Leveraging OPNFV test tools beyond the NFV domain

Georg Kunz, Emma Foley & the OPNFV testing community
Goals of this talk

1. Create awareness for OPNFV test tools
   - Targeting users outside of NFV domain and telcos not active in OPNFV
   - Beneficial for most cloud operators and developers
   - Leverage the extensive tooling OPNFV has built over 4 years

2. Trigger a discussion about the evolution of the OPNFV test tools
   - How to evolve the test tools to address emerging use cases?
   - Learn from people outside of NFV domain about their needs
What does OPNFV do?

- Development / Integration
- Images
- Packages
- Platform Deployment
- Systems
- Testing & Reporting

Federated OPNFV labs

Requirements

Images

Packages
What does OPNFV do?
OPNFV Test Ecosystem

Functional Testing

- Functest: NFVI, VIM APIs + Functions
- RefStack: openstack
- Tempest: openstack
- Cloudify: VNF
- Kubernetes

OPNFV Feature Tests

Non-functional Testing

- Yardstick: NFVI & VNF Performance
- Bottlenecks: Load tests staging manager
- vsperf, NFV Bench: NFVI dataplane Performance
- StorPerf: NFVI Storage Performance

Test Result Database

Dovetail

Compliance Verification

OPNFV Compliance Verification
OPNFV Test Ecosystem

Functional Testing
- NFVI, VIM APIs + Functions
- RefStack
- Tempest
- Cloudify
- VNF
- Kubernetes

Non-functional Testing
- Yardstick
  - NFV & VNF Performance
- Bottlenecks
  - Load tests staging manager
- vsperf, NFVBench
  - NFVI dataplane Performance
- StorPerf
  - NFVI Storage Performance

Test Result Database
- Dovetail
  - OPNFV Compliance Verification

Compliance Verification
Description
Functional verification of OpenStack and K8s

Components tested
Cloud infrastructure control plane

Stage deployed
From patch set verification to release gating

Collected metrics
Pass / fail

Project packaging/release
Multiple docker containers

Extensibility
Build with extensibility in mind: based on Xtesting
OPNFV Test Ecosystem

**Functional Testing**
- OPNFV Feature Tests
  - RefStack
  - Tempest
  - Cloudify
  - Kubernetes
  - NFVI, VIM APIs + Functions

**Non-functional Testing**
- Yardstick
  - NFVI & VNF Performance
- Bottlenecks
  - Load tests staging manager
- vsperf, NFVbench
  - NFVI dataplane Performance
- StorPerf
  - NFVI Storage Performance

**Report and Compliance**
- Test Result Database
- Dovetail
  - OPNFV Compliance Verification
Yardstick

**Description**
Infrastructure Verification and NFVI/VNF characterisation

**Components tested**
Cloud infrastructure resources

**Stage deployed**
CI and pre-production verification

**Collected metrics**
Performance metrics and pass/fail metrics (HA tests)

**Project packaging/release**
Docker container

**Extensibility**
Test cases integrated as scenarios
OPNFV Test Ecosystem

Functional Testing
- OPNFV Feature Tests
  - Functest
    - NFVI, VIM APIs + Functions
  - RefStack
  - Tempest
  - Cloudify
  - VNF
  - Kubernetes
- NFVI, VIM Performance
- VNF Performance

Non-functional Testing
- Yardstick
  - Load tests
  - Staging manager
- vsperf, NFVbench
  - NFVI dataplane Performance
- StorPerf
  - NFVI Storage Performance

Compliance Verification
- Dovetail
  - OPNFV Compliance Verification

Test Result Database
**Description**
- Simulates extreme or long term product usage

**Components tested**
- Cloud infrastructure control plane

**Stage deployed**
- CI and performance tuning of infrastructure

**Collected metrics**
- pass/fail metrics

**Project packaging/release**
- Docker container

**Extensibility**
- Test scheduler for other OPNFV tools, e.g. Yardstick, StorPerf

---

**Virtual Network Functions**

**Management and Orchestration**

- **OpenStack + Kubernetes**
  - KVM
  - LXD
  - Compute Virtualization
  - Storage Virtualization
  - Network Virtualization

**Data Plane**

- **FD.io**
- **DPDK**

**Infrastructure**

- **Pharos Community Labs**
- **OPNFV Bare Metal Lab**
OPNFV Test Ecosystem

**OPNFV Feature Tests**
- **Functest**
  - NFVI, VIM APIs + Functions
- **RefStack**
- **Tempest**
- **Cloudify**
- **VNF**
- **Kubernetes**

**Functional Testing**

**Non-functional Testing**

- **Yardstick**
  - NFVI & VNF Performance
- **Bottlenecks**
  - Load tests staging manager
- **vsperf, NFVBench**
  - NFVI dataplane Performance
- **StorPerf**
  - NFVI Storage Performance

**Test Result Database**
- **Dovetail**
- **OPNFV Compliance Verification**
vsperf

Description
Optimizing switching technologies and NFVI data path components

Components tested
Virtual switch and packet processing components

Stage deployed
Pre-deployment evaluation

Collected metrics
Performance metrics as reported by traffic generators

Project packaging/release
Source code package

Extensibility
Integration of custom tests possible
OPNFV Test Ecosystem

Functional Testing
- RefStack
- Tempest
- Cloudify
- VNF
- Kubernetes

Non-functional Testing
- Yardstick
- Bottlenecks
- StorPerf
- vsperf
  NFVI dataplane Performance

OPNFV Feature Tests
- NFVI, VIM APIs + Functions

Test Result Database
- Dovetail
- OPNFV Compliance Verification

Performance
- NFVI & VNF Performance
- Load tests staging manager
- NFVI Storage Performance

Load tests
- VNF
- Kubernetes

Compliance Verification
**NFVBench**

**Description**
Full stack data plane performance measurements

**Components tested**
Full data plane stack: packet forwarding and virtualization components

**Stage deployed**
Pre-production, performance tuning and monitoring

**Collected metrics**
Metrics reported by T-Rex Project packaging/release
Single self-contained Docker container

**Extensibility**
Wide range of parameters in PVP, PVVP, SR-IOV etc. scenarios
OPNFV Test Ecosystem

- **Functional Testing**
  - OPNFV Feature Tests
    - NFVI, VIM APIs + Functions
    - RefStack (openstack)
    - Tempest (openstack)
    - Cloudify
    - Kubernetes
  - NFVI & VNF Performance
    - Yardstick
  - Load tests staging manager
    - Bottlenecks
  - NFVI dataplane Performance
    - vsperf, NFVBen
  - NFVI Storage Performance
    - StorPerf

- **Non-functional Testing**
  - OPNFV Compliance Verification
  - Dovetail

- **Test Result Database**
**Description**
Performance measurements of block & ephemeral storage at the VM level

**Components tested**
Storage subsystem

**Stage deployed**
Pre-production and lab environment

**Collected metrics**
Performance metrics in steady state, test failed if no stabilization

**Project packaging/release**
Docker container

**Extensibility**
Wide array of parameter: e.g. nr of VMs, queue depth, I/O access pattern
OPNFV Test Ecosystem

**Functional Testing**
- **OPNFV Feature Tests**
  - NFVI, VIM APIs + Functions

**Non-functional Testing**
- **Yardstick**
  - NFVI & VNF Performance
- **Bottlenecks**
  - Load tests staging manager
- **vsperf, NFVBench**
  - NFVI dataplane Performance
- **StorPerf**
  - NFVI Storage Performance

**Test Result Database**

**Dovetail**
- **OPNFV Compliance Verification**

**Compliance Verification**
OPNFV Test Ecosystem

OPNFV Feature Tests

- NFVI, VIM APIs + Functions

Functional Testing

- RefStack (openstack)
- Tempest (openstack)
- Cloudify
- VNF (Cleanwater)
- Kubernetes

Non-functional Testing

- Yardstick: NFVI & VNF Performance
- Bottlenecks: Load tests, staging manager
- vsperf, NFVbench: NFVI dataplane Performance
- StorPerf: NFVI Storage Performance

Test Result Database

Dovetail

OPNFV Compliance Verification

Compliance Verification
OPNFV Compliance Program

• OPNFV Verified Program (OVP) verifies that a commercial cloud platform exposes the same
  – key APIs,
  – behaviors, and
  – characteristics
  as a reference platform **defined through a specific selection of test cases**

• Main objective: Reduce vendor selection and application onboarding cost
  – Establish industry-accepted technical baseline
  – Simplify RFIs and RFPs

• Main components of OVP
  1. OPNFV test frameworks providing the actual OPNFV and upstream test cases
  2. Dovetail: Wrapper for OPNFV test tools and reporting tool
OPNFV Compliance Program

- **Functional Testing**
  - Functest
  - NFVI, VIM APIs + Functions
  - RefStack
  - Tempest
  - Cloudify
  - Kubernetes

- **Non-functional Testing**
  - Yardstick (NFVI & VNF Performance)
  - Bottlenecks (Load tests, staging manager)
  - vsperf, NFV/Bench (NFVI dataplane Performance)
  - StorPerf (NFVI Storage Performance)

- **Compliance**
  - Dovetail (OPNFV Compliance Verification)
  - Test Result Database

- **Utilized by OVP 2018.09**

- **Candidates for later releases**
Addressing emerging use cases
Addressing emerging use cases

• OPNFV traditionally focused on NFVi data center scenarios
  – Medium to large scale deployments in centralized data centers
  – VNFs = legacy Network Functions in VMs

• Emerging use cases impose new requirements on test tools
  – Edge computing
  – Cloud native computing

⇒ How to address those requirements?
Edge Computing

- Impact of edge computing on test tools and methods
  - Test topology
    - Automatic deployment of multiple sites
    - Inter-site connectivity
  - Consideration of networking effects
    - Control and data plane latency
    - Limited bandwidth, jitter, packet drops
  - Hardware resources
    - Limited resources in the edge: 1-4 servers
Virtual Edge in a Box

- **OPNFV XCI**
  - Mini flavor installs OpenStack from master in VMs
  - Can itself be in a VM
  - 2 full OpenStack environments in 1 server
Modeling of Edge Networking Environment

Site 1 VM

Keystone

Network Impairment VM

nic1

Netem delay N ms

nic2

Site 2 VM

Keystone
Cloud Native Computing

- Monolithic App

- Break down into smaller chunks

- Microservice architecture puts functionality into separate services:
  - Iterative development
  - Division of labor
  - Reduce single point of failure
  - Language/deployment flexibility
  - Build different apps using subsets of services
Cloud Native & OPNFV Test Projects

- Consider cloud native for OPNFV test projects
  - Package as micro-services
  - Many are already containerized
    - Functest divided into 8+
  - Add GRPC or REST server interfaces
  - Make actions more atomic within each
  - Orchestrate system level tests using different combinations of services/actions
  - Deploy all OPNFV test services in a single manifest potentially
  - Use tool-chains such as Spinnaker for CI/CD
  - Installer projects are also considering cloud native for some services
Summary

- **Join us!**
  - OPNFV test working group
    - [https://wiki.opnfv.org/display/testing/TestPerf](https://wiki.opnfv.org/display/testing/TestPerf)
  - OPNFV
  - OPNFV Verified
    - [https://www.opnfv.org/](https://www.opnfv.org/)

- Provide feedback and input!
opnfv-users@lists.opnfv.org
#functest
#yardstick #nsb
#bottlenecks
#nfvbench #vsperf
#dovetail
Backup slides
Functest in a nutshell

• Verify any kind of OpenStack and Kubernetes deployments (OPNFV model) or production environments
• Conform with upstream rules (OpenStack gate jobs and Kubernetes conformance tests)
• Ensure that the platforms meet Network Functions Virtualization requirements
Functest suites

- All functional tests as defined by the upstream communities (e.g. Tempest, neutron-tempest-api, Barbican, Patrole...)
- Upstream API and dataplane benchmarking tools (Rally, Vmtpl and Shaker)
- Virtual Network Function deployments and testing (vIMS, vRouter and vEPC)
Yardstick’s Goal is to verify infrastructure compliance from the perspective of a Virtual Network Function (VNF).

Yardstick’s scope is the development of a testing framework, test cases and test stimuli to enable NFVI verification. Yardstick also includes NSB (Network services benchmarking).
Bottlenecks

3. Upstream Develop

1. Classified bottlenecks

2. Feedback bottlenecks

VSPERF

StorPerf

Cperf

Yardstick

DB

Test Cases

Test Results

Network
Storage
Compute
Middleware
APP

Bottlenecks Testing Results

Monitoring

OPNFV Reference Platform

OpenStack
Kubernetes
VSPPerf

- Automated Framework for dataplane performance benchmarking,
  - Switching Technologies with Physical and Virtual Interfaces
- Configuration and control of topology, vswitch, VNF, traffic-generator and other software components are performed by VSPERF.
  - VSPERF provides the user the ability to choose the vswitch, Traffic-generator, VNF, etc.
- VSPERF is used as a tool for optimizing switching technologies, qualifying packet processing components and for pre-deployment evaluation of the NFV platform datapath.
- Virtual Switches:
  - OVS, VPP
- Traffic Generators
  - T-Rex, Spirent, Ixia, Xena, Moongen
- Deployment Scenarios
  - Phy2Phy, PVP, PVVP, Custom.
- VSPERF tests are defined and driven by Level Test Design (LTD) Specification.
  - VSPERF supports designing and implementing custom tests through its 'integration-tests' feature.
- VSPERF supports multiple modes:
  - Ex: Trafficgen-off mode: VSPERF will do setup of DUT, but no control the traffic-generator.
NFV Bench

- Tool that provides an automated way to measure the network performance for the most common data plane packet flows on any OpenStack system.
- Designed to be easy to install and easy to use by non-experts
  - there is no need to be an expert in traffic generators and data plane performance testing.
- The tool is built around the open source T-Rex traffic generator and is useful for testing a full NFVI subsystem that includes ToR switches.
- The key areas of strength for NFVbench are in its automation of the traffic generator, ability to test a full subsystem, and to perform this testing on a production cloud.