



Network Application Testing Platform using Openstack and OpenDaylight

Ritesh Singh Parihar¹, Sarwan Kumar Rai², Janardan Singh Parihar³, Yogita T. Hambir⁴
^{1,2,3,4}Department of Computer Engineering, Army Institute of Technology

Abstract —The networking environment today is increasingly complex. The convergence of data center, business expansion and new service deployments need to combine several different technologies - including physical and virtual devices - into a single network infrastructure. These networks must handle massive amounts of traffic with critical applications on both issues to the point and multicast network strategies. You need high-capacity network devices to route and switch huge amounts of traffic. Network equipment manufacturers are positive that their equipment can not only handle the needs of a complex network infrastructure, but also easily expand - often at massive scales. suppliers and network service companies must be able to highlight their infrastructure to see if they can handle the traffic volumes and provide the best quality service.

With our project, we will make an assessment and gather the requirements to provide our developers, multi-node OpenStack test environments on demand where they can, in a whiteboard like fashion, configure different network functions, edit them in all different types of distributions. The main objectives are the ease and speed of the test lab configuration, safety, repeatability and cost.

Our benchmarks conduct functional and performance testing and probing the weaknesses and vulnerabilities. As such, we have to push the changes to a variety of environments with multiple physical and virtual components.

Keywords- Opendaylight; Openstack; RESTful API; Virtual Machine; Devstack

I. INTRODUCTION

A centralized platform for testing and analyzing the performance of network application is going to developed. The platform will provide the administrator with features to test:

1. Multiple OS support
2. Direction and rate of traffic flow
3. Resources required and usage
4. Security
5. Cost
6. Integration for already present and future technologies

It is done by using Opendaylight Restful and neutron APIs which is the local API support provided for Opendaylight and Openstack Integration.

II. RELATED WORK

OpenDaylight is an open platform for network programmability to enable Software-Defined Networking (SDN) deployments on heterogeneous modern multi-vendor networks. OpenDaylight provides a platform abstraction model oriented service that allows users to easily write applications that work across a variety of hardware and South terminal protocols.

OpenStack is a platform for free and open-source software cloud computing, mostly deployed as an infrastructure -as-a-service (IaaS). The software platform consists of interrelated elements that control pools of processing equipment, storage, and network resources across a data center. Users manage either through a dashboard based on the Web, through command line tools, or through a RESTful API.

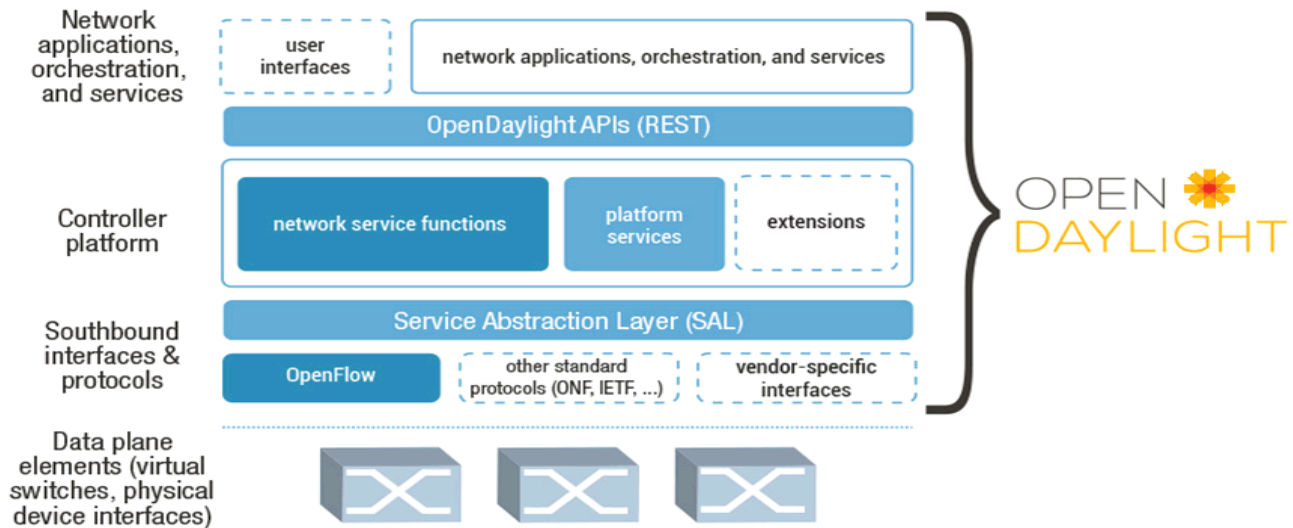


Figure 1: OpenDaylight Architecture Overview

OpenStack allows users to deploy virtual machines and others that handle different tasks for managing a cloud environment on the fly. It makes easy horizontal level, which means that the tasks have to run simultaneously can easily serve more or less users on the fly by simply turning up several instances.

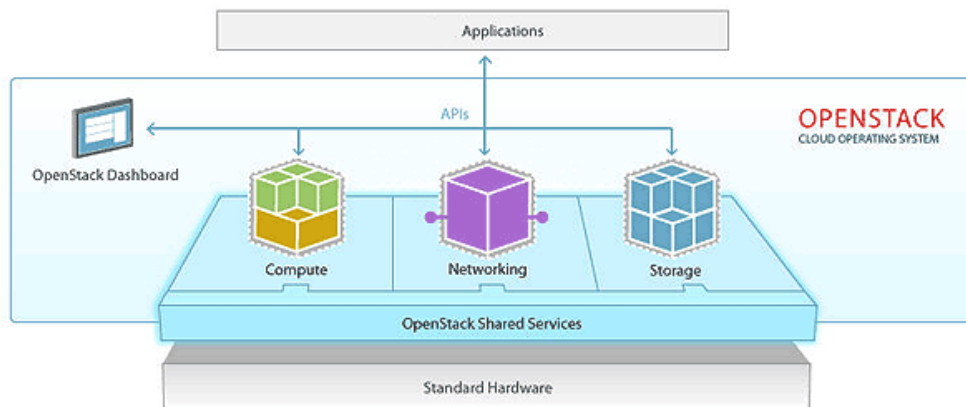


Figure 2: OpenStack Architecture Overview

OpenFlow is a communication protocol that allows access to the forwarding plane of a network switch or router on the network. It allows network controllers to determine the path of network packets through a network of switches. The controllers are separate switches.

OpenFlow allows switches from different vendors often each with their own proprietary interfaces and scripting languages to manage remotely using a single, open protocol. The inventors consider OpenFlow protocol software an enabler of Software defined networking (SDN).

III. OPENSTACK INTEGRATION WITH OPENDAYLIGHT

OpenDaylight is an open platform for network programmability to enable software-defined networking (SDN). OpenDaylight contains a driver for Neutron ML2 (Modular Layer 2) Plugin to allow communication between Neutron and OpenDaylight. On the side of the SDN controller, a OpenDaylight has northbound APIs to interact with Neutron and

useOVSDB (Open vSwitch Database Management Protocol) for southbound configuration of vSwitches on compute nodes. Thus OpenDaylight can handle network connectivity and launch GRE or VXLAN tunnels for the compute nodes.

OpenDaylight exposes one common OpenStack service to the north. - Exposed API Neutron corresponds precisely - Multiple implementations of neutron networks OpenDaylight.

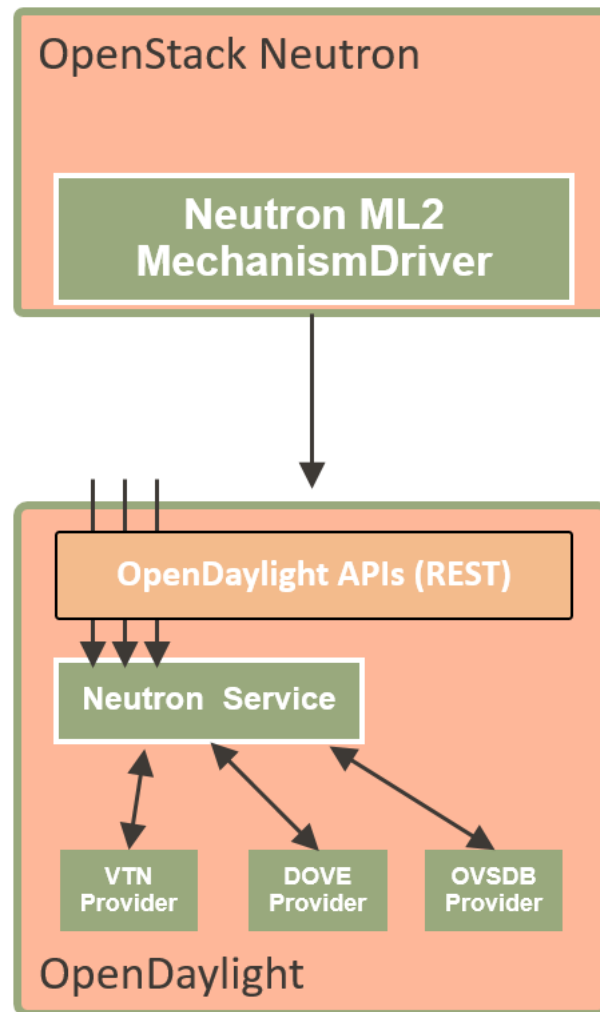


Figure 3: OpenStack and OpenDaylight Integration

IV. PROPOSED SOLUTION

Our proposed solution can be explained by the following steps:

1. Launch OpenDaylight Controller

Get OpenDaylight virtualization edition from OpenDaylight Release Download. OpenDaylight can either run on same host of OpenStack control node or on separate Host. To run the controller

```
$ ./run.sh -virt ovsdb -of13
```

2. Launch DevStack control and compute nodes.

In this step, to have a minimal multi-node OpenStack deployment we can launch one or more compute node(s) along with control node.

The local.conf file in DevStack contains the list of all the services to be enabled or disabled. Also the IP settings of the SERVICE HOST and odl controller is specified in the file. To launch Devstack nodes:

3. Spawn VM instances on DevStack node

Export and launch of VM instances can be done either by using CLI interface or simply by using Devstack Horizon dashboard which is web based GUI. In case of CLI, first step is to export VM images that are required. For example

```
$ export IMAGE=cirros-0.3.0-i386-disk.img
```

Example to boot up VM instances

```
$ nova boot --flavor <flavor> --image <image id>--nic net-id=<network id> test1
```

Therefore, in essence, following is the overall process of OpenStack and OpenDaylight. On the front of OpenStack, Neutron consists of ML2 driver mechanism, which acts as a proxy REST and passes all API calls into Neutron OpenDaylight. OpenDaylight contains a service to northern REST (called API Service Neutron) that caches data from these proxies API calls and makes it available to other services within OpenDaylight. Shown below, when describing the two components in detail, these RESTful API make the connection of OpenStack and OpenDaylight.

REFERENCES

- [1] [1] Bruno Astuto A. Nunes, A Survey of SoftwareDened Networking: Past, Present, and Future of Programmable Networks," IEEE Xplore
- [2] [2] Raj Jain, Subharthi Paul, Network virtualization and software dened networking for cloud computing: a survey" IEEE Xplore
- [3] [3] DevStack Documentation, <http://docs.openstack.org/developer/devstack/>
- [4] [4] OpenDaylight Wiki, <https://wiki.opendaylight.org/>
- [5] [5] Brent Salisbury, "OpenDaylight OpenStack Integration with DevStack on Fedora" <http://networkstatic.net/opendaylight-openstack-integration-devstack-fedora-20/>
- [6] [6] Sridhar Rao, "OpenDaylight is One of the Best Controllers for OpenStack Heres How to Implement It" <http://thenewstack.io/opendaylight-is-one-of-the-best-controllers-for-openstack-heres-how-to-implement-it/>