E2E Automation vLB & vFW w/ CDS Use Case - ONAP-01-Installation-Required Component (El Alto)

ONAP Dublin Deployment Components necessary for the vFW & VLB use case with CDS:

In order to run the VLB with CDS use case, we need an ONAP Dublin Release instance that includes the following components::

ONAP Component Name	Description	
A&AI	Required for **Inventory** Cloud Owner, Customer, Owning Entity, Service, Generic VNF, VF Module	
SDC	VSP , VF, and Service **Modeling of the VLB**	
DMAAP	**Distribution of the CSAR** to all ONAP components	
SO	Requires for **Macro Orchestration using the generic building blocks**.	
SDNC needs to include: • CDS UI mS • CDS Blueprint Processor mS • CDS Controller Blueprint mS • Netbox • Naming Generation mS	Used for ONAP E2E **Zero Touch Declarative Provisioning & Configuration Management for VNF/CNF /PNF.**	
Policy	Used to Store **Naming Policy**	
VID	Used for **User Interface** for Run Time Execution of the Marco Instantiation flow.	
AAF	Used for Authentication and Authorization of requests	
Portal	Used for accessing the ONAP Components GUI like SDC, VID, etc ...	
Robot	Used for running automated tasks, like provisioning cloud customer, cloud region, service subscription, etc ...	
Shared Cassandra DB	Used as a **shared storage** for ONAP components that rely on Cassandra DB, like **AAI**	
Shared Maria DB	Used as a **shared storage** for ONAP components that rely on Maria DB, like **SDNC, and SO**	

ONAP Deployment Guide:

In order to deploy such an instance, we can follow the ONAP deployment guide in this link: https://docs.onap.org/en/elalto/submodules/oom.git/docs/oom_quickstart_guide.html#quick-start-label

As we can see from the guide, we can use an override file that helps us customize our ONAP deployment, without modifying the OOM Folder, so you can download this override file here, that includes the necessary components mentioned above.

override.yaml for vDNS

```
# Copyright © 2019 Amdocs, Bell Canada, Orange
#
# Licensed under the Apache License, Version 2.0 (the "License");
# you may not use this file except in compliance with the License.
# You may obtain a copy of the License at
#
#       http://www.apache.org/licenses/LICENSE-2.0
#
# Unless required by applicable law or agreed to in writing, software
# distributed under the License is distributed on an "AS IS" BASIS,
# WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
# See the License for the specific language governing permissions and
# limitations under the License.

################################################################
# Global configuration overrides.
```

```
#
# These overrides will affect all helm charts (ie. applications)
# that are listed below and are 'enabled'.
################################################################
global:
  # Change to an unused port prefix range to prevent port conflicts
  # with other instances running within the same k8s cluster
  nodePortPrefix: 302
  nodePortPrefixExt: 304

  # ONAP Repository
  # Uncomment the following to enable the use of a single docker
  # repository but ONLY if your repository mirrors all ONAP
  # docker images. This includes all images from dockerhub and
  # any other repository that hosts images for ONAP components.
  #repository: nexus3.onap.org:10001

  # readiness check - temporary repo until images migrated to nexus3
  readinessRepository: oomk8s
  # logging agent - temporary repo until images migrated to nexus3
  loggingRepository: docker.elastic.co

  # image pull policy
  pullPolicy: IfNotPresent

  # override default mount path root directory
  # referenced by persistent volumes and log files
  persistence:
    mountPath: /dockerdata-nfs

  # flag to enable debugging - application support required
  debugEnabled: true

################################################################
# Enable/disable and configure helm charts (ie. applications)
# to customize the ONAP deployment.
################################################################
aaf:
  enabled: true
aai:
  enabled: true
  global:
    cassandra:
      replicas: 1
  aai-cassandra:
    replicaCount: 1
appc:
  enabled: false
cassandra:
  enabled: true
  replicaCount: 1
clamp:
  enabled: false
cli:
  enabled: false
consul:
  enabled: false
contrib:
  enabled: true
dcaegen2:
  enabled: false
dmaap:
  enabled: true
esr:
  enabled: false
log:
  enabled: false
  log-logstash:
    replicaCount: 1
sniro-emulator:
  enabled: false
```

```
oof:
  enabled: false
mariadb-galera:
  enabled: true
msb:
  enabled: false
multicloud:
  enabled: false
nbi:
  enabled: false
policy:
  enabled: true
  config:
    preloadPolicies: true
pomba:
  enabled: false
portal:
  enabled: true
robot:
  enabled: true
  config:
    openStackEncryptedPasswordHere: <openStackEncryptedPassword>
  openStackFlavourMedium: <openStackFlavourMedium>
  openStackKeyStoneUrl: <openStackKeyStoneUrl>
  openStackPublicNetId: <openStackPublicNetId>
  openStackPassword: <openStackPassword>
  openStackRegion: <openStackRegion>
  openStackTenantId: <openStackTenantId>
  openStackUserName: <openStackUserName>
  ubuntu14Image: <ubuntu14Image>
  ubuntu16Image: <ubuntu16Image>
  openStackPrivateNetId: <openStackPrivateNetId>
  openStackSecurityGroup: <openStackSecurityGroup>
  openStackPrivateSubnetId: <openStackPrivateSubnetId>
  openStackPrivateNetCidr: "10.0.0.0/8"
  vnfPubKey: <vnfPubKey>

sdc:
  enabled: true
  global:
    cassandra:
      replicaCount: 1
sdnc:
  enabled: true

  replicaCount: 1

  mysql:
    replicaCount: 1


so:
  enabled: true

  replicaCount: 1

  liveness:
    # necessary to disable liveness probe when setting breakpoints
    # in debugger so K8s doesn't restart unresponsive container
    enabled: true

  # so server configuration
  config:
    # message router configuration
    dmaapTopic: "AUTO"
    # openstack configuration
    openStackUserName: <openStackUserName>
    openStackRegion: <openStackRegion>
    openStackKeyStoneUrl: <openStackKeyStoneUrl>
    openStackServiceTenantName: <openStackServiceTenantName>
    openStackEncryptedPasswordHere: <openStackEncryptedPasswordHere>
```

```
  # configure embedded mariadb
  mariadb:
    config:
      mariadbRootPassword: password

uui:
  enabled: false
vfc:
  enabled: false
vid:
  enabled: true
vnfsdk:
  enabled: false
```

As we can see in the override.yaml file above, we can enable or disable the deployment of a specific ONAP component, and in our case, we enabled only the necessary components to run the VLB demo.

We can also see that we can configure the parameters needed for the use case to run, like OpenStack Username, Passowrd, Region, Tenant, Image Names, and Flavors.

We have also enable "PreloadPolicies", so that the Naming Policy is deployed when ONAP POLICY components starts.

Post Deployment:

After completing the first part above, we should have a functional ONAP deployment for the Dublin Release.

We will need to apply a few modifications to the deployed ONAP Dublin instance in order to run the VLB use case:

Initialize cloud account:

Our use case will need a customer, service subscription, cloud-region, tenant, and complex.

We can initialize this information by running the robot script "./demo-k8s.sh onap init_customer".

Robot initialize cloud account

```
root@olc-dublin-rancher:~# cd oom/kubernetes/robot/
root@olc-dublin-rancher:~/oom/kubernetes/robot# ./demo-k8s.sh onap init_customer
Number of parameters:
2
KEY:
init_customer
++ kubectl --namespace onap get pods
++ sed 's/ .*//'
++ grep robot
+ POD=onap-robot-robot-6dd6bfbd85-nl29n
+ ETEHOME=/var/opt/ONAP
++ kubectl --namespace onap exec onap-robot-robot-6dd6bfbd85-nl29n -- bash -c 'ls -1q /share/logs/ | wc -l'
+ export GLOBAL_BUILD_NUMBER=3
+ GLOBAL_BUILD_NUMBER=3
++ printf %04d 3
+ OUTPUT_FOLDER=0003_demo_init_customer
+ DISPLAY_NUM=93
+ VARIABLEFILES='-V /share/config/vm_properties.py -V /share/config/integration_robot_properties.py -V /share
/config/integration_preload_parameters.py'
+ kubectl --namespace onap exec onap-robot-robot-6dd6bfbd85-nl29n -- /var/opt/ONAP/runTags.sh -V /share/config
/vm_properties.py -V /share/config/integration_robot_properties.py -V /share/config
/integration_preload_parameters.py -d /share/logs/0003_demo_init_customer -i InitCustomer --display 93
Starting Xvfb on display :93 with res 1280x1024x24
Executing robot tests at log level TRACE
==============================================================================
Testsuites
==============================================================================
Testsuites.Demo :: Executes the VNF Orchestration Test cases including setu...
==============================================================================
Initialize Customer                                                   | PASS |
------------------------------------------------------------------------------
Testsuites.Demo :: Executes the VNF Orchestration Test cases inclu... | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
==============================================================================
Testsuites                                                            | PASS |
1 critical test, 1 passed, 0 failed
1 test total, 1 passed, 0 failed
==============================================================================
Output:  /share/logs/0003_demo_init_customer/output.xml
Log:     /share/logs/0003_demo_init_customer/log.html
Report:  /share/logs/0003_demo_init_customer/report.html
root@olc-dublin-rancher:~/oom/kubernetes/robot#
```

SO Orchestration Transition Table Updates:

```
INSERT INTO `catalogdb`.`orchestration_status_state_transition_directive` (`RESOURCE_TYPE`,
`ORCHESTRATION_STATUS`, `TARGET_ACTION`, `FLOW_DIRECTIVE`)
VALUES
('VNF', 'CONFIGURED', 'ACTIVATE', 'CONTINUE');
```

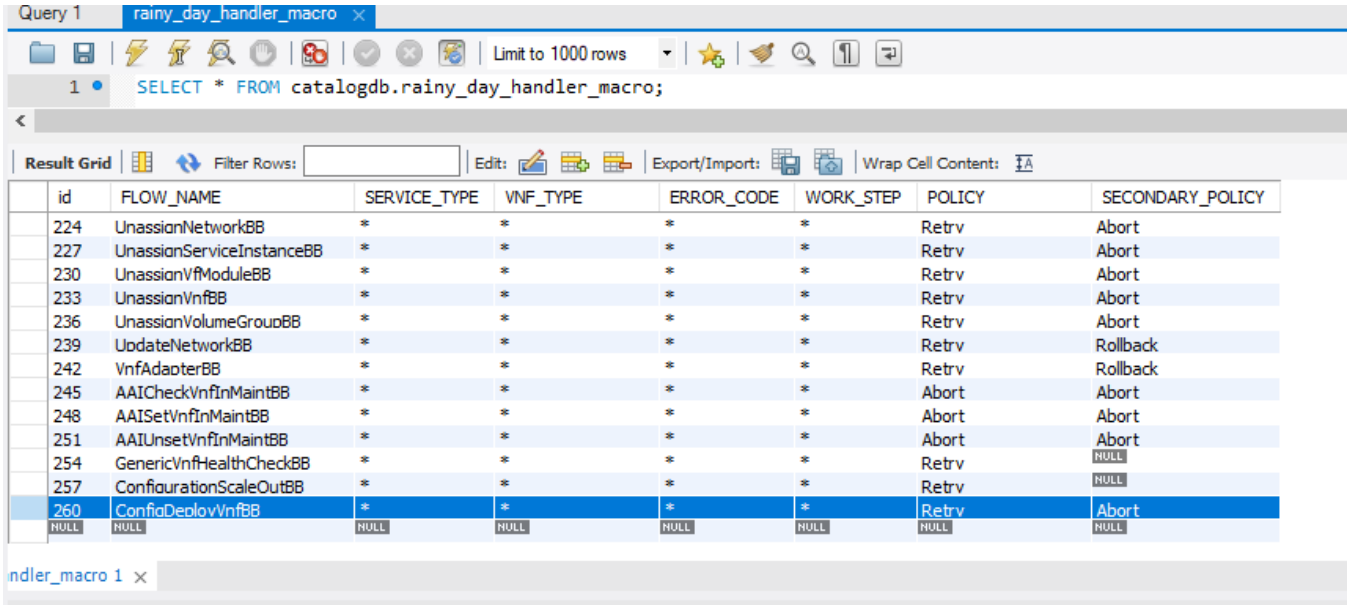
Retry for ConfigDeployVnfBB:

The vLB VM will download source code and will build the netconf simulator that will receive the configuration updates from CDS. This process might take more than 5 minutes.

Due to a default 5 minute timeout value in Camunda BPMN Execution Engine, this means the ConfigDeployVnfBB might fail, which will break the E2E automation of the instantiation.

Thus, we need to tell SO that if ConfigDeployVnfBB fails, it should retry ConfigDeployVnfBB.

We can do this by adding an entry in the SO CATALOG DB rainy_day_handler_macro table, as shown below:



We can do this by connecting to the SO CATALOG DB, and running the following SQL command:

update rainy_day_handler_macro

```
INSERT INTO rainy_day_handler_macro (FLOW_NAME, SERVICE_TYPE, VNF_TYPE, ERROR_CODE, WORK_STEP, POLICY,
SECONDARY_POLICY)
VALUES
('ConfigDeployVnfBB', '*', '*', '*', '*' , 'Retry', 'Abort');
```

Also, in order to allow SO to retry the ConfigDeployVnfBB if it fails the 1st time, we can update the override.yaml file for the SO BPMN INFRA chart in OOM like below:

override.yaml for so-bpmn-infra chart

```
mso:
  rainyDay:
    retryDurationMultiplier: 2
    maxRetries: 5
```

Naming Policy:

The override.yaml file above has an option "preload=true", that will tell the POLICY component to run the push_policies.sh script as the POLICY PAP pod starts up, which will in turn create the Naming Policy and push it.

To check that the naming policy is created and pushed OK, we can run the commands below.

SDNC Naming policy

```
bash-4.4$ curl -k --silent -X POST --header 'Content-Type: application/json' --header 'ClientAuth:
cHl0aG9uOnRlc3Q=' --header 'Authoment: TEST' -d '{ "policyName": "SDNC_Policy.
Config_MS_ONAP_VNF_NAMING_TIMESTAMP.1.xml"}' 'https://pdp:8081/pdp/api/getConfig'
[{"policyConfigMessage":"Config Retrieved! ","policyConfigStatus":"CONFIG_RETRIEVED","type":"JSON","config":"{\"
service\":\"SDNC-GenerateName\",\"version\":\"CSIT\",\"content\":{\"policy-instance-name\":\"ONAP_VNF
_NAMING_TIMESTAMP\",\"naming-models\":[{\"naming-properties\":[{\"property-name\":\"AIC_CLOUD_REGION\"},{\"
property-name\":\"CONSTANT\",\"property-value\":\"ONAP-NF\"},{\"property-name\":\"TIMESTAMP\"},{\"property-
value\":\"_\",\"property-name\":\"DELIMITER\"}],\"naming-type\":\"VNF\",\"naming-recipe\":\"
AIC_CLOUD_REGION|DELIMITER|CONSTANT|DELIMITER|TIMESTAMP\"},{\"naming-properties\":[{\"property-name\":\"
VNF_NAME\"},{\"property-name\":\"SEQUENCE\",\"increment-sequence\":{\"max\":\"zzz\",\"scope\":\"ENTIRETY\",\"
start-value\":\"001\",\"length\":\"3\",\"increment\":\"1\",\"sequence-type\":\"alpha-numeric\"}},{\"property-
name\":\"NFC_NAMING_CODE\"},{\"property-value\":\"_\",\"property-name\":\"DELIMITER\"}],\"naming-type\":\"
VNFC\",\"naming-recipe\":\"VNF_NAME|DELIMITER|NFC_NAMING_CODE|DELIMITER|SEQUENCE\"},{\"naming-properties\":[{\"
property-name\":\"VNF_NAME\"},{\"property-value\":\"_\",\"property-name\":\"DELIMITER\"},{\"property-name\":\"
VF_MODULE_LABEL\"},{\"property-name\":\"VF_MODULE_TYPE\"},{\"property-name\":\"SEQUENCE\",\"increment-
sequence\":{\"max\":\"zzz\",\"scope\":\"PRECEEDING\",\"start-value\":\"01\",\"length\":\"3\",\"increment\":\"
1\",\"sequence-type\":\"alpha-numeric\"}}],\"naming-type\":\"VF-MODULE\",\"naming-recipe\":\"
VNF_NAME|DELIMITER|VF_MODULE_LABEL|DELIMITER|VF_MODULE_TYPE|DELIMITER|SEQUENCE\"}]}}","policyName":"SDNC_Policy.
Config_MS_ONAP_VNF_NAMING_TIMESTAMP.1.xml","policyType":"MicroService","policyVersion":"1","matchingConditions":
{"ECOMPName":"SDNC","ONAPName":"SDNC","service":"SDNC-GenerateName"},"responseAttributes":{},"property":null}]
```

In case the policy is missing, we can manually create and push the SDNC Naming policy. E2E Automation vLB w/ CDS Use Case - ONAP-02-Design Time (El Alto)#1091812835

Network Naming mS: Remove data from EXTERNAL_INTERFACE database

naming db

from rancher remove

```
root@sb04-rancher:~# kubectl -n onap get pod |grep neng
dev-sdnc-nengdb-0 1/1 Running 0 21d

root@sb04-rancher:~# kubectl -n onap exec -it dev-sdnc-nengdb-0 bash
bash-4.2$ mysql -unenguser -pnenguser123
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 373324
Server version: 10.1.24-MariaDB MariaDB Server

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> show DATABASES;
+--------------------+
| Database |
+--------------------+
| information_schema |
| nengdb |
+--------------------+
2 rows in set (0.00 sec)

MariaDB [(none)]> use nengdb
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MariaDB [nengdb]> show tables;
+-----------------------+
| Tables_in_nengdb |
+-----------------------+
| DATABASECHANGELOG |
| DATABASECHANGELOGLOCK |
| EXTERNAL_INTERFACE |
| GENERATED_NAME |
| IDENTIFIER_MAP |
| NELGEN_MESSAGE |
| POLICY_MAN_SIM |
| SERVICE_PARAMETER |
+-----------------------+
8 rows in set (0.00 sec)


MariaDB [nengdb]> delete from EXTERNAL_INTERFACE;
Query OK, 0 rows affected (0.00 sec)

MariaDB [nengdb]>
```