Workgroup: Transport Working Group
Internet-Draft: draft-morton-tsvwg-sce-04
Updates: 3168, 8311 (if approved)
Published: 14 October 2022
Intended Status: Experimental
Expires: 17 April 2023
Authors: J. Morton P. Heist 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

This Internet-Draft will expire on 17 April 2023.

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 in [RFC3168] (section 20.2). 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]

Table 1

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]

Binary	Keyword	References
01	SCE (Some Congestion Experienced)	[This draft]
10	ECT (ECN-Capable Transport)	[RFC3168]
11	CE (Congestion Experienced)	[RFC3168]

Table 2

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.

The permitted ECN codepoint transitions by middleboxes are:

From	To
Not-ECT	Not-ECT
ECT	ECT or SCE or CE
SCE	SCE or CE
CE	CE

Table 3

Note that dropping a packet is an allowed action for any ECN codepoint. While that is the only way of indicating congestion with Not-ECT, it may also be used to both indicate and reduce congestion in any state.

To re-state the allowed transitions another way: 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 are left to additional Internet Drafts. 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 [RFC3168] 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. Design Rationale

The SCE design sees ECN as a "network feature". The risks with ECN signaling (Section 5.1), the need to handle unresponsive flows (Section 5.2), the utility of fairness (Section 5.3), and the availability of only one ECN codepoint all influenced the SCE signaling design. This section discusses these related concerns, along with what is needed from middleboxes to address them, and how that ultimately led to the selection of ECT(1) as an additional signal of lesser congestion (Section 5.4).

5.1. Risks with ECN Signaling

The safety and effectiveness of ECN signaling depends upon the unaltered transmission of the ECN bits, both for the indication of ECN support, and for ECN signaling. Unlike a drop, which is reliably and irrevocably signaled, ECN signals may be erased or manipulated.

Specifically, any of the following results in the lack of a congestion response, which is likely to lead to the near starvation of competing flows:

- if transports indicate ECT(0) but do not respond to CE
- if packets are erroneously changed from Not-ECT to ECT(0) in the network
- if CE marks are erased after a bottleneck
- if ECE marks are erased post-negotiation

Although the lack of a congestion response is similar to when transports do not respond appropriately to drop, the difference is that with ECN, the behavior can be brought about in the network, without changes to the endpoint. This may happen by accident, for example due to a broken network configuration or endpoint implementation, or on purpose, e.g. using a simple firewall rule.

Unresponsive flow mitigation, discussed in the next section, deals with flows that are not responding to congestion signals, including for the reasons listed above.

5.2. Unresponsive Flows

A single unresponsive flow has the potential to nearly starve all other competing flows in a congested bottleneck, resulting in unacceptable network delays and collapses in throughput. The need to handle unresponsive flows is corroborated in [RFC7567] (section 4), stating:

> "Research, engineering, and measurement efforts are needed regarding the design of mechanisms to deal with flows that are unresponsive to congestion notification or are responsive, but are more aggressive than present TCP."

The source language from [RFC2309] (section 5) is more direct:

> "It is urgent to begin or continue research, engineering, and measurement efforts contributing to the design of mechanisms to deal with flows that are unresponsive to congestion notification or are responsive but more aggressive than TCP."

The [COBALT] AQM algorithm is one example of how unresponsive flows can be dealt with, using the [BLUE] algorithm to detect overload and trigger drops.

Regardless of how it's done exactly, unresponsive flow mitigation is most effectively implemented with some level of flow awareness, so that drops may be directed to the offending flow/s. Once flow awareness is available, fairness steering becomes possible, discussed further in the following section.

5.3. Fairness

In order for SCE flows to compete fairly with non-SCE flows, at least one of the following is required: some form of fairness steering, or some way of separating SCE and non-SCE flows. Following is a non-exhaustive list of options:

- FQ (fair queueing), to isolate and schedule flows fairly from separate queues

- AF (approximate fairness), so that SCE and non-SCE flows can share the same queue, e.g. [AFD], [I-D.morton-tsvwg-codel-approx-fair], [I-D.morton-tsvwg-lightweight-fair-queueing]

- DSCP [RFC2474], to explicitly separate SCE and non-SCE flows (see Section 6)

When available, fairness is viewed as an advantage, in that it:

- controls aggressive flows

- prevents network bias

- promotes the fair interoperation between the ever-expanding matrix of new congestion control mechanisms

The abundance of new and proposed congestion controls is making their fair competition across bandwidths, RTTs and network conditions more difficult if not impossible to ensure in the endpoint alone [CC-REVOLUTION] [CC-COMPAT]. Congestion control implementations may dominate one another under different conditions, e.g. [BBR-CUBIC], while the widespread deployment of potentially beneficial congestion controls that seek to minimize delay is discouraged by the fact that they are often out-competed in bottlenecks by standard TCP. Fairness in the network both improves these conditions and assists transports responding to SCE.

5.4. ECT(1) as SCE

With only a single ECN codepoint remaining, options are limited for how to signal congestion with high fidelity. Meanwhile, the recent rise in ECN signaling prevalence in the Internet makes backwards compatibility with [RFC3168] a requirement. The existence of two distinct levels of ECN signalling also potentially enables new congestion control paradigms, eg. max-min-fair or power-fair instead of RTT-fair, to coexist on the Internet, even in the presence of legacy infrastructure and traffic.

Fortunately, the same network technologies that mitigate the well recognized risks listed in Section 5 above, also make the use of ECT(1) as defined by SCE possible, without a separate traffic identifier. Where those technologies cannot be deployed, Diffserv may be used to identify SCE traffic (see Section 6), a purpose for which it was expressly designed. Where that is impossible, SCE allows a graceful fallback to [RFC3168] ECN. SCE's usage of ECT(1) provides a safe and solid foundation on which future innovations in the network can improve the availability and performance of high-fidelity congestion signaling.

6. Diffserv Usage

SCE is not dependent on Diffserv [RFC2474] for its signaling, but makes use of it in the following ways:

- to mark SCE traffic for experimental or private use

- to assist middleboxes in their operation

- to request separation of traffic having different classes of SCE response

6.1. SCE Diffserv Codepoints (DSCPs)

All SCE DSCPs indicate SCE support in the originating endpoint. This MAY assist SCE marking middleboxes in their operation, but MUST NOT be depended upon for effective congestion control, because the DSCP field cannot be relied upon to survive end-to-end in the Internet. See Section 7.3 for an example of such a usage.

SCE middleboxes MUST retain any SCE DSCPs that arrive on incoming packets, and MUST NOT set them on packets that do not already have them. The DSCP field MAY be translated between Diffserv domains by the SCE middlebox, whilst retaining the sense of the SCE-related meaning thus encoded.

The SCE DSCPs MAY be set on TCP ACK and control packets which have the Not-ECT codepoint set in the ECN field, provided the TCP connection as a whole is SCE capable (or in the process of being negotiated as such). This allows all packets relating to that connection to be treated equally by middleboxes which distinguish them. Should ECN negotiation fail, the DSCP should be changed to some non-SCE value for subsequent traffic on that connection.

SCE DSCPs are not intended to imply a priority class of service. Legacy middleboxes are expected to map SCE DSCPs to a best-effort

PHB, and the DSCP numerical value should be chosen to make this mapping natural.

6.1.1. SCE-RTT-FAIR

The SCE-RTT-FAIR DSCP indicates SCE support, with standard, best-effort service implied. The response to SCE signals is in the "RTT fair" class.

6.1.2. SCE-MAX-MIN-FAIR

The SCE-MAX-MIN-FAIR DSCP indicates SCE support, with standard, best-effort service implied. The response to SCE signals is in the "max-min fair" class.

6.1.3. SCE-POWER-FAIR

The SCE-POWER-FAIR DSCP indicates SCE support, with standard, best-effort service implied. The response to SCE signals is in the "power fair" class.

6.2. Diffserv Codepoints for Experimental and Private Use

Prior to approval for public experiment, the SCE DSCPs are defined in the experimental pool xxxx11, and the following rules MUST be observed to contain SCE traffic within the experimental network:

- SCE senders SHOULD set one of the SCE DSCPs when participating in an SCE experimental network.

- SCE middleboxes MUST NOT mark SCE on packets lacking an SCE DSCP, or packets that may leave the experimental network.

- SCE receivers MUST check that one of the SCE DSCPs is present before returning SCE feedback.

- All SCE DSCPs MUST be bleached at the experimental network boundaries.

The following values are proposed for guidance only. Because they are in the experimental pool, they may be changed to suit the environment:

Name	Value (Binary)	Value (Decimal)
SCE-RTT-FAIR	000011	3
SCE-MAX-MIN-FAIR	000111	7
SCE-POWER-FAIR	001011	11

Table 4

6.3. Diffserv Codepoints for Public Use

In the event that SCE is approved for public experiment, the DSCPs will be allocated in an appropriate standards action pool, using a value that is intended to be treated as best-effort traffic by existing deployed devices.

One of the SCE DSCPs SHOULD be set by sending endpoints on all SCE capable traffic. However, they neither need to be checked by middleboxes that do not require them before marking SCE, nor by receiving endpoints before returning SCE feedback. That way, they can serve as hints for middleboxes, but the SCE signaling mechanism is not dependent on end-to-end DSCP traversal.

Unless and until a public experiment is approved, the guidance in Section 6.2 MUST be followed.

7. Examples of use

7.1. 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.

7.2. 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.

7.3. Simple Two-Queue Middleboxes

In high-capacity or resource constrained SCE marking middleboxes, DSCP may be used to select one of two queues, in lieu of implementing fairness steering. Packets marked with an SCE DSCP are placed in an SCE queue, where an AQM instance may mark congestion with either SCE or CE. Packets not marked with an SCE DSCP are placed in a second [RFC3168] queue, whose AQM instance may only mark congestion with CE. For approximate flow fairness, the queues may be scheduled in proportion to the number of flows they contain.

Note that as long as the SCE DSCP remains intact from the sending endpoint to the marking queue, the SCE queue may be used. If it has been erased or altered to a non-SCE DSCP, the packet will be placed in the [RFC3168] queue, and may still benefit from standard ECN.

If this middlebox is to be used in public environments, some form of unresponsive flow mitigation is warranted to ensure that flows haven't indicated their support for either SCE or [RFC3168] ECN incorrectly. If flows do not respond to the signals they advertise support for, they will dominate competing traffic in the same queue.

7.4. TCP

The proposed mechanism for TCP to feed back SCE signals to the sender is outlined in [I-D.grimes-tcpm-tcpsce]. Use is 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.

7.5. 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.

8. Compatibility

8.1. 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.

8.2. L4S

L4S [I-D.ietf-tsvwg-l4s-arch] also claims the ECT(1) codepoint, with significantly different semantic meaning than SCE, so a discussion around the potential for L4S and SCE compatibility is warranted. 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. However, because SCE middleboxes are likely to upgrade ECT(1) marked packets to CE at a higher threshold than L4S middleboxes would, L4S flows will outcompete non-L4S flows in a single SCE-aware queue. This is the same known safety concern with L4S deployment in regards to existing [RFC3168] queues, resulting from the redefinition of CE in L4S. Fairness steering in SCE middleboxes could mitigate this.

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. It may also result in the reordering of packets for both SCE and non-SCE aware flows through L4S middleboxes, as packets marked ECT(1) will on average traverse the bottleneck with lower delay than packets not marked ECT(1). Although this could be mitigated by [I-D.ietf-tcpm-rack], it may lead to reduced throughput and head-of-line blocking for flows that traverse both SCE and L4S bottlenecks.

There are at least two secondary concerns brought about by the L4S use of ECT(1) as a traffic identifier:

- If 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.

- If it is found necessary to bleach ECT(1) in order to disable L4S in a network, this would erase SCE signals sent to endpoints. Although not ideal, SCE transports would still safely fall back to relying on CE for congestion notification.

Lastly, an ambiguous definition of ECT(1) complicates network debugging with packet captures, since it would be unclear whether a packet was marked ECT(1) due to congestion at an SCE bottleneck, or because it is an L4S flow. Although examination of other packets in the flow could reduce this ambiguity, the necessity of observing flow state is generally discouraged for debugging purposes.

Thus far, the working group is operating under the assumption that coexistence of SCE and L4S is not an option.

9. Ongoing Research and Development

The SCE proposal is a work in progress, with ongoing or planned work in at least the following areas:

*AQM strategies for a small number of FIFO queues

*Tunnel traversal, with possible updates to [RFC3168] and [RFC6040]

*Research ways of reducing RTT dependence (Prague requirement #5)

*Performance in environments with jitter and burstiness

*New testing tools that cover many short flows, and VBR UDP flows

*Testing, with guidance from [RFC2914], [RFC7141] and [RFC5033]

10. Related Work

[RFC8087] [RFC7567] [RFC7928] [RFC8290] [RFC8289] [RFC8033] [RFC8034] [I-D.morton-tsvwg-interflow-intraflow-delays]

11. IANA Considerations

There are no IANA considerations.

12. 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.

13. 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.

14. 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>.

15. Informative References

[AFD] Pan, R., Breslau, L., Prabhakar, B., and S. Shenker, "Approximate fairness through differential dropping", in ACM SIGCOMM Computer Communication Review, April 2003, <https://dl.acm.org/doi/10.1145/956981.956985>.

[BBR-CUBIC] Borgli, R.J. and J. Misund, "Comparing BBR and CUBIC Congestion Controls", in University of Oslo, INF5072, 2018, <https://www.uio.no/studier/emner/matnat/ifi/INF5072/v18/inf5072_example1.pdf>.

[BLUE] Feng, W., Kandlur, D.D., Saha, D., and K.G. Shin, "BLUE: A New Class of Active Queue Management Algorithms", in

Computer Science Technical Report, April 1999, <http://www.eecs.umich.edu/techreports/cse/99/CSE-TR-387-99.pdf>.

[CC-COMPAT] Fejes, F., Gombos, G., Laki, S., and S. Nadas, "Compatibility of Scalable Congestion Controls", in Second Workshop on the Future of Internet Transport - FIT 2020, Paris, France (Virtual), 2020, <http://ppv.elte.hu/scalable-cc-comp>.

[CC-REVOLUTION] Fejes, F., Gombos, G., Laki, S., and S. Nadas, "Who will Save the Internet from the Congestion Control Revolution?", in Workshop on Buffer Sizing, Stanford University, 2019, <http://ppv.elte.hu/buffer-sizing>.

[COBALT] Palmei, J., Gupta, S., Imputato, P., Morton, J., Tahiliani, M.P., Avallone, S., and D. Taht, "Design and Evaluation of COBALT Queue Discipline", in 2019 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), September 2019, <https://ieeexplore.ieee.org/abstract/document/8847054>.

[I-D.grimes-tcpm-tcpsce] Grimes, R. W. and P. G. Heist, "Some Congestion Experienced in TCP", Work in Progress, Internet-Draft, draft-grimes-tcpm-tcpsce-01, 4 November 2019, <https://www.ietf.org/archive/id/draft-grimes-tcpm-tcpsce-01.txt>.

[I-D.ietf-tcpm-rack] Cheng, Y., Cardwell, N., Dukkipati, N., and P. Jha, "The RACK-TLP Loss Detection Algorithm for TCP", Work in Progress, Internet-Draft, draft-ietf-tcpm-rack-15, 22 December 2020, <https://www.ietf.org/archive/id/draft-ietf-tcpm-rack-15.txt>.

[I-D.ietf-tsvwg-l4s-arch] Briscoe, B., Schepper, K. D., Bagnulo, M., and G. White, "Low Latency, Low Loss, Scalable Throughput (L4S) Internet Service: Architecture", Work in Progress, Internet-Draft, draft-ietf-tsvwg-l4s-arch-20, 29 August 2022, <https://www.ietf.org/archive/id/draft-ietf-tsvwg-l4s-arch-20.txt>.

[I-D.morton-tsvwg-cheap-nasty-queueing] Morton, J. and P. G. Heist, "Cheap Nasty Queueing", Work in Progress, Internet-Draft, draft-morton-tsvwg-cheap-nasty-queueing-01, 4 November 2019, <https://www.ietf.org/archive/id/draft-morton-tsvwg-cheap-nasty-queueing-01.txt>.

[I-D.morton-tsvwg-codel-approx-fair] Morton, J. and P. G. Heist, "Controlled Delay Approximate Fairness AQM", Work in Progress, Internet-Draft, draft-morton-tsvwg-codel-approx-fair-01, 9 March 2020, <https://www.ietf.org/archive/id/draft-morton-tsvwg-codel-approx-fair-01.txt>.

[I-D.morton-tsvwg-interflow-intraflow-delays] Morton, J. and P. G. Heist, "Interflow vs Intraflow Delays", Work in Progress, Internet-Draft, draft-morton-tsvwg-interflow-intraflow-delays-00, 17 May 2021, <https://www.ietf.org/archive/id/draft-morton-tsvwg-interflow-intraflow-delays-00.txt>.

[I-D.morton-tsvwg-lightweight-fair-queueing] Morton, J. and P. G. Heist, "Lightweight Fair Queueing", Work in Progress, Internet-Draft, draft-morton-tsvwg-lightweight-fair-queueing-00, 2 July 2019, <https://www.ietf.org/archive/id/draft-morton-tsvwg-lightweight-fair-queueing-00.txt>.

[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>.

[RFC2309]

Braden, B., Clark, D., Crowcroft, J., Davie, B., Deering, S., Estrin, D., Floyd, S., Jacobson, V., Minshall, G., Partridge, C., Peterson, L., Ramakrishnan, K., Shenker, S., Wroclawski, J., and L. Zhang, "Recommendations on Queue Management and Congestion Avoidance in the Internet", RFC 2309, DOI 10.17487/RFC2309, April 1998, <https://www.rfc-editor.org/info/rfc2309>.

[RFC2474] Nichols, K., Blake, S., Baker, F., and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, DOI 10.17487/RFC2474, December 1998, <https://www.rfc-editor.org/info/rfc2474>.

[RFC2914] Floyd, S., "Congestion Control Principles", BCP 41, RFC 2914, DOI 10.17487/RFC2914, September 2000, <https://www.rfc-editor.org/info/rfc2914>.

[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>.

[RFC5033] Floyd, S. and M. Allman, "Specifying New Congestion Control Algorithms", BCP 133, RFC 5033, DOI 10.17487/

RFC5033, August 2007, <https://www.rfc-editor.org/info/rfc5033>.

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, DOI 10.17487/RFC6040, November 2010, <https://www.rfc-editor.org/info/rfc6040>.

[RFC7141] Briscoe, B. and J. Manner, "Byte and Packet Congestion Notification", BCP 41, RFC 7141, DOI 10.17487/RFC7141, February 2014, <https://www.rfc-editor.org/info/rfc7141>.

[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>.

[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>.

[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>.

[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>.

[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>.

[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>.

[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>.

Authors' Addresses

Jonathan Morton
Kokkonranta 21
FI-31520 Pitkajarvi
Finland

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

Peter G. Heist
Redacted
463 11 Liberec 30
Czech Republic

Email: pete@heistp.net

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

Email: rgrimes@freebsd.org