Internet Engineering Task Force (IETF)
Request for Comments: 9600
Category: Standards Track
ISSN: 2070-1721

D. Eastlake 3rd
B. Briscoe
Independent

August 2024

TRansparent Interconnection of Lots of Links (TRILL): Explicit Congestion Notification (ECN) Support

Abstract

Explicit Congestion Notification (ECN) allows a forwarding element to notify downstream devices, including the destination, of the onset of congestion without having to drop packets. This can improve network efficiency through better congestion control without packet drops. This document extends ECN to TRansparent Interconnection of Lots of Links (TRILL) switches, including integration with IP ECN, and provides for ECN marking in the TRILL header extension flags word (RFC 7179).

Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at https://www.rfc-editor.org/info/rfc9600.

Table of Contents

1. Introduction

Explicit Congestion Notification (ECN) [RFC3168] [RFC8311] allows a forwarding element (such as a router) to notify downstream devices, including the destination, of the onset of congestion without having to drop packets. This can improve network efficiency through better congestion control without packet drops. The forwarding element can explicitly mark a proportion of packets in an ECN field instead of dropping packets. For example, a 2-bit field is available for ECN marking in IP headers.

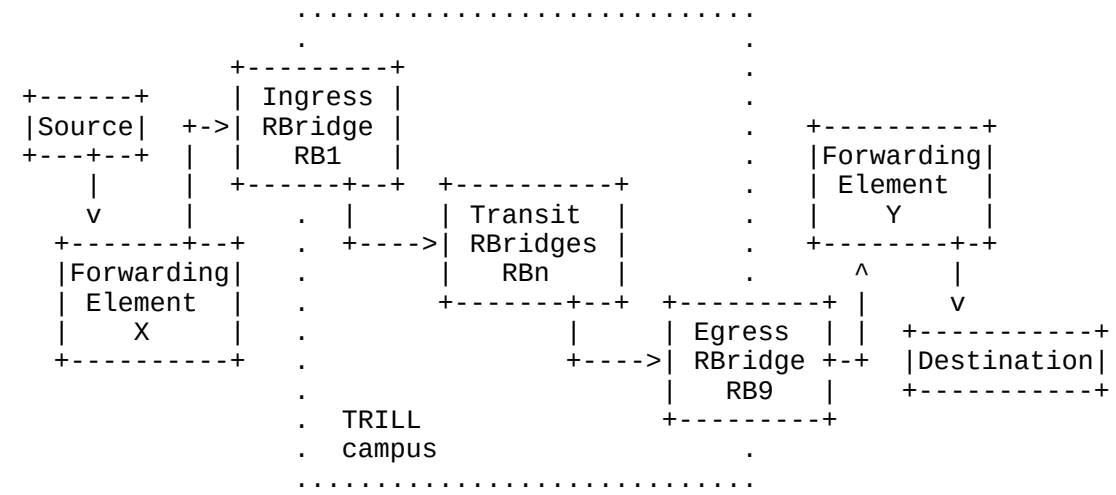


Figure 1: Example Path-Forwarding Nodes

In [RFC3168], it was recognized that tunnels and lower-layer protocols would need to support ECN, and ECN markings would need to be propagated, as headers were encapsulated and decapsulated. [RFC9599] gives guidelines on the addition of ECN to protocols like TRILL that often encapsulate IP packets, including propagation of ECN from and to IP.

In Figure 1, assuming IP traffic, RB1 is an encapsulator and RB9 is a decapsulator. Traffic from Source to RB1 might or might not get marked as having experienced congestion in forwarding elements, such as X, before being encapsulated at ingress RB1. Any such ECN marking is encapsulated with a TRILL header [RFC6325].

This document specifies how ECN marking in traffic at the ingress is copied into the TRILL extension header flags word and requires such copying for IP traffic. It also enables congestion marking by a congested RBridge (such as RBn or RB1 above) in the TRILL header extension flags word [RFC7179].

At RB9, the TRILL egress, it specifies how any ECN markings in the TRILL header flags word and in the encapsulated traffic are combined so that subsequent forwarding elements, such as Y and the Destination, can see if congestion was experienced at any previous point in the path from the Source.

A large part of the guidelines for adding ECN to lower-layer protocols [RFC9599] concerns safe propagation of congestion notifications in scenarios where some of the nodes do not support or understand ECN. Such ECN ignorance is not a major problem with RBridges using this specification, because the method specified assures that, if an egress RBridge is ECN ignorant (so it cannot further propagate ECN) and congestion has been encountered, the egress RBridge will at least drop the packet, and this drop will itself indicate congestion to end stations.

1.1. Conventions Used in This Document

The terminology and acronyms defined in [RFC6325] are used herein with the same meaning.

In this documents, "IP" refers to both IPv4 and IPv6.

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 BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Abbreviations:

AQM: Active Queue Management

CCE: Critical Congestion Experienced

CE: Congestion Experienced

CItE: Critical Ingress-to-Egress

ECN: Explicit Congestion Notification

ECT: ECN-Capable Transport

L4S: Low Latency, Low Loss, and Scalable throughput

NCHbH: Non-Critical Hop-by-Hop

NCCE: Non-Critical Congestion Experienced

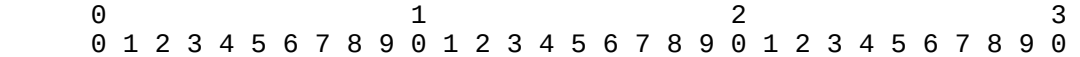
Not-ECT: Not ECN-Capable Transport

PCN: Pre-Congestion Notification

2. The ECN-Specific Extended Header Flags

The extension header fields for ECN in TRILL are defined as a 2-bit TRILL-ECN field and a one-bit CCE field in the 32-bit TRILL header extension flags word [RFC7780].

These fields are shown in Figure 2 as "ECN" and "CCE". The TRILL-ECN field consists of bits 12 and 13, which are in the range reserved for NCHbH bits. The CCE field consists of bit 26, which is in the range reserved for CItE bits. The CRItE bit is the critical Ingress-to-Egress summary bit and will be one if, and only if, any of the bits in the CItE range (21-26) are one or there is a critical feature invoked in some further extension of the TRILL header after the extension flags word. The other bits and fields shown in Figure 2 are not relevant to ECN. See [RFC7780], [RFC7179], and [IANAthFlags] for the meaning of these other bits and fields.



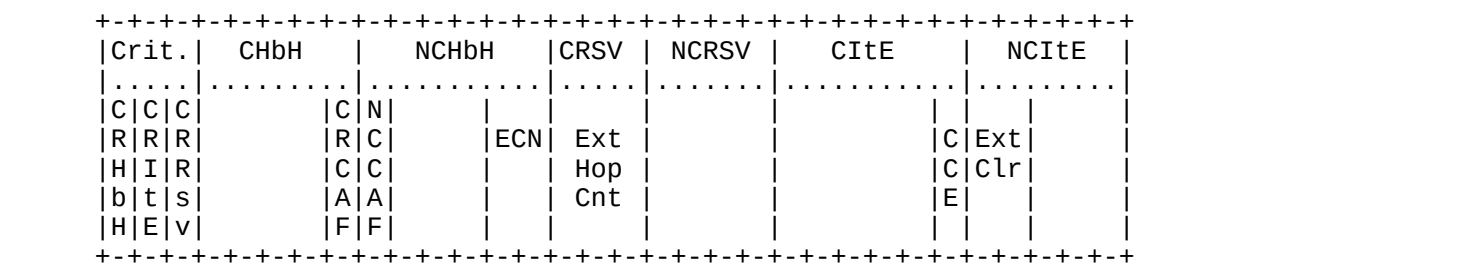


Figure 2: The TRILL-ECN and CCE TRILL Header Extension Flags Word Fields

Table 1 shows the meaning of the codepoints in the TRILL-ECN field. The first three have the same meaning as the corresponding ECN field codepoints in the IP header, as defined in [RFC3168]. However, codepoint 0b11 is called NCEE to distinguish it from CE in IP.

Binary	Name	Meaning
00	Not-ECT	Not ECN-Capable Transport
01	ECT(1)	ECN-Capable Transport (1)
10	ECT(0)	ECN-Capable Transport (0)
11	NCCE	Non-Critical Congestion Experienced

Table 1: TRILL-ECN Field Codepoints

3. ECN Support

This section specifies interworking between TRILL and the original standardized form of ECN in IP [RFC3168].

The subsections below describe the required behavior to support ECN at TRILL ingress, transit, and egress. The ingress behavior occurs as a native frame is encapsulated with a TRILL header to produce a TRILL Data packet. The transit behavior occurs in all RBridges where TRILL Data packets are queued, usually at the output port (including the output port of the TRILL ingress). The egress behavior occurs where a TRILL Data packet is decapsulated and output as a native frame through an RBridge port.

An RBridge that supports ECN MUST behave as described in the relevant subsections below, which correspond to the recommended provisions in Section 3 of this document and Sections 4.2 through 4.4 of [RFC9599]. Nonetheless, the scheme is designed to safely propagate some form of congestion notification even if some RBridges in the path followed by a TRILL Data packet support ECN and others do not.

3.1. Ingress ECN Support

The behavior at an ingress RBridge is as follows:

* When encapsulating an IP frame, the ingress RBridge MUST:
  - set the F flag in the main TRILL header [RFC7780];
  - create a flags word as part of the TRILL header;
  - copy the two ECN bits from the IP header into the TRILL-ECN field (flags word bits 12 and 13); and

- ensure the CCE flag is set to zero (flags word bit 26).

* When encapsulating a frame for a non-IP protocol (where that protocol has a means of indicating that ECN is understood by the ingress RBridge), the ingress RBridge MUST follow the guidelines in Section 4.3 of [RFC9599] to add a flags word to the TRILL header. For a non-IP protocol with an ECN field similar to IP, this would be achieved by copying into the TRILL-ECN field from the encapsulated native frame.

3.2. Transit ECN Support

The transit behavior, shown below, is required at all RBridges where TRILL Data packets are queued, usually at the output port.

* An RBridge that supports ECN MUST implement some form of AQM according to the guidelines of [RFC7567]. The RBridge detects congestion either by monitoring its own queue depth or by participating in a link-specific protocol.

* If the TRILL header flags word is present, whenever the AQM algorithm decides to indicate critical congestion on a TRILL Data packet, it MUST set the CCE flag (flags word bit 26). Note that Classic ECN marking [RFC3168] only uses critical congestion indications, but the two variants in Section 4.1 use a combination of critical and non-critical congestion indications.

* If the TRILL header flags word is not present, the RBridge will either drop the packet or it MAY do all of the following instead to indicate congestion:

  - set the F flag in the main TRILL header;

  - add a flags word to the TRILL header;

  - set the TRILL-ECN field to Not-ECT (00); and

  - set the CCE flag and the critical Ingress-to-Egress summary bit (CRItE).

Note that a transit RBridge that supports ECN does not refer to the TRILL-ECN field before signaling CCE in a packet. It signals CCE irrespective of whether the packet indicates that the transport is ECN capable. The egress/decapsulation behavior ensures that a CCE indication is converted to a drop if the transport is not ECN capable.

3.3. Egress ECN Support

3.3.1. Non-ECN Egress RBridges

If the egress RBridge does not support ECN, that RBridge will ignore bits 12 and 13 of any flags word that is present because it does not contain any special ECN logic. Nonetheless, if a transit RBridge has set the CCE flag, the egress will drop the packet. This is because drop is the default behavior for an RBridge decapsulating a CItE flag when it has no specific logic to understand it. Drop is the intended behavior for such a packet, as required by Section 4.4 of [RFC9599].

3.3.2. ECN Egress RBridges

If an RBridge supports ECN, for the two cases of an IP and a non-IP inner packet, the egress behavior is as follows:

Decapsulating an inner IP packet: The RBridge sets the ECN field of the outgoing native IP packet using Table 3. It MUST set the ECN

field of the outgoing IP packet to the codepoint at the intersection of the row for the arriving encapsulated IP packet and the column for 3-bit ECN codepoint in the arriving outer TRILL Data packet TRILL header. If no TRILL header extension flags word is present, the 3-bit ECN codepoint is assumed to be all zero bits.

The name of the TRILL 3-bit ECN codepoint used in Table 3 is defined using the combination of the TRILL-ECN and CCE fields in Table 2. Specifically, the TRILL 3-bit ECN codepoint is called CE if either NCCE or CCE is set in the TRILL header extension flags word. Otherwise, it has the same name as the 2-bit TRILL-ECN codepoint.

In the case where the TRILL 3-bit ECN codepoint indicates CE but the encapsulated native IP frame indicates a Not-ECT, it can be seen that the RBridge MUST drop the packet. Such packet dropping is necessary because a transport above the IP layer that is not ECN capable will have no ECN logic, so it will only understand dropped packets as an indication of congestion.

Decapsulating a non-IP protocol frame: If the frame has a means of indicating ECN that is understood by the RBridge, it MUST follow the guidelines in Section 4.4 of [RFC9599] when setting the ECN information in the decapsulated native frame. For a non-IP protocol with an ECN field similar to IP, this would be achieved by combining the information in the TRILL header flags word with the encapsulated non-IP native frame, as specified in Table 3.

TRILL-ECN Name	TRILL-ECN Bits	CCE	Arriving TRILL 3-Bit ECN Codepoint Name
Not-ECT	00	0	Not-ECT
ECT(1)	01	0	ECT(1)
ECT(0)	10	0	ECT(0)
NCCE	11	0	CE
Not-ECT	00	1	CE
ECT(1)	01	1	CE
ECT(0)	10	1	CE
NCCE	11	1	CE

Table 2: Mapping of TRILL-ECN and CCE Fields to the TRILL 3-Bit ECN Codepoint Name

Inner Native Header	Arriving TRILL 3-Bit ECN Codepoint Name			
	Not-ECT	ECT(0)	ECT(1)	CE
Not-ECT	Not-ECT	Not-ECT(*)	Not-ECT(*)	<drop>
ECT(0)	ECT(0)	ECT(0)	ECT(1)	CE
ECT(1)	ECT(1)	ECT(1)(*)	ECT(1)	CE
CE	CE	CE	CE(*)	CE

Table 3: Egress ECN Behavior

An asterisk in Table 3 indicates a combination that is currently unused in all variants of ECN marking (see Section 4) and therefore SHOULD be logged.

With one exception, the mappings in Table 3 are consistent with those for IP-in-IP tunnels [RFC6040], which ensures backward compatibility with all current and past variants of ECN marking (see Section 4). It also ensures forward compatibility with any future form of ECN marking that complies with the guidelines in [RFC9599], including cases where ECT(1) represents a second level of marking severity below CE.

The one exception is that the drop condition in Table 3 need not be logged because, with TRILL, it is the result of a valid combination of events.

4. TRILL Support for ECN Variants

This section is informative, not normative; it discusses interworking between TRILL and variants of the standardized form of ECN in IP [RFC3168]. See also [RFC8311].

The ECN wire protocol for TRILL (Section 2) and the ingress (Section 3.1) and egress (Section 3.3) ECN behaviors have been designed to support the other known variants of ECN as detailed below. New variants of ECN will have to comply with the guidelines for defining alternative ECN semantics [RFC4774]. It is expected that the TRILL ECN wire protocol is generic enough to support such potential future variants.

4.1. Pre-Congestion Notification (PCN)

The PCN wire protocol [RFC6660] is recognized by the use of a PCN-compatible Diffserv codepoint in the IP header and a nonzero IP-ECN field. For TRILL or any lower-layer protocol, equivalent traffic-classification codepoints would have to be defined, but that is outside the scope of this document.

The PCN wire protocol is similar to ECN, except it indicates congestion with two levels of severity. It uses:

* 11 (CE) as the most severe, termed the Excess-Traffic-Marked (ETM) codepoint

* 01 ECT(1) as a lesser severity level, termed the Threshold-Marked (ThM) codepoint. This difference between ECT(1) and ECT(0) only applies to PCN, not to the classic ECN support specified for TRILL in this document before Section 4.

To implement PCN on a transit RBridge would require a detailed specification. In brief:

* the TRILL CCE flag would be used for the Excess-Traffic-Marked (ETM) codepoint;

* ECT(1) in the TRILL-ECN field would be used for the Threshold-Marked codepoint.

Then, the ingress and egress behaviors defined in Section 3 would not need to be altered to ensure support for PCN as well as ECN.

4.2. Low Latency, Low Loss, and Scalable Throughput (L4S)

L4S is currently on the IETF's experimental track. An outline of how a transit TRILL RBridge would support L4S [RFC9331] is given in Appendix A.

5. IANA Considerations

IANA has updated the "TRILL Extended Header Flags" registry by replacing the lines for bits 9-13 and 21-26 with the following:

Bits	Purpose	Reference
9-11	available non-critical hop-by-hop flags	[RFC7179]
12-13	TRILL-ECN (Explicit Congestion Notification)	RFC 9600
21-25	available critical ingress-to-egress flags	[RFC7179]
26	Critical Congestion Experienced (CCE)	RFC 9600

Table 4: Updated "TRILL Extended Header Flags" Registry

6. Security Considerations

TRILL support of ECN is a straightforward combination of previously specified ECN and TRILL with no significant new security considerations.

For general security considerations regarding adding ECN to lower layer protocols, see [RFC9599] and [RFC6040].

For general TRILL protocol security considerations, see [RFC6325].

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[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>.

[RFC4774] Floyd, S., "Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field", BCP 124, RFC 4774, DOI 10.17487/RFC4774, November 2006, <https://www.rfc-editor.org/info/rfc4774>.

[RFC6325] Perlman, R., Eastlake 3rd, D., Dutt, D., Gai, S., and A. Ghanwani, "Routing Bridges (RBridges): Base Protocol Specification", RFC 6325, DOI 10.17487/RFC6325, July 2011, <https://www.rfc-editor.org/info/rfc6325>.

[RFC7179] Eastlake 3rd, D., Ghanwani, A., Manral, V., Li, Y., and C. Bestler, "Transparent Interconnection of Lots of Links (TRILL): Header Extension", RFC 7179, DOI 10.17487/RFC7179, May 2014, <https://www.rfc-editor.org/info/rfc7179>.

[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>.

[RFC7780] Eastlake 3rd, D., Zhang, M., Perlman, R., Banerjee, A., Ghanwani, A., and S. Gupta, "Transparent Interconnection of Lots of Links (TRILL): Clarifications, Corrections, and Updates", RFC 7780, DOI 10.17487/RFC7780, February 2016, <https://www.rfc-editor.org/info/rfc7780>.

[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>.

[RFC9599] Briscoe, B. and J. Kaippallimalil, "Guidelines for Adding Congestion Notification to Protocols that Encapsulate IP", RFC 9599, DOI 10.17487/RFC9599, August 2024, <https://www.rfc-editor.org/info/rfc9599>.

7.2. Informative References

[IANAthFlags] IANA, "TRILL Extended Header Flags", <http://www.iana.org/assignments/trill-parameters/>.

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

[RFC6660] Briscoe, B., Moncaster, T., and M. Menth, "Encoding Three Pre-Congestion Notification (PCN) States in the IP Header Using a Single Diffserv Codepoint (DSCP)", RFC 6660, DOI 10.17487/RFC6660, July 2012, <https://www.rfc-editor.org/info/rfc6660>.

[RFC9331] De Schepper, K. and B. Briscoe, Ed., "The Explicit Congestion Notification (ECN) Protocol for Low Latency, Low Loss, and Scalable Throughput (L4S)", RFC 9331, DOI 10.17487/RFC9331, January 2023, <https://www.rfc-editor.org/info/rfc9331>.

Appendix A. TRILL Transit RBridge Behavior to Support L4S

The specification of the Low Latency, Low Loss, and Scalable throughput (L4S) wire protocol for IP is given in [RFC9331]. L4S is one example of the ways TRILL ECN handling may evolve [RFC8311]. It is similar to the original ECN wire protocol for IP [RFC3168], except:

* An AQM that supports L4S classifies packets with ECT(1) or CE in the IP header into an L4S queue and a "Classic" queue otherwise.

* The meaning of CE markings applied by an L4S queue is not the same as the meaning of a drop by a "Classic" queue (contrary to the original requirement for ECN [RFC3168]). Instead, the likelihood that the Classic queue drops packets is defined as the square of the likelihood that the L4S queue marks packets -- e.g., when there is a drop probability of 0.0009 (0.09%), the L4S marking probability will be 0.03 (3%).

This seems to present a problem for the way that a transit TRILL

RBridge defers the choice between marking and dropping to the egress. Nonetheless, the following pseudocode outlines how a transit TRILL RBridge can implement L4S marking in such a way that the egress behavior already described in Section 3.3 for Classic ECN [RFC3168] will produce the desired outcome.

```
/* p is an internal variable calculated by any L4S AQM
 *  dependent on the delay being experienced in the Classic queue.
 * bit13 is the least significant bit of the TRILL-ECN field
 */

% On TRILL transit
if (bit13 == 0 ) {
      % Classic Queue
      if (p > max(random(), random()) )
         mark(CCE)                          % likelihood: p^2

} else {
      % L4S Queue
      if (p > random() ) {
         if (p > random() )
            mark(CCE)                       % likelihood: p^2
         else
            mark(NCCE)                      % likelihood: p - p^2
      }
}
```

With the above transit behavior, an egress that supports ECN (Section 3.3) will drop packets or propagate their ECN markings depending on whether the arriving inner header is from an ECN-capable or not ECN-capable transport.

Even if an egress has no L4S-specific logic of its own, it will drop packets with the square of the probability that an egress would if it did support ECN, for the following reasons:

* Egress with ECN support:

  - L4S: Propagates both the Critical and Non-Critical CE marks (CCE and NCCE) as a CE mark.

    Likelihood: p^2 + p - p^2 = p

  - Classic: Propagates CCE marks as CE or drop, depending on the inner header.

    Likelihood: p^2

* Egress without ECN support:

  - L4S: Does not propagate NCCE as a CE mark, but drops CCE marks.

    Likelihood: p^2

  - Classic: Drops CCE marks.

    Likelihood: p^2

Acknowledgements

The helpful comments of Loa Andersson and Adam Roach are hereby acknowledged.

Authors' Addresses

Donald E. Eastlake, 3rd

Independent
2386 Panoramic Circle
Apopka, FL 32703
United States of America
Phone: +1-508-333-2270
Email: d3e3e3@gmail.com

Bob Briscoe
Independent
United Kingdom
Email: ietf@bobbriscoe.net
URI: http://bobbriscoe.net/