

HPA Beijing Testing

- TESTING PROGRESS: 95%
- Pairwise:
 - Updates:
 - Thursday 5/31/18
 - Tuesday 5/29/18
 - Wednesday 5/23/18
 - Tuesday 5/22/18
 - Thursday 5/17/18
 - Wednesday 5/16/18
 - Tuesday 5/15/18
 - Monday 5/14/18
 - Friday 5/11/18
- Integration:
 - vCPE Regression (ensure HPA changes do not break vCPE)
 - vCPE + HPA
 - vCPE + HPA Integration Test Plan
 - Updates:
 - Thursday 5/31/18
 - Tuesday 5/29/18
 - Wednesday 5/16/18
- HPA Updated CSAR's
- Postman Collection

TESTING PROGRESS: 95%

Pairwise:

Dependencies: OOF, Multicloud, SO, Policy, AAI

Components	Status
AAI – Multicloud	Done - Working
OOF – AAI	Done - Working
OOF – SO	Done - Working
OOF – Policy	Done - Working per Ankit

HPA is unique in that it relies on a multitude of projects and OOF is new and untested. Most of HPA work is gated by OOF, but we are doing our best to help facilitate OOF testing.

Updates:

Thursday 5/31/18

We were able to send a homing request to OOF and receive HPA flavor labels. Solving the last issue from OOF. See

[SO-661](#) - Getting issue details...

STATUS

Tuesday 5/29/18

Fixed [SO-573](#) - Getting issue details...

STATUS

 - we are able to send homing request and get correct homing request back.

Wednesday 5/23/18

OOF - SO, Fixing some API conformance issues, for example OOF does not support License Homing for Beijing, removed from the homing call <http://gerrit.onap.org/r/#/c/48825/>, retesting and working through [SO-573](#) - Getting issue details...

STATUS

Tuesday 5/22/18

OOF - SO, We have had multiple working meetings since last week and have made a lot of progress. We got SB01 configured and have been running tests, filing bugs and fixing bugs and retesting.

SO Issues: [SO-637](#) - Getting issue details... [STATUS](#) [SO-636](#) - Getting issue details... [STATUS](#)

OOF Issues: [OPTFRA-239](#) - Getting issue details... [STATUS](#) [OPTFRA-241](#) - Getting issue details... [STATUS](#)

[OPTFRA-239](#) - Getting issue details... [STATUS](#) [OPTFRA-232](#) - Getting issue details... [STATUS](#)

Thursday 5/17/18

OOF - Policy Done | OOF - AAI Done | OOF - SO, testing started in SB01

Wednesday 5/16/18

Example policies and modified CSAR's have been created and attached to this page. | OOF - Policy Debug Still in Progress | OOF - AAI Waiting for multicloud fix | OOF - SO, testing started in SB01

Tuesday 5/15/18

OOF -AAI, AAI connectivity resolved | OOF - Policy, Still testing/Troubleshooting with Policy | OOF - SO, exact regression CSAR's acquired, modification started | AAI – Multicloud, A couple issues with data model used have been fixed that blocked AAI

Monday 5/14/18

OOF - Policy, Bugs Found in Final Tests, Policy Team working on fixes per Ankit | OOF - AAI Needed further data populated by Multicloud in AAI - In Progress | OOF - SO, Testing Setup but blocked by other parts | AAI - Multicloud Working

Friday 5/11/18

OOF -AAI, Bugs found, Fixed and Retested | OOF - SO, Testing Setup but blocked by other parts | OOF - Policy, Testing In Progress - Will be completed Today | AAI - Multicloud, Working

Integration:

vCPE Regression (ensure HPA changes do not break vCPE)

- [Kang is working through this](#)

vCPE + HPA

- Gated by vCPE Regression, If vCPE doesn't work in regression it won't work with HPA
- The work to test HPA over vCPE regression is only + 10%

vCPE + HPA Integration Test Plan

- Setup vCPE the same as regression; up to Service Instantiation
 - Before vCPE Service Instantiation
 - Create HPA policies – See wiki page [Policy OOF HPA](#)
- Example policies:

HPA Policy Example 1

```
{
  "riskLevel": "1",
  "riskType": "SampleRiskType",
  "policyName": "PolicyHPA3",
  "service": "hpaPolicy",
  "guard": "False",
  "description": "Must have Huge pages support",
  "templateVersion": "OpenSource.version.1",
  "priority": "1",
  "version": "CSIT",
  "content": {
    "identity": "hpaPolicy_set3",
    "policyScope": [
      "INTERNATIONAL"
    ],
    "policyType": "hpaPolicy",
    "resources": "vGMuxInfra",
    "flavorFeatures": [
      {
        "flavorLabel": "vcpe.vgmx",
        "flavorProperties": [
          {
            "hpa-feature-attributes": [
              {
                "hpa-attribute-key": "numVirtualCpu",
                "hpa-attribute-value": "4",
                "operator": "="
              },
              {
                "hpa-attribute-key": "virtualMemSize",
                "hpa-attribute-value": "8192",
                "operator": "=",
                "unit": "MB"
              }
            ],
            "mandatory": "True",
            "hpa-feature": "basicCapabilities",
            "architecture": "generic",
            "hpa-version": "v1"
          },
          {
            "hpa-feature-attributes": [
              {
                "hpa-attribute-key": "logicalCpuPinningPolicy",
                "hpa-attribute-value": "dedicated",
                "operator": "="
              }
            ],
            "mandatory": "True",
            "hpa-feature": "cpuPinning",
            "architecture": "generic",
            "hpa-version": "v1"
          }
        ]
      }
    ]
  }
}
```

HPA Policy Example 2

```
{
  "riskLevel": "1",
  "riskType": "SampleRiskType",
  "policyName": "PolicyHPA3",
  "service": "hpaPolicy",
  "guard": "False",
  "description": "Must have Huge pages support",
  "templateVersion": "OpenSource.version.1",
  "priority": "1",
  "version": "CSIT",
  "content": {
    "identity": "hpaPolicy_set3",
    "policyScope": [
      "INTERNATIONAL"
    ],
    "policyType": "hpaPolicy",
    "resources": "vGMuxInfra",
    "flavorFeatures": [
      {
        "flavorLabel": "vcpe.vgmx",
        "flavorProperties": [
          {
            "hpa-feature-attributes": [
              {
                "hpa-attribute-key": "numVirtualCpu",
                "hpa-attribute-value": "4",
                "operator": ">="
              },
              {
                "hpa-attribute-key": "virtualMemSize",
                "hpa-attribute-value": "8",
                "operator": "=",
                "unit": "MB"
              }
            ],
            "mandatory": "True",
            "hpa-feature": "basicCapabilities",
            "architecture": "generic",
            "hpa-version": "v1"
          }
        ]
      }
    ]
  }
}
```

This example will change based on flavors in OpenStack/Multicloud in the lab.

- Create flavors in OpenStack
 - flavors which is used to discover HPA information should named with prefix of "onap.", otherwise there will no HPA information can be extracted by multicloud plugins for OpenStack
 - the number of flavors to be created for ONAP is determined by number of the generic flavors multiplied by the combination of HPA specification.
 - In this example, there are 2 generic flavors:
 - medium: 4 vcpu, 4GB memory, 40GB storage
 - large: 6 vcpu, 8GB memory, 80GB storage
 - In this example, there are 2 set of HPA specification:
 - set 1: hw:numa_nodes=2, hw:cpu_policy=dedicated, hw:mem_page_size=2M;
 - set 2: hw:cpu_policy=dedicated, hw:mem_page_size=2M
 - Hence the flavors named with prefix of "onap." are:
 - onap.flavor2.medium
 - medium with HPA set 1
 - onap.flavor2.large
 - large with HPA set 1
 - onap.flavor3.medium
 - medium with HPA set 2
 - onap.flavor3.large

- large with HPA set 2

- Use modified CSARs flavorLabel attributes to match the ones created in policy - See below 'HPA Updated CSAR's' section.
- Distribute CSARs to SO
- At vCPE Service Instantiation
 - Send updated REST service instantiation request to SO that includes user param: Customer_Location
- Continue the same as vCPE Regression and confirm all is well

Updates:

Thursday 5/31/18

We were able to send a homing request to OOF and receive HPA flavor labels. SO then processes these correctly. However we need to make it through vCPE Use Case to the portion where we instantiat vGW to verify. Made it through vCPE Use Case until "PostProcess SDNC Create" where I hit bug:

2018-06-01 00:32:35,497 | ERROR | tp1609525416-571 | GenericResourceApiProvider | 374 - org.onap.sdnc.northbound.generic-resource-api-provider - 1.3.3 - - | Caught exception executing service logic for tunnelxconn-topology-operation
org.onap.ccsdk.sli.core.sli.SvcLogicException: Invalid index values [0,]

Brian Freeman told me I could solve bug by pulling the latest containers and updating a few config to match SB07. Trying to pull the latest containers has been in progress for three hours. I will retest HPA when containers finish pulling.

Tuesday 5/29/18

Started integration testing by setting up vCPE use case in SB01. Will update wiki with instructions on the additional procedures.

Wednesday 5/16/18

Example policies and modified CSAR's have been created and attached to this page. On policy side more will need to be created for all vCPE CSARs.

HPA Updated CSAR's

[csars_20180516_HPA.zip](#)

The above CSAR's are copies of the ones Kang is using for vCPE Regression Integration with one minor change.

Each CSAR has heat Environment File update to vcpe_flavor_name as follows:

CSAR Name	vcpe_flavor_name
service-VcpesvcInfra0412a-csar	vcpe.infra
service-VcpesvcRescust0412a-csar	vcpe.vgw
service-VcpesvcVbng0412a-csar	vcpe.bng
service-VcpesvcVbrg0412a-csar	vcpe.brg
service-VcpesvcVgmux0412a-csar	vcpe.vgmux

HPA Code uses the vcpe_flavor_name to match HPA flavors to the correct policy and service.

Postman Collection

```
{
  "variables": [],
  "info": {
    "name": "SO Pairwise",
    "_postman_id": "2cf34bf0-16d9-e9ad-0948-c0ded70fed44",
    "description": "",
    "schema": "https://schema.getpostman.com/json/collection/v2.0.0/collection.json"
  }
}
```

```

},
    "item": [
        {
            "name": "SO Create vCPE",
            "request": {
                "url": "http://{SO Host}:8080/ecomp/mso/infra/serviceInstances/v5",
                "method": "PUT",
                "header": [
                    {
                        "key": "Content-Type",
                        "value": "application/json",
                        "description": ""
                    },
                    {
                        "key": "Authorization",
                        "value": "Basic SW5mcmFQb3J0YWxDbGllbnQ6cGFzc3dvcmQxJA==",
                        "description": ""
                    }
                ],
                "body": {
                    "mode": "raw",
                    "raw": "{\r\n  \requestDetails\": {\r\n    \"cloudConfiguration\": {\r\n      \"lcpCloudRegionId\": \"RegionOne\", \r\n      \"tenantId\": \"1e097c6713e74fd7ac8e4295e605ee1e\"\r\n    }, \r\n    \"modelInfo\": {\r\n      \"modelInvariantId\": \"a3ebfaf2-3dea-43b0-bb8e-566f95735230\", \r\n      \"modelName\": \"vcpesvc_rescust_0412a\", \r\n      \"modelType\": \"service\", \r\n      \"modelVersion\": \"1.0\", \r\n      \"modelVersionId\": \"dc316752-561f-48a3-8354-d873c813735d\" \r\n    }, \r\n    \"owningEntity\": {\r\n      \"owningEntityId\": \"520cc603-a3c4-4ec2-9ef4-ca70facd79c0\", \r\n      \"owningEntityName\": \"OE-Demonstration\" \r\n    }, \r\n    \"project\": {\r\n      \"projectId\": \"Project-Demonstration\" \r\n    }, \r\n    \"requestInfo\": {\r\n      \"instanceName\": \"vcpe_svc_vcpesvc_rescust_0412a_201805102300\", \r\n      \"productFamilyId\": \"a9a77d5a-123e-4ca2-9eb9-0b015d2ee0fb\", \r\n      \"requestorId\": \"vCPE-Robot\", \r\n      \"source\": \"VID\", \r\n      \"suppressRollback\": \"true\" \r\n    }, \r\n    \"requestParameters\": {\r\n      \"aLaCarte\": \"false\", \r\n      \"subscriptionServiceType\": \"vCPE\", \r\n      \"userParams\": [\r\n        {\r\n          \"name\": \"BRG_WAN_MAC_Address\", \r\n          \"value\": \"fa:16:3e:c0:34:ef\" \r\n        }, \r\n        {\r\n          \"name\": \"Customer_Location\", \r\n          \"value\": {\r\n            \"customerLatitude\": \"32.897480\", \r\n            \"customerLongitude\": \"-97.040443\" \r\n          }, \r\n          \"customerName\": \"some_company\" \r\n        } \r\n      ] \r\n    }, \r\n    \"subscriberInfo\": {\r\n      \"globalSubscriberId\": \"SDN-ETHERNET-INTERNET\", \r\n      \"subscriberName\": \"Kaneohe\" \r\n    } \r\n  }, \r\n  \"description\": \"\" \r\n}\r\n\r\n\", \r\n    \"response\": [] \r\n  }\r\n}\r\n\r\n\", \r\n    {
            "name": "Policy vCPE HPAL",
            "request": {
                "url": "http://{Policy Host}/pdp/api/createPolicy",
                "method": "PUT",
                "header": [
                    {
                        "key": "Content-Type",
                        "value": "application/json",
                        "description": ""
                    },
                    {
                        "key": "Authorization",
                        "value": "Basic dGVzdHBHbkBBoYTEyMw==",
                        "description": ""
                    },
                    {
                        "key": "cache-control",
                        "value": "no-cache",
                        "description": ""
                    },
                    {
                        "key": "clientauth",
                        "value": "ch10ag9uOnRlc3Q=",
                        "description": ""
                    }
                ],
                "body": {
                    "key": "content-type",

```

```

        "value": "application/json",
        "description": ""
    },
    {
        "key": "environment",
        "value": "TEST",
        "description": ""
    }
],
"body": {
    "mode": "raw",
    "raw": "{\r\n  \\"riskLevel\":"\r\n  \\"riskType\":"\r\n
SampleRiskType\"," \\"policyName\":"\r\n  \\"service\":"\r\n  \\"guard\":"\r\n
False\"," \\"description\":"\r\n  \\"Must have Huge pages support\"," \\"templateVersion\":"\r\n  \\"OpenSource.version.
1\"," \\"priority\":"\r\n  \\"version\":"\r\n  \\"CSIT\"," \\"content\":"\r\n  \\"identity\":"\r\n
hpaPolicy_set3\"," \\"policyScope\":"\r\n  \\"INTERNATIONAL\"," \\"policyType\":"\r\n
\\"hpaPolicy\"," \\"resources\":"\r\n  \\"vGMuxInfra\"," \\"flavorFeatures\":"\r\n
{\r\n  \\"flavorLabel\":"\r\n  \\"vcpe.vgmux\"," \\"flavorProperties\":"\r\n
{\r\n  \\"hpa-feature-attributes\":"\r\n  {\r\n  \\"hpa-
attribute-key\":"\r\n  \\"numVirtualCpu\"," \\"hpa-attribute-value\":"\r\n
\r\n  \\"operator\":"\r\n  },\r\n
{\r\n  \\"hpa-attribute-key\":"\r\n  \\"virtualMemSize\"," \\"hpa-
attribute-value\":"\r\n  \\"operator\":"\r\n  \\"unit\":"\r\n
\\"MB\"," \"}\r\n  },\r\n  \\"mandatory\":"\r\n
\r\n  \\"hpa-feature\":"\r\n  \\"basicCapabilities\"," \\"architecture\":"\r\n
\r\n  \\"hpa-version\":"\r\n  },\r\n  {\r\n  \\"
hpa-feature-attributes\":"\r\n  {\r\n  \\"hpa-attribute-key\":"\r\n
logicalCpuPinningPolicy\"," \\"hpa-attribute-value\":"\r\n
\r\n  \\"operator\":"\r\n  }\r\n  },\r\n
\r\n  \\"mandatory\":"\r\n  \\"hpa-feature\":"\r\n
\r\n  \\"architecture\":"\r\n  \\"hpa-version\":"\r\n
\r\n  }\r\n  }\r\n  }\r\n  }\r\n  }\r\n}"
    },
    "description": ""
},
"response": []
},
{
    "name": "Policy vCPE HPA2",
    "request": {
        "url": "http://{Policy Host}/pdp/api/createPolicy",
        "method": "PUT",
        "header": [
            {
                "key": "Content-Type",
                "value": "application/json",
                "description": ""
            },
            {
                "key": "Authorization",
                "value": "Basic dGVzdHBkcDphbHB0YTEyMw==",
                "description": ""
            },
            {
                "key": "cache-control",
                "value": "no-cache",
                "description": ""
            },
            {
                "key": "clientauth",
                "value": "cHl0aG9uOnRlc3Q=",
                "description": ""
            },
            {
                "key": "content-type",
                "value": "application/json",
                "description": ""
            },
            {
                "key": "environment",
                "value": "TEST",

```


Attach to robot VM, execute following script:

```
$ /opt/demo.sh init_customer
```

The default cloud-region (CloudOwner,RegionOne) and the default complex (cli1) and the association between them will be populated .

Caveat: The cloud-region populated by this script is not conform to the cloud region expected by multicloud. So it is not possible to discover/populate HPA information under this default cloud region.