TCPM Working Group
Internet-Draft
Intended status: Experimental
Expires: 21 January 2026

R. Bonica
T. Li
HPE

20 July 2025

TCP Extended Options
draft-bonica-tcpm-extended-options-01

Abstract

The TCP header can accommodates 40 octets of TCP options. However, modern applications may require more than 40 octets of TCP Options. Therefore, this document describes an experiment that extends the TCP Options field. If this experiment is successful, it will demonstrate that the extension procedures described herein are implementable and deployable. It will also demonstrate that they maintain backwards compatibility.

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 21 January 2026.

Copyright Notice

Table of Contents

1. Introduction

Figure 1 depicts a TCP [RFC9293] segment.

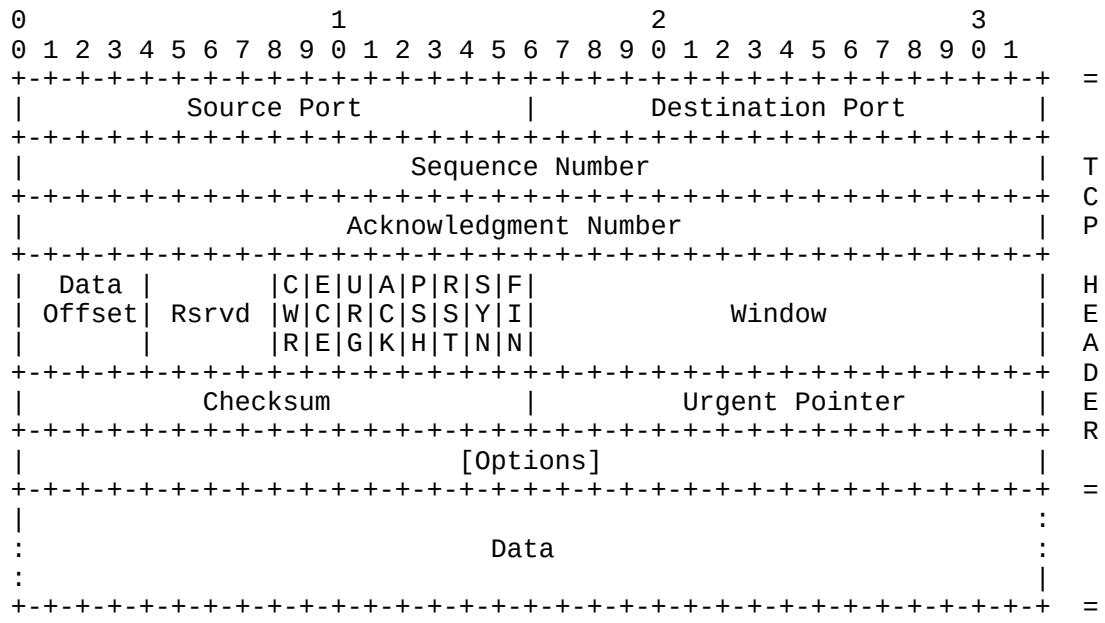

Figure 1: TCP Segment

Every TCP segment contains a header. Some TCP segments also contain data.

Each field in the TCP header, except for the last, has a fixed length. The fixed length fields in the TCP header occupy 20 octets. One of these fields is called the Data Offset field.

The last field in the TCP header is called the Options field. When present, the Options field contains a list of TCP options [TCPOPTS]. Its length varies from 0 to 40 octets.

The Data Offset field represents the offset of the Data field in the TCP segment, measured in 4-octet units. The Data Offset field also determines the length of the Options field. This is because the Options field consumes all of the space between the fixed length fields in the TCP header and the Data field.

The Data Offset field contains 4 bits. So, its value ranges nominally from 0 to 15. However, the value of the Data Offset field must be 5 or greater. This is because the Data field must follow the fixed length fields in the TCP header. As stated above, those fields occupy 20 octets.

Because the value of the Data Offset field cannot exceed 15, the offset of the Data field cannot exceed 60 and the length of the Options field cannot exceed 40 (i.e., 60 minus 20).

Modern applications may require more than 40 octets of TCP Options. Therefore, this document describes an experiment that extends the TCP Options field. If this experiment is successful, it will demonstrate that the extension procedures described herein are implementable and deployable. It will also demonstrate that they maintain backwards compatibility.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Extending The TCP Options field

According to [RFC9293]:

* The Data Offset field must have a value of 5 or greater

* TCP Options can be present only when the Data Offset field has a value greater than 5.

This document allows the Data Offset field to have a value of 0. When the Data Offset field has a value of 0, the format of the TCP Options field is as depicted in Figure 2.

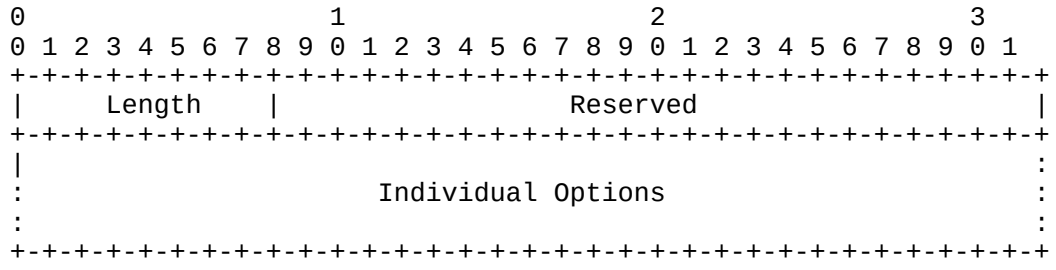

Figure 2: TCP Options

* Length: 8-bit unsigned integer. Represents the length of the TCP Options field, including the length and reserved fields. Measured in 4-octet units. Value MUST be greater than 10.

* Reserved: MUST be set to 0 by the sender and MUST be ignored by the receiver.

* Individual Options: Defined in [RFC9293].

When a TCP segment contains 40 bytes of TCP Options or fewer, it MUST be formatted exactly as described in [RFC9293]. When a TCP segment contains more than 40 bytes of TCP Options, the Data Offset field MUST be equal to 0 and the Options field MUST be formatted as described in this section.

When the Data Offset field is equal to 0, a TCP segment can include over 1,000 bytes of options. While this may be required in the distant future, it is RECOMMENDED that TCP options not exceed 256 bytes.

4. Checksum Considerations

As per [RFC9293], checksums are calculated over the entire Options field.

5. Signaling

TCP endpoints that can process segments with Data Offset equal to 0 MUST signal their ability to do so. They can use the Extended Options Permitted (EOP) Option to do so. If a TCP endpoint sends an EOP Option, it must do so during connection establishment. Figure 3 depicts the EOP Option.

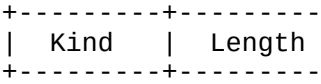

Figure 3: Extended Options Permitted Option

* Kind - Extended Options Permitted (EPO). Value TBD

* Length - MUST be equal to 2.

TCP endpoints also can signal their ability to process segments with Data Offset equal to 0 by sending such a segment during connection establishment.

TCP endpoints MUST maintain per-connection state regarding their peers' ability to process segments with Data Offset equal to 0.

6. Backwards Compatibility Considerations

6.1. Extended Options Used During Connection Establishment

When one TCP endpoint is not capable of processing segments with Data Offset equal to 0, and such a segment is sent during connection establishment, the segment will be ignored and the connection will not establish.

6.2. Extended Options Used During Connection Establishment

Once a TCP session has been established, TCP implementations MUST NOT send segments with Data Offset equal to 0 unless their peer has signaled the ability to process them.

6.3. Middleboxes and Accelerators

Legacy middleboxes and hardware accelerators discard packets with Data Offset equal to 0. Therefore, this experiment should not be performed in the presence of such devices.

7. Security Considerations

TBD

8. IANA Considerations

IANA is requested to add the following entry to the "TCP Option Kind Numbers" registry (https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml#tcp-parameters-1).

Kind	Length	Meaning	Reference
TBD	2	Extended Options Permitted	This document

Figure 4: IANA Actions

9. Experimental Results

Parties participating in this experiment should publish experimental results within one year of the publication of this document Experimental results should address the following:

* Effort required to deploy
  - Was deployment incremental or network-wide?
  - Was there a need to synchronize configurations at each node or could nodes be configured independently?

- Did the deployment require hardware upgrade?

* Scale of deployment

* Interoperability

  - Did you deploy two interoperable implementations?

  - Did you experience interoperability problems?

* Effectiveness and sufficiency of OAM mechanisms

  - Did Wireshark work?

  - Did TCPDUMP work?

10. Acknowledgements

The authors wish to acknowledge Keshawn Hamlin, Jordan Head, Rahul Khali, Prashant Kumar, Amalesh Maity, Erin MacNeil, Joe Touch and Michael Tuexen for their review and helpful comments.

11. References

11.1. 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/rfc/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

[RFC9293] Eddy, W., Ed., "Transmission Control Protocol (TCP)", STD 7, RFC 9293, DOI 10.17487/RFC9293, August 2022, <https://www.rfc-editor.org/rfc/rfc9293>.

11.2. Informative References

[TCPOPTS] Internet Assigned Numbers Authority (IANA), "Transmission Control Protocol (TCP) Parameters -", Web https://www.iana.org/assignments/tcp-parameters/tcp-parameters.xhtml#tcp-parameters-1.

Authors' Addresses

Ron Bonica
HPE
Herndon, Virginia
United States of America
Email: rbonica@juniper.net

Tony Li
HPE
Sunnyvale, California
United States of America
Email: tli@juniper.net