INTERNET-DRAFT
Intended Status: Standard
Expires: November 2018

T. Herbert
Quantonium

May 29, 2018

ICMPv6 errors for discarding packets due to processing limits
draft-ietf-6man-icmp-limits-00

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 ICMPv6 errors that can be sent by a node that discards packets because it is unable to process the protocol headers. A node that receives such an ICMPv6 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 ICMPv6 errors that can be sent when a node discards a packet due to it being unable to process the necessary protocol headers because of processing constraints or limits. New ICMPv6 code points are defined as an update to [RFC4443].

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 [RFC8200]. 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 Headers 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 an option (non-pad type) 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 [RFC8200], 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, high limits of lengths of extension headers, or potential for Denial of Service attack; many devices impose 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 ICMPv6 error can modify the use of extension headers in subsequent packets to the destination in order to avoid further occurrences of packets 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 bytes 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.

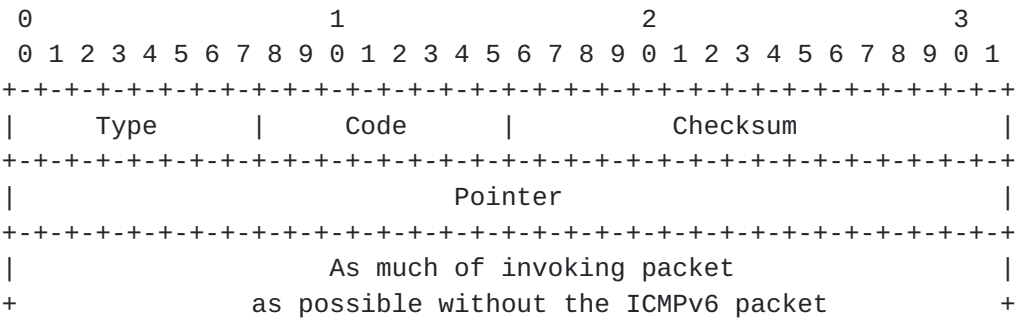
This document defines one code for ICMPv6 Destination Unreachable that is sent by a node that is unable to process the headers of a packet due to the aggregate size of the packet headers exceeding a processing limit (e.g. exceeding the size of a parsing buffer). A source host that receives an ICMPv6 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 ICMPv6 errors for extension header limits

Three new codes are defined for Parameter Problem type and applicability of one existing code is extended for ICMPv6 errors for extension header limits.

2.1 Format

The format of the ICMPv6 message for an extension header limit exceeded error is:



| exceeding the minimum IPv6 MTU [IPv6] |

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

Pointer
Identifies the octet offset within the invoking packet where the problem occurred.

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

2.2 Unrecognized Next Header type encountered (code 1)

[RFC8200] 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 ICMPv6 Pointer field is set to the offset of the unrecognized value within the original packet.

Note that when the original sender receives the ICMPv6 error it can differentiate between the message being sent by a destination host, per [RFC4443], and an error sent by an intermediate host based on matching the source address of the ICMPv6 packet and the destination address of the packet in the ICMPv6 data.

2.3 Extension header too big (code 4)

An ICMPv6 Parameter Problem with code for "extension header too big"

SHOULD be sent when a node discards a packet because the size of an extension header exceeds its processing limit. The ICMPv6 Pointer field is set to the offset of the first octet in the extension header that exceeds the limit.

2.4 Extension header chain too long (code 5)

An ICMPv6 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.

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 a size limit is exceeded, the ICMPv6 Pointer is set to first octet beyond the limit. In the case that the number of extension headers is exceeded, the ICMPv6 Pointer is set to the offset of first octet of the first extension header that is beyond the limit.

2.5 Too many options in extension header (code 6)

An ICMPv6 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 ICMPv6 Pointer field is set to the first octet of the first option that exceeds the limit.

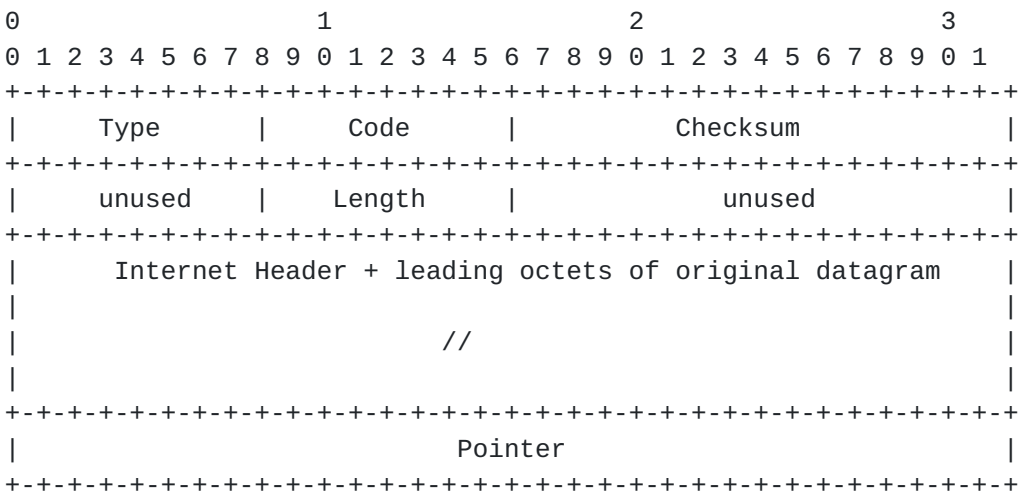
3 ICMPv6 error for aggregate header limits

One code is defined for Destination Unreach type for aggregate header limits.

3.1 Format

The error for aggregate header limits employs a multi-part ICMPv6 message format as defined in [RFC4884]. The extended structure contains a pointer to the octet beyond the limit.

The format of the ICMPv6 message for an aggregate header limit exceeded is:



IPv6 Fields:

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

ICMPv6 Fields:

Type
1 (Destination Unreachable type)

Code (pertinent to this specification)
8 - Headers too long

Length
Length of the "original datagram" measured in 64 bit words

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

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

3.2 Usage

An ICMPv6 Destination Unreachable error 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 Pointer in the extended ICMPv6 structure is set to the offset of the first octet that exceeds the limit.

4 Operation

Nodes that send or receive ICMPv6 errors due to header processing limits MUST generally comply with ICMPv6 processing as specified in [RFC4443].

4.1 Priority of reporting

More than one ICMPv6 error may be applicable to report for a packet. For instance, the number of extension headers in a packet might exceed a limit, and the aggregate length of protocol headers might also exceed a limit. Only one ICMPv6 error should be sent for a packet, so a priority is defined to determine which error to report.

The reporting priority of ICMPv6 errors for processing limits is from highest to lowest priority:

1) Real error (existing codes)

2) Unrecognized Next Header type encountered by an intermediate node

3) Too many options in an extension header

4) Extension header too big

5) Extension header chain too long for number of extension headers limit exceeded

6) Extension header chain too long for size of the extension header chain exceeding a limit

7) Headers too long

4.2 Host response

When a source host receives an ICMPv6 error for a processing limit being exceeded, it SHOULD verify the ICMPv6 error is valid and take an appropriate action.

The ICMPv6 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.

A host MAY modify its usage of protocol headers in subsequent packets

to avoid repeated occurrences of the same error.

For ICMPv6 errors cause by extension header limits being exceeded:

* 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 ICMPv6 error, or attempt modify its use of extension headers or headers to avoid the packet drop.

* A host system SHOULD take appropriate action if it is automatically inserting extension headers into packets unbeknownst to the application. If the offending extension header is not required for communication, the host MAY either stop sending it or otherwise modify its use in subsequent packets sent to the destination indicated in the ICMPv6 error.

5 Security Considerations

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

6 IANA Considerations

6.1 Parameter Problem codes

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

6.2 Destination Unreachable codes

IANA is requested to assign the following codes for ICMPv6 type 1 "Destination Unreachable":

8 - Headers too long

8 Acknowledgments

The authors would like to thank Ron Bonica, Bob Hinden for their comments and suggestions that improved this document.

7 References

7.1 Normative References

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[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>.

[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>.

[RFC4884] Bonica, R., Gan, D., Tappan, D., and C. Pignataro, "Extended ICMP to Support Multi-Part Messages", RFC 4884, DOI 10.17487/RFC4884, April 2007, <https://www.rfc-editor.org/info/rfc4884>.

7.2 Informative References

Author's Address

Tom Herbert
Quantonium
Santa Clara, CA
USA

Email: tom@qantonium.net