MEC Practical Realization using OSM

Pedro Escaleira (IT Aveiro)
escaleira@av.it.pt
A Quick Recap
B2B2C Model

- **NSP**
  - Offers
  - Client of
    - **CSP**
      - To
      - **End Consumer**
        - Offers
        - Client of
          - **End Consumer**

- **End Consumer**
  - To
  - Offers
  - Client of
    - **CSP**
      - To
      - **NSP**
        - Offers
        - Client of
          - **NSP**

- **Edge Infrastructure**
  - Platform, where other Operators can request new Edge resources

- **Simple Edge platform**
  - where Consumers can deploy their MEC APPs

© ETSI
A Year of Progress
1. Now, we have a *Real* Edge Environment
Real Edge Environment Scenario
However, to achieve the requirements of an Edge Scenario, we needed to modify our previous Edge NS and VNFs/CNFs.
Last Year Edge Environment NS and VNFs/CNFs
Edge Scenario Requirement
New Edge Environment NS and VNFs/CNFs
2. We made changes to the Day 1 and 2 Scripts
Execution Environments

Juju-based EEs

- Poorly documented
- Language-dependant (Python)
- On OSM, it requires the usage of *weird* software libraries (e.g., *chams.osm*)
- Performance issues

Helm-based EEs

- Very well documented
- Not language-dependant
- On OSM, it only requires an RPC API
- It is probably faster (?)
With these differences in mind, we decided to give it a try to Helm-based EEs
Helm/Ansible EEs Setup

- Helm
  - Kubernetes Controller Installer EE (container)
  - Kubernetes Worker 1 Installer EE (container)
  - Kubernetes Worker n Installer EE (container)
- Ansible
  - Kubernetes Controller VM
  - Kubernetes Worker 1 VM
  - Kubernetes Worker n VM
- OSM Kubernetes Cluster

© ETSI
Edge NS Instantiation Time

- Helm/Ansible-based: ~6.4 min
- Juju-based EE: ~10.7 min

~65% more
3. Our MEC Applications need to communicate in more than one network
Kubernetes' Usual Scenario
What we required
Multus Configuration

Flannel Network Fabric

Pod (MEC Application)

eth0
net1

PoP (Node Host)

eth5

Factory Network

Edge Equipment

Another PoP (Node Host)
4. The PoPs of the factory with cameras had a GPU
GPU Passthrough with OSM

Old OSM feature now implemented: https://osm.etsi.org/gitlab/osm/features/issues/7095

```
vnfd:
  virtual-compute-desc:
    id: worker-compute
    extra-specs:
      spec: pci_passthrough:alias
      value: gpu:1
```

Injests extra steps as a Flavor Metadata
Thank You!

This work is supported by the European Regional Development Fund (FEDER), through the Regional Operational Programme of Lisbon (POR LISBOA 2020) and the Competitiveness and Internationalization Operational Programme (COMPETE 2020) of the Portugal 2020 framework [Project NETEDGE with Nr. 069977 (POCI-01-0247-FEDER-069977)].