VNF Performance Validation and Onboarding

© 2019 Aarna Networks, Inc.
Objectives

Demonstrate the full life cycle of a sample VNF using OPNFV and ONAP:

- Characterizing a VNF’s performance by using OPNFV Yardstick NSB
- Onboarding the VNF onto MANO software (ONAP)
- Deploying the VNF using ONAP, onto OPNFV scenario using Aarna Networks’ ONAP distribution (ANOD)
Performance validation

Challenges: Comparing VNF performance is a lot more difficult than before for the following reasons:

- Differences between vendor environments
- Differences from a real CSP environment

The OPNFV Yardstick Network Service Benchmarking (NSB) tool is useful in solving the above problem.
ONAP Within the Overall Stack

ONAP requires integration and design before it can generate value.
L2 Forwarder Network Topology

```
<table>
<thead>
<tr>
<th></th>
<th>192.168.10.x/24</th>
<th>10.1.1.x/24</th>
<th>10.1.0.x/24</th>
<th>10.0.1.x/24</th>
</tr>
</thead>
<tbody>
<tr>
<td>External</td>
<td>l2fwd__0</td>
<td>Downlink</td>
<td>Uplink</td>
<td>Mgmt</td>
</tr>
<tr>
<td>tg__0</td>
<td>Up</td>
<td>Downlink</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

© 2019 Aarna Networks, Inc.
Onboarding onto ONAP

- Create Openstack Cloud image with required software (prox/DPDK library etc.)
- Create required resources on Openstack (Flavor, Networks/Subnets)
- Create HEAT templates and environment files
  - Method 1: Create VNF HEAT templates from scratch
  - Method 2: Use the HEAT template file generated by NSB as part of its execution
  Note: We have used Method 1, but Method 2 can be done as well.
  Note: In Method 1, we created 2 VNFs - Traffic Generator (TG) and L2 forwarder (L2FWD), and 1 Network Service
- Add the required scripts in cloud-init section of VNF’s YAML files
  - In case of L2FWD, this requires running prox utility on both VNFs
- Onboard the ZIP files in SDC, and create VSP’s & NS
Deployment using ONAP

- Run SDN-C Preload scripts that set the required parameters in SDN-C
- Instantiate VNF L2FWD, which creates the VM and runs prox process with appropriate configuration file
- Instantiate VNF TG, which creates the VM and runs prox process with traffic generator options
End-to-end orchestration needs to support EPA/HPA

① Understand NFVI (cloud) attributes
② Read VNF requirements
③ Place VNF in the right cloud
④ Consume EPA/HPA

Note: Not shown in this Demo
Goal: Help VNF vendors and network service designers to figure out right hardware requirements for various network functions

How:

1. Enhance NSB to output hardware features and configuration of hardware features to meet SLA such as throughput, latency and jitter using NSB.
2. Auto-creation of VNF hardware requirements (VNFD) based on the output of previous step.

*Note: Inputs from Intel*
Thank You!!!