Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 9, 2020

Z. Li
S. Peng
Huawei Technologies
C. Xie
C. Li
China Telecom

July 8, 2019

Application-aware IPv6 Networking
draft-li-6man-app-aware-ipv6-network-00

Abstract

A multitude of applications are carried over the network, which have varying needs for network bandwidth, latency, jitter, and packet loss, etc. Some applications such as online gaming and live video streaming have very demanding network requirements thereof require special treatments in the network. However, in current networks, the network and applications are decoupled, that is, the network is not aware of the applications' requirements in a finer granularity. Therefore, it is difficult to provide truly fine-granular traffic operations for the applications and guarantee their SLA requirements.

This document proposes a new framework, named Application-aware IPv6 Networking, and also the solution to guarantee SLA for applications, which makes use of IPv6 extensions header in order to convey the application related information including its requirements along with the packet to the network so to facilitate the service deployment and network resources adjustment. Then, it defines the application-aware options which can be used in the different IPv6 extension headers for the purpose.

Requirements Language

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].

This Internet-Draft will expire on January 9, 2020.

Copyright Notice

Table of Contents

1. Introduction

A multitude of applications are carried over the network, which have varying needs for network bandwidth, latency, jitter, and packet loss, etc. Some applications such as online gaming and live video streaming have very demanding network requirements thereof require special treatments in the network. However, in current networks, the network and applications are decoupled, that is, the network is not aware of the applications' requirements in a finer granularity. Therefore, it is difficult to provide truly fine-granular traffic operations for the applications and guarantee their SLA requirements. Such guarantee would also be provided by adopting the hierarchical orchestration and the interactions between the orchestrator and multiple controllers, but it would take a very long time by going through the control and management elements. Moreover, the interfaces between those elements require standardizations.

This document proposes a new framework, named Application-aware IPv6 Networking, and also the solution to guarantee SLA for applications, which makes use of IPv6 extensions header (i.e. Hop-by-Hop Options Header (HBH), Destination Options Header (DOH), Segment Routing Header(SRH)) to convey the applications related information including their identifiers and requirements along with the packet to the network to facilitate the service deployment and network resource adjustment. Then it defines the application-aware options (i.e. application-aware ID option and service-aware para option), which can be used in the above listed different IPv6 extension headers for the purpose.

2. Terminologies

ASBR: Autonomous System Boundary Router

ASG: Aggregation Service Gateway

CPE: Customer-Premises Equipment

CSG: Cell Site Gateway

FBB: Fixed Broadband

MBB: Mobile Broadband

RG: Residential Gateway

RSG: Radio Service Gateway

3. Demanding Applications

This section shows the various demanding requirements of some applications. The traffic of these applications needs to be differentiated from other types of traffic and applied with special treatments in the network, that is, the network needs to be able to provide fine-granular traffic operations and acceleration to these demanding applications. In return, the operators will get their networks' revenue increased.

3.1. Online Gaming

Good network performance is normally a prerequisite for satisfactory game play, especially for the online gaming. Shooting or racing online gaming is normally based on quick action and needs to update the game status in real time by continuously sending and receiving updates to/from the game server and/or other players. The online gaming is very sensitive to the network latency.

3.2. Video streaming

The network latency, jitter, bandwidth, and packet loss are the key factors for the video streaming. Live video streaming has even more strict requirements. High quality video source require more bandwidth in order to stream properly. Real time streaming services also require real time content delivery from the web server to the end user ideally via carefully planned explicit TE paths. The online gaming often involves live video streaming.

4. Problem Statement

[RFC3272] reviews a number of IETF activities which are primarily intended to evolve the IP architecture to support new service definitions which allow preferential or differentiated treatment to be accorded to certain types of traffic. The challenge when using traditional ways to guarantee SLA is that the packets are not able to carry enough information of service requirements of applications. The network devices mainly rely on the 5-tuple of the packets which cannot provide fine-grained service process. If more information is needed, it has to refer to DPI which will introduce more cost in the network and impose security challenges.

In the era of SDN the orchestrator is introduced for the orchestration of applications and the network. The SDN controller can be aware of the service requirements of the applications on the network through the interface interworking with the orchestrator. The service requirements is used by the controller for traffic management. The method raises the following problems: 1) The whole

loop is long and time-consuming which is not suitable for the real-time adjustment for applications; 2) Too many interfaces are involved in the loop which proposes more challenges of standardization and inter-operability, and it is difficult to be standardized for easy interworking.

5. App-aware IPv6 Networking Framework

```
 Client                                                        Server
 +------+                                                      +------+
 |App x|-\                                                  /->|App x|
 +------+ |   +-----+  +---------+   +----------+   +----------+ |  +------+
          \->|App- |  |App-     |-A-|App-      |-A-|App-      |-/
 User side   |aware|--|aware    |-B-|aware     |-B-|aware     |
          /->|Edge |  |Head-End|-C-|Mid-Point|-C-|End-Point|-\
 +------+ |   +-----+  +---------+   +----------+   +----------+ |  +------+
 |App y|-/                                                  \->|App y|
 +------+           ----------  Uplink   ---------->            +------+
```

Figure 1 App-aware IPv6 Network

In the application-aware IPv6 network shown in Figure 1, there are following components:

1. Application-aware Apps: The IPv6 enabled applications runs in the host which can optionally add the service requirements of the applications in an IPv6 extension header ([RFC8200]) or remove it from the IPv6 extension header. The service requirement information includes:

o An IPv6 application-aware ID which identifies the IPv6 packets as part of the traffic flow belonging to a specific SLA level/Application/User;

o A set of parameters for the specific service such as bandwidth, delay, delay variation, packet loss ratio, etc.

The service requirements will be processed by the IPv6 enabled nodes along the path or the SRv6 ([I-D.filsfils-spring-srv6-network-programming]) nodes along the SRv6 path.

2. App-aware Edge Device: The Edge Device can add the service requirements of the applications on network through the IPv6 extension header on behalf of the IPv6 enabled applications or change the service requirements conveyed by the packets of the application-aware applications according to local policies which is out of the scope of this document. The service requirements will be processed

by the IPv6 enabled nodes along the path or the SRv6 enabled node along the SRv6 path which be programmed by the Edge Device. The application information can also be encapsulated through L2 encapsulation or some tunnel encapsulations appended to the packet depending on different application scenarios and device capability.

3. App-aware-process Head-End: The service requirements may be processed as a service path such as a SRv6 Policy path of SFC at the App-aware-process Head-End. The service requirements conveyed in the IPv6 packets can be mapped to an existing service path or an existing Policy which satisfies the specific requirement, trigger to set up the new service path by the Head-End, or trigger the global traffic adjustment by the controller according to the information provided by the network devices. The process depends on the local policy which is out of the scope this document. The application information conveyed by the packet received from the App-aware Edge Device can also be copied or be mapped to the out IPv6 extension header for further application-aware process.

4. App-aware-process Mid-Point: The Mid-Point provides the path service according to the service path set up by the Head-End which satisfies the service requirement conveyed by the IPv6 packets. The Mid-Point may also adjust the resource locally to guarantee the service requirements depending on a specific policy and the applicaton-aware information conveyed by the packet. Policy definitions and mechanisms are out of the scope of this document.

5. App-aware-process End-Point: The process of the specific service path will end at the End-Point. The service requirements information can be removed at the End-Point or go on to be conveyed with the IPv6 packets.

In this way the network is able to be aware of the service requirements expressed by the applications explicitly. According to the service requirement information carried in the IPv6 packets the network is able to adjust its resources fast in order to satisfy the service requirement of applications. The flow-driven method also reduces the challenges of inter-operability and long control loop.

6. Application-aware Options

In order to support the Application-aware IPv6 networking, two application-aware options are defined:

- o Application-aware ID option

- o Service-Para Option

6.1. Application-aware ID Option

The Application-aware ID option indicates the information of the applications, users, and application requirements, as illustrated in the following figure:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
                                +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                |  Option Type  |  Opt Data Len |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+                    IPv6 Application-aware ID                  +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3. IPv6 Application-aware ID Option

Option Type: TBD_1

Opt Data Len: 16 octets.

The IPv6 Application-aware ID is 128bits long which has one of the following structures:

-- Structure I: Any combination of SLA level (e.g. Gold, Silver, Bronze), APP ID, and/or user ID. The length of each field is variable, as shown in the following diagram:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    SLA Level    |     APP ID     |    User ID     |    Flow ID    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 4. IPv6 Application-aware ID Structure I

-- Structure II: Any combination of SLA level (e.g. Gold, Silver, Bronze), APP ID, and/or user ID plus the arguments which indicating the service requirements of the identified application, as shown in the following diagram:

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| SLA Level|  APP ID  |  User ID  |    Flow ID    |   Arguments    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 5. IPv6 Application-aware ID Structure II

-- Structure III: An SRv6 SID, with its arguments as the information specified in Structure II, as shown in the following diagram:

```
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |      Locator Address      |   Function ID |     Arguments     |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 6. IPv6 Application-aware ID Structure III

6.2. Service-Para Option

The Service-Para Option is a variable-length option carrying multiple service requirement parameters for specific application. Each service requirement parameter is put into the corresponding Service-Para Sub-TLV, as shown in Figure 6. This Option can be put into the IPv6 Hop-by-Hop Options, Destination Options, and SRH TLV.

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
                                   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
                                   |  Option Type  |  Opt Data Len |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   .                                                               .
   .                 Service-Para Sub-TLVs(Variable)               .
   .                                                               .
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 7. IPv6 Service-Para Option

Option Type	TBD
Opt Data Len	8-bit unsigned integer. Length of the Service-Para Sub-TLVs.
Service-Para Sub-TLVs	Variable-length field with Service-Para Sub-TLVs.

The corresponding Service-Para Sub-TLVs are shown in the following figures respectively.

1. BW Sub-TLV

This BW sub-TLV indicates the bandwidth requirement of applications. The format of this sub-TLV is shown in the following diagram:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |     Length    |  Class Type   |    RESERVED   |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |                           Bandwidth                           |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 8. BW Sub-TLV

where:

Type: TBD

Length: 4

Class Type: The Bandwidth Type.

RESERVED: This field is reserved for future use. It MUST be set to 0 when sent and MUST be ignored when received.

Bandwidth: This field carries the bandwidth requirement along the path.

2. Delay Sub-TLV

This Delay Sub-TLV indicates the delay requirement of applications. The format of this sub-TLV is shown in the following diagram:

```
     0                   1                   2                   3
     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |     Type      |     Length    |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    |    RESERVED   |                  Delay                        |
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 9. Delay Sub-TLV

where:

Type: TBD

Length: 4

RESERVED: This field is reserved for future use. It MUST be set to 0 when sent and MUST be ignored when received.

Delay: This 24-bit field carries the delay requirements in microseconds, encoded as an integer value. When set to the maximum value 16,777,215 (16.777215 sec), then the delay is at least that value and may be larger. This value is the highest delay that can be tolerated.

3. Delay Variation Sub-TLV

This Delay Variation Sub-TLV indicates the delay variation requirement of applications. The format of this sub-TLV is shown in the following diagram:

```
    0                   1                   2                   3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |   Type        |     Length    |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  RESERVED     |               Delay Variation                 |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 10. Delay Variation Sub-TLV

where:

Type: TBD

Length: 4

RESERVED: This field is reserved for future use. It MUST be set to 0 when sent and MUST be ignored when received.

Delay Variation: This 24-bit field carries the delay variation requirements in microseconds, encoded as an integer value.

4. Packet Loss Ratio Sub-TLV

This Packet Loss Ratio Sub-TLV indicates the packet loss ratio requirement of applications. The format of this sub-TLV is shown in the following diagram:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Type        |     Length    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    RESERVED   |                    Packet Loss Ratio          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 11. Packet Loss Ratio Sub-TLV

where:

Type: TBD

Length: 4

RESERVED: This field is reserved for future use. It MUST be set to 0 when sent and MUST be ignored when received.

Link Loss: This 24-bit field carries link packet loss ratio requirement. This value is the highest packet-loss ratio that can be tolerated.

7. Locations for placing the Application-aware Options

The Application-aware options can be placed in several locations in the IPv6 packet header depend upon the scenarios and implementation requirements.

7.1. Hop-by-Hop Options Header (HBH)

The application-aware options can be carried in the Hop-by-Hop Options Header as new options. By using the HBH Options Header, the information carried can be read by every node along the path. However, the current processing of the HBH Options Header goes to the slow path, which will decrease the forwarding performance. Therefore, we propose a new enhanced HBH Options Header in [I-D.li-6man-enhanced-extension-header].

7.2. Destination Options Header (DOH)

The application-aware options can be carried in the Destination Options Header as new options.

7.3. Segment Routing Header (SRH)

The Application-aware options can be placed in the segment routing header (SRH), e.g., in the SRH TLV, the SID Arguments field, or the TAG field.

7.3.1. SRH TLV

The Application-aware options can be placed in the SRH TLV.

7.3.2. SID Arguments Field

The Application-aware ID option can be put in the SID Arguments field, which can be read by each node indicated by the SID in the SID list.

7.3.3. SRH TAG field

The Application-aware ID option can be put in the TAG field, which can be read by each node that processes the SRH.

8. IANA Considerations

IANA maintains the registry for the Options and Sub-TLVs.

Application-Para Option will require one new type code per sub-TLV defined in this document:

Type Value

TBD Application-aware ID Option

TBD Application-Para Option

TBD BW Sub-TLV

TBD Delay Sub-TLV

TBD Delay Variation Sub-TLV

TBD Packet Loss Sub-TLV

9. Security Considerations

TBD

10. References

10.1. Normative References

[I-D.filsfils-spring-srv6-network-programming]
Filsfils, C., Camarillo, P., Leddy, J., daniel.voyer@bell.ca, d., Matsushima, S., and Z. Li, "SRv6 Network Programming", draft-filsfils-spring-srv6-network-programming-07 (work in progress), February 2019.

[I-D.ietf-6man-segment-routing-header]
Filsfils, C., Dukes, D., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-21 (work in progress), June 2019.

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

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

10.2. Informative References

[RFC3272] Awduche, D., Chiu, A., Elwalid, A., Widjaja, I., and X. Xiao, "Overview and Principles of Internet Traffic Engineering", RFC 3272, DOI 10.17487/RFC3272, May 2002, <https://www.rfc-editor.org/info/rfc3272>.

Authors' Addresses

Zhenbin Li
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
Huawei Bld., No.156 Beiqing Rd.
Beijing 100095
China

Email: pengshuping@huawei.com

Chongfeng Xie
China Telecom
China Telecom Information Science&Technology Innovation park, Beiqijia
Town,Changping District
Beijing 102209
China

Phone: +86-10-50902116
Email: xiechf.bri@chinatelecom.cn

Cong Li
China Telecom
China Telecom Information Science&Technology Innovation park, Beiqijia
Town,Changping District
Beijing 102209
China

Phone: +86-10-50902556
Email: licong.bri@chinatelecom.cn