Exercises - OSM

These exercises will make you feel comfortable with OSM.

This exercise is based on OSM Documentation Tutorials: https://osm.etsi.org/docs/user-guide/04-vim-setup.html#what-if-i-do-not-have-a-vim-at-hand-use-of-sandboxes.

This exercise is divided in three parts:

1. Starting OSM
2. An Emulated OSM VIM
3. Deploying a NSD

Part 1: Understanding the OSM components

Open a terminal and start OSM:

```
$ docker stack deploy -c /etc/osm/docker/docker-compose.yaml osm
```

Expect the following service to be started:

Creating service osm_mongo osm_nbi osm_kafka osm_grafana osm_prometheus-cadvisor osm_pol osm_prometheus osm_zookeeper osm_mon osm_mysql osm_light-ui osm_ro osm_keystone osm_lcm

Check the started osm service containers.

```
docker stack ps osm | grep -i running
```

Open the browser, and enter the OSM web UI at http://127.0.0.1/ with login admin and password admin.

Part 2: An Emulated OSM VIM

Vim-emu emulation platform was created to support network service developers to locally prototype and test their network services in realistic end-to-end multi-PoP scenarios. It allows the execution of real network functions, packaged as Docker containers, in emulated network topologies running locally on the developer's machine. The emulation platform also offers OpenStack-like APIs for each emulated PoP so that it can integrate with MANO solutions, like OSM. The core of the emulation platform is based on Containernet.

Known limitations of VIM Emulator:

- VIM Emulator requires special VM images, suitable for running in a VIM Emulator environment.
- Day-1 and Day-2 procedures of OSM are a work in progress in VIM Emulator, and hence are not available as of the date of this publication.

Notice, if you plan to use this emulation platform for academic publications, please cite the following paper:

- M. Peuster, H. Karl, and S. v. Rossem: MeDICINE: Rapid Prototyping of Production-Ready Network Services in Multi-PoP Environments. IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN), Palo Alto, CA, USA, pp. 148-153. doi: 10.1109/NFV-SDN.2016.7919490. (2016)

In the host machine, check if the emulator is running:

```
$ docker ps | grep vim-emu
```

If not, start it with the following command:

```
$ docker run --name vim-emu -t -d --rm --privileged --pid='host' --network=netosm -v /var/ru
```

You need to set the correct environment variables, i.e., you need to get the IP address of the vim-emu container to be able to add it as a VIM to your OSM installation:

```
$ export VIMEMU_HOSTNAME=$(sudo docker inspect -f '{{range .NetworkSettings.Networks}}{{.IPA
```

Attach OSM to vim-emu

```
$ osm vim-create --name emu-vim1 --user username --password password --auth_url http://$VIME

$ osm vim-list
```

Part 3: Deploying a NSD

Onboarding the VNFs Descriptors

```
$ osm vnfd-create vim-emu/examples/vnfs/ping.tar.gz
$ osm vnfd-create vim-emu/examples/vnfs/pong.tar.gz
```

Onboarding the NSD

```
$ osm nsd-create vim-emu/examples/services/pingpong_nsd.tar.gz
```

You can now check OSM's GUI to see the VNFs and NS in the catalog. Or:

```
$ osm vnfd-list

$ osm nsd-list
```

Check the OSM GUI interface. Click in the left side of the screen, Packages -> VNF Packages or NS Packages. There you should see the ping and pong VNFDs and the pingpong NSD.

Instantiate example pingpong service

```
$ osm ns-create --nsd_name pingpong --ns_name test --vim_account emu-vim1
```

Check service instance using OSM client

```
$ osm ns-list
```

Interact with deployed VNFs

```
$ docker exec -it mn.dc1_test-1-ubuntu-1 /bin/bash
```

Ping the pong VNF over the attached management network

```
#root@dc1_test-1-ubuntu-1:/# ping 192.168.100.4
```

```
$ osm ns-delete test
```

connect to vim-emu Docker container to see its logs (do in another terminal window)

```
$ docker logs -f vim-emu
```

check if the emulator is running in the container

```
$ docker exec vim-emu vim-emu datacenter list
```

check running service

```
$ docker exec vim-emu vim-emu compute list
```

Congratulations!

Now you know a little about OSM!

For more info and tutorials, check https://osm.etsi.org/wikipub/index.php/Latest_OSM_Hackfest_Material