Internet Engineering Task Force
INTERNET-DRAFT
Intended Status: Informational
Expires: January 4, 2015

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

July 3, 2014

Tunnel Congestion Feedback
draft-wei-tsvwg-tunnel-congestion-feedback-02

Abstract

This document describes a mechanism to calculate congestion of a tunnel segment based on RFC 6040 recommendations, and a feedback protocol by which to send the measured congestion of the tunnel from egress to ingress router. A basic model for measuring tunnel congestion and feedback is described, and a protocol for carrying the feedback data is outlined.

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 current practice of Internet protocol, encapsulation of IP headers is always the technical proposal for overlay networking scenarios. For example, mobile network are designed to encapsulate inner IP header and application layer header chain through IP header, UDP header and GTP-U header. It is also designed to fulfill the mobility, QoS control, bearer management and other specific application of the mobile network. Some organization's private network encrypt IP header by Internet tunnel solutions with private key or certification approaches to setup VPN (virtual private network) over WAN (wide area network).

Congestion is the situation that traffic input exceeds throughput of any segment of transmission path, which can result from transportation constraints and interface/processor overload. In general, congestion seen as the cause of packet loss or unexpected delay to network end points. End to end congestion protocols (e.g. ECN [RFC 3168] and ECN handling for tunneling scenario [RFC6040]) are discussed in IETF.

In IP header encapsulation cases, IP headers should be carried over transportation protocol like TCP or UDP, which influents the explicit congestion control feedback, since the receiver should mark ECN in TCP acknowledgment. On the other hand, packet loss and performance degradation should not be recognized by network elements, for instance the tunnel ingress and egress entity, when network segment is encapsulated by IP header and UDP header chain. That causes management problem when tunnel segment is considered as an independent administration domain, and network operator intents to keep network operation reliable.

This document describes a mechanism for feedback of congestion observed in IP tunnels usages. Common tunnel deployments such as mobile backhaul networks, VPNs and other IP-in-IP tunnels can be congested as a result of sustained high load.

Network providers use a number of methods to deal with high load conditions including proper network dimensioning, policies for preferential flow treatment and selective offloading among others. The mechanism proposed in this document is expected to complement them and provide congestion information that to allow making better, policies and decisions.

The model and general solution proposed in chapter 4 consist of identifying congestion marks set in the tunnel segment, and feeding back the congestion information from the egress to the ingress of the tunnel. Measuring congestion of a tunnel segment is based on counting

outer packet CE marks for packets that have ECT marks in the inner packet. This proposal depends on statistical marking of congestion and uses the method described in RFC 6040 [RFC6040], Appendix C.

Chapter 5 describes the protocol mechanisms to feed back the calculated congestion information from egress to ingress. The desired properties of the congestion information conveying protocol are outlined, and IPFIX [RFC5101] as a candidate protocol for these extensions is explored further.

2. Conventions and Terminology

2.1 Conventions

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]

2.2 Terminology

Tunnel: A channel over which encapsulated packets traverse across a network.

Encapsulation: The process of adding control information when it passes through the layered model.

Encapsulator: The tunnel endpoint function that adds an outer IP header to tunnel a packet, the encapsulator is considered as the "ingress" of the tunnel.

Decapsulator: The tunnel endpoint function that removes an outer IP header from a tunneled packet, the decapsulator is considered as the "egress" of the tunnel.

Outer header: The header added to encapsulate a tunneled packet.

Inner header: The header encapsulated by the outer header.

E2E: End to End.

VPN: Virtual Private Network is a technology for using the Internet or another intermediate network to connect computers to isolated remote computer networks that would otherwise be inaccessible.

GRE: Generic Routing Encapsulation.

IPFIX IP Flow Information Export. An IETF protocol to export flow information from routers and other devices.

RED Random Early Detection

NFV Network Functions Virtualization is an alternative design approach for building complex IT applications, particularly in the telecommunications and service provider industries, that virtualizes entire classes of function into building blocks that may be connected, or chained, together to create services.

VNF Virtualized Network Function may consist of one or more virtual machines running different software and processes, which form the building blocks for NFV.

SFC Service Function Chain is a group of connected VNF in a specific sequence/map using NFV approach, in order to deliver a specific service.

3. Problem Statement

Network traffic congestion control plays a significant role in network performance management, and sustaining congestion could impact subscriber's experience. Currently the solution of network congestion problem mainly focuses on end-to-end method, i.e. ECN [RFC3168], and the traffic sender are in charge of reducing traffic rates in case of network congested. But sometimes it's not always reliable to dependent on end hosts to solve the congestion situation, because some end hosts may not support ECN, or even ECN is supported by end hosts some traffics, e.g. UDP-based traffic, may not support ECN.

Though the congestion happens in operator's network, in case that the congestion information is transparent to operator, network administration would be hard to take action to control the network traffic of reason to network congestion. To improve the performance of the network, it's better for operator to take network congestion situation into network traffic management.

Many kinds of tunnels are widely deployed in current networks, even in some scenarios all traffics transmitted through designated tunnel(s).

Because the ingress and egress of tunnel are usually deployed by operator, so it's easy for operator to execute operator's policy, for

example gating, flow control and dropping. The tunnel feedback mechanism should be feasible for operator to collect network congestion information in encapsulation segment. After obtaining congestion information, operator could make policy at tunnel ingress for traffic management taking these information into consideration.

ECN handling mechanisms in RFC 6040 specifies how ECN should be handled for tunneling. In addition, RFC 6040, Appendix C provides guidance to calculate congestion experienced in the tunnel itself. However, there is no standardized mechanism by which the congestion information inside the tunnel can be fed back from egress to ingress router.

In the following sub-sections, some network tunnel scenarios are discussed.

3.1 3GPP network scenario

Tunnels, including GRE [RFC2784], GTP [TS29.060], IP-in-IP [RFC2003] or IPSec [RFC4301] etc, are widely deployed in 3GPP networks. And in 3GPP network tunnels are used to carry end user flows within the backhaul network such as shown in Figure 1.

IP backhaul networks such as those of mobile networks are provisioned and managed to provide the subscribed levels of end user service. These networks are traffic engineered, and have defined mechanisms for providing differentiated services and QoS per user or flow. Policy to configure per user flow attributes in these networks have traditionally been based on monitoring and static configuration.

Currently, these networks are increasingly used for applications that demand high bandwidth. The nature of the flows and length of end user sessions can lead to significant variability in aggregate bandwidth demands and latency. In such cases, it would be useful to have a more dynamic feedback of congestion information. In addition, eNB, SGW and PGW are administrated by one mobile operator, mobile backhaul to carry IP/UDP/GTP encapsulation is regally administrated by back haul service operator. This aggregate congestion feedback could be used to determine flow handling and admission control.

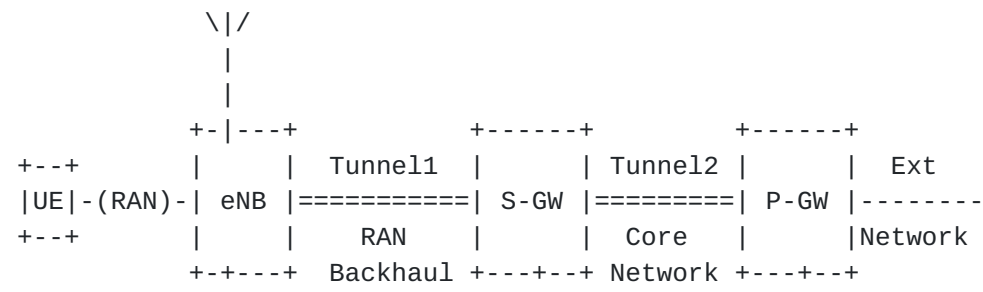


Figure 1: Example - Mobile Network and Tunnels

3.2 Network Function Virtualization Scenario

Telecoms networks contain an increasing variety of proprietary hardware appliances, leading to increasing difficulty in lauching new network services, as well as the complexity of integrating and deploying these appliances in a network.

Network Functions Virtualisation (NFV) aims to address these problems by decoupling the software from dedicated hardware platforms to a range of industry standard server hardware for various network services, through IT virtualization technology that can be moved to, or instantiated in, various locations in the network as required. In this way, it is expected to provide significant benefits for network operators (reduced expenditures for network construction and maintenance) and their customers (shortened time-to-market for new network services).

Furthermore, service functions are preferred to be deployed and managed in a data center manner, rather than being inserted on the data-forwarding path between communicating peers as today. SFC WG is currently working on a new framework to cope with this highly dynamic routing problem for a network service, which requires that the relevant data traffic be traversing a group of virtualized network function nodes (VNFs), each of which could be applied at any layer within the network protocol stack (network layer, transport layer, application layer, etc.). [SFC]

As shown in Figure 2, in a SFC-enabled domain (e.g. with or across network operator's deployed data centers), a PDP (Policy Decision Point) is the central entity which is responsible for maintaining SFC Policy Tables (rules for the boundary nodes on deciding which IP flow to traverse which service function path), and enforcing appropriate policies in SF Nodes and SFC Boundary Nodes. Beginning at the Ingress node, at each hop of a given service function path (as decided by a matched SFC policy rule/map), if the next function node is not an

immediate (L3) neighbor, packet are encapsulated and forwarded to correspondent downstream function node, as shown in Figure 3.

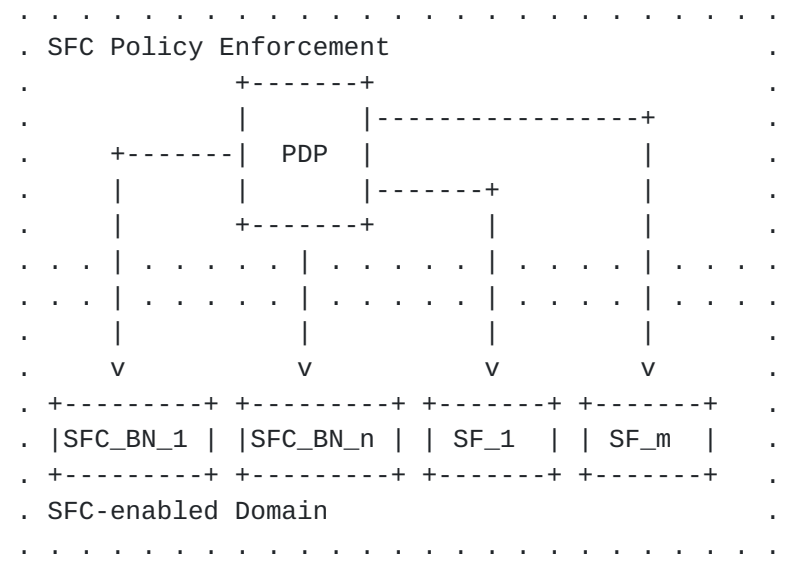


Figure 2: SFC Policy Enforcement Scheme

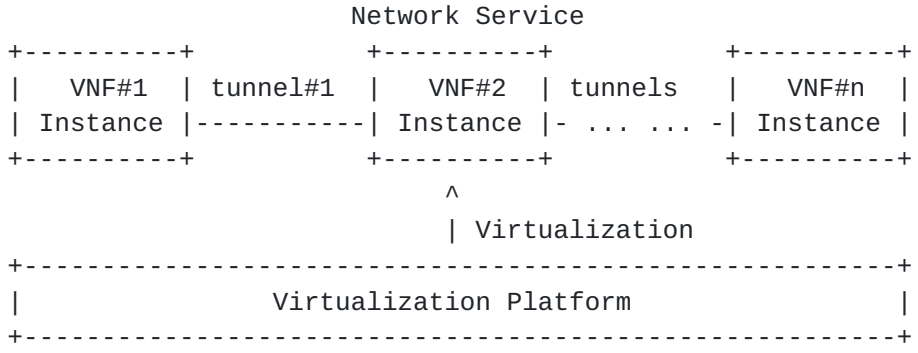


Figure 3: Example - Mobile Network service chaining and Tunnels

However, using VNFs running commodity platforms can introduce additional points of failure beyond those inherent in a single specialized server, and therefore poses additional challenges on reliability. [VNFPOOL] proposes using pooling techniques in response, which requires maintaining a backup mapping among running VNF instances for a given service function, and choosing from them for a specific data flow. It is clear that it would be helpful to make more efficient use of network capacity in case of local congestion, if the choice is based on the ECN feedback as well as the running status and/or physical resources accommodation of a candidate VNF instance.

4. Congestion Model

To support traffic management and congestion information feedback in tunnel, there are mainly two issues that this document discusses: calculation of congestion level information, and feeding back the congestion information from egress to ingress router.

In this solution, we assume the tunnel ingress/egress is compliant with RFC6040 and the tunnel interior routers are compliant with RFC3168.

In addition, it should be noted that these tunnels may carry ECT or Not-ECT traffic. A well defined mechanism for aggregate congestion calculation should be able to work in the presence of all kinds of traffic and would benefit from a common feedback mechanism and protocol.

4.1 Congestion Calculation

Calculation of congestion in the tunnel is based on the method described in RFC 6040, Appendix C.

The egress can calculate congestion using moving averages. The proportion of packets not marked in the inner header that have a CE marking in the outer header is considered to have experienced congestion in the tunnel. Note that the packets are ECN capable and not congestion-marked before tunnel. Since routers implementing RED randomly select a percentage of packets to mark, this method can be effectively used to expose congestion in the tunnel.

When the ingress is RFC6040 compliant, the packets collected by egress can be divided into to 4 categories, shown in figure 2. The tag before "|" stands for ECN field in outer header; and the tag after "|" stands for ECN field in inner header.

"Not-ECN|Not-ECN" indicates traffic that does not support ECN, for example UDP and Not-ECT marked TCP; "CE|CE" indicates ECN capable packets that have CE-mark before entering the tunnel; "CE|ECT" indicates ECN capable packets that are CE-marked in the tunnel; "ECT|ECT" indicates ECN capable packets that have not experienced congested in tunnel (or outside the tunnel).

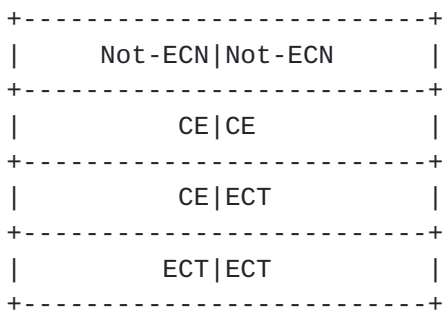


Figure 2: ECN marking categories by outer/inner packet

Out of the total number of packets, if the quantity of CE|ECT packets is A, the quantity of ECT|ECT packets is B, then the congestion level (C) can be calculated as follows:

$$C=A/(A+B)$$

As an example, consider 100 packets to calculate the moving average as shown in RFC 6040, Appendix C. Say that there are 12 packets that have CE|ECT marks indicating that they have experienced congestion in the tunnel. And, there are 58 packets that have ECT|ECT marks indicating that there was no congestion in either the tunnel or elsewhere. The egress can calculate congestions as:

$$C = 12/ (12 + 58)$$
$$= 12/70 \text{ (17\% congestion)}$$

4.2 Congestion Feedback

The figure below introduces an abstract view of the tunnel and outlines a tunnel congestion feedback model.

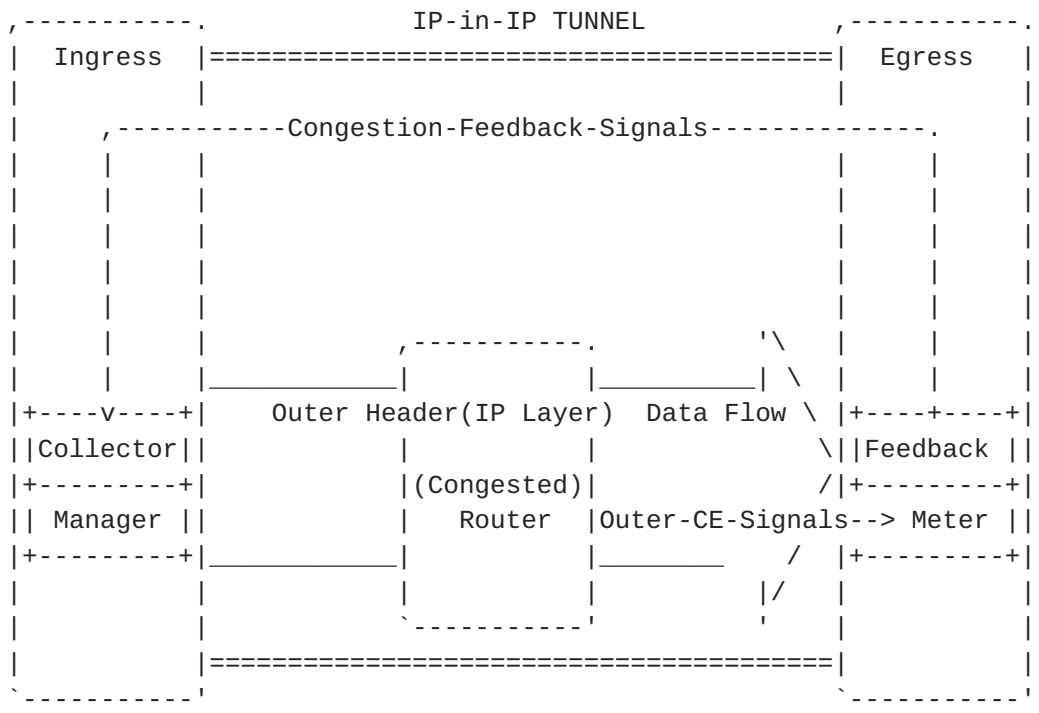


Figure 3: Basic Feedback Model

The basic model consists of the following components: Ingress, Egress, Feedback, Meter, Collector and Manager.

At egress, a module named Meter is used to estimate the congestion level of in the tunnel as described in the section above. A congestion information feedback module, called Feedback, is used to control the congestion information feedback.

The metering module (Meter) in the Egress node accounts the congestion marks it receives. The Feedback module calculates the amount of congestion and feeds back the congestion information to the Ingress node. The Collector at the Ingress receives the congestion information which is fed back from the Egress node. The Manager has admission control and flow control functions which are out of the scope of this document.

It should be assumed that the ingress and egress of the tunnel are ECN-enabled and the intermediate routers in the tunnel path are also ECN-enabled. Congestion feedback signals in the figure are fed back using protocols described in section 5.

5. Congestion Feedback Protocol

5.1 Properties of Candidate Protocol

To feedback congestion efficiently there are some properties that are desirable in the feedback protocol.

1. Congestion friendliness. The feeding back traffics are coexistence with other traffics, so when congestion happens in the network, the feeding back traffic should be reduced, So that feedback itself will not congest the network further when the network is already getting congested. In other words, feedback frequency should adjust to network's congestion level.

2. Extensibility. The authors consider that using an existing protocol, or extensions to an existing protocol is preferable. The ability of a protocol to support modular extensions to report congestion level as feedback is a key attribute of the protocol under consideration.

3. Compactness. In different situations, there may be different congestion information to be conveyed, and in order to reduce network load, the information to be conveyed should be selectable, i.e. only the required information should be possible to convey.

5.2 IPFIX Extensions for Congestion Feedback

This section outlines IPFIX extensions for feedback of congestion. The authors consider that IPFIX is a suitable protocol that is reasonably easy to extend to carry tunnel congestion reporting.

Since IPFIX is preferred to use SCTP as transport, it has the foundation for congestion-friendly behavior, and because SCTP allows partially reliable delivery [RFC3758] - IPFIX message channels can be tagged so that SCTP does not retransmit certain losses. This makes it safe during high levels of congestion in the reverse direction, to avoid a congestion collapse.. When congestion occurs in the network, the Exporter (Egress) can reduce the IPFIX traffic. Thus the feedback itself will not congest the network further when the network is already getting congested. When the Exporter detects network congestion, it can also reduce IPFIX traffic frequency to avoid more congestion in network while being able to sufficiently convey congestion status.

Because the template mechanism in IPFIX is flexible, it allows the export of only the required information. Sending only the required

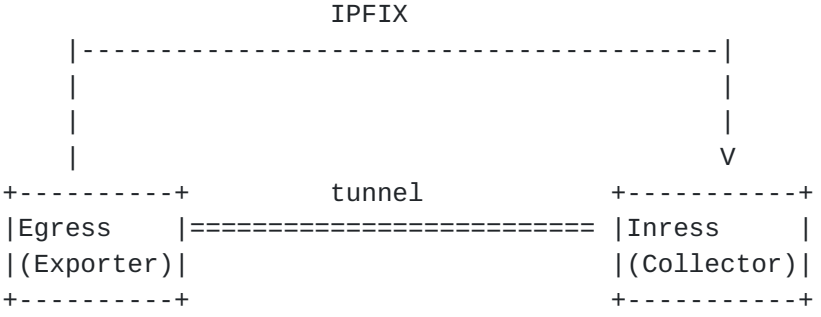
information can also reduce network load.

The basic procedure for feedback using IPFIX is as follows:

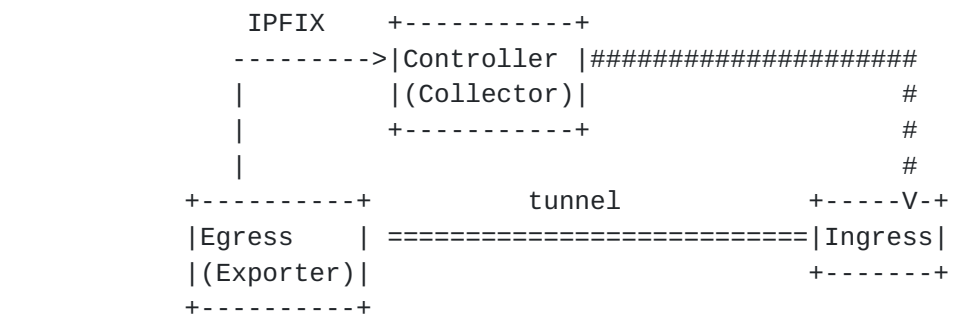
(1)The exporter inform the collector how to interpret the IEs in IPFIX message using template. Collector just accepts template passively; which IEs to send is configured by other means that not included in IPFIX specification.

(2)The exporter meters the traffic and sends the congestion level to collector.

Congestion feedback using IPFIX is shown in the figures below. There are two variations to congestion feedback model using IPFIX. In the first one shown in Figure 4(a), congestion information is sent directly from egress to ingress and ingress makes decisions according this information. In the second case shown in Figure 4(b), congestion information is sent to a mediation controller instead of tunnel ingress; the controller is in charge of making decisions according to network congestion and control the behavior of ingress node, for example, reducing traffic or forbidding new traffic flows. In this model the congestion information from egress to controller is conveyed by IPFIX, but how controller controls the behavior of ingress is out of scope of this document.



(a) Direct Feedback.



(b) Mediated Feedback.

Figure 4: IPFIX Congestion Feedback Models

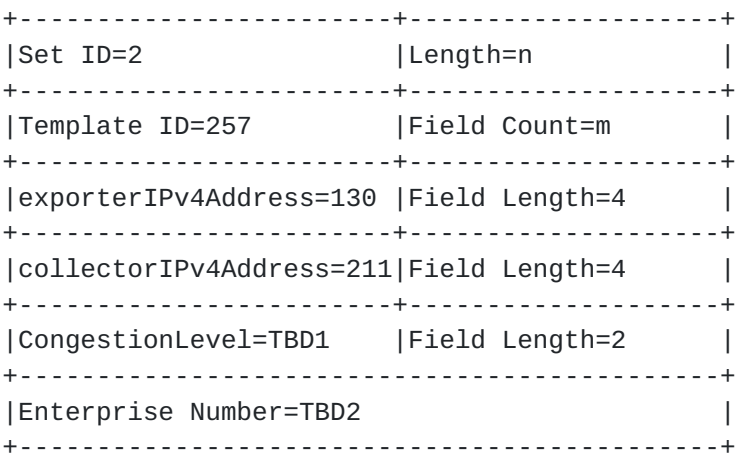
To support feeding back congestion information, some extensions to the IPFIX protocol are necessary. A new IE conveying congestion level is defined for this purpose.

Definition of new IE indicating congestion level.
Description:
The congestion level calculated by exporter.
Abstract Data Type: unsigned8
Data Type Semantics: quantity
ElementId: TBD.
Status: current

The example below shows how IPFIX can be used for congestion feedback.

[NOTE: the information conveyed here may be incomplete, and what information should be conveyed needs to be determined.]

(1) Sending Template Set The exporter use Template Set to inform the collector how to interpret the IEs in the following Data Set.



(2) Sending Data Set The exporter meters the traffic and sends the congestion information to collector by Data Set.

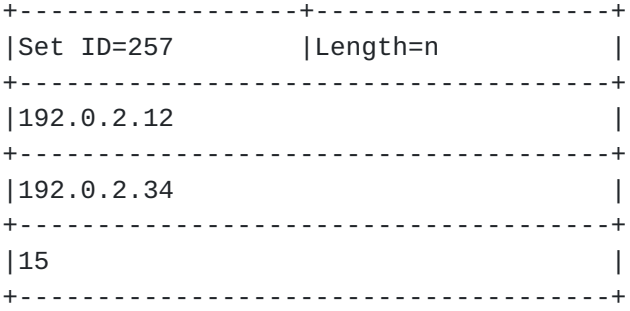


Figure 5: IPFIX Congestion Flow

The Exporter can send congestion information periodically or when triggered by the Collector. Before sending congestion information to collector, the exporter sends a Template set to Collector. The Template set specifies the structure and semantics of the subsequent Data Set containing congestion-related information. The Collector understands the Data Sets that follow according to Template Set that was sent previously. The exporting Process transmits the Template Set in advance of any Data Sets that use that Template ID, to help ensure that the Collector has the Template Record before receiving the first Data Record. Data Records that correspond to a Template Record may appear in the same and/or subsequent IPFIX Message(s).

The Exporter meters the traffic passing through it and generates flow records. At this point, the Exporter may cache the records and then send congestion cumulative information to the collector. When Exporter detects that the network is heavily congested, it can change the feedback frequency to avoid adding more congestion to network.

When receiving congestion related information, the Collector will make decisions to control the traffic entering the tunnel to reduce tunnel congestion.

5.3 Other Protocols

A thorough evaluation of other protocols have not been performed at this time.

6. Security Considerations

This document describes the tunnel congestion calculation and feedback. For feeding back congestion, security mechanisms of IPFIX are expected to be sufficient. No additional security concerns are expected.

7. IANA Considerations

IANA assignment of parameters for IPFIX extension may need to be considered in this document.

8. References

8.1 Normative References

[RFC2003] Perkins, C., "IP Encapsulation within IP", RFC 2003, October 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2784] Farinacci, D., Li, T., Hanks, S., Meyer, D., and P. Traina, "Generic Routing Encapsulation (GRE)", RFC 2784, March 2000.

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

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

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC5101] Claise, B., Ed., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.

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

[I-D.boucadair-sfc-framework] Boucadair, M. etc, "Service Function Chaining: Framework & Architecture", draft-boucadair-sfc-framework-00(work in progress), October 2013.

[I-D.zong-vnfpool-problem-statement] Zong, N. etc, "Virtualized Network Function (VNF) Pool Problem Statement", draft-zong-vnfpool-problem-statement-02(work in progress), January 2014.

8.2 Informative References

[TS29.060]3GPP TS 29.060: "General Packet Radio Service (GPRS); GPRS Tunnelling Protocol (GTP) across the Gn and Gp interface".

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