

Open Source
MANO

OSM Hackfest – Session 1

Installation and first use

Benjamín Díaz (Whitestack)
Guillermo Calviño (Altran)

Installing OSM

- Official REL5 installation procedure:
 - https://osm.etsi.org/wikipub/index.php/OSM_Release_FIVE#Install_OSM_Release_FIVE
- Using Vagrant:
 - https://osm.etsi.org/wikipub/index.php/How_to_run_OSM_on_Vagrant
 - Instructions when using a downloaded image:
 - Create a folder and copy the image there
 - Open the windows console and move to the folder created:
 - `cd Documents\Vagrant`
 - Initialize the environment:
 - `vagrant init osm/releasefive --box-version 0`
 - Copy the image to the same folder (if not done yet) and add it as a 'vagrant box':
 - `vagrant box add osm/releasefive virtualbox.box`
 - Modify the Vagrantfile your exposed ports preferences:
 - `config.vm.network "forwarded_port", guest: 80, host: 8080`
 - `vagrant up`
 - `vagrant ssh`

Troubleshooting Vagrant: Users with old windows

- In case you need see the following error after trying to power on the virtual machine:

```
C:\Users\guillermo.calvino\Documents\Vagrant>vagrant up
The version of powershell currently installed on this host is less than
the required minimum version. Please upgrade the installed version of
powershell to the minimum required version and run the command again.

Installed version: 2

Minimum required version: 3
```

- You need to upgrade the powershell version:
 - <http://www.microsoft.com/en-us/download/details.aspx?id=34595>
 - For example:
 - Windows 7 Service Pack 1 - 64-bit versions: Windows6.1-KB2506143-x64.msu

Upgrade OSM version

- To get the latest version of OSM we will follow these instructions:
 - Stop the OSM stack:
 - `docker stack rm osm`
 - Check if all containers have been removed, and remove the remaining ones in “Exited” status:
 - `docker ps -a`
 - `docker container prune`
 - Remove old images:
 - `docker image prune -a`
 - Start the stack:
 - `docker stack deploy -c /etc/osm/docker/docker-compose.yaml osm`

After installing OSM

- Test OSM client
 - Try 'osm'
- Test UI:
 - Access UI:
<http://localhost:8080>
Credentials are admin/admin

Local OSM instances

- osm-plugtest1: <http://172.21.1.4>
- osm-plugtest2: <http://172.21.1.5>
- osm-hackfest1: <http://172.21.1.9>
- osm-hackfest2: <http://172.21.1.10>

For people using ETSI OSM servers

Install OSM client



- Instructions could be found here:
 - https://osm.etsi.org/wikipub/index.php/How_to_install_OSM_client

Steps:

- `curl https://osm-download.etsi.org/repository/osm/debian/ReleaseFIVE/OSM%20ETSI%20Release%20Key.gpg | sudo apt-key add -`
 - `sudo add-apt-repository -y "deb [arch=amd64] https://osm-download.etsi.org/repository/osm/debian/ReleaseFIVE stable osmclient"`
 - `sudo apt-get update`
 - `sudo apt-get install -y python-osmclient`
-
- After the installation completes, you might want to add the following environment variables to your `.bashrc` file:
 - `export OSM_HOSTNAME=<OSM_host> # IP of the OSM server (default: 127.0.0.1)`

Adding VIM accounts

- VIMs:

Name	Type	AUTH URL	tenant	user	Password	SDN controller
openstack1	openstack	http://172.21.2.20:5000/v2.0	xxx	xxx	xxx	YES
openstack2	openstack	https://172.21.6.140:5000/v3	xxx	xxx	xxx	YES
• openstack3	openstack	https://172.21.6.26	xxx	xxx	xxx	NO

- ping <IP>
- curl -k http://<IP>:5000/v2.0 (or v3)
- Install python-openstackclient
 - sudo apt-get install python-openstackclient

Adding VIM accounts

- Load Openstack credentials and run some commands for testing:
 - export OS_AUTH_URL=xxx
 - export OS_USERNAME=xxx
 - export OS_TENANT_NAME=xxx
 - export OS_PASSWORD=xxx
 - export OS_IDENTITY_API_VERSION=3
 - openstack image list
 - openstack network list
 - openstack flavor list
 - openstack server list

OpenStack1 VIM Details



URL:

- <http://172.21.2.20:5000/v2.0>

Credentials

- User: etsi
- Tenant: etsi
- Password: etsiosm

Images:

- ubuntu1604
- US1604
- hackfest3-mgmt
- cirros034

Networks:

- Public & Management: mgmt

Adding OpenStack1 using “osm” cli:

```
osm vim-create --name openstack1 --account_type openstack --auth_url http://172.21.2.20:5000/v2.0 \  
--user etsi --password etsiosm --tenant etsi --description "Hackfest OpenStack1"
```

```
osm vim-list
```

OpenStack2 VIM Details



URL:

- <http://172.21.5.4:5000/v3>

Credentials

- Tenant: osm
- Tenant user: osm
- Tenant password: osm@W1nd

Images:

- ubuntu1604
- US1604
- hackfest3-mgmt
- cirros034

Networks:

- Provider/Management: external

Adding Openstack2 VIM using “osm” cli:

```
osm vim-create --name openstack2 --account_type openstack --auth_url http://172.21.5.4:5000/v3
--user osm --password osm@W1nd --tenant osm --description "ETSI openstack2 Windriver
Openstack, with tenant osm"
```

OpenStack3 VIM Details



URL:

- <http://172.21.7.5:5000/v3>

Credentials

- Tenant: osmX
- Tenant user: osmX
- Tenant password: osmX

Where X goes from 1 to 40

Images:

- ubuntu1604
- US1604
- hackfest3-mgmt
- cirros034

Networks:

- Provider/Management: PUBLIC

Adding Openstack3 VIM using “osm” cli:

```
osm vim-create --name openstack3 --account_type openstack --auth_url  
http://172.21.7.5:5000/v3 --user osm1 --password osm1 --tenant osm1 --description "ETSI  
openstack3 WhiteCloud Openstack, with tenant osm"
```

Adding images to the VIM

- Image management is not implemented in OSM today. It has to be done independently on each VIM.

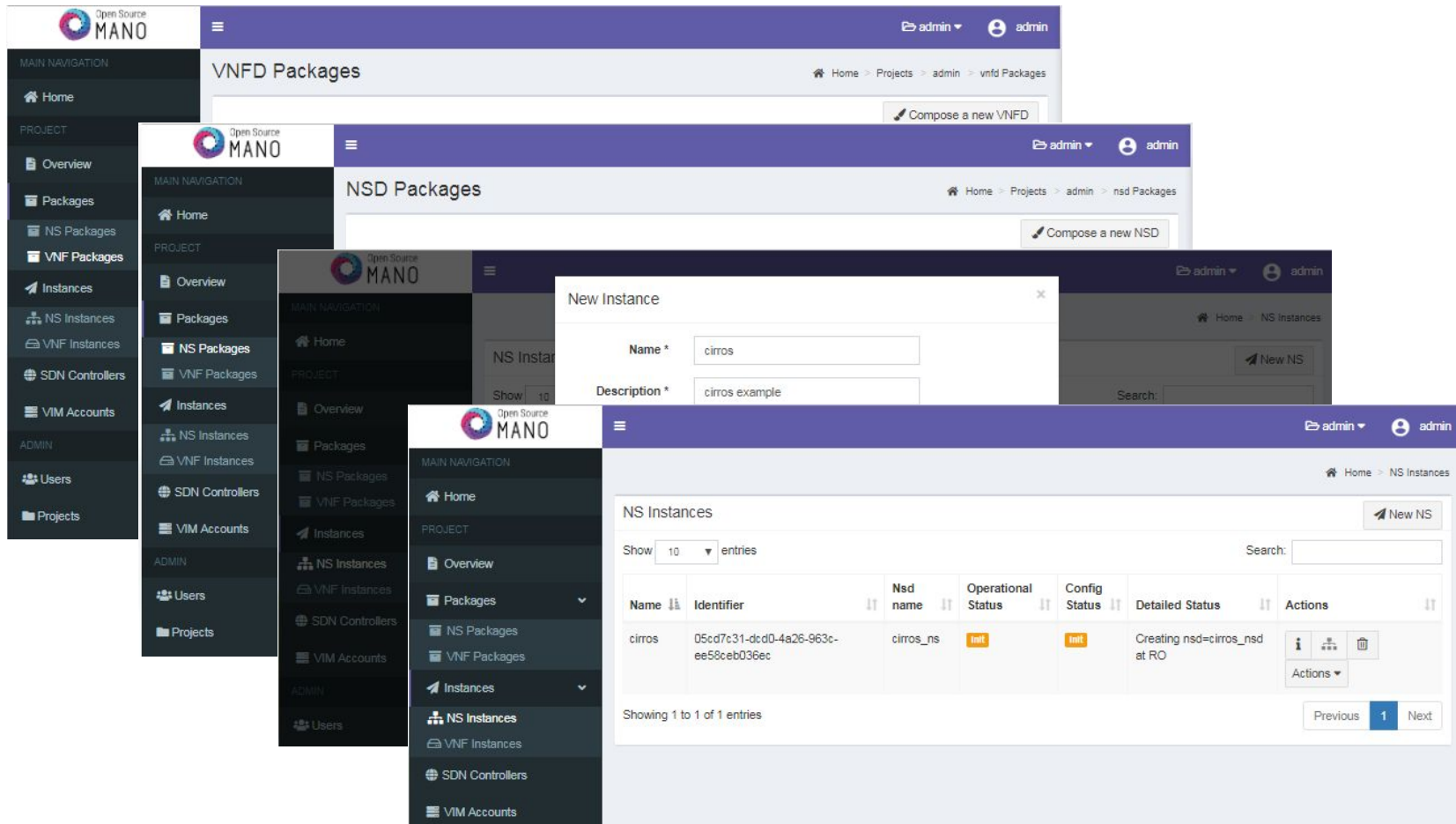
- **IMAGES HAVE BEEN ALREADY ADDED TO THE REMOTE VIMS IN THE HACKFEST**

- Example for Openstack:
 - `openstack image create --file="./cirros-0.3.4-x86_64-disk.img" --container-format=bare --disk-format=qcow2 --public cirros034`

Adding images to the VIM

Image name in descriptors	Filename
ubuntu1604	xenial-server-cloudimg-amd64-disk1.img (you can get it from https://cloud-images.ubuntu.com/xenial/current/)
US1604	US1604.qcow2
hackfest3-mgmt	hackfest3-mgmt-qcow2
hackfest-pktgen	hackfest-pktgen-qcow2
cirroso34	cirros-0.3.4-x86_64-disk.img

Deploying our first NS with OSM UI



The image shows a sequence of four overlapping screenshots from the Open Source MANO web interface, illustrating the process of creating a new Network Service (NS) Instance.

- Top Screenshot:** The 'VNFD Packages' page. The breadcrumb trail is 'Home > Projects > admin > vnfd Packages'. A button 'Compose a new VNFD' is visible.
- Second Screenshot:** The 'NSD Packages' page. The breadcrumb trail is 'Home > Projects > admin > nsd Packages'. A button 'Compose a new NSD' is visible.
- Third Screenshot:** A 'New Instance' modal dialog box. The 'Name' field contains 'cirros' and the 'Description' field contains 'cirros example'.
- Bottom Screenshot:** The 'NS Instances' page. The breadcrumb trail is 'Home > NS Instances'. A table lists the instance 'cirros' with the following details:

Name	Identifier	Nsd name	Operational Status	Config Status	Detailed Status	Actions
cirros	05cd7c31-dcd0-4a26-963c-ee58ceb036ec	cirros_ns	init	init	Creating nsd=cirros_nsd at RO	[Info] [Add] [Delete] [Actions]

The page also shows 'Showing 1 to 1 of 1 entries' and navigation buttons for 'Previous', '1', and 'Next'.

Deploying our first NS with the UI

- Add VNF package (drag&drop)
- Add NS package (drag&drop)
- Instantiate
- Get VNF record and obtain mgmt IP address
- Access to the VNF via SSH (user: “cirros”, pass: “cubswin:”) or “gocubsgo” depending on the VIM)
- Delete NS instance
- Delete NS
- Delete VNF

Deploying our first NS with OSM client

- **Add VNF and NS packages**

- `osm vnfd-create cirros_vnf.tar.gz`
- `osm vnfd-list`
- `osm nsd-create cirros_2vnf_ns.tar.gz`
- `osm nsd-list`

- **Instantiate**

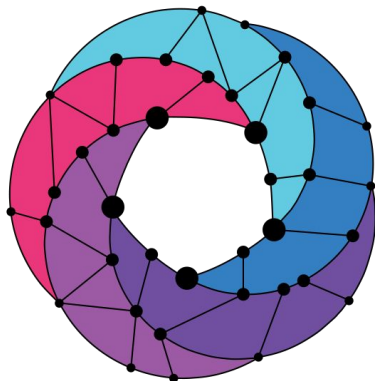
- `osm ns-create --nsd_name cirros_2vnf_ns --ns_name <ns-instance-name> --vim_account <data-center-name>`
- `osm ns-list`

- **Delete NS instance**

- `osm ns-delete <ns-instance-name>`
- `osm ns-list`

- **Delete VNF and NS package**

- `osm nsd-delete cirros_2vnf_ns`
- `osm nsd-list`
- `osm vnfd-delete cirros_vnfd`
- `osm vnfd-list`



Open Source
MANO

Find us at:

osm.etsi.org
osm.etsi.org/wikipub