ENSURING OPEN VSWITCH PERFORMANCE FOR A PREDICTABLE NFV INFRASTRUCTURE

A Linux Foundation Collaborative Project

INTRODUCTION

The success of NFV applications—such as Evolved Packet Core (vEPC), Customer Premises Equipment (vCPE) and Broadband Remote Access Services (vBRAS)—depends on a predictable infrastructure. Accordingly, the Virtual Network Functions (VNFs) that comprise these applications need deterministic behavior from the NFV infrastructure (NFVI).

Although not a required component for NFVI[1], many cloud and Telco infrastructure providers are evaluating virtual switches—particularly Open vSwitch (OvS)[2] but others as well—as a component of the NFVI in order to direct frames between VNFs. The performance of a virtual switch can directly affect the number of subscribers/VNFs that can be supported on a single blade.

While virtual switches were not originally designed for NFV use cases, they have evolved in recent years to become production quality with higher performance, and they offer more flexible configuration of the data path[3] than their hardware-based predecessors[4].

OPNFV's VSPerf project[5] supports a predictable infrastructure by developing an architecture-independent vSwitch testing framework and associated tests that validate the performance of different vSwitch implementations in NFV deployments. The project evaluates virtual switches in order to identify performance limitations and to drive architectural changes to improve vSwitch throughput and determinism.

---

[1] For the original ETSI NFV Overview, see https://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.02.01_60/gs_NFV002v010201p.pdf.
[2] See www.openvswitch.org.
[3] This is useful, for instance, in service function chaining: https://wiki.opnfv.org/display/sfc/.
[4] See Open vSwitch Enables SDN and NFV Transformation at https://networkbuilders.intel.com/docs/open-vswitch-enables-sdn-and-nfv-transformation-paper.pdf.
[5] See https://wiki.opnfv.org/display/vsperf/.

ABOUT VSPERF

To achieve its goals, VSPerf implements a modular test framework, combining traffic generation, virtual switches, virtual network functions, and network configuration test cases (*Figure 1*). It then generates a performance benchmark report for all supported test cases.

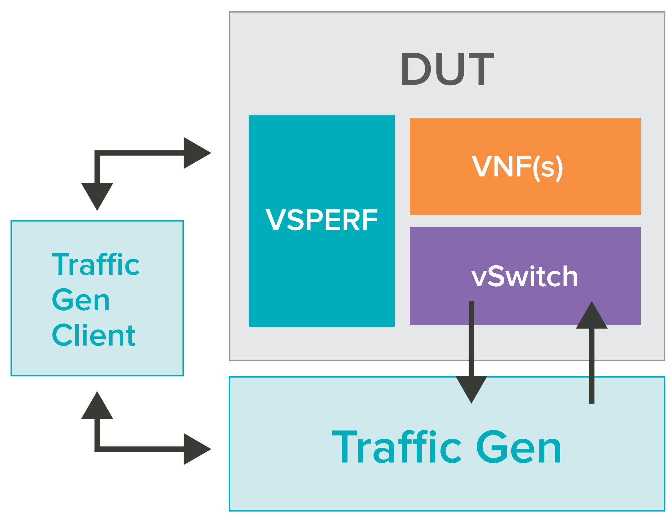

Figure 1: VSPerf's Modular Test Framework

As with all OPNFV projects, and in the spirit of NFV itself, each of the modules is componentized and thus interchangeable. For example, a test setup using any traffic generator can be supported within this framework.

VSPerf Test Matrix Overview

The test matrix for OPNFV draws largely from two RFC's: RFC2544 *Benchmarking Methodology for Network Interconnect Devices*, and RFC2889 *Benchmarking Methodology for LAN switching Devices*.

These reports are referenced in *Benchmarking Virtual Switches in OPNFV*[6], a public IETF draft that describes the tests and benchmarks used to evaluate virtual switches.

VSPerf measures speed, accuracy, reliability, and scale for activating flows and processing them over a sustained period of time. This is summarized in the following table.

Table 1: The Level Test Design (LTD) for VSPerf

	Speed	Accuracy	Reliability	Scale
Activation	• Address Learning Rates and Caching Capacities • Packet Processing Latencies and Variations	• Flow Addition	• RFC2544. System RecoveryTime • RFC2544. ResetTime	• RFC2889. AddressCaching Capacity
Operation	• RFC2544.PacketLossRatio • RFC2544.PacketLossRate FrmMod • RFC2544.BackToBack Frames • RFC2889.MaxForwarding Rate • RFC2889.ForwardPressure • RFC2889.BroadcastFrame Forwarding • RFC2889.BroadcastFrame Latencytest • CPU.RFC2544.0PacketLoss • RFC2544.WorstN-BestN • InterPAcketDelayVariation. RFC5481	• Frame Filtering	• RFC2889. Soak • RFC2889. SoakFrame Modification • PacketDelay Variation. RFC3393. Soak	• Scalability.RFC25 44.0PacketLoss • MemoryBandwid th.RFC2544.0Pac ketLoss.Scalability

[6] See https://tools.ietf.org/html/draft-vsperf-bmwg-vswitch-opnfv-02. The older drafts on testing methodologies (RFC2544, RFC2889, and others) are referenced in that draft.

In order to match real-world deployments, the test design is implemented on a variety of physical and virtual interfaces; for instance, some baseline scenarios include:

- Physical to physical interfaces (phy2phy)
- Physical to virtual to physical, via a single virtual machine loopback framework (PVP)
- A two VM loopback framework inside two physical interfaces (PVVP)

An example of the last (PVVP) is shown in Figure 2. There is a virtual switch and two VNFs (shown as "Hypervisor+VM") running on a host.

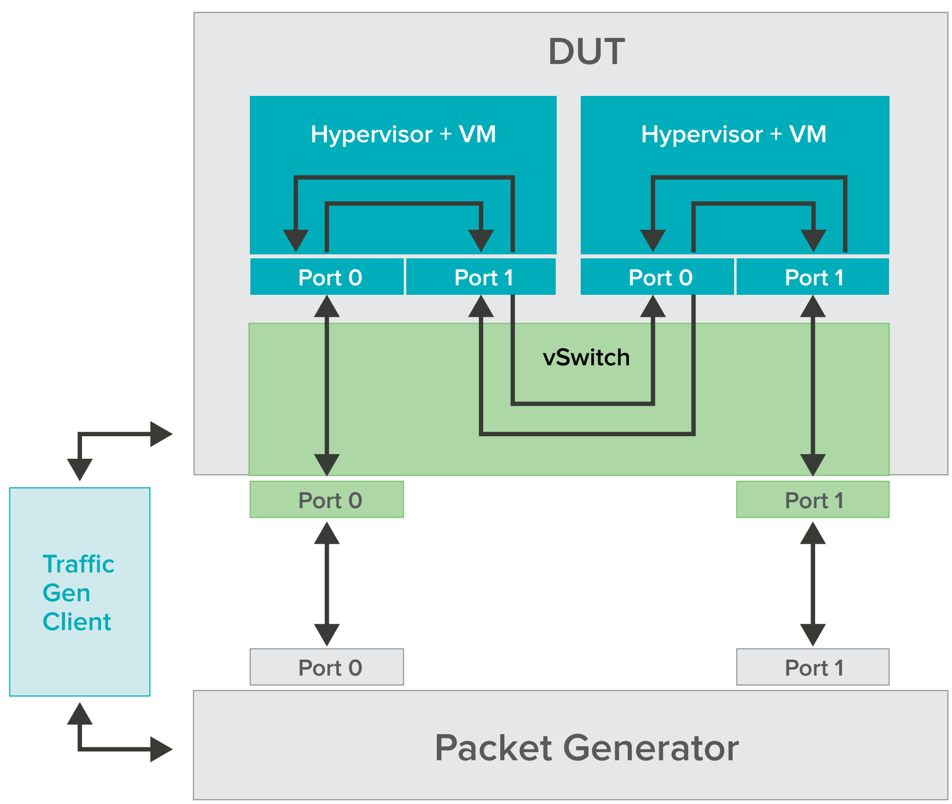

Figure 1: VSPerf's Modular Test Framework

The flow here is bidirectional between the two physical ports (Ports 0 and 1) on the packet generator and on the host. On the host, traffic flows bidirectionally between the VNFs via the vSwitch.

Categories of VSPerf Tests

The major categories of tests include:

- Throughput for maximum forwarding rate
- Packet and Frame Delay Distribution to measure delay for constant traffic loads over time
- Scalability to understand how virtual switch performance may change according to number of flows, active ports, or complexity of the forwarding logic's configuration
- Stream Performance to measure bulk data transfer
- Control and Data Path Coupling to measure, for instance, how long it may take the first packet in a flow to be queued for transmission
- CPU and Memory Consumption to understand the virtual switch's footprint on the system

Additionally, there are so-called "soak" tests, where the selected test is conducted over a longer period of time, say from 6-24 hours. These tests capture transient changes in performance which could not otherwise be captured in a test run of shorter duration.

VSPERF'S THEORY OF OPERATIONS

Some key considerations when benchmarking vSwitch performance are:

- To allow for a direct comparison, vSwitches should be benchmarked—when possible—in a similar way to physical switches
- Accuracy, consistency, stability and repeatability of the results should be ensured between test runs.
- It is essential to limit, and if possible, eliminate, any noise that may interfere with the accuracy of the metrics collected by the test.

To compare the performance of virtual designs and implementations with their physical counterparts, identical benchmarks are needed. It is important to note the number of parallel cores needed to achieve comparable performance with a given physical device, or whether some limit of scale was reached before the cores could achieve the comparable level. Other factors to consider include CPU utilization, cache utilization, and memory usage.

When benchmarking the performance of a vSwitch there are many factors that can affect the consistency of results—one key factor is matching the various hardware and software details of the system under test.

VSPERF'S RESULTS AS OF OPNFV BRAHMAPUTRA

The results dashboard for VSPerf is available at:
http://testresults.opnfv.org/proto/index-vsperf.html

Furthermore, a full test report in PDF format can be found at
http://artifacts.opnfv.org/vswitchperf/brahmaputra/report/report.pdf

In general, the test results are very stable and predictable as of the Brahmaputra release of OPNFV.

All of the OPNFV results are available in a dashboard at:
https://www.opnfv.org/opnfvtestgraphs/per-test-projects/vsperf

CONCLUSION

The objective of VSPerf is to evaluate a virtual switch to identify its suitability for Network Functions Virtualization (NFV) in telecom environments. VSPerf also aims to identify any gaps or bottlenecks in order to drive architectural changes to improve virtual switch performance and determinism.

The move to NFV Infrastructure has resulted in many new benchmarking initiatives across the industry. Performance of virtual devices (virtual switches and other VNFs) and virtual connectivity (VNF-to-NIC, VNF-to-VNF, NIC-to-NIC), is a key consideration in NFV design and infrastructure.

In providing a best practice reference for benchmarking the virtual networking platform, the VSPerf project is continuing to automate virtual switch benchmarking across defined dimensions, focused on achieving repeatable and reusable results so that switching performance with NFV infrastructure and NFV deployments can be more reliably planned. VSPerf is applicable to real deployments through a feedback loop with industry communities.

VSPerf finds the boundaries of deterministic performance of virtual switch and virtual networking topologies—both in single instances and at full density on compute machines. The project achieves this by identifying and quantifying resources that matter by publishing benchmarks with telemetry data, and driving the community to optimize virtual networking infrastructures by lowering the cost of moving packets within computers.

Ultimately, VSPerf brings forth many lessons in VNF benchmarking methodology[7], extended RFC2544 methodology, results highlighting, and runtime x86 resource analysis.

---

[7] Benchmarking Methodology for Network Interconnect Devices: https://www.ietf.org/rfc/rfc2544.txt.