Network Working Group   Z. Li
Internet-Draft   S. Peng
Intended status: Standards Track   Huawei Technologies
Expires: September 12, 2019   March 11, 2019

Service-aware IPv6 Network
draft-li-6man-service-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, since the current network is lack of enough information of service requirements of such applications it is difficult to guarantee the SLA or it may take long time to provide such guarantee. This document proposes the solution to make use of IPv6 extensions header to convey the service requirement information along with the packet to the network to facilitate the service deployment and network resource adjustment to guarantee SLA for applications. Then it defines the service-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].

Status of This Memo

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

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on September 12, 2019.

Copyright Notice

Copyright (c) 2019 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

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, since the current network is lack of enough information of service requirements of such applications it is difficult to guarantee the SLA or it may take long time to provide such guarantee. This document proposes the solution to take use of IPv6 extensions header to convey the service requirement information along with the packet to the network to facilitate the service deployment and network resource adjustment to guarantee SLA for applications. Then it defines the service-aware

options which can be used in the different IPv6 extension headers for the purpose.

2. Use Cases

This section shows the various demanding requirements of some applications in the following use cases. The traffic of these applications needs to be differentiated from other traffic and applied with special treatments in the network.

2.1. Online Gaming

Good network performance is normally a prerequisite for satisfactory game play, especially for the online gaming. The maximum allowable ping rate (network latency) and the required minimum download/upload speed (network bandwidth) are the key factors to make the online gaming playable. 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 network paths with low latency and low packet loss need to be explicitly selected from the game players to the game server.

2.2. 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 (e.g. from Netflix) require more bandwidth in order to stream properly. Real time streaming services also requires 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.

3. 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 relies 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.

4. Framework

```
+-----
+
+-----+
|App x|----
\                                                                     /---->|
App x|
+-----+    |    +-----------+    +--------+       +---------+       +---------
+   |    +-----+
           \--->|           |    |        |---A---|         |---A---|
|---/
                |Edge Device|----|Head-End|---B---|Mid-Point|---B---|End-Point|
           /--->|           |    |        |---C---|         |---C---|
|---\
+-----+    |    +-----------+    +--------+       +---------+       +---------
+   |    +-----+
|App
y|----/
\---->|App y|
+-----
+
+-----+
```

Figure 1 Service-aware IPv6 Network

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

1. Service-aware Apps: The IPv6 enabled applications runs in the host which can add the service requirements of the applications on network through the IPv6 extension header ([RFC8200]) or remove it from the IPv6 extension header. The service requirement information includes the IPv6 service-aware ID which identifies the IPv6 packets of the traffic belongs to the specific SLA level/Applications/User and the 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]) enabled node along the SRv6 path which be programmed in the host. The Apps can also need not to add any service requirement information in the IPv6 extension header.

2. Service-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 service-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.

3. Service-process Head-End: The service requirements may be processed as a service path such as SRv6 TE path of SFC at the Service-process Head-End. The service requirements conveyed in the IPv6 packets can be mapped to a service path 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.

4. Service-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 specific policies which is out of the scope of this document.

5. Service-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 of the applications explicitly. According to these service requirement information carried in the IPv6 packets the network is able to adjust its resource fast to satisfy the service requirement of applications. The flow-driven method also reduces the challenges of inter-operability and loop control loop.

5. Service-aware Options

Two service-aware options are defined, i.e. Service-aware ID option and Service-Para Option to support the Service-aware IPv6 network.

5.1. Service-aware ID Option

The Service-aware ID option indicates the information of the applications, users, and service requirements, which is defined 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 Service-aware ID                      +
|                                                               |
+                                                               +
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2. IPv6 Service-aware ID Option

Option Type: TBD

Opt Data Len: 16 octets.

The IPv6 Service-aware ID is 128bits long which can have 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, which is shown in the following diagram:

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

Figure 3. IPv6 Service-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 indicates the service requirements of the identified application, which is shown in the following diagram:

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

Figure 4. IPv6 Service-aware ID Structure II

-- Structure III: An SRv6 SID, with its arguments as the information specified in Structure 2, which is shown in the following diagram:

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

Figure 5. IPv6 Service-aware ID Structure III

This Option can be put into the IPv6 Hop-by-Hop Options, Destination Options, and SRH TLV ([I-D.ietf-6man-segment-routing-header]).

5.2. Service-Para Option

The Service-Para Option is a variable-length option carrying multiple service requirement parameters. 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 6. 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 7. 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 8. 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 9. 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 10. 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.

6. IANA Considerations

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

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

Type Value

TBD Service-aware ID Option

TBD Service-Para Option

TBD BW Sub-TLV

TBD Delay Sub-TLV

TBD Delay Variation Sub-TLV

TBD Packet Loss Sub-TLV

7. Security Considerations

TBD

8. References

8.1. 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., Previdi, S., Leddy, J., Matsushima, S., and d. daniel.voyer@bell.ca, "IPv6 Segment Routing Header (SRH)", draft-ietf-6man-segment-routing-header-16 (work in progress), February 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>.

8.2. 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