Workgroup: Network Working Group
Internet-Draft:
draft-li-rtgwg-apn-app-side-framework-00
Published: 22 October 2023
Intended Status: Standards Track
Expires: 24 April 2024
Authors: Z. Li
Huawei Technologies
S. Peng
Huawei Technologies

Extension of Application-aware Networking (APN) Framework for Application Side

Abstract

The Application-aware Networking (APN) framework defines that application-aware information (i.e. APN attribute) including APN identification (ID) and/or APN parameters (e.g. network performance requirements) is encapsulated at network edge devices and carried in packets traversing an APN domain in order to facilitate service provisioning, perform fine-granularity traffic steering and network resource adjustment. This document defines the extension of the APN framework for the application side. In this extension, the APN resources of an APN domain is allocated to applications which compose and encapsulate the APN attribute in packets. When the network devices in the APN domain receive the packets carrying APN attribute, they can directly provide fine-granular traffic operations according to these APN attributes in the packets.

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 24 April 2024.

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 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 as well as orchestration and SDN-based solution causing long control loops.

[I-D.li-apn-framework] proposes the framework of Application-aware Networking (APN), where application-aware information (APN attribute) including application-aware identification (APN ID) and application-aware parameters (APN Parameters), is encapsulated at network edge devices and carried along with the encapsulation of the tunnel used by the packet when traversing an APN domain. By APN domain we intend the operator infrastructure where APN is used from edge to edge (ingress to egress) and where packets are encapsulated

using an outer header incorporating the APN information. The APN attribute will facilitate service provisioning and provide fine-granular services in the APN domain.

In the APN framework the APN attribute is acquired at the ingress of the APN domain based on the existing information in the incoming packet header (i.e. source and destination addresses, incoming L2 (or) MPLS encapsulation, incoming physical/virtual port information, the other fields of the 5-tuple if they are not encrypted) in the edge devices. The APN information is then added to the data packets along with the tunnel encapsulation. The packet traverses the domain and, when exiting the domain, the outer header along with the APN information is removed.

This document defines the extensions of the APN framework for application side. In this extension, the APN resources of the APN domain is allocated to applications which compose and encapsulate the APN attribute in packets. When network devices in the APN domain receives packets carrying APN attribute, they can directly apply policies for these traffic flows according to the APN attribute encapsulated by applications.

2. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 RFC 2119 [RFC2119] RFC 8174 [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Terminology

ACL: Access Control List

APN: Application-aware Networking

APN6: Application-aware Networking for IPv6/SRv6

MPLS: Multiprotocol Label Switching

PBR: Policy Based Routing

QoE: Quality of Experience

SDN: Software Defined Networking

SLA: Service Level Agreement

4. APN Framework for Application Side

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

Packets carry application characteristic information (i.e. APN attribute) which includes the following information:

*Application-aware identification (APN ID): identifies the set of attributes, indicating that all packets belonging to the same flow will be given the same treatment by the network. APN ID is mandatory.

*Application-aware parameters (APN parameters): The typical application-aware parameters are the network performance requirement parameters including bandwidth, delay, delay variation, packet loss ratio, etc. APN parameters are optional.

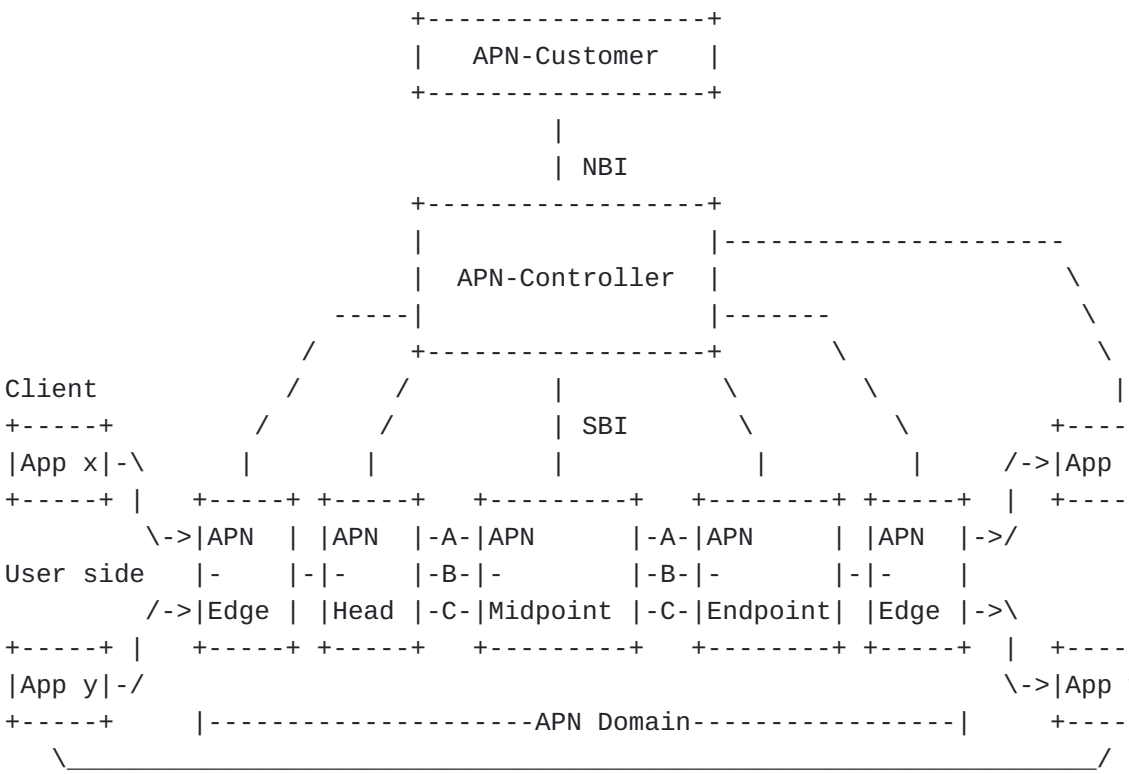

Figure 1: APN Framework for Application Side

In the extension of the APN framework for application side, the new key components, APN-capable Application Server (AAS) and APN-capable Application Client (AAC), are introduced as follows:

*APN-capable Application Server (AAS): The AAS requests the APN-Controller to allocate the APN resources of an controlled APN domain. And the AAS allocates the APN resources received from the APN-Controller to the AAC to compose APN attribute according to the requirement from the AAC. When the AAS sends packets to the AAC, it adds APN attribute in these packets. The request sent by the AAS to the APN-Controller includes the application information, network service requirement, etc. The APN resources allocated by the APN-controller to the AAS includes sets of APN IDs and corresponding network service attributes.

*APN-capable Application Client (AAC): The AAC requests the AAS to allocate the APN resources. The AAC composes the APN attribute according to the allocated APN resources from the AAS. When the AAC sends packets to the AAS, it adds the APN attribute in these packets. The APN resources allocated by the AAS to the AAC includes the unique APN ID and the corresponding network service attributes.

In the extension of the APN framework for the application side, the functionalities of the following key components are extended or changed:

*APN-Edge: In the extension of APN framework for the application side, since the APN attribute is added by the application, the functionalities of the APN-Edge needs to be changed. The APN-Edge can directly transmit the packets without encapsulating tunnels for the purpose of carrying APN attribute. If the APN-Edge needs to encapsulate a tunnel for packets, it can directly obtain the APN attribute from these packets sent by the AAS/AAC and the APN attribute can be copied or be mapped into the outer tunnel header.

*APN-Head: The APN-Head can directly obtain the APN attribute from packets sent by the AAS/AAC to apply corresponding policies.

*APN-Midpoint: If policies need to be adjusted on the APN-Midpoint, the APN-Midpoint can also directly obtain the APN attribute from packets sent by the AAS/AAC.

*APN-Endpoint: The APN-Endpoint MUST keep the APN attribute in packets sent by the AAS/AAC without any change.

*APN-Controller: In the extension of APN framework for the application side, the APN-Controller is responsible for

processing the request from the AAS and allocating the APN resources of the controlled APN domain to the AAS.

The APN attribute need to be transmitted among the AAS, AAC and APN domain. The security mechanism MUST be introduced to guarantee the security of the transmission. The details of the security mechanism will be proposed in future versions of the draft.

5. Requirements

According to the extension of APN framework for the application side, there are following basic protocol extension requirements:

[REQ01] Protocol extensions MUST be defined for the AAS to request the APN-Controller to allocated the APN resources of the APN domain.

[REQ02] Protocol extensions MUST be defined for the APN-Controller to notify the allocated APN resources to the AAS.

[REQ03] Protocol extensions MUST be defined for the AAC to request the AAS to allocate the APN resources.

[REQ04] Protocol extensions MUST be defined for the AAS to notify the allocated APN resources to the AAC.

[REQ05] Security mechanism MUST be defined to guarantee for that the APN attribute being securely transmitted among the AAS, AAC and the APN domain.

6. IANA Considerations

This document does not include an IANA request.

7. Security Considerations

In the extension of the APN Framework for the application side, the security issues are proposed because the APN attributes need to be transmitted among the AAS, AAC and the APN domain. The security mechanism MUST be introduced to guarantee the secure transmission. The details of the security mechanism and the security consideration will be proposed in the future version of the draft or an independent draft.

8. References

8.1. Normative References

[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>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[I-D.li-apn-framework] Li, Z., Peng, S., Voyer, D., Li, C., Liu, P., Cao, C., and G. S. Mishra, "Application-aware Networking (APN) Framework", Work in Progress, Internet-Draft, draft-li-apn-framework-07, 3 April 2023, <https://datatracker.ietf.org/doc/html/draft-li-apn-framework-07>.

8.2. Informative References

[I-D.li-apn-problem-statement-usecases]
Li, Z., Peng, S., Voyer, D., Xie, C., Liu, P., Qin, Z., and G. S. Mishra, "Problem Statement and Use Cases of Application-aware Networking (APN)", Work in Progress, Internet-Draft, draft-li-apn-problem-statement-usecases-08, 3 April 2023, <https://datatracker.ietf.org/doc/html/draft-li-apn-problem-statement-usecases-08>.

Authors' Addresses

Zhenbin Li
Huawei Technologies
China

Email: lizhenbin@huawei.com

Shuping Peng
Huawei Technologies
China

Email: pengshuping@huawei.com