Internet Engineering Task Force
INTERNET-DRAFT
Intended Status: Informational
Expires: July 29, 2017

X. Wei
Huawei Technologies
L.Zhu
Huawei Technologies
L.Deng
China Mobile

January 25, 2017

Tunnel Congestion Feedback
draft-ietf-tsvwg-tunnel-congestion-feedback-04

Abstract

This document describes a method to measure congestion on a tunnel segment based on recommendations from RFC 6040, "Tunneling of Explicit Congestion Notification", and to use IPFIX to communicate the congestion measurements from the tunnel's egress to a controller which can respond by modifying the traffic control policies at the tunnel's ingress.

Status of this Memo

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

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

Copyright and License Notice

Table of Contents

1. Introduction

In IP networks, persistent congestion[RFC2914] lowers transport throughput, leading to waste of network resource. Appropriate congestion control mechanisms are therefore critical to prevent the network from falling into the persistent congestion state. Currently, transport protocols such as TCP[RFC793], SCTP[RFC4960], DCCP[RFC4340], have their built-in congestion control mechanisms, and even for certain single transport protocol like TCP there can be a couple of different congestion control mechanisms to choose from. All these congestion control mechanisms are implemented on host side, and there are reasons that only host side congestion control is not sufficient for the whole network to keep away from persistent congestion. For example, (1) some protocol's congestion control scheme may have internal design flaws; (2) improper software implementation of protocol; (3) some transport protocols, e.g. RTP[RFC3550] do not even provide congestion control at all.

Tunnels are widely deployed in various networks including public Internet, data center network, and enterprise network etc. A tunnel consists of ingress, egress and a set of intermediate routers. For the tunnel scenario, a tunnel-based mechanism is introduced for network traffic control to keep the network from persistent congestion. Here, tunnel ingress will implement congestion management function to control the traffic entering the tunnel.

This document provides a mechanism of feeding back inner tunnel congestion level to the ingress. Using this mechanism the egress can feed the tunnel congestion level information it collects back to the ingress. After receiving this information the ingress will be able to perform congestion management according to network management policy.

2. Conventions And Terminologies

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this

document are to be interpreted as described in RFC 2119 [RFC2119]

DP: Decision Point, an logical entity that makes congestion management decision based on the received congestion feedback information.

AP: Action Point, an logical entity that implements congestion management action according to the decision made by Decision Point.

ECT: ECN-Capable Transport code point defined in RFC3168.

3. Congestion Information Feedback Models

The feedback model mainly consists of tunnel egress and tunnel ingress. The tunnel egress composes of meter function and exporter function; tunnel ingress composes AP (Action Point) function, collector function and DP (Decision Point) function.

The Meter function collects network congestion level information, and conveys the information to Exporter which feeds back the information to the collector function.

The collector collects congestion level information from exporter, after that congestion management Decision Point (DP) function will make congestion management decision based on the information from collector.

The action point controls the traffic entering tunnel, and it implements traffic control decision of DP.

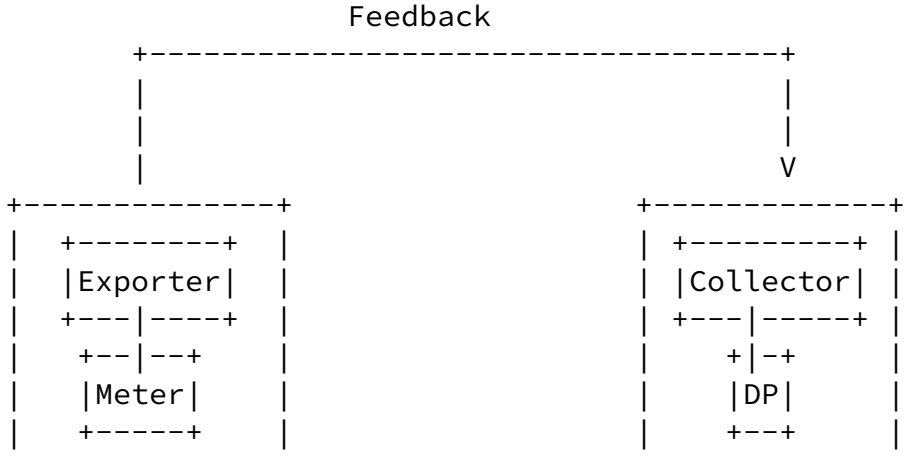


Figure 1: Feedback Model.

4. Congestion Level Measurement

This section describes how to measure congestion level in a tunnel.

The congestion level measurement is based on ECN (Explicit Congestion Notification) [RFC3168] and packet drop. If the routers support ECN, after router's queue length is over a predefined threshold, the routers will mark the ECN-capable packets as Congestion Experienced (CE) or drop not-ECT packets with the

probability proportional to queue length; if the queue overflows all packets will be dropped. If the routers do not support ECN, after router's queue length is over a predefined threshold, the routers will drop both the ECN-capable packets and the not-ECT packets with the probability proportional to the queue length.

The network congestion level could be indicated through the ratio of CE-marked packet and the ratio of packet drop, the relationship between these two kinds of indicator is complementary. If the congestion level in tunnel is not high enough, the packets would be 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 packet 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. Because loss is only ever a sign of serious congestion, so it doesn't need to measure loss ratio accurately.

Faked ECN-capable transport (ECT) is used at ingress to defer

packet loss to egress. The basic idea of faked ECT is that, when encapsulating packets, ingress first marks tunnel outer header according to RFC6040, and then remarks outer header of Not-ECT packet as ECT, there will be three kinds of combination of outer header ECN field and inner header ECN field: CE|CE, ECT|N-ECT, ECT|ECT (in the form of outer ECN| inner ECN); when decapsulating packets at egress, RFC6040 defined decapsulation behavior is used, and according to RFC6040, the packets marked as CE|N-ECT will be dropped by egress.

To calculate congestion level, for the same span of packets, the number of each kind of ECN marking packet at ingress and egress will be compared to get the volume of CE-marked packet in the tunnel; and the total number of packets at ingress and egress will be compared to detect the packet loss.

The basic procedure of congestion level measurement is as follows:

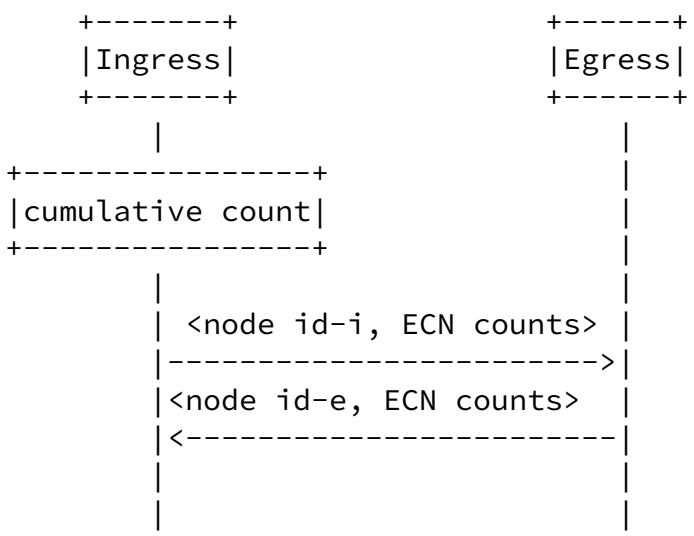


Figure 2: Procedure of Congestion Level Measurement

Ingress encapsulates packets and marks outer header according to faked ECT as described above. Ingress cumulatively counts packets for three types of ECN combination (CE|CE, ECT|N-ECT, ECT|ECT) and then the ingress regularly sends cumulative packet counts message of each type of ECN combination to the egress. When each message arrives, the egress cumulatively counts packets coming from the ingress and adds its own packet counts of each type of ECN combination (CE|CE, ECT|N-ECT, CE|N-ECT, CE|ECT, ECT|ECT) to the message and returns the whole message to the ingress.

The counting of packets can be at the granularity of the 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.

5. Congestion Information Delivery

As described above, the tunnel ingress needs to convey a message containing cumulative packet counts of each type of ECN combination to tunnel egress, and the tunnel egress also needs to feed back the message of cumulative packet counts of each type of ECN combination to the ingress. This section describes how the messages should be conveyed.

The message travels along the same path with network data traffic, referred as in-band signal. Because the message is transmitted in band, so the message packet may get lost in case of network congestion. To cope with the situation that the message packet gets lost, the packet counts values are sent as cumulative counters. Then

if a message is lost the next message will recover the missing information. Even though the missing information could be recovered, the message should be transmitted in a much higher priority than users' traffic flows.

IPFIX [RFC7011] is selected as information feedback protocol. IPFIX uses preferably SCTP as transport. SCTP allows partially reliable delivery [RFC3758], which ensures the feedback message will not be

blocked in case of packet loss due to network congestion.

Ingress can do congestion management at different granularity which means both the overall aggregated inner tunnel congestion level and congestion level contributed by certain traffic(s) could be measured for different congestion management purpose. For example, if the ingress only wants to limit congestion volume caused by certain traffic(s),e.g UDP-based traffic, then congestion volume for the traffic will be fed back; or if the ingress do overall congestion management, the aggregated congestion volume will be fed back.

When sending message 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 procedure should be as following:

The ingress determines IPFIX template record to be used. The template record can be preconfigured 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.

Meter on ingress measures traffic volume according to template record chosen and then the measurement records are sent to egress in band.

Meter on egress measures congestion level information according to template record, the content of template record should be the same as template record of ingress.

Exporter of egress sends measurement record together with the measurement record of ingress back to the ingress.

5.1 IPFIX Extensions

This sub-section defines a list of new IPFIX Information Elements

according to RFC7013 [RFC7013].

5.1.1 tunnelEcnCeCePacketTotalCount

Description: The total number of incoming packets with CE|CE ECN marking combination for this Flow 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: packets

5.1.2 tunnelEcnEct0NectPacketTotalCount

Description: The total number of incoming packets with ECT(0)|N-ECT ECN marking combination for this Flow 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: packets

5.1.3 tunnelEcnEct1NectPacketTotalCount

Description: The total number of incoming packets with ECT(1)|N-ECT ECN marking combination for this Flow 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: packets

5.1.4 tunnelEcnCeNectPacketTotalCount

Description: The total number of incoming packets with CE|N-ECT ECN marking combination for this Flow 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: packets

5.1.5 tunnelEcnCeEct0PacketTotalCount

Description: The total number of incoming packets with CE|ECT(0) ECN marking combination for this Flow 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: packets

5.1.6 tunnelEcnCeEct1PacketTotalCount

Description: The total number of incoming packets with CE|ECT(1) ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD6

Statues: current

Units: packets

5.1.7 tunnelEcnEct0Ect0PacketTotalCount

Description: The total number of incoming packets with ECT(0)|ECT(0) ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD7

Statues: current

Units: packets

5.1.8 tunnelEcnEct1Ect1PacketTotalCount

Description: The total number of incoming packets with ECT(1)|ECT(1) ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

ElementId: TBD8

Statues: current

Units: packets

6. Congestion Management

After tunnel ingress receives congestion level information, then congestion management actions could be taken based on the information, e.g. if the congestion level is higher than a predefined

threshold, then action could be taken to reduce the congestion level.

The design of network side congestion management SHOULD take host side e2e congestion control mechanism into consideration, which means the congestion management needs to avoid the impacts on e2e congestion control. For instance, congestion management action must be delayed by more than a worst-case global RTT (e.g. 100ms), otherwise tunnel traffic management will not give normal e2e congestion control enough time to do its job, and the system could go

unstable.

The detailed description of congestion management is out of scope of this document, as examples, congestion management such as circuit breaker [CB] could be applied. Circuit breaker is an automatic mechanism to estimate congestion, and to terminate flow(s) when persistent congestion is detected to prevent network congestion collapse.

6.1 Example

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 record, the example here is to measure the congestion level for the overall tunnel (caused by all the traffic in tunnel). After the negotiation is finished, ingress sends in-band message to egress, the message contains the number of each kind of ECN-marked packets (i.e. CE|CE, ECT|N-ECT and ECT|ECT) received until the sending of message.

After egress receives the message, the egress counts number of different kinds of ECN-marking packets received until receiving the message, then the egress sends a feedback message containing the counts together with the information in ingress's message to ingress.

Figure 3 to Figure 6 below show the example procedure between ingress and egress.

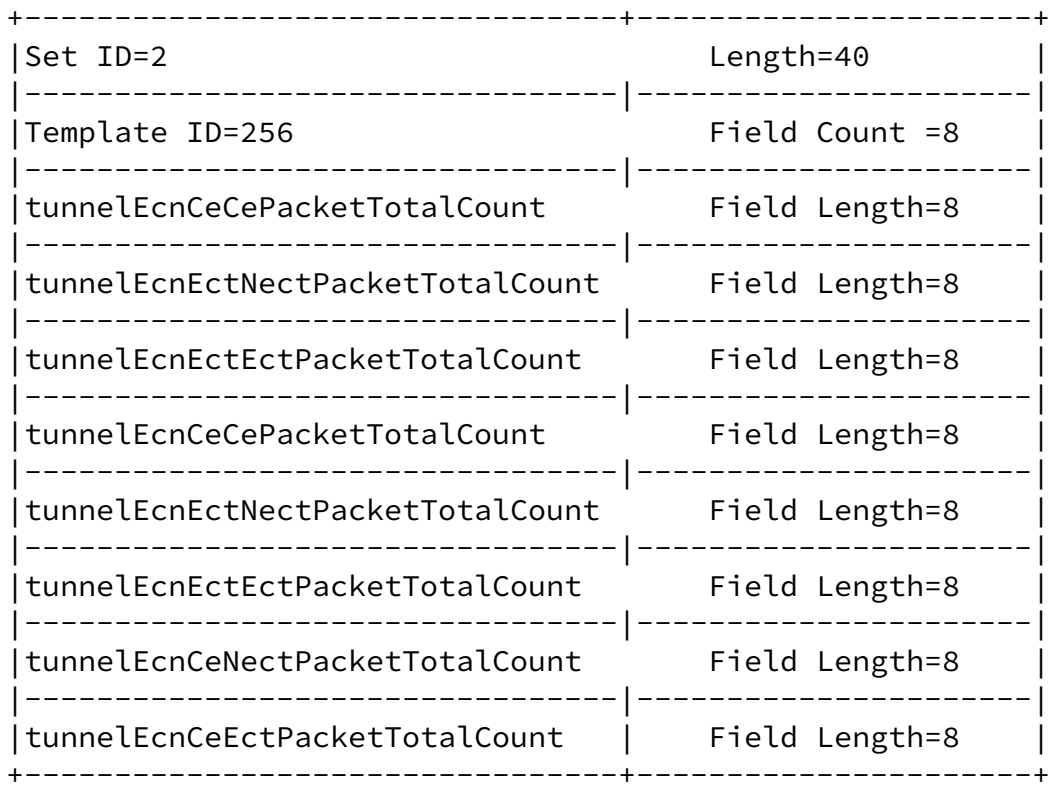


Figure 3: Template Record Sent From Egress to Ingress

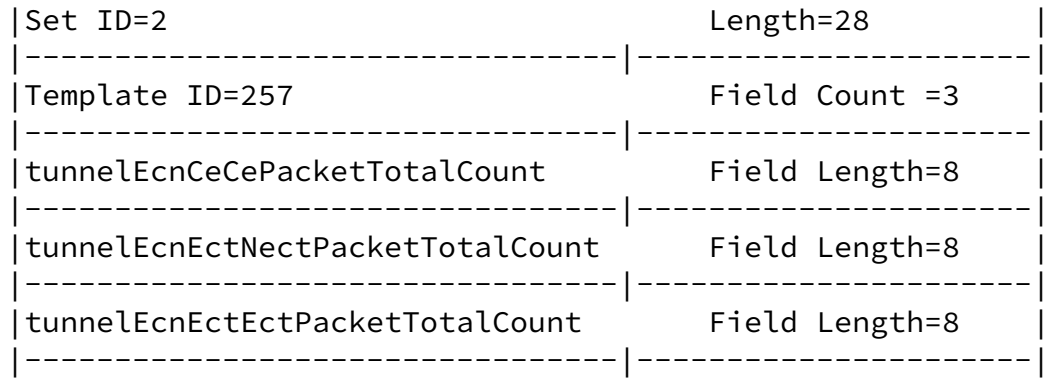
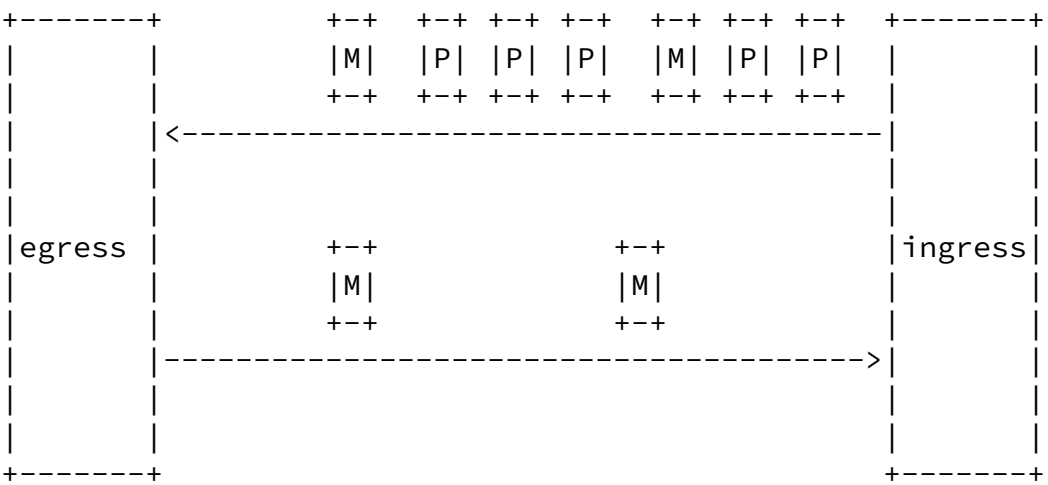


Figure 4: Template Record Sent From Ingress to Egress



```
+-+
|P| : User Packet
+-+
```

Figure 5 Traffic flow Between Ingress and Egress

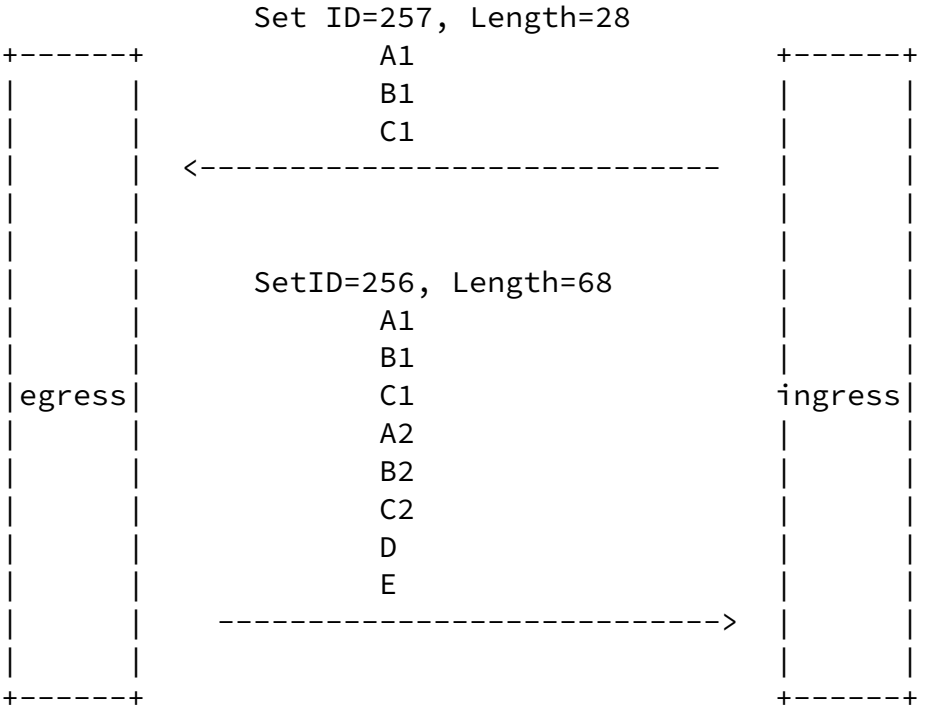


Figure 6: Message Between Ingress and Egress

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

Congestion Level could be divided into two categories:(1)slight congestion(no packets dropped); (2)serious congestion (packet dropping happen).

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

ce_marked = (A2 + D + E) - A1;

For serious congestion, the congestion level is indicated as the number of lost packets:

total_ingress = (A1 + B1 + C1)

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

packet_loss = (total_ingress - total_egress)

7. Security Considerations

This document describes the tunnel congestion calculation and feedback.

The tunnel endpoints are assumed to be deployed in the same administrative domain, so the ingress and egress will trust each other, the signaling traffic between ingress and egress will be protected utilizing security mechanism provided IPFIX (see section 11 in RFC7011).

From the consideration of privacy point of view, in case of fine grained congestion management, ingress is aware of the amount of traffic for specific application flows inside the tunnel which seems to be an invasion of privacy. But in any way, the ingress could The solution doesn't introduce more privacy problem.

8. IANA Considerations

This document defines a set of new IPFIX Information Elements (IE),which need to be registered at IANA IPFIX Information Element Registry.

ElementID: TBD1
Name:tunnelEcnCeCePacketTotalCount
Data Type: unsigned64

Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with CE|CE ECN marking combination for this Flow at the Observation Point since the

Metering Process (re-)initialization for this Observation Point.
Units: packets

ElementID: TBD2
Name:tunnelEcnEct0NectPacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with ECT(0)|N-ECT ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: packets

ElementID: TBD3
Name: tunnelEcnEct1NectPacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with ECT(1)|N-ECT ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: packets

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

ElementID: TBD5
Name:tunnelEcnCeEct0PacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with CE|ECT(0) ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: packets

ElementID: TBD6

Name:tunnelEcnCeEct1PacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with CE|ECT(1) ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: packets

ElementID: TBD7
Name:tunnelEcnEct0Ect0PacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with ECT(0)|ECT(0) ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: packets

ElementID: TBD8
Name:tunnelEcnEct1Ect1PacketTotalCount
Data Type: unsigned64
Data Type Semantics: totalCounter
Status: current
Description:The total number of incoming packets with ECT(1)|ECT(1)ECN marking combination for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.
Units: packets

[TO BE REMOVED: This registration should take place at the following location: http://www.iana.org/assignments/ipfix/ipfix.xhtml#ipfix-information-elements]

9. References

9.1 Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, 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, September 2001, <http://www.rfc-editor.org/info/rfc3168>.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003, <http://www.rfc-editor.org/info/rfc3550>.

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

[RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", RFC 4340, March 2006, <http://www.rfc-editor.org/info/rfc4340>.

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

[RFC6040] Briscoe, B., "Tunnelling of Explicit Congestion Notification", RFC 6040, 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, September 2013, <http://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, September 2013, <http://www.rfc-editor.org/info/rfc7013>.

[CONEX] Matt Mathis, Bob Briscoe. "Congestion Exposure (ConEx) Concepts, Abstract Mechanism and Requirements", RFC7713, December 2015

9.2 Informative References

[CB] G. Fairhurst. "Network Transport Circuit Breakers", draft-ietf-tsvwg-circuit-breaker-01, April 02, 2015

10. Acknowledgements

Thanks Bob Briscoe for his insightful suggestions on the basic mechanisms of congestion information collection and many other useful comments. Thanks David Black for his useful technical suggestions.

Also, thanks Anthony Chan, Jake Holland, John Kaippallimalil and Vincent Roca for their careful reviews.

Authors' Addresses

Xinpeng Wei
Beiqing Rd. Z-park No.156, Haidian District,
Beijing, 100095, P. R. China
E-mail: weixinpeng@huawei.com

Zhu Lei
Beiqing Rd. Z-park No.156, Haidian District,
Beijing, 100095, P. R. China
E-mail:lei.zhu@huawei.com

Lingli Deng
Beijing, 100095, P. R. China
E-mail: denglingli@gmail.com