Abstract

This document describes the architecture of remote service installation and configuration in the provider’s network through a centralized management system. This is a typical scenario for virtual network function installation and dynamic configuration in network function virtualization (NFV) context. It is also a typical scenario in cloud computing environment where end users do not have to install applications on their end hosts, but can install their own featured powerful software application in the cloud. This specification describes an architecture of virtual network function installation and configuration. It is also applicable to other use cases with similar requirements.

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 March 27, 2014.

Copyright Notice

Copyright (c) 2013 IETF Trust and the persons identified as the document authors. All rights reserved.
This document describes the architecture to solve the problems described in [I-D.song-opsawg-virtual-network-function-config], in virtual network function configuration. The network function virtualization (NFV) controller is a broker for various software vendors. It manages how to install a virtual network function in the provider’s network according to user’s requirements. There needs standard protocols between user and NFV controller to describe the components requirements and resource requirements for a (or a batch of) virtual network function installation and configuration. There also needs a standard protocol between software vendor and NFV controller to describe the software basic information, running environment resource requirements, components information description. The component of a virtual network function may be another virtual network function. And the NFV controller may combine some virtual network functions together to form another virtual network function. If layer N (N can be 1, 2, 3...) virtual network function is consisted of several layer N-1 virtual network functions, the NFV controller must communicate with the user on how to construct the upper layer VNF with lower layer VNFS, and what are the
forwarding sequences and what are the conditions to trigger that forwarding sequence. The resource requirements from the user can be either explicit or implicit. If the resource requirement is "on-demand", then the NFV controller has to communicate with the virtual network function layer to create new service instances when the service load is increasing or delete extra service instances when the service load is decreasing. The resource requirements for each service instance is based on the recommendation from the software vendor.

This specification gives a basic management architecture for virtual network function (VNF) configuration. It lists the main functional modules that plays important roles in VNF configuration, and specifies what each functional module does, and what are the contents communicated among these functional modules, as depicted in Figure 1. The main concepts from this document are from ETSI NFV ISG [NFVE2E]. But there is slightly difference from ESTI. For example, we have a central NFV controller in this architecture that contains OSS/BSS, VNF management and Infrastructure management, unlike ESTI NFV ISG, where it separates the OSS/BSS and a management and orchestration system. In real world implementation, these functional units usually reside in the same physical equipment(s) called network management system.

OSS/BSS (operations support system / business support system) functional module is responsible for communication with users. It provides software (VNF) provider’s information to the users, and get components requirements and resource requirements from the users. It also communicates with VNF software vendors to get the software information and the recommended resource information, and etc.. VNF management module is responsible for VNF instance creation/deletion, VNF status management, VNF resource management, and VNF connection management when one VNF is represented as a conditional/static connection of multiple VNFs. Infrastructure management module is responsible for physical layer resource (CPU, memory, storage, bandwidth etc.) allocation for the virtual machine that a VNF is resided on.

This document will describe the basic principle for the architecture first, then give the description of the interfaces and what contents are exchanged among those interfaces. At the end, it will discuss the security threats for the VNF configuration.
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 from the NFV end to end architecture [NFVE2E].

User: a person, home or an enterprise customer who installs their own virtual network function (such as virtual enterprise firewall, virtual home gateway) in the provider’s network. A user can also be the provider’s network system administrator who want to install a general virtual network function (such like the carrier grade network address translator (NAT) ) that can server various customers inside the provider’s network.

NFV: network function virtualization. NFV technology uses the commodity servers to replace the dedicated hardware boxes for the network functions, for example, home gateway, enterprise access router, carrier grade NAT and etc. So as to improve the reusability, allow more vendors into the market, and reduce time to market. NFV
architecture includes a NFV controller (orchestrator) to manage the virtual network functions and the infrastructure resources.

NF: A functional building block within an operator’s network infrastructure, which has well-defined external interfaces and a well-defined functional behavior. Note that the totality of all network functions constitutes the entire network and services infrastructure of an operator/service provider. In practical terms, a Network Function today is often a network node or physical appliance.

vNF: virtual network function, an implementation of an executable software program that constitutes the whole or a part of an NF that can be deployed on a virtualization infrastructure.

VM: virtual machines, a program and configuration of part of a host computer server. Note that the Virtual Machine inherits the properties of its host computer server e.g. location, network interfaces.

3. Architecture

3.1. Design Principals

There are five key modules in the architecture. They are controller, software vendor, user, VNF, and infrastructure. And the controller has three key functional modules, OSS/BSS, VNF management, and infrastructure management.

(1) Controller is the brain of all operations. Controller creates virtual network functions in the provider’s network. Although the users may communicate with the virtual network functions in the service layer directly, but it has not to directly communicate with the VNF in the management level. A user can have multiple VNFs in the provider’s network, and can configure them or a subset of them through a simple communication with the controller. The controller is also a broker for various software vendors. It provides standard interfaces to convey information model with software vendors and users. It also monitors the VNF resource consumption status when the resource consumption model is "on-demand".

(2) Controller MUST NOT be aware of the service logic of each VNF. When a user configures multiple VNFs by sending a configuration template to the controller. The controller does not understand the contents in the configuration template. It only has to know how to apply the configuration to the appropriate VNFs.
(3) The key point for the protocol development SHOULD focus on the information model to describe the VNF itself, the resource requirements, the service/forwarding graph, and the status report.

3.2. User-controller

User communicates with the OSS/BSS module of the controller, to query, buy, or configure VNFs. User gets the VNF list from the query interface. Each VNF in the list contains: VNF type, function description, description of components that can be installed optionally, installation resource requirements, possible performance standard (which may include capable number of users per instance, throughput, concurrent connections, and etc.), and pricing information. User can select the appropriate VNF to install from the list. The information that the user can provide to the OSS/BSS before the installation can be the VNF name, quantity, the preferred components of the VNF, the preferred installation locations in high level (user do not have to know the detailed location such like on which virtual machine), and the possible performance requirement at the beginning (which will be considered when allocating resources, so that even the same VNF provided by the same software vendor can have different initial resource allocation when it is installed to serve different customers that have different amount of users/traffic). User should also tell the controller whether it needs specific resources or it only needs resources on-demand. In the meantime, user should also provide the data forwarding graph to the controller, so that the controller can figure out how the new VNF is connected to the existing network, and how the data flow should forward after the installation of this new VNF. Such forwarding graph should be written in a fixed format in spite of the difference of users. User can get the relative report from the controller, the report contains information of which VNFs have been installed, the status of each VNF, the number of VNF instances, logging information, and accounting information. User can also configure its VNFs from the controller, the details of the configuration are relative to the specific VNF vendors. User needs to specify the ID of which VNFs the configuration file is used to configure. When update package is released by the vendor, possible procedure may also need for the OSS/BSS to inquire the user whether he would like to update the VNF of his own. The protocol between the user and OSS/BSS module of controller could be a new protocol or an extension of existing protocols such like HTTP, NetConf, YANG or XML.
3.3. Software Vendor-Controller

Software Vendor communicates with the OSS/BSS module of the controller, to publish a VNF, update a VNF, off-the-shelf a VNF. To the details, if a software vendor sends a request to publish a VNF, the identity of the software vendor must be verified before any further action. The software vendor needs to provide the software description information and the software package itself. The description information should include the following: VNF name, classification (classification types can be provided by the controller to the software vendor in advance), function description, components description, installation environment description (operation system, CPU, memory, storage and etc.), and the capacity description under the recommended installation environment (for example, the capable number of users per instance, throughput, concurrent connections, and etc.), as well as the pricing information if needed. Please note that the software package can be submitted to the OSS/BSS together with the description information, or be provided from some out-of-band methods. For example, the software vendor can provide a URL where the OSS/BSS can get this VNF package, or the OSS/BSS provides a URL where the software vendor can upload its package. The software vendor should provide the VNF name and the update package to the controller when updating a certain VNF. Modification of functions, components, and the other basic information should also be provided to the controller. The protocol between software vendor and the OSS/BSS module of controller could be a new protocol or an extension of existing protocols such like HTTP, NetConf, YANG and XML.

3.4. Controller-VNF

Another important role for the controller is the VNF management. VNF management module collaborates with the OSS/BSS and infrastructure management module for the VNF creation, deletion, update, monitoring, scale-out/scale-in, and software configuration. OSS/BSS receives the VNF request from the user, and then invokes its interface with the VNF management module to create VNFs. VNF management module gets VMs with requested resources from the infrastructure module, or directly gets the physical host resources. And then it mounts operation system on the VM or the host, then installs the customized VNF(s), sets up the forwarding rules between VNFs according to the forwarding graph provided by the user. VNF management is also in charge of VNF update when update request is made by the vendor. When update request is agreed by the user, the OSS/BSS invoke the update procedure of the VNF management to install the update version VNF(s) on the corresponding VM. Each VNF will report its status to the VNF management module, such like the CPU, memory, storage usage, current traffic volume information, the link usage on the forwarding graph.
User can configure his VNFs through the OSS/BSS interface, but the configurations are finally carried out by the VNF management module. VNF management module can automatically scale-out or scale-in the number of VNF instances according to the load information on the current VNFs. For example, when a user's resource model is "on-demand", then when the VNF management module finds a relative VNF is beyond the load threshold, then it creates another same VNF to offload the relative VNF. And if the VNF function is process network traffic for the user, then VNF management module needs to coordinate with the NFV infrastructure to configure the relative switches (including soft switches), routers (including soft routers), so as to make traffic be divided to different VNFs. And user SHOULD be agnostic of this traffic division operation.

The protocol between the VNF management module can be a new protocol or the extension of existing protocols, such like HTTP, NetConf, and YANG.

3.5. Controller-Infrastructure

The third module in the controller is the infrastructure management module. It manages the infrastructure resources that the VNFs are using. It can create, delete, query about VMs. It configures the underlying network, including IP address, VLAN, ACL rules and flow tables. It can utilize OpenStack, CloudStack for some network management system similar to that. Infrastructure management can use the existing open interfaces. Infrastructure management also configure the network for the forwarding graph among the VNFs.

4. Security Considerations

Because VNFs are running in the provider’s network, so the privacy concern should be the most important aspect that a user would think about. The operator of NFV must guarantee that no third party can access the user’s VNFs or any information of the VNFs without the user’s permission.

VNFs are generally considered to be run on cloud computing environment, so the security threats to a cloud computing system are also applicable here.

5. IANA Considerations

There is no IANA considerations in this document.

6. References
6.1. Normative References


[I-D.song-opsawg-virtual-network-function-config]

6.2. Informative References


Authors’ Addresses

Hong Zhou
Huawei

Email: zhouhong@huawei.com

Haibin Song
Huawei

Email: haibin.song@huawei.com

Fu Qiao
China Mobile

Email: fuqiao@chinamobile.com