

Cloud Computing with Open Source Tool :OpenStack

Dr. Urmila R. Pol

Department Of Computer Science , Shivaji University, Kolhapur.

ABSTRACT: *OpenStack is a especially scalable open source cloud operating system that is a global alliance of developers and cloud computing technologists producing the ubiquitous open source cloud computing platform for public and private clouds. OpenStack provides series of interrelated projects delivering various components for a cloud infrastructure solution as well as controls large pools of storage, compute and networking resources throughout a datacenter that all managed through a Dashboard(Horizon) that gives administrators control while empowering their users to provision resources through a web interface. In this paper, we present a overview of Cloud Computing Platform such as, Openstack, Eucalyptus ,CloudStack and Opennebula which is open source software, cloud computing layered model, components of OpenStack, architecture of OpenStack. The aim of this paper is to show mainly importance of OpenStack as a Cloud provider and its installation.*

KEYWORDS: *Cloud Computing, OpenStack ,Eucalyptus, cloudStack , Open Nebula*

I.

INTRODUCTION

Cloud computing is relatively a new concept for which the resources are dynamically extended, virtualized as well as provided as a service on the Internet, it also allow providers to give users access to a virtually unlimited number of resources i-e outsourcing of resources[1]. OpenStack was founded by NASA and Rackspace Hosting which is rapidly grown to be a global software community of developers collaborating on a standard and massively scalable open-source cloud operating system. Cloud computing is a quite new concept that brings together all technologies (Web services, virtualization, service oriented architecture, grid computing, etc.) and business models used to deliver IT capabilities (software, platforms, hardware) as a service request, scalable and elastic. OpenStack is an industry initiative based on a global collaboration of developers and cloud computing technologists producing the open standard cloud computing operating system for both public and private clouds.

II. CLOUD COMPUTING

Cloud Computing [5] is a modern computing paradigm that providing IT infrastructure and it is very essential requirement for the IT companies. Cloud Computing providing essential service i.e. infrastructure as a service (IaaS), network as a service (NaaS), platform as a service (PaaS), software as a service (SaaS). Cloud computing is a model for enabling ubiquitous, on-demand network access to a shared pool of configurable computing resources such as network, servers, storage, applications, and services that can be rapidly provisioned and released with minimal management effort. Cloud Clients Can access and use the services of cloud applications using browsers, mobile devices, while all the data as well as software is stored on servers at a remote location, which are also used to perform all the heavy duty processing.

Infrastructure as a service (IaaS), is the most basic and essential cloud service model under which virtual machines, load balancers, raw block storage, firewalls and networking services are provided. In The platform as a service model, a computing platform including APIs, operating system, development environments, programming language execution environment and web server are typically provided. But In the software as a service model, cloud providers install and operate application software in the cloud. Overall, a cloud computing layered model is very important as well as main aims to provide benefits in terms of lesser up-front investment in infrastructure during deployment, higher scalability, lower operating costs, ease of access through the Web, reduced business risks and maintenance expenses.

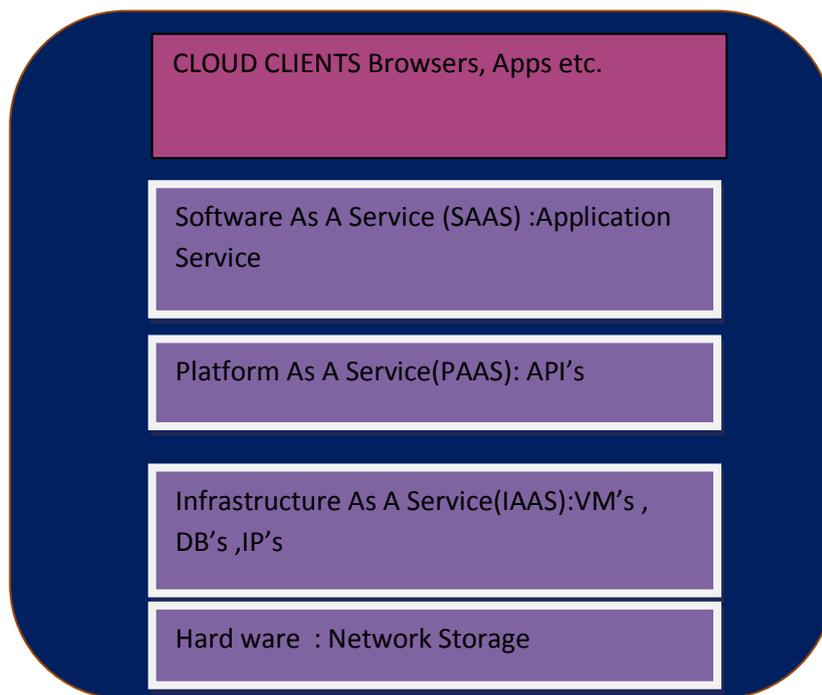


Fig 1. The Cloud-computing layered model

III. OPEN SOURCE CLOUD PLATFORM

A. OpenStack : OpenStack [3,5] is the fastest growing free open source software was announced in July 2010, but initial contributors are NASA and Rackspace. Rackspace contributed their "Cloud Files" platform (code) while NASA contributed their "Nebula" platform (code). OpenStack open source software is a collection of open source software projects that cloud computing technologists can use to setup and run their cloud compute and storage infrastructure. OpenStack provides an Infrastructure-as-a-Service (IaaS) solution through a variety of complementary services. Each service offers an application programming interface (API) that facilitates this integration. Dashboard Service project name Horizon Provides a web-based self-service portal to interact with underlying OpenStack services, such as launching an instance, assigning IP addresses and configuring access controls. Compute service project name Nova Manages the lifecycle of compute instances in an OpenStack environment. Responsibilities include spawning, scheduling and decommissioning of virtual machines on demand. Networking service project name Neutron Enables Network-Connectivity-as-a-Service for other OpenStack services, such as OpenStack Compute. Provides an API for users to define networks and the attachments into them. Has a pluggable architecture that supports many popular networking vendors and technologies. Object Storage project name Swift Stores and retrieves arbitrary unstructured data objects via a RESTful, age HTTP based API. It is highly fault tolerant with its data replication and scale out architecture. Its implementation is not like a file server with mountable directories.

Block Storage service project name Cinder Provides persistent block storage to running instances. Its pluggable driver architecture facilitates the creation and management of block storage devices. Identity service project name Keystone Provides an authentication and authorization service for other OpenStack services. Provides a catalog of endpoints for all OpenStack services. Image Service project name Glance Stores and retrieves virtual machine disk images. OpenStack Compute makes use of this during instance provisioning. Telemetry service project name Ceilometer Monitors and meters the OpenStack cloud for billing, benchmarking, scalability, and statistical purposes. Orchestration service project name Heat Orchestrates multiple composite cloud applications by using either the native HOT template format or the AWS CloudFormation template format, through both an OpenStack-native REST API and a CloudFormation-compatible Query API. Database Service project name Trove Provides scalable and reliable Cloud Database-as-a-Service functionality for both relational and non-relational database engines.

B. Eucalyptus : EUCALYPTUS[3,5] is the acronym for Elastic Utility Computing Architecture for Linking Your Program to Useful System, which is an open source private cloud software for building private or hybrid cloud resources for compute, network, and storage that are compatible with Amazon Web Service (AWS) APIs. It was developed by University of California-Santa Barbara for Cloud Computing to implement Infrastructure as a Service (IaaS). Eucalyptus [3] provide an Elastic Compute Cloud (EC2) -compatible cloud Computing Platform and Simple Storage Service (S3)-compatible Cloud Storage. Eucalyptus has some high-level components such as Cloud Controller (CLC), Cluster Controller (CC), Storage Controller (SC), and Node Controller (NC). The main benefits to use this open source software for private clouds which provide highly efficient, scalability, organization agility.

C. CloudStack : CloudStack,[5] initially developed by Cloud.com, it was purchased by Citrix then later on released into the Apache Incubator program. The first stable version of CloudStack was released in 2013 and in this time governed by the Apache Software Foundation and supported by Citrix. CloudStack support some commendable features such as storage independent compute, new security features, Smooth Deployment, Scalability, Multi Hypervisor support, Detailed Documentation and Interactive Web UI. **CloudStack's followed monolithic** architecture which posed some challenges one of them being reduced installation flexibility.

D. OpenNebula : OpenNebula [3,5,6] was first established as a research project back in 2005 by Ignacio M. Liorente and Ruben S. Montero, which is used by many enterprises as an open, flexible alternative to vCloud on their VMware-based data center. OpenNebula is primarily used as a virtualization tool to manage virtualized infrastructure in the data center, which is usually referred as private cloud and supports hybrid cloud to combine local infrastructure with public cloud-based infrastructure, enabling highly scalable hosting environments. OpenNebula cloud infrastructure provide users with an elastic platform for fast delivery and scalability of services and also support Public cloud by providing cloud interfaces to expose its functionality for virtual machine, storage and network management.

IV. OPENSTACK

OpenStack [4] is a set of software tools for building and managing cloud computing platforms for public and private clouds as well as it is a collection of open source software projects which provides an Infrastructure-as-a-Service (IaaS) solution through a set of interrelated services. In July 2010 NASA and Rackspace Hosting combindly launched an open-source cloud-software initiative which is known as OpenStack. OpenStack [3] code came from NASA's Nebula platform and from Rackspace's Cloud Files platform. OpenStack, project is an open source cloud computing platform which provide the ubiquitous open source cloud computing platform for public as well as private clouds, also it is free and open-source software released under the terms of the Apache License. The cloud is mainly providing computing features for end users in a remote environment, where the actual software runs as a service on reliable, scalable servers rather than on each end users computer. OpenStack give facilities for deploy virtual machines (VMs) and other instances which handle different tasks for managing a cloud environment. It provides horizontal scaling very easy, which means that tasks which benefit from running concurrently can easily serve more as well as less users on the fly by just spinning up more instances[2].

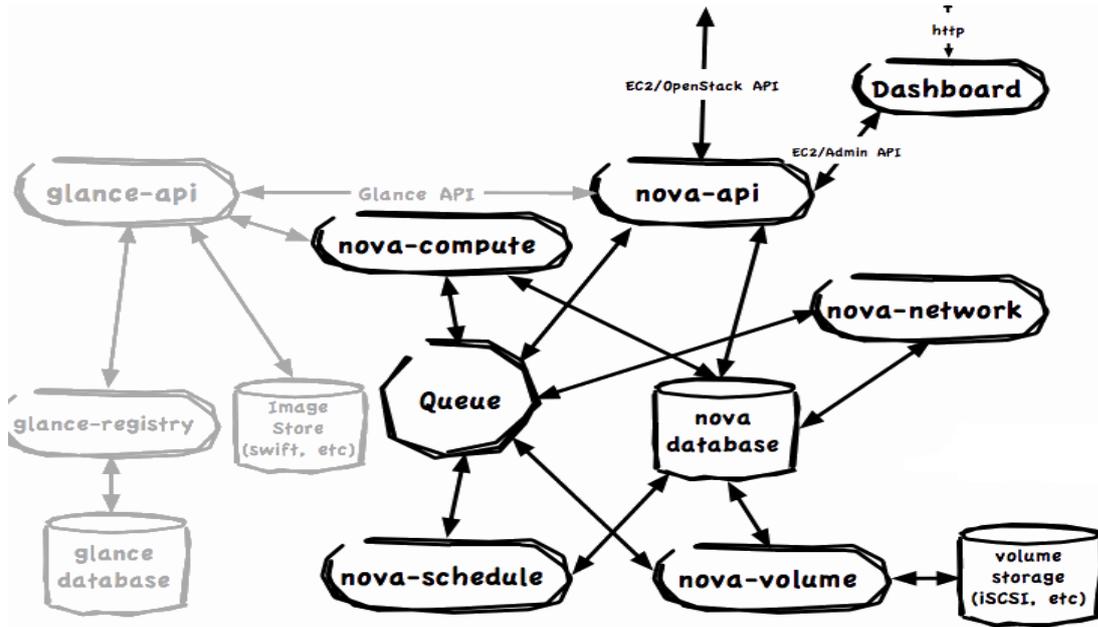


Fig 2. OpenStack Architecture

Components of OpenStack:

- **Compute (Nova):** OpenStack Compute (Nova) is a cloud computing fabric controller, which is used for deploying and managing large numbers of virtual machines and other instances to handle computing tasks.
- **Object Storage (Swift):** OpenStack Object Storage (Swift) is a scalable redundant storage system for objects and files. Objects as well as files are written to multiple disk drives spread throughout servers in the data center, OpenStack software only responsible for ensuring data replication and integrity across the cluster.
- **Block Storage (Cinder):** OpenStack Block Storage (Cinder) is a block storage component, which is more analogous to the traditional notion of a computer being able to access specific locations on a disk drive as well as it provides persistent block-level storage devices for use with OpenStack compute instances. In OpenStack, the block storage system manages the creation, attaching, detaching of the block devices to servers.
- **Networking (Neutron):** OpenStack Networking (Neutron) provides the networking capability for OpenStack and it is a system for managing networks and IP addresses easily, quickly and efficiently.
- **Dashboard (Horizon):** OpenStack Dashboard (Horizon) is the dashboard behind OpenStack which provides administrators and users a graphical interface to access, provision and automate cloud-based resources.
- **Identity Service (Keystone):** OpenStack Identity (Keystone) provides identity services for OpenStack or it is a central directory of users mapped to the OpenStack services they can access. It provides multiple means of access, and acts as a common authentication system across the cloud operating system and can integrate with existing backend directory services like LDAP.
- **Image Service (Glance):** OpenStack Image Service (Glance) provides image services to OpenStack, discovery, registration and delivery services for disk and server images, it also allows these images to be used as templates when deploying new virtual machine instances.
- **Telemetry (Ceilometer):** OpenStack Telemetry Service (Ceilometer) provides telemetry services, which allow the cloud to provide billing services to individual users of the cloud, it keeps a verifiable count of **each user's system usage of each of the various components of an OpenStack cloud.**

- **Orchestration (Heat):** OpenStack Orchestration (Heat) is a service which allows developers to store the requirements of a cloud application in a file that defines what resources are necessary for that application.
- **Database (Trove):** OpenStack (Trove) is a database as a service which provides relational and non-relational database engines.

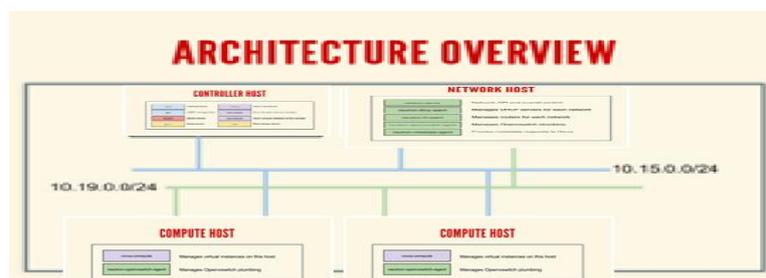
V. RDO HANGOUT: MULTINODE OPENSTACK WITH PACKSTACK

A multinode OpenStack install using packstack contains **One controller, One network host and Two compute hosts**. Following open tools are needed for installation **CentOS 6.5 ,RDO Havana, Packstack**.

What is packstack?

A command-line tool for automating the deployment of simple OpenStack clouds.

- Single host (--allinone) or multinode
- Proof of Concept ("PoC") deployments



Above figure describes architectural overview of openstack. It contains Controller node, Network host and two compute hosts. In controller host OpenStack provides an Infrastructure as a Service (IaaS) solution through a set of interrelated services. Each service offers an application programming interface (API) that facilitates this integration. Depending on your needs, you can install some or all services. The following table describes the OpenStack services that make up the OpenStack architecture:

Table 1.1. OpenStack services

TABLE 1.1. OPENSTACK SERVICES

Service	Project name	Description
Dashboard	Horizon	Provides a web-based self-service portal to interact with underlying OpenStack services, such as launching an instance, assigning IP addresses and configuring access controls.
Compute	Nova	Manages the lifecycle of compute instances in an OpenStack environment. Responsibilities include spawning, scheduling and decommissioning of machines on demand.
Networking	Neutron	Enables network connectivity as a service for other OpenStack services, such as OpenStack Compute. Provides an API for users to define networks and the attachments into them. Has a pluggable architecture that supports many popular networking vendors and technologies.
Storage		
Object Storage	Swift	Stores and retrieves arbitrary unstructured data objects via a RESTful, HTTP based API. It is highly fault tolerant with its data replication and scale out architecture. Its implementation is not like a file server with mountable directories.
Block Storage	Cinder	Provides persistent block storage to running instances. Its pluggable driver architecture facilitates the creation and management of block storage devices.
Shared services		
Identity Service	Keystone	Provides an authentication and authorization service for other OpenStack services. Provides a catalog of endpoints for all OpenStack services.
Image Service	Glance	Stores and retrieves virtual machine disk images. OpenStack Compute makes use of this during instance provisioning.
Telemetry	Ceilometer	Monitors and meters the OpenStack cloud for billing, benchmarking, scalability, and statistical purposes.
Higher-level services		
Orchestration	Heat	Orchestrates multiple composite cloud applications by using either the native HOT template format or the AWS CloudFormation template format, through both an OpenStack-native REST API and a CloudFormation-compatible Query API.

Getting started**Set up ssh**

Make sure that you can ssh as root to all your hosts from wherever you're running packstack.

Install packstack

Make the RDO repositories available:

```
# yum install -y http://rdo.fedorapeople.org/rdo-release.rpm
```

And install packstack:

```
# yum -y install openstack-packstack
```

The answers file

You can set all sorts of parameters on the command line...

```
packstack --allinone --os-quantum-install=y --provision-demo=n \
  --provision-all-in-one-ovs-bridge=n
```

...but I like to generate an "answers" file and edit it:

```
# packstack --gen-answer-file packstack-answers.txt
```

Run packstack

```
# packstack --answer-file packstack-answers.txt
```

Welcome to Installer setup utility

Packstack changed given value to required value /root/.ssh/id_rsa.pub

Installing:

```
Clean Up... [ DONE ]
Setting up ssh keys... [ DONE ]
Discovering hosts' details... [ DONE ]
Adding pre install manifest entries... [ DONE ]
Adding MySQL manifest entries... [ DONE ]
Adding QPID manifest entries... [ DONE ]
Adding Keystone manifest entries... [ DONE ]
Adding Glance Keystone manifest entries... [ DONE ]
Adding Glance manifest entries... [ DONE ]
Installing dependencies for Cinder... [ DONE ]
Adding Cinder Keystone manifest entries... [ DONE ]
```

What just happened?

- Lots of packages installed
- Configuration in /etc/{nova,glance,cinder,neutron,...}/
- Firewall rules for all services
- MySQL users and databases created
- Services started

Post-install Configuration**Fix Horizon**

You may need to fix ALLOWED_HOSTS:

```
# sed -i '^ALLOWED_HOSTS/ s/=.*/= [ "*" ]/' \
  /etc/openstack-dashboard/local_settings
```

```
# service httpd restart
```

Source your admin credentials:

```
# . /root/keystonerc_admin
```

Create a disk image:

```
glance image-create \
  --copy-from http://download.cirros-cloud.net/0.3.1/cirros-0.3.1-x86_64-disk.img \
  --is-public true \
  --container-format bare \
  --disk-format qcow2 \
  --name cirros
```

Create external network:

```
# neutron net-create external --router:external=True
# neutron subnet-create --disable-dhcp external 172.16.13.0/24
```

Create a flavor for testing:

```
# nova flavor-create m1.nano auto 128 1 1
```

This flavor consumes minimal memory and disk so it is better than the default flavors for testing in constrained environments.

Create a non-admin user:

```
# keystone tenant-create --name demo
# keystone user-create --name demo --tenant demo --pass demo
```

And store the credentials in /root/keystonerc_demo:

```
export OS_USERNAME=demo
export OS_TENANT_NAME=demo
export OS_PASSWORD=demo
export OS_AUTH_URL=http://192.168.2.131:35357/v2.0/
export PS1='\u@\h \W(keystone_demo)]\$ '
```

Switch credentials

From this point on we're going to be operating as the "demo" user:

```
# ./root/keystonerc_demo
```

Create an ssh keypair

Create a keypair:

```
# ssh-keygen -t rsa -b 2048 -N "" -f id_rsa_demo
```

Create tenant networks

Create a private network:

```
# neutron net-create net0
# neutron subnet-create --name net0-subnet0 \
  --dns-nameserver 8.8.8.8 net0 10.0.0.0/24
```

Create security rules

Make sure we allow ICMP and SSH traffic to instances:

```
# neutron security-group-rule-create --protocol icmp default
# neutron security-group-rule-create --protocol tcp \
  --port-range-min 22 --port-range-max 22 default
```

Booting an instance

We'll need the UUID for network net0 that we created in the previous step:

```
# nova boot --poll --flavor m1.nano --image cirros \
  --nic net-id=77cafb07-a793-41cb-8a96-58d04408e10d \
  --key-name demo test0
```

Create and assign a floating ip

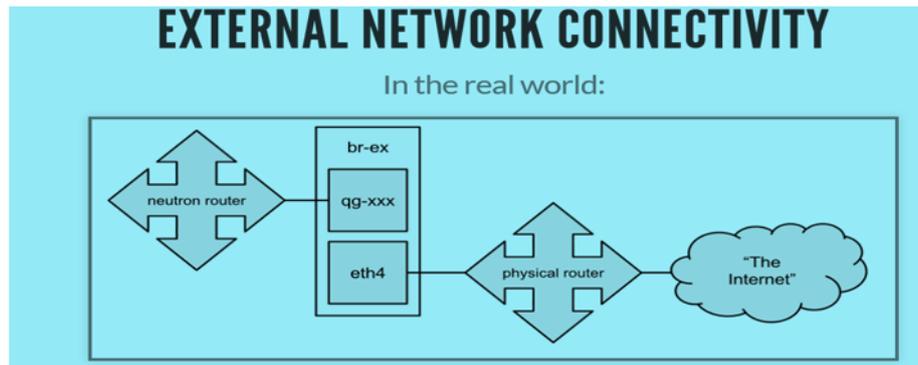
Allocate a floating ip address from the *external* network:

```
# nova floating-ip-create external
```

```
+-----+-----+-----+-----+
| Ip      | Instance Id | Fixed Ip | Pool    |
+-----+-----+-----+-----+
| 172.16.13.3 | None      | None     | external |
+-----+-----+-----+-----+
```

Assign it to the new instance:

```
# nova add-floating-ip test0 172.16.13.3
```



VI. CONCLUSIONS

This paper compares the four most popular and commonly used open source software which are OpenStack, Eucalyptus, CloudStack, OpenNebula and; these all open source software allow users to choose better services according to their requirement. OpenStack is an open source software used for designed to allow researchers and administrators to deploy infrastructure as a service (IaaS) and provide tools for creating and managing virtual machines (VMs) on top of existing resources.

REFERENCES

- [1] Paramjot Singh, Vishal Pratap Singh, Gaurav Pachauri, "Critical Analysis of Cloud Computing Using OpenStack" IJCSMC, Vol. 3, Issue. 3, March 2014, pg.121 – 127
- [2] OpenFlow as a Service; Fred Hsu, M. Salman Malik, Soudeh Ghorbani {fredhsu2,mmalik10,ghorban2}@illinois.edu
- [3] Omar SEFRAOUI, Mohammed AISSAOUI, Mohsine ELEULDJ; "*OpenStack: Toward an Open-Source Solution for Cloud Computing*", International Journal of Computer Applications (0975 - 8887) Volume 55 - No. 03, October 2012
- [4] OpenStack URL: <http://www.openstack.org/>
- [5] CloudStack URL: <http://www.cloudstack.apache.org/>
- [6] Open Nebula URL: <http://openebula.org/>