Software-Defined Network Functions Virtualization (SDN & NFV)

This paper discusses two key networking trends: SDN and NFV. It shows how the two are not only complementary, but, when deployed together, are synergistic. Whereas NFV focuses on network platform virtualization, SDN is focused on network virtualization.

When deploying NFV platforms, the number of end points can grow exponentially due to the nature of the many virtual machines (VMs) being spawned. SDN can be used to provide the connectivity between these VM-based end points in a dynamic way.

Additionally, since NFV focuses on higher-layer (L4-L7) devices (e.g., network security devices and gateways), there is a recognized need to extend SDN-OpenFlow protocol to move beyond the current L2-L4 control and into the higher layers.

NFV: A Cross-operator’s Initiative

Network operators aiming at reducing their TCO while accelerating new service deployment, have been working on a new initiative termed Network Functions Virtualization, or NFV. The goal is, for example, instead of procuring a proprietary network appliance, the operator can deploy applications and services that can run on top of virtual machines (VMs) in a Commercial Off The Shelf (COTS) server architecture platform. Such applications are referred to as VNFs, or virtual network functions, and can be dynamically chained to form a multi-service platform.

NFV Value Proposition

The operator’s goal is to reduce Capex/Opex, and allow for faster services deployment, greater flexibility and differentiation, while having a low barrier to entry introducing new services for additional revenue.

In trying to capitalize on the virtualization gain achieved by the IT industry, NFV targets many network equipment applications, from residential gateways at the low-end to a high-performance firewall at the high-end, that can run on a range of industry-standard hardware, all of which require processing traffic at all 7-layers of the OSI protocol stack.

ETSI ISG NFV

NFV, part of ETSI, is an industry specifications group (ISG) aiming at fostering faster interoperable implementations, for a broad number of use cases, rather than necessarily spawning new standards activities.

Figure 1 below highlights the three functional working groups within the ISG NFV effort. The management and orchestration (MANO) WG is responsible for the management and orchestration of network services

This paper examines the architecture model of both SDN and NFV, discussing possible integration models, and addressing deployment scenarios to extend SDN-OpenFlow to support in-line (in the forwarding plane) L4-L7 networking services to better manage NFV platforms.

The paper concludes by highlighting Netronome’s strategy, value proposition and product portfolio in SDN and NFV.

---

Figure 1. High-level NFV framework. Three functional working groups: MANO, SWA and INF. Source: ISG Architecture and Framework document.
on NFV infrastructure. The software architecture WG (SWA) is responsible for the decomposition of the VNF into functional blocks and their requirements. The infrastructure WG (INF) is responsible for specifying the requirements of three domains: Compute (include storage), hypervisor and network.

**Emergence of SDN**

In the meantime, an orthogonal new concept has been gaining ground in the data center and WAN. Software-defined networking, or SDN, is about separating the control plane from the data plane, making the latter simple and fast, dealing mostly with the lower layers (L2-L4). The interface between the two is referred to as the southbound interface. The Open Networking Foundation (ONF) has defined the OpenFlow protocol as the standard southbound interface between the SDN controller and the OpenFlow switch.

The SDN controller has a bird’s-eye view of the whole network and can dynamically steer the traffic based on statistics gathered from the network. In addition, third-party applications, management and orchestration functions can run north of the SDN controller. In SDN, these applications are network-aware, and can communicate their requirements to the network via the SDN controller, resulting in dynamically changing the network topology, for example. At the same time, the application can monitor network state and adapt accordingly.

**ONF’s SDN-OpenFlow**

Figure 2 depicts the ONF-OpenFlow architecture model. In this architecture, the control layer is the reference point. ONF has defined a standard OpenFlow protocol, south of the controller. Business applications and networking services run north of the controller over a northbound interface, or NBI. The latter is only now starting to be standardized by ONF.

The infrastructure layer is where all the OpenFlow switches reside. Their main function is to forward data plane traffic based on match-action rules defined by the SDN controller. These switches are equivalent to switches and routers in today’s legacy networks.

**SDN-OpenFlow Supports L4-L7 for Best Synergy with NFV**

Today’s limited deployment of SDN-OpenFlow networks calls for higher-layer networking services to be north of the SDN controller. Although this model works for some higher-layer services, such as DDoS attack prevention, it falls short for many other networking services, such as Firewall, load balancing and IPS/IDS.

In order to meet network scaling, low-latency and high-performance, some networking services need to be inserted in the data path or forwarding plane. Some of the challenges include:

1. Not every service can be run satisfactorily north of the controller due to performance, latency and scalability issues.
2. While some can, many other services can’t and need to be run inside an enhanced OF switch. Examples of such services that need to reside in the switch include Intrusion Prevention (IPS) and SSL Virtual Private Networks (VPNs).
3. In addition, legacy L4-L7 devices, such as load balancers and firewalls, can reside in the infrastructure layer (forwarding plane) where the controller can steer traffic based on some criteria to such legacy devices.
4. The above can be satisfied by extending the OF protocol, or devising a new protocol focused on L4-L7 while maintaining state information, but this will be up to the ONF’s technical WGs to decide.

**Adding L4-L7 Intelligence in SDN-OpenFlow**

Figure 3 depicts some of the deployment models to allow SDN-OpenFlow networks to effectively support L4-L7 services. There are many deployment models geared toward adding L4-L7 support in SDN-OpenFlow networks (see Figure 3):

1. Running as applications on the controller
   - Controller programs SDN switch on per-flow basis
2. Standalone network appliance
   - Traffic directed to appliance either based on static policy or dynamically driven by controller, or simply in-line with traffic
   - E.g., Firewall, Load balancer

![Figure 2. ONF’s SDN reference architecture](image)

![Figure 3. Adding L4-L7 support in SDN-OpenFlow networks.](image)
3. Full Layer 4-7 network services running on an intelligent switch
   • Intelligent switch becomes Layer 2-7 device
   • E.g., IPS/IDS

Note: Any network device in the infrastructure layer, including network security devices and gateways, are referred to as “switches.”

Combining NFV and SDN
Table 1 depicts how SDN & NFV are complementary. SDN can be used to interconnect the many VM-based end points in NFV platforms.

Figure 4 depicts a typical deployment scenario in the data center. The SDN fabric focus is on fast L2 / L3 forwarding, while the application servers, appliances and gateways are intelligent edge devices, optionally architected using NFV concepts (i.e., COTS server architecture).

Table 1. SDN and NFV are complementary and synergistic.

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SDN</strong></td>
<td></td>
</tr>
<tr>
<td>✓ Software-based architecture</td>
<td>✓ Fast OpenFlow switches (L2-L4)</td>
</tr>
<tr>
<td>✓ About network virtualization</td>
<td>✓ Centralized controller with bird’s-eye view of the network</td>
</tr>
<tr>
<td>✓ Network-aware applications</td>
<td>✓ Middleboxes and gateways (L4-L7)</td>
</tr>
<tr>
<td><strong>NFV</strong></td>
<td></td>
</tr>
<tr>
<td>✓ About network equipment virtualization</td>
<td>✓ Virtualized appliances (L4-L7 middleboxes and gateways)</td>
</tr>
<tr>
<td>✓ Uses COTS server architecture</td>
<td>✓ Network equipment implemented using server architecture</td>
</tr>
<tr>
<td>✓ SDN to control VM connectivity</td>
<td>✓ Offload / acceleration, when needed</td>
</tr>
</tbody>
</table>

NFV is about virtualizing appliances. SDN is about virtualizing the network.

From an architectural model perspective, SDN can be integrated into the NFV architecture framework model shown in Figure 5. The orchestration and the management applications can themselves be VNFs running on one or more VMs. In such cases, the OpenFlow protocol is part of the NF-Vi interface; or it can be part of the Vn-Nf interface.

NFV and SDN can be viewed as synergistic and complementary. When deployed together, network equipment can be built using virtualized COTS server architecture, while the overall network is controlled through SDN. The resulting network of NFV equipment can be dynamically reconfigured, based on applications and traffic patterns. In addition, the SDN controller can steer traffic from one network equipment to another if required. An example can be a legacy firewall that is dynamically inserted in the traffic path when network monitoring statistics show abnormality in the traffic.

Figure 5. This figure depicts the synergy between SDN and NFV in the data center.
Service Chaining: Example of SDN-NFV Working Together
With NFV, the server platform hosts the various network functions (applications and services) on its VMs. Some VNFs need a single VM, while others may require multiple VMs to meet their performance target. Allocating a number of VMs to a specific VNF is best done dynamically through a management and orchestration entity that can be either local or remote. In addition, the resulting large number of VM-based end points will need to be interconnected in a dynamic fashion based on network topology and network traffic.

One advantage of SDN-based NFV is simplifying the ability to implement service chaining of multiple VNFs on the same platform. The orchestration layer can dynamically define a series of services to be implemented on the virtualized platform based on intelligent information gathered from the network. An example of a chain includes a firewall followed by a load balancer, which, in turn, is followed by a WAN optimization function.

Netronome Strategy, Architecture and Products
Netronome offers integrated solutions for NFV & SDN, adhering to the Open COTS server architecture. Today, the Netronome architecture uses a three-tier processing architecture (Figure 6) based on two different processing chips. When migrating to the new NFP-6xxxx Flow Processor, which combines both packet and flow processing on one-chip, this architecture will be reduced to only two chips.

Tier 1
- Application / Control Plane Processing
- Deep Packet Inspection / Application ID
- Content Inspection, Behavioral Heuristics
- Forensics, PCRE, Analytics

Tier 2 & 3: Packet & Flow Processing
- Packet Classification / Filtering
- L2-L4 Packet Classification
- Stateful Flow Processing
- Cryptography / PKI
- Flow-based, Dynamic Load Balancing
- L2 Switching, L3 Routing, NAPT
- IPsec / SSL VPNs
- SSL Inspection
- 4x100, 12x40, 48x10 GigE I/O

The Netronome FlowNIC
Netronome’s flow processor-based NICs, or FlowNICs (Figures 7a, 7b), tightly coupled to multi-socket (up to four) x86 CPUs over a standard PCIe3 interface, provides an ideal component to extend a COTS server architecture to support both SDN and NFV.

With 216 programmable packet and flow processing cores, it caters to over 400 Gbps of networking I/O, while relaying 200 Gbps to up to four x86 devices. The result is one universal hardware design that can meet not only DPI and network analytics applications, but also firewall and load balancing applications.
NFV Platform and SDN Gateways

The software-defined NFV platform (Figure 8a) is an open platform for hosting various virtual network functions (VNFs) as standalone or in a service chain configuration. The platform integrates multi-core x86 with Netronome offload and acceleration card(s) and software. The integrated system can be fully virtualized.

The NFV platform can be used in the following applications:
1. NFV / Middlebox platform
2. Host various applications, such as fire wall, IPS/IDS and load balancing (VNF as a Service)
3. As a NFV Infrastructure as a Service (IaaS)
4. Accelerated Open vSwitch platform

The SDN gateway is an integrated hardware / software solution focusing on the gateway function between a private or public cloud and the WAN. It is also ideal for interconnecting geographically disparate enterprises over the WAN.

As shown in Figure 8b, the SDN Gateway reference platform can be used as:
1. Multi-tenant DC to MPLS WAN, under SDN OpenFlow 1.3 control
2. Enterprise-to-enterprise connectivity over the WAN under SDN OpenFlow 1.3 control
3. Accelerated (OpenVSwitch) OVS with OVSDB and OF-Config support

Challenges Ahead

The latest industry efforts in drafting NFV requirements, and its strategy to solicit Proof of Concepts, or PoCs, will trigger multi-vendor interoperable implementations aligning with specific operator use cases. It remains to be seen how SDN can be integrated in such PoCs and how these can meet the challenges in scalability, low-latency and reliability. In any case, SDN-OpenFlow needs to better address control of inline L4-L7 devices, as these will be based on NFV implementations.

Summary

SDN and NFV are two key inflection points addressing complementary goals: NFV is about network equipment virtualization, while SDN is about network virtualization. Netronome is capitalizing on both key networking trends by offering turnkey reference platforms addressing both SDN and NFV simultaneously, while meeting the latest OpenFlow protocol and NFV infrastructure domain requirements.