Abstract

Currently, there are network management applications that present specific demands on a communication network. This document describes the APONF basic architecture, its elements and interfaces. The main APONF architecture entities are the Network Management Application Agent (NMAA), which is a network entity that creates and runs network services, and Application-based Policy Decision (ABPD), which supports classified application models. Each of these models support application demands that are similar in nature and therefore can be grouped/classified together. Moreover, the ABPD maps the classified application models into network capabilities, e.g., network management and traffic policies.

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 http://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 January 3, 2015.
1. Introduction

As the Internet grows, more and more new services keep on arising, and network traffic is rapidly increased, which may result in slow performance of network devices (e.g., BRAS) and poor end-user experience. In addition, especially for cloud applications, the cloud tenants and developers usually need to use the communication network capabilities, such as dynamic network management and dynamic traffic steering, easily, accurately and efficiently. In this way, the deployment of new applications and services may be accelerated and the user experience can be improved.

In particular, today network operators are challenged to create an abstract view of their network infrastructure and help service developers on using and programming this abstraction rather than manipulating individual devices. In this context, network management applications can be used to provide the required configuration and
application programming interfaces to such service developers. Subsequently, a network management application can use the application based demands and possibly update its associated network configuration and/or network topology model. Examples of network management applications that can modify the network configuration and/or network topology models are for example, Distributed Data Center Application and IPv6 transitions, need to change the network infrastructure configuration.

The up to date network configuration and network topology model needs to (1) be communicated to e.g., the network management and controlling systems, (2) map the network configuration and network topology models into specific network management policies, i.e., device level configuration models.

Currently, there are no IETF standard mechanisms or modeling languages that can directly be applied to model the network configuration nor the network topology. IETF has however, created the IETF SFC WG [SFC] to document a new approach to service delivery and operation, where one of its goals is to realize an abstract view of a network by using a service graph instance denoted as the Service Function Path (SFP). This will enable the development of suitable models for network configuration and network topology.

The main goal of this document is to specify the APONF basic architecture, its elements and interfaces. The main APONF architecture entities are the Network Management Application Agent (NMAA) and the Application-based Policy Decision (ABPD). NMAA is a network entity that creates and runs network services and is able to use the application based demands and possibly update their associated SPF based network configuration and/or network topology model. The ABPD is able to map the SFP based network configuration and network topology models into specific network management policies, i.e., device level configuration models. The definition of these network management policies is out of the APONF scope.

2. Terminology

AECON (Application Enabled Collaborative Network): The main goal of the AECON activity (currently BOF) is to allow applications to explicitly signal their flow characteristics to the network.

Device level configuration model: supports the description of the network management policies and it describes the configuration details at the device level.

Network Management Application: Network management are Operational Support System (OSS) like applications that help a communication
service provider to monitor, control, analyze and manage a communication network.

Network configuration model: provides a declarative configuration of the network.

Network topology model: describes the topology of a multi-layer network.

NFVcon (Network Functions Virtualization configuration): The main goal of this activity (BOF status) is to support the dynamic configuration of NFV instances.

Service Function Chain (SFC): A service Function chain defines an ordered set of service functions that must be applied to packets and/ or layer-2 frames selected as a result of classification. The implied order may not be a linear progression as nodes may copy to more than one branch. The term service chain is often used as shorthand for service function chain.

Service Function Path (SFP): The instantiation of a service function chain in the network. Packets follow a service function path from a classifier through the required instances of service functions in the network.

VNF (Virtualized Network Function): An implementation of an executable software program that constitutes the whole or a part of an NF and can be deployed on a virtualization infrastructure.

3. Overview of the APONF Architecture

This section depicts an overview of the architecture of application-based policy on network functions. Figure 1 shows APONF architecture. The basic components of the APONF architecture are:

Network Management Application: Operational Support System (OSS) like applications that help a communication service provider to monitor, control, analyze and manage a communication network. Several network management applications MAY communicate with the Application Based Policy Decision block via the Network Management Application Agent.

The Network Management Application Agent (NMAA): The NMAA is part of the network management application and is a network entity that creates and runs network services. These network services should be developed by an operator, which are assumed to be already available. The NMAA is able to generate for each of these network services and using application based demands a SFP based network configuration and network topology model.
Application Based Policy Decision (ABPD): A network entity which provides an interface to NMAA(s) and is able to map the classified application based models, which are including the classified application based demands and the SFP based network configuration and network topology models, into specific network management policies, i.e., device level configuration models, which are used by the communication network. ABPD can communicate with multiple NMAAs simultaneously.

Network Element (NE): handles incoming packets based on the ABDP network management policies and the corresponding network management and manipulation procedures.

Figure 1 shows the basic architecture of application-based policy on network functions.
4. Network Management Applications

This architecture is expected to be used for several categories of network management applications. Such network management applications are representing the realizations of the APONF use cases, which are: "Distributed Data Center " [ID.cheng-aponf-ddc-use-cases], "IPv6 transition " [ID.sun-aponf-openv6-use-cases],
These network management applications are represented by a set of network services. Each network service can be represented by a classified application based policy model, since it can model the group of demands coming from a bundle of applications that impose similar requirements on the communication network. Such network services can be "Distributed Data Center", "IPv6 transition", "Virtualized Enterprise Applications" and "Source Address Validation and Traceback (SAVI)" and "Using the abstract view of network by service developers".

4.1. Network Management Application Agent (NMAA)

The NMAA is part of the network management application and is a network entity that creates and runs network services. These network services should be developed by an operator, which are assumed to be already available.

The assumption here is that the network management application has a complete view of the available network and network capabilities that it can use. Moreover, it is assumed that the network management application is able to have the abstract view of the network and on how the network service is mapped into this abstract view. This network abstract view is defined using the Service Function Path (SFP), specified in the IETF SFC WG. It is assumed that the NMAA can use the network service description and create a SFP based network configuration and network topology model, by using the guidelines provided by the SFC WG.

An NMAA is a typical OSS gateway or Network Management Station entity, that needs to support the following new functional blocks as shown in Figure 2:
o Typical OSS (Operations Support System) features.

o Create/Update SFPs: this is a NMAA functional block and is used by the NMAA to use the network service and create or update an SFP. The assumption used here is that the description of the network services is provided to end user applications in such a way that the end user application developer can use and program certain network capabilities in such a way that the end user application can increase its support for end user QoE. The modified versions of the network service are made known to the network management application and NMAA. This event initiates the update of the SFP.

o End User Application Interaction: this functional block is used to provide and receive information to/from the end user application engine. This functional block is in charge to provide the description of the network services to end user applications in such a way that the end user application developer can use and program certain network capabilities in such a way that the end user application can increase its support for end user QoE. This functional block is also used to receive the modified versions of the network service from the end user application and to inform the Create/Update SFPs functional block about this change. This event initiates the update of the SFP. Note that it is assumed that the realization of this functional block and the interface with the end user are out of the APONF’s scope.

Figure 2: NMAA Functionality Block Diagram
NMAA NSIS (Next Steps in Signaling) interface: this functional block is used to support the enhanced APONF NSIS protocol engine as described in Section 7.

The Network Management Application Agent (NMAA) will use the APONF interface to communicate with the Application Based Policy Decision (ABPD) entity.

5. Application Based Policy Decision

The Application-Based Policy Decision (ABPD) block, is an entity used between the Network Management Applications and the network elements to provide and maintain the application based policies. It supports the APONF interface/protocol and is a software repository, which stores the information associated with each NE, and maps the classified application models, i.e., application based demands and the SPF based network configuration and network topology models, into existing network management policies, i.e., device configuration models. In particular, by creating application based policies that mirror application semantics, a better mapping to existing network management policies can be realized. This provides a simple, self-documenting mechanism for capturing application-based policy requirements and mapping them to existing network management policies. This will allow applications to use the network capabilities in a more accurate and efficient way.

Figure 3 illustrates the ABPD functionality block diagram, which is based on [ID.farrkingel-pce-abno-architecture] and enhanced to satisfy the demands of the APONF use cases.
The Application Based Policy Decision (ABPD) block includes all the functional blocks provided in Figure 1 of [ID.farrkingel-pce-abno-architecture], together with the following new defined functional blocks:

- Fresh SFPs Maintenance: maintains a fresh abstract view of the network. Note that this is realized using the Service Function Path (SFP), specified in the IETF SFC WG. Important to note is that for each network service / classified application model that is managed by a network management application a different SFP is needed. So in order to support this the APONF architecture needs to support a functional block that stores all these abstract views
of the network in different SFPs that are identified by an unique ID.

- Application to Network Mapping: the following features are supported by this functional block:

  1. Translates the actions and the changed SFP received from the network management application, see explanation below, to a new SFP. This is accomplished by using application based demands generated by network management applications systems to map the SFP based network configuration and network topology models into specific network management policies, i.e., into device level configuration models. Such application based demands are:

    2. Encapsulating, de-encapsulating packets associated with a flow into a tunnel (for example, VPN service, IPv6 transition service demands on the network).

    Blocking, or dropping packets associated with a flow in (the edge of) the network element when the network security service is aware of the attack (for example, SAVI service, Anti-DoS service demands on the network).

    Configure and dynamically reconfigure data centers to the steer and reroute traffic associated with a specific flow.

    Configure and dynamically reconfigure data centers to change priorities of different types of traffic associated with a specific flow.

    logging the traffic associated with a flow for network security service, optimization of the traffic based on the IETF ALTO [ID.ietf-alto-protocol].

    Other actions defined by the administrator.

3. if required updates all databases, see Section 2.3.1.8 of [ID.farrkingel-pce-abno-architecture].

4. Uses existing network management and signaling protocols, i2rs [I2RS], SFC [SFC], NETCONF [NETCONF], NFVcon, etc., to request the implementation of the changes into the network.
o ABPD Network Management Interface: this functional block provides the interface with existing network management, i2rs, SFC, NETCONF, NFVcon, etc. protocols to request and negotiate the implementation of the changes into the network configuration.

o ABPD NSIS (Next Steps in Signaling) interface: this functional block is used to support the enhanced APONF NSIS protocol engine. The definition of the network management policies is out of the APONF scope.

These application-based policy models can meet the application’s demands on the communication network and map these demands to network management policies that can be understood by the communication network.

6. Network Elements

The Network Element (NE) handles incoming packets based on the policy information communicated with the ABPD block and makes corresponding policy enforcement, which is based on existing network management policies, see Section 5.

A NE may be a physical entity or a virtual entity and is locally managed, whether via CLI, SNMP, or NetConf. Examples of NEs can include:

- A router that has an extended function module. The extended module handles incoming packets basing on the flow table of the module.
- A server that runs vRouter or vSwitch.
- A CGN that runs NAT, Tunnel En/De-capsulation functions.
- A virtual network function entity.

7. The APONF Interface

This APONF Interface/Protocol, needs to be specified by the APONF effort and is used to support the communication between the NMAA entity and the ABPD entity. The IETF Next Steps in Signaling (NSIS) protocol may be extended in two ways to support this interface, see [ID.karagiannis-aponf-problem-statement]:

1. Extend NSIS GIST [RFC5971] in such a way that it can be used for off-path support;
2. Specify a new signaling protocol (NSIS Signaling Layer Protocol), similar to the NAT/Firewall NSLP [RFC5973] that can be applied and support the APONF use cases. This signaling protocol is denoted as APONF NSLP.

8. Security Considerations

Security is a key aspect of any protocol that allows state installation and extracting of detailed configuration states. More investigation remains to fully define the security requirements, such as authorization and authentication levels.

9. IANA Considerations

No IANA considerations.

10. Acknowledgements

The authors of this draft would like to thank the following persons for the provided valuable feedback: Jose Saldana, Spencer Dawkins, Jun Bi, Xing Li, Chongfeng Xie, Benoit Claise, Ian Farrer, Marc Blanct, Zhen Cao, Hosnieh Rafiee, Mehmet Ersue, Mohamed Boucadair.

11. informative References

[ID.farrkingel-pce-abno-architecture]

[ID.karagiannis-aponf-problem-statement]

[ID.montpetit-transport-use-cases]


Authors’ Addresses

Cathy Zhou
Huawei Technologies
Bantian, Longgang District
Shenzhen  518129
P.R. China

Email: cathy.zhou@huawei.com

Tina Tsou
Huawei Technologies
Bantian, Longgang District
Shenzhen  518129
P.R. China

Email: Tina.Tsou.Zouting@huawei.com

Diego Lopez
Telefonica

Email: diego@tid.es

Georgios Karagiannis
University of Twente

Email: g.karagiannis@utwente.nl

Qiong Sun
China Telecom
No.118 Xizhimennei street, Xicheng District
Beijing  100035
P.R. China

Email: sungiong@ctbri.com.cn