Whitepaper

Simplify Your Network Infrastructure with Network Functions Virtualization

Author: Krishna Kunapuli

Contents

Abstract

When it comes to networking equipment, traditional hardware architectures are growing too diverse, making the operation of service additions and upgrades increasingly difficult for service providers and data center operators. The NFV approach helps service providers simplify operations by virtualizing network functions previously performed by proprietary hardware. This paper discusses how virtualization of network functions also provides the ability for dynamic service chaining, resource allocation, and scale-in or scale-out.

Introduction

Network Functions Virtualization (NFV) was originally conceptualized in early 2000s through a collective effort of service providers and equipment manufacturers as a technology for replacing physical network devices with virtual devices that could be deployed on demand. Networks had a higher number of devices, which were becoming larger in size with extra space and power requirements. There was a need for more ports with higher port speeds with an increasing number of IP-enabled endpoints such as peripherals, phones, surveillance cameras, etc. To cater to the service provider transport networks, manufacturers had to come up with multi-chassis devices with terabit throughput. In addition, an increase in the number of devices resulted in more complex cabling requirements in equipment rooms, which became unmanageable. The growing need to scale out led to the origin of NFV.

Even today, global networks are disrupted constantly to accommodate newer connectivity requirements, changing application architectures and data demands of an organization. This is driven by the increased number of connected devices within an enterprise due to BYOD and seamless mobility requirements, a higher number of end-user applications and rich audio-visual content, and the collaboration requirements of the smart workplace. There is also a greater need for ubiquitous access to SaaS

applications over Internet due to the global nature of a flexible and disparate business needs. Network Functions Virtualization (NFV) is one such major disruption that is enabling digital transformation by removing the rigidity associated with traditional networks.

This whitepaper considers the problems in traditional networks and challenges posed by ever-increasing data demands. It also looks at the origin of an Enterprise NFV, and how the standards bodies and the industry are working to establish management frameworks that make interoperability possible. It further introduces concepts such as Service Chains and gives examples of common use-cases for NFV. Finally, it lists out important criteria for choosing an NFV solution and gives an example of a turn-key NFV solution available in the market.

Exponential Increase in Data Demand

As enterprise users started consuming data rich applications, including real-time video for collaboration, the demands put on network became significantly higher. FastEthernet (100Mbps), which was the highest access switchport speed for a long time, quickly became obsolete as networks moved to 1 gigabit or higher speeds in the access layer. Multi-gigabit network backbones became possible with improvements in silicon packaging densities and availability of high speed fibers. This bandwidth eventually trickled down to the access layer.

Today we have **10 gigabit ports** available on almost every shipped network equipment. With the advent of **Wi-Fi6** we can move **1 gigabit** per stream. But handling such large data volumes requires more expensive silicon on each equipment. There needs to be a better' way of segregating this traffic in software, with fewer interconnections between the devices, to reduce the overall cost of the infrastructure.

Network Device as a VM – The Origin of Enterprise NFV

By 2005, OEMs started manufacturing software versions, or virtual machines (VMs), of their hardware devices. These were called Virtual Network Functions (VNFs), and the earliest commercially available network functions that gained mainstream attention were the Cisco Cloud Services Router 1000v, Juniper virtual SRX, and Riverbed virtual Steelhead, among others. These VNFs could be installed as VMs on commodity servers. The opensource community also embraced NFV quickly. There were some opensource projects, such as the VyattaOS, which furthered the development of VNF as a concept.

However, there was the problem of managing these appliances and sending traffic to and from these virtual appliances for different flows. This was also a big shift away from hardware for network OEMs. Adoption was slow as administrators were not confident of these experimental appliances with unclear management strategies.

Introducing MANO - The Need for Standardization

Telcos and OEMs were instrumental in the development of NFV due to their participation, along with ETSI, in creating standards that ensured interoperability across the industry. ETSI was influential in developing the Management and Orchestration architecture for NFV (MANO), which is widely accepted across the industry and now is an opensource project called the Opensource MANO. MANO is key in choosing any NFV architecture and defines what an NFV deployment should contain at a minimum, i.e. an NFV Orchestrator, a VNF manager, and a Virtual Infrastructure Manager. MANO compliance is key in choosing any NFV architecture.

MANO and Service Chains

MANO introduced a simple concept called Service Graphs that are hop by hop connections between different network functions. These are analogous to cables in the physical world. A service graph delivered a meaningful end-to-end network service by interconnecting different network functions, or what we can call a service chain. With the increase in dynamic on-demand applications and connectivity needs, enterprise users needed access to specific network functions for a given amount of time. Each such scenario required a service chain, as permanent connectivity and rigid application architectures started becoming obsolete.

Let's look at an example. Users of a specific line of business (LoB) want to test a new partner solution and to achieve this, need secure access to the partner's datacenter. This access has to be restricted to a particular user group and the partner data needs to be stored in redundant server clusters.

To implement this requirement, the LoB needs a firewall, a load balancer, access to server farm, the switching layer to carry the VLANs and a router-to-route the traffic to a partner datacenter. This connectivity requires the laying of cables that will only be required until the activity lasts, i.e. a few weeks. The cost involved can be avoided if NFVs were used.

Here is how an NFV implementation of this requirement would look like. A VNF manager would deploy one VNF instance/one configuration item for each for the functions. It would deploy these VNFs on the virtualized compute and storage provided by the Virtual Infrastructure Manager. It would also inform the NFV orchestrator about how these VMs can be accessed such as IP address/authentication parameters, etc. The NFV orchestrator would take the necessary VLAN and IP subnets needed for this service from a dynamic pool of available addresses and provision the same on the devices after logging into them. It would present the service with a name and unique identifier to the business user, who can monitor the service using APIs available on the orchestrator.

NFV Use Cases

Use Case 1

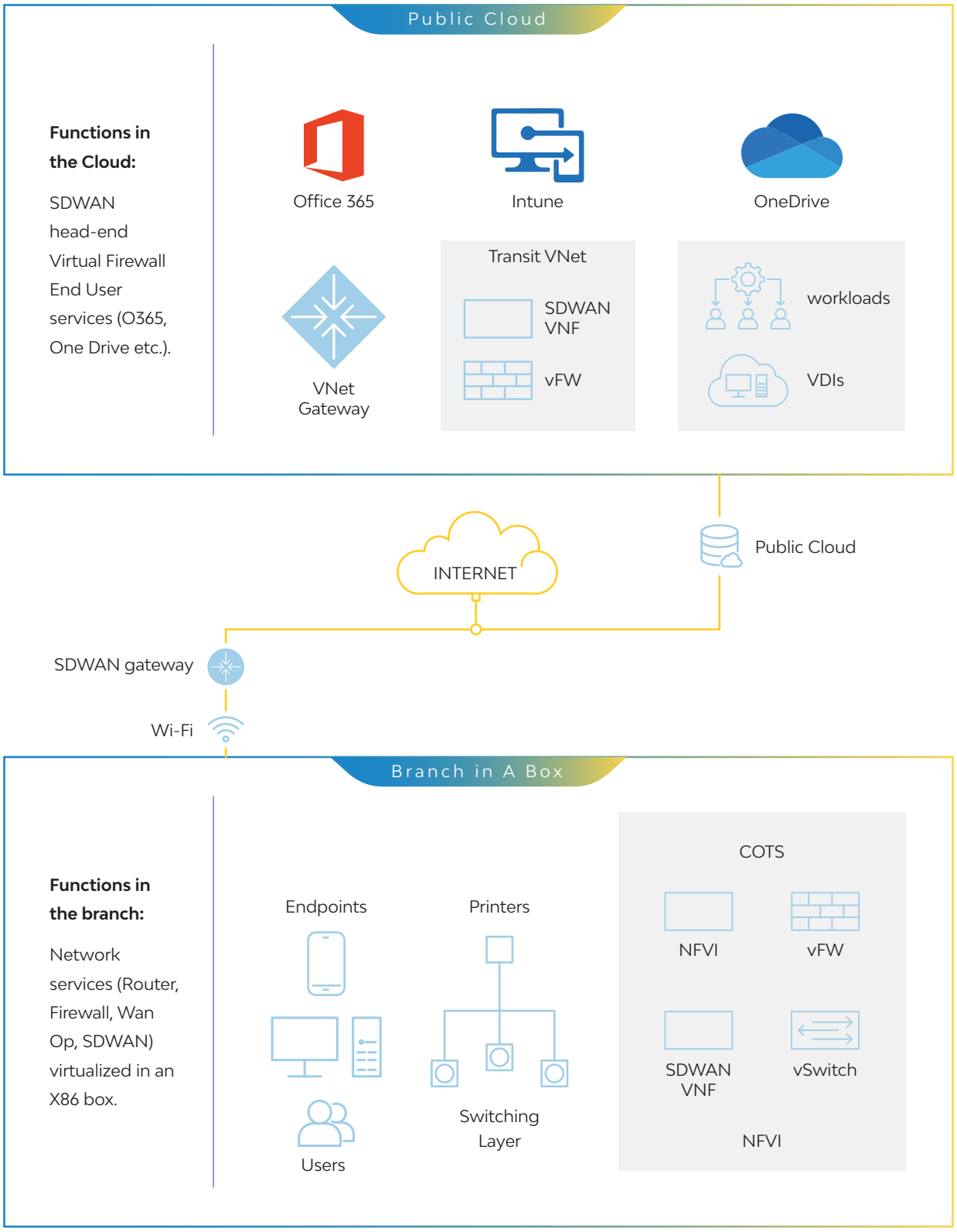
Branch-in-a-Box – Virtualized Small Branches

A common use-case for Enterprise NFV is plug-&-play branches that can be configured in a few minutes and would take not more than 3-4 RU of rack space for realizing the network functions.

Below are the key business and operational drivers for this use case:

- Lean hardware with COTS servers.
- On-demand provisioning of network resources.
- Single pane of glass for Network Management.
- Network functions realized in software reducing power needs.
- Switching in software for intelligent forwarding decisions.
- Zero-touch deployment.
- Savings in time, cost, and resources for implementing new branches.
- Dynamic scale up and down in functionality in a few clicks.

Below is an illustration of a typical Branch-in-a-box solution using NFV. In this example, a COTS server in the branch is used to implement network functions such as Firewall, SDWAN, and Wan Optimizer in software.


Public Cloud
Functions in the Cloud:
SDWAN head-end Virtual Firewall End User services (O365, One Drive etc.).
Office 365
Intune
OneDrive
VNet Gateway
Transit VNet
SDWAN VNF
vFW
workloads
VDIs
INTERNET
Public Cloud
SDWAN gateway
Wi-Fi
Branch in A Box
Functions in the branch:
Network services (Router, Firewall, Wan Op, SDWAN) virtualized in an X86 box.
Endpoints
Printers
COTS
NFVI
vFW
SDWAN VNF
vSwitch
NFVI
Switching Layer
Users

Use Case 2

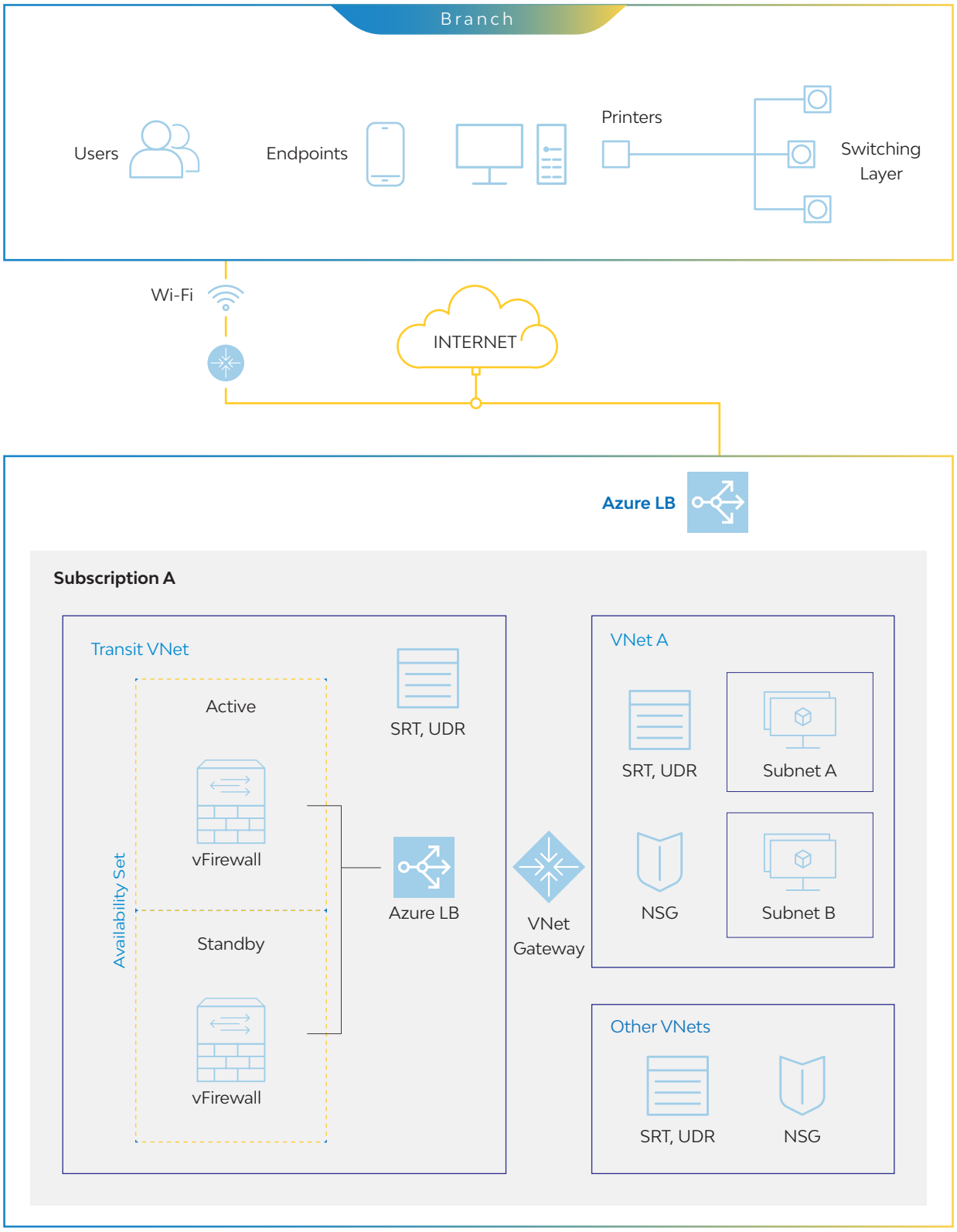
Cloud Edge Gateway VNFs

Another use-case for Enterprise NFV is deploying VNFs as Cloud Edge Gateways. This is becoming a necessity as customers want some flexibility in choosing features that are not available in native Cloud solutions for VPN connectivity and Security. While it is likely that more features will be added to cloud- native solutions, such as the Transit Gateway in AWS, there will always be customers with a bespoke need that is satisfied by a commercially available VNF that can be deployed in a few minutes.

Below are the key business and operational drivers for this use case:

- VNF with the best feature set can be selected.
- Additional layer of security on top of native security on Cloud.
- UTM and DPI firewall features.
- Cloud-integrated Enterprise WAN.
- Consistent policy across enterprise irrespective of location.
- Reduced need for private WAN connectivity to cloud such as Direct Connects.
- Quick deployment times.
- On-demand scale out (more VNFs can be added/throughput can be purchased).

Below is an illustration of a typical Branch-in-a-box solution using NFV. In this example, a COTS server in the branch is used to implement network functions such as Firewall, SDWAN, and Wan Optimizer in software.


Branch
Users
Endpoints
Printers
Switching Layer
Wi-Fi
INTERNET
Azure LB
Subscription A
Transit VNet
Active
SRT, UDR
vFirewall
Availability Set
Azure LB
Standby
vFirewall
VNet Gateway
VNet A
SRT, UDR
Subnet A
NSG
Subnet B
Other VNets
SRT, UDR
NSG

Elements of an NFV Solution

A good NFV solution should have good integration between NFVO, VNFM, and VIM, with the exception of handling routines. Each component of the solution is managed by its own manager and they should intercommunicate to handle exceptions and to perform reconciliation.

Some of the important operational considerations for any NFV solution are:

- How will my team be alerted if there is a service disruption?
- What happens when my VNF crashes - can I recover its configuration?
- Can I spin up a new VNF in time with minimal service disruption?
- What happens when my capacity limits are exhausted on the server hardware?
- What happens if my VNF's logical throughput limit is exhausted?
- How can I ensure that I can dynamically increase capacity while my service is up and running?
- How can I create a new service chain or delete an existing one?
- How can I secure my NFV deployment?

Choosing the Right NFV Solution

It is recommended to select the components of an NFV solution based on the merits of each function rather than a turn-key solution from a single OEM covering all components. Typically, enterprises that have a DevOps-centric operational approach are better suited to NFV deployments as they can ensure better integration between the different components and can also better able to utilize the breadth of capabilities available via REST API on the products available in the market.

When it comes to choosing VNFs, preference must be given to VNFs that have:

- High available architecture (redundant cores/ redundant design with 2 or more VNFs)
- Standards-based data-ingestion and modeling (e.g. TOSCA/YANG)
- Open API
- Scalable for high throughput
- Good logging and reconciliation mechanisms
- Call home features
- Zero-touch provisioning features
- Support for multi-tenant environments
- Support for multiple hypervisors
- Support for containerisation
- Flexible configuration knobs to support variety of use cases
- OEM ecosystem for VNF development
- Readily available cross VNF integrations (VNF to VNF compatibility)

A Peek Into Cisco's NFV

Some organizations may prefer a turn-key approach to NFV using a single OEM solution. That's because it helps bootstrap the operations quickly even with a little knowhow about the technology. Here is a description of an E2E NFV solution using Cisco's Enterprise NFV portfolio.

Below are the solution components:

NFVI	VIM	VNFM
NFV Infrastructure (hardware-agnostic)	Virtual Infrastructure Manager (VCenter)	VNF Manager (Cisco Elastic Services Controller)

ESC will be used as the VNF manager and will integrate into VCenter which acts as the VIM. The solution is compute-hardware agnostic. ESC will be deployed as a VM (HA supported).

The components of ESC are as follows:

- **Core engine:** provides the central VNF life cycle management functions of ESC. In addition, it handles duties, such as applying policy from higher layers in the orchestration stack (VNF placement, start-up order, etc.), coordinating and tracking multi-step and/or multi-VNF life cycle requests, and a database-style ability to implement, roll back, and resume transactions.

- **MONA:** provides sophisticated instrumentation and analytics of VNFs and includes a rules engine that triggers predefined or customer-defined actions based on metric thresholds and life cycle stage.

- Beyond these two key components, ESC also has components to monitor ESC for HA, a logging module, and a ConfD module for northbound NETCONF/YANG clients.

Conclusion

NFV for Enterprise is geared for big growth, both due to simplified architectures reducing the number of on-premise network devices and due to better suitability for dynamic and elastic applications. NFV is one of the most promising trends in virtualization, freeing enterprises from the limitations imposed by their existing infrastructure and hardware. The NFV approach is helping enterprises use virtualization to reduce the costs associated with managing and powering physical infrastructure. It is also better suited to cloud deployments as the network functions can easily be deployed as instances in public clouds, giving real flexibility not just in design, but also in consumption models. Though NFV is not a fix-all solution and comes with its own set of challenges, such as network stability and security, a lot of these risks can be mitigated by understanding the technology.

A top challenge for service providers is transitioning network devices to software functions. Provisioning resources in these environments is an area where NFV has a lot to offer as it allows businesses to update and configure software on demand without glitches. Ultimately, NFV aims to transform the way network operators' architect and operate their networks. Enterprises must have a clear NFV strategy with a view to reducing operations overhead and cost of doing business.

About the Author

Krishna Kunapuli

Sr. Solutions Architect, Cloud and Infrastructure Services, LTIMindtree

Krishna Kunapuli is a Network architect with LTIMindtree CIS (Cloud and Infrastructure Services), with 15 years of experience in design consulting and implementation. He has designed and deployed several Service Provider and Enterprise networks worldwide and has also participated in major technology events such as the SDN and NFV world congress. He is an early enthusiast of NFV and has over 6 years of experience in designing and deploying NFV solutions for Enterprises and Telcos.