Developing ONAP Environment for COP (Certified ONAP Professional) Exam

23 April 2020

Vivekanandan Muthukrishnan

vmuthukrishnan@aarnanetworks.com

Agenda

- Requirements for COP Environment
- Challenges
- Future recommendations

Note: We will not talk about specific items in the COP

Platform Requirements

- Each candidate taking a test gets a dedicated environment
  - Single server/VM with nested VMs
  - VM Cluster for ONAP
  - VM for Openstack
  -
- The environment should be
  - Fully provisioned
  - ONAP is installed and fully functional
  - All workarounds (if any) should be run
  - Access to all publicly available material (Wiki pages etc.)

Test Requirements

- Each item in the test may require different setup (starting point), so the provisioning has to be different.
- Some test items may need part of a step (eg., partial creation of an artifact), so that the candidate completes the remaining
- Needs fine control over what needs to be done
- All steps need to be automated (details in next few slides)

Automation Requirements

- Step 1: Preparing for a test item:
  - For each test item, all the required environment (for solving the particular item) should be fully provisioned (eg., distributing services, deploying NS).

- Step 2: Performing the test item:
  - This is done by the candidate but we need to automate the steps (for comparison)
  - For each test item, whatever the candidate does manually, should be automated (to verify the test results)

Automation Requirements

- Step 3: Validation:
  - For each test item, the validation process should be automated (which will be run after candidate completes the item).

- Step 4: Clean up:
  - After each test item, the state should be restored before provisioning for next test item.

Provisioning Automation

- Deploy ONAP and verify all services are up
  - Apply required patches & workarounds
  - Health check of all components

- For each test item, before candidate starts, the environment should be fully prepared for that particular test

- This requires automating several steps that are typically not readily available in the community (eg., forcing some workflows to fail)

Test Automation

- Automate the same steps that the candidate does manually
  - Design time activities
  - Runtime activities

Validation Automation

- This involves validating the item after it is completed by the candidate
  - Design time & Run time
- The final state of the environment (after test) should be compared with expected state of the system
  - Collect and verify all test items with expected results

Challenges

- Constrained environment (single server/VM on the cloud) with all ONAP components and Openstack
  - Cost should be taken into consideration
- Repeatability
  - Each deployment should result in exactly same state for each candidate
- Predictability
  - There should be no surprises for the candidate when the test is started (eg., no option to run workarounds)
- Stability
  - The environment should stay healthy while the test is being taken by the candidate
- It is not trivial to switch to new releases, so all bug fixes have to be backported!

Future Recommendations

- Avoid SSL certificate expiry issues, since the Certification platform may not be frequently upgraded to newer releases.
  - Option to bypass certificate validation
- Deploy only a subset of components for each test item (minimum requirement).
- Consider the possibility of a smaller sandbox with specific components
  - This will reduce both the cost of the environment as well as the complexity
- All test item developers could prepare and present a demo of their item on the specific release on which the test is planned
  - Few items could not be completed and replaced/rewritten since the functionality was broken on Dublin release

THANK YOU !