NFVbench Project Proposal

Project Name

- Proposed name for the project: NFVbench
- Proposed name for the repository: nfvbench

Project description

The NFVbench project develops a toolkit that allows developers, integrators, testers and customers to measure and assess the L2/L3 forwarding performance of an NFV-infrastructure solution stack (i.e. OPNFV scenario) using a black-box approach.

Scope and detailed description

It is currently very difficult to assess realistic performance of an NFVI solution due to the complexity of the system and the wide range of standards and tools that are applicable. As a result, performance benchmarks published today are difficult to compare as they always come with different variations in method of measurement, conditions of measurements and ways to report results.

To illustrate the problem, virtual switch benchmarks (such as OVS-DPDK, FD.io VPP) provide a good assessment of the inherent efficiency of a vswitch implementation but the benchmark numbers reported are far from representing actual performance of a full system because the vswitch is only one component of the system. Furthermore, conditions of measurement of vswitches are often tailored to favor the highest efficiency (and benchmark numbers) but in ways that are not sustainable or not representative in a production system:

- Load conditions and workload mix
- System resources sharing with other critical components such as system services, OpenStack services, VNFs (CPU, cache, memory, NIC, top of rack switches)

The NFVbench project develops a toolkit that allows developers, integrators, testers and customers to measure and assess the network performance of a full NFV-infrastructure solution stack. The resulting performance measurements are much more meaningful for the final users because they take into account all the trade-offs that are necessary and unavoidable to build a viable production platform.

NFVbench is to leverage existing literature describing packet paths (OPNFV, ETSI NFV TST), filter/refine and pick only those that are relevant for real NFVI deployments, and augment them with scale out feature in order to reflect more accurately real NFV deployments. The main goal is to provide a representative NFVI data plane benchmark using representative, realistic and reproducible (blueprint) deployment conditions that are as close to production conditions as possible:

- Realistic and most common packet paths (single and multi VNF chains, multi-flow)
- Production deployment scale with fully loaded compute nodes (multi-chaining)
- Network performance modular efficiency (performance per smallest amount of resources i.e. per core or socket, and verification of linear scale i.e. when increasing available resources by one core or socket)

NFVbench can automatically stage service chains using standard OpenStack APIs to launch one or more fast L2 forwarders as VNFs and form any of the pre-built packet paths. Once staged, the performance of these chains is measured through the integrated TRex traffic generator (developed as part of the FD.io project) following a well-defined subset of RFC-2544 that is the most representative of NFV traffic.

Any measurement using virtualized traffic generators requires calibration to ensure that the system under test is measured, not just the performance limit of the traffic generator itself. It is a common issue in the industry that performance test software is misconfigured and test results are misunderstood. To counter this problem, NFVbench conducts an automated calibration in advance of each test campaign. If the reference test results are in the same order of magnitude as the perceived NFVI benchmarking results, NFVbench warns the user that the results are bogus.

NFVbench always gathers complete information about the test environment and documents it as part of the test report. This includes hardware capabilities and options (NICs!), firmware (BIOS) configuration, OPNFV/OpenStack software versions and configuration.

NFVbench can perform at least 3 types of measurements:

- fixed rate at fixed duration (with detailed parameters to be defined as part of the project)
- NDR (no drop rate binary search: highest throughput without dropping packets)
- PDR (partial drop rate binary search: highest throughput with a maximum allowed drop rate).
- Additional measurement types aligning with relevant IETF and ETSI standards.

The whole system is used in conditions comparable with real operations:

- NFVbench uses OpenStack APIs to stage the service chains
- Data plane traffic is sent to the compute nodes through a physical switch that can be optionally controlled by NFVbench (for stitching traffic)
Alternatively, NFVbench can also measure the basic L3 performance using any L3 forwarder.

NFVbench is to enforce packet paths to test, the exact scale out conditions (number of service chains) and the exact traffic patterns (number of flows x number of service chains).

Dependencies - OPNFV and upstream projects

Dependencies: Overview

- OPNFV projects:
  - Testing projects (FuncTest, YardStick, …)
  - OPNFVdocs
  - Octopus
  - Releng
  - Bamboo
  - Sample-vnf
  - VSpert
  - Bottlenecks
- Upstream projects:
  - OpenStack
  - FD.IO (in particular the TRex traffic generator project)
  - PNDA.IO

Positioning with other OPNFV and open source projects

The main focus of NFVbench is the NFVI full stack data plane benchmarking using realistic production deployment conditions.

NFVbench does not focus on the following areas and will align with, complement and leverage projects that already cover properly these areas:

- Vswitch performance benchmarking is current handled by the OPNFV vsperf project and FD.io CSIT projects
- General compute performance benchmarking – covered by yardstick to some degree
- Control plane performance benchmarking – covered by yardstick/OpenStack Rally
- Storage performance benchmarking – covered by OPNFV storperf and OpenStack KloudBuster
- Simple VM to VM data plane benchmarking – covered by OpenStack VMTP and Shaker
- Stateful VNF performance benchmarking – covered by the current contribution from Intel to the yardstick project (work in progress)
- Benchmark results storage and publication – covered by OPNFV yardstick, qtip, bamboo, PNDA.io
- Overall lab level test orchestration – covered by OPNFV CI/CD
Committers and Contributors

- Project Lead: Alec Hothan (ahothan@cisco.com)
- Committers
  - Alec Hothan (ahothan@cisco.com)
  - Carsten Rossenhövel (cross@eantc.com)
  - Frank Brockners (fbrockne@cisco.com)
  - Yichen Wang (yicwang@cisco.com)
  - Al Morton (acmorton@att.com)
- Contributors
  - ...

Planned deliverables

Project deliverables:

- Toolkit to allow for black box testing of NFVI solution stacks – open source with Apache 2.0 license
- Results into testresults databases and Bamboo
- Integration of the NFVbench toolkit into OPNFV’s CI/CD pipeline

Proposed Release Schedule:

The project initial deliverable is expected to be available in time for the OPNFV “Euphrates” release.

Support for at least 1 OPNFV scenario is targeted for the Euphrates release. Integration in the OPNFV CI/CD may be available in the Euphrates or F release.