

Appsawg  
Internet-Draft  
Intended status: Informational  
Expires: January 16, 2014

H. Song  
Huawei  
July 15, 2013

The Problems of Service Configuration in NFV Context  
draft-song-appsawg-service-template-00

Abstract

This document describes the problem space of virtual function installation and dynamic configuration in network function virtualization (NFV) context. And identify the scope that needs standardization.

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 16, 2014.

Copyright Notice

Copyright (c) 2013 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 (<http://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. Background . . . . .	2
2. Terminology . . . . .	3
3. Problems of Service Configuration . . . . .	3
4. Scope for Standardization . . . . .	4
5. Normative References . . . . .	4
Author's Address . . . . .	4

## 1. Background

Users always have different requirements when they need a software. So a software vendor often provides a lot of functions to satisfy a majority of its users. But one user might only need a few sub-set functions. For example, a firewall could protect against various attacks, but a user's environment decides that he only needs to be protected from several attacks, and other attacks may not happen in this context. As a result, these functions exist in the software as components. There are several possibilities:

- (1) A software vendor distinguish users as several classes, and provide related versions of software to the users accordingly, for example, "home edition".
- (2) When user request a software, a person negotiate with the software vendor, and the software vendor makes a specified version of software to the user, in this version, it enables the components that the user need, and disables those unneeded.
- (3) The user get a license and software packet, and with the license, it allows the user to choose inside a range of components for installation. The user enables components that he wants in that range.

But these methods either too complex, or authorize the user with more components than what he wants.

In the context of network function virtualization (NFV), more and more network functions become to be available in a virtualized function way. It adopts the common IT infrastructure instead of physical hardware box to implement these network functions. The benefits of this method is to reduce cost through improved infrastructure reusability and lower entry of the industry, which allows more software vendors. Various virtual functions exist in the network. They are deployed to virtual machines through the NFV control center. These virtual functions can be replaced with new virtual functions when needed, with only re-configuring it with a new software through the control center. In this case, NFV control center is just like a broker for many software applications.

## 2. Terminology

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 [RFC2119]. And the following terms used in this document have their definitions below.

## 3. Problems of Service Configuration

There are several problems in the context of NFV to configure a virtual function, which makes it different from the traditional ways.

First, in the NFV framework, a software is installed according to the user (enterprise and etc.) requirements. And it is not installed locally in the user's equipment, but remotely in the network. NFV's control center needs to coordinate the necessary infrastructure resources for the installation. So the user does not have direct control over the software installation position or the hardware/software resources. But the control center has the direct control. In a result, the user needs to interact with the control center to accomplish the configuration of the components in a software installation.

Second, the NFV control center is just like a broker for various software applications. If every software vendor has its own proprietary messages for the software installation component configuration, then it will make the control center and the user environment more complex. A uniform and standard component configuration is more appropriate for this context.

Third, if the software vendor does not provide a clear description of these software components, then users do not know how to choose among those components. So the control center also needs a standard format to communicate with the software vendors, so as to acquire the detailed descriptions of the software components.

Fourth, dynamic configuration is another problem. A user may want to change its service configuration when the software is running. In the traditional context, a user logs into the server, and changes the service template in the server, then save it. It may become effect immediately or after reboot. But in the context of NFV, a user's virtual function may be installed in many virtual machines. It gets too complex if we let the user maintains the installed virtual machines information and logs into each virtual machine to reconfigure the service template one by one. A centralized service template configuration modification is much more easier. The control center may be or not be aware of the meaning of these dynamic configurations, But it needs to know that this is a configuration file and the range VMs that it applies to.

#### 4. Scope for Standardization

- (1) An interface between user and NFV control center for software installation configuration, about the components.
- (2) An interface between software vendor and NFV control center for the detailed description of the software components.
- (3) An interface between user and NFV control center for dynamic configuration.

The data model of the service components is the key point for the standardization.

#### 5. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

#### Author's Address

Haibin Song  
Huawei

Email: haibin.song@huawei.com