How do I learn to stop worrying about OpenStack and OpenDaylight, and just bring them together?
What are we going to do?
Agenda

• High level overview of OpenStack and Neutron
• A bit deeper look into Neutron ML2
• Higher level overview of OpenDaylight
• OpenStack and OpenDaylight integration
• Hands on workshop
• Exercises/Presentations
OpenStack for beginners
OpenStack: The Open Source Cloud Platform

- **Compute (Nova)**
  Self-service provisioning of virtual machines through a software API

- **Network Service (Neutron)**
  For tenant created, virtual isolated networks and subnets, and services

- **Object Storage (Swift)**
  Massively scalable, distributed object store

Your Application

Diagram showing OpenStack components:
- OpenStack Dashboard
- Compute
- Networking
- Storage
- APIs
- OpenStack Shared Services
- Standard Hardware
OpenStack Architecture
What is Neutron?

- Neutron is an OpenStack project to provide “Networking as a Service” between interface devices managed by other OpenStack services.
- Basic API Abstraction (port, subnet, network)
- Operator selects backends to implement the core API (ML2, OpenvSwitch, Linux Bridge etc.)
- Extendable API to provide advanced services (LBaaS, FWaaS etc.)
What is this Neutron ML2?
Now a core plugin since Havana, Icehouse and beyond!

Deprecates the OVS, LinuxBridge, and Hyper-V plugins!
ML2 Use Cases

- Replaces existing monolithic plugins, eases development of new plugins
  - Eliminates redundant code
  - Reduce development and maintenance effort
- New features
  - Top-of-Rack switch control
  - Avoid tunnel flooding via L2 population
  - Modular Agents
- Heterogeneous deployments
  - Specialized hypervisor nodes with distinct network mechanisms
  - Integrate *aaS appliances
  - Roll new technologies into existing deployments
## ML2 Architecture Diagram

<table>
<thead>
<tr>
<th>Neutron Server</th>
<th>ML2 Plugin</th>
<th>API Extensions</th>
<th>Type Manager</th>
<th>Mechanism Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRE Type Driver</td>
<td>VLAN Type Driver</td>
<td>VXLAN Type Driver</td>
<td>Type Manager</td>
<td>Mechanism Manager</td>
</tr>
<tr>
<td>Arista</td>
<td>Nexus</td>
<td>Cisco</td>
<td>Hyper-V</td>
<td>L2 Pop</td>
</tr>
<tr>
<td>Linuxbridge</td>
<td>Open vSwitch</td>
<td>Tail-F NCS</td>
<td>??</td>
<td></td>
</tr>
</tbody>
</table>
Let there be OpenDaylight
What is OpenDaylight
Architecture overview
When OpenStack met OpenDaylight
OpenStack Integration status

• ODL ML2 driver available in Icehouse release
  • Supported VXLAN and GRE
  • Devstack support
  • Focused on core Neutron Functionality, still used DHCP, L3 Agents

• ODL ML2 in Kilo (current)
  • Changed auth to basic auth
  • Vendor decomposition – networking-odl in stackforge
  • Support for L3 Service, FWaaS, LBaaS
OpenStack and ODL

Neutron Node

- Neutron Server
  - ML2 Plugin w/ OpenDaylight Driver

OpenDaylight Node

- OpenDaylight Server
  - Neutron API Service
  - OVSDB Plugin

Compute Node

- VM1
- VM2

Network Node

- L3 Agent
- DHCP Agent

OVS

- REST API
- RPC
- OpenFlow & OVSDB

VM1
VM2
L3 Agent
DHCP Agent
OVS
OVS
ML2 Plugin w/ OpenDaylight Driver
Neutron API Service
OVSDB Plugin
REST API
RPC
OpenFlow & OVSDB

OpenDaylight FORUM INDIA 2015
OpenDaylight NorthBound API Layer - REST APIs

OpenDaylight Neutron REST-API

OVSDDB Neutron Application

API Driven SAL (ADSAL)
- Configuration Service
- Inventory Service
- Connection Service
- Flow Programmer

Model Driven SAL (MDSAL)
- Inventory Service
- Connection Service
- Flow Programmer

OVSDDB South-bound Plugin
- OVSDDB Protocol Library
- Bidirectional JSON-RPC Library
  - Netty.io

OpenFlow 1.0 SB Plugin
- OpenFlow 1.0 Plugin
- OpenFlow 1.0 Library
  - java.nio.socket

OpenFlow 1.3 SB Plugin
- OpenFlow 1.3 Plugin
- OpenFlow 1.3 Library
  - Netty.io

OpenFlow 1.0

OpenFlow 1.3

OVSDDB Protocol

OpenVSwitch

OpenDaylight NorthBound API Layer - REST APIs

OpenDaylight Neutron REST-API

OVSDDB Neutron Application

API Driven SAL (ADSAL)
- Configuration Service
- Inventory Service
- Connection Service
- Flow Programmer

Model Driven SAL (MDSAL)
- Inventory Service
- Connection Service
- Flow Programmer

OVSDDB South-bound Plugin
- OVSDDB Protocol Library
- Bidirectional JSON-RPC Library
  - Netty.io

OpenFlow 1.0 SB Plugin
- OpenFlow 1.0 Plugin
- OpenFlow 1.0 Library
  - java.nio.socket

OpenFlow 1.3 SB Plugin
- OpenFlow 1.3 Plugin
- OpenFlow 1.3 Library
  - Netty.io

OpenFlow 1.0

OpenFlow 1.3

OVSDDB Protocol

OpenVSwitch
Let’s play!
What you need - OpenStack

• Ubuntu 14.0.x VM with a NAT and a Host-Only Adaptor – 4GB RAM recommended, minimum 2GB
• Clone devstack repository
• Internet access from your VM
• Good to have: Try out devstack from https://wiki.opendaylight.org/view/OVSD B:OVSDB_OpenStack_Guide
What you need - ODL

• Running on your laptop or VM (4GB RAM minimum for VM)
• Setup dev environment: Java, Maven
• Clone Integration repo and compile, or get Helium SR3 binary
• Eclipse – Optional
• Good to have: Run ODL and `feature:install odl-ovsdb-openstack` to make sure all artifacts are available in local repository
Workshop Walkthrough

• Run ODL on your laptop/VM
• OpenStack VMs provided as OVAs, boot and login as stack/stack.
• ‘cd devstack’ and edit local.conf so that IP Addresses match your setup
• Run devstack: ‘~/devstack/stack.sh’
Workflow - Installation

Stack.sh

CLI/Horizon

Set-manager

Open vSwitch

Configure bridge

OVSDDB Connect

OpenFlow Connect

ODL

Neutron

Nova

libvirt
Workflow – Create Network/Subnet

CLI/Horizon

Create Network

Create Subnet

ODL

Neutron

Nova

Open vSwitch

libvirt
Log in to Horizon (admin/admin)
Workflow – Create VM

CLI/Horizon → Nova → libvirt
Create VM

Create Neutron Port
Neutron

Create Port
ODL
OpenFlow Port Status Notification
OpenFlow Flows

Create TAP Port
Open vSwitch
Horizon Dashboard

Overview

Usage Summary

Select a period of time to query its usage:


The date should be in YYYY-mm-dd format.

Active Instances: 0 Active RAM: 0 bytes This Period's VCPU-Hours: 0 This Period's GB-Hours: 0

Usage

<table>
<thead>
<tr>
<th>Project Name</th>
<th>VCPUs</th>
<th>Disk</th>
<th>RAM</th>
<th>VCPU Hours</th>
<th>Disk GB Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

No items to display.
Launch VM
Launch VM
Launch VM (contd.)
Instance is booted

<table>
<thead>
<tr>
<th>Instance Name</th>
<th>Image Name</th>
<th>IP Address</th>
<th>Size</th>
<th>Key Pair</th>
<th>Status</th>
<th>Availability Zone</th>
<th>Task</th>
<th>Power State</th>
<th>Uptime</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>demo</td>
<td>cirros-0.3.1-x86_64-uec</td>
<td>10.0.0.2</td>
<td>m1.nano 64MB RAM 1 VCPU 0Bytes Disk</td>
<td>-</td>
<td>Active</td>
<td>nova</td>
<td>None</td>
<td>Running</td>
<td>0 minutes</td>
<td>More</td>
</tr>
</tbody>
</table>
Boot another instance
Playtime’s over
Links

• https://wiki.opendaylight.org/view/OpenStack_and_OpenDaylight
• https://github.com/vthapar/odl-openstack/
• https://github.com/romilgupta/Openstack-ODL-Script
• https://ask.opendaylight.org/questions/
• IRC: #opendaylight, #opendaylight-ovsdb, #sdn-bangalore
• Meetup: http://www.meetup.com/Bangalore-SDN-and-NFV-meetup/
Questions?
Thank you…

Kyle Mestery, Brent Salisbury, Madhu Venogopal for their informative blogs and videos

Sam Hague and Flavio Fernandes for their prompt responses in ovsdb-dev

Everyone else in ovsdb-dev and #.opendaylight-ovsdb for being a great community