

Network Working Group  
Internet-Draft  
Intended status: Standards Track  
Expires: August 26, 2021

Z. Li  
S. Peng  
Huawei Technologies  
D. Voyer  
Bell Canada  
C. Li  
China Telecom  
P. Liu  
China Mobile  
C. Cao  
China Unicom  
K. Ebisawa  
Toyota Motor Corporation  
S. Previdi  
Huawei Technologies  
J. Guichard  
Futurewei Technologies Ltd.  
February 22, 2021

**Application-aware Networking (APN) Framework**  
**draft-li-apn-framework-02**

**Abstract**

A multitude of applications are carried over the network, which have varying needs for network bandwidth, latency, jitter, and packet loss, etc. Some new emerging applications (e.g. 5G) have very demanding performance requirements. However, in current networks, the network and applications are decoupled, that is, the network is not aware of the applications' requirements in a fine granularity. Therefore, it is difficult to provide truly fine-granularity traffic operations for the applications and guarantee their SLA requirements.

This document proposes a new framework, named Application-aware Networking (APN), where application characteristic information such as application-aware identification and network performance requirements is carried in the packet encapsulation in order to facilitate service provisioning, perform fine-granularity traffic steering and network resource adjustment.

**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 August 26, 2021.

## Copyright Notice

Copyright (c) 2021 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

<a href="#">1.</a>	<a href="#">Introduction</a>	<a href="#">3</a>
<a href="#">2.</a>	<a href="#">Specification of Requirements</a>	<a href="#">3</a>
<a href="#">3.</a>	<a href="#">Terminology</a>	<a href="#">3</a>
<a href="#">4.</a>	<a href="#">APN Framework and Key Components</a>	<a href="#">4</a>
<a href="#">5.</a>	<a href="#">APN Requirements</a>	<a href="#">6</a>
<a href="#">5.1.</a>	<a href="#">Application-aware Information Conveying Requirements</a>	<a href="#">6</a>
<a href="#">5.2.</a>	<a href="#">Application-aware Information Handling Requirements</a>	<a href="#">8</a>
<a href="#">5.2.1.</a>	<a href="#">App-aware SLA Guarantee</a>	<a href="#">8</a>
<a href="#">5.2.2.</a>	<a href="#">App-aware network slicing</a>	<a href="#">8</a>
<a href="#">5.2.3.</a>	<a href="#">App-aware deterministic networking</a>	<a href="#">9</a>
<a href="#">5.2.4.</a>	<a href="#">App-aware service function chaining</a>	<a href="#">9</a>
<a href="#">5.2.5.</a>	<a href="#">App-aware network measurement</a>	<a href="#">10</a>
<a href="#">5.3.</a>	<a href="#">Security requirements</a>	<a href="#">10</a>
<a href="#">6.</a>	<a href="#">IANA Considerations</a>	<a href="#">10</a>
<a href="#">7.</a>	<a href="#">Security Considerations</a>	<a href="#">10</a>
<a href="#">8.</a>	<a href="#">Acknowledgements</a>	<a href="#">10</a>
<a href="#">9.</a>	<a href="#">Contributors</a>	<a href="#">10</a>
<a href="#">10.</a>	<a href="#">References</a>	<a href="#">11</a>
<a href="#">10.1.</a>	<a href="#">Normative References</a>	<a href="#">11</a>
<a href="#">10.2.</a>	<a href="#">Informative References</a>	<a href="#">12</a>
	<a href="#">Authors' Addresses</a>	<a href="#">12</a>



## **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 has very demanding network requirements and therefore require special treatment 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 fine granularity. Therefore, it is difficult to provide truly fine-granularity traffic operations for the applications and guarantee their SLA requirements accordingly.

[[I-D.li-apn-problem-statement-usecases](#)] describes the challenges of traditional differentiated service provisioning methods, such as five tuples used for ACL/PBR causing coarse granularity, DPI imposing high CAPEX & OPEX and security issues, as well as orchestration and SDN-based solution causing long control loops.

This document proposes a new framework, named Application-aware Networking (APN), aiming to guarantee fine-granularity SLA requirements of applications, where application characteristic information such as application-aware identification and network performance requirements is carried in the packet encapsulation in order to facilitate service provisioning, perform traffic steering, and network resource adjustment.

## **2. Specification of Requirements**

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 document is not a protocol specification and the key words in this document are used for clarity and emphasis of requirements language.

## **3. Terminology**

ACL: Access Control List

APN: Application-aware Networking

APN6: Application-aware Networking for IPv6/SRV6

DPI: Deep Packet Inspection

MPLS: Multiprotocol Label Switching



PBR: Policy Based Routing

QoE: Quality of Experience

SDN: Software Defined Networking

SLA: Service Level Agreement

SR: Segment Routing

SR-MPLS: Segment Routing over MPLS dataplane

SRv6: Segment Routing over IPv6 dataplane

#### 4. APN Framework and Key Components

The APN framework is shown in Figure 1. The key components include Service-aware App, App-aware Edge Device, App-aware-process Head-End, App-aware-process Mid-Point, and App-aware-process End-Point.

Packets carry application characteristic information (i.e. application-aware information) which includes the following information:

- o Application-aware identification information: identifies the application (group), the user (group), and their SLA requirements, indicating that all packets belonging to the same flow will be given the same treatment by the network. ;
- o Network performance requirements information that specify at least one of the following parameters: bandwidth, delay, delay variation, packet loss ratio, security, etc.

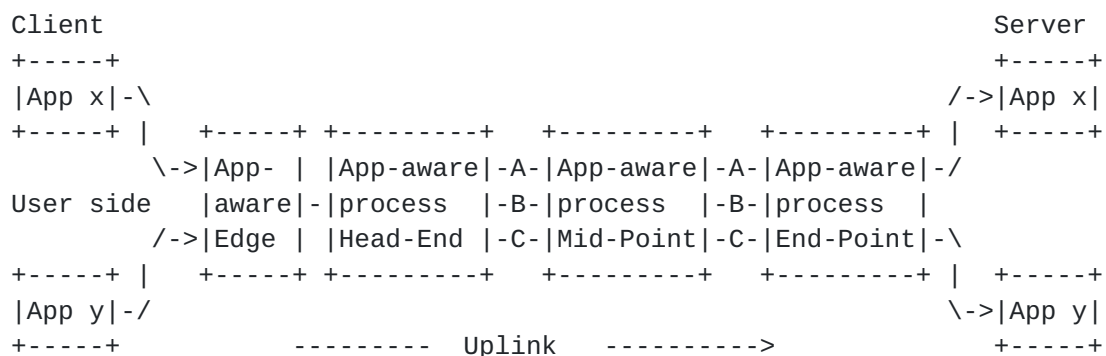


Figure 1: Framework and Key Components

The key components are introduced as follows.



1. Service-aware App: the host obtains the application characteristic information of the Service-aware App and generates the packets which carry the application characteristic information in the encapsulation. If carried in the packets, this information is used by the App-aware-process Head-End to determine the path between the App-aware-process Head-End and the App-aware-process End-Point for forwarding the packets to their destination, that is, to steer the packet into a given policy which satisfies the application requirements. With the Service-aware App, the network is aware of the service requirements expressed by the applications explicitly. In the APN framework, the application is not mandatory to be service-aware.
2. App-aware Edge Device: this network device receives packets from applications and obtains the application characteristic information. If the application is not Service-aware App, the application characteristic information can be retrieved by packet inspection, derived from services information such as double VLAN tagging (C-VLAN and S-VLAN), or added according to the local policies which is out of the scope of this document. The App-aware Edge Device adds the application characteristic information in the encapsulation on behalf of the application. The packets carrying the application characteristic information will be sent to the App-aware-process Head-End, and the application characteristic information will be used to determine the path between the App-aware-process Head-End and the App-aware-process End-Point for forwarding the packets.
3. App-aware-process Head-End: This network device receives packets and obtains the application characteristic information. A set of paths, tunnels or SR policy, exist between the App-aware-process Head-End and the App-aware-process End-Point. The App-aware-process Head-End maintains the matching relationship between the application characteristic information and the paths between the App-aware-process Head-End and the App-aware-process End-Point. The App-aware-process Head-End determines the path between the App-aware-process Head-End and the App-aware-process End-Point according to the application characteristic information carried in the packets and the matching relationship with it, which satisfies the service requirements of the application. If there is no such matching path found, the App-aware-process Head-End can set up a path towards the App-aware-process End-Point, and the matching relationship will be stored. The App-aware-process Head-End forwards the packets along the path. The application information conveyed by the packet received from the App-aware Edge Device can also be copied or be mapped to the outgoing packet header, e.g, IPv6 header followed by an extension header for further application-aware process.





4. App-aware-process Mid-Point: the Mid-Point provides the path service according to the path set up by the App-aware-process Head-End which satisfies the service requirements conveyed by the packets. The Mid-Point may also adjust the resource locally to guarantee the service requirements depending on a specific policy and the application-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 together with the outer encapsulation or go on to be conveyed with the packets.

According to the application characteristic information carried in the 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 interoperability and long control loop.

## **5. APN Requirements**

APN doesn't mandate a specific encapsulation however it is reasonable to assume that most of the APN benefits are achieved when utilizing IPv6 encapsulation (e.g. IPv6 header as well as, possibly, extension headers). APN6 (the APN architecture applied to the IPv6/SRv6 data plane) consists of the application-aware information conveyed into the network through the use of IPv6 header and Extension Headers and where the network performs service provisioning, traffic steering, and SLA guarantee according to such information. This section specifies the requirements for supporting the APN framework, including the requirements for conveying and handling the application-aware information and related security requirements. Other encapsulation may be used with some obvious constraint such as, as in the case of MPLS, the limited space available in the header (i.e., 20-bit label size).

### **5.1. Application-aware Information Conveying Requirements**

The application-aware information includes application-aware identification information and network performance requirements information.

1. Application-aware identification information includes the following identifiers (IDs),



- \* SLA level: indicates the level of SLA requirement such as Gold, Silver, Bronze. In some cases, color (e.g. red, green) can be used to indicate the SLA level.
- \* Application ID: identifies an application (group) of the traffic.
- \* User ID: identifies the user (group) of the traffic.
- \* Flow ID: identifies the key session of the application traffic.

The different combinations of the IDs can be used to provide different granularity of the service provisioning and SLA guarantee for the traffic.

2. Network performance requirements information includes the following parameters:

- \* Bandwidth: the bandwidth requirement
- \* Latency: the latency requirement
- \* Packet loss ratio: the packet loss ratio requirement
- \* Jitter: the jitter requirement

The different combinations of the parameters are for further expressing the more detailed service requirements, conveyed together with the Application-aware identification information, which can be used to match to appropriate tunnels/SR Policies, queues that can satisfy these service requirements. If not available, new tunnels/SR Policies can also be triggered to be set up.

[REQ 1a]. Application-aware identification information MUST include Application ID to indicate the application (group) that generates the packet.

[REQ 1b]. SLA level is RECOMMENDED to be included in the Application-aware identification information.

[REQ 1c]. User ID and Flow ID are OPTIONAL to be included in the Application-aware identification information.

[REQ 1d]. Network performance requirements information is OPTIONAL.



[REQ 1e]. All the nodes along the path SHOULD be able to process the application-aware information if needed.

[REQ 1f]. The application-aware information can be generated directly by application, or by the application-aware edge devices though packet inspection or local policy.

[REQ 1g]. The application-aware information SHOULD be kept intact when directly copied from the application-aware edge devices and carried in the packet.

## **5.2. Application-aware Information Handling Requirements**

The app-aware-process Head-End and app-aware-process Mid-Point perform matching operation against the application-aware information, that is, to match IDs and/or service requirements to the corresponding network resources (tunnels/SR policies, queues).

### **5.2.1. App-aware SLA Guarantee**

In order to achieve better Quality of Experience (QoE) of end users and engage customers, the network needs to be able to provide fine-granularity and even application-level SLA guarantee [[I-D.li-apn-problem-statement-usecases](#)].

[REQ 2-1a]. With the application-aware information, the App-aware-process Head-End SHOULD be able to steer the traffic to the tunnel/SR policy that satisfies the matching operation.

[REQ 2-1b]. With the application-aware information, the App-aware-process Head-End SHOULD be able to trigger the setup of the tunnel/SR policy that satisfies the matching operation.

[REQ 2-1c]. With the application-aware information, the App-aware-process Head-End and Mid-Point SHOULD be able to steer the traffic to the queue that satisfies the matching operation.

[REQ 2-1d]. With the application-aware information, the App-aware-process Head-End and Mid-Point SHOULD be able to trigger the configuration of the queue that satisfies the matching operation.

### **5.2.2. App-aware network slicing**

Network slicing provides ways to partition the network infrastructure in either control plane or data plane into multiple network slices that are running in parallel. The resources on each node need to be associated to corresponding slices.



[REQ 2-2a]. With the application-aware information, the App-aware-process Head-End SHOULD be able to steer the traffic to the slice that satisfies the matching operation.

[REQ 2-2a]. With the application-aware information, the App-aware-process Mid-Point SHOULD be able to associate the traffic to the resources in the slice that satisfies the matching operation.

### **5.2.3. App-aware deterministic networking**

Along the path each node needs to provide guaranteed bandwidth, bounded latency, and other properties relevant to the transport of time-sensitive data for the Detnet flows that coexist with the best-effort traffic.

[REQ 2-3a]. With the application-aware information, the App-aware-process Head-End SHOULD be able to steer the traffic to the appropriate path that satisfies the matching operation.

[REQ 2-3b]. With the application-aware information, the App-aware-process Head-End SHOULD be able to trigger the setup of the appropriate path that satisfies the matching operation for the Detnet flows.

[REQ 2-3c]. With the application-aware information, the App-aware-process Mid-Point SHOULD be able to associate the traffic to the resources along the path that satisfies the performance guarantee.

[REQ 2-3d]. With the application-aware information, the App-aware-process Mid-Point SHOULD be able to reserve the resources for the Detnet flows along the path that satisfies the performance guarantee.

### **5.2.4. App-aware service function chaining**

The end-to-end service delivery often needs to go through various service functions, including traditional network service functions such as firewalls, DPI as well as new application-specific functions, both physical and virtual. SFC is applicable to both fixed and mobile networks as well as data center networks.

[REQ 2-4a]. With the application-aware information, the App-aware-process devices SHOULD be able to steer the traffic to the appropriate service function.

[REQ 2-4b]. The App-aware-process devices SHOULD be able to process the application-aware information carried in the packets.





#### **5.2.5. App-aware network measurement**

Network measurement can be used for locating silent failure and predicting QoE satisfaction, which enables real-time SLA awareness/proactive OAM.

[REQ 2-5a]. The App-aware-process devices SHOULD be able to perform IOAM based on the Application-aware identification information.

[REQ 2-5b]. The network measurement results can be reported based on the Application-aware identification information and verify whether the performance requirements are satisfied.

#### **5.3. Security requirements**

[REQ 3a]. The security mechanism defined for APN MUST allow an operator to prevent applications sending arbitrary application-aware information without agreement with the operator.

[REQ 3b]. The security mechanism defined for APN MUST prevent an application requesting a service which it is not entitled to get.

[REQ 3c]. The APN mechanisms MUST preserve user privacy. More details related to this requirement can be found in [\[I-D.peng-apn-security-privacy-consideration\]](#).

### **6. IANA Considerations**

This document does not include an IANA request.

### **7. Security Considerations**

[I-D.li-apn-problem-statement-usecases] and [\[I-D.peng-apn-security-privacy-consideration\]](#) describe the security considerations and requirements for APN.

### **8. Acknowledgements**

The authors would like to acknowledge Robert Raszuk (Bloomberg LP), and Yukito Ueno (NTT Communications Corporation) for their valuable reviews and comments.

### **9. Contributors**

Daniel Bernier  
Bell Canada

Email: [daniel.bernier@bell.ca](mailto:daniel.bernier@bell.ca)



Chongfeng Xie  
China Telecom

Email: xiechf@chinatelecom.cn

Feng Yang  
China Mobile

Email: yangfeng@chinamobile.com

Zhuangzhuang Qin  
China Unicom

Email: qinzhuangzhuang@chinaunicom.cn

Chang Liu  
China Unicom

Email: liuc131@chinaunicom.cn

Gyan Mishra  
Verizon

Email: hayabusagsm@gmail.com

Luis M. Contreras  
Telefonica

Email: contreras.ietf@gmail.com

Luc-Fabrice Ndifor Ngwa  
MTN

Email: Luc-Fabrice.Ndifor@mtn.com

## **10. References**

### **10.1. Normative References**

[I-D.li-apn-problem-statement-usecases]

Li, Z., Peng, S., Voyer, D., Xie, C., Liu, P., Qin, Z., Ebisawa, K., Previdi, S., and J. Guichard, "Problem Statement and Use Cases of Application-aware Networking (APN)", [draft-li-apn-problem-statement-usecases-01](#) (work in progress), September 2020.



[I-D.peng-apn-security-privacy-consideration]

Peng, S., Li, Z., Voyer, D., Li, C., Liu, P., and C. Cao,  
"APN Security and Privacy Considerations", [draft-peng-apn-security-privacy-consideration-00](#) (work in progress),  
September 2020.

[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>>.

[RFC7665] Halpern, J., Ed. and C. Pignataro, Ed., "Service Function  
Chaining (SFC) Architecture", [RFC 7665](#),  
DOI 10.17487/RFC7665, October 2015,  
<<https://www.rfc-editor.org/info/rfc7665>>.

[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>>.

[RFC8578] Grossman, E., Ed., "Deterministic Networking Use Cases",  
[RFC 8578](#), DOI 10.17487/RFC8578, May 2019,  
<<https://www.rfc-editor.org/info/rfc8578>>.

## **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  
China

Email: [lizhenbin@huawei.com](mailto:lizhenbin@huawei.com)

Shuping Peng  
Huawei Technologies  
China

Email: [pengshuping@huawei.com](mailto:pengshuping@huawei.com)



Daniel Voyer  
Bell Canada  
Canada

Email: daniel.voyer@bell.ca

Cong Li  
China Telecom  
China

Email: licong@chinatelecom.cn

Peng Liu  
China Mobile  
China

Email: liupengyjy@chinamobile.com

Chang Cao  
China Unicom  
China

Email: caoc15@chinaunicom.cn

Kentaro Ebisawa  
Toyota Motor Corporation  
Japan

Email: ebisawa@toyota-tokyo.tech

Stefano Previdi  
Huawei Technologies  
Italy

Email: stefano@previdi.net

James N Guichard  
Futurewei Technologies Ltd.  
USA

Email: jguichar@futurewei.com



