Experimental Cloud Using Commodity Hardware

A Project Report Submitted
in Partial Fulfillment of the Requirements
for the Degree of

Bachelor of Technology

by

Kaushal Kishore
(111601008)

under the guidance of

Dr. Sandeep Chandran

INDIAN INSTITUTE
OF TECHNOLOGY
PALAKKAD

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

*This is to certify that the work contained in this thesis entitled "**Experimental Cloud Using Commodity Hardware**" is a bonafide work of **Kaushal Kishore** (**Roll No. 111601008**), carried out in the Department of Computer Science and Engineering, Indian Institute of Technology Palakkad under my supervision and that it has not been submitted elsewhere for a degree.*

Dr. Sandeep Chandran

Assistant Professor

Department of Computer Science & Engineering

Indian Institute of Technology Palakkad

Contents

Chapter 1

Introduction

1.1 Cloud

Cloud computing is the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet.

Cloud services refer to any IT services that are provisioned and accessed from a cloud computing provider. This is a broad term that incorporates all delivery and service models of cloud computing and related solutions. Cloud services are delivered over the internet and accessible globally from the internet. There are three basic types of cloud services:

- Software as a Service (SaaS)
- Platform as a service (PaaS)
- Infrastructure as a service (IaaS)

1.2 Cloud Services

1.2.1 SaaS

SaaS is a software distribution model in which applications are hosted by a vendor or service provider and made available to customers over a network, typically the internet. Examples include G Suite – formerly Google Apps, Microsoft Office 365, Salesforce and Workday.

1.2.2 PaaS

PaaS refers to the delivery of operating systems and associated services over the internet without downloads or installation. The approach lets customers create and deploy applications without having to invest in the underlying infrastructure. Examples include Amazon Web Services' Elastic Beanstalk, Microsoft Azure – which refers to its PaaS offering as Cloud Services – and Salesforce's App Cloud.

1.2.3 IaaS

IaaS involves outsourcing the equipment used to support operations, including storage, hardware, servers and networking components, all of which are made accessible over a network. Examples include Amazon Web Services, IBM Bluemix and Microsoft Azure.

1.3 Experimental Cloud using Commodity Hardware

The objective of this project is to create an experimental cloud by repurposing commodity hardware. The cloud we create would be made available to students as virtual desktops which may be used to host web services which can vary from simple static page to complex web applications.

1.4 Organization of The Report

This chapter provides an overview of cloud computing and cloud services. In the next chapter we will introduce MaaS(Metal as a Service), which is a relatively new approach for cloud based service. In chapter 3, we will discuss some of the tools that we need to be familiar with to break the ice. In chapter 4 and 5, we will discuss the approach by which we can create a MaaS based cloud environment. And finally in chapter 6, we conclude with some future works.

Chapter 2

MaaS : Metal as a Service

IaaS customers are given access to servers which can be dedicated or, more often, virtual and free to install the OS and applications of their choice. The customer doesn't host or manage the underlying infrastructure but is able to use the resources as they wish.

As with all 'as a Service' computing models, customers benefit from access to the resources they need without having to invest in expensive hardware upfront, instead they pay monthly and only for what they use.

2.1 Bare metal cloud

Bare metal cloud is an environment in which physical, dedicated servers can be provisioned to customers with cloud-like ease and speed. Bare metal cloud customers are given access to the entire processing power of individual servers, as well as any storage, networking or other services they require.

Within a bare metal infrastructure there is **no multi-tenanting** (sharing of machines) and the servers provisioned are not virtual ones created on top of any hypervisor.

Customers of bare metal cloud are free to use their dedicated servers in any way they want, including running any OS and applications as well as installing hypervisors to create their own virtual machines if they want. And bare metal cloud is provided as a service.

2.2 IaaS vs. MaaS

Is there any difference between IaaS and Maas?

This depends on your view point. Many define IaaS as the provision of virtual resources only. Some include dedicated servers in their definition. In our view, bare metal cloud is the true IaaS whereas virtualised versions are really a form of Platform as a Service (PaaS).

In all scenarios you gain access to a server on which you can install and run you chosen OS and applications. In this sense, IaaS and bare metal cloud are the same.

On a virtual IaaS however, you have no knowledge of or control over the actual infrastructure on which your services are built. The provider has control of these and your services are abstracted from them.

With bare metal cloud on the other hand, you are provisioned full dedicated servers, with no virtualisation or sharing. It's up you how you use these and, in the case of installing a hypervisor, how many virtual machines you run on each.

With bare metal you get control of the full stack, from the tin right up to the user interface, and can optimise utilisation and performance to a granular level, something you simple cannot do in a virtualised environment.

2.3 Canonical's MAAS

https://maas.io/

Metal-as-a-Service (MASS) is a provisioning construct created by Canonical, developers of the Ubuntu Linux-based operating system. MAAS is designed to help facilitate and automate the deployment and dynamic provisioning of hyperscale computing environments such as big data workloads and cloud services.

Chapter 3

MAAS in VENV - I : Requirements

In this chapter we will discuss some of the tools that we will be using in the next chapter to create a MAAS based virtual cloud environment.

3.1 Software Requirements

- libvirt
- QEMU
- OpenSSH
- Ubuntu 18.04 Server ISO

3.2 Hardware Requirements

It is recommended to have atleast 16 GB RAM and 100 GB free space in the host system for deploying virtual cloud environment.

Chapter 4

MAAS in VENV - II : Controller

In this chapter we will discuss the steps required to create a basic MAAS based virtual cloud environment. Remember that the problem statement is to repurpose the commodity hardware to create an experimental cloud. At present we don't have access to those hardwares hence we are conducting our experiments in a virtualized environment.

4.1 Creating an Isolated Network

MAAS uses DHCP and DNS with PXE boot to enlist the nodes, so naturally you could have a conflict if you deploy this on a network with an existing DHCP server. Hence create a new (virtual) network with the following given steps from the QEMU interface:

1. Network name - maasisotest
2. Disable DHCP configuration
3. Choose IPv4 address which is not in use by any other connected network. I prefer to use 10.17.17.0/24.
4. Select forwarding to physical network option and set destination to any physical device and mode to NAT

The network configuration for 'maasisotest' should look similiar to Fig. 4.1.

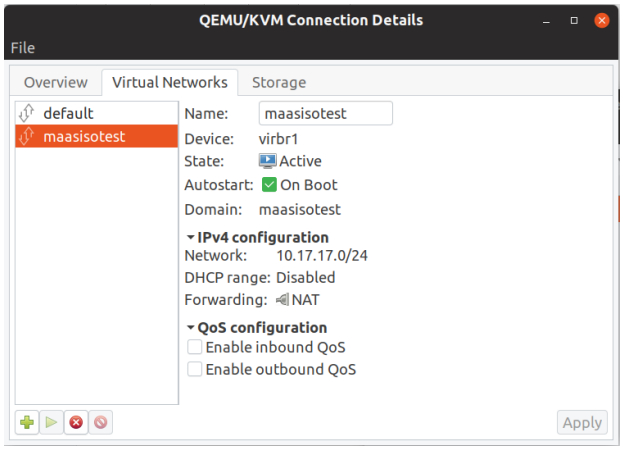

Fig. 4.1 Isolated network configuration

This will essentially create a router at the edge of this virtual network.

4.2 Creating a maas controller

Configurations for creating a new virtual machine with ubuntu 18.04 server installation. This machine will act as controller for the maas nodes.

- Memory - 1536 MB
- CPU Cores - 3
- Storage - 20 GB qcow2
- Name - maas-controller
- Network - maasisotest
- NIC Interface
  - Network source - maasisotest
  - Device model - virtio
- Disk bus - virtio
- Remove unnecessary virtual hardware from the list for eg. sound

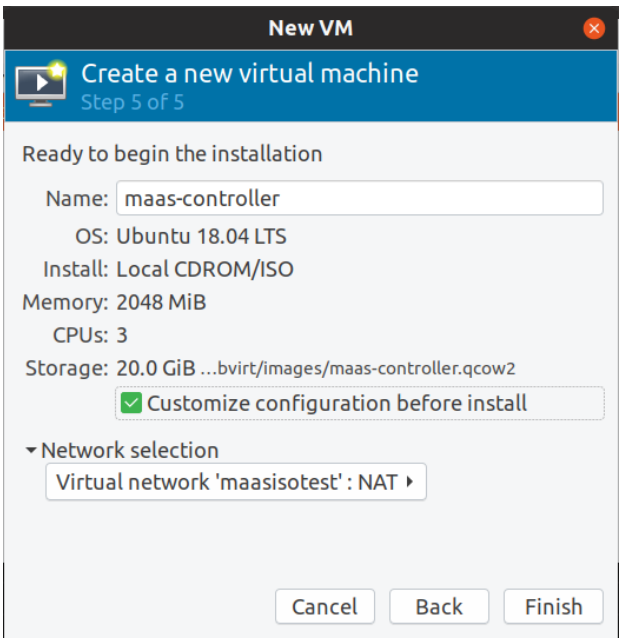

Fig. 4.2 maas-controller configuration

The basic configuration for maas-controller should look similar to Fig. 4.2.

Once the configuration is done we can proceed to installation.

4.3 Installating ubuntu server image

During the installation it is expected that DHCP acquisition will fail, since there is no DHCP server on that network. Hence, we need to configure this manually with a statiic IP address.

Fig. 4.3 IPv4 configuration

Kindly refer to the Fig. 4.3 for the IPv4 configuration.

Once the installation is finished, the system should reboot.

4.4 Setup MAAS

Use the following commands to install some utility packages and maas related packages on maas-controller. Then verify that you are able to connect through ssh from the host.

$ sudo apt-get install ssh iptraf htop wget lynx dnsutils
$ sudo apt-get install maas maas-dhcp maas-dns

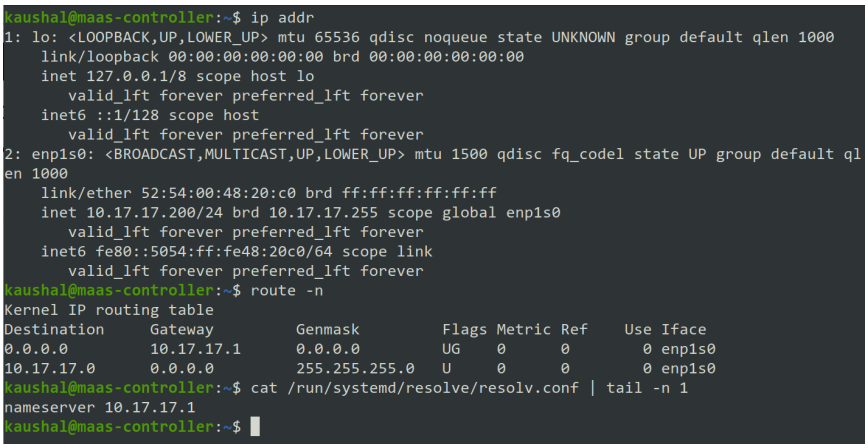

Fig. 4.4 verify internet connectivity

Verify the network connectivity by using the following commands:

$ ip addr
$ route -n
$ cat /run/systemd/resolve/resolv.conf — tail -n 1
$ dig canonical.com
$ ping google.com

The results should look similar to Fig. 4.4.

Verify that the following link works: `http://10.17.17.200:5240/`. It will open a MAAS interface which looks something similar to Fig. 4.5. Create a super-user by using the following command:

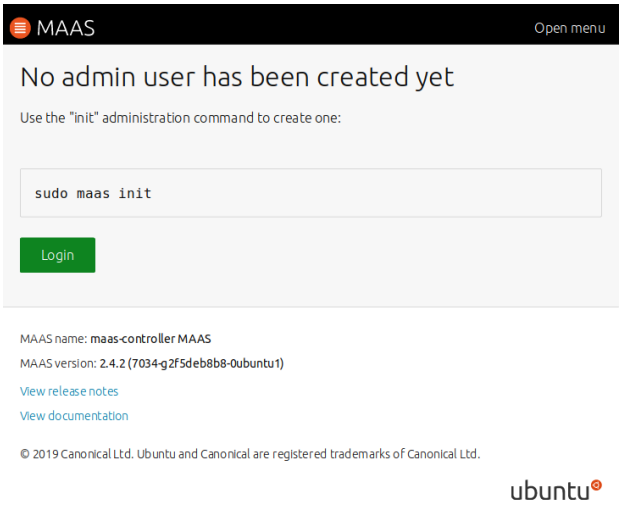

Fig. 4.5 MAAS initial page

$ sudo maas init

After setting up the maas superuser open the maas webui and login. You will land up on the page similar to Fig. 4-6.

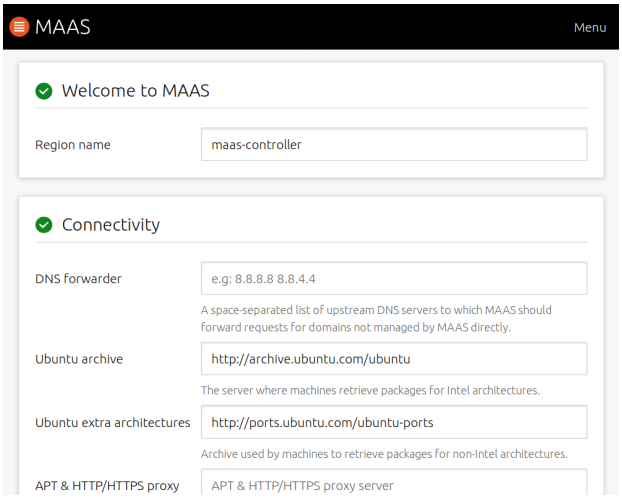

Fig. 4.6 MAAS intro page

4.5 SSH & DHCP configuration for MAAS

Change the default shell of the 'maas' user as shown in the Fig. 4.7.

Login to the maas user and generate an ssh key and copy the public key to the host for

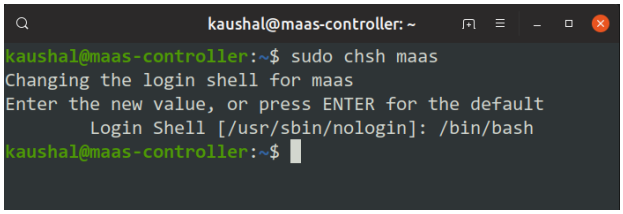

Fig. 4.7 Default shell for maas user

passwordless ssh authentication. That will enable us to open a ssh session at that virtual machine host without being prompted for a password. Verify that you are able to open an ssh session to the host system from the controller without being asked for the password.

```
$ sudo su - maas
$ ssh-keygen -t rsa
$ ssh-copy-id -i ~/.ssh/id_rsa.pub kaushal@10.128.0.132
```

Verify that maas will be able to control the virtual machines power at the host level. To do that we need to verify that the virsh command is working.

```
virsh -c qemu+ssh://kaushal@10.128.0.132/system list --all
```

It should list all the virtual machines at the host.

The next thing we need to do is to import the ssh key of maas controller into maas, so that when maas provisions a new node and brings up an image on that node it can inject our ssh key allowing us to remotely access those managed nodes without that we really have no way aside from backdoor and breaking in to access those nodes that we have created.

To do this first create a ssh key by following the above mentioned steps.

Go to the following link: `http://10.17.17.200:5240/MAAS/#/intro/user` and upload your ssh key.

Next we need to configure the DHCP configuration for the maas. To enable MAAS-managed DHCP, under the 'Subnets' page select the desired fabric and then VLAN.

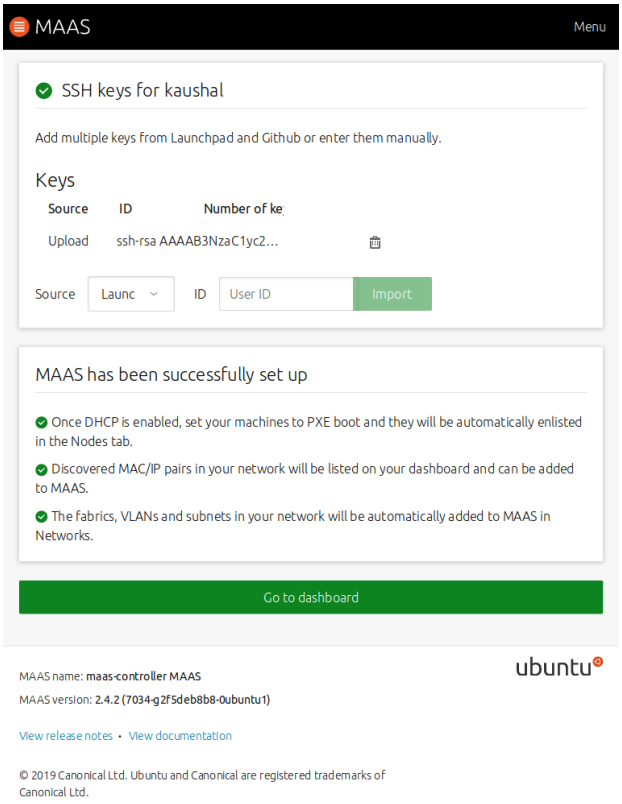

Fig. 4.8 Uploading the ssh key of maas controller

4.5.1 IP Ranges

During the DHCP configuration we need to configure the IP ranges. IP addresses can be reserved by adding one or more reserved ranges to a subnet configuration. There are two types of ranges that can be defined: reserved range and reserved dynamic range.

Reserved range mode operates differently depending on whether the subnet is **managed** or **unmanaged**. In managed (subnet), MAAS will never assign IP addresses inside this range. They can be used for anything (e.g. infrastructure systems, network hardware, or external DHCP). While in unmanaged (subnet), MAAS will only assign IP addresses inside this range.

A **Reserved dynamic range** is an IP range that MAAS will use for enlisting, commissioning and, if MAAS-managed DHCP is enabled on the node's VLAN during commissioning, deploying. An initial range is created as part of the DHCP enablement process if

done with the web UI. For an unmanaged subnet, this range is never used.

To keep things simple we will use the reserved dynamic range mode with managed subnet. By default, a subnet is managed, but it is easy to change this. To disable (or re-enable) subnet management navigate to the 'Subnets' page and select the subnet. Press the 'Edit' button so that changes can be made. The 'Managed allocation' field will become a slide switch. Click the label (or the switch icon itself) to toggle between enabled (dark blue) and disabled (grey) and click 'Save summary'.

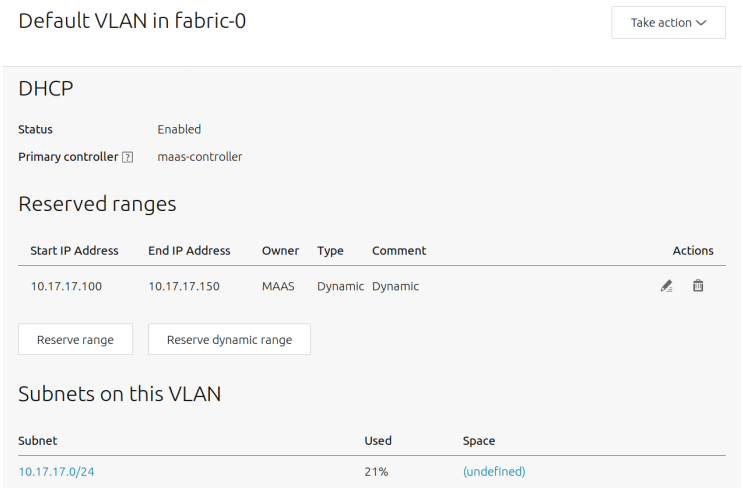

Fig. 4.9 Provide DHCP

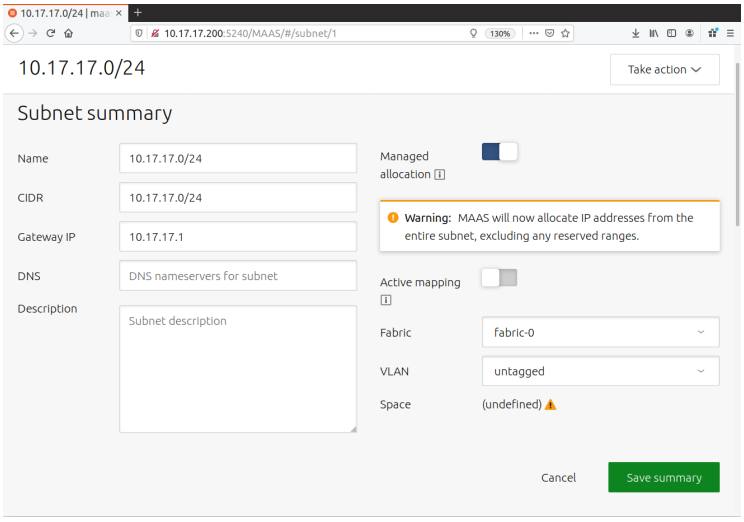

Fig. 4.10 Editing subnet

4.6 Images

MAAS is only useful once it has images available to provision its nodes with. Therefore, one key task once MAAS is installed is to select and import images from the boot source(place from where a region controller downloads its images). MAAS comes configured with a boot source that should suffice for most users:

`https://images.maas.io/ephemeral-v3/daily/`

Once images have been imported MAAS will update them on an hourly basis (a sync mechanism is enabled by default).

The 'Images' page shows what images and architectures have been selected and downloaded.

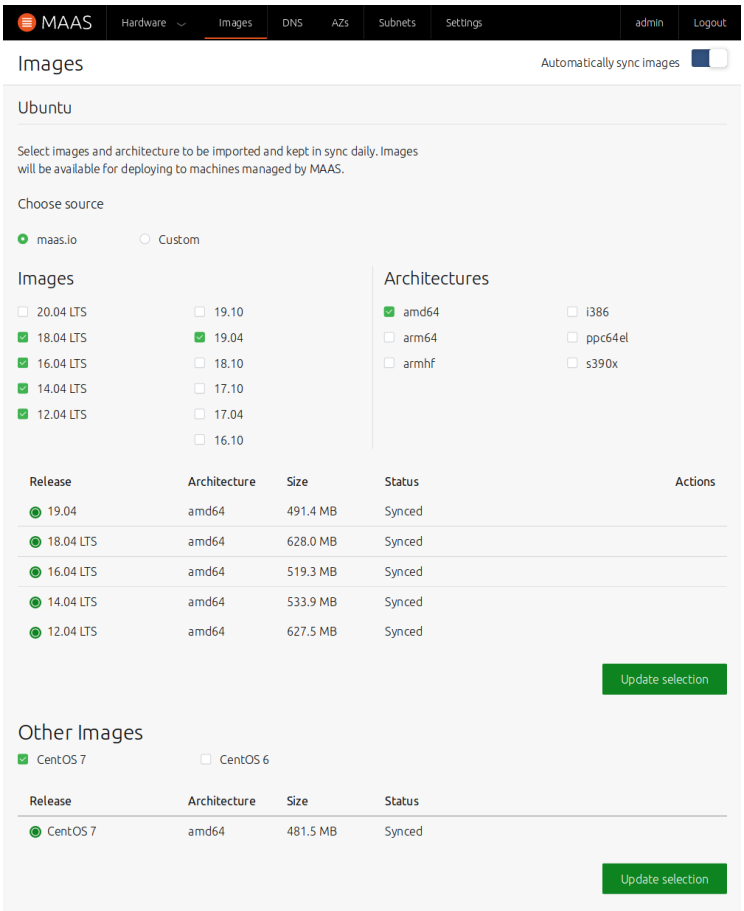

Fig. 4.11 Images

With this our basic configuration of the maas controller is done and in the next chapter we will create some nodes and add it to the maas network and perform node acquisition.

Chapter 5

MAAS in VENV - III : Nodes

5.1 Creating a few nodes

There is a restriction on creating virtual machine that it won't let you proceed until you specify an iso file. Hence, create a dummy iso file by using the command:

$ touch nothing.iso

We will start creating a single node and then we will clone that to make a few more nodes.

Configurations for creating a new virtual machine with dummy nothing.iso image:

- Memory - 1024 MB
- CPU Cores - 2
- Storage - 20 GB qcow2
- Name - node0
- Boot order - 1.) PXE 2.) HDD
- Network - maasisotest
- NIC Interface
  - Network source - maasisotest

  - Device model - virtio

- Disk bus - virtio
- Remove unnecessary virtual hardware from the list for eg. sound

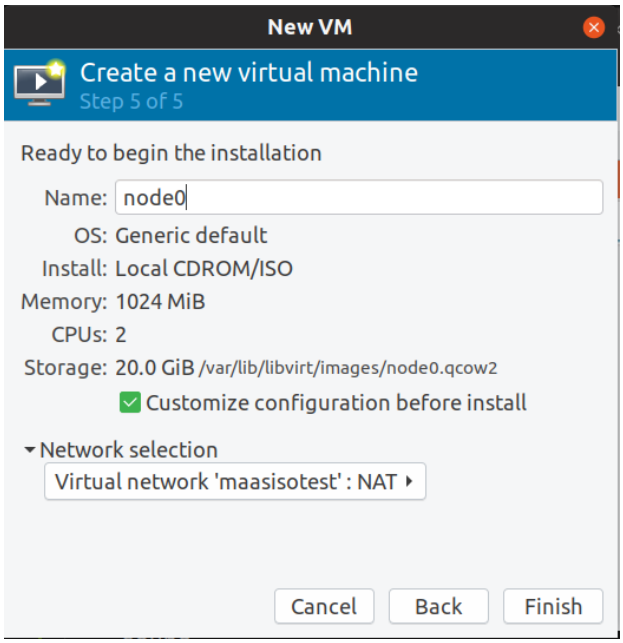

Fig. 5.1 Node 0

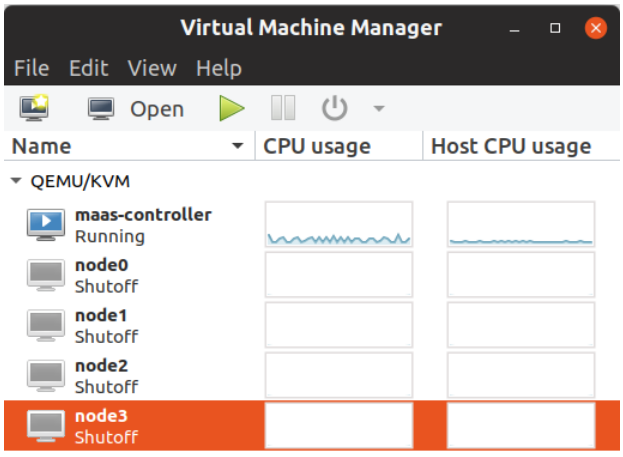

Fig. 5.2 Clones of node 0

Begin the installation and once the machine is open, pull the virtual power plug by force off, to create a few virtual machine clones.

Once the clones are created, power on all the virtual machines. These machines will boot for enlistment procedure. They will register themselves and shut themselves down.

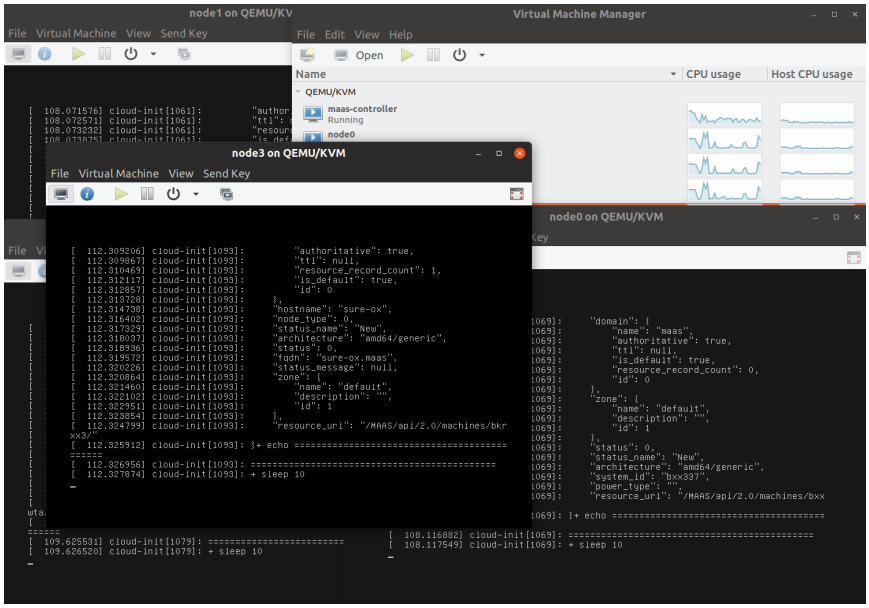

Fig. 5.3 Node enlisting themselves

5.2 Power configurations

As soon as they shutdown, we will start configuring the power parameters. Before we do that, we need to change the identifications names of the nodes in the maas interface. By default, the node's hostname according to MAAS is a randomly chosen string(Fig. 5.4).

First we need to find the mac address of the interface of a particular node. We can find this in the xml configuration file for the corresponding VM(Fig. 5.5).

$ virsh dumpxml node0 |grep mac

And then check for the device which has the same mac address in the maas interface and then set the name to node0. In the same way do for the other nodes.

Even though a script can be written to automate this task for the virtual environment

case but while operating with the actual hardware there is no other way to completely automate this task.

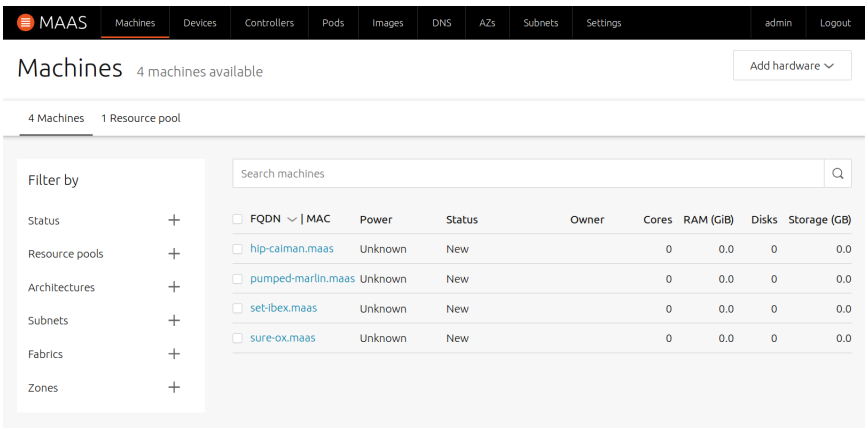

Fig. 5.4 Change these names to the meaningful names

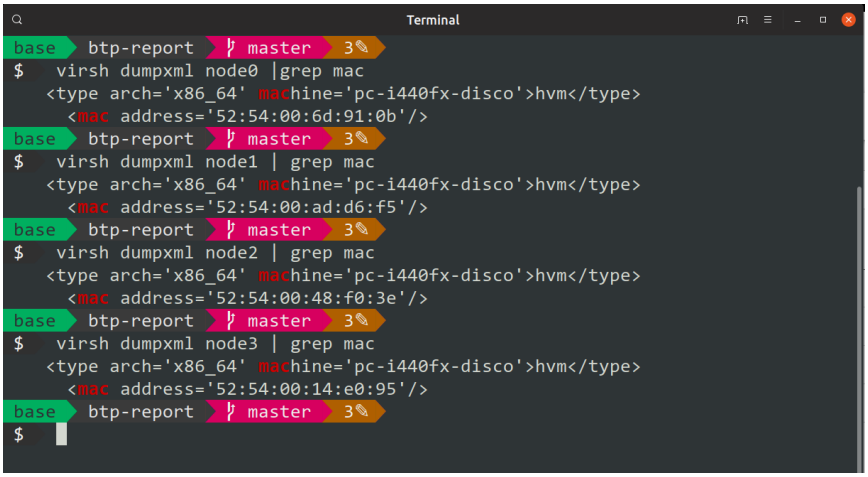

Fig. 5.5 Finding the mac address of each node

Once the names are changed, navigate to the power configuration under the configuration tab for each node and set the power parameters as follows:

Power type: Virsh

Virsh Address: `qemu+ssh://<USERNAME>@<IP>/system`

Virsh VM ID: `node0`

Where USERNAME is related to the host machine and the IP is the local ip of that host machine. For my case USERNAME is ‘kaushal’ and IP is ‘10.128.1.220’.

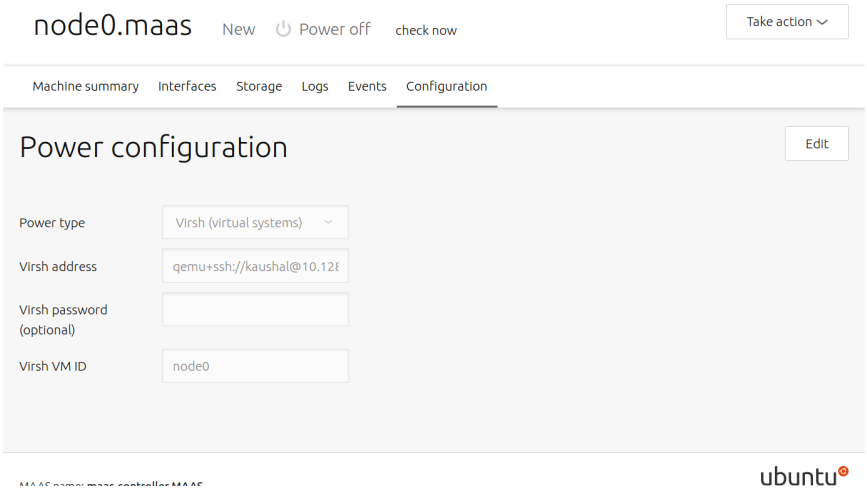

Fig. 5.6 Power configuration for node0

5.3 Commissioning the nodes

Once the power configuration is done select all nodes and commission all of them (fig 5.7). During the commissioning procedure the machine will also go through some hardware testing procedure (fig 5.8).

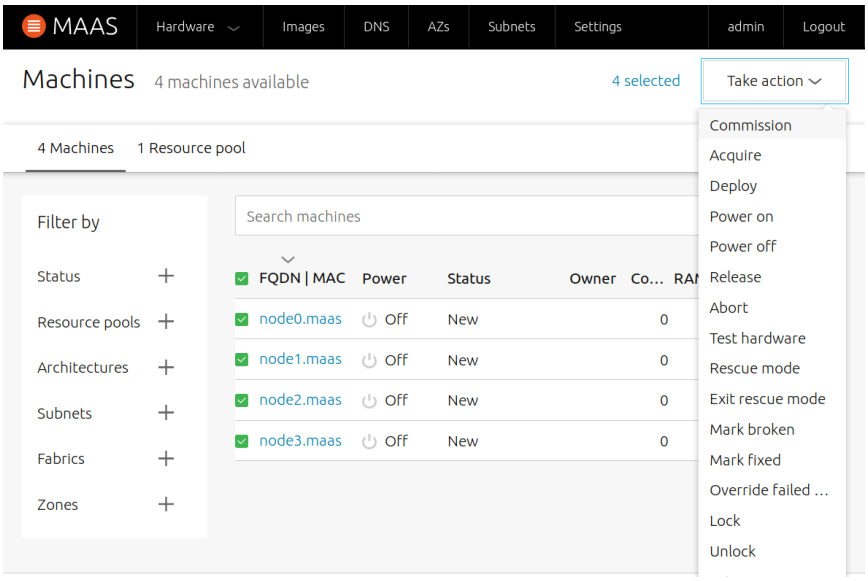

Fig. 5.7 Commissioning all nodes

Once the commissioning procedure is complete the machines will gracefully shutdown and on the MAAS web interface you will have the basic hardware specification(fig. 5.9).

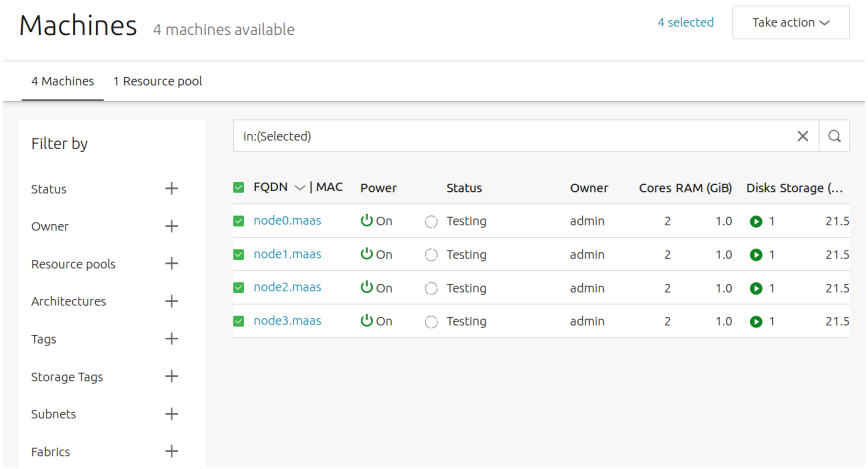

Fig. 5.8 Hardware testing phase

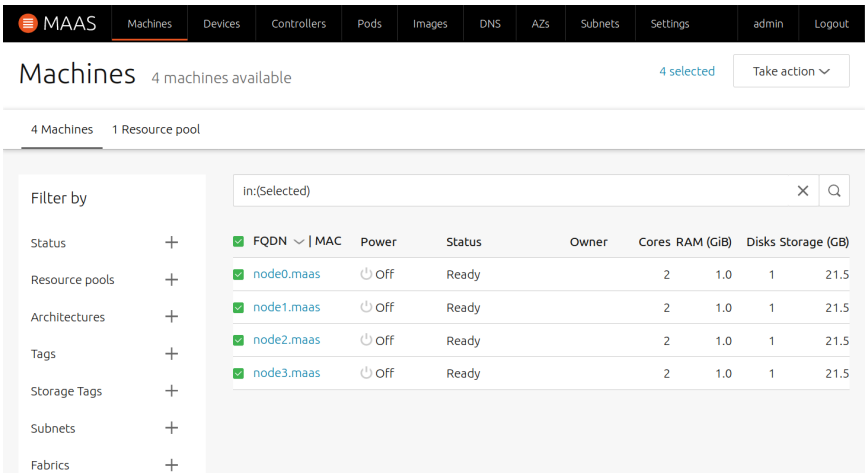

Fig. 5.9 Hardware information

This is a basic procedure of how to create MAAS nodes and commission them.

In the next chapter we will discuss the machine acquisition and deployment.

Chapter 6

MAAS in VENV - IV : Acquisition

After setting up the nodes and commissioning them, the next logical step is machine acquisition and deployment.

6.1 User management

One need to have a account to acuire a machine. An account can only be created by an admin. The account information can be managed in Users section in settings on the MAAS web interface(fig. 6.1).

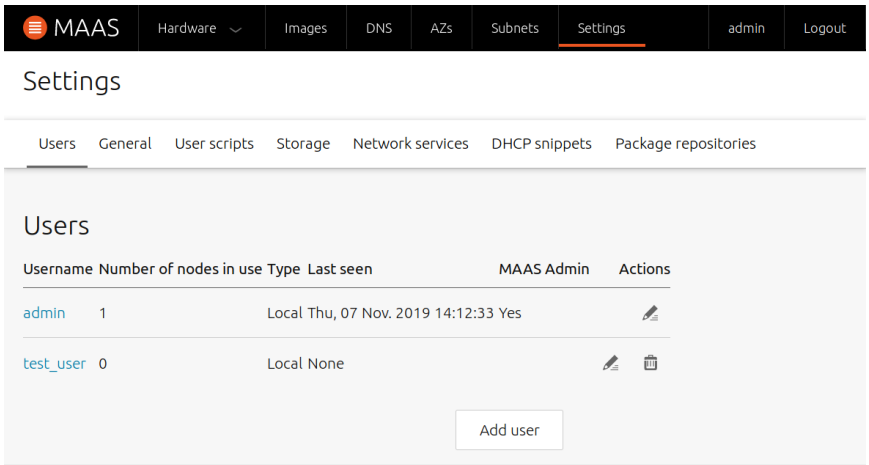

Fig. 6.1 User management

We will create a test user with no administrator privileges.

6.2 Acquiring machine

Chapter 7

Future Work

- Fix some occasional errors while commissioning nodes due to netowork and connectivity related problems.
- Production ready maas based virtual cloud.
- Moving out of the virtual platform to physical racks and controllers.