Network Working Group
Internet-Draft
Intended status: Informational
Expires: September 23, 2015

G. Fairhurst
University of Aberdeen
M. Welzl
University of Oslo

March 22, 2015

The Benefits of using Explicit Congestion Notification (ECN)
draft-ietf-aqm-ecn-benefits-02

Abstract

This document describes the potential benefits when applications enable Explicit Congestion Notification (ECN). It outlines the principal gains in terms of increased throughput, reduced delay and other benefits when ECN is used over network paths that include equipment that supports ECN-marking. It also identifies some potential problems that might occur when ECN is used. The document does not propose new algorithms that may be able to use ECN or describe the details of implementation of ECN in endpoint devices, routers and other network devices.

Status of This Memo

Copyright Notice

1. Introduction

Internet Transports (such as TCP and SCTP) have two ways to detect congestion: the loss of a packet and, if Explicit Congestion Notification (ECN) [RFC3168] is enabled, by reception of a packet with a Congestion Experienced (CE)-marking in the IP header. Both of these are treated by transports as indications of (potential) congestion. ECN may also be enabled by other transports: UDP applications may enable ECN when they are able to correctly process the ECN signals (e.g. ECN with RTP [RFC6679]).

A network device (router, middlebox, or other device that forwards packets through the network) that does not support AQM, typically uses a drop-tail policy to discard excess IP packets when its queue becomes full. The discard of packets serves as a signal to the end-to-end transport that there may be congestion on the network path being used. This triggers a congestion control reaction to reduce the maximum rate permitted by the sending endpoint.

When an application uses a transport that enables the use of ECN, the transport layer sets the ECT(0) or ECT(1) codepoint in the IP header of packets that it sends. This indicates to network devices that they may mark, rather than drop, packets in periods of congestion. This marking is generally performed by Active Queue Management (AQM) [RFC2309.bis] and may be the result of various AQM algorithms, where the exact combination of AQM/ECN algorithms does not need to be known by the transport endpoints. The focus of this document is on usage of ECN by transport and application layer flows, not its implementation in hosts, routers and other network devices.

ECN makes it possible for the network to signal the presence of congestion without incurring packet loss. This lets the network deliver some packets to an application that would otherwise have been dropped if the application or transport did not support ECN. This packet loss reduction is the most obvious benefit of ECN, but it is often relatively modest. However, enabling ECN can also result in a number of beneficial side-effects, some of which may be much more significant than the immediate packet loss reduction from ECN-marking instead of dropping packets. Several of these benefits have to do with reducing latency in some way (e.g., reduced Head-of-Line Blocking and potentially smaller queuing delay, depending on the marking rules in network devices).

The remainder of this document discusses the potential for ECN to positively benefit an application without making specific assumptions about configuration or implementation.

[RFC3168] describes a method in which a network device sets the CE codepoint of an ECN-Capable packet at the time that the router would otherwise have dropped the packet. While it has often been assumed that network devices should CE-mark packets at the same level of congestion at which they would otherwise have dropped them, separate configuration of the drop and mark thresholds is known to be supported in some network devices and this is recommended [RFC2309.bis]. Some benefits of ECN that are discussed rely upon network devices marking packets at a lower level of congestion, before they would otherwise drop packets from queue overflow [KH13].

The ability to use ECN relies upon using a transport that can support ECN. Some benefits are also only realised when the transport endpoint behaviour is also updated, this is discussed further in Section 5.

2. ECN Deployment

For an application to use ECN requires that the endpoint first enables ECN within the transport.

The ability to use ECN requires network devices along the path to at least pass IP packets that set ECN codepoints, and do not drop packets because these codepoints are used Section 2.2. This is the recommended behaviour for network devices [RFC2309.bis] [RFC3168]. Applications and transports (such as TCP or SCTP) can be designed to fall-back to not using ECN when they discover they are using a path that does not allow use of ECN (e.g., a firewall or other network device configured to drop the ECN codepoint) Section 6.1.

For an application to gain benefit from using a transport that enables ECN, network devices need to enable ECN marking. However, not all network devices along the path need to enable ECN, for the application to benefit. Any network device that does not mark an ECN-enabled packet with a CE-codepoint can be expected to drop packets under congestion. Applications that experience congestion in these network devices do not see any benefit from using ECN, but would see benefit if the congestion were to occur within a network device that did support ECN.

ECN can be deployed both in the general Internet and in controlled environments:

- ECN can be incrementally deployed in the general Internet. The IETF has provided guidance on configuration and usage in [RFC2309.bis]. A recent survey reported growing support for ECN on common network paths [TR15].

- ECN may also be deployed within a controlled environment, for example within a data centre or within a well-managed private network. In this case, the use of ECN may be tuned to the specific use-case. An example is Datacenter TCP (DCTCP) [AL10].

Some mechanisms that can assist in using ECN across paths that only partially supports ECN are noted in Section 6.

2.1. Enabling ECN in network devices

Network deployment needs also to consider the requirements for processing ECN at tunnel endpoints of network tunnels, and guidance on the treatment of ECN is provided in [RFC6040].

Further guidance on the encapsulation and use of ECN by non-IP network devices is provided in [ID.ECN-Encap].

2.2. Bleaching and middlebox requirements to deploy ECN

Cases have been noted where a sending endpoint marks a packet with a non-zero ECN mark, but the packet is received with a zero ECN value by the remote endpoint.

The current IPv4 and IPv6 specifications assign usage of 2 bits in the IP header to carry the ECN codepoint[RFC2474] [RFC3168]. A previous usage assigned these bits as a part of the now deprecated Type of Service (ToS) field [RFC1349]. Network devices that conform to this older specification may still remark or erase the ECN codepoints, and such equipment needs to be updated to the current specifications to support ECN . This remarking has also been called "ECN bleaching".

Some network devices have been observed to implement a policy that erases or "bleaches" the ECN marks at a network edge (resetting these to zero). This may be implemented for various reasons (including normalising packets to hide which equipment supports ECN). This policy prevents use of ECN by applications. A network device should therefore not remark an ECT(0) or ECT(1) mark to zero.

A network device must not change a packet with a CE mark to a zero codepoint (if the CE marking is not propagated, the packet must be discarded). Such a packet has already received ECN treatment in the

network, and remarking it would then hide the congestion signal from the endpoints.

Some networks may use ECN internally or tunnel ECN for traffic engineering or security. Guidance on the correct use of ECN in this case is provided in [RFC6040].

3. Benefit of using ECN to avoid congestion loss

When packet loss is a result of (mild) congestion, an ECN-enabled router may be expected to CE-mark, rather than drop an ECN-enabled packet [RFC2309.bis]. An application can benefit from this marking in several ways:

3.1. Improved Throughput

ECN can improve the throughput performance of applications, although this increase in throughput offered by ECN is often not the most significant gain.

When an application uses a light to moderately loaded network path, the number of packets that are dropped due to congestion is small. Using an example from Table 1 of [RFC3649], for a standard TCP sender with a Round Trip Time, RTT, of 0.1 seconds, a packet size of 1500 bytes and an average throughput of 1 Mbps, the average packet drop ratio is 0.02. This translates into an approximate 2% throughput gain if ECN is enabled. In heavy congestion, packet loss may be unavoidable with, or without, ECN.

3.2. Reduced Head-of-Line Blocking

Many transports provide in-order delivery of received data segments to the applications they support. This requires that the transport stalls (or waits) for all data that was sent ahead of a particular segment to be correctly received before it can forward any later data. This is the usual requirement for TCP and SCTP. PR-SCTP [RFC3758], UDP, and DCCP [RFC4340] provide a transport that does not have this requirement.

Delaying data to provide in-order transmission to an application results in additional latency when segments are dropped as indications of congestion. The congestive loss creates a delay of at least one RTT for a loss event before data can be delivered to an application. We call this Head-of-Line (HOL) blocking.

In contrast, using ECN can remove the resulting delay following a loss that is a result of congestion:

- First, the application receives the data normally. This also avoids the inefficiency of dropping data that has already made it across at least part of the network path. It also avoids the additional delay of waiting for recovery of the lost segment.

- Second, the transport receiver notes that it has received CE-marked packets, and then requests the sender to make an appropriate congestion-response to reduce the maximum transmission rate for future traffic.

3.3. Reduced Probability of RTO Expiry

In some situations, ECN can help reduce the chance of a retransmission timer expiring (e.g., expiry of the TCP or SCTP retransmission timeout, RTO [RFC5681]. When an application sends a burst of segments and then becomes idle (either because the application has no further data to send or the network prevents sending further data - e.g., flow or congestion control at the transport layer), the last segment of the burst may be lost. It is often not possible to recover this last segment (or last few segments) using standard methods such as Fast Recovery [RFC5681], since the receiver generates no feedback because it is unaware that the lost segments were actually sent.

In addition to avoiding HOL blocking, this allows the transport to avoid the consequent loss of state about the network path it is using, which would have arisen had there been a retransmission timeout. Typical impacts of a transport timeout are to reset path estimates such as the RTT, the congestion window, and possibly other transport state that can reduce the performance of the transport until it again adapts to the path.

Avoiding timeouts can hence improve the throughput of the application. This benefits applications that send intermittent bursts of data, and rely upon timer-based recovery of packet loss. It can be especially significant when ECN is used on TCP SYN/ACK packets [RFC5562] where the RTO interval may be large because in this case TCP cannot base the timeout period on prior RTT measurements from the same connection.

3.4. Applications that do not retransmit lost packets

Some latency-critical applications do not retransmit lost packets, yet they may be able to adjust the sending rate in the presence of congestion. Examples of such applications include UDP-based services that carry Voice over IP (VoIP), interactive video or real-time data. The performance of many such applications degrades rapidly with increasing packet loss, and many therefore employ loss-hiding

mechanisms (e.g., packet forward error correction, or data duplication) to mitigate the effect of congestion loss on the application. However, such mechanisms add complexity and can themselves consume additional network capacity reducing the capacity for application data and contributing to the path latency when congestion is experienced.

By decoupling congestion control from loss, ECN can allow the transports supporting these applications to reduce their rate before the application experiences loss from congestion, especially when the congestion is mild and the application/transport can react promptly to reception of a CE-marked packet. Because this reduces the negative impact of using loss-hiding mechanisms, ECN can have a direct positive impact on the quality experienced by the users of these applications.

4. Benefit from Early Congestion Detection

An application can further benefit from using ECN, when the network devices are configured such that they mark packets at a lower level of congestion before they would otherwise have dropped packets from queue overflow:

4.1. Avoiding Capacity Overshoot

Internet transports do not know apriori how much capacity exists along a network path. Transports therefore try to measure the capacity available to an application by probing the network path with increasing traffic to the point where they detect the onset of congestion (such as TCP or SCTP Slow Start).

ECN can help capacity probing algorithms (such as Slow Start) from significantly exceeding the bottleneck capacity of a network path. Since a transport that enables ECN can receive congestion signals before there is significant congestion, an early-marking method in network devices can help a transport respond before it induces significant congestion with resultant loss to itself or other applications sharing a common bottleneck. For example, an application/transport can avoid incurring significant congestion during Slow Start, or a bulk application that tries to increase its rate as fast as possible, may quickly detect the presence of congestion, causing it to promptly reduce its rate.

Use of ECN is more effective than schemes such as Limited Slow-Start [RFC3742] because it provides direct information about the state of the network path. An ECN-enabled application/transport that probes for capacity can reduce its rate as soon as it discovers CE-marked packets are received, and before the applications increases its rate

to the point where it builds a queue in a network device that induces congestion loss. This benefits an application seeking to increase its rate - but perhaps more significantly, it eliminates the often unwanted loss and queueing delay that otherwise may be inflicted on flows that share a common bottleneck.

4.2. Making Congestion Visible

A characteristic of using ECN is that it exposes the presence of congestion on a network path to the transport and network layers. This information can be used for monitoring performance of the path, and could be used to directly meter the amount of congestion that has been encountered upstream on a path; metering packet loss is harder. ECN measurements are used by Congestion Exposure (CoNex) [RFC6789].

A network flow that only experiences CE-marks and no loss implies that the sending endpoint is experiencing only congestion and not other sources of packet loss (e.g., link corruption or loss in middleboxes). The converse is not true - a flow may experience a mixture of ECN-marks and loss when there is only congestion or when there is a combination of packet loss and congestion [RFC2309.bis]. Recording the presence of CE-marked packets can therefore provide information about the performance of the network path.

5. Other forms of ECN-Marking/Reactions

ECN requires a definition of both how packets are CE-marked and how applications/transports need to react to reception of CE-marked packets. This section describes the benefits when updated methods are used.

ECN-capable receiving endpoints may provide more detailed feedback describing the ECN codepoints that they observe using [ID.Acc-ECN]. This can provide more information to a sending endpoint's congestion control mechanism.

Benefit has been noted when packets are CE-marked earlier than they would otherwise be dropped, using an instantaneous queue, and if the receiver provides precise feedback about the number of packet marks encountered, a better sender behavior is possible. This has been shown by Datacenter TCP (DCTCP) [AL10].

Precise feedback about the number of packet marks encountered is supported by the Real Time Protocol (RTP) when used over UDP [RFC6679] and proposed for SCTP [ST14] and TCP [ID.Acc-ECN]. An underlying assumption of DCTCP is that it is deployed in confined environments such as a datacenter. It is currently unknown whether or how such behaviour could be safely introduced into the Internet.

6. ECN transport mechanisms for paths with partial ECN support

Early deployment of ECN encountered a number of operational difficulties when the network only partially supports the use of ECN, or to respond to the challenges due to misbehaving network devices and/or endpoints. These problems have been observed to diminish with time, but may still be encountered on some Internet paths [TR15].

This section describes transport mechanisms that allow ECN-enabled endpoints to continue to work effectively over a path with partial ECN support.

6.1. Verifying whether a path really supports ECN

ECN transport and applications need to implement mechanisms to verify ECN support on the path that they use and fallback to not using ECN when it would not work. This is expected to be a normal feature of IETF-defined transports supporting ECN.

Before a transport relies on the presence or absence of CE-marked packets, it may need to verify that any ECN marks applied to packets passed by the path are indeed delivered to the remote endpoint. This may be achieved by the sender setting known ECN codepoints into specific packets in a network flow and then verifying that these reach the remote endpoint [ID.Fallback], [TR15].

Endpoints also need to be robust to path changes. A change in the set of network devices along a path may impact the ability to effectively signal or use ECN across the path, e.g., when a path changes to use a middlebox that bleaches ECN codepoints. As a necessary, but short term fix, transports could implement mechanisms that detect this and fall-back to disabling use of ECN [BA11].

6.2. Detecting ECN receiver feedback cheating

It is important that receiving endpoints accurately report the loss they experience when using a transport that uses loss-based congestion control. So also, when using ECN, a receiver must correctly report the congestion marking that it receives and then provide a mechanism to feed the congestion information back to the sending endpoint.

The transport at endpoint receivers must not try to conceal reception of CE-marked packets in the ECN feedback information that they provide to the sending endpoint [RFC2309.bis]. Transport protocols are actively encouraged to include mechanisms that can detect and appropriately respond to such misbehavior (e.g., disabling use of ECN, and relying on loss-based congestion detection [TR15]).

7. Conclusion

Network devices should enable ECN and people configuring host stacks should also enable ECN. Specifically network devices must not change a packet with a CE mark to a zero codepoint (if the CE marking is not propagated, the packet must be discarded). These are prerequisites to allow applications to gain the benefits of ECN.

Prerequisites for network devices (including IP routers) to enable use of ECN include:

- should not reset the ECN codepoint to zero by default Section 2.2.

- should correctly update the ECN codepoint in the presence of congestion.

- should correctly support alternate ECN semantics ([RFC4774]).

Prerequisites for network endpoints to enable use of ECN include:

- should use transports that can set and receive ECN marks.

- should correctly return feedback of congestion to the sending endpoint.

- must use transports that react appropriately to received ECN feedback Section 6.2.

- should use transports that can detect misuse of ECN and detect paths that do not support ECN, providing fallback to loss-based congestion detection when ECN is not supported Section 6.1.

Application developers should where possible use transports that enable the benefits of ECN. Applications that directly use UDP need to provide support to implement the functions required for ECN. Once enabled, an application that uses a transport that supports ECN will experience the benefits of ECN as network deployment starts to enable ECN. The application does not need to be rewritten to gain these benefits. Table 1 summarises some of these benefits.

Section	Benefit
3.1	Improved Throughput
3.2	Reduced Head-of-Line
3.3	Reduced Probability of RTO Expiry
3.4	Applications that do not retransmit lost packets
4.1	Avoiding Capacity Overshoot
4.2	Making Congestion Visible

Table 1: Summary of Key Benefits

8. Acknowledgements

The authors were part-funded by the European Community under its Seventh Framework Programme through the Reducing Internet Transport Latency (RITE) project (ICT-317700). The views expressed are solely those of the authors.

The authors would like to thank the following people for their comments on prior versions of this document: Bob Briscoe, David Collier-Brown, John Leslie, Colin Perkins, Richard Scheffenegger, Dave Taht, Wes Eddy.

9. IANA Considerations

XX RFC ED - PLEASE REMOVE THIS SECTION XXX

This memo includes no request to IANA.

10. Security Considerations

This document introduces no new security considerations. Each RFC listed in this document discusses the security considerations of the specification it contains.

11. Revision Information

XXX RFC-Ed please remove this section prior to publication.

Revision 00 was the first WG draft.

Revision 01 includes updates to complete all the sections and a rewrite to improve readability. Added section 2. Author list reversed, since Gorry has become the lead author. Corrections following feedback from Wes Eddy upon review of an interim version of this draft.

Note: Wes Eddy raised a question about whether discussion of the ECN Pitfalls could be improved or restrcutured - this is expected to be addressed in the next revision.

Revision 02 updates the title, and also the description of mechanisms that help with partial ECN support.

We think this draft is ready for wider review. Comments are welcome to the authors or via the IETF AQM or TSVWG mailing lists.

12. References

12.1. Normative References

[RFC2309.bis]
Baker, F. and G. Fairhurst, "IETF Recommendations Regarding Active Queue Management", Internet-draft draft-ietf-aqm-recommendation-06, October 2014.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

12.2. Informative References

[AL10] Alizadeh, M., Greenberg, A., Maltz, D., Padhye, J., Patel, P., Prabhakar, B., Sengupta, S., and M. Sridharan, "Data Center TCP (DCTCP)", SIGCOMM 2010, August 2010.

[BA11] Bauer, Steven., Beverly, Robert., and Arthur. Berger, "Measuring the State of ECN Readiness in Servers, Clients, and Routers, ACM IMC", 2011.

[ID.Acc-ECN]
Kuehlewind, Mirja., Scheffenegger, Richard., and Bob. Briscoe, "Problem Statement and Requirements for a More Accurate ECN Feedback", Internet-draft, IETF work-in-progress draft-ietf-tcpm-accecn-reqs, 2015.

[ID.ECN-Encap]
Briscoe, B., Kaippallimalil, J., and P. Thaler, "Guidelines for Adding Congestion Notification to Protocols that Encapsulate IP", Internet-draft, IETF work-in-progress draft-ietf-tsvwg-ecn-encap-guidelines, .

[ID.Fallback]
Kuehlewind, Mirja. and Brian. Trammell, "A Mechanism for ECN Path Probing and Fallback, draft-kuehlewind-tcpm-ecn-fallback, Work-in-Progress", .

[KH13] Khademi, N., Ros, D., and M. Welzl, "The New AQM Kids on the Block: Much Ado About Nothing?", University of Oslo Department of Informatics technical report 434, October 2013.

[RFC1349] "Type of Service in the Internet Protocol Suite", .

[RFC2474] "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", .

[RFC3649] Floyd, S., "HighSpeed TCP for Large Congestion Windows", RFC 3649, December 2003.

[RFC3742] Floyd, S., "Limited Slow-Start for TCP with Large Congestion Windows", RFC 3742, March 2004.

[RFC3758] Stewart, R., Ramalho, M., Xie, Q., Tuexen, M., and P. Conrad, "Stream Control Transmission Protocol (SCTP) Partial Reliability Extension", RFC 3758, May 2004.

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, March 2006.

[RFC4774] Floyd, S., "Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field", BCP 124, RFC 4774, November 2006.

[RFC5562] Kuzmanovic, A., Mondal, A., Floyd, S., and K. Ramakrishnan, "Adding Explicit Congestion Notification (ECN) Capability to TCP's SYN/ACK Packets", RFC 5562, June 2009.

[RFC5681] Allman, M., Paxson, V., and E. Blanton, "TCP Congestion Control", RFC 5681, September 2009.

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, November 2010.

[RFC6679] Westerlund, M., Johansson, I., Perkins, C., O'Hanlon, P., and K. Carlberg, "Explicit Congestion Notification (ECN) for RTP over UDP", RFC 6679, August 2012.

[RFC6789] Briscoe, B., Woundy, R., and A. Cooper, "Congestion Exposure (ConEx) Concepts and Use Cases", RFC 6789, December 2012.

[ST14] Stewart, R., Tuexen, M., and X. Dong, "ECN for Stream Control Transmission Protocol (SCTP)", Internet-draft draft-stewart-tsvwg-sctpecn-05.txt, January 2014.

[TR15] Tranmmel, Brian., Kuehlewind, Mirja., Boppart, Damiano, Learmonth, Iain., and Gorry. Fairhurst, "Enabling internet-wide deployment of Explicit Congestion Notification Tramwell, B., Kuehlewind, M., Boppart, D., Learmonth, I., Fairhurst, G. & Scheffnegger, Passive and Active Measurement Conference (PAM)", March 2015.

Authors' Addresses

Godred Fairhurst
University of Aberdeen
School of Engineering, Fraser Noble Building
Aberdeen AB24 3UE
UK

Email: gorry@erg.abdn.ac.uk

Michael Welzl
University of Oslo
PO Box 1080 Blindern
Oslo N-0316
Norway

Phone: +47 22 85 24 20
Email: michawe@ifi.uio.no