Workgroup:
Deterministic Networking Working Group
Internet-Draft:
draft-peng-detnet-traffic-shaping-solutions-00
Published: 29 September 2022
Intended Status: Informational
Expires: 2 April 2023
Authors: G. Peng
Beijing University of Posts and Telecommunications
S. Wang
Beijing University of Posts and Telecommunications
Z. Cheng
New H3C Technologies
L. Zhou
New H3C Technologies
P. Liu
China Mobile

Traffic Shaping Solutions for Bounded Latency in Large-scale Networks

Abstract

This document presents a traffic shaping solution for DetNet service with bounded latency in large-scale networks. The traffic shaping solution includes the edge access control, enqueue cycle mapping and jitter compression mechanisms. These mechanisms support appropriate resource reservation algorithms, reasonably calculate the end-to-end delay in DetNet IP network in advance, and adjust, manage and control the resources after real-time detection. Using the traffic shaping solution, it is possible for an implementer, user, or standards development organization to realize bounded delay based on the existing TSN/DetNet queuing models.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The standard documents related to deterministic networks provide bounded latency and zero congestion loss for time-sensitive services (or real-time services). e.g., IETF Deterministic Networking (DetNet) and IEEE 802.1 Time-sensitive Networking [IEEE802.1TSN]. DetNet enables these capabilities based on the following aspects [I-D.ietf-detnet-bounded-latency]: A) configuring and allocating network resources for the exclusive use of DetNet flows; B) identifying, in the data plane, the resources to be utilized by any given packet, and C) the detailed behavior of those resources, especially transmission queue selection.

In [RFC8655], DetNet flows are set with maximum bandwidth and the worst-case end-to-end transmission latency, which is usually ensured by strict input metering and forwarding policies. The bounded transmission latency of DetNet flows can provide appropriate buffer space for devices in the same network domain, further ensuring zero congestion loss for DetNet services. To meet such strictly bounded latency, DetNet flows need to ensure that their explicit routes, queue buffers, and bandwidth requirements are computable before arrival. This document refers to the relevant queuing models in TSN [IEEE802.1Qbv][IEEE802.1Qch] and DetNet [RFC8655] documents, which guarantee the Quality of Service (QoS) of DetNet flows by controlling packet forwarding and transmission on each node. In this document, a traffic shaping solution is proposed to provide edge access control, cycle mapping and jitter compression mechanisms to enhance the typical TSN/DetNet queue models, so as to support end-to-end bounded latency and jitter transmission across network domains. The above mechanisms in the traffic shaping solution are based on the DetNet timing model [I-D.ietf-detnet-bounded-latency]. This document improved the bounded latency timing model so that it could be applied to large-scale deterministic network for traffic scheduling.

Using the traffic shaping solution presented in this document, it is possible for an implementer, user, or standards development organization to realize bounded regulation delay and queuing delay based on the existing queuing models. The edge access control, enqueue cycle mapping and delay detection operations in this document support appropriate resource reservation algorithms so that the end-to-end latency in the DetNet IP network can be reasonably calculated in advance, and resources can be adjusted and managed and controlled after real-time detection.

This document does not specify any resource reservation protocol, transmission selection algorithm, and control plane function. It does describe methods for the regulation of DetNet flows with existing queuing models. Any protocol and model can be applied as long as it complies with the traffic shaping solution rules.

2. Terminology and Definitions

This document uses the terms defined in [RFC8655]. Moreover, the following terms are used in this document:

TSN
Time-Sensitive Networking.

CNC
Central Network Controller.

TAS
Time Awareness Shaper.

CQF
Cyclic Queuing and Forwarding.

TSN
Time-Sensitive Networking.

CSQF
Cycle Specified Queuing and Forwarding [I-D.qiang-detnet-large-scale-detnet].

SQ/RQ
Sending Queue and Receiving Queue.

SID
Segment Routing Identifier.

3. Bounded Latency Model for Large-scale Networks

This section presents the DetNet basic model for traffic shaping solutions in large-scale networks. We establish the flow admission paradigm of DetNet flow scheduling in large-scale networks, and propose DetNet Relay Nodes and Edge Nodes to build the end-to-end transport model, which further supports our solution of bounded delay in large-scale networks.

3.1. Flow admission

1. Describe the characteristics of the newly arrived DetNet flow, such as the worst-case end-to-end delay, jitter, bandwidth requirements, and flow sending frequency, packet number, etc.

2. The end-to-end latency model of DetNet transit nodes includes DetNet edge nodes and DetNet relay nodes. For aggregation of DetNet flows, any configuration required by DetNet relay nodes in the network can be performed. The configuration is done beforehand, and not tied to any particular DetNet flow. The configuration of DetNet edge nodes supports edge access control and cycle mapping operation.

3. The cooperative work of DetNet edge nodes and DetNet relay nodes supports the traffic shaping solution for DetNet flows (time-sensitive traffic) across domains in a large-scale network.

4. Establish the explicit route that the DetNet flow will take through the network from the source to the destination(s). This can be a point-to-point or a point-to-multipoint path.

5. Performs the cross-domain end-to-end transmission of DetNet flows over large-scale networks. The traffic shaping solution can realize the cross-domain end-to-end explicit route transmission after DetNet flows are injected into the network domain. In this process, delay detection is used to calibrate the jitter compressible range of DetNet flows to ensure the bounded latency and jitter requirements.

6. Assuming that the resources are available, commit those resources to the DetNet flow. This may require dynamic adjustment of control filtering rules or enqueue cycle mapping parameters at each hop along the explicit route.

This paradigm can implement unified management and control based on Centralized User Configuration (CUC)/ Centralized Network Configuration (CNC) node's requirements for collecting flow characteristics and sending DetNet relay/edge node configurations.

3.2. Relay node model and edge node model

A relay node model for the operation of a DetNet transit node is detailed in [I-D.ietf-detnet-bounded-latency]. The per-hop delay experienced by a packet passing through a DetNet transit node is decomposed into six types of delays: 1) output delay; 2) link delay; 3) frame preemption delay; 4) processing delay; 5) regulation delay; 6) queuing delay. This decomposition applies to the calculation of hop-by-hop delay and hop-by-hop buffer requirements.

An edge node model makes some changes based on the existing DetNet relay node model, adding additional buffers before entering the regulator to dynamically adjust the interdomain timeslot (cycle) offset and absorb additional jitter at the network edge. The regulation delay included in processing delay is the extra time slot offset to be mapped plus the delay of node forwarding operation. The regulation delay contained in per-hop delay introduces an additional timeslot offset for traffic shaping.

```
   DetNet transit node A            DetNet transit node B
   +----------------------+     +-------------------------------+
   |            Queuing   |     |                      Queuing  |
   | Regulator subsystem  |     |  Buffer   Regulator subsystem |
   |  +-+-+-+   +-+-+-+   |     |  +-+-+-+   +-+-+-+   +-+-+-+  |
-->+  | | | |   | | | |   +---->+  | | | |   | | | |   | | | |  +-->
   |  +-+-+-+   +-+-+-+   |     |  +-+-+-+   +-+-+-+   +-+-+-+  |
   |  DetNet relay node   |     |       DetNet edge node        |
   +----------------------+     +-------------------------------+
     4     5       6     1  2,3   4         5           6     1
    <-> <----> <------> <-> <--> <-> <-------------> <-----> <->

             1: Output delay             4: Processing delay
             2: Link delay               5: Regulation delay
             3: Frame preemption delay   6: Queuing delay
```

Figure 1: Relay node and edge node models for DetNet transit nodes

In Figure 1, the two DetNet nodes are connected via a link. Transit nodes A and B represent the DetNet relay node and the DetNet edge node respectively. In each transit node, a packet experiences six delays from hop to hop. Among them, link propagation, receiving processing, frame preemption and output delay are affected by hardware, Precise Time Protocol ([IEEE8023] [RFC8655]) and other factors, but are relatively a constant value. So, in order to obtain hop-by-hop bounded delay, the key of traffic shaping solution is to get the regulation delay and queuing delay bounds. The edge access control, enqueue cycle mapping and delay detection operations are proposed to adjust these two kinds of delay in DetNet transit node models in Section 4 and Section 5 .

3.3. End-to-end transmission model for large-scale networks

In Figure 2, the end-to-end transmission model consists of TSN end systems, TSN domains, DetNet relay nodes and DetNet edge nodes. Because in large-scale networks, DetNet service flows need to be transmitted across multiple network domains, new requirements are put forward for DetNet nodes to deal with transmission delay of network edge and interdomain communication [I-D.liu-detnet-large-scale-requirements]. The edge nodes in this model can perform edge access control when flows are injected into the DetNet domain, and perform timeslot offset after flows injecting entering the edge node. When leaving the DetNet domain, bounded delay and jitter are controlled by jitter compression scheme at DetNet edge nodes. The whole deterministic communication in a large-scale network includes: TSN end system access -> TSN network domain -> DetNet edge node -> DetNet relay node -> ... -> DetNet edge node -> peer TSN network domain -> TSN end system.

```
                 DetNet service flows
        <--------------------------------------->

  TSN       DetNet     DetNet        DetNet      TSN
end system edge node  relay node    edge node  end system
+------+   +------+   +------+      +------+   +------+
|      |   |     +<-->+     +<-...->+      |   |      |
+--+--+   +--+--+   +------+      +--+--+   +--+--+
   ^          ^                        ^          ^
   |  +----+  |                        |  +----+  |
   +->+    +<-+                        +->+    +<-+
      +----+                              +----+
     TSN domain                        TSN domain
```

Figure 2: End-to-end transmission model in large-scale networks

4. Traffic Shaping Mechanisms

For the cross-domain traffic scheduling in large-scale networks, this document presents a traffic shaping mechanism between network domains (e.g., TSN domain, DetNet domain) for edge access control and management of DetNet flows. The traffic shaping mechanism establishes cross-domain cycle mapping relationship between different network domains according to the requirements of upper-layer application latency and jitter, and supports deterministic queuing model of different domains (such as CQF and CSQF mechanisms).

In this document the traffic shaping mechanism solves problems such as inter/intra-domain multi-flow aggregation, traffic burst, uncertain enqueue selection, and bandwidth resource mismatch. Based on the cross-domain one-to-one deterministic cycle mapping relationship, end-to-end DetNet flows are scheduled within each domain, and bounded latency guarantee is realized by inter-domain cooperation.

4.1. Traffic shaping at the network edge

The idea of network edge traffic shaping mechanism is to plan the injection time of time-sensitive traffic at the network edge to achieve edge access control. As each packet enters a new network domain, it gains a timeslot offset in the buffer. With such mechanism, we can centrally plan and manage the timeslot offset of time-sensitive traffic based on the state information of interdomain network and characteristic parameters of DetNet flow. For example, In Figure 3, When end systems send/receive time-sensitive application traffic to a network domain (TSN/DetNet), we leverage the additional edge buffer to adjust the injection timeslot offset of traffic entering the queue model (e.g., CQF). Based on network information and traffic characteristics, we can dynamically adjust the timeslot offset when cross-domain traffic enters different domains, plan queue resources in advance, and alleviate inter-domain flow aggregation and burst.

The traffic shaping at the network process is as follows:

1. CNC discovers and connects terminal/network devices through API interface. Obtain the information of network topology, link capacity, and port transmission rate in the current network domain. Obtain time-sensitive traffic information injected into the network domain, including end-to-end delay bounds requirements, packet sending frequency, packet size and quantity, and source/destination address.

2. CNC performs DetNet flow scheduling for the traffic in this domain and obtains the traffic output timeslot from the initial TSN domain to the DetNet domain. The specific flow scheduling algorithm is not restricted in this traffic shaping solution.

3. CNC plans the injection timeslot offset of time-sensitive traffic injected into the network domain according to the existing timeslot conflict situation, and comprehensively considers the link capacity, end-to-end delay bounds requirements and other factors to adjust the injection timeslot of some packets.

4. Based on the edge access control of time-sensitive flow, CNC reschedules the flow offset by injection time to reduce the aggregation and burst of some timeslot flows transmitted between domains.

```
+-------------------------------------------+
|                  CNC/CUC                  |
+----+----------------+----------------+----+
     |                |API             |
     v                v                v
End systems      TSN domain      DetNet domain
+---------+     +----------+     +----------+
|         +<--->+          +<--->+          |
+---------+     +----------+     +----------+
             ^      Edge      ^
             | access control |
+------------+----------------+-------------+
|                         +-----------+     |
|          Edge buffer +->     RQ     |     |
| packets   +-------+  |  +-----------+     |
| +------>  |       +--+  +-----------+     |
|           +-------+     |    SQ     ->    |
|                         +-----------+     |
|    Timeslot offset    Cyclic forwarding   |
+-------------------------------------------+
```

Figure 3: Traffic shaping at the network edge: edge access control

4.2. Inter/Intra-domain traffic shaping

This section proposes a one-to-one deterministic cycle mapping relationship for inter/intra-domain traffic shaping. After edge access control management of cross-domain traffic by offsetting the injection timeslot at the network edge, we establish a enqueue cycle mapping relationship between cross-domain traffic from TSN domain to DetNet domain (or DetNet domain to TSN domain). When there is no serious inter-domain multi-flow aggregation and burst phenomenon between domains, this mechanism needs to obtain the parameter of the queue models (e.g., CQF and CSQF model) between adjacent domain, including queue number, cycle time slot size, output port bandwidth, transmission delay, clock synchronization of fixed frequency offset parameters, and then establishes a cross-domain one-to-one enqueue cycle mapping relationship. The cycle mapping relationship is defined as follows: if packets sent in cycle X in a node A will all be received no later than cycle Y in the downstream node B. It can be expressed by the formula:

Cycle_mapping(A,B)(X)=Y

In Figure 4, after the clock synchronization of edge devices, the controller can obtain the fixed clock frequency difference between devices, and then establish a one-to-one cycle mapping relationship: from the sending queue in TSN domain to the next hop receiving queue in DetNet domain. The mapping information is added to the packet's

Segment Routing Identifier (SID) tag for enqueue selection after packets exiting the edge buffer. Because the controller can plan the inter-domain enqueue selection in advance, it can ensure that the upper and lower bounds of regulation delay and queuing delay of cross-domain traffic are deterministic.

```
     TSN domain          DetNet domain
+-----------------+    +------------------+
|    TSN devices  |    | DetNet devices   |
|  +----+  +----+ |    | +----+   +----+  |
|  |    +--+    +------->+    +---+    |  |
|  +----+  +----+ |    | +----+   +----+  |
|                 | ^^ |                  |
+-----------------+ || +------------------+
                    ||
+-------------------++--------------------+
|                          +---------+    |
| The last hop   Edge          RQ    |    |
|sending queue  Buffer     +---------+    |
|   +-------+   +----+     +---------+    |
|   |  SQ    -> |    +--->     RQ    |    |
|   +-------+   +----+     +---------+    |
|                             ....        |
|  SID specifies the next  +---------+    |
|  hop receiving queue     |   SQ     --> |
|                          +---------+    |
+-----------------------------------------+
  1. Edge devices clock synchronization.
  2. Enqueue cycle mapping relationship.
```

Figure 4: Inter/Intra-domain traffic shaping: cycle mapping

5. Jitter Compression for Large-scale Networks

A large-scale network may span multiple networks, and one of the goals of DetNet is to connect each network domain to provide end-to-end deterministic delay service. The adoption techniques and capabilities of each network are different, and the corresponding topology models are either piecewise or nested. In this way, mutual coupling (dependency) should be reduced as much as possible. As long as the network meets certain range requirements, the jitter compression of the two-end device with Asynchronous/synchronous clocks can support end-to-end deterministic delay service.

In this document, the jitter compression scheme is compatible with the edge access control and enqueue cycle mapping mechanisms (Section 4.1 and Section 4.2). The jitter compression utilizes the explicit route planning, delay detection, jitter compression mechanisms to support end-to-end time-sensitive traffic scheduling across multiple domains to ensure bounded and jitter DetNet service requirements.

```
                        +-----------------+
                TS1 XXX>+ SDN controller  +<XXXX TS7
                    X   +-----------------+    X   Control plane
+-------------------X--------------------------X------------------+
        Gateway A   X                          X     Data plane
   +--+   +---+   +---+   +---+   +---+        X
   |  +-->+   +-->+ R1+-->+ R2+---+ R3+------+ X
   +--+   +---+   +-+-+   +-+-+   +-+-+      | X   Gateway B
  Host A        ^   |       |       |       ++--+    +---+   +--+
                |   |       |       |       | R7+--->+   +-->+  |
       Add SIDs +   |       v       |       ++--+    +---+   +--+
                  +-+-+   +-+-+   +-+-+      ^    +         Host B
                  | R4+---+ R5+-->+ R6+------+    |
                  +---+   +---+   +---+           v Remove SIDs

      Explicit route: R1->R2->R5->R6->R7   Data packets: ----->
      Relay nodes: R2, R3, R4, R5, R6    INT/NQA packets:  XXXX>
      Edge nodes: R1, R7
```

Figure 5: Explicit route planning and delay detection for jitter compression scheme

5.1. Explicit route planning

In this document, explicit routing planning adopts the new SID type for DetNet transit nodes (e.g., edge nodes, relay nodes). The SIDs contain information about the output port interface, queue (e.g., receiving queue, sending queue), and control gate period. Relay nodes and edge nodes interact with each other through protocols to learn mapping of gating cycles. SIDs can be configured by the SDN controller or generated on the device side and reported to the SDN controller.

In Figure 5, the SDN controller uses BGP-LS to collect topology information of the entire network, and uses the detection technology to collect end-to-end SLA network service quality information, including latency, packet loss, and jitter, between edge devices (R1 and R7). Based on the quality constraint requirements of different SLA levels, the SDN controller can generate feasible explicit routes (e.g., R1->R2->R5->R6->R7) that can meet the SLA requirements and sends the corresponding SID stack to the edge nodes (R1). This document is not limited to the specific techniques used to generate SIDs.

5.2. Delay detection

In this document, network telemetry technologies such as INT/NQA can be used to detect end-to-end delay on the network. INT-Band telemetry packets are encapsulated by inserting the INT header into the TCP or UDP header of the original packet, such as INT over TCP, INT over UDP. As shown in Figure 6, INT probe HDR is the inherent header of INT, and the device identifies INT packets through this field. MD #1-N is meta-data, and Timestamp (TS) is the INT information added at the end of the packet. Each TS includes an ingress timestamp and an egress timestamp.

```
UDP/TCP Packets
+------+-------+-----+----+---+----+--------+----+---+-----
|ETH/IP|UDP/TCP| INT |MD N|...|MD 1|Playload|TS 1|...|TS M|
| HDR  |  HDR  | HDR |    |   |    |        |    |   |    |
+------+-------+-----+----+---+----+--------+----+---+----+
               |<----------------->|        |<----------->|
                 INT Encapsulation         INT encapsulation
```

Figure 6: INT-Band telemetry packet encapsulation

In Figure 5, the source node (R1) periodically sends NQA packets to the destination node (R7). After receiving the probe packets, the destination device replies the packets. The source node calculates the packet delay based on the time of receiving and replying packets and reports the packet delay to the controller. If the network scale is large, end-to-end detection causes heavy pressure on the device. Alternatively, the device can only perform detection between neighboring devices and report the detection to the controller. The controller collects information and calculates the end-to-end delay. The maximum and minimum end-to-end delay values are calculated as follows:

End-to-End min = min (TS7 - TS1)

End-to-End max = max (TS7 - TS1)

5.3. Jitter compression

Since the end side network is carried by the carrier's network, only the carrier's network promises its end-to-end delay, jitter and reliability capabilities for deterministic flows. In this document, the terminals can use the carrier's network as a tunnel, deploy the gateway on the end side to perform edge access control, traffic shaping, and deterministic scheduling, and perform jitter compression on the peer end side to meet the end-to-end bounded latency service.

This document implements global control and jitter compression based on end-to-end deterministic transmission in SDN management. The specific process is as follows:

1. Hosts A and B are located at the two ends of the network in Figure 5. Each end uses its own clock. To prevent clock drift, the SDN controller needs to calibrate the time slots at both ends. End-to-end deterministic transmission is required between hosts A and B to ensure bounded low latency and small jitter. It may span a WAN or multiple DetNet transit nodes. R1 and R7, as DetNet transit nodes, are key nodes of end-to-end deterministic transmission.

2. The SDN controller implements end-to-end viewing, explicit routes planning, and bandwidth reservation through segment routing technology (e.g., SRv6).

3. Cycle mapping is performed based on the specified SID tag to specify the jitter range of data packets in a receiving queue. R1 and R7 are DetNet transit nodes with the same scheduling frequency synchronization clock. The scheduler divides many cycles according to the same frequency and adopts the DetNet queue model (e.g., CSQF) of a specified cycle to schedule forwarding.

For example, if A new DetNet service needs deterministic transmission between hosts A and B, A request is sending to the SDN Controller via API. Based on the detection and analysis in advance, the SDN controller plans the corresponding explicit routes and distributes SIDs mapping rules to the transit nodes along the path. The edge node encapsulates the packets according to the rules and forwards the packets through reassembly, caching, and scheduling, thus realizing the end-to-end deterministic transmission with bounded latency and jitter. In SIDs, the FlowID can be used for reassembly and out-of-order recovery on the peer end side. According to Cycle, enqueue cycle mapping scheduling can be carried out at the peer end side. When uneven cycle mapping occurs on the peer device, the controller can adjust the arrival time of DetNet flows so that the flows can be mapped to different cycles. Thus, some queues won't be completely filled and some queues won't starve to death. In this way, it is possible to realize bounded latency and jitter for end-to-end communication in large-scale networks.

6. Security Considerations

This section will be described later.

7. IANA Considerations

This document has no IANA actions.

8. Acknowledgements

9. Contributors

[RFC7322] limits the number of authors listed on the front page to a maximum of 5. The editor wishes to thank and acknowledge the following author for contributing text to this document.

Tao Huang
Beijing University of Posts and Telecommunications
100876
Email: htao@bupt.edu.cn

Yunjie Liu
Beijing University of Posts and Telecommunications
100876
Email: liuyj@bupt.edu.cn

Wei Wang
New H3C Technologies
100094
Email: david_wang@h3c.com

10. Normative References

[I-D.ietf-detnet-bounded-latency] Finn, N., Boudec, J. L., Mohammadpour, E., Zhang, J., and B. Varga, "DetNet Bounded Latency", Work in Progress, Internet-Draft, draft-ietf-detnet-bounded-latency-10, 8 April 2022, <https://www.ietf.org/archive/id/draft-ietf-detnet-bounded-latency-10.txt>.

[I-D.liu-detnet-large-scale-requirements] Liu, P., Li, Y., Eckert, T., Xiong, Q., Ryoo, J., Zhu, S., and X. Geng, "Requirements for Large-Scale Deterministic Networks", Work in Progress, Internet-Draft, draft-liu-detnet-large-scale-requirements-04, 6 September 2022, <https://www.ietf.org/archive/id/draft-liu-detnet-large-scale-requirements-04.txt>.

[I-D.qiang-detnet-large-scale-detnet] Qiang, L., Geng, X., Liu, B., Eckert, T., Geng, L., and G. Li, "Large-Scale Deterministic IP Network", Work in Progress, Internet-Draft, draft-qiang-detnet-large-scale-detnet-05, 2 September 2019, <https://www.ietf.org/archive/id/draft-qiang-detnet-large-scale-detnet-05.txt>.

[IEEE802.1Qbv] IEEE, "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks - Amendment 25: Enhancements for Scheduled Traffic", IEEE 802.1Qbv-2015, DOI 10.1109/IEEESTD.2016.8613095, 18 March 2016, <https://doi.org/10.1109/IEEESTD.2016.8613095>.

[IEEE802.1Qch] IEEE, "IEEE Standard for Local and metropolitan area networks -- Bridges and Bridged Networks - Amendment 29: Cyclic Queuing and Forwarding", IEEE 802.1Qch-2017, DOI 10.1109/IEEESTD.2017.7961303, 28 June 2017, <https://doi.org/10.1109/IEEESTD.2017.7961303>.

[IEEE802.1TSN] IEEE Standards Association, "IEEE 802.1 Time-Sensitive Networking Task Group", <https://www.ieee802.org/1/pages/tsn.html>.

[IEEE8023] IEEE 802.3, "IEEE Std 802.3-2018: IEEE Standard for Ethernet", 2018, <http://ieeexplore.ieee.org/document/8457469>.

[RFC7322] Flanagan, H. and S. Ginoza, "RFC Style Guide", RFC 7322, DOI 10.17487/RFC7322, September 2014, <https://www.rfc-editor.org/info/rfc7322>.

[RFC8655] Finn, N., Thubert, P., Varga, B., and J. Farkas, "Deterministic Networking Architecture", RFC 8655, DOI 10.17487/RFC8655, October 2019, <https://www.rfc-editor.org/info/rfc8655>.

Authors' Addresses

Guoyu Peng
Beijing University of Posts and Telecommunications
Beijing
100876
China

Email: guoyupeng@bupt.edu.com

Shou Wang
Beijing University of Posts and Telecommunications
Beijing
100876
China

Email: shuowang@bupt.edu.com

Zuopin Cheng
New H3C Technologies
Beijing
100094
China

Email: czp@h3c.com

Lei Zhou
New H3C Technologies
Beijing
100094
China

Email: zhou.leih@h3c.com

Peng Liu
China Mobile
Beijing
100053
China

Email: liupengyjy@chinamobile.com