INTERNET-DRAFT
Intended status: Proposed Standard

D. Eastlake
Futurewei Technologies
B. Briscoe
Independent
Y. Li
Huawei Technologies
A Malis
Malis Consulting
X. Wei
Huawei Technologies

Expires: October 1, 2021

April 2, 2021

Explicit Congestion Notification (ECN) and Congestion Feedback Using the Network Service Header (NSH) and IPFIX

<draft-ietf-sfc-nsh-ecn-support-05.txt>

Abstract

Explicit congestion notification (ECN) allows a forwarding element to notify downstream devices of the onset of congestion without having to drop packets. Coupled with a means to feed information about congestion back to upstream nodes, this can improve network efficiency through better congestion control, frequently without packet drops. This document specifies ECN and congestion feedback support within a Service Function Chaining (SFC) architecture domain through use of the Network Service Header (NSH, RFC 8300) and IP Flow Information Export (IPFIX, RFC 7011).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Distribution of this document is unlimited. Comments should be sent to the SFC Working Group mailing list <sfc@ietf.org> or to the authors.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/1id-abstracts.html. The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

Table of Contents

1. Introduction....4
1.1 NSH Background....4
1.2 ECN Background....6
1.3 Tunnel Congestion Feedback Background....6
1.4 Conventions Used in This Document....8

2. The NSH ECN Field....10

3. ECN Support in the NSH....12
3.1 At The Ingress....13
3.2 At Transit Nodes....14
3.2.1 At NSH Transit Nodes....14
3.2.2 At an SF/Proxy....15
3.2.3 At Other Forwarding Nodes....15
3.3 At Exit/Egress....16
3.4 Conservation of Packets....16

4. Tunnel Congestion Feedback Support....18
4.1 Congestion Level Measurement....18
4.3 Congestion Information Delivery....19
4.3 IPFIX Extensions....21
4.3.1 nshServicePathID....21
4.3.2 tunnelEcnCeCeByteTotalCount....21
4.3.3 tunnelEcnEctNectBytetTotalCount....22
4.3.4 tunnelEcnCeNectByteTotalCount....22
4.3.5 tunnelEcnCeEctByteTotalCount....22
4.3.6 tunnelEcnEctEctByteTotalCount....23
4.3.7 tunnelEcnCEMarkedRatio....23

5. Example of Use....24

6. IANA Considerations....27
6.1 SFC NSH Header ECN Bits....27
6.2 IPFIX Information Element IDs....27

7. Security Considerations....29
8. Acknowledgements....29

Normative References....30
Informative References....31

Authors' Addresses....32

1. Introduction

Explicit Congestion Notification (ECN [RFC3168]) allows a forwarding element to notify downstream devices of the onset of congestion without having to drop packets. Coupled with a means to feed information about congestion back to upstream nodes, this can improve network efficiency through better congestion control, frequently without packet drops. This document specifies ECN and congestion feedback support within a Service Function Chaining (SFC [RFC7665]) architecture domain through use of the Network Service Header (NSH [RFC8300]) and IP Flow Information Export (IPFIX [RFC7011]).

It requires that all ingress and egress nodes of the SFC domain implement ECN. While congestion management will be the most effective if all interior nodes of the SFC domain implement ECN, some benefit is obtained even if some interior nodes do not implement ECN. Congestion at any interior bottleneck where ECN marking is not implemented will be unmanaged.

The subsections below in this section provide background information on NSH, ECN, congestion feedback, and terminology used in this document.

1.1 NSH Background

The Service Function Chaining (SFC [RFC7665]) architecture calls for the encapsulation of traffic within a service function chaining domain with a Network Service Header (NSH [RFC8300]) added by the "Classifier" (ingress node) on entry to the domain and the NSH being removed on exit from the domain at the egress node. The NSH is used to control the path of a packet in an SFC domain. The NSH is a natural place, in a domain where traffic is NSH encapsulated, to note congestion, avoiding possible confusion due, for example, to changes in the outer transport header in different parts of the domain.

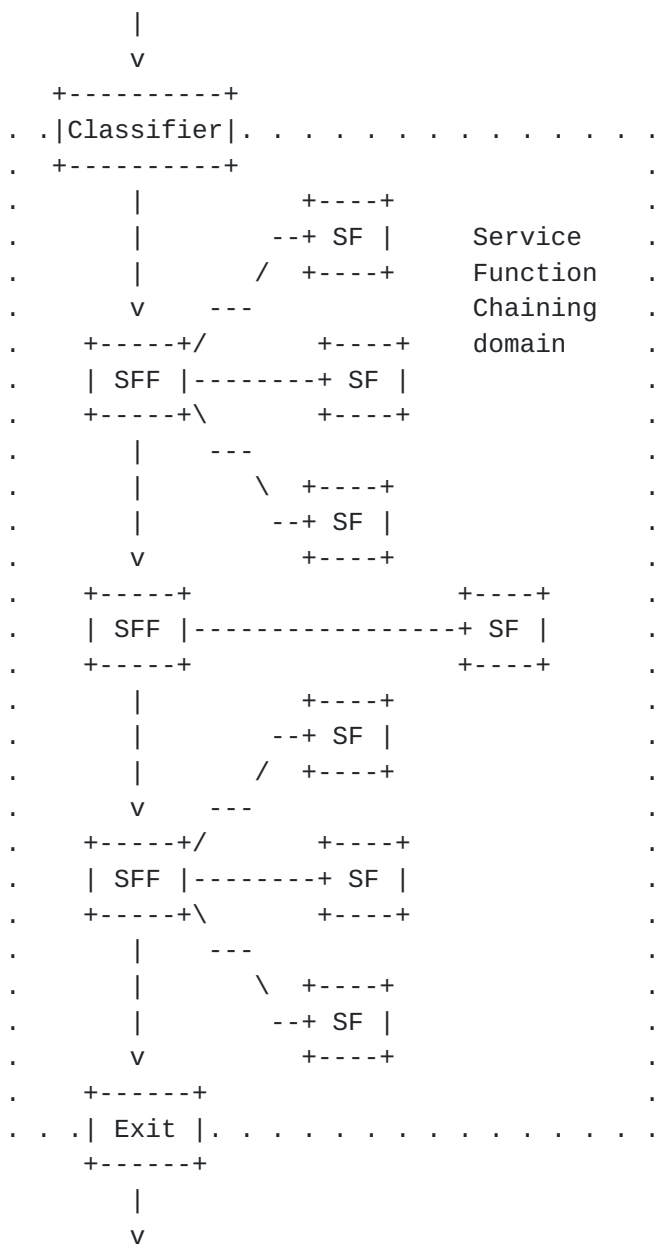


Figure 1. Example SFC Path Forwarding Nodes

Figure 1 shows an SFC domain for the purpose of illustrating the use of the NSH. Traffic passes through a sequence of Service Function Forwarders (SFFs) each of which sends the traffic to one or more Service Functions (SFs). Each SF performs some operation on the traffic, for example firewall or Network Address Translation (NAT) or load balancer, and then returns it to the SFF from which it was received.

Logically, during the transit of each SFF, the outer transport header

that got the packet to the SFF is stripped (see Figure 3), the SFF decides on the next forwarding step, either adding a new transport

header or, if the SFF is the exit/egress, removing the NSH header. The transport headers added may be different in different regions of the SFC domain. For example, IP could be used for some SFF-to-SFF communication and MPLS used for other such communication.

1.2 ECN Background

Explicit congestion notification (ECN [RFC3168]) allows a forwarding element (such as a router or a Service Function Forwarder (SFF) or Service Function (SF)) to notify downstream devices of the onset of congestion without having to drop packets. This can be used as an element in active queue management (AQM) [RFC7567] to improve network efficiency through better traffic control without packet drops. The forwarding element can explicitly mark some packets in an ECN field instead of dropping the packet. For example, a two-bit field is available for ECN marking in IP headers [RFC3168].

1.3 Tunnel Congestion Feedback Background

Tunnels are widely deployed in various networks including data center networks, enterprise network, and the public Internet. A tunnel consists of ingress, egress, and a set of intermediate nodes including routers. Tunnel Congestion Feedback (Section 4) is a building block for congestion mitigation methods. It supports feedback of congestion information from an egress node to an ingress node. This document treats the SFC domain as a tunnel with the initial Classifier node being the ingress; however, the Tunnel Congestion Feedback facilities specified in this document MAY be used in other contexts besides SFC domains.

Examples of actions that can be taken by an ingress node when it has knowledge of downstream congestion include those listed below. Details of implementing these traffic control methods, beyond those given here, are outside the scope of this document.

Any action by a tunnel ingress to reduce congestion needs to allow sufficient time for the end-to-end congestion control loop to respond first, otherwise the system could go unstable. For instance by the ingress taking a smoothed average of the level of congestion signaled by feedback from the tunnel egress or delaying any action for at least the worst case global round trip time (for example 100 milliseconds).

(1) Traffic throttling (policing), where the downstream traffic flowing out of the ingress node is limited to reduce or eliminate

congestion.

(2) Upstream congestion feedback, where the ingress node sends messages upstream to or towards the ultimate traffic source, a function that can throttle traffic generation/transmission.

(3) Traffic re-direction, where the ingress node configures the NSH of some future traffic so that it avoids congested paths. Great care must be taken with this option to avoid (a) significant re-ordering of traffic in flows that it is desirable to keep in order and (b) oscillation/instability in traffic paths due to alternate congestion of previously idle paths and the idling of previously congested paths. For example, it is preferable to classify traffic into flows of a sufficiently coarse granularity that the flows are long lived and then use a stable path per flow, sending only newly appearing flows on apparently uncongested paths.

Figure 2 shows an example path from an original sender to a final receiver passing through an example chain of service functions between the ingress and egress of an SFC domain. The path is also likely to pass through other network nodes outside the SFC domain (not shown) before entering the SFC domain and after leaving the SFC domain.

The figure shows typical congestion feedback that would be expected from the final receiver to the origin sender, which controls the load the origin sender applies to all elements on the path. The figure also shows the congestion feedback from the egress to the ingress of the SFC domain that is described in this document, to control or balance load within the SFC domain.

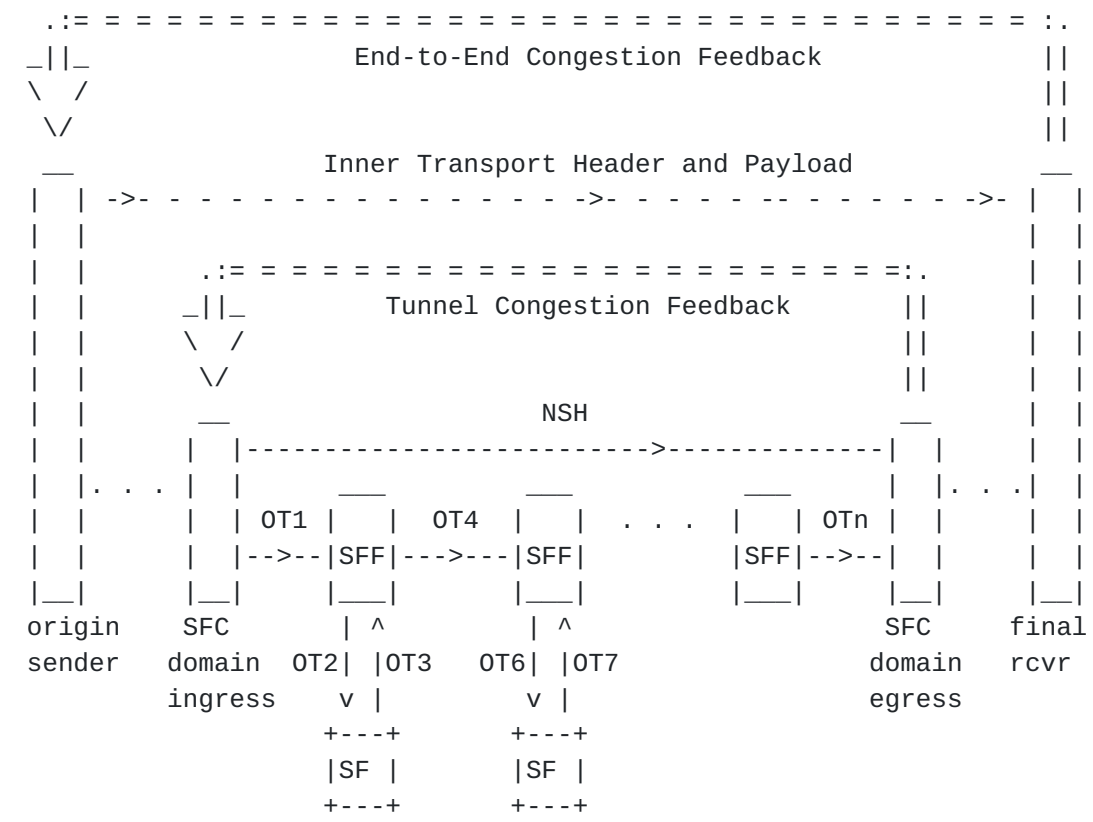


Figure 2. Congestion Feedback across an SFC Domain

SFC Domain congestion feedback in Figure 2 is shown within the context of an end-to-end congestion feedback loop. Also shown is the encapsulated layering of NSH headers within a series of outer transport headers (OT1, OT2, ... OTn).

1.4 Conventions Used in This Document

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.

Acronyms:

AQM - Active Queue Management [RFC7567]

CE - Congestion Experienced [RFC3168]

downstream - The direction from ingress to egress

ECN - Explicit Congestion Notification [RFC3168]

ECT - ECN Capable Transport [RFC3168]

IPFIX - IP Flow Information Export [RFC7011]

Not-ECT - Not ECN-Capable Transport [RFC3168]

NSH - Network Service Header [RFC8300]

SF - Service Function [RFC7665]

SFC - Service Function Chaining [RFC7665]

SFF - Service Function Forwarder [RFC7665] - A type of node that forwards based on the NSH.

TLV - Type Length Value

upstream - The direction from egress to ingress

2. The NSH ECN Field

The NSH header is used to encapsulate and control the subsequent path of traffic (see Section 2 of [RFC8300]). The NSH also provides for optional metadata inclusion, as shown in Figure 3.

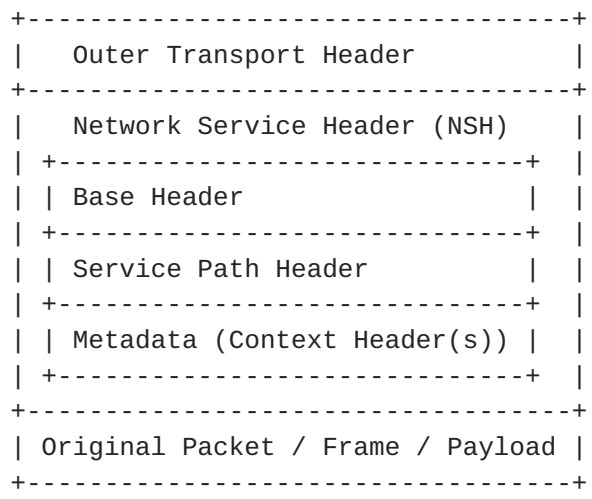


Figure 3. Data Encapsulation with the NSH

Two currently unused bits (indicated by "U") in the NSH Base Header (Section 2.2 of [RFC8300]) are allocated for ECN indication as shown in Figure 4.

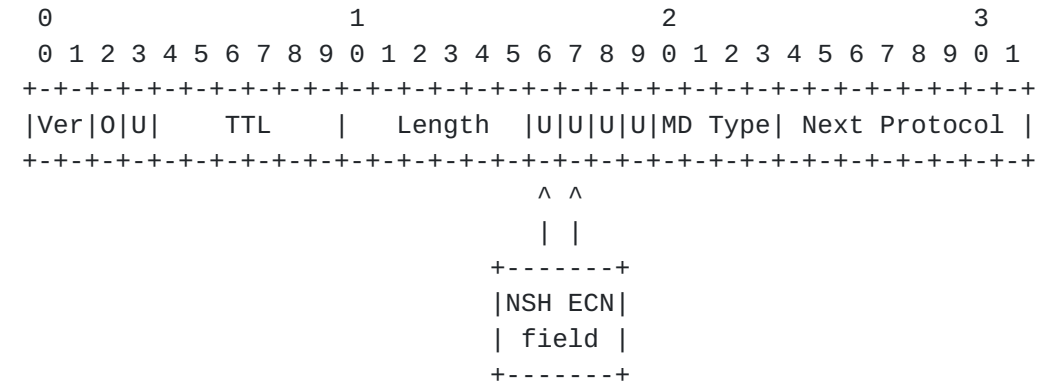


Figure 4. NSH Base Header

Note to RFC Editor: The above figure should be adjusted based on the bits assigned by IANA (see Section 5) and this note deleted.

Table 1 shows the meaning of the code points in the NSH ECN field. These have the same meaning as the ECN field code points in the IPv4 or IPv6 header as defined in [RFC3168].

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

Table 1. ECN Field Code Points

3. ECN Support in the NSH

This section describes the required behavior to support ECN using the NSH. There are two aspects to ECN support:

1. ECN propagation during encapsulation or decapsulation
2. ECN marking during congestion at bottlenecks.

While this section covers all combinations of ECN-aware and ECN-unaware, it is expected that in most cases the NSH domain will be uniform so that, if this document is applicable, all SFFs will support ECN; however, some legacy SFs might not support ECN.

ECN Propagation:

The specification of ECN tunneling [RFC6040] explains that an ingress must not propagate ECN support into an encapsulating header unless the egress supports correct onward propagation of the ECN field during decapsulation. We define Compliant ECN Decapsulation here as decapsulation compliant with either [RFC6040] or an earlier compatible equivalent ([RFC4301], or the full functionality mode of [RFC3168]).

The procedures in Section 3.2.1 ensure that each ingress of the large number of possible transport links within the SFC domain does not propagate ECN support into the encapsulating outer transport header unless the corresponding egress of that link supports Compliant ECN Decapsulation.

Section 3.3 requires that all the egress nodes of the SFC domain support Compliant ECN Decapsulation in conjunction with tunnel congestion feedback, otherwise the scheme in this document will not work.

ECN Marking:

At transit nodes the marking behavior specified in Section 3.2.1 is recommended and if not implemented at such transit nodes, there may be unmanaged congestion.

Detection of congestion will be most effective if ECN marking is supported by all potential bottlenecks inside the domain in which NSH is being used to route traffic as well as at the ingress and egress. Nodes that do not support ECN marking, or that support AQM but not ECN, will naturally use drop to relieve congestion. The gap in the end-to-end packet sequence will be detected as congestion by the final receiving endpoint, but not by the NSH egress (see Figure 2).

3.1 At The Ingress

When the ingress/Classifier encapsulates an incoming IP packet with an NSH, it MUST set the NSH ECN field using the "Normal mode" specified in [RFC6040] (i.e., copied from the incoming IP header).

Then, if the resulting NSH ECN field is Not-ECT, the ingress SHOULD set it to ECT(0). This indicates that, even though the end-to-end transport is not ECN-capable, the egress and ingress of the SFC domain are acting as an ECN-capable transport. This approach will inherently support all known variants of ECN, including the experimental L4S capability [RFC8311] [ecnL4S].

Packets arriving at the ingress might not use IP. If the protocol of arriving packets supports an ECN field similar to IP, the procedures for IP packets can be used. If arriving packets do not support an ECN field similar to IP, they MUST be treated as if they are Not-ECT IP packets.

Then, as the NSH encapsulated packet is further encapsulated with a transport header, if ECN marking is available for that transport (as it is for IP [RFC3168] and MPLS [RFC5129]), the ECN field of the transport header MUST be set using the "Normal mode" specified in [RFC6040] (i.e., copied from the NSH ECN field).

A summary of these normative steps is given in Table 2.

Incoming Header (also equal to departing Inner Header)	Departing NSH and Outer Headers
Not-ECT	ECT(0)
ECT(0)	ECT(0)
ECT(1)	ECT(1)
CE	CE

Table 2. Setting of ECN fields by an ingress/Classifier

The requirements in this section apply to all ingress nodes for the domain in which NSH is being used to route traffic.

3.2 At Transit Nodes

This section described behavior at nodes that forward based on the NSH such as SFF and other forwarding nodes such as IP routers. Figure 5 shows a packet on the wire between forwarding nodes.

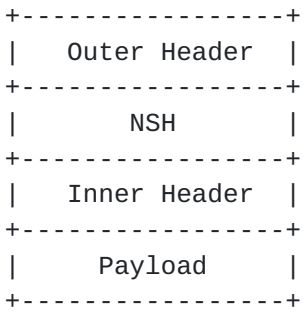


Figure 5. Packet in Transit

3.2.1 At NSH Transit Nodes

When a packet is received at an NSH based forwarding node such as an SFF, say N1, the outer transport encapsulation is removed and its ECN marking SHOULD be combined into the NSH ECN marking as specified in [RFC6040]. If this is not done, any congestion encountered at non-NSH transit nodes between N1 and the next upstream NSH based forwarding node will be lost and not transmitted downstream.

The NSH forwarding node SHOULD use a recognized AQM algorithm [RFC7567] to detect congestion. If the NSH ECN field indicates ECT, it will probabilistically set the NSH ECN field to the Congestion Experienced (CE) value or, in cases of extreme congestion, drop the packet.

When the NSH encapsulated packet is further encapsulated for transmission to the next SFF or SF, ECN marking behavior depends on whether or not the node that will decapsulate the outer header supports Compliant ECN Decapsulation (see Section 3). If it does, then the encapsulating node propagates the NSH ECN field to this outer encapsulation using the "Normal Mode" of ECN encapsulation [RFC6040] (the ECN field is copied). If it does not, then the encapsulating node MUST clear ECN in the outer encapsulation to non-ECT (the "Compatibility Mode" of [RFC6040]).

3.2.2 At an SF/Proxy

If the SF is NSH and ECN-aware, the processing is essentially the same at the SF as at an SFF as discussed in Section 3.2.1.

If the SF is NSH-aware but ECN-unaware, then the SFF transmitting the packet to the SF will use Compatibility Mode. Congestion encountered in the SFF to SF and SF to SFF paths will be unmanaged.

If the SF is not NSH-aware, then an NSH proxy will be between the SFF and the SF to avoid exposure of the NSH to the SF that does not understand NSHs as shown in Figure 6. This is described in Section 4.6 of [RFC7665]. The SF and proxy together look to the SFF like an NSH-aware SF. The behavior at the proxy and SF in this case is as below:

If such a proxy is not ECN-aware then congestion in the entire path from SFF to proxy to SF back to proxy to SFF will be unmanaged.

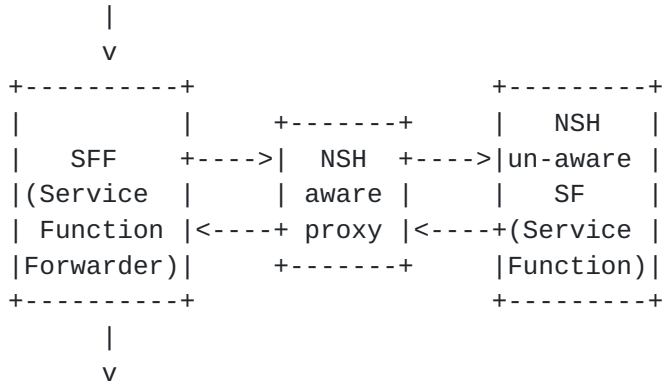


Figure 6. Proxy for NSH Un-aware SFF

If the proxy is ECN-aware, the proxy uses an AQM to indicate congestion within the proxy in the NSH that it returns to the SFF. The outer header used for the proxy-to-SF path uses Normal Mode. The outer head used for the proxy-to-SFF path uses Normal Mode based copying of the NSH ECN field to the outer header. Thus congestion in the proxy will be managed. Congestion in the SF will be managed only if the SF is ECN-aware and implements an AQM.

3.2.3 At Other Forwarding Nodes

Other forwarding nodes, that is non-NSH forwarding nodes between NSH forwarding nodes, such as IP or label switched routers, might also contain potential bottlenecks. If so, they SHOULD implement an AQM

algorithm to update the ECN marking in the outer transport header as

specified in [RFC3168].

3.3 At Exit/Egress

At the SFC domain egress node, first any actions are taken based on Congestion Experienced or other values of ECN marking, such as accumulating statistics to send back to the ingress (see Section 4) or for other uses. If the packet being carried inside the NSH is IP, when the NSH is removed the NSH ECN field MUST be combined with the IP ECN field as specified in Table 3 that was extracted from [RFC6040]. This requirement applies to all egress nodes for the domain in which NSH is being used to route traffic.

Arriving Inner Header	Arriving Outer Header			
	Not-ECT	ECT(0)	ECT(1)	CE
Not-ECT	Not-ECT	Not-ECT	Not-ECT	<drop>
ECT(0)	ECT(0)	ECT(0)	ECT(0)	CE
ECT(1)	ECT(1)	ECT(1)	ECT(1)	CE
CE	CE	CE	CE	CE

Table 3. Exit ECN Fields Merger

All the egress nodes of the SFC domain MUST support Compliant ECN Decapsulation as specified in this section. If this is not the case, the scheme described in this document will not work, and cannot be used.

3.4 Conservation of Packets

The SFC specification permits an SF to absorb packets and to generate new packets as well as simply processing and forwarding the packets it receives. Such actions might appear to be packet loss due to congestion or might mask the loss of packets by generating additional packets.

The tunnel congestion feedback approach (Section 4) detects loss by counting payload bytes in at the ingress and counting them out at the egress. This does not work unless nodes conserve the amount of payload bytes. Therefore, it will not be possible to detect loss using this technique if they are not conserved.

Nonetheless, if a bottleneck supports ECN marking, it will be possible to detect the very high level of CE markings that are associated with congestion that is so excessive that it leads to loss. However, it will not be possible for the tunnel congestion feedback approach to detect any congestion, whether slight or severe, if it occurs at a bottleneck that does not support ECN marking.

4. Tunnel Congestion Feedback Support

The collection and storage of congestion information at the egress may be useful for later analysis but, unless it can be fed back to a point which can take action to reduce congestion, it will not be useful in real time. Such congestion feedback to the ingress enables it to take actions such as those listed in Section 1.3.

IP Flow Information Export (IPFIX [RFC7011]) provides a standard for communicating traffic flow statistics. As extended by this document, IPFIX messages from the egress to the ingress are used to communicate the extent of congestion between an ingress and egress based on ECN marking in the NSH.

4.1 Congestion Level Measurement

The congestion level measurement is based on ECN marking in the NSH and packet drop, particularly the fraction of packets that are CE-marked packet and fraction that are dropped. If the congestion level is not high enough, the packets are marked as CE instead of being dropped, and then it is easy to calculate congestion level according to the ratio of CE-marked packets. If the congestion level is so high that ECT packets will be dropped, then the packet loss ratio could be calculated by comparing total packets entering ingress and total packets arriving at egress over the same span of packets. If packet loss is detected, it could be assumed that severe congestion has occurred in the tunnel.

The egress calculates the CE-marked packet ratio by counting packets with different ECN markings. The CE-marked packet ratio will be used as an indication of tunnel load level. It is assumed that nodes between the ingress and egress will not drop packets biased towards certain ECN codepoints, so calculating of CE-marked packet ratio is not affect by packet drop.

The calculation of volumes of packet drop is by comparing the traffic volumes between ingress and egress.

Faked ECN-Capable Transport (ECT) is used at the ingress to defer packet loss to the egress. The basic idea of faked ECT is that, when encapsulating packets, the ingress first marks the tunnel outer header (NSH for an SFC domain) according to [RFC6040], and then remarks the outer header of Not-ECT packets as ECT. (ECT(0) and ECT(1) are treated as the same.) Thus, as transmitted by the ingress node, there will be one of three combinations of outer header ECN field and inner header ECN field: CE|CE, ECT|N-ECT, and ECT|ECT (in the format of outer-ECN|inner-ECN); when decapsulating packets at the

egress, [RFC6040] defined decapsulation behavior is used, and

according to [RFC6040], the packets marked as CE|N-ECT will be dropped. Faked-ECT is used to shift some drops to the egress in order to allow the egress to calculate the CE-marked packet ratio more precisely.

The ingress encapsulates packets and marks their outer header according to faked ECT as described above. The ingress cumulatively counts packet bytes for three types of ECN combination (CE|CE, ECT|N-ECT, and ECT|ECT) and then the ingress regularly sends cumulative bytes counts message of each type of ECN combination to the egress.

When each message arrives at the egress, (1) egress calculates the ratio of CE-marked packet; (2) the egress cumulatively counts packet bytes coming from the ingress and adds its own bytes counts of each type of ECN combination (CE|CE, ECT|N-ECT, CE|N-ECT, CE|ECT, and ECT|ECT) to the message for ingress to calculate packet loss. The egress feeds back CE-marked packet ratio and bytes counts information to the ingress for evaluating congestion level in the tunnel.

The counting of bytes can be at the granularity of all traffic from the ingress to the egress to learn about the overall congestion status of the path between the ingress and the egress. The counting can also be at the granularity of individual customer's traffic or a specific set of flows to learn about their congestion contribution.

For example, the tunnelEcnCEMarkedRatio field (specified below) indicates the fraction of traffic that has been marked in the ECN field of the NSH as Congestion Experienced (CE).

4.3 Congestion Information Delivery

As described above, the tunnel ingress needs to send a messages containing cumulative bytes counts of packets of each type of ECN combination to the tunnel egress, and the tunnel egress also needs to feed back messages with cumulative bytes counts of packets of each type of ECN combination and CE-marked packet ratio to the ingress. This section specifies how the messages should be conveyed.

IPFIX recommends, but does not require, use of SCTP [RFC4960] in partial reliability mode [RFC3758] for the transport of its messages. This mode allows loss of some packets, which is tolerable because IPFIX communicates cumulative statistics. IPFIX over SCTP over IP SHOULD be used directly where there is IP connectivity between the ingress and egress; however, there might be different transport protocols or address spaces used in different regions of an SFC domain that make such direct IP connectivity problematic. The NSH provides the general method of routing traffic within an SFC domain

so the encapsulation of the required IPFIX traffic in NSH MUST be

implemented and, when IP connectivity is not available, IPFIX over NSH SHOULD be used along with configuration of appropriate SFC paths for the IPFIX over NSH traffic.

Typically IPFIX messages could travel along the same path as network data traffic. In any case, an IPFIX message packet may get lost in case of network congestion. Even though the missing information could be recovered because of the use of cumulative counts, the message SHOULD be transmitted at a higher priority than users' traffic flows.

The ingress node can do congestion management at different granularity which means both the overall aggregated inner tunnel congestion level and congestion level contributed by certain traffic flows could be measured for different congestion management purpose. For example, if the ingress only wants to limit congestion volume caused by certain traffic flows, such as UDP-based traffic, then congestion volume for that traffic will be fed back; or if the ingress is doing overall congestion management, the aggregated congestion volume will be fed back.

When sending IPFIX messages from ingress to egress, the ingress acts as IPFIX exporter and egress acts as IPFIX collector; When feedback congestion level information from egress to ingress, then the egress acts as IPFIX exporter and ingress acts as IPFIX collector.

The combination of congestion level measurement and congestion information delivery procedures should be as following:

- The ingress node determines the IPFIX template record to be used. The template record can be pre-configured or determined at runtime, the content of template record will be determined according to the granularity of congestion management; if the ingress wants to limit congestion volume contributed by specific traffic flow then the elements such as source IP address, destination IP address, flow id and CE-marked packet volume of the flow, etc., will be included in the template record.

- Metering on the ingress measures traffic volume according to template record chosen and then the measurement records are sent to the egress.

- Metering on the egress measures congestion level information according to template record which should be the same as the template record sent by the ingress.

- The egress sends measurement records together with the measurement records of ingress back to the ingress.

4.3 IPFIX Extensions

This section specifies new IPFIX Information Elements according to [RFC7013].

4.3.1 nshServicePathID

In order to identify SFC flows, so that congestion can be measured and reported at that granularity, it is necessary for IPFIX to be able to classify traffic based on the Service Path Identifier field of the NSH [RFC8300]. Thus an NSH Service Path Identifier (nshServicePathID) IPFIX Information Element [RFC7012] is specified.

Name: nshServicePathID

Description: Network Service Header [RFC8300] Service Path Identifier. This is a 24-bit value which is left justified in the Information Element. The low order byte MUST be sent as zero and ignored on receipt.

Abstract Data Type: unsigned32

Data Type Semantics: identifier

ElementId: tbd0

Status: current

4.3.2 tunnelEcnCeCeByteTotalCount

Description: The total number of bytes of incoming packets with CE|CE ECN marking combination at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD1

Statues: current

Units: bytes

4.3.3 tunnelEcnEctNectBytetTotalCount

Description: The total number of bytes of incoming packets with ECT|N-ECT ECN marking combination (ECT(0) and ECT(1) are treated as the same) at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD2

Statues: current

Units: bytes

4.3.4 tunnelEcnCeNectByteTotalCount

Description: The total number of bytes of incoming packets with CE|N-ECT ECN marking combination at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD3

Statues: current

Units: bytes

4.3.5 tunnelEcnCeEctByteTotalCount

Description: The total number of bytes of incoming packets with CE|ECT ECN marking combination (ECT(0) and ECT(1) are treated as the same) at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD4

Statues: current

Units: bytes

4.3.6 tunnelEcnEctEctByteTotalCount

Description: The total number of bytes of incoming packets with ECT|ECT ECN marking combination (ECT(0) and ECT(1) are treated as the same) at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD5

Statues: current

Units: bytes

4.3.7 tunnelEcnCEMarkedRatio

Description: The ratio of CE-marked Packet at the Observation Point.

Abstract Data Type: float32

ElementId: TBD6

Statues: current

5. Example of Use

This subsection provides an example of how the solution described in this document could work.

First of all, IPFIX template records are exchanged between ingress and egress to negotiate the format of data records. The example here is to measure the congestion level for the overall tunnel caused by all the traffic. After the negotiation is finished, the ingress sends in-band messages to egress containing the number of each kind of ECN-marked packets (i.e.. CE|CE, ECT|N-ECT and ECT|ECT) received until it sent the message.

After the egress receives the message, the egress calculates the CE-marked packet ratio and counts the number of different kinds of ECN-marking packets received until it received the message. Then the egress sends a feedback message containing the counts together with the information in the ingress's message back to the ingress.

Figures 7 to 10 below show the example procedure between ingress and egress.

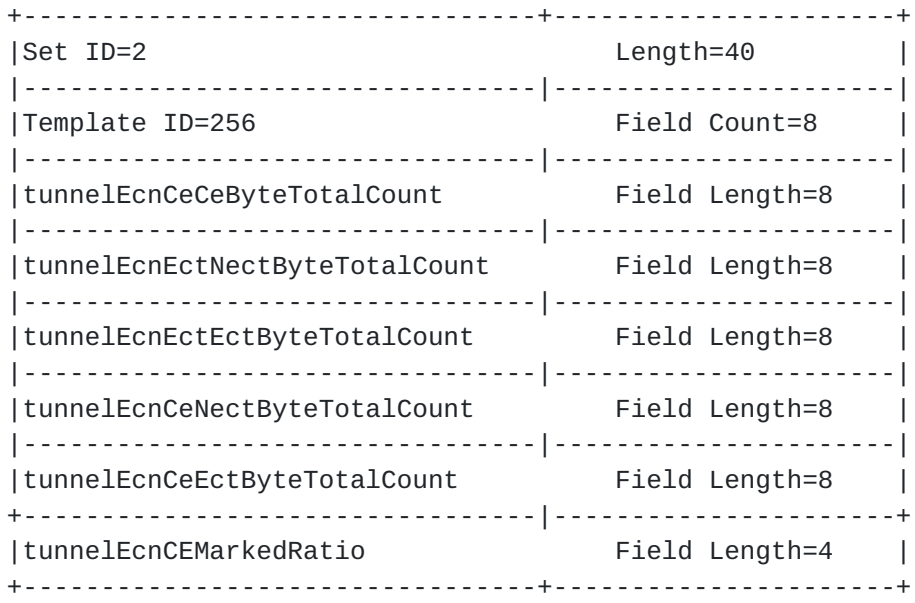


Figure 7. Template Record Sent From Egress to Ingress

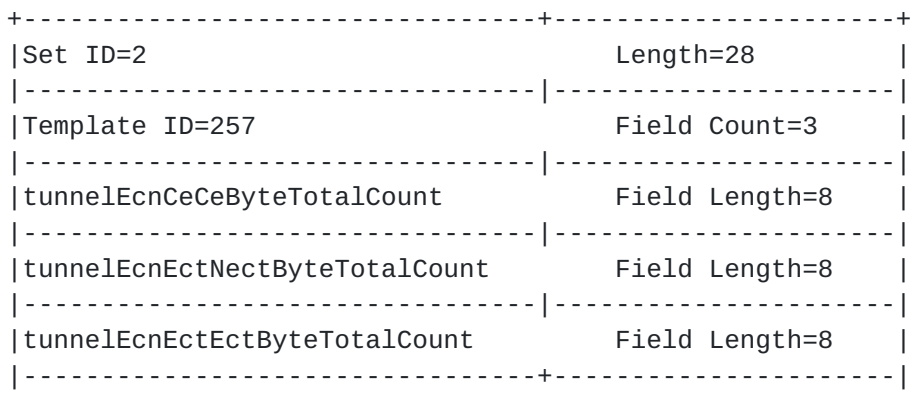


Figure 8. Template Record Sent From Ingress to Egress

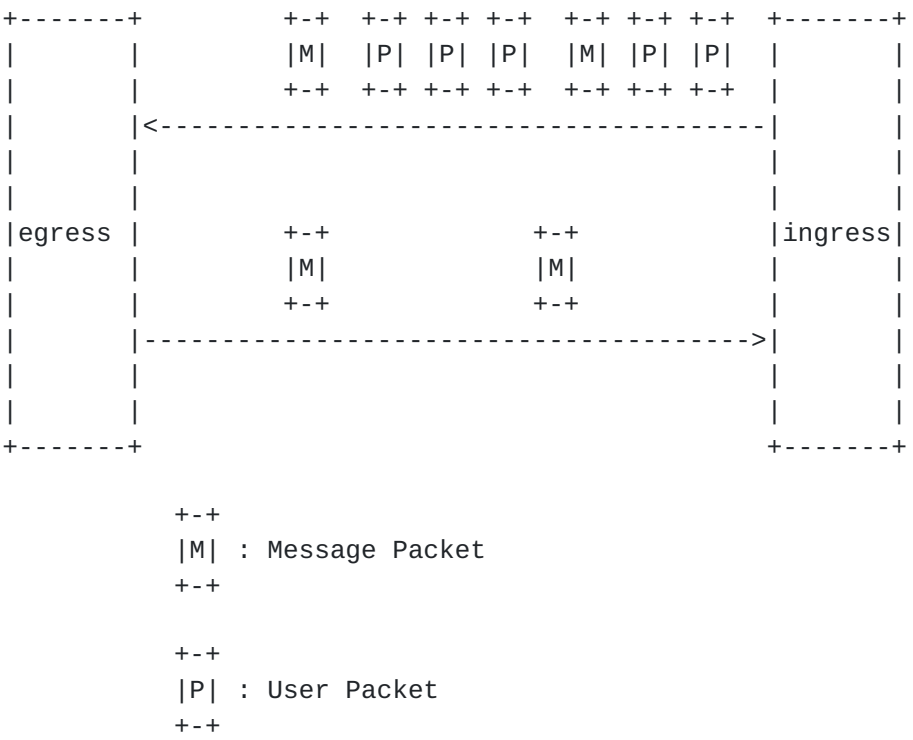


Figure 9. Traffic flow Between Ingress and Egress

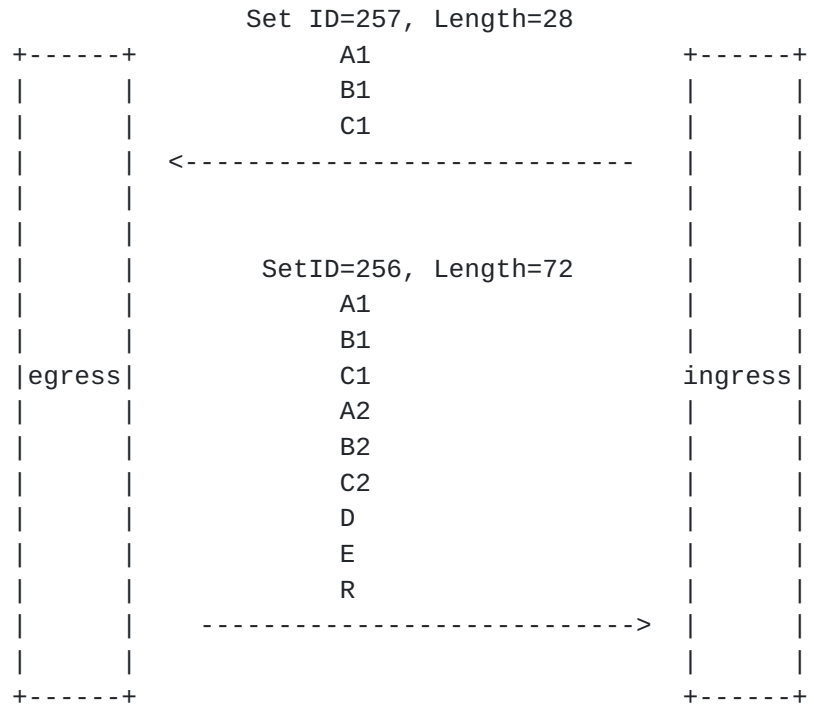


Figure 10. Messages Between Ingress and Egress

The following provides an example of how the tunnel congestion level could be calculated:

The congestion Level could be divided into two categories: (1) slight congestion (no packets dropped); (2) serious congestion (packets are being dropped).

For slight congestion, the congestion level is indicated as the ratio of CE-marked packet:

ce_marked = R;

For serious congestion, the congestion level is indicated as the number of volume loss:

total_ingress = (A1 + B1 + C1)

total_egress = (A2 + B2 + C2 + D + E)

volume_loss = (total_ingress - total_egress)

6. IANA Considerations

The following subsections provide IANA assignment considerations.

6.1 SFC NSH Header ECN Bits

IANA is requested to assign two contiguous bits in the NSH Base Header Bits registry for ECN (bits 16 and 17 suggested) and note this assignment as follows:

Bit	Description	Reference
tbd(16-17)	NSH ECN	[this document]

6.2 IPFIX Information Element IDs

IANA is requested to assign IPFIX Information Element IDs as follows:

ElementID: tbd0
Name: nshServicePathID
Data Type: unsigned32
Data Type Semantics: identifier
Status: current
Description: The Network Service Header [RFC8300] Service Path Identifier.

ElementID: TBD1
Name: tunnelEcnCeCePacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description: The total number of bytes of incoming packets with CE|CE ECN marking combination at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: octets

ElementID: TBD2
Name: tunnelEcnEctNectPacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description: The total number of bytes of incoming packets with ECT|N-ECT ECN marking combination at the Observation Point since the Metering Process (re-)initialization for this

Observation Point.

```
      Units: octets

      ElementID: TBD3
      Name: tunnelEcnCeNectPacketTotalCount
      Data Type: unsigned64
      Data Type Semantics: totalCounter
      Status: current
      Description: The total number of bytes of incoming packets with
         CE|N-ECT ECN marking combination at the Observation Point since
         the Metering Process (re-)initialization for this Observation
         Point.
      Units: octets

      ElementID: TBD4
      Name: tunnelEcnCeEctPacketTotalCount
      Data Type: unsigned64
      Data Type Semantics: totalCounter
      Status: current
      Description: The total number of bytes of incoming packets with
         CE|ECT ECN marking combination at the Observation Point since
         the Metering Process (re-)initialization for this Observation
         Point.
      Units: octets

      ElementID: TBD5
      Name: tunnelEcnEctEctPacketTotalCount
      Data Type: unsigned64
      Data Type Semantics: totalCounter
      Status: current
      Description: The total number of bytes of incoming packets with
         CE|ECT(0) ECN marking combination at the Observation Point
         since the Metering Process (re-)initialization for this
         Observation Point.
      Units: octets

      ElementID: TBD6
      Name: tunnelEcnCEMarkedRatio
      Data Type: float32
      Status: current
      Description: The ratio of CE-marked Packet at the Observation
         Point.
```

7. Security Considerations

For general NSH security considerations, see [RFC8300].

For security considerations concerning tampering with ECN signaling, see [RFC3168]. For security considerations concerning ECN and encapsulation, see [RFC6040].

For general IPFIX security considerations, see [RFC7011]. If deployed in an untrusted environment, the signaling traffic between ingress and egress can be protected utilizing the security mechanisms provided by IPFIX (see Section 11 in [RFC7011]).

The tunnel endpoints (the ingress and egress for an SFC domain) are assumed to be in the same administrative domain, so they will trust each other.

The solution in this document does not introduce any greater potential to invade privacy than would have been available without the solution.

8. Acknowledgements

Most of the material on Tunnel Congestion Feedback was originally in draft-ietf-twvwg-tunnel-congestion-feedback. After discussion with the authors of that draft, the authors of this draft, and the Chairs of the TSVWG and SFC Working Groups, the Tunnel Congestion Feedback draft was merged into this draft.

The authors wish to thank the following for their comments, suggestions, and reviews:

David Black, Sami Boutros, Anthony Chan, Lingli Deng, Liang Geng, Joel Halpern, Jake Holland, John Kaippallimalil, Tal Mizrahi, Vincent Roca, Lei Zhu

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, <http://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, <http://www.rfc-editor.org/info/rfc3168>.

[RFC3758] - Stewart, R., Ramalho, M., Xie, Q., Tuexen, M., and P. Conrad, "Stream Control Transmission Protocol (SCTP) Partial Reliability Extension", RFC 3758, DOI 10.17487/RFC3758, May 2004, <https://www.rfc-editor.org/info/rfc3758>.

[RFC5129] - Davie, B., Briscoe, B., and J. Tay, "Explicit Congestion Marking in MPLS", RFC 5129, DOI 10.17487/RFC5129, January 2008, <https://www.rfc-editor.org/info/rfc5129>.

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

[RFC7011] - Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <https://www.rfc-editor.org/info/rfc7011>.

[RFC7013] - Trammell, B. and B. Claise, "Guidelines for Authors and Reviewers of IP Flow Information Export (IPFIX) Information Elements", BCP 184, RFC 7013, DOI 10.17487/RFC7013, September 2013, <https://www.rfc-editor.org/info/rfc7013>.

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

[RFC8174] - Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <http://www.rfc-editor.org/info/rfc8174>

[RFC8300] - Quinn, P., Ed., Elzur, U., Ed., and C. Pignataro, Ed., "Network Service Header (NSH)", RFC 8300, DOI 10.17487/RFC8300, January 2018, <https://www.rfc-editor.org/info/rfc8300>.

Informative References

[RFC4301] - Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, DOI 10.17487/RFC4301, December 2005, <https://www.rfc-editor.org/info/rfc4301>.

[RFC4960] - Stewart, R., Ed., "Stream Control Transmission Protocol", RFC 4960, DOI 10.17487/RFC4960, September 2007, <https://www.rfc-editor.org/info/rfc4960>.

[RFC7012] - Claise, B., Ed., and B. Trammell, Ed., "Information Model for IP Flow Information Export (IPFIX)", RFC 7012, DOI 10.17487/RFC7012, September 2013, <https://www.rfc-editor.org/info/rfc7012>.

[RFC7665] - Halpern, J., Ed., and C. Pignataro, Ed., "Service Function Chaining (SFC) Architecture", RFC 7665, DOI 10.17487/RFC7665, October 2015, <https://www.rfc-editor.org/info/rfc7665>.

[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>.

[ecnL4S] - De Schepper, K., and B. Briscoe, "Identifying Modified Explicit Congestion Notification (ECN) Semantics for Ultra-Low Queuing Delay (L4S)", draft-ietf-tsvwg-ecn-l4s-id, work in progress.

Authors' Addresses

Donald E. Eastlake, 3rd
Futurewei Technologies
2386 Panoramic Circle
Apopka, FL 32703 USA

Tel: +1-508-333-2270
Email: d3e3e3@gmail.com

Bob Briscoe
Independent
UK

Email: ietf@bobbriscoe.net
URI: http://bobbriscoe.net/

Yizhou Li
Huawei Technologies
101 Software Avenue,
Nanjing 210012, P. R China

Phone: +86-25-56624584
EMail: liyizhou@huawei.com

Andrew G. Malis
Malis Consulting

Email: agmalis@gmail.com

Xinpeng Wei
Huawei Technologies
Beiqing Rd. Z-park No.156, Haidian District,
Beijing, 100095, P. R. China

EMail: weixinpeng@huawei.com

Copyright and IPR Provisions

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License. The definitive version of an IETF Document is that published by, or under the auspices of, the IETF. Versions of IETF Documents that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of IETF Documents. The definitive version of these Legal Provisions is that published by, or under the auspices of, the IETF. Versions of these Legal Provisions that are published by third parties, including those that are translated into other languages, should not be considered to be definitive versions of these Legal Provisions. For the avoidance of doubt, each Contributor to the IETF Standards Process licenses each Contribution that he or she makes as part of the IETF Standards Process to the IETF Trust pursuant to the provisions of RFC 5378. No language to the contrary, or terms, conditions or rights that differ from or are inconsistent with the rights and licenses granted under RFC 5378, shall have any effect and shall be null and void, whether published or posted by such Contributor, or included with or in such Contribution.