Workgroup: Transport Working Group
Internet-Draft: draft-morton-tsvwg-sce-01
Updates: 3168, 8311 (if approved)
Published: 4 November 2019
Intended Status: Experimental
Expires: 7 May 2020
Authors: J. Morton R.W. Grimes, Ed.

The Some Congestion Experienced ECN Codepoint

Abstract

This memo reclassifies ECT(1) to be an early notification of congestion on ECT(0) marked packets, which can be used by AQM algorithms and transports as an earlier signal of congestion than CE. It is a simple, transparent, and backward compatible upgrade to existing IETF-approved AQMs, RFC3168, and nearly all congestion control algorithms.

Status of This Memo

Copyright Notice

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

Traditional TCP congestion control exhibits a "sawtooth" pattern which, in the most favourable cases, oscillates around the optimum operating point of maximum throughput and minimum delay, which exists at the point where the congestion window equals path BDP. The term "sawtooth" brings to mind the straight-edged graphs of TCP Reno, but the equally common TCP CUBIC is essentially similar in character, as are other AIMD-derived algorithms.

A number of proposals have sought to improve this, but introduce various other tradoffs in return. TCP Vegas is consistently outcompeted by standard TCPs, DCTCP proved to be too aggressive for deployment in the public Internet, and while BBR appears to have avoided both of these problems, its complexity makes it difficult to implement correctly. Each of these proposals is characterised by primarily changing only the endpoints, not the network nodes on the path between them; though DCTCP is intended for use with a specific style of AQM, it can work with standard AQMs as long as there is no competing non-DCTCP traffic.

Some other proposals have attempted to convey information about the network path explicitly, by having network nodes inject data about link capacity and/or utilisation into passing traffic. These proposals have generally been unsuccessful due to the complex slow-path processing required in network nodes, and are not widely deployed. The only successful proposal of this type is Explicit Congestion Notification [RFC3168] which allows an AQM to signal congestion by marking packets with (essentially) a one-bit signal in preference to dropping them.

ECN defines a two-bit field supporting four codepoints, of which three are in active use and the fourth is a semantic duplicate. It was explicitly suggested during ECN's development that new meaning could be given to this spare codepoint, including as a lesser indication of congestion. With an alternative use of this codepoint having fallen out of favour, the time is right to revisit this suggestion and propose a workable method of applying it.

In so doing, care must be taken that backwards compatibility is maintained with existing traffic, endpoints and network nodes that are known or suspected to have been deployed. Keeping the changes to on-wire protocols minimal, and the complexity of implementation low, are also highly desirable.

This memo reclassifies ECT(1) to be an early notification of congestion on ECT(0) marked packets, which can be used by AQM algorithms and transports as an earlier signal of congestion than CE ("Congestion Experienced").

This memo also briefly discusses how transports should respond to ECT(1) marked packets. Detailed specifications of this behaviour are left to transport-specific memos.

3. Background

[RFC3168] defines the lower two bits of the (former) TOS byte in the IPv4/6 header as the ECN field. This may take four values: Not-ECT, ECT(0), ECT(1) or CE.

Binary	Keyword	References
00	Not-ECT (Not ECN-Capable Transport)	[RFC3168]
01	ECT(1) (ECN-Capable Transport(1))	[RFC3168]
10	ECT(0) (ECN-Capable Transport(0))	[RFC3168]
11	CE (Congestion Experienced)	[RFC3168]

Research has shown that the ECT(1) codepoint goes essentially unused, with the "Nonce Sum" extension to ECN having not been implemented in practice and thus subsequently obsoleted by [RFC8311] (section 3). Additionally, known [RFC3168] compliant senders do not emit ECT(1), and compliant middleboxes do not alter the field to ECT(1), while compliant receivers all interpret ECT(1) identically to ECT(0). These are useful properties which represent an opportunity for improvement.

Experience gained with 7 years of [RFC8290] deployment in the field suggests that it remains difficult to maintain the desired 100% link utilisation, whilst simultaneously strictly minimising induced delay due to excess queue depth - irrespective of whether ECN is in use. This leads to a reluctance amongst hardware vendors to implement the most effective AQM schemes because their headline benchmarks are throughput-based.

The underlying cause is the very sharp "multiplicative decrease" reaction required of transport protocols to congestion signalling (whether that be packet loss or CE marks), which tends to leave the congestion window significantly smaller than the ideal BDP when triggered at only slightly above the ideal value. The availability of this sharp response is required to assure network stability (AIMD principle), but there is presently no standardised and backwards-compatible means of providing a less drastic signal.

4. Some Congestion Experienced

As consensus has arisen that some form of ECN signaling should be an earlier signal than drop, this memo changes the meaning of ECT(1) to SCE, meaning "Some Congestion Experienced". Since there is no longer ambiguity between two ECT codepoints, ECT(0) is referred to as ECT. The ECN-field codepoint table then becomes:

Binary	Keyword	References
00	Not-ECT (Not ECN-Capable Transport)	[RFC3168]
01	SCE (Some Congestion Experienced)	[This Internet-draft]
10	ECT (ECN-Capable Transport)	[RFC3168]
11	CE (Congestion Experienced)	[RFC3168]

This permits middleboxes implementing AQM to signal incipient congestion, below the threshold required to justify setting CE, by converting some proportion of ECT codepoints to SCE ("SCE marking"). Existing [RFC3168] compliant receivers MUST transparently ignore this new signal with respect to congestion control, and both existing and SCE-aware middleboxes SHOULD convert SCE to CE in the same circumstances as for ECT, thus ensuring backwards compatibility with [RFC3168] ECN endpoints.

Permitted ECN codepoint packet transitions by middleboxes are:

Not-ECT	->	Not-ECT (or drop)
ECT	->	ECT or SCE or CE
SCE	->	SCE or CE
CE	->	CE

In other words, for ECN-aware flows, the ECN marking of an individual packet MAY be increased by a middlebox to signal congestion, but MUST NOT be decreased, and packets SHALL NOT be altered to appear to be ECN-aware if they were not originally, nor vice versa. Note however that SCE is numerically less than ECT, but semantically greater, and the latter definition applies for this rule.

Receivers and transport protocols conforming to this specification SHALL continue to apply the [RFC3168] interpretation of the CE codepoint, that is, to signal the sender to back off send rate to the same extent as if a packet loss were detected. This maintains compatibility with existing middleboxes, senders and receivers.

New SCE-aware receivers and transport protocols SHOULD interpret the SCE codepoint as an indication of mild congestion, and respond accordingly by applying send rates intermediate between those resulting from a continuous sequence of ECT codepoints, and those resulting from a CE codepoint. The ratio of ECT and SCE codepoints received indicates the relative severity of such congestion, with a higher proportion of SCE codepoints indicating more congestion.

The intent of SCE marking is a "cruise control" signal which permits middleboxes to request relatively small reductions in send rate, or merely a slowing of send rate growth. Accordingly, SCE marks SHOULD progressively trigger exit from exponential slow-start growth, then reduction to Reno-linear growth (for congestion control algorithms which support higher growth rates in congestion-avoidance phase), then a halt to send rate growth, then a gradual reduction of send rate. For immediate large reductions of send rate, the CE mark MUST retain its original Multiplicative Decrease power as per [RFC8511], and compliant AQMs SHOULD retain the ability to employ it where appropriate.

Details of how to implement SCE awareness at the transport layer will be left to additional Internet Drafts yet to be submitted. To ensure RTT-fair convergence with single-queue SCE AQMs, transports SHOULD stabilise at lower SCE-mark ratios for higher BDPs, and MAY reduce their response to CE marks IFF they are responding to SCE signals received at around the same time (eg. within 1-2 RTTs) in the same flow.

To maximise the benefit of SCE, middleboxes SHOULD begin to produce SCE marks at lower congestion levels than they begin to produce CE marks. This will usually ensure that SCE-aware flows avoid receiving CE marks. When a single-queue AQM is upgraded to SCE awareness, this will tend to cause SCE flows to give way to non-SCE flows; to avoid this behaviour, single-queue AQMs MAY be left as RFC-3168 compliant without SCE support.

For the avoidance of doubt, a decision to mark CE or to drop a packet always takes precedence over SCE marking.

5. Examples of use

5.1. Codel-type AQMs

A simple and natural way to implement SCE in a Codel-type AQM is to mark all ECT packets as SCE if they are over half the Codel target sojourn time, and not marked CE by Codel itself. This threshold function does not necessarily produce the best performance, but is very easy to implement and provides useful information to SCE-aware flows, often sufficient to avoid receiving CE marks whilst still efficiently using available capacity.

For a more sophisticated approach avoiding even small-scale oscillation, a stochastic ramp function may be implemented with 100% marking at the Codel target, falling to 0% marking at or above zero sojourn time. The lower point of the ramp should be chosen so that SCE is not accidentally signalled due to CPU scheduling latencies or serialisation delays of single packets. Absent rigorous analysis of these factors, setting the lower limit at half the Codel target should be safe in many cases.

The default configuration of Codel is 100ms interval, 5ms target. A typical ramp function for these parameters might cease marking below 2.5ms sojourn time, increase marking probability linearly to 100% at 5ms, and mark at 100% for sojourn times above 5ms (in which CE marking is also possible).

In single-queue AQMs, the above strategy will result in SCE flows yielding to pressure from non-SCE flows, since CE marks do not occur until SCE marking has reached 100%. A balance between smooth SCE behaviour and fairness versus non-SCE traffic can be found by having the marking ramp cross the Codel target at some lower SCE marking rate, perhaps even 0%. A two-part ramp, reaching 1/sqrt(X) at the Codel target (for some chosen X, a cwnd at which the crossover between smoothness and fairness occurs) and ramping up more steeply thereafter, has been implemented successfully for experimentation.

The CNQ algorithm [I-D.morton-tsvwg-cheap-nasty-queueing] offers a relatively simple way to limit this yielding behaviour and ensure that, even in competition with non-SCE flows, SCE flows maintain a reasonable minimum throughput capability. This may be sufficient to avoid the need for the two-part ramp described above.

Flow-isolating AQMs, including especially CNQ and DRR++ based algorithms, should avoid signalling SCE to flows classified as "sparse", in order to encourage the fastest possible convergence to the fair share.

5.2. RED-type AQMs (including PIE)

There are several reasonable methods of producing SCE signals in a RED-type AQM.

The simplest would be a threshold function, giving a hard boundary in queue depth between 0% and 100% SCE marking. This could be a sensible option for limited hardware implementations. The threshold should be set below the point at which a growing queue might trigger CE marking or packet drops.

Another option would be to implement a second marking probability function, occupying a queue-depth space just below that occupied by the main marking probability function. This should be arranged so that high marking rates (ideally 100%) are achieved at or before the point at which CE marking or packet drops begin.

For PIE specifically, a second marking probability function could be added with the same parameters as the main marking probability function, except for a lower QDELAY_REF value. This would result in the SCE marking probability remaining strictly higher than the CE marking probability for ECT flows.

5.3. TCP

Some mechanism should be defined to feed back SCE signals to the sender explicitly. Details of this are left to [I-D.grimes-tcpm-tcpsce]; use could be made of the redundant NS bit in the TCP header, which was formerly associated with ECT(1) in the Nonce Sum specification.

The recommended response to each single segment marked with SCE 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.

5.4. Other

New transports under development, such as QUIC, may implement a fine-grained signal back to the sender based on SCE. QUIC itself appears to have this sort of feedback already (counting ECT(0), ECT(1) and CE packets received), and the data should be made available for congestion control.

6. Compatibility

6.1. Existing ECN & AQM Deployments

SCE explicitly retains [RFC8511] compliant Multiplicative Decrease responses to CE marks, and conventional Multiplicative Decrease responses to packet loss. SCE senders' behaviour is thus naturally compliant with existing specifications when running over existing networks.

Existing endpoints, supporting Not-ECT or [RFC3168] compliant congestion control, are required to treat SCE marks (that is, ECT(1)) as identical to ECT(0), and will thus transparently ignore SCE marks. This is allowed for in SCE's design, and allows SCE middleboxes to be deployed into a heterogeneous network.

Hence the incremental deployability of SCE endpoints and middleboxes is good.

6.2. L4S

L4S also claims the ECT(1) codepoint, with significantly different semantic meaning than SCE. In the L4S system, ECT(1) is used to identify L4S flows, to distinguish them from [RFC3168] flows - necessary since in L4S, the semantic meaning of CE marks is also changed.

Since L4S connections are explicitly negotiated through support of AccECN, and AccECN doesn't support SCE, there is no ambiguity regarding the mode of the connection as far as endpoints are concerned.

SCE middleboxes will treat L4S flows in the same way as [RFC3168] does.

L4S middleboxes may interpret ECT packets which have received SCE markings at some other SCE-aware middlebox as though they were L4S traffic. This may result in a higher CE marking rate and/or different queuing behaviour. Though undesirable, this appears to be safe from SCE's point of view. Since the steady-state rate of SCE marking is likely to be low, the impact on L4S is also likely to be tolerable.

Accordingly, it appears as though the two experiments can coexist.

However, there is a secondary concern brought about by the L4S use of ECT(1) as a traffic identifier. If, as presently seems likely, it is found necessary to firewall L4S traffic off from the general Internet, then SCE-marked packets are also likely to be dropped at this boundary. This could have a significantly detrimental effect on ECT traffic traversing both an SCE and an L4S enabled network, even if the endpoints are not explicitly SCE aware.

7. Related Work

[RFC8087][RFC7567][RFC7928][RFC8290][RFC8289][RFC8033][RFC8034]

8. IANA Considerations

There are no IANA considerations.

9. Security Considerations

An adversary could inappropriately set SCE marks at middleboxes he controls to slow down SCE-aware flows, eventually reaching a minimum congestion window. However, the same threat already exists with respect to inappropriately setting CE marks on normal ECN flows, and this would have a greater impact per mark. Therefore no new threat is exposed by SCE in practice.

An adversary could also simply ignore SCE marks at the receiver, or ignore SCE information fed back from the receiver to the sender, in an attempt to gain some advantage in throughput. Again, the same could be said about ignoring CE marks, so no truly new threat is exposed. Additionally, correctly implemented SCE detection may actually improve long-term goodput compared to ignoring SCE.

An adversary could erase congestion information by converting SCE marks to ECT or Not-ECT codepoints, thus hiding it from the receiver. This has equivalent effects to ignoring SCE signals at the receiver. An identical threat already exists for erasing congestion information from CE marked packets, and may be mitigated by AQMs switching to dropping packets from flows observed to be non-responsive to CE.

An adversary could drop SCE-marked packets, believing them to be bogons (see also L4S Compatibility, above). Endpoints should be able to recover from this through retransmission and a reduction of cwnd. However, it is possible for this to lead to a significant denial of service. A workaround is to disable ECN for connections over the affected path.

10. Acknowledgements

Thanks to Dave Taht for his contributions to the SCE effort, and his work on writing the original draft-morton-taht-sce-00 that was submitted for IETF/104 on which this draft is based.

Many thanks to John Gilmore, the members of the ecn-sane project and the cake@lists.bufferbloat.net mailing list, and the former IETF AQM working group.

11. Normative References

[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>.

12. Informative References

[RFC8033] Pan, R., Natarajan, P., Baker, F., and G. White, "Proportional Integral Controller Enhanced (PIE): A Lightweight Control Scheme to Address the Bufferbloat Problem", RFC 8033, DOI 10.17487/RFC8033, February 2017, <https://www.rfc-editor.org/info/rfc8033>.

[I-D.grimes-tcpm-tcpsce] Grimes, R. and P. Heist, "Some Congestion Experienced in TCP", Work in Progress, Internet-Draft, draft-grimes-tcpm-tcpsce-00, 8 July 2019, <https://tools.ietf.org/html/draft-grimes-tcpm-tcpsce-00>.

[RFC7567] Baker, F., Ed. and G. Fairhurst, Ed., "IETF Recommendations Regarding Active Queue Management", BCP 197, RFC 7567, DOI 10.17487/RFC7567, July 2015, <https://www.rfc-editor.org/info/rfc7567>.

[I-D.morton-tsvwg-cheap-nasty-queueing]
Morton, J. and P. Heist, "Cheap Nasty Queueing", Work in Progress, Internet-Draft, draft-morton-tsvwg-cheap-nasty-queueing-00, 22 July 2019, <https://tools.ietf.org/html/draft-morton-tsvwg-cheap-nasty-queueing-00>.

[RFC8087] Fairhurst, G. and M. Welzl, "The Benefits of Using Explicit Congestion Notification (ECN)", RFC 8087, DOI 10.17487/RFC8087, March 2017, <https://www.rfc-editor.org/info/rfc8087>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC8290] Hoeiland-Joergensen, T., McKenney, P., Taht, D., Gettys, J., and E. Dumazet, "The Flow Queue CoDel Packet Scheduler and Active Queue Management Algorithm", RFC 8290, DOI 10.17487/RFC8290, January 2018, <https://www.rfc-editor.org/info/rfc8290>.

[RFC8511] Khademi, N., Welzl, M., Armitage, G., and G. Fairhurst, "TCP Alternative Backoff with ECN (ABE)", RFC 8511, DOI 10.17487/RFC8511, December 2018, <https://www.rfc-editor.org/info/rfc8511>.

[RFC7928] Kuhn, N., Ed., Natarajan, P., Ed., Khademi, N., Ed., and D. Ros, "Characterization Guidelines for Active Queue Management (AQM)", RFC 7928, DOI 10.17487/RFC7928, July 2016, <https://www.rfc-editor.org/info/rfc7928>.

[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>.

[RFC8289] Nichols, K., Jacobson, V., McGregor, A., Ed., and J. Iyengar, Ed., "Controlled Delay Active Queue Management", RFC 8289, DOI 10.17487/RFC8289, January 2018, <https://www.rfc-editor.org/info/rfc8289>.

[RFC8034] White, G. and R. Pan, "Active Queue Management (AQM) Based on Proportional Integral Controller Enhanced PIE) for Data-Over-Cable Service Interface Specifications

(DOCSIS) Cable Modems", RFC 8034, DOI 10.17487/RFC8034, February 2017, <https://www.rfc-editor.org/info/rfc8034>.

Authors' Addresses

Jonathan Morton
Kokkonranta 21
FI-31520 Pitkajarvi
Finland

Phone: +358 44 927 2377
Email: chromatix99@gmail.com

Rodney W. Grimes (editor)
Redacted
Portland, OR 97217
United States

Email: rgrimes@freebsd.org