OSM Hackfest – Session 4.1
Modeling EPA capabilities in VNF
Eduardo Sousa (Canonical)
Guillermo Calviño (Altran)
Felipe Vicens (ATOS)
EPA (Enhanced Placement Awareness)

- **EPA features** like use of large hugepages memory, dedicated CPUs, strict NUMA node placement, the use of passthrough and SR-IOV interfaces, can be used in OSM's VNF descriptors since Rel Zero.

- If your VIM supports EPA, then you don't need to do anything extra to use it from OSM. VIM connectors in OSM take advantage of EPA capabilities if the VIM supports it. All you need to do is build your descriptors and deploy.

- Openstack configuration for EPA ([reference guide](#))
EPA support combined with **SDN Assist** enables chaining of high performance VNFs

1. Accurate assignment of resources at VM level
2. Proper assignment of I/O interfaces to the VM
3. **SDN gives the ability to create underlay L2 connections**
   - Interconnecting VMs
   - Attaching external traffic sources
Adding new VIM account: openstack-epa

• VIM:
  • openstack-epa: <vim_ip>

• Test VIM:
  • ping <vim_ip>
  • curl http://<vim_ip>:5000/v2.0

Load Openstack credentials:

  export OS_AUTH_URL=http://<vim_ip>:5000/v2.0
  export OS_USERNAME=osm
  export OS_TENANT_NAME=osm
  export OS_PASSWORD=osm

• Run some commands:
  • openstack image list
  • openstack network list
  • openstack flavor list
  • openstack server list
Adding new VIM account: openstack-epa

- Add your second VIM ‘openstack-epa’ with the OSM client:
  
  ```
  osm vim-create --name openstack-epa --account_type openstack
  --auth_url http://<vim_ip>:5000/v2.0 
  --user <username> --password <password> --tenant <tenant> 
  --description “ETSI openstack site 2, with EPA, with tenant <tenant>” 
  --config '{dataplane_physical_net: physnet_sriov, microversion: 2.32}'
  ```

  - osm vim-list
  - osm vim-show openstack-epa

- Config options:
  - dataplane_physical_net:
    - Used to instantiate VMs with SR-IOV and Passthrough interfaces
    - Value: The physical network label used in Openstack both to identify SRIOV and passthrough interfaces (nova configuration) and also to specify the VLAN ranges used by SR-IOV interfaces (neutron configuration).
  - microversion:
    - Used for device role tagging
    - Value: 2.32
VNF: hackfest_epasriov_vnfd

External Connection point: vnf-mgmt

VDU: mgmtVM
- Image name: US1604
- VM Flavor: 1 CPU, 1024MB RAM, 10 GB disk
- Interfaces:
  - mgmtVM-eth0: VIRTIO
  - mgmtVM-eth1: VIRTIO

ICP: mgmtVM-internal

External Connection point: vnf-data

VDU: dataVM
- Image name: US1604
- VM Flavor: 1 CPU, 1024MB RAM, 10 GB disk
- Interfaces:
  - dataVM-eth0: VIRTIO
  - dataVM-xe0: VIRTIO
- Guest EPA

ICP: dataVM-internal

VL: internal
User Interface

- Clone hackfest_multivdu_vnfd in the user interface

- A new hackfest_multivdu_vnfd appears:
Creating the VNFD

- Edit the new descriptor
- Modify the name and id: hackfest_epasriov_vnfd
- Modify VDU dataVM:

```yaml
guest-epa:
  cpu-pinning-policy: DEDICATED
  cpu-thread-pinning-policy: PREFER
  mempage-size: LARGE
  numa-node-policy:
    mem-policy: STRICT
    node-cnt: '1'
    node:
      - id: '1'
```

And finally, this is the sample file: Hackfest EPA SRIOV VNF Descriptor
[https://osm-download.etsi.org/ftp/osm-6.0-six/7th-hackfest/packages/hackfest_epasriov_vnf.tar.gz](https://osm-download.etsi.org/ftp/osm-6.0-six/7th-hackfest/packages/hackfest_epasriov_vnf.tar.gz)
NS diagram - Changes highlighted in yellow

NS: hackfest_epasriov_nsd

VNF: hackfest_epasriov_vnfd
CP: vnf-data
VL: datanet
CP: vnf-mgmt

VNF: hackfest_epasriov_vnfd
CP: vnf-data
VL: mgmtnet
CP: vnf-mgmt

© ETSI 2019
User Interface

- Steps:
  - Compose a new NS
  - Create new Package

- NS PACKAGES

- COMPOSE A NEW NS

- hackfest_epasriov_nsd
NSD Composer

- Steps
  - NSD Composer
  - Keyboard shortcuts
Creating the NSD (1/2)

- **Steps**
  - **Select VNFs:**
    - [VNF](#)
      - member-vnf-index: 1
      - vnf-id-ref: hackfest_spassiov_vnfd
    - [VNF](#)
      - member-vnf-index: 2
      - vnf-id-ref: hackfest_spassiov_vnfd
  - **Create VLs:**
    - mgmtnet
    - datanet

(Drag and drop)
Creating the NSD (2/2)

● Steps
  ○ Link VLs with VNFs (Shift + Left Click)
    ■ Select the name for the CPs (vnf-data and vnf-mgmt)

● Final Scenario

And finally, this is the sample file: Hackfest EPA SRIOV NS Descriptor
https://osm-download.etsi.org/ftp/osm-6.0-six/7th-hackfest/packages/hackfest_epasriov_ns.tar.gz
Deploying NS in the UI (1/4)

- Onboard VNFD and NSD to catalog using the UI
- Launch the NS from the UI
  - Depending on the VIM, specify a VIM network name to map `mgmtnet`
  - If you need to change the VIM, change the network name using `config:
    {vld: [{name: mgmtnet, vim-network-name: osm-ext}]}`
- Click the info button to see the mgmt IP address of each VNF
- Connect to management VNF:
  - `ssh osm@<IP>`
    - password: `osm4u`
• There are several methods to check if the NUMA and HugesPages was applied. We are going to show one method. This is to check the Openstack Flavors assigned to the created VM.

• List the servers: `openstack server list`

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Status</th>
<th>Networks</th>
</tr>
</thead>
<tbody>
<tr>
<td>76a03cc4-e8df-4d63-8e66-b1a6a1e40576</td>
<td>epa_test-2-dataVM-1</td>
<td>ACTIVE</td>
<td>epa_test-dataonet=192.168.255.10; epa_test-internal=192.168.101.13</td>
</tr>
<tr>
<td>5f1a46dc-12a6-4b71-83a9-417236525374</td>
<td>epa_test-2-mgmtVM-1</td>
<td>ACTIVE</td>
<td>epa_test-dataonet=192.168.255.2; epa_test-internal=192.168.150.13</td>
</tr>
<tr>
<td>e03b9177-ef94-4fad-9ca9-2ea80b37be7e</td>
<td>epa_test-1-dataVM-1</td>
<td>ACTIVE</td>
<td>epa_test-dataonet=192.168.255.2; epa_test-internal=192.168.150.6</td>
</tr>
<tr>
<td>0162b119-5589-4b75-9615-981936666186</td>
<td>epa_test-1-mgmtVM-1</td>
<td>ACTIVE</td>
<td>epa_test-dataonet=192.168.255.10; epa_test-internal=192.168.101.13</td>
</tr>
</tbody>
</table>
Deploying NS in the UI (3/4)

- Show one of the mgmt servers
  
  ```bash
  openstack server show <mgmt_server_uuid>
  ```

- Show the flavor of the server
  
  ```bash
  openstack flavor show <mgmt_server_flavor_uuid>
  ```
Deploying NS in the UI (4/4)

- Show one of the data servers
  
  ```
  openstack server show <data_server_uuid>
  ```

- Show the flavor of the server
  
  ```
  openstack flavor show <data_server_flavor_uuid>
  ```