

Internet Engineering Task Force
Internet-Draft
Updates: [4271](#) (if approved)
Intended status: Standards Track
Expires: February 22, 2021

E. Chen
Cisco Systems
J. Scudder
Juniper Networks
August 21, 2020

Extended Optional Parameters Length for BGP OPEN Message
draft-ietf-idr-ext-opt-param-09

Abstract

The Optional Parameters in the BGP OPEN message as defined in the base BGP specification are limited to 255 octets due to a one-octet length field. BGP Capabilities are carried in this field and may foreseeably exceed 255 octets in the future, leading to concern about this limitation.

In this document we update [RFC 4271](#) by extending, in a backward-compatible manner, the length of the Optional Parameters in the BGP OPEN. The Parameter Length field of individual Optional Parameters is also extended.

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 <https://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 February 22, 2021.

Copyright Notice

Copyright (c) 2020 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 (<https://trustee.ietf.org/license-info>) in effect on the date of

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.

1. Introduction

The Optional Parameters Length field in the BGP OPEN message is defined in the base BGP specification [[RFC4271](#)] as one octet, thus limiting the Optional Parameters field in the OPEN message to 255 octets. Since BGP Capabilities [[RFC5492](#)] are carried in the Optional Parameters field, and new BGP capabilities continue to be introduced, the limitation is a concern for BGP development.

In this document we update [[RFC4271](#)] by extending, in a backward-compatible manner, the length of the Optional Parameters in BGP OPEN. This is done by using Optional Parameter Type 255 as a distinguished value, that indicates an extended Optional Parameters Length field follows. In this case the Parameter Length field of the individual Optional Parameters in the BGP OPEN message is also extended.

1.1. Requirements Language

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](#) [[RFC2119](#)].

2. Protocol Extensions

This document reserves Optional Parameter Type code 255 as the "Extended Length" type code.

In the event that the length of Optional Parameters in the BGP OPEN message does not exceed 255, the encodings of the base BGP specification [[RFC4271](#)] MUST be used without alteration. However, an implementation MUST be prepared to accept an OPEN message that uses the encoding of this specification for Optional Parameters of any length.

If the length of Optional Parameters is greater than 255, the extended encoding defined here MUST be used. The (non-extended) length field MUST be set to 255. The subsequent octet (which would be the first Optional Parameter Type in the non-extended format) MUST be set to 255 as well. The subsequent two octets carry the actual length. In addition, the "Parameter Length" field of each Optional Parameter is enlarged to two octets. Other than the larger sizes of

the given fields, there is no change to the BGP OPEN message defined in [\[RFC4271\]](#).

When receiving an OPEN, a BGP speaker determines the extended encoding is in use if the first Optional Parameter Type field is 255. In this case, the BGP speaker **MUST** use the Extended Optional Parameters Length field in lieu of the [\[RFC4271\]](#) encoding to determine the length of Optional Parameters contained in the message.

Accordingly, when the length of Optional Parameters in the BGP OPEN message is greater than 255, the OPEN message format is modified as follows, using the first Optional Parameter Type field to indicate the use of the extended format:

```

      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
+-+--+--+--+--+--+--+
|      Version      |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|      My Autonomous System      |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|              Hold Time              |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|                                BGP Identifier                                |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|Non-Ext OP Len.|Non-Ext OP Type|  Extended Opt. Parm. Length  |
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
|
|              Optional Parameters (variable)              |
|
+-+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

The non-extended Optional Parameters Length field **MUST** be set to 255 on transmission, and **MUST** be ignored on receipt once the use of the extended format is determined positively by inspection of the (non-extended) Optional Parameters Type field.

The subsequent one-octet field, that in the non-extended format would be the first Optional Parameter Type field, **MUST** be set to 255 on transmission. On receipt, a value of 255 for this field is the indication that the extended format is in use.

In this extended encoding, the subsequent two-octet field, termed the Extended Optional Parameters Length field, is an unsigned integer indicating the total length of the Optional Parameters field in octets. If the value of this field is zero, no Optional Parameters are present.

Likewise, in that situation the Optional Parameters encoding is modified to be the following:

```

      0                               1                               2
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
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
| Parm. Type |      Parameter Length      |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+
~      Parameter Value (variable)      ~
|                                         |
+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+

```

The rules for encoding Optional Parameters are unchanged with respect to those given in [\[RFC4271\]](#) other than the extension of the Parameter Length field to be a two-octet unsigned integer.

In parsing an OPEN message, if the one-octet "Optional Parameters Length" field is non-zero, a BGP speaker MUST use the value of the octet following the one-octet "Optional Parameters Length" field to determine both the encoding of the Optional Parameters length and the size of the "Parameter Length" field of individual Optional Parameters. If the value of this field is 255, then the encoding described above is used for the Optional Parameters length. Otherwise the encoding defined in [\[RFC4271\]](#) is used.

This encoding is chosen for backward compatibility reasons -- a BGP speaker which has not been upgraded to support this specification may legitimately send Optional Parameters whose length equals exactly 255, thus the Optional Parameters Length field alone is insufficient as an indicator. However, such a speaker would never legitimately send an Optional Parameter whose type code is 255, since that value has been reserved by this specification.

The choice to mandate that when the extended encoding is in use, the (non-extended) Optional Parameters Length field must be 255 was made for backward compatibility with implementations of earlier versions of this specification. When the extended encoding is in use, the value 0 MUST NOT be used in that field since the presence of that value could have the effect of causing a message parser to never inspect the following octet.

3. Errors

If a BGP speaker supporting this specification (a "new speaker") is peering with one which does not (an "old speaker") no interoperability issues arise unless the new speaker needs to encode Optional Parameters whose length exceeds 255. In that case, it will transmit an OPEN message which the old speaker will interpret as

containing an Optional Parameter with type code 255. Since by definition the old speaker will not recognize that type code, the old speaker may be expected to close the connection with a NOTIFICATION with an Error Code of OPEN Message Error and an Error Subcode of Unsupported Optional Parameters, according to [Section 6.2 of \[RFC4271\]](#).

Although the above is the most likely error to be sent, it is not impossible that the old speaker might detect some other error first, such as a length error, depending on the details of the implementation. In no case would the peering be expected to establish successfully; the only question is which NOTIFICATION would be generated.

We note that in any case, such a peering could not succeed, since by definition the extended length encoding would not be used by the new speaker unless the basic encoding was insufficient.

Although the Optional Parameter Type code 255 is used in this specification as the indication that the extended encoding is in use, it is not a bonafide Optional Parameter Type in the usual sense, and MUST NOT be used other than as described above. If encountered as an actual Optional Parameter Type code, it MUST be treated as an unrecognized Optional Parameter and handled according to [\[RFC4271\] Section 6.2](#).

It is not considered an error to receive an OPEN message whose Extended Optional Parameters Length value is less than or equal to 255, any value SHOULD be silently accepted. It is not considered a fatal error to receive an OPEN message whose (non-extended) Optional Parameters Length value is not 255, and whose first Optional Parameter type code is 255 -- in this case the encoding of this specification MUST be used for decoding the message. A warning MAY be logged.

[4.](#) IANA Considerations

IANA is requested to designate type code 255 in the BGP OPEN Optional Parameter Types registry as the Extended Length type code.

[5.](#) Security Considerations

This extension to BGP does not change the underlying security issues inherent in the existing BGP [\[RFC4272\]](#).

6. Acknowledgements

The authors would like to thank Yakov Rekhter and Srihari Sangli for discussing various options to enlarge the Optional Parameters field. We would also like to thank Matthew Bocci, Bruno Decraene, John Heasley, Jakob Heitz, Pradosh Mohapatra, Keyur Patel and Hannes Gredler for their valuable comments.

7. References

7.1. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC4271] Rekhter, Y., Ed., Li, T., Ed., and S. Hares, Ed., "A Border Gateway Protocol 4 (BGP-4)", [RFC 4271](#), DOI 10.17487/RFC4271, January 2006, <<https://www.rfc-editor.org/info/rfc4271>>.

7.2. Informative References

- [RFC4272] Murphy, S., "BGP Security Vulnerabilities Analysis", [RFC 4272](#), DOI 10.17487/RFC4272, January 2006, <<https://www.rfc-editor.org/info/rfc4272>>.
- [RFC5492] Scudder, J. and R. Chandra, "Capabilities Advertisement with BGP-4", [RFC 5492](#), DOI 10.17487/RFC5492, February 2009, <<https://www.rfc-editor.org/info/rfc5492>>.

Authors' Addresses

Enke Chen
Cisco Systems

Email: enkechen@cisco.com

John Scudder
Juniper Networks

Email: jgs@juniper.net

