International Journal of Engineering and Technical Research (IJETR)
ISSN: 2321-0869 (O) 2454-4698 (P), Volume-5, Issue-1, May 2016

Orchestration and Life Cycle Management of Virtualized Network Functions

Sharath Kumar V

***Abstract*— Network functions virtualization separates the software from hardware and runs it on commodity hardware like servers, storages and switches. Virtualization abstracts the details of physical elements and enables the logical detachment between execution environment and infrastructure. The disassociate exposes a new set of entities, the virtualized network functions, and a new set of relationships between them and the NFV Infrastructure .VNFs can be chained with other VNFs and physical network functions to realize a Network Service. This paper describes the life cycle management of VNF instances and how resources are allocated in NFV infrastructure to using constant and continuous orchestration and management functions.**

***Index Terms*— NFV, Virtualization, Orchestrator, Cloud computing, VNF, VNFM, VIM, Openstack.**

I. INTRODUCTION

NFV is a modern way to define, create and manage networks by replacing committed network equipment's with software and automation. It is a continuation of IT practice away from visible hardware that's inflexible, proprietary and expensive.

In an NFV environment, a virtualized network function (VNF) takes on the responsibility of managing specific network functions that will run on one or more virtual machines (VMs), on hypervisors, or in containers, on top of the physical infrastructure.

Virtualized network functions will range starting from mobile deployments, where mobile gateways (e.g. serving gateway, packet data network gateway, etc.) and related functions (e.g. Multimedia equipment, Home location register, PCRF, etc.) are deployed as VNFs, to deployments with "virtual" customer premises equipment (CPE), burrowing gateways (e.g. Virtual private network, gateways), to test and characterize equipment (e.g. to monitor service level agreements).

The benefits of NFV is derived from the fact that it will run on common purpose industry servers and switches in virtual machines and is built with standard open APIs. NFV depends on open source development and offers a vast range of networking capabilities which grows dynamically and adaptively.

Sharath Kumar V, Department of Electronics and communication Engineering, SJBIT, Bangalore, Karnataka, 560060, India

In general, the aim of NFV is to offer adaptability, flexibility, and simplicity. The exact and comprehensive operational and technical benefits that network operators gain from NFV implementation include:

- Reduced equipment costs and reduced power consumption through integration of equipments and exploiting the economies of scale of the IT industry.
- Network flexibility via programmatic tools
- Taking advantage of the open source pace of revolution and ever-emerging improvements in both the telecom and the traditional IT space
- Extensive choice of modular drivers and plugins
- Convenience and connectivity via API, enabling faster deliverable time for new capabilities
- Lower costs by replacing with commercial off the shelf hardware, better price/performance
- Space utilization
- Operational efficiency across datacenters through orchestration by managing thousands of devices from one aid
- Visibility: automated tracking, troubleshooting and actions across physical and virtual networks and devices
- Increase performance by raising network device utilization
- Service level agreement driven resource allocation
- Quality of service: performance, scalability, footprint, resilience, integration, conformability
- Policy-driven noticeability
- Application based infrastructure support

The OpenStack platform will provide the premise for the NFV architecture, which is fundamentally a cloud for deploying, orchestrating and managing virtualized network functions. OpenStack allows multiple datacenter management from unified source, complete with common security, identity services, APIs, and user interfaces. OpenStack is a software that handles a large pool of compute, storage, and networking resources throughout a datacenter.

II. SYSTEM OVERVIEW

VNF (Virtual Network Function)

A VNF is the most basic block in NFV Architecture. It is the virtualized network element. For example we call a virtualized router a router VNF another example is base station VNF.A sub function or block which is virtualized, then it is called VNF. For example in the case of router, each sub-functions of the router can be described as separate VNFs which together function as consolidated virtual router.

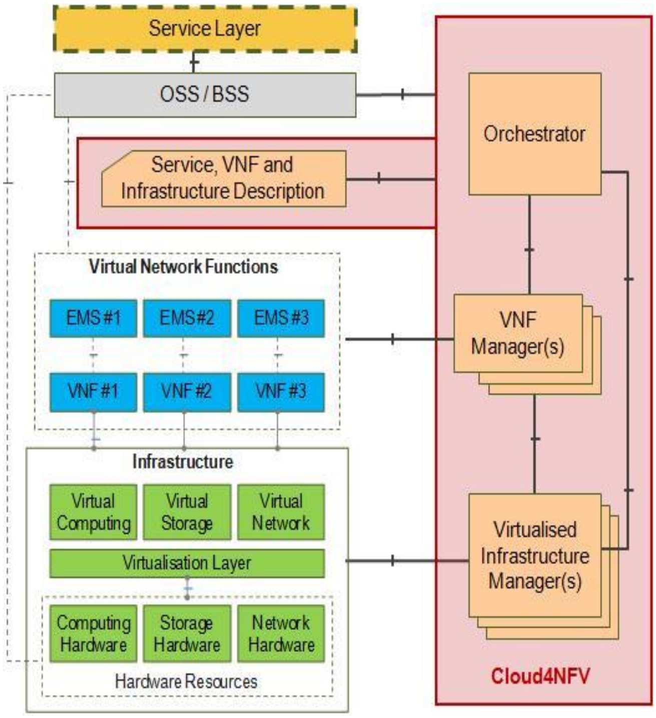

Figure 1: NFV architectural framework

A. *EMS (Element Management System)*

B. *This block is mainly responsible for the management of VNF operation, in the same way as visible network elements are managed by their respective EMS of VNFs.This may handle the VNFs through correct interfaces. There may be one EMS for each VNF or an EMS can manage multiple VNFs. EMS itself can be a VNF.*

C. *NFVI (Network Function Virtualization Infrastructure)*

D. *NFV Infrastructure is the environment in which VNFs will run. This includes visible resources, virtual resources and virtualization layer.*

Compute, Memory and Networking Resources

This is the physical part in NFV Infrastructure. Virtual resources are instantiated on these visible resources. Any standard switch or visible server/storage server belongs to this category.

Virtual Computing, Virtual Memory and Virtual Networking Resources

This service is the virtual part in NFV Infrastructure. The visible resources are hidden into virtual resources that are utilized by VNFs.

Virtualization Layer

This layer or block is responsible for abstracting visible resources into virtual resources. The common and standard term for this layer is Hypervisor. The function of this layer is to decouple software from hardware which enables the software to progress independently from hardware.

In a scenario where there is no virtualization layer, one may think that VNFs can run on visible resources directly; however by the definition of VNF we cannot call them VNF nor it would be part of NFV architecture. They may properly be called PNFs (Physical Network Functions).

OSS/BSS (Operation Support System/Business Support System)

Operation support system compacts with network management, fault management, configuration management and service management. Business

Support system compacts with customer management, product management and order management etc.

Orchestrator

The key responsibilities of NFV Orchestrator:

• Fulfilling the Resource Orchestration functions and the orchestration of NFVI resources across multiple VIM instances.

• The lifecycle management of Network Services, fulfilling the Network Service Orchestration functions.

The two responsibilities are kept within one functional block and sharing a common interface represented by the NFV Instances and NFV Resources repository. To support different multi-vendor deployments and/or different mappings of functionality to Administrative Domains, the two responsibilities may be separated in future. The NFVO uses the Network Service Orchestration functions to coordinate groups of VNF instances as Network Services that jointly realize a more complex function, including joint instantiation and configuration, configuring required connections between different VNFs, and managing dynamic changes of the configuration, e.g. for scaling the capacity of the Network Service. The Network Service Orchestration function uses the services exposed by the VNF Manager function and by the Resource Orchestration function.

VNF manager (VNFM)

The VNF Manager block is mainly responsible for the lifecycle management of VNF instances. Each VNF instance is expected to have a related VNF Manager. A VNF manager can be assigned with the task of managing a single VNF instance, or multiple VNF instances of the same or different types. Most of the VNF Manager functions are assumed to be general common functions applicable to any type of VNF. However, the NFV architectural framework must also support the cases where VNF instances need particular functionality for their lifecycle management and such certain functionality may be specified in the VNF Package.

Virtualized infrastructure manager (VIM)

The virtualized Infrastructure Manager block is responsible for controlling and managing the NFV infrastructure compute, storage and network resources, usually within the ever operator's Infrastructure Domain (e.g. all resources within an NFVI-PoP, resources across multiple NFVI-POPs, or a subset of resources within an NFVI-PoP). A Virtualized infrastructure manager may be specialized in controlling a certain type of NFVI resource (e.g. compute-only, storage-only, networking-only), or may be capable of managing multiple types of NFV Infrastructure resources (e.g. in NFVI-Nodes).

Implementation

Virtualized network functions are managed by VNF manager (VNFM).The VNFM is responsible from the onboarding of VNF template package to their termination or deletion. Network service package describes the link between the VNFs and the link between VNF and PNF.Network service also describes the link between multiple VNF's associated with a network service.

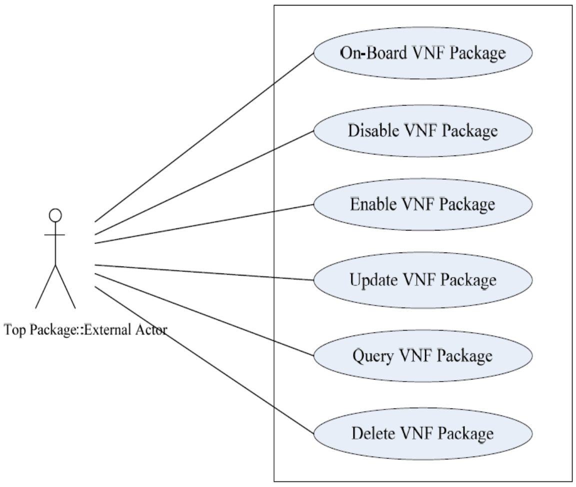

Figure 2: Diagram for VNF package on-boarding

- VNF Package on-boarding refers to the process of submitting VNF Package in the form of XML to the NFVO to be included in the Catalog.
- Disabling a VNF package refers to the process of marking a VNF Package as disabled in the catalogue, so that it is not possible to instantiate VNFs with it any further.
- Enabling a VNF Package refers to the process of marking a VNF Package as enabled in the catalogue, so that it can be used to instantiate VNFs again.
- Updating VNF Package refers to the process of submitting a modified VNF Package to the NFVO to be included in the catalog.
- Querying VNF Packages allows returning from the catalog the information of the VNF Packages.
- The final step is deleting a VNF package.

III. RESULTS AND DISCUSSION

We analyze the implementation of NFV on a virtual machine platform and open switches are used for multiple VNFs chaining.

Performance of VNFs can be evaluated on two parameters:

a) Throughput

b) Latency

NFV is deliberately based on openstack cloud computing technologies.

- VNFs executable in virtual machines or hypervisors
- Paths which are implemented through Switches

The figure shows the throughput for single VNF chain and multiple VNF chain and the latency obtained on a virtual machine (VM).

Single Chain output

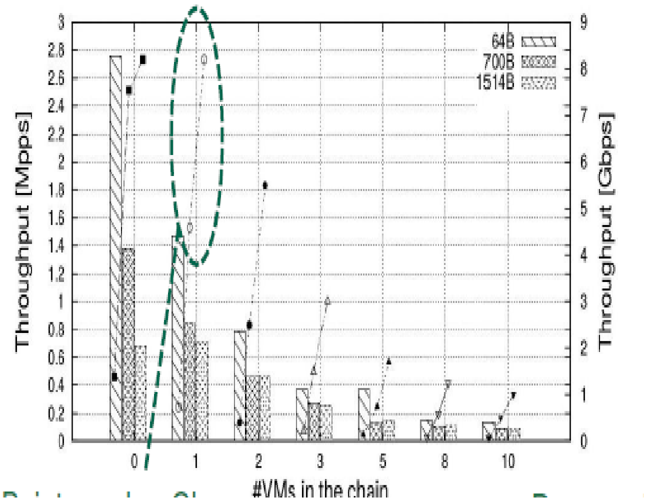

Figure 3: Single VNF throughput obtained with virtual machine

Multiple chain output

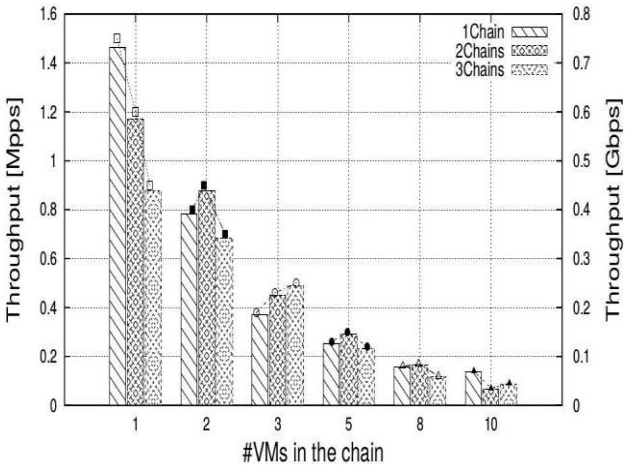

Figure 4: Multiple VNF Chain throughput obtained with Virtual machine

Latency

As per the throughput, the latency also worsens sensibly when exceeding the CPU Cores.

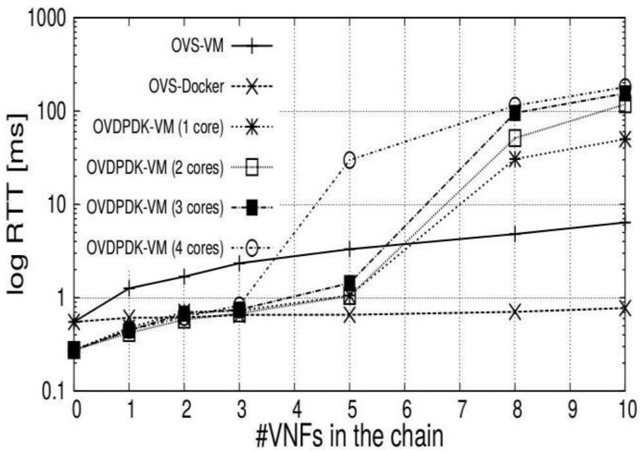

Figure 5: Latency obtained with virtual machine

CONCLUSIONS

Virtualized network functions provide an effective way to deploy, operate and orchestrate network services with much less capital and operational investment cost.

Network operators can easily deploy any network service without the need to install the hardware equipment. This provides a flexible and adaptable manner to deploy network service in the network service provider's server or datacenter.

In this paper we discuss how to implement virtualized network functions, and the resources needed for a network service to be deployed.

ACKNOWLEDGMENTS

This paper is made possible through the guidance and support of many people. First and foremost; I would like to thank Dr.K.R.Nataraj, head of our department and Bhaskar.B, Associate professor for their support and encouragement.

Finally, I sincerely thank to my parents, family, and friends, who gave me the advice and financial support.

REFERENCES

1. ETSI, Network Functions Virtualization (NFV): Use Cases, Technical Report ETSI GS NFV 001 v1.1.1, Oct. 2013.
2. ETSI, Network Functions Virtualization (NFV): Architectural Framework, Technical Report ETSI GS NFV 002 v1.1.1, Oct. 2013.
3. Assessing the Performance of Virtualization Technologies for NFV: Preliminary Benchmarking ,Fulvio Risso ,Politecnico di Tornio –EWSDN 2015
4. J. Carapinha, et.al. "D2.1: Reference Scenarios and Technical System Requirements Definition", Mobile Cloud Networking FP7 project, April 2013.
5. G. Xilouris1, et al., "T-NOVA: A Marketplace for Virtualized Network Functions", European Conference on Networks and Communications (EUCNC), June 2014.
6. Batalle, J.; et al., "On the Implementation of NFV over an OpenFlow Infrastructure: Routing Function Virtualization," Future Networks and Services (SDN4FNS), IEEE SDN, vol., no., pp.1,6, 11-13 Nov. 2013
7. J. Soares, C. Gonçalves, et.al, "Towards a Telco Cloud Environment for Service Functions", submitted to IEEE Communication Magazine.
8. H. Puthalath, J. Soares, et al., "Negotiating On-demand connectivity between clouds and wide area networks," IEEE CloudNet, Paris, 2012.

Mr. Sharath Kumar.V, Master in Digital communication and networking from SJB Institute of Technology, Bangalore, Karnataka