Running an OPNFV Academy Lab on AWS and Google - Brahmaputra Edition

Setup and operate your own OPNFV laboratory for dev, test, training using Ravello Systems on AWS and Google Cloud.

What is OPNFV

The Genesis project of OPNFV defines a number of core technologies that are part of the open source NFV platform. These include:

- Openstack with various core projects such as Horizon, Nova, Neutron, Glance, Cinder, etc...
- Open vSwitch
- Integration with an approved network controller

The Brahmaputra release of OPNFV from February 2016 includes support for 4 different “bare-metal” installers for OpenStack integrated with 4 different network controller options.

- APEX
- Compass4NFV
- FUEL
- JOID

These installers deploy OpenStack per the Genesis guidelines on hardware you provide in a network test lab or “in the cloud” with various approved network controller options such as:

- OpenContrail from Juniper
- OpenDaylight
- ONOS from ON.Labs
- No SDN controller using default OpenStack Neutron options

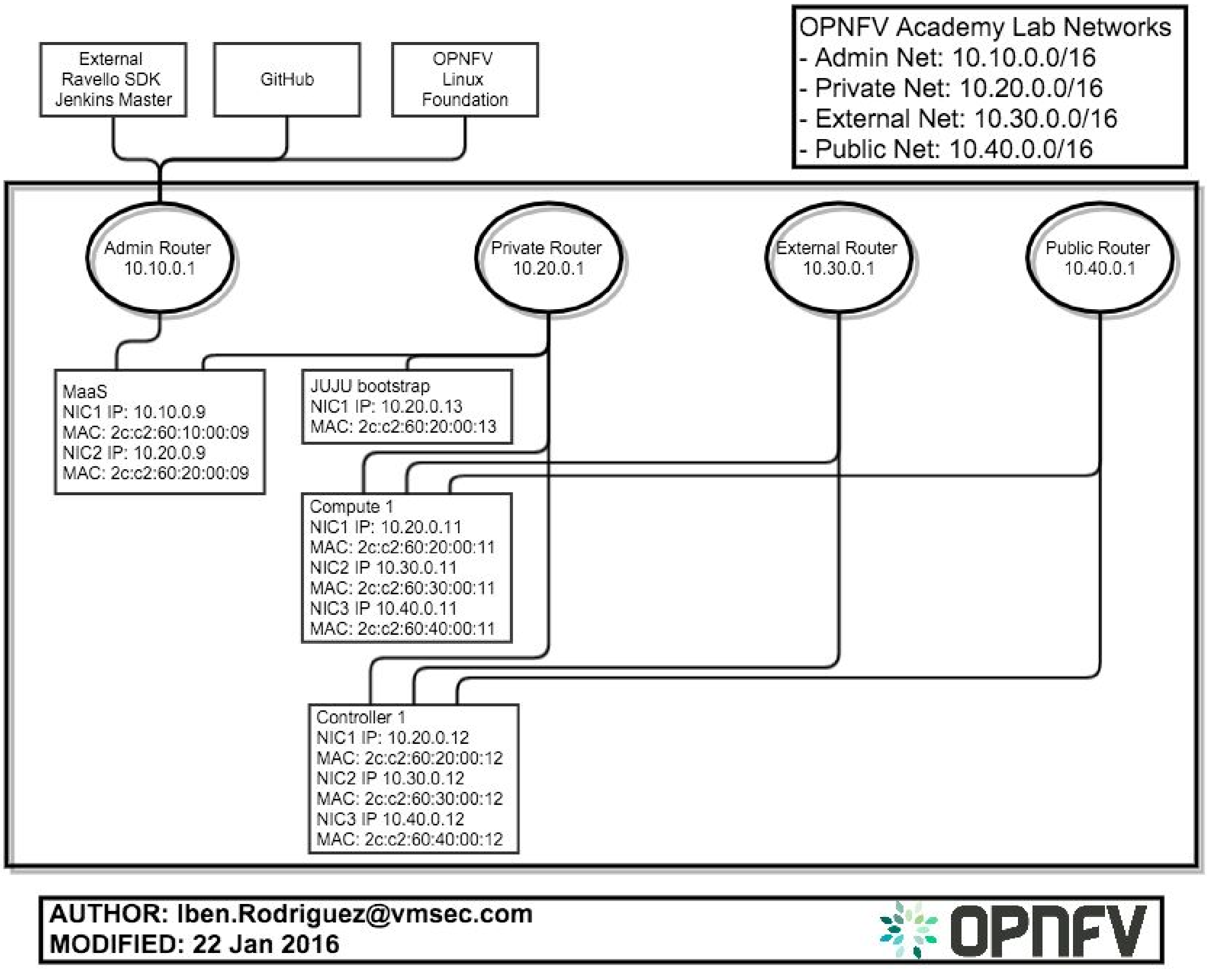
There are a number of feature and test projects that use these environments as a platform after it’s built.

If you’re new to OPNFV and DevOps there can be a pretty steep learning curve to get up to speed with all the components needed to get a working platform going and maintain it.

Organizations wishing to participate in the development and testing of OPNFV should follow the guidelines established by the Pharos project which specify 6 physical servers connected to a Jenkins build server that uses scripts to issue commands to a Jump Box machine. This jump box then installs an Operating System onto the target nodes and then configures the OpenStack, network controller, storage, and compute functions. Preparing the environment and running the scripts can take a few hours even for an automated script and that doesn’t include all the time spent planning and debugging a new installation.

These blueprints we provide are based on the the OPNFV Copper Academy work done by Bryan Sullivan from AT&T which provides a design for a lighter weight 4 node design that can run virtual either on premise or hosted “in the cloud”.

Here’s a screenshot of the network layout for the blueprints covered by this blog:



This series of posts and blueprints is intended to make it easier (and cheaper) to get started. All you will need to get started is a web browser and a free account on Ravello. The following 5 blueprint configurations are being made available and will be kept up to date on a regular basis as new OPNFV releases are available:

1. Builder blueprint with a MaaS controller pre-installed and three empty nodes for bootstrap, controller, and compute. All ready for configuration and OPNFV build out with your SDN choice. It will take a few hours to deploy this blueprint, configure maas, deploy, and produce a working OPNFV customize exactly to your parameters. This is NOT a self-contained blueprint and you must provide ssh keys, github, and complicated file editing. Intended for the more advanced developer.

2. Contrail blueprint including an already deployed OPNFV with OpenStack Kilo and Juniper Contrail SDN Controller. Spin up an app from this blueprint and in 20 minutes you will have a working OpenStack environment. Beginner level.
3. ODL blueprint including an already deployed OPNFV with OpenStack Liberty and Open Daylight SDN controller.
4. ONOS blueprint including an already deployed OPNFV with OpenStack Liberty and ON Labs: ONOS SDN controller.
5. NOSDN blueprint including an already deployed OPNFV with OpenStack Tip/Master and no SDN controller.

Find these blueprints by going to the Ravello Repo and searching for OPNFV.

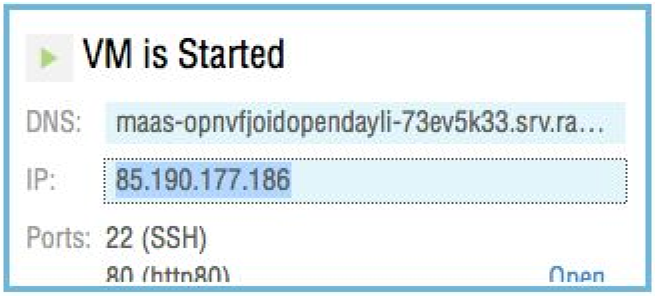
What is Ravello

Insert Blurb about Ravello Here

Getting Started with the OPNFV Academy Blueprints

Here are the general steps needed to get started with these blueprints and get up and running quickly with one of the pre-built configurations we have provided. See the readme from the github repo for more detailed steps.

1. Open a new web browser window and Add this blueprint to your library
2. Create an application from the blueprint
3. Start the application - wait 10 to 15 minutes for the machines to spin up and be ready
4. Once the VM is started you can find the IP address for the MAAS server in the Ravello dashboard.



VM is Started

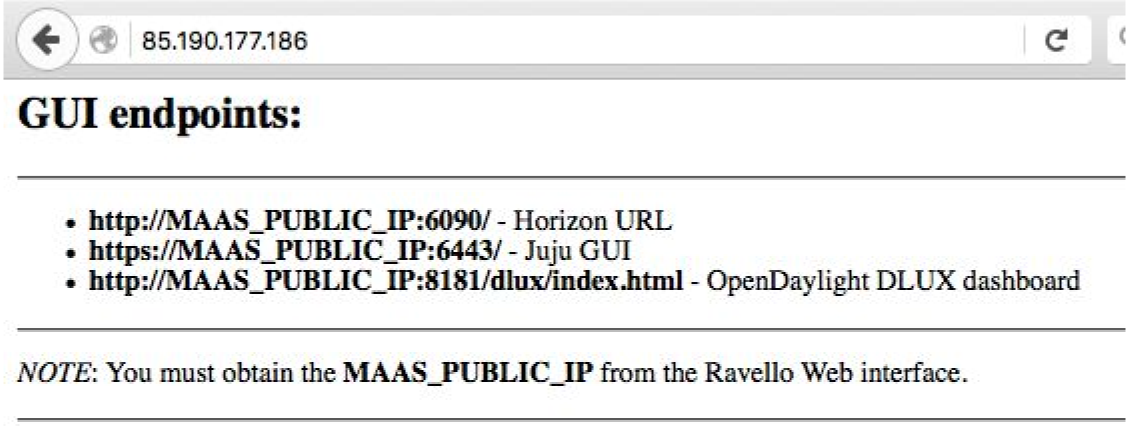
DNS: maas-opnvfjoidopendayli-73ev5k33.srv.ra…

IP: 85.190.177.186

Ports: 22 (SSH)

80 (http80) Open

5. Perform a Basic Functional Test to ensure the admin console for each function is working.
    a. open a new web browser window to the IP address of the MAAS server.



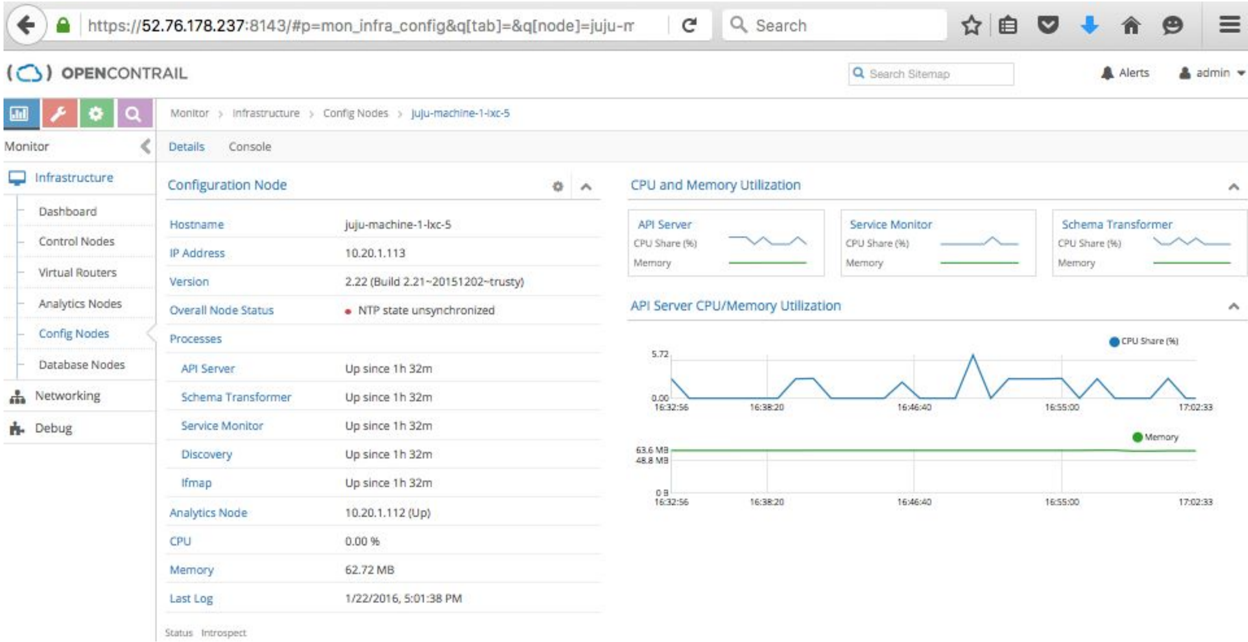
GUI endpoints:

- **http://MAAS_PUBLIC_IP:6090/** - Horizon URL
- **https://MAAS_PUBLIC_IP:6443/** - Juju GUI
- **http://MAAS_PUBLIC_IP:8181/dlux/index.html** - OpenDaylight DLUX dashboard

NOTE: You must obtain the **MAAS_PUBLIC_IP** from the Ravello Web interface.

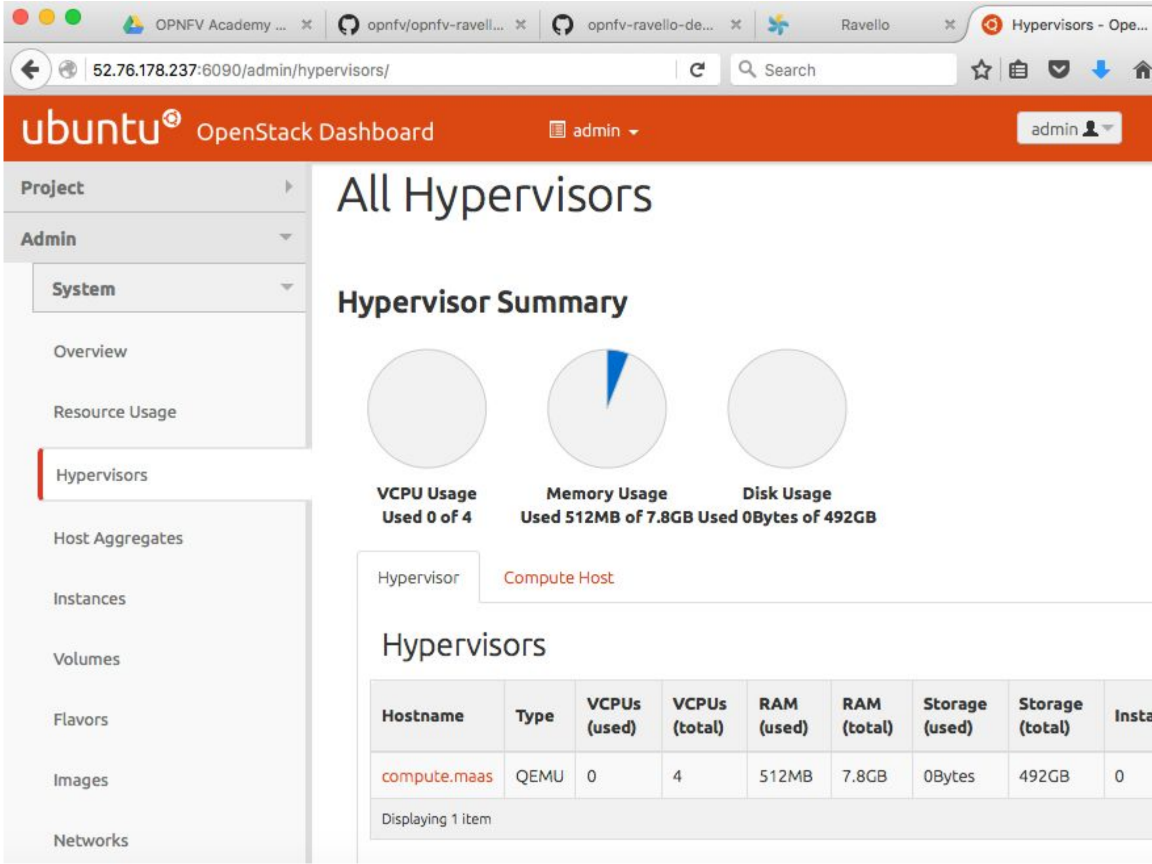
b. open new web browser windows with the MAAS server IP address followed by the port for the function you wish to use (insert screenshot)

i. Juniper OpenContrail



Hostname	juju-machine-1-lxc-5
IP Address	10.20.1.113
Version	2.22 (Build 2.21~20151202~trusty)
Overall Node Status	● NTP state unsynchronized
Processes	
API Server	Up since 1h 32m
Schema Transformer	Up since 1h 32m
Service Monitor	Up since 1h 32m
Discovery	Up since 1h 32m
Ifmap	Up since 1h 32m
Analytics Node	10.20.1.112 (Up)
CPU	0.00 %
Memory	62.72 MB
Last Log	1/22/2016, 5:01:38 PM

ii. OpenStack Horizon



All Hypervisors

Hypervisor Summary

VCPU Usage
Used 0 of 4

Memory Usage
Used 512MB of 7.8GB

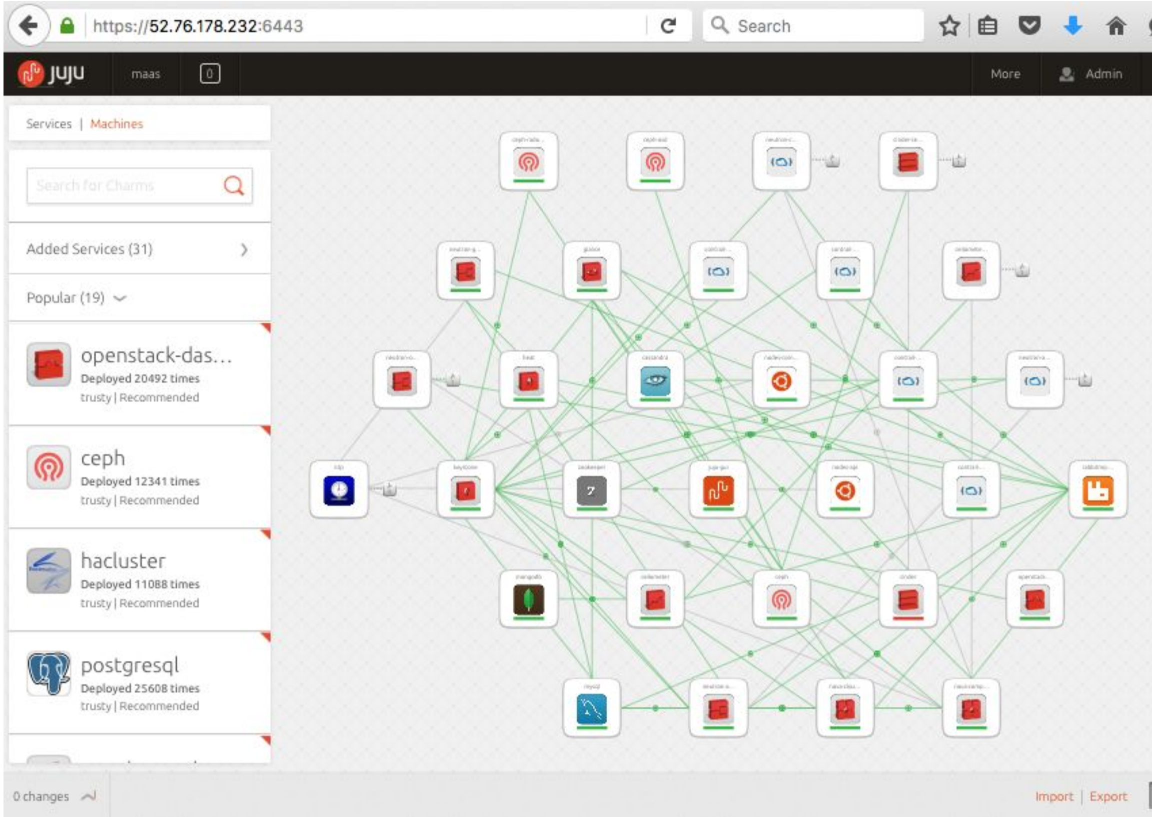
Disk Usage
Used 0Bytes of 492GB

Hypervisors

Hostname	Type	VCPUs (used)	VCPUs (total)	RAM (used)	RAM (total)	Storage (used)	Storage (total)	Insta
compute.maas	QEMU	0	4	512MB	7.8GB	0Bytes	492GB	0

Displaying 1 item

iii. JuJu admin console



https://52.76.178.232:6443

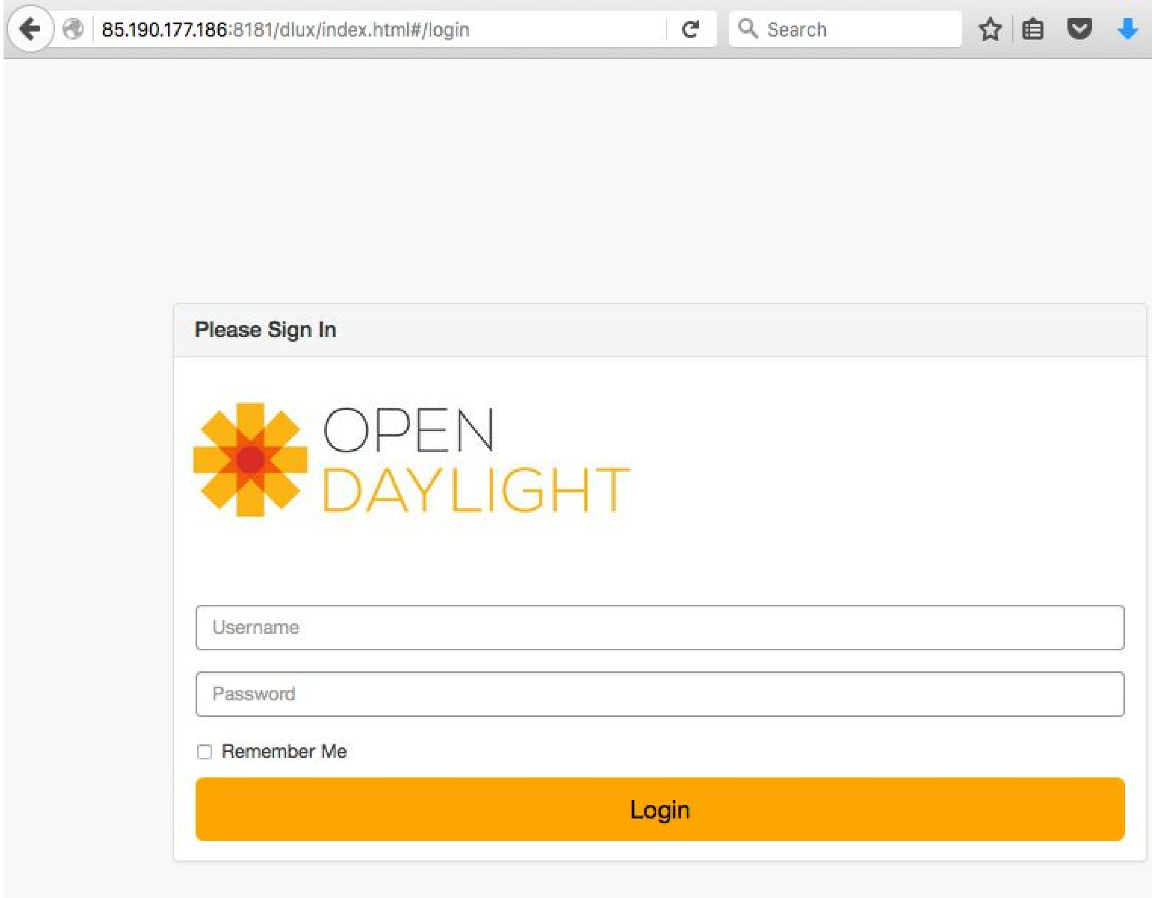
openstack-das…
Deployed 20492 times
trusty | Recommended

ceph
Deployed 12341 times
trusty | Recommended

hacluster
Deployed 11088 times
trusty | Recommended

postgresql
Deployed 25608 times
trusty | Recommended

iv. Open Daylight DLUX console



85.190.177.186:8181/dlux/index.html#/login

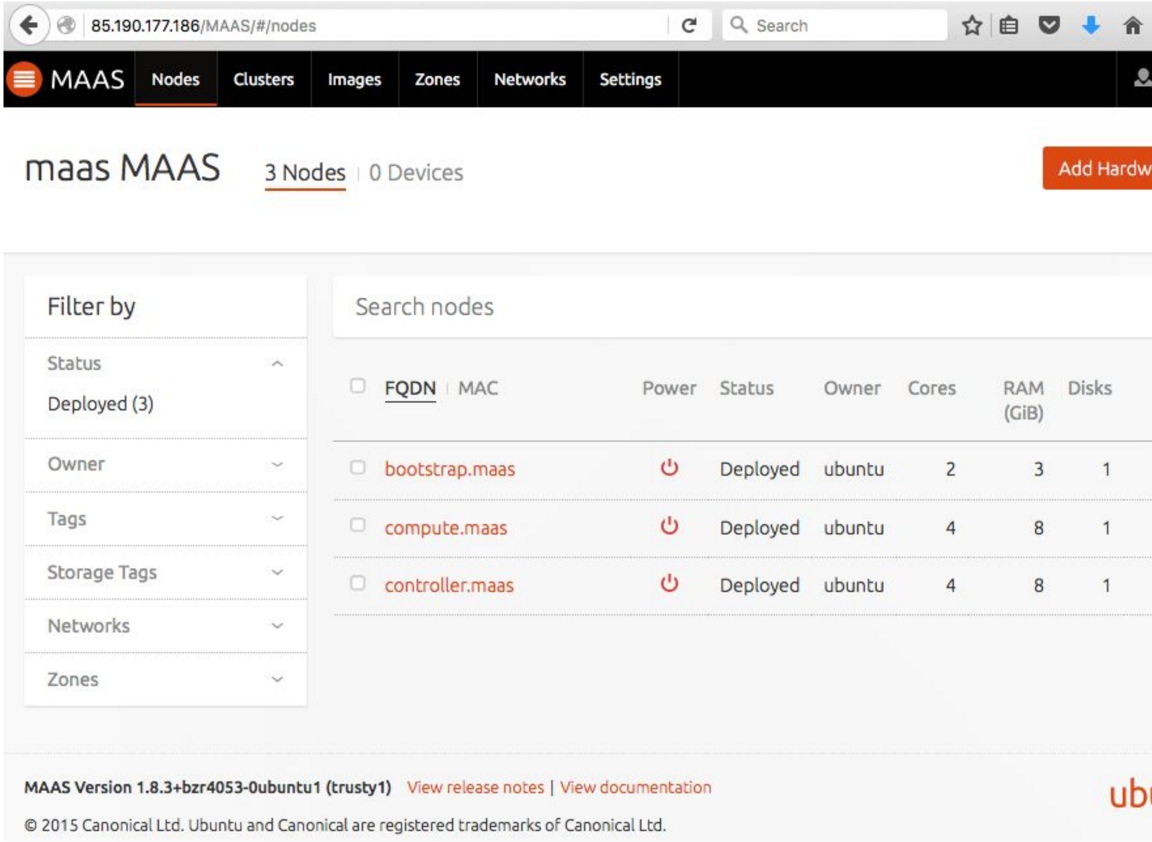
Please Sign In

Username

Password

□ Remember Me

v. MaaS admin console



☐	FQDN | MAC	Power	Status	Owner	Cores	RAM (GiB)	Disks
☐	bootstrap.maas	⏻	Deployed	ubuntu	2	3	1
☐	compute.maas	⏻	Deployed	ubuntu	4	8	1
☐	controller.maas	⏻	Deployed	ubuntu	4	8	1

MAAS Version 1.8.3+bzr4053-0ubuntu1 (trusty1)

© 2015 Canonical Ltd. Ubuntu and Canonical are registered trademarks of Canonical Ltd.

Next Steps

After this the possibilities are endless. Sign up for an account with the Linux Foundation that will give you access to update the wiki, post patches to Gerrit, update JIRA issues, and use Jenkins. Modify and create your own blueprints on Ravello to share them with others. A REST API and Python SDK are also available allowing automation of Ravello workloads as part of the product lifecycle for your company.

End of Blog