeds the size of the parsing buffer, a device will typically either discard the packet or defer processing to a software slow path. In either case, no indication of a problem is sent back to the sender.

This document defines one code for ICMPv6 Parameter Problem type that is sent by a node that is unable to process the headers of a packet due to the aggregate size of the packet headers exceeding a processing limit (e.g. exceeding the size of a parsing buffer). A source host that receives an ICMP error can modify the headers used in subsequent packets to try to avoid further occurrences of packets being discarded or relegated to a slow path.

2 ICMP message format

The ICMP errors defined in this document are Parameter Problem messages. Four new codes are defined for Parameter Problem type and applicability of one existing code is extended.

The format of the ICMP message is:



IPv6 Fields:

Destination Address
Copied from the Source Address field of the invoking packet.

ICMPv6 Fields:

Type
4 (Parameter Problem type)

Code (pertinent to this specification)
1 - Unrecognized Next Header type encountered
4 - Extension header too big
5 - Extension header chain too long
6 - Too many options in extension header
7 - Headers too long

Pointer
Identifies the octet offset within the invoking packet where a limit was exceeded.

The pointer will point beyond the end of the ICMPv6 packet if the field exceeding the limit is beyond what can fit in the maximum size of an ICMPv6 error message.

3 Descriptions of codes

3.1 Unrecognized Next Header type encountered (code 1)

[RFC2460] specifies that a destination host should send an "unrecognized next header type" when a Next Header value is unrecognized in a packet. This document extends this to allow intermediate nodes to send this same error for a packet that is discarded because a node does not recognize a Next Header type.

This code SHOULD be sent by an intermediate node that discards a packet because it encounters a Next Header type that is unknown in its examination. The ICMP Pointer field is set to the offset of the unrecognized value within the original packet.

Note that when the original sender receives the ICMP error it can differentiate between the message being sent by a destination host, per [RFC2460], and an error sent by an intermediate host based on matching the source address of the ICMP packet and the destination address of the packet in the ICMP data.

3.2 Extension header too big (code 4)

An ICMP Parameter Problem with code for "extension header too big" SHOULD be sent when a node discards a packet because the size of extension exceeds its processing limit. The ICMP Pointer field should be set to the offset of length field for the extension header that is too big.

3.3 Extension header chain too long (code 5)

An ICMP Parameter Problem with code for "extension header chain too long" SHOULD be sent when a node discards a packet with an extension header chain because an extension header chains exceeds it processing limit. The ICMP Pointer field should be set to the offset of the first octet that exceeds the limit.

Note there are two different limits that might be applied: a limit on the total size in octets of the header chain, and a limit on the number of extension headers in the chain. This error code is used in both cases. In the case that the an octet limit is exceeded, the ICMP Pointer should be set to first octet beyond the limit. In the case that the number of extension headers is exceeded, the ICMP Pointer should be set to the offset of first octet of the first extension header that is beyond the limit.

3.4 Too many options in extension header (code 6)

An ICMP Parameter Problem with code for "too many options in extension header" SHOULD be sent when a node discards a packet with an extension header that has a number of options that exceed the processing limits of the node. This code is applicable for Destination options or Hop by Hop options. The ICMP Pointer field should be set to the first octet of the first option that exceeds the limit.

3.5 Headers too long (code 7)

An ICMP Parameter Problem with code for "headers too long" SHOULD be sent when a node discards a packet because the aggregate length of headers in the packet exceeds the processing limits of the node. The ICMP Pointer should be set to the offset of the first octet that exceeds the limit.

4 Host response

When a source host receives an ICMP Parameter Problem error for one of the codes described in section 3, it SHOULD verify the ICMP error is valid and take an appropriate action. Possible actions are:

* The error SHOULD be logged with sufficient detail for

debugging packet loss. The details of the error, including the addresses and the offending extension header or data, should be retained. This would be useful for instance to debug when a node is mis-configured and unexpectedly discarding packets, or when a new extension header is being deployed.

* An error SHOULD be reported to an application if the application enabled extension headers for its traffic. The application MAY either terminate a connection if extension headers are required, stop using extension headers in packets to the destination indicated in packet of the ICMP error, or attempt modify its use of extension headers or headers to avoid the packet drop.

* A host system SHOULD take action if it is automatically inserting extension headers into packets unbeknownst to the application. The host system SHOULD either stop using extension headers or modify its used of extension headers for subsequent packets sent to the destination indicated in the packet of the ICMP error.

5 Security Considerations

This document does not introduce any new security concerns for use of ICMP errors. The security considerations for ICMPv6 described in [RFC4443] are applicable.

6 IANA Considerations

IANA is requested to assign the following codes for ICMPv6 type 4 "Parameter Problem":

4 - Extension header too big

5 - Extension header chain too long

6 - Too many options in extension header

7 - Headers too long

7 References

7.1 Normative References

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

[RFC7045] Carpenter, B. and S. Jiang, "Transmission and Processing of IPv6 Extension Headers", RFC 7045, DOI 10.17487/RFC7045, December 2013, <http://www.rfc-editor.org/info/rfc7045>.

[RFC4443] Conta, A., Deering, S., and M. Gupta, Ed., "Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 4443, DOI 10.17487/RFC4443, March 2006, <http://www.rfc-editor.org/info/rfc4443>.

7.2 Informative References

[I-D.ietf-6man-rfc2460bis] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", draft-ietf-6man-rfc2460bis-05.

Author's Address

Tom Herbert
Quantonium
Santa Clara, CA
USA

Email: tom@herbertland.com