Department of Computer Science
Computer Networks Research Group

Technology Review

Management of ServiCes Across MultipLE clouds

Authors:

Arkajit Dhar
Ashwin Prasad Shivarpatna Venkatesh
Bhargavi Mohan
Deeksha Mysore Ramesh
Harshitha Pandithanahalli Somashekaraiah
Sanket Kumar Gupta
Suheel Shrirangapura Nazeersab
Vivek Jaganath

Supervisors:

Prof. Dr. Holger Karl | Sevil Dräxler | Hadi Razzaghi Kouchaksaraei

Paderborn, December 29, 2018

Contents

List of Figures

List of Tables

1

Open Source MANO

2

OpenBaton

3

SONATA/PISHAHANG

3.1 Configuration requirements to run Pishahang on a single server or VM

- Operating System: Ubuntu 16.04 as base image (http://releases.ubuntu.com/16.04/)
- Minimum Requirements: 4GB RAM, 40GB hard disk and a non-root user account

3.2 OpenStack Installation

We set up an OpenStack environment using DevStack, which is installed via a configuration file named local.conf. The installation guide can also be found at https://docs.openstack.org/devstack/latest/

- Other references [1] [2]

3.2.1 Steps of installation:

- Create a user "stack"

```
sudo useradd -s /bin/bash -d /opt/stack -m stack
echo "stack ALL=(ALL) NOPASSWD: ALL" | sudo tee \ /etc/sudoers.d/stack
sudo su - stack
```

- Clone the devstack repository

```
git clone \ https://git.openstack.org/openstack-dev/devstack
cd devstack
```

[1] Refer DevStack heat documentation to enable heat service
[2] Refer DevStack networking-sfc documentation for service chaining

- Create and configure the local.conf file

```
[[local|localrc]]
ADMIN\_PASSWORD=password
DATABASE\_PASSWORD=$ADMIN_PASSWORD
RABBIT_PASSWORD=$ADMIN\_PASSWORD
SERVICE\_PASSWORD=$ADMIN_PASSWORD
```

- Execute the command

```
./stack.sh
```

- After installation check and verify from openstack horizon GUI

  Access http://1.2.3.4, replace 1.2.3.4 with the IP address of your host Login using user id: admin, password: admin

3.3 Pishahang installation

The Below steps of installation are performed from the non-root user account

- Installing packages

```
sudo apt-get install -y software-properties-common
sudo apt-add-repository -y ppa:ansible/ansible
sudo apt-get update
sudo apt-get install -y git ansible
```

- Clone repository

```
git clone https://github.com/CN-UPB/Pishahang.git
cd Pishahang/son-install
echo sonata | tee ~/.ssh/.vault_pass
```

- Start Installation Replace "<your_ip4_address>" with the IP address SONATA should be available at , also check the command

```
ansible-playbook utils/deploy/sp.yml -e \ "target=localhost
    public_ip=<your_ip4_address>" -v
```

- Verify Installation Open your browser and navigate to http://public_ip. Login using the username sonata and password 1234. If the installation was successful, you should now see the dashboard of the service platform

- Installation of son-cli The SONATA CLI toolset can also be installed via the Python setup script

```
git clone https://github.com/sonata-nfv/son-cli.git
cd son-cli
python3 setup.py install
```

- Test if its working by invoking

```
son-workspace -h
son-package -h
son-publish -h
son-push -h
son-monitor -h
```

Reference Link - https://github.com/sonata-nfv/son-cli#all-dists-using-setuptools

3.4 Service Descriptor Packaging and uploading

We also need the son-cli to be installed and son-examples repository to be cloned in the environment

- Add WIM
  - Open your browser and navigate to http://public_ip
  - Open the "WIM/VIM Settings" tab

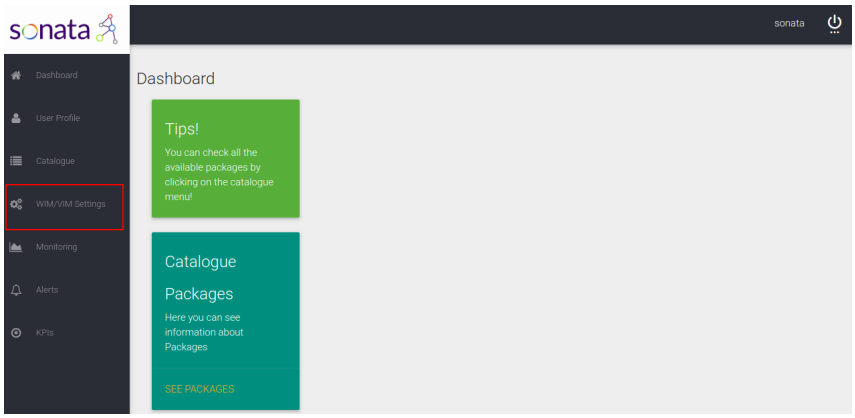

Figure 3.1: Sonata Dashboard

  - click on add a WIM
  - Select "Mock" WIM vendor
  - Enter any WIM name(e.g. Sonata Test), WIM address(e.g. local host), username(e.g., Sonata) and password(e.g. 1234)
  - Confirm by clicking "SAVE"

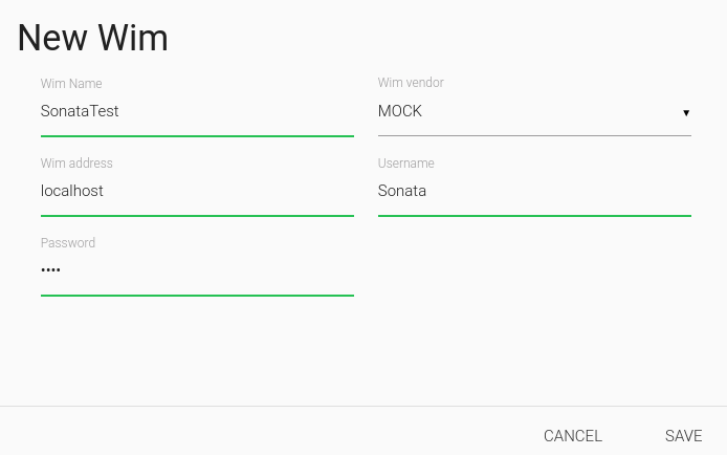

Figure 3.2: Add WIM

- Adding OpenStack VIM
  - Click on add a VIM
  - Enter the VIM name(e.g. DevStack) , select the WIM just created, enter the country(e.g. germany) and city(Paderborn)
  - Select "Heat" VIM vendor

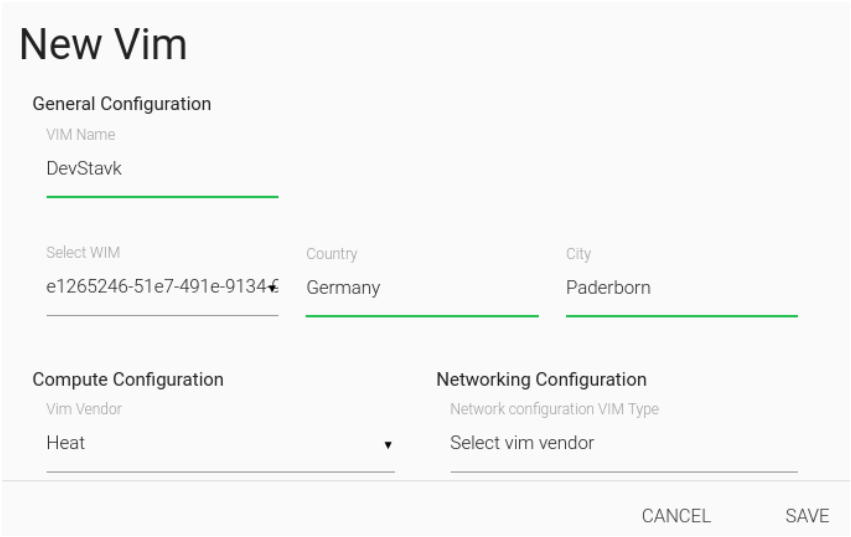

Figure 3.3: Add VIM

  - Tenant ID: DevStack project id (e.g. sonatademo),Tenant External Netwrok ID: DevStack ID of the public network and Tenant External Router ID: DevStack ID of the router created under sonatademo user i.e. sonata-router as shown below

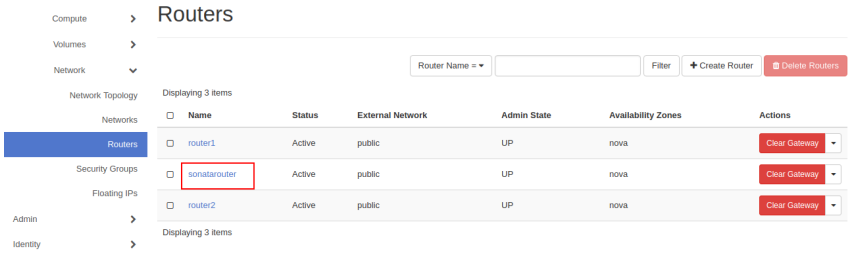

Figure 3.4: Select Router

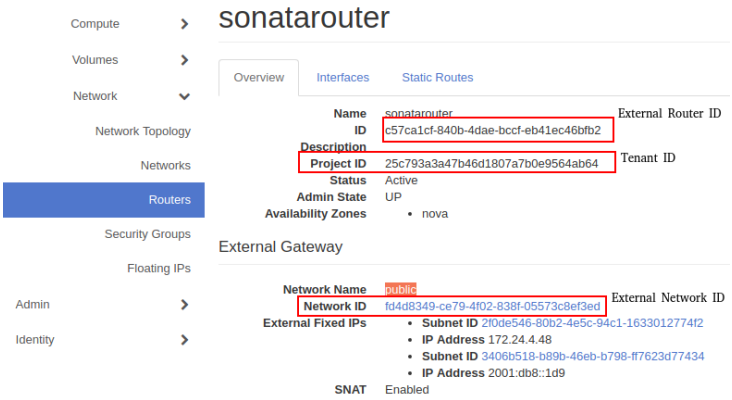

Figure 3.5: Select IDs

- VIM Address: DevStack (131.234.29.34)
- Vim Vendor: “OVS”,Username: sonatademo,Password: password of the user sonatademo (e.g. sonata),Domain: Default
- Click on "Save"

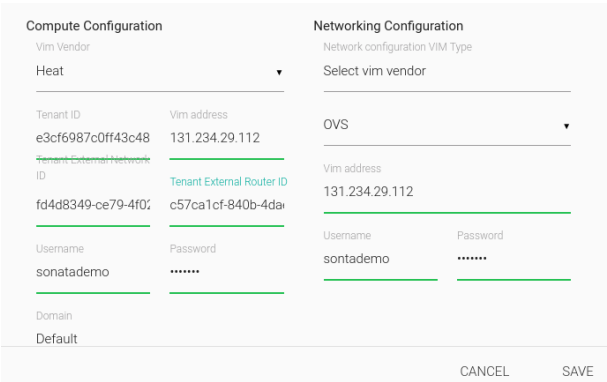

Figure 3.6: VIM Details

– On-boarding Service Package

```
git clone \ https://github.com/sonata-nfv/son-examples.git

son-workspace --init

son-validate --project \ son-examples/service-projects/sonata-demo

son-package --project \ son-examples/service-projects/sonata-demo
    -n \ service_package

son-access config --platform_id ServicePlatform \ --new --url
    http://131.234.29.102 --default

son-access auth -u sonata -p 1234

son-access push --upload service_package.son
```

Reference video - https://www.youtube.com/watch?v=RsXUIt4rzF0

3.5 Linking VIM to sonata

Login to the DevStack dashboard: http://131.234.29.34/dashboard.There are two users created during installation admin and demo. Password for both users is sonata

- Create New User and Project
  - Login as admin user in domain Default and create new user (e.g. sonatademo)
  - In the menu, go to Identity->User (Create User)
  - Give the admin role to the new user

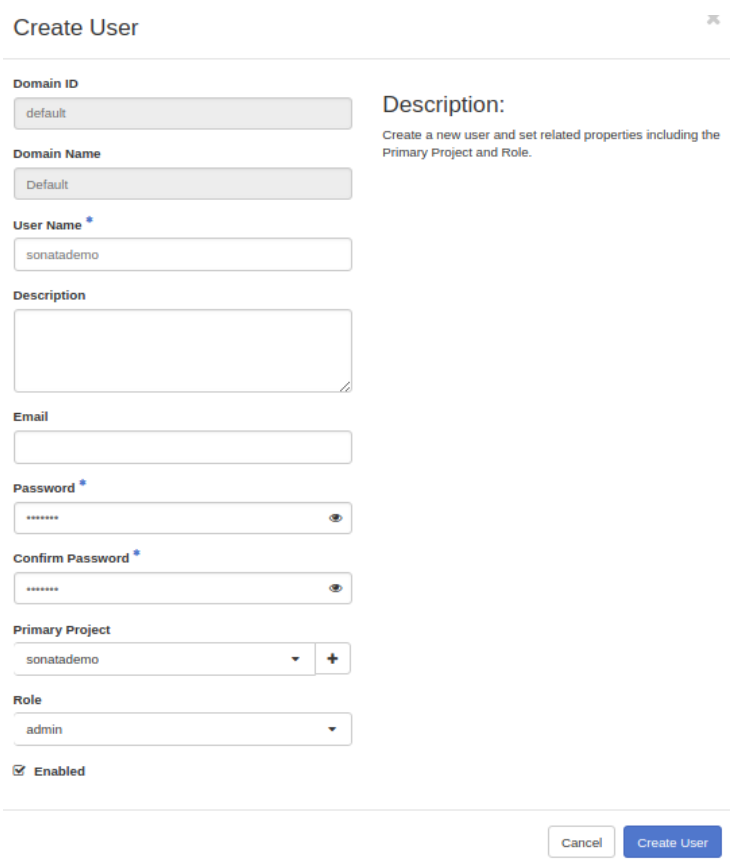

Figure 3.7: Create user

- Add a new project with the below details
  - Project name/tenant name: sonatademo
  - Allocate maximum number of resources for that project under Quotas tab

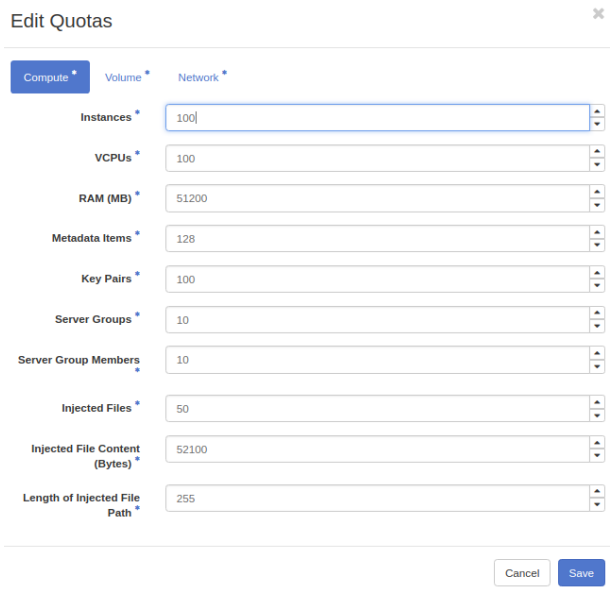

Figure 3.8: Edit project quotas

- Create Private Network
  - Login as new user(e.g. sonatademo)
  - Create a network(e.g. sonata-priv) and add the subnet as well (e.g. sonata-priv-sub)
  - Add the router
  - Use any private network address, for example 192.168.x.0/24. While creating the router select the External Network as public (Error: Reference source not found). Add the sonata-priv-sub as the interface to the router

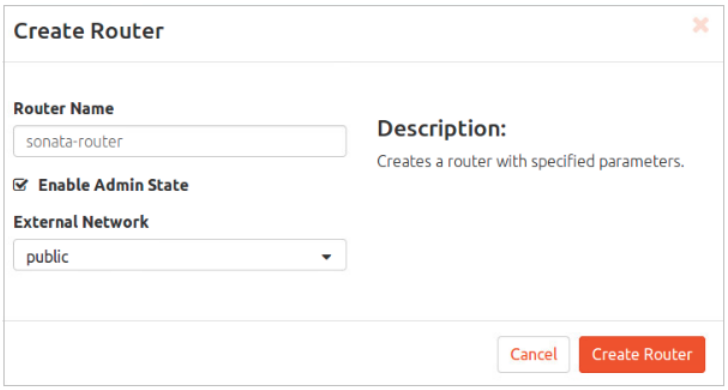

Figure 3.9: Create router

3.6 Network Service Instantiation

- Open your browser and navigate to http://public_ip:25001
- Open the "Available Complex Services" tab
- Click the "Instantiate" button of the service you want to deploy
- Confirm the instantiate modal (ingress and egress can be empty)