

# VESMapper (BBS Usecase)

- [Overview](#)
- [Blueprint/model/image](#)
- [Deployment Prerequisite/dependencies](#)
- [Deployment Steps](#)
- [Validation](#)
- [Functional testing of VES-Mapper individually](#)

## Overview

VES-Mapper microservice of DCAE can help to collect telemetry data / events / alerts from VNFs and collectors of DCAE and convert them into VES compliant events. It supports both VES-5.4 and VES-7.0 specifications.

Based on the configuration, the VES-Mapper subscribes to non-VES notifications published on the DMaaP. By using mapping files, the mapper converts the received notifications into VES events and publish them back on DMaaP on the configured topic. It uses a Java framework called Smooks ([smooks.org](https://smooks.org)) for the mapping files.

In case of BroadBand Service (BBS), VES-Mapper will receive the PNF registration notifications from DMaaP that are published by RestConf Collector, convert them into VES-7.0 events and publish them back on DMaaP for PNF registration handler service to pick them.

## Blueprint/model/image

- VES-Mapper blueprint is available @ <https://gerrit.onap.org/r/gitweb?p=dcaegen2/services/mapper.git;a=blob;f=UniversalVesAdapter/dpo/blueprints/k8s-vesmapper.yaml-template.yaml;h=refs/head/master>
- VES-Mapper docker image is available in Nexus repo @ [nexus3.onap.org:10001/org.onap.dcaegen2.services.mapper.vesadapter.universalesadaptor:1.0.0-SNAPSHOT](https://nexus3.onap.org/repository/org.onap.dcaegen2.services.mapper.vesadapter.universalesadaptor/1.0.0-SNAPSHOT)

## Deployment Prerequisite/dependencies

VES-Mapper can be deployed individually though it will throw errors if it can not reach to DMaaP instance's APIs. To functionally test the mapper, only DMaaP is needed.

## Deployment Steps

- Transfer blueprint file in DCAE bootstrap POD under /blueprints directory
- Log-in to the DCAE bootstrap POD's main container

### To Deploy::

1. Validate blueprint

```
cfy blueprints validate /blueprints/k8s-vesmapper.yaml-template.yaml
```

2. Upload validated blueprint

```
cfy blueprints upload -b ves-mapper /blueprints/k8s-vesmapper.yaml-template.yaml
```

3. Create Cloudify Deployment

```
cfy deployments create -b ves-mapper ves-mapper
```

4. Deploy component

```
cfy executions start -d ves-mapper install
```

## To Undeploy::

1. Uninstall component

```
cfy uninstall ves-mapper
```

2. Delete blueprint

```
cfy blueprints delete ves-mapper
```

## Validation

1. Check that VES-Mapper POD is created and it is in running state

```
kubect1 get pods -n onap | grep mapper  
dep-s36f1a2fadd614a42872bd680516e271a-dcae2-svc-mapper-l2cms 1/1 Running 0 1d
```

2. Check the logs. Here are some important log statements that could be looked for.

```
kubect1 logs -n onap dep-s36f1a2fadd614a42872bd680516e271a-dcae2-svc-mapper-l2cms | more  
11:10:12.078 [main] INFO debugLogger - Triggering controller's start url  
11:10:12.078 [main] INFO debugLogger - Running curl command for url:http://localhost:8085/start  
11:10:12.186 [http-nio-8085-exec-1] INFO o.a.c.c.C.[Tomcat].[localhost].[/] - Initializing Spring  
DispatcherServlet 'dispatcherServlet'  
11:10:12.186 [http-nio-8085-exec-1] INFO o.s.web.servlet.DispatcherServlet - Initializing Servlet  
'dispatcherServlet'  
11:10:12.200 [http-nio-8085-exec-1] INFO o.s.web.servlet.DispatcherServlet - Completed initialization in  
14 ms  
11:10:12.243 [http-nio-8085-exec-1] INFO metricsLogger - UniversalVesAdapter Application starting...  
11:10:12.243 [http-nio-8085-exec-1] INFO debugLogger - Creating Subscriber and Publisher with  
creator.....  
...  
...  
...  
11:10:13.219 [http-nio-8085-exec-1] INFO debugLogger - Created scriber topic:unauthenticated.  
DCAE_RCC_OUTPUTpublisher topic:unauthenticated.VES_PNFREG_OUTPUT  
11:10:13.230 [rcc-notification] INFO metricsLogger - fetch and publish from and to Dmaap started:rcc-  
notification  
...  
...  
...  
11:10:13.322 [http-nio-8085-exec-1] INFO debugLogger - Created scriber topic:unauthenticated.ONAP-  
COLLECTOR-SNMPTRAPPublisher topic:unauthenticated.SEC_FAULT_OUTPUT  
11:10:13.326 [snmp-notification] INFO metricsLogger - fetch and publish from and to Dmaap started:snmp-  
notification  
...  
...  
...  
11:13:11.873 [snmp-notification] DEBUG debugLogger - number of messages to be converted :0  
11:13:13.044 [rcc-notification] DEBUG debugLogger - number of messages to be converted :0
```

## Functional testing of VES-Mapper individually

1. Send a PNF registration notification on behalf of RestConf Collector by calling DMaaP API

```
curl -v -k -H 'Content-Type:application/json' -POST http://message-router.onap.svc.cluster.local:3904  
/events/unauthenticated.DCAE_RCC_OUTPUT -d @rcc_noti.json
```

[rcc\\_noti.json](#)

2. Invoke DMaaP API to fetch message on the topic `unauthenticated.VES_PNFREG_OUTPUT` . VES-Mapper should have published converted VES event on this topic.

```
curl -X GET 'http://message-router.onap.svc.cluster.local:3904/events/unauthenticated.VES_PNFREG_OUTPUT/grp2/con2?timeout=1600'
```

#### Converted VES event

3. VES-Mapper logs

```
12:35:27.886 [rcc-notification] DEBUG debugLogger - number of messages to be converted :1
12:35:27.889 [rcc-notification] INFO debugLogger - Retrieved JSONArray from Collector Config File
12:35:27.889 [rcc-notification] INFO debugLogger - returning identifier array from Collector Config
12:35:27.890 [rcc-notification] DEBUG debugLogger - Using Default Mapping file as Mapping file is not
available for Enterprise Id / identifier ID:2541256
12:35:28.075 [rcc-notification] INFO debugLogger - Transforming incoming json
12:35:28.114 [rcc-notification] INFO debugLogger - Context created
12:35:28.125 [rcc-notification] DEBUG debugLogger - conversion successful to VES Event
12:35:28.125 [rcc-notification] INFO debugLogger - Incoming json transformed to VES format successfully:
rcc-notification
12:35:28.137 [rcc-notification] INFO debugLogger - Serialized VES json
12:35:28.138 [rcc-notification] DEBUG debugLogger - Batch Queue capacity exceeds messages size. Flushing
of all pending messages to DMaaP MR Publisher Topic.
12:35:28.138 [rcc-notification] DEBUG debugLogger - Force publishing messages to DMaaP MR Topic.
Messages Size: 1
12:35:28.138 [rcc-notification] DEBUG debugLogger - DMaaP MR Publisher Authentication is disabled as
username or password is not present.
12:35:28.167 [rcc-notification] DEBUG debugLogger - DMaaP MR Messages published successfully. DMaaP
Response Code: 200. DMaaP Response Body: {
  "serverTimeMs": 0,
  "count": 1
}, Number of Messages published: 1
12:35:28.168 [rcc-notification] INFO metricsLogger - Message successfully published to DMaaP Topic-
{"event":{"commonEventHeader":{"domain":"pnfRegistration","eventId":"registration_1556627728115","
eventName":"pnfRegistration","eventType":"pnfRegistration","lastEpochMicrosec":1.556627728115E12,"
priority":"Normal","reportingEntityName":"VESMapper","sequence":0,"sourceName":"Huawei-HWTCC01B7504","
startEpochMicrosec":1.556627728115E12,"version":"4.0.1","vesEventListenerVersion":"7.0.1"},"
pnfRegistrationFields":{"additionalFields":{"remote-id":"some-remote-id","attachment-point":"
NE=167772165-5-1","cvlan":"1005","svlan":"100"},"lastServiceDate":"somestring","manufactureDate":"","
modelName":" somestring","oamV4IpAddress":"somestring","oamV6IpAddress":"somestring","
pnfRegistrationFieldsVersion":"2.0","serialNumber":"HWTCC01B7504","softwareVersion":"v1","unitFamily":"
somestring","unitType":"somestring","vendorName":"Huawei"}}}}
```