TCP Maintenance and Minor Extensions
Internet-Draft
Intended status: Standards Track
Expires: 9 January 2020

R.W. Grimes
P. Heist

8 July 2019

Some Congestion Experienced in TCP
draft-grimes-tcpm-tcpsce-00

Abstract

This memo classifies a TCP code point ESCE ("Echo Some Congestion Experienced") for use in feedback of IP code point SCE ("Some Congestion Experienced").

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 9 January 2020.

Table of Contents

1. Terminology

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 [RFC2119] and [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Introduction

This memo reclassifies the former TCP NS ("Nonce Sum") codepoint as ESCE.

This memo limits its scope to the redefinition of the TCP NS codepoint as ESCE, with a few brief illustrations of how it may be used.

SCE provides early and proportional feedback to the CC (congestion control) algorithms for transport protocols, including but not limited to TCP. The [sce-repo] is a Linux kernel modified to support SCE, including:

* Enhancements to Linux's Cake (Common Applications Kept Enhanced) AQM to support SCE signaling

* Modifications to the TCP receive path to reflect SCE signals back to the sender

* The addition of three new TCP CC algorithms that modify the

originals to add SCE support: Reno-SCE, DCTCP-SCE and Cubic-SCE (work in progress as of this writing)

3. Background

[I-D.morton-tsvwg-sce] defines the SCE codepoint and [RFC8311] (section 3) obsoletes the NS codepoint making it avaliable valiable for use.

4. TCP Receiver

The mechanism defined to feed back SCE signals to the sender explicitly makes use of the ESCE ("Echo Some Congestion Experienced") code point in the TCP header.

4.1. Single ACK implementation

Upon receipt of a packet an ACK is immediatly generated, the SCE codepoint is copied into the ESCE codepoint of the ACK. This keeps the count of bytes SCE marked or not marked properly reflected in the ACK packet(s). This valid implementation has the downside of increasing ACK traffic. This implementation is NOT RECOMMENDED, but useful for experimental work.

4.2. Simple Delayed ACK implementation

Upon receipt of a packet without an SCE codepoint traditional delayed ACK processing is performed. Upon receipt of a packet with an SCE codepoint immediate ACK processing SHOULD be done, this allows some delaying of ACK's, but creates earlier feedback of the congested state. This has the negative effect of over signalling ESCE.

4.3. Dithered Delayed ACK implementation

Upon receipt of a packet the SCE codepoint is stored in the TCP state. Multiple packets state may be stored. Upon generation of an ACK, normal or delayed, the stored SCE state is used to set the state of ESCE. If no SCE state is in the TCP state, then the ESCE code point MUST NOT be set. If all of the packets to be ACKed have SCE state set then the ESCE code point MUST be set in the ACK. If some of the packets to be ACKed have SCE state set then some proportional number of ACK packets SHOULD be sent with the ESCE code point set. The goal is to have the same number of bytes marked with ESCE as arrived with SCE.

5. TCP Sender

The recommended response to each single segment marked with ESCE is to reduce cwnd by an amortised 1/sqrt(cwnd) segments. Other responses, such as the 1/cwnd from DCTCP, are also acceptable but may perform less well.

This is still an area of continued investigation.

6. Related Work

TBD

7. IANA Considerations

There are no IANA considerations.

8. Security Considerations

There are no Security considerations.

9. Acknowledgements

TBD

10. Normative References

[I-D.morton-tsvwg-sce]
Morton, J. and R. Grimes, "The Some Congestion Experienced ECN Codepoint", draft-morton-tsvwg-sce-00 (work in progress), 2 July 2019, <https://www.ietf.org/archive/id/draft-morton-tsvwg-sce-00>.

[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>.

[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/info/rfc8174>.

[RFC8311] Black, D., "Relaxing Restrictions on Explicit Congestion Notification (ECN) Experimentation", RFC 8311, DOI 10.17487/RFC8311, January 2018, <https://www.rfc-editor.org/info/rfc8311>.

11. Informative References

[sce-repo] "Some Congestion Experienced Reference Implementation GitHub Repository", July 2019, <https://github.com/chromi/sce/>.

Authors' Addresses

Rodney W. Grimes
Redacted
Portland, OR 97217
United States

Email: rgrimes@freebsd.org

Peter G. Heist
Redacted
463 11 Liberec 30
Czech Republic

Email: pete@heistp.net