White Paper

NFV Validation across Boundaries

Executive Summary

Network Functions Virtualization (NFV) offers many advantages over dedicated network devices including potential reduction in costs for network and data center operators and increased revenue opportunities with the agility it affords. However, the agility potential also introduces more performance, reliability and security uncertainty with so many alternatives in the manner in which the infrastructure is designed. Holistic and continual validation across all the boundaries in a software-defined cloud stack is required for NFV to achieve successful deployments by operators. This whitepaper address the complex layers that require a validation solution capable of emulation and orchestration in order to make NFV infrastructure to be integrated and validated as a system.

Introduction

NFV is an industry term used to denote the capability of delivering network functions through virtualized compute infrastructure. These network functions span capabilities that have traditionally been delivered using dedicated hardware-based networking solutions. Within NFV topologies, some or all of the network elements that were physical and interconnected with cables are replaced with virtual machines within a virtual compute infrastructure. The virtual machines serving each function are termed Virtualized Network Functions (VNFs) in NFV terminology. The VNFs are deployed within a commercial or open-source hypervisor and interconnected using software implementations of switches or bridges, typically referred to as virtual switches. These virtual network devices can also be open-source or commercial and interconnected to physical network adaptors integrated within commercial off-the-shelf (COTS) computing/server hardware.

Routing, security, and other network functions like load balancing can be implemented using virtual machines in single-VM or multi-VM service chained topologies. General purpose computing hardware coupled with NFV software can be leveraged to reduce capital expenditures for dedicated networking hardware by leveraging the diminishing costs and increasing scale of multi-core processors combined with high-density network adaptors. Data center and network operators have incentives to move towards NFV because of the agility and flexibility it promises. Additional operating cost reductions can potentially be gained by employing automation to orchestrate different network topologies for different tenants (customers) or systems (applications/services) in cloud environments.

NFV Validation across Boundaries

Validation of NFV environment

The industry broadly views NFV as having applications in a diverse set of data center and transport network interconnect use cases. Service Providers and Data Center Operators can use NFV to deliver services with increased depth and breadth, enabling them to broaden revenue opportunities. They can also reconfigure network infrastructure to satisfy elastic, on-demand use cases with NFV service chains that combine network functions in different combinations per tenant. Data center and network operators, and Cloud Providers can use NFV to solve shard cloud infrastructure multi-tenancy issues such as on-demand data center interconnects, tenant isolation/security risks, and noisy neighbor performance degradation.

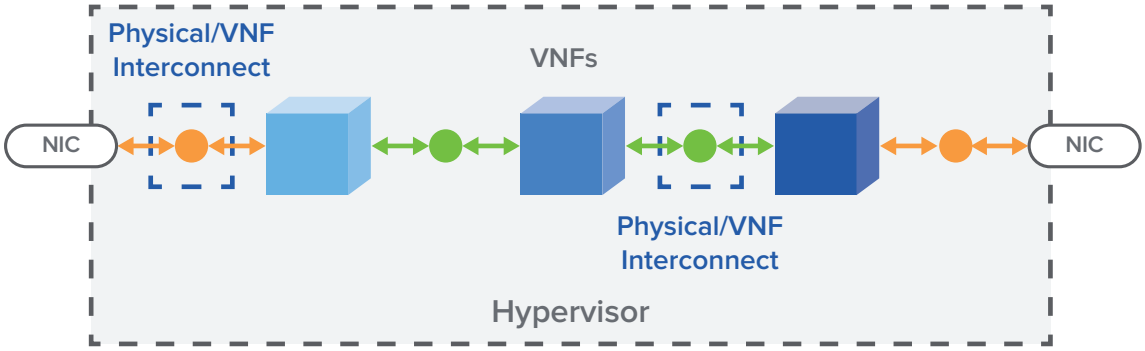

Figure 1. Virtual and Physical VNF Interconnects

Figure 1 shows the interconnections between VNFs in a service chain with circles, which is a simplification of the underlying virtual devices. The VNF to physical network interface card (NIC) connections are depicted in orange and can either be similar to the connections between VNFs shown in blue or a special type of interface that allows the VNF direct access to the physical Ethernet interface. This direct access usually facilities improved packet forwarding performance between the physical network interfaces and the VNFs over having virtual devices in between. Technologies employed for this direct access include PCI pass-through (Intel VT-d or AMD-Vi) and single root I/O virtualization and sharing (SR-IOV) and must be supported by the underlying hardware (compute platform, NIC) and hypervisor software.

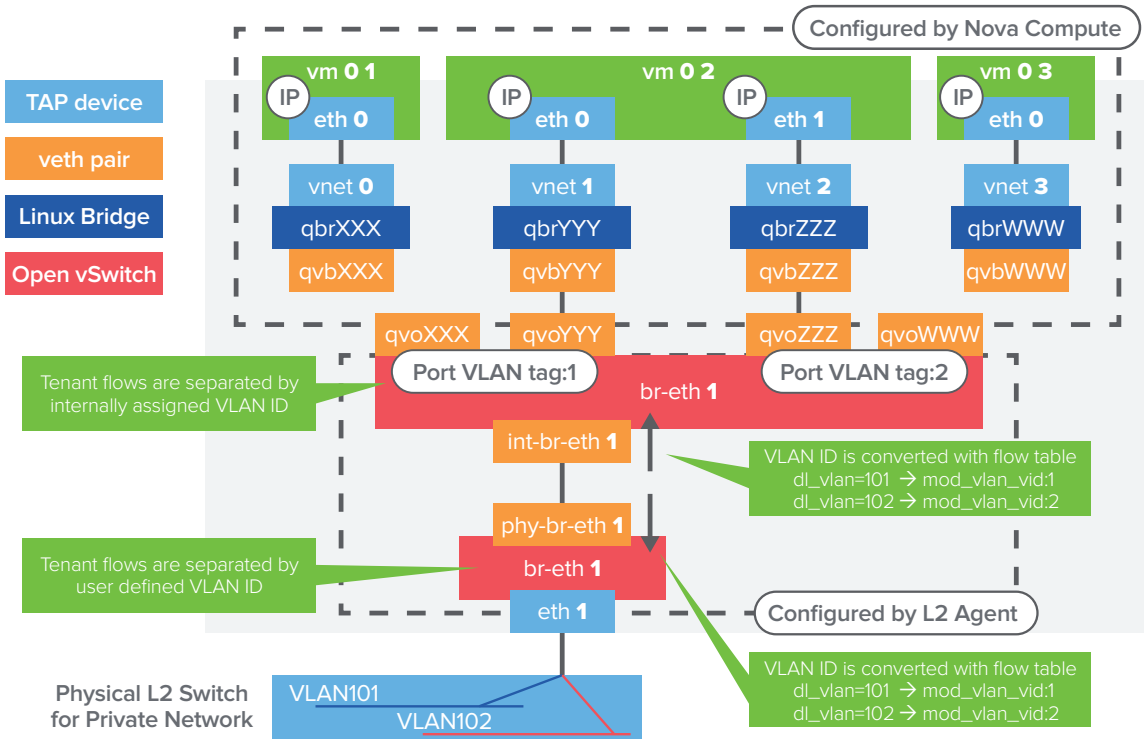

Figure 2. Various Linux virtual network devices between a guest OS within a virtual machine / hypervisor and a shared physical network interface card within the host OS. Source: OpenStack Cloud Administrator Guide

Figure 2 is taken from the OpenStack Cloud Administrator Guide and depicts the interfaces between virtual machines (or potentially VNFs) at the top of the diagram and a physical Ethernet interface at the bottom of the diagram in one potential virtual topology. There are various alternative methods to interconnect virtual machines. These methods are highly dependent on the hypervisor being used and may employ native models inherent in the hypervisor or by employing shared devices (ex, Linux bridges) or by using software-defined networking (SDN) open-source such as Open vSwitch (OVS) or even commercial virtual switching solutions. Figure 2 shows the use of OVS in an OpenStack context with VLAN encapsulation for distribution across the physical switching tier. OVS can be used in a SDN setting to dynamically control the virtual switching tier by explicitly controlling the contents of forwarding tables. However, it can be also be used in static environment where the virtual switch operates in a similar fashion to a physical switch with L2 bridging domains, MAC address learning and features like VLAN tagging etc. In this latter scenario, OVS can act as a standard Linux bridge but will offer the advantage of having more options (ex. easier to setup port mirroring for monitoring) and configurability.

The details of the virtual devices shown in Figure 2 are beyond the scope of this paper. However, the main point of including the diagram is to highlight that there are nine interfaces or boundaries between the virtual machine and the physical network interface in this scenario. The impact of all these software layers should be obvious to the reader – packet forwarding performance in this scenario is not going to be on par with physical network devices built to handle wire speed load on high speed Ethernet interfaces (10/40GbE) at small packet sizes. The server computing landscape is evolving continually and virtual switching deficiencies are improving with technology advancements in virtual to physical interactions provided by computing vendors such as Intel and AMD. The virtual device layers in Figure 2 can also be condensed with a Linux bridge depending on the deployment scenario.

NFV Validation across Boundaries

NFV Uncertainty

While NFV offers many benefits to network and data center operators, there is an abundance of uncertainty in selecting technologies, configuring infrastructure, optimizing performance and hardening for security. The flexibility NFV promises also increases the number of permutations that must be considered in a nearly exponential fashion. Vendors that release proprietary hardware-based networking products have the luxury of confining their solutions to a finite number of use-cases and can optimize the interaction of embedded software with hardware elements for operators. Traditionally, the burden of validating the core functions is shouldered by the network vendors. Today's NFV landscape shifts some of this burden to operators deploying NFV topologies while not absolving the need for network vendors to validate NFV software within different hardware (compute platforms, physical switching tiers) and software infrastructures (hypervisor, holistic cloud stacks public clouds, etc).

Spirent's experience in empirical testing of COTS compute coupled with NFV software shows that performance can vary widely and is dependent on several key variables including:

- Compute resource assignment to virtual machines serving as VNFs and more does not always translate to additional performance gains
- Configuration of the interfaces between the physical networking environment and virtual network elements (virtual infrastructure)
- Selection of virtual networking alternatives that interconnect VNFs

The validation burden is compounded beyond performance testing with the need to assess whether networking capabilities will functionally operate in different cloud stack environments.

Cloud Infrastructure Validation Challenges

NFV presents a myriad of validation challenges, as it is delivered through a cloud stack consisting of many hardware and software layers. A typical cloud stack is depicted in Figure 3 with hardware elements highlighted in dark blue and virtual/application software elements highlighted in light blue. Depending on the deployment scenario and specific technologies employed, the diagram below may look quite different and may not have all the depicted components. For example, a carrier NFV use-case to provide aggregation routing for multiple tenants may not have application/db, guest OS storage virtualization or NAS/SAN storage elements. But a data center operator NFV use-case will likely have everything depicted.

VNFs are deployed within a hypervisor to provide network virtualization. These virtual machines will have compute and VM image storage resources assigned and can dynamically be orchestrated through cloud management software such as OpenStack Nova Compute residing in the middleware tier. A subset of the VNFs can also have SDN interfaces such as the usage of the OpenFlow protocol to dynamically control the forwarding tables of OVS instances. In this case, OpenFlow controllers may also reside in the middleware tier.

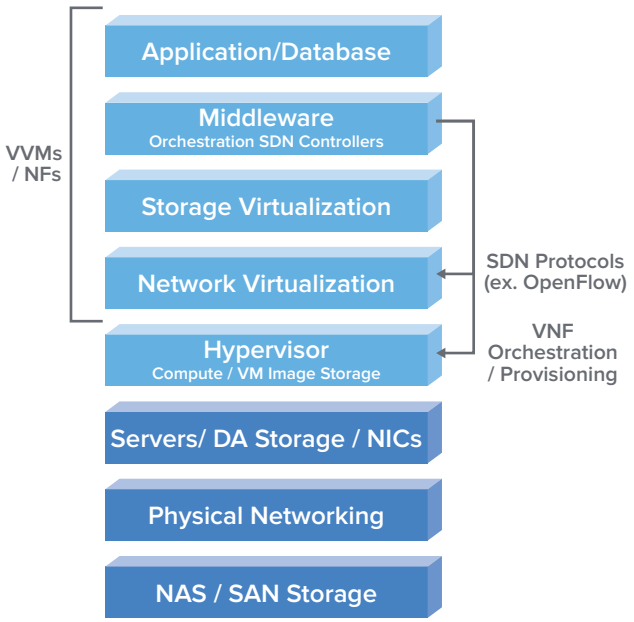

Figure 3. Cloud Infrastructure for NFV Deployment

VNFs may be combined in a service chain in front of the application/service tiers to provide security and load balancing services in cloud data center use-cases. Unless an application/service workload is localized on a compute node, VNFs will likely be connected to the external physical switching fabric through the server NICs and often employ a form of encapsulation such as VLAN or VXLAN to facilitate traffic isolation and forwarding across the physical fabric. In data center use-cases such a big data / data analytics, external storage is usually used over and above any available direct attached storage (DAS) and may have its own physical networking fabric; especially for block technologies such as fiber channel (FC). VNFs will likely not employ external storage but traffic might use the same NIC / physical fabric converged Ethernet conduits and hence must be considered in validating the entire cloud stack environment for NFV deployments.

A validation solution that can test across all the boundaries outlined in Figure 3 requires test solutions that are both and physical and virtual in nature with test interfaces on the inside and outside of the infrastructure. Cloud stack components must be able to be isolated through multi-dimensional emulations to pinpoint functional and performance issues. System testing must also be performed to validate multiple layers simultaneously to root out complex issues that arise due to their interdependencies. L2/3 host and router emulation is required to validate forwarding and IP services that may reside in VNFs that interplay with the physical switching fabric. Storage and application protocol emulation is essential to validate stateful L4-7 VNFs such as load balancers and security functions. It is also required to validate the end services/applications that the NFV infrastructure may be serving.

NFV increases the number of test permutations that must be performed significantly over traditional dedicated hardware networking topologies because of its flexibility so without automation in both the test/emulation and orchestration planes, adequate test coverage is unfeasible with manual methods. Automation of the orchestration plane allows the cloud stack / NFV infrastructure under test to be modified to run the battery of required test-cases continually during key development and integration checkpoints.

Spirent NFV Validation Solution

Spirent aims to support a Continuous Cloud Validation (CCV) process to empower all NFV and cloud stakeholders to eliminate the uncertainty inherent in NFV infrastructures due to their multi-layered complexity and flexibility. Spirent is uniquely positioned to facilitate this with its breadth of test solutions that allow automation in all required dimensions and support validation across all key NFV and cloud stack boundaries. These capabilities are outlined in subsequent sections.

Validation across Boundaries

One of Spirent's key core competencies over the last two decades has been building dedicated hardware (FPGA-based) capable of L2/3 traffic generation/analysis at wire speed on high speed Ethernet test interfaces. This allows for deterministic and precise control of all key L2/3 traffic attributes including frames sizes, encapsulations, host emulation scale (millions of MAC/IP addresses) and load. Realistic IP diversity can be emulated with these generation engines and is paired with an equally capable analyzer engine capable of advanced per-flow tracking and statistics. The Spirent TestCenter platform is a unified architecture that delivers this capability in different chassis form factors with shared test modules and have traditionally been sold for testing of dedicated routing/switching network devices. However, these solutions can also be used to validate NFV infrastructure.

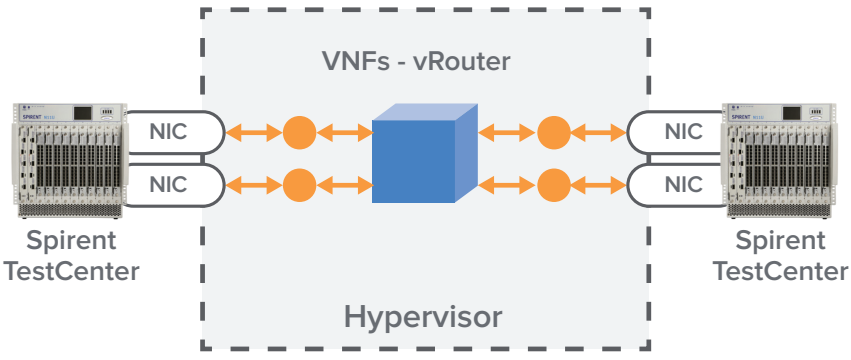

Figure 4. End-to-end validation with Spirent TestCenter L2/3 traffic generation/analysis

NFV Validation across Boundaries

Figure 4 shows how Spirent TestCenter L2/3 test modules can be used to test a simple virtual routing VNF that is mapped to four NIC interfaces. This form of testing allows the virtual infrastructure plumbing housing NFV to be characterized. Bottlenecks can be exposed in the underlying plumbing to find maximum packet forwarding/processing performance of the compute platform end-to-end from physical network cards within the server through the VNF under test and back out the physical network cards across virtual/physical boundaries. The advantage of this form is testing is a deterministic measured offered load at ingress (source for flow) ports and measured receive load at egress ports (sink for flow). One-way (as opposed to round trip) packet latencies over time, latency distributions and latency variance can all be measured accurately with 10 nanosecond resolution to consider NFV packet forwarding performance versus dedicated switch/router hardware solutions. The entire compute environment under test can be benchmarked using de facto RFC standard methodologies in different scenarios including:

- NIC to PCI-E slot population with consideration of PCI-E lane width and the interconnect to CPUs
- Multi-port NICs can be evaluated with bi-directional traffic
- Port-pair, partial mesh and full mesh traffic patterns with different packet sizes
- Compute resource (logical CPU core and memory) assignment to the VNF including memory backing schemes like hugepages
- Different disk/storage mappings to the VNF (if VNF requires frequent read/writes to storage)
- PCI passthrough support with Intel VT-d with input/output memory management unit (IOMMU) enabled

In many cases end-to-end NFV validation with L2/3 traffic is not feasible due to the NFV topology being deployed or the need to validate VNF to VNF interconnects that do not have an egress physical network interface mappings. This is depicted in Figure 5. Here the same VNF virtual router is being tested. However, the vRouter will be deployed with additional virtual machines attached to its right to host application/service tiers. The blue circles representing VM-to-VM interconnects can be shared Linux or OVS bridges that will ultimately connect to the application/service tiers. This topology allows the vRouter to be validated across the physical and virtual device (bridges) boundaries in an isolated fashion. Test traffic will traverse from physical Spirent TestCenter test ports attached to the physical NICs to virtual Spirent test ports attached to the shared bridges. This introduces yet another critical component of Spirent's solution for validating NFV – virtual test ports. Spirent offers the same generation/analysis engines in the form of a virtual machine. Test traffic can be generated and analyzed from physical to virtual ports allowing NFV/virtual infrastructure to be tested from the outside in and inside out.

Spirent's physical and virtual test interfaces have near parity in feature set with the only major exception being diminished resolution for packet latency measurements.

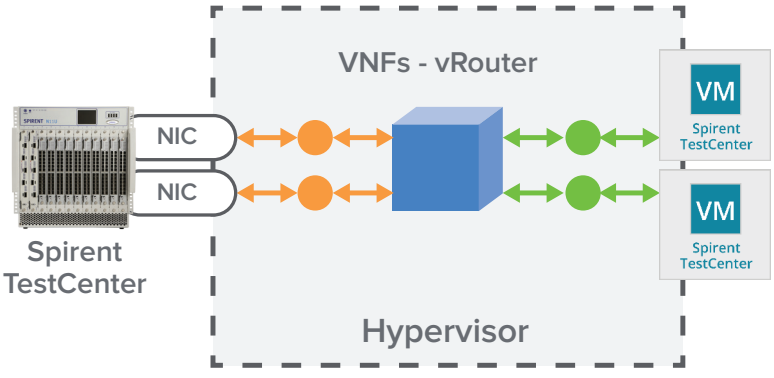

Figure 5. Physical to virtual validation across virtual bridging using Spirent's physical and virtual test ports

Multi-Dimension Automation

Characterizing the L2/3 packet forwarding performance of the host compute environment, interconnecting virtual devices and the L3 VNFs outlined in the previous section is crucial, as virtual switching infrastructure still lacks the performance of dedicated router/switch hardware devices with higher packet loss and latency under many conditions. But NFV validation requires emulation of many diverse control-plane protocols and the use of L4-7 application protocols under many topologies. The ability to automate in all of these dimensions is key since changing hypervisor or virtual switching parameters can change the NFV performance profile drastically. The ability to provision or orchestrate the underlying NFV/virtual infrastructure is also desirable since complete automation cannot be achieved without being able to control this dimension programmatically.

Figure 6 shows another NFV topology that requires L4-7 application traffic generation. Security devices such as stateful firewalls and other application-aware devices such as load balancers will not propagate L3 traffic and require real application workloads to validate them. NFV service chains (sequences of connected VNFs) will typically contain these types of functions, as shown in Figure 6. Spirent has spent well over a decade developing the Avalanche platform, which is an advanced application protocol emulation solution. Millions of stateful (TCP) connections or emulated users can be generated concurrently and per second. Spirent has packaged the Avalanche in hardware-based form-factors for high performance scenarios but also in a virtual machine, as shown in Figure 6 on the left side.

Stateful NFV service chains should be benchmarked with the Avalanche platform for user scalability with connection capacity tests to measure the total number of open or concurrent connections service chain can maintain with application session transactions occurring. The rate at which the NFV service chain can process connections with application transactions is important to verify, as many use cases have end-user traffic that exhibits sudden spikes in connections rates, such as users logging in all at once in the morning or web search storms that occur during news-making events. A peak rate test case is used to measure the maximum rate of connections/transactions the NFV service chain can handle per second.

Using Spirent Avalanche in figure 6 allows the NFV service chain (firewall, load balancer and VNF interconnects) and the end application/services to be validated together, as a system. The Avalanche can emulate clients and their application transactions across web (HTTP), file, instant messaging, streaming video, storage and many more protocols. The Avalanche platform also includes high performance application server emulation (named Reflector instances) to allow the NFV service chains to be isolated without the need for real application/service VMs (replace right side of figure 6 with Reflector instances).

Figure 6 also depicts how the Spirent Virtual Deployment Manager can orchestrate or provision the NFV infrastructure under test (NVF VNFs or other VMs) and the Spirent virtual test interfaces (Spirent Avalanche virtual or STC virtual instances). This adds the orchestration dimension to the other emulation dimensions Spirent test ports (physical or virtual) support to provide a complete automation solution. Optimal configurations for NFV infrastructure can be converged on by automating the placement of NFV service chains with benchmarking test-cases.

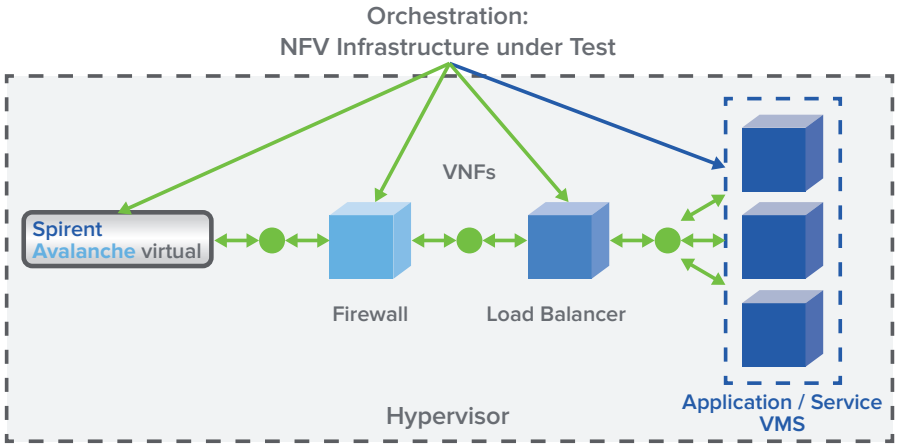

Figure 6. Spirent Virtual Deployment Manager orchestrating Spirent test virtual machines and NFV infrastructure under test

NFV Validation across Boundaries

Figure 7 shows how Spirent test interfaces are capable of control-plane emulation in the routing and SDN dimensions to validate more complex NFV topologies. A Spirent TestCenter virtual interface (top) can emulate an OpenFlow controller in the SDN dimension to control the forwarding tables of Open vSwitch (OVS) bridges that interconnect pools of database VMs with a virtual router VNF. Database replication may occur within pools and across different data centers by way of the vRouter VNF. Business logic would control which database instances are active and when replication events take place in the actual deployment. DevOps engineers can consider the design of this business logic by using the test topology depicted. Another Spirent TestCenter virtual interface (left) may be used to emulate routers and inject routes to the vRouter VNF and another (bottom) to send traffic destined to the advertised routes. The control-plane scalability of the vRouter VNF can be evaluated and the L3 throughput of the OVS/vRouter combination can be benchmarked. The latter measurements will provide DevOps engineers with an idea of how quickly database replication events may occur.

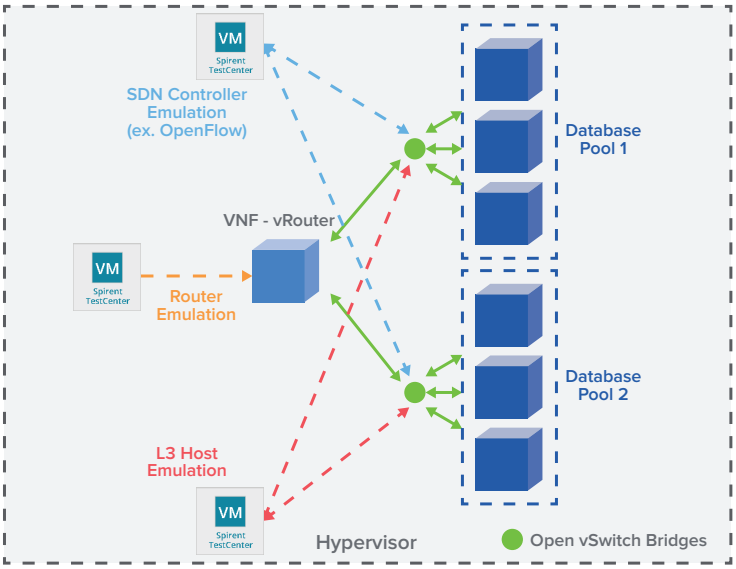

Figure 7. Illustrative topology for database replication use case

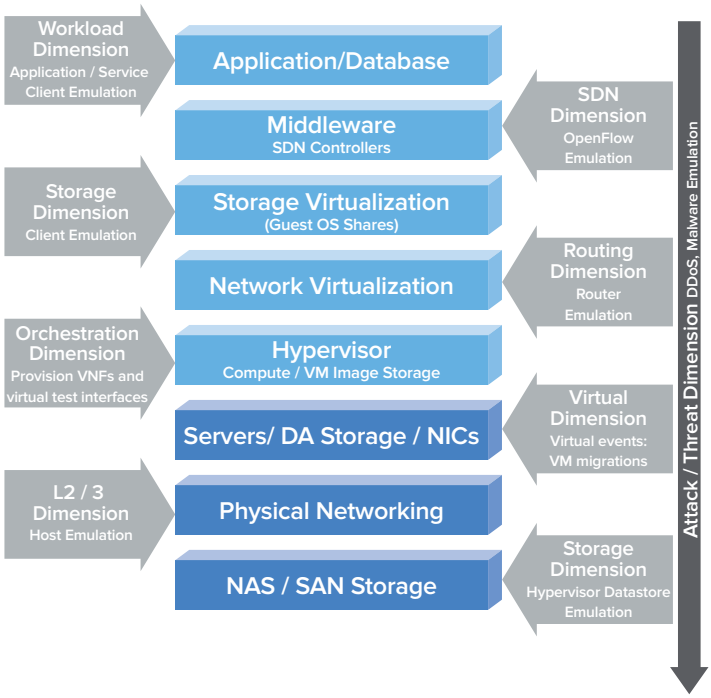

Figure 8. Spirent Multi-dimension automation: illustrates how Spirent can emulate all required dimensions to holistically validate NFV/Virtual infrastructure within a cloud stack

Figure 8 repeats the cloud stack diagram represented in figure 3 but superimposes all of the data-plane, control-plane, orchestration-plane and attack-plane dimensions Spirent supports across its broad portfolio. Data-plane emulation spans the application, storage and L2/3 traffic generation and analysis capabilities highlighted in blue. Spirent can emulate control-plane protocols highlighted in yellow including SDN/Openflow, routing and the ability to add additional stress to the virtual infrastructure with hypervisor events such as VM migrations. Spirent Virtual Deployment Manager can be employed to control the orchestration-plane placing both NFV VNFs and Spirent virtual test interfaces into the NFV topology. Finally, the Avalanche platform (both virtual/physical) can be used to also control the attack-plane in addition to storage/application protocols. The attack-plane includes the ability to launch distributed denial of service (DDoS) attacks that can disrupt network infrastructure and web services. Client-side vulnerabilities and malware can be injected together with legitimate traffic to validate the security effectiveness of firewalls, intrusion prevention systems (IPS) and other network-level security devices. These same threats can be made to target applications and services that NFV infrastructure front-ends to harden them against application-level vulnerabilities such as cross-site scripting or SQL injection attacks.

NFV Validation across Boundaries

About Spirent

Spirent provides software solutions to high-tech equipment manufacturers and service providers that simplify and accelerate device and system testing. Developers and testers create and share automated tests that control and analyze results from multiple devices, traffic generators, and applications while automatically documenting each test with pass-fail criteria. With Spirent solutions, companies can move along the path toward automation while accelerating QA cycles, reducing time to market, and increasing the quality of released products. Industries such as communications, aerospace and defense, consumer electronics, automotive, industrial, and medical devices have benefited from Spirent products.

For more information

To learn more about Spirent's test and lab automation solutions, visit www.spirent.com/automation.

Conclusion

Spirent's NFV solutions validates all of the cloud stack layers and NFV elements to be tested as a system and also isolated to pinpoint the root of any issues that are exposed. This helps vendors and operators improve their products and services and make more informed decisions during the design and integration processes. With so many permutations to test, automation of the validation process is imperative. Complete automation can only be achieved in NFV environments with the ability to orchestrate the components under test and the test emulation interfaces.

spirent.com

AMERICAS 1-800-SPIRENT
+1-818-676-2683 | sales@spirent.com

EUROPE AND THE MIDDLE EAST
+44 (0) 1293 767979 | emeainfo@spirent.com

ASIA AND THE PACIFIC
+86-10-8518-2539 | salesasia@spirent.com