Testing Network Softwarization

Pierre Lynch
Lead Technologist, Ixia Solutions Group, Keysight Technologies
Chair, TST WG, ETSI NFV ISG
AGENDA

- Introduction and Background
- Testing Networking in Software
  - New features require new methodologies
- Industry Activities
  - ETSI NFV, Open Source
INTRODUCTION AND BACKGROUND
What’s so hard about that?
ISOLATING THE SYSTEM UNDER TEST (SUT)

Physical Elements

- HSS
- MME
- eNodeB UEs
- SGW/PGW
- PCRF
- Charging

Services
- Internet
• Some VNF functionality has been moved out
• The SUT requires the complete environment in order to run
ISOLATING THE SUT

• Some VNF functionality has been moved out
• The SUT requires the complete environment in order to run
• Test VNFs can be placed to perform simulations
ISOLATING THE SUT

- Some VNF functionality has been moved out
- The SUT requires the complete environment in order to run
- Test VNFs can be placed to perform simulations
- Must maintain the environment constant
TEST UNIT SELECTION AND PLACEMENT

HW or SW-only

HW-Based

SW-Based

© 2018 Keysight Technologies  ALL RIGHTS RESERVED.
**NFVI**

Performance Testing

- **Performance Verification**
  Validating the advertised performance of a SUT

- **Benchmarking**
  Determining the maximum performance of a platform

- **Dimensioning**
  Determining the amount of platform resources required to achieve a performance target
MULTI-TENANCY

- Shared resources change the behavior and predictability
- VNFs can become “noisy neighbors”
- Achieving predictable behavior for other VNFs is the goal
- Platform can become “noisy” too
ACCELERATION

• Various acceleration techniques have appeared
  DPDK
  SR-IOV
  Fast Data - FD.io

• Have various impacts on performance and latency
VNFFG AND NETWORK FORWARDING PATH

- Definition of a network service using VNF Forwarding Graphs
- Establishment of the forwarding path has many options: vSwitch, SDN, etc
- Performance will vary based on networking technologies used
- Slicing!
SCALING

- Dynamic allocation of resources based on trigger criteria
- Criteria can be network, compute or storage based
SCALING

• Dynamic allocation of resources based on trigger criteria
• Criteria can be network, compute or storage based
• Once trigger is reached, resources allocated to instantiate a new VNF-C
• Dynamic allocation of resources based on trigger criteria
• Criteria can be network, compute or storage based
• Once trigger is reached, resources allocated to instantiate a new VNF-C
• Test VNFs placed to simulate the required trigger traffic
  SW or HW configurations
MANO FUNCTIONS

- VNF Package Mgmt
- Onboarding
- VNF Lifecycle Management
- VNF Performance Management
- VNF Fault Management
- NS Lifecycle Management
- NS Performance Management
- NS Fault Management
DEVOPS AND CI/CD

Opportunity!

• DevOps
  Combination of different operational areas into one cohesive service delivery team:
    Dev, QA, Operations, Security, Others as needed
  Typically associated with an Agile delivery process
    Small changes - easier to pinpoint failures

• CI/CD: Continuous Integration and Continuous Delivery/Deployment
  All new system updates and additions are immediately integrated and tested
  Purpose is to find failures immediately (fail fast, fail forward)

• Automation!
  Continuous testing, monitoring and feedback
• SW deployment enables the deployment of test units along with the system
• Sanity tests (and more) can be triggered automatically upon deployment
INDUSTRY ACTIVITIES
Who’s doing what
Paris, 16-18 October 2018
Paris, 16-18 October 2018

ETSI NFV TST
Testing, Experimentation and Open Source
TST001 – PRE-DEPLOYMENT TESTING

• Target audience:
  All companies wanting to validate new SW, SW updates
  CI/CD pipeline

• Content summary
  Definition of SUTs
  Test methods for pre-deployment validation of SUTs
  Pre-deployment validation of NFV Infrastructure
  Pre-deployment validation of VNFs
  Pre-deployment validation of Network Services

ETSI GS NFV-TST 001 V1.1.1 (2016-04)
TST004 – PATH IMPLEMENTATION TESTING

- Guidelines for test plan on path implementation through NFVI
- SUT options
  - Fct placement
  - SDN application type
  - SDN controller type
- Metrics
  - VNFC instantiation time
  - Path instantiation
  - 1st packet latency
  - Subsequent packet latency
  - Std pkt transfer measurements
- Procedures
- Examples
TST007 – GUIDELINES FOR INTEROP TESTING

• Test Descriptions (Test Cases)
• Interoperability Features Statement
  List of all features that need to be supported
  Referenced by the individual Test Descriptions
  VIM, NFVO, VNFM, EM/VNF
• Features taken from IFA documents
  VNF Package Mgmt, LCM, Fault Mgmt,
  Performance Mgmt
  NS LCM, Update, Healing, Termination
• Interoperability Testing Guidelines for NFVI-VIM, MANO and VNF
• Detailed collection of test descriptions for most functionality

ETSI GR NFV-TST 007 V1.1.1 (2017-11)

Network Functions Virtualisation (NFV);
Testing;
Guidelines on Interoperability Testing for MANO
TST009 – NFVI NETWORK BENCHMARKS AND MEASUREMENT METHODS

• Expands the Requirements and Methods of RFC2544
  New reality of NFVI platforms are different than dedicated “boxes” of the past

• Benchmark definition
• Test setups
• Test tool requirements
• Methods of Measurement

ETSİ GS NFV-TST009 V0.0.15 (2018-08)

Network Functions Virtualisation (NFV); Testing;
 Specification of Networking Benchmarks and Measurement
  Methods for NFVI

Rapporteur: Al Morton (AT&T Labs)
### BENCHMARKS

- **Throughput**
  - Offered Load Frame Size
  - Offered Load Step Size
  - Min Trial Repetition Interval
  - Trial Duration
  - Max X% Loss Ratio
  - Max # of Trials

- **Latency**

- **Delay Variation**

- **Loss**

For each Benchmark:

- **Background**
- **Name**
- **Parameters**
- **Scope**
- **Units of Measure**
- **Definition**
- **Units of Measure**
- **Sources of Error**
- **Discussion**
- **Reporting Format**
TEST SETUP EXAMPLES
MITIGATING BACKGROUND PROCESSES THAT CAUSE ERRORS (LOSS)


Because of the nature of NFVI platform
BINARY SEARCH WITH LOSS VERIFICATION

• Goal
  Separate resource exhaustion and loss due to transient processes
  They are dealt with in separate ways

• Solution
  If a trial fails because of loss (< z), run the trial again with the same stimulus (Max (r) = 2)
  Keep trials short to avoid transients
  Isolate loss due to transients
    Run long duration tests to characterize effects and frequency

Prototyped with OPNFV
Showed marked success in repeatability
TST010 – MANO API CONFORMANCE TEST SUITE

• For 3 Reference Points:
  Os-Ma-Nfvo - ETSI GS NFV-SOL 005
  Or-Vnfm - ETSI GS NFV-SOL 003
  Ve-Vnfm - ETSI GS NFV-SOL 002

• Document + Automatable Test Descriptions
  OpenAPIs developed by the SOL WG
  Using the open source Robot Framework

Collaboration with OPNFV:
• Functest Integration
• Potential Instrumented Platform
TST011 – TEST DOMAIN AND DESCRIPTION
LANGUAGE RECOMMENDATIONS

• NFV Test Domain with automation ecosystem
• Recommendations for a DSL (Domain Specific Language)
TST012 - VIM & NFVI CONTROL AND MANAGEMENT PERFORMANCE EVALUATION

• Focus on the control plane performance of VIM + NFVI
• Based on functional requirements in ETSI GS NFV-IFA010
• Potential Metrics:
  - Virtualization container instantiation
  - Scaling
  - Migration
• Delicate!
  - VNFs can impact these metrics
  - Care will be taken to define the metrics and methods to be independent of VNF (maybe use standard samples)

Rapporteur: Huang Cheng (Huawei)
3RD NFV PLUGTESTS & OPNFV PLUGFEST
CROSS-COMMUNITY SYNERGY & JOINT ACTIVITIES

• Several cross-community activities:
  • TST009 - Testing specification of networking benchmarks and measurement methods for NFVI
  • TST010 - API conformance testing specification
  • NFVI validation track (Dovetail)
  • OSM integration in OPNFV XCI
  • NSH based SFC Testing
RESULTS HIGHLIGHTS

• Most of the interop testing focused on multi-vendor NS
  • More Test Cases run in fewer (but longer) Test Sessions
  • Interop rates similar to January (slightly higher)

• +125% of API testing

• +175% of automated interop testing

• Learnings to guide TST010 NFV Conformance Testing
OVERALL RESULTS

Table 27a: IOP Overall Results

<table>
<thead>
<tr>
<th>Overall Results</th>
<th>Number of Test Sessions</th>
<th>Interoperability (TCs Run)</th>
<th>TCs Not Run</th>
<th>TCs Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>OK</td>
<td>NA</td>
<td>Run</td>
</tr>
<tr>
<td>Overall Results</td>
<td></td>
<td>NA</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 27b: API Track Overall Results

<table>
<thead>
<tr>
<th>Overall Results</th>
<th>Number of Test Sessions</th>
<th>API Validation (TCs Run)</th>
<th>TCs Not Run</th>
<th>TCs Totals</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>OK</td>
<td>NA</td>
<td>Run</td>
</tr>
<tr>
<td>Overall Results</td>
<td></td>
<td>NA</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
OPNFV: UPSTREAM FIRST

- OPNFV is an integration project, incorporating other open source components to create a platform for NFV
- Most of the development in OPNFV is actually on upstream projects
<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pharos</td>
<td>Infrastructure - Federated labs for CI and feature testing</td>
<td>12+ labs available</td>
</tr>
<tr>
<td>Functest</td>
<td>Feature testing (production deployment)</td>
<td>Umbrella test project with database/API and dashboard</td>
</tr>
<tr>
<td>Yardstick</td>
<td>Framework with generic test cases for NFVI (compute, storage and networking)</td>
<td>Framework and methodology for other test projects to plugin</td>
</tr>
<tr>
<td>VSPERF</td>
<td>Virtual switch benchmarking</td>
<td>Methodology and tools</td>
</tr>
<tr>
<td>CPERF</td>
<td>Controller performance benchmarking</td>
<td></td>
</tr>
<tr>
<td>Qtip</td>
<td>Platform component benchmarking</td>
<td>Compute, storage, NW tests</td>
</tr>
<tr>
<td>Storeperf</td>
<td>Tool to measure block and object storage performance of an NFVI</td>
<td></td>
</tr>
<tr>
<td>Bottlenecks</td>
<td>Framework to look for system constraints</td>
<td></td>
</tr>
<tr>
<td>NFVBench</td>
<td>L2/L3 forwarding performance</td>
<td>Black Box approach using open source tools (T-Rex)</td>
</tr>
</tbody>
</table>
OPNFV YARDSTICK

• Framework and methodology for NFVI testing
OPNFV YARDSTICK (CONT)

• Part of CI pipeline – daily execution
• Test cases
  Generic Test Cases for NFVI verification (compute, storage, network)
  Test cases for OPNFV Projects: HA, SFC, SDNVPN, IPVSIX, VNFFG, KVM
  Execute Yardstick Generic Test cases for OVS4NFV, ARMBAND
CROSS-COMMUNITY CI (XCI)
SUMMARY

• New considerations for testing methodologies
  SUT Isolation
  Test Unit Selection and Placement

• New functionality
  Multi-Tenancy
  Acceleration
  VNFFG & Network Forwarding Path, Slicing
  Scaling
  MANO

• New possibilities
  DevOps and CI/CD

• SDOs and Open Source very active!