IPv4 Hop-by-Hop Options and Destination Options Extension Headers
draft-herbert-ipv4-hbh-destopt-00

Abstract

This specification enables the use of Hop-by-Hop Options and Destination Options extension headers which are defined for IPv6 to be used with IPv4. The goal is to provide a uniform and feasible method of extensibility that is common between IPv4 and IPv6.

Status of this Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/idabstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

Copyright and License Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of

T. Herbert Expires March 27, 2020 [Page 1]
publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1 Introduction .................................................. 3
  1.1 Motivation .................................................. 3
  1.2 Enabling IPv4 extension headers ............................. 4
2 Format .......................................................... 5
  2.1 Option format ................................................ 6
  2.2 Extension header formats ................................... 6
    2.2.1 Hop-by-Hop Options ................................... 6
    2.2.2 Destination Options Header ............................. 6
  2.3 Extension Header Order ..................................... 6
  2.4 Experimental options ....................................... 7
3 ICMP errors ..................................................... 7
  3.1 Parameter Problem codes .................................. 7
  3.2 Destination Unreachable codes ............................... 9
4 Requirements and operation .................................... 10
  4.1 Requirements ............................................... 10
  4.2 Interaction with standard IPv4 mechanisms .................. 11
    4.2.1 IPv4 options ........................................... 11
    4.2.2 IPv4 fragmentation .................................... 11
  4.3 Processing limits .......................................... 11
5 Deployability .................................................. 12
6 Security Considerations ....................................... 13
7 IANA Considerations ........................................... 13
  7.1 Protocol descriptions ...................................... 13
  7.2 IPv4 Parameters registry ................................... 13
  7.3 ICMP parameters ........................................... 15
    7.3.1 Parameter Problem codes ............................... 15
    7.3.2 Destination Unreachable codes ......................... 15
8 References ..................................................... 16
  8.1 Normative References ...................................... 16
  8.2 Informative References .................................... 16
Author’s Address ................................................. 18
1 Introduction

This specification enables the Hop-by-Hop Options and Destination Options extension headers which are defined for IPv6 to be used with IPv4. The goal is to provide an extensible mechanism in IPv4 that is unified with IPv6 and facilitates leveraging common protocol and implementation for extensibility between the two versions of the Internet Protocol.

Hop-by-Hop Options and Destination Options extension headers are defined in [RFC8200]. The respective IP protocol numbers, 0 and 60, are defined for use with IPv6 as Next Header values, but are reserved for IPv4 and are not valid values in the IPv4 Protocol field. This document permits the use of protocol numbers 0 and 60 in the IPv4 protocol field and defines the semantics of processing Hop-by-Hop Options and Destination Options extension headers in the context of IPv4. Note that the use of the Hop-by-Hop Options and Destination Options protocol numbers the IPv4 Protocol field effectively designates the field to be the IPv4 Next Header field.

The use of extension headers in IPv4 is not without precedent. Both the Authentication header (AH, protocol number 51) [RFC4302] and Encapsulating Security Payload (ESP, protocol number 50) [RFC4303] are defined for use with IPv4.

Note that this specification only enables the use of Hop-by-Hop Options and Destination Options extension headers in IPv4. It does not define the use of any other extension headers with IPv4 including the Routing Header and Fragmentation Header.

1.1 Motivation

IPv6 is intended to become the standard protocol of the Internet, however it is clear that there is a large segment of users that will be using IPv4 for the foreseeable future. This is particularly true in many enterprises where a business case for transitioning to IPv6 hasn’t yet emerged [V6STATE].

In lieu of sun-setting IPv4 and expecting all users to move to IPv6 in some time frame that is unlikely to be met, this specification suggests an alternative to enable the extensibility mechanisms of IPv6 in IPv4. The rationale for this is two fold:

1) Users benefit from forward looking features being actively defined and developed for IPv6 without requiring them to first transition to IPv6.
2) In making IPv4 look more like IPv6, the work required to complete a future transition to IPv6 may be reduced or simplified.

Various proposals that would use Hop-by-Hop Options or Destination Options are currently being developed in IETF. These include Path MTU Option [MTUOPT], In-situ OAM [IOAM], Service-aware IPv6 Network [SAIN], and Firewall and Service Tickets [FAST]. These proposals leverage the extensibility mechanisms of extension headers defined for IPv6. These proposals, in some form, could be of value for use with IPv4.

Unfortunately, IPv4 does not define a mechanism that meets reasonable requirements for extensibility. IP options are quite limited and have long been considered obsolete. There have been proposals for encoding host to network signaling in UDP (e.g. [SPUD], IOAM over encapsulation in Geneve [IOAMGEN]), however these are shown to neither be generic nor robust especially in the case that encapsulated data must be modified in flight [RFC7605].

The proposal in this document is to enable IPv4 packets to carry Hop-by-Hop Options and Destination Options extension headers in the same manner that IPv6 does. In doing so, the various options defined for IPv6 can be used with IPv4 to the benefit of the user. It is expected that in many cases, such as the IOAM and Path MTU options, options are protocol agnostic and would be applicable to use in IPv4 with little or no change. In other cases, particularly when an option carries an IP address, the options might be defined to be IPv6 specific and require some adaptation to work with IPv4.

1.2 Enabling IPv4 extension headers

IPv4 options were defined in [RFC0791] as the means of extending the IP protocol. IPv4 options have not been successful. Early router implementations, and even those today, either don’t process IPv4 options or relegate them to a slow path effectively making them unusable for serious applications. IPv4 options are limited to forty bytes length and, unlike TCP options, no IP options have been defined that are critical to communications. The upshot is that IPv4 options have long not been considered an option for deployment [IPNOOP].

IPv6 took a different approach. Extensibility of IPv6 is provided by extension headers. Optional internet-layer information is encoded in separate headers that may be placed between the IPv6 header and the upper-layer header in a packet [RFC8200]. IPv6 extension headers have had mixed success in deployment in that some intermediate devices have trouble processing them [RFC7872], however there are several active proposals in IETF that would make use of them (e.g. [FAST]),
Using extension headers with IPv4 is logically straightforward. The IPv4 Protocol field is effectively re-designated to be a Next Header field with the same meaning and semantics as the IPv6 Next Header field. In this manner, an IPv4 packet can contain IPv6 extension headers that are recast as IPv4 extension headers. This specification describes the use of Hop-by-Hop Options and Destination Options extension headers in IPv4.

A number of ICMP errors may be sent when an error condition is encountered in processing extension headers. The relevant ICMPv6 errors are defined in [RFC4443] and [ICMPLIM]. This specification adapts these ICMPv6 errors for use in IPv4.

2 Format

IPv4 extension headers are optional internet-layer information encoded in separate headers that may be placed between the IPv4 header and the upper-layer header in a packet. IPv4 Hop-by-Hop Options and Destination Options extension headers are based on the corresponding IPv6 extension headers and share the same basic properties and semantics [RFC8200].

As illustrated in these examples, an IPv4 packet MAY carry zero, one, or more extension headers, each identified by Protocol field of the IPv4 header or the Next Header field of a preceding extension header:

```
+---------------+------------------------
|  IPv4 header  | TCP header + data      |
| Protocol =    | TCP                    |
+---------------+------------------------

+---------------+------------------------+------------------------
|  IPv4 header  | Hop-by-Hop             | TCP header + data      |
| Protocol =    | Next Header =          | TCP                    |
| Hop-by-Hop    | TCP                    |                         |
+---------------+------------------------+------------------------

+---------------+------------------------+------------------------+------------------------
|  IPv4 header  | Hop-by-Hop             | Destination Opt        | TCP header + data      |
| Protocol =    | Next Header =          | Next Header =          | TCP                    |
| Hop-by-Hop    | DestOpt                |                         |                         |
+---------------+------------------------+------------------------+------------------------
```
2.1 Option format

The format and processing of options in IPv4 Hop-by-Hop Options and Destination Options extension headers is the same as that defined for IPv6 options in section 4.2 of [RFC8200] with the following modifications:

* In the case that an option is unrecognized by a receiver and the highest-order 2 bits specify that an ICMP error should be sent (values of 10 and 11) then an ICMPv4 Parameter Problem, Code 5 is sent (see section 3).

Note that PAD1 and PADN are defined for IPv4 Hop-by-Hop Options and Destination Options with the same format and semantics as defined for IPv6.

2.2 Extension header formats

2.2.1 Hop-by-Hop Options

The format of the IPv4 Hop-by-Hop Options extension header is the same as the corresponding format of IPv6 Hop-by-Hop Options defined in section 4.3 of [RFC8200] with the following modifications:

* The Next Header field MUST contain a protocol number that is defined to be usable with IPv4 or 60 (IPv4 Destination Options extension header).

2.2.2 Destination Options Header

The format of the IPv4 Destination Options extension header is the same as the corresponding format of IPv6 Destination Options defined in section 4.6 of [RFC8200] with the following modifications:

* The Next Header field MUST contain a protocol number that is defined to be usable with IPv4 or 60 (IPv4 Destination Options extension header).

2.3 Extension Header Order

When more than one IPv4 extension header is used in the same packet, it is RECOMMENDED that those headers appear in the following order:

IPv4 header
Hop-by-Hop Options header
Destination Options header (note 1)
Authentication header (note 2)
Encapsulating Security Payload header (note 2)
Upper-Layer header

note 1: unlike IPv6, routing headers are not defined for IPv4 so there is no distinction between Destination Options that appear before or after the routing header.

note 2: additional recommendations regarding the relative order of the Authentication and Encapsulating Security Payload headers are given in [RFC4303].

2.4 Experimental options

This document assigns a single option type for experimental purposes, with all possible values of the "act" and "chg" fields, resulting in eight distinct option type codes. These are the same values defined for experimental Hop-by-Hop and Destination Options in IPv6 [RFC4727].

Experimental IPv4 Hop-by-Hop Options and Destination Options Types are:

<table>
<thead>
<tr>
<th>HEX</th>
<th>act</th>
<th>chg</th>
<th>rest</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x1e</td>
<td>00</td>
<td>0</td>
<td>11110</td>
</tr>
<tr>
<td>0x3e</td>
<td>00</td>
<td>1</td>
<td>11110</td>
</tr>
<tr>
<td>0x5e</td>
<td>01</td>
<td>0</td>
<td>11110</td>
</tr>
<tr>
<td>0x7e</td>
<td>01</td>
<td>1</td>
<td>11110</td>
</tr>
<tr>
<td>0x9e</td>
<td>10</td>
<td>0</td>
<td>11110</td>
</tr>
<tr>
<td>0xbe</td>
<td>10</td>
<td>1</td>
<td>11110</td>
</tr>
<tr>
<td>0xde</td>
<td>11</td>
<td>0</td>
<td>11110</td>
</tr>
<tr>
<td>0xfe</td>
<td>11</td>
<td>1</td>
<td>11110</td>
</tr>
</tbody>
</table>

3 ICMP errors

ICMP errors are defined to be sent in response to errors that occur in processing extension headers. Six ICMPv4 Parameter Problem codes are defined and one ICMPv4 Destination Unreachable code is defined. These codes are adaptations of ICMPv6 codes defined in [RFC4443] and [ICMPLIM].

3.1 Parameter Problem codes

The format for ICMP Parameter Problem errors related to extension headers employs a multi-part ICMPv4 message format as defined in [RFC4884]. The extended structure contains a pointer to the octet beyond the limit.
The format of the ICMPv4 Parameter Problem for extension headers is:

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     unused    |    Length     |             unused            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Internet Header + leading octets of original datagram |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                            Pointer                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

IPv4 Header Fields:

- **Destination Address**: Copied from the Source Address field of the invoking packet.

ICMPv4 Fields:

- **Type**: 12 (Parameter Problem Message)

- **Code**: (pertinent to this specification)
  
  - 3 - Erroneous header field encountered
  - 4 - Unrecognized Next Header type encountered
  - 5 - Unrecognized IPv4 option encountered
  - 6 - Extension header too big
  - 7 - Extension header chain too long
  - 8 - Too many options in extension header
  - 9 - Option too big

- **Length**: Length of the padded "original datagram" field, measured in 32-bit words.

- **Pointer**: Identifies the octet offset within the invoking packet where the error was detected.

Codes 3, 4, and 5 are analogues of Parameter Problem codes 0, 1, and 2 defined for IPv6 in [RFC4443]. Operation and semantics of these
codes are the same as their counterparts in [RFC4443] with the following modifications:

* The multi-part ICMP format of [RFC4884] is used and its fields are set appropriately.

* The pointer to the offending byte in the invoking packet is contained in the 32 bit Pointer field in the extended format.

Codes 6, 7, 8, and 9 are analogues of Parameter Problem codes 4, 5, 6, and 7 defined for IPv6 in [ICMPLIM]. Operation and semantics of these codes are the same as their counterparts in [ICMPLIM] with the following modifications:

* The multi-part ICMP format of [RFC4884] is used and its fields are set appropriately.

* The pointer to the offending byte in the invoking packet is contained in the 32 bit Pointer field in the extended format.

3.2 Destination Unreachable codes

This specification defines one IPv4 Destination Unreachable code for aggregate header limits being exceeded as described in [ICMPLIM]. The error for aggregate header limits employs a multi-part ICMPv4 message format as defined in [RFC4884]. The extended structure contains a pointer to the octet beyond the limit.

The format of the ICMPv4 message for an aggregate header limit exceeded is:

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|     unused    |    Length     |             unused            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|      Internet Header + leading octets of original datagram    |
|                                                               |
|                           //                                  |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|                            Pointer                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```
IPv4 Header Fields:

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

ICMPv4 Fields:

Type
3 (Destination Unreachable Message)

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

Length
Length of the padded "original datagram" field, measured in 32-bit words.

Pointer
Identifies the octet offset within the invoking packet where the error was detected.

Code 16 is an analogue of Destination Unreachable code 8 defined in [ICMPLIM]. Operation and semantics of the code should be the same as the counterpart in [ICMPLIM].

4 Requirements and operation

4.1 Requirements

IPv4 Hop-by-Hop and Destination Options extension headers normatively assume the requirements of IPv6 extension headers as defined in [RFC8200] section 4, with the following modifications:

* References to the IPv6 header are replaced by references to the IPv4 header.

* ICMP errors sent in the course of processing Hop-by-Hop Options and Destination Options extension headers use ICMPv4 instead of ICMPv6. Applicable ICMPv4 errors for extension header processing are specified in section 3.

* The IPv4 header Protocol field assumes the same role and semantics with respect to extension headers as the IPv6 Next Header field.

* The Hop-by-Hop Options header is used to carry optional information that MAY be examined and processed by any node along
a packet’s delivery path.

* If a legacy IPv4 destination node, one that does not support IPv4 Hop-by-Hop Options or Destination Options extension headers, receives a packet with those extension headers then the packet will be processed as having an unknown protocol. It is expected that the packet will be discarded and an ICMP error may be generated.

* References to the Payload Length are reinterpreted as being the computed IPv4 payload length (i.e. IPv4 Total Length minus the length of the IPv4 header).

### 4.2 Interaction with standard IPv4 mechanisms

IPv4 Hop-by-Hop Options and Destination Options extension headers may be used concurrently with IPv4 mechanisms such as IPv4 options and IPv4 fragmentation. This section discusses the interactions.

**4.2.1 IPv4 options**

An IPv4 packet MAY contain both IPv4 options and Hop-by-Hop Options or Destination Options extension headers. IPv4 options are completely independent of IPv4 extension headers. IPv4 options MUST be processed before processing any extension headers per normal requirements of processing the IP header before the IP payload.

**4.2.2 IPv4 fragmentation**

A packet containing Hop-by-Hop Options and Destination Options extension headers may be fragmented using standard IPv4 fragmentation.

At a destination, if a received packet was fragmented by standard IPv4 fragmentation, it MUST be reassembled before processing any IPv4 extension headers.

If an IPv4 packet containing Hop-by-Hop Options is fragmented using standard IPv4 fragmentation, the Hop-by-Hop Options are not set in each of the packet fragments. An intermediate node MAY process the Hop-by-Hop options in the first fragment if the complete Hop-by-Hop extension header is contained within the fragment.

### 4.3 Processing limits

Section 5.3 of [RFC8504] describes the use of limits in processing extension headers in order to protect a node from excessive extension header options. These limits are adapted for use with IPv4 extension...
headers. The requirements in [ICMPLIM] for sending and processing ICMP errors related to processing limits are applicable.

A node MAY impose limits on processing IPv4 Destination Options and Hop-by-Hop Options extension headers. This includes limits on length or number of consecutive padding options, disallowing unknown options, and limits on the number of options or length of options. If a limit is exceeded in processing a received packet, the packet is discarded and an appropriate ICMP error SHOULD be sent (Section 3, [ICMPLIM]).

A node MAY allow configuration of different limits for processing IPv4 and IPv6 options. The default limits for IPv4 are assumed to be the same as those defined for IPv6 in [RFC8504].

### 5 Deployability

If a legacy host device receives an IPv4 packet with IPv4 Hop-by-Hop Options or Destination extension headers, the packet will be treated as having an unknown protocol and should dropped. Intermediate nodes might also observe to packets with a protocol number that is unrecognized and will forward the packet inasmuch as they would forward any packet with an unknown protocol.

In the public Internet, it is well known that there are some intermediate nodes that will drop packets with protocols that are unknown to them (firewalls would commonly to this for instance). Therefore, it is unlikely that packets with IPv4 Hop-by-Hop Options or Destination Options extension headers can be ubiquitously deployed over the Internet.

In a limited domain [LIMDOM], an operator would have control over intermediate nodes and could ensure that at a minimum they properly forward packets with IPv4 extension headers. Routers in a limited domain can be updated to process IPv4 Hop-by-Hop Options provide the functionality of features like IOAM.
6 Security Considerations

This specification enables use of Hop-by-Hop Options and Destination Options extension headers in IPv4. Related security mechanisms of IPv6 extension headers can be applied for use with IPv4 extension headers.

7 IANA Considerations

7.1 Protocol descriptions

IANA is requested to change the descriptions of the Hop-by-Hop Options and Destination Options extension headers to reflect that they are not IPv6 specific.

In the "Assigned Internet Protocol Numbers Registry", the modified protocols descriptions are:

<table>
<thead>
<tr>
<th>Decimal</th>
<th>Keyword</th>
<th>Protocol</th>
<th>IPv6 Extension header</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>HOPOPT</td>
<td>Hop-by-Hop Option</td>
<td></td>
<td>[RFC8200][RFCXXXX]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>[This document]</td>
</tr>
<tr>
<td>60</td>
<td>Opts</td>
<td>Destination Options</td>
<td></td>
<td>[RFC8200][RFCXXXX]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>[This document]</td>
</tr>
</tbody>
</table>

7.2 IPv4 Parameters registry

IANA is requested to create a parameters registry in "Internet Protocol Version 4 (IPv4) Parameters". This registry contains the assigned IPv4 Hop-by-Hop Options and Destination Options numbers.

The description of the registry shall contain:

Name: Destination Options and Hop-by-Hop Options

Registration Procedure(s): IESG Approval, IETF Review or Standards

Action Reference: This specification

Note: From (This specification) IPv4 Option Types are 8-bit values, structured as three subfields, are defined in Section 4.2 of [RFC8200].
Each distinct 8-bit Option Type identifies a different option, i.e., the high-order 3 bits are considered part of the option identification. However, it is recommended that Option Types be assigned with distinct values in the "rest" subfield, until and unless that 5-bit space becomes full.

Available Formats: <CSV link>

For each option number, the value, description, and document reference are listed. The value is provided in both hexadecimal as well as binary. The binary value is split into action, change, and rest bits which refer to the semantics of the three high-order bits of the Option Type.

The initial set of assigned IPv4 Option Types are:

<table>
<thead>
<tr>
<th>Hex value</th>
<th>Binary value</th>
<th>Description</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>act chg rest</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x00</td>
<td>00 0 00000</td>
<td>Pad1</td>
<td>[This document]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[RFC8200]</td>
</tr>
<tr>
<td>0x01</td>
<td>00 0 00001</td>
<td>PadN</td>
<td>[This document]</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>[RFC8200]</td>
</tr>
<tr>
<td>0x1e</td>
<td>00 0 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0x3e</td>
<td>00 1 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0x5e</td>
<td>01 0 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0x7e</td>
<td>01 1 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0x9e</td>
<td>10 0 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0xbe</td>
<td>10 1 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0xde</td>
<td>11 0 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
<tr>
<td>0xfe</td>
<td>11 1 11110</td>
<td>Experimental</td>
<td>[This document]</td>
</tr>
</tbody>
</table>
7.3 ICMP parameters

7.3.1 Parameter Problem codes

IANA is requested to assign the following codes in "ICMP Type Numbers" for type 12 "Parameter Problem":

3 - Erroneous header field encountered
4 - Unrecognized Next Header type encountered
5 - Unrecognized IPv4 option encountered
6 - Extension header too big
7 - Extension header chain too long
8 - Too many options in extension header
9 - Option too big

7.3.2 Destination Unreachable codes

IANA is requested to assign the following codes in "ICMP Type Numbers" for type 3 "Destination Unreachable":

16 - Headers too long
8 References

8.1 Normative References


8.2 Informative References


[IOAM] F. Brockners, S. Bhandari, V. Govindan, C. Pignataro, H. Gredler, J. Leddy, S. Youell, T. Mizrahi, D. Mozes, P. Lapukhov, R. Chang, "Encapsulations for In-situ OAM Data" draft-brockners-inband-oam-transport-05


[FAST] Herbert, T., "Firewall and Service Tickets", draft-herbert-fast-03


[IOAMGEN] Brockners, F. et al., "Geneve encapsulation for In-situ OAM Data", draft-brockners-ippm-ioam-geneve-01

[IPNOOP] Rodrigo Fonseca, George Manning Porter, Randy H. Katz, Scott Shenker and Ion Stoica, "IP Options are not an option", <https://www2.eecs.berkeley.edu/Pubs/TechRpts/2005/EECS-2005-24.html>

[ICMPLIM] Herbert, T., "ICMPv6 errors for discarding packets due to processing limits", draft-ietf-6man-icmp-limits-01


Author’s Address

Tom Herbert
Intel
Santa Clara, CA, USA
USA

Email: tom@herbertland.com