Private Cloud for Organizations: An Implementation using OpenStack

P Aruna, L Yamuna Devi, D Sudha Devi, N Priya, Dr. S.Vasantha and Dr. K.Thilagavathy

Abstract— Cloud Computing has been rapidly and radically changing the dynamics of IT consumption. The core cloud computing proposition is being able to buy compute technology as a service in an on-demand, elastic, and pay-as-you-go model. Public cloud offerings to the IT industry are very prominent. But the organizations, which have already invested money in their own infrastructure, are interested in setting up the private clouds within their organizational boundaries to avail the benefits of cloud computing technologies. Private Cloud solutions can reduce costs, help to efficiently utilize the resources, minimize security risks, and help meet many core IT goals and requirements. The fundamental resources like processing power, storage capacity and network can be efficiently used by leveraging the benefit of private cloud in an institution. As Public cloud is proprietary, the user could not customize it according to their requirements. Hence for researchers, students and institutions that needs customization, private cloud is a treat because they could implement it in their laptop, experience the cloud environment, carry out the experiments and execute their own algorithms. This paper tries to address the challenges towards Public Cloud, a possible alternative, that is, “Private Cloud” and emphasis the advantages of moving to Private Cloud. This paper also describes the process of setting up a Private Cloud environment in an institution using the open source technology Openstack, the various services it offers to the different stakeholders of the institution.

Index Terms— Cloud Computing, IaaS, OpenStack, Private Cloud, Resource Utilization, Virtualization.

1 Introduction

The NIST definition of Cloud Computing is “Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction”[5]. Several trends are opening up the era of Cloud Computing, which is an Internet-based development and pay per use model. This model appeals especially to startups and small organizations that have not invested in hardware resources and are looking for ways to avoid the large capital expenditure involved in procuring infrastructure upfront. Virtualization is the key technique for cloud computing. Virtualization is a technical innovation designed to increase the level of system abstraction and enable IT users to harness ever-increasing levels of computer performance.[1] This Cloud computing is available in three service models, and four deployment models.

A service is defined as a fine-grained reusable resources (i.e., infrastructure or business processes) available from a service provider; this is now what is popularly called —as a service. Cloud computing offers the following service models:

Software as a Service (SaaS) is a kind of application that is available as a service to users; it delivers software as a service over the Internet, eliminating the need to install and run the application on local computers in order to simplify the maintenance and support. The benefits of SaaS are lower cost, user familiarity with WWW, and web availability and reliability.

Platform as a Service (PaaS) model enables the deployment of applications without the cost and complexity of buying and managing the underlying hardware and software layers. A customer can deploy an application directly on the cloud infrastructure (without managing and controlling that infrastructure) using the programming languages and tools supported by a provider. A customer has the control over its applications and hosting environment’s configurations.

Infrastructure as a Service (IaaS) delivers a computer infrastructure that is a fundamental resource like processing power, storage capacity and network to customers; instead of building data centers, purchasing servers, software or network equipments, a customer buys the resources as a fully outsourced service; a customer does not manage the underlying infrastructure but has full control over the operating systems and the applications running on it. IaaS models often provide automatic support for on demand scalability of computing and storage resources [1].

Public cloud like Amazon AWS[2] , Google AppEngine[3], Microsoft Azure[4] offer infrastructure and platform as a service that can be procured by the user over the internet. Even though there are many public cloud offering there is in need of private cloud for some organization which already invested most of their capital expenses for building the infrastructure such as computing power, storage etc.

This paper is organized as follows: Section 2 introduces the various deployment models of Cloud computing, compares the major deployment models namely public and private clouds, and justifies the adoption of private cloud deployment for institution, Section 3 gives the architecture of our private cloud solution OpenStack, Section 4 gives the steps to implement a private cloud using the open source technology “Openstack”, Section 5 briefs about the creation of virtual machine, access the VM across different platforms and the different services availed from the private cloud, and Section 6 concludes this paper.

2 Deployment models of cloud computing

Fig. 1 shows the most commonly known deployment models as defined by National Institute of Standards and Technology (NIST) [5].

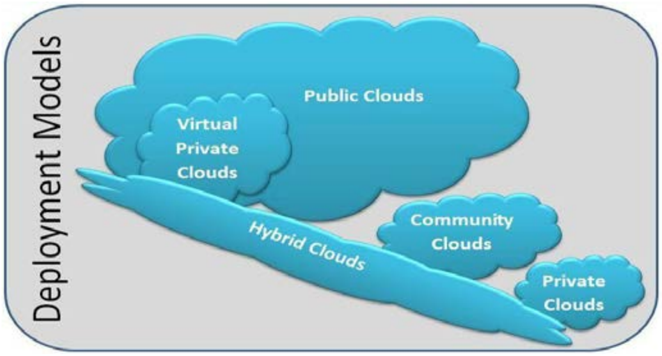


Fig. 1. Deployment models of Cloud

Private cloud: The cloud infrastructure is operated solely for an organization. It may be managed by the organization or a third party and may exist on premise or off premise.

Community cloud: The cloud infrastructure is shared by several organizations and supports a specific community that has shared concerns (e.g., mission, security requirements, policy, and compliance considerations). It may be managed by the organizations or a third party and may exist on premise or off premise.

Public cloud: The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

Hybrid cloud: The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).

Among the many available deployment models of cloud computing, the most widely used ones are Public and Private Clouds. There are many risks in selecting a public cloud offering such as security, privacy, interoperability and performance. Public Cloud offerings generally come with standard Service Level Agreement (SLA)s. They are not flexible to use according to the specific needs of institutions.

Public cloud is beneficial for the start up industry to cut down its capital expenses. But for the institution which have already invested their money in the infrastructure they can utilize their resources efficiently with the help of private cloud setup. Whenever there is any demand for higher end resources, they could do cloud bursting (hybrid cloud) to meet the demands. This implies that they need not invest the money on infrastructure once again.

For research people, students and an institution whoever doing their research work in the virtual machine is beneficial, since any crash in the machine does not affect the hardware. Licensed software loaded in the virtual machine can be leveraged as a service so that the amount spent on getting license would be minimized.

3 OPENSTACK ARCHITECTURE

Open source software for setting up a private cloud is widely available. OpenStack, Eucalyptus, OpenNebula are the main competitors in the private cloud area. Among them OpenStack has a more user-friendly GUI. OpenStack is a cloud computing project for providing the services. It was founded by NASA and RackSpace in 2010. It is a cloud operating System, a new management layer that adds automation and control for pool of resources such as computation, storage and network, empower administrators & users via self-service portals and empower developers to make apps cloud-aware via APIs. The project is managed by the OpenStack Foundation, a non-profit corporate entity established in September 2012 to promote, protect and empower OpenStack software and its community.

The major advantage of OpenStack is using the same API's as Amazon, which makes cloud bursting or a hybrid cloud solution easily implementable. OpenStack is a cloud computing platform which is EC2 and S3 compatible, and since its services can be reached using EC2 and S3 compatible APIs, any client tools written for AWS can communicate with OpenStack as well [1]. It is customizable thus, OpenStack seems like a treat to researchers who are working in cloud environment. They could experience the cloud, make studies on the algorithms used in cloud and exploit it as a real test bed for their work. They can implement a private cloud in their laptop and do their work not depending on the public cloud which has all proprietary components.

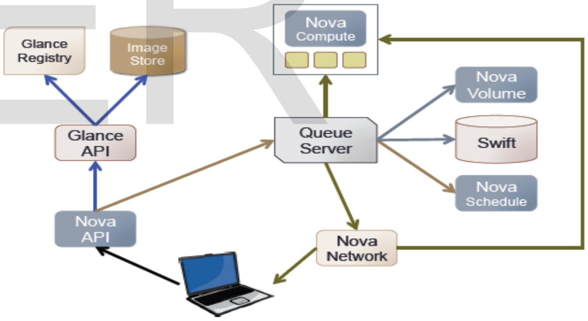


Fig. 2. OpenStack Architecture

OpenStack is horizontally and massively scalable, hardware agnostic and hypervisor agnostic. It support s for Xen , Citrix XenServer, Microsoft Hyper-V, KVM, UML, LXC and ESX.

Fig. 2 gives the architecture diagram of OpenStack. OpenStack (Essex) has five components namely Keystone, Glance, Nova, Swift and Horizon. Keystone provides the following services. It authenticates the user, enforces policies for different users and maintains the service catalog. Glance keeps a database of metadata associated with an image for registering. It is used to retrieve and discover the image when needed. It is built on top of Swift where images are stored. Glance has two services, Glance-api: public interface for uploading and managing images; Glance-registry: private interface to metadata database. Glance supports multiple image formats like iso, ami and vdi etc.

Among the current core projects of OpenStack, Nova project is the core of the cores. The nova-compute of Nova has five major components. Nova-API is the public facing interface. Message Queue acts as a Broker to handle interactions between services, currently based on RabbitMQ. The Central component that manages the allocation of virtual resources for a cloud infrastructure's physical resources is known as the cloud scheduler. Openstack refers to its cloud scheduler component using the name "nova-scheduler" [6]. Nova-Scheduler coordinates all services, determines the placement of new resources requested. Nova-compute hosts VMs, controls hypervisor and VMs when receives commands on Message Queue. Nova-volume manages permanent storage. Nova-network provides networking to the Virtual Machines.

Swift is the object storage device which provides service for storing and retrieving arbitrary data. It uses the native API and S3 compatible API. Horizon is the web interface for Openstack. Horizon, the Openstack dashboard is used to provide simple self service user interface for end users.

4 Setting up the Private Cloud using Openstack

To deploy a minimal cloud infrastructure, at least two dedicated systems are needed: a front end, one or more node(s) with minimum 2GB memory. In our Institution, we have set up the private cloud using a cloud controller with three compute nodes. The scalability of the cloud can be extended through adding any number of compute nodes with the cloud controller. Fig. 3. gives the implementation diagram of the cloud which we have implemented. Table 1 describes the hardware details of the private cloud setup we implemented.

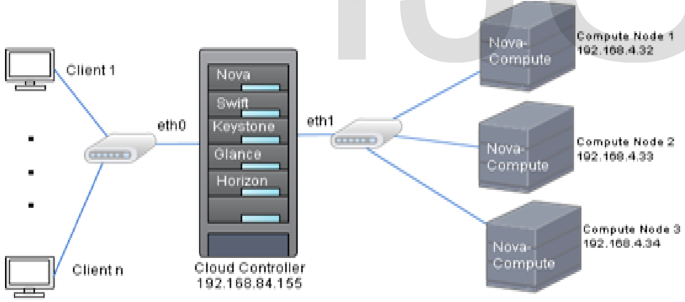


Fig. 3. Implementation diagram of cloud controller and compute node of OpenStack

Table 1. Hardware used to setup the Cloud

Hard ware	**Cloud Controller** (Server 1)	**Compute Node 1** (S2)	**Compute Node 2** (S3)	**Compute Node 3** (S4)	**Client**
CPU	VT Enabled 64 bit Dual Core	VT Enabled 64 bit Dual Core	VT Enabled 64 bit Dual Core	VT Enabled 64 bit Dual Core	VT Enabled 64 bit Dual core
RAM	4 GB	16 GB	4 GB	4 GB	1 GB
Hard disk Size	320 GB	500 GB	320 GB	320 GB	320 GB
Network	1 Gbps	1 Gbps	1 Gbps	1 Gbps	1 Mbps

4.1 Cloud Controller (Server1) Installation

1. In the cloud controller Ubuntu 12.04 version has been installed as the base operating system. Server1 contains all nova- services (nova-compute, nova-api, nova-volume, nova-scheduler, nova-network), Glance, Swift, Keystone and Horizon. Server1 has two network interface cards (NICs).
2. Set the network configurations (Public Ip and Private IP). Restart the network configuration.
3. Edit the file /etc/ntp.conf to make sure that the time on the server stays in sync with an external server. If the Internet connectivity is down, the NTP server uses its own hardware clock as the fallback. Restart the NTP server.
4. Openstack allows us to use MySQL, PostgreSQL or SQLite for Nova and Glance. We have chosen MySql and installed it. Using MySql we created user, databases for Nova, Glance, Keystone and gave the privileges to the users.
5. Installed keystone. Created tenants as admin and service. Created users as nova, glance, admin and swift. Created roles as admin and member. Adding Roles to Users in Tenants. Created Services and Endpoints.
6. Installed glance component by installing the packages glance-api, glance-client, glance-common, glance-registry and python-glance
7. Installed nova component by installing the packages nova-api, nova-cert, nova-compute, nova-compute-kvm, nova-doc, nova-network, nova-objectstore, nova-scheduler, nova-volume, rabbitmq-server, novnc and nova-consoleauth
8. Create nova-volume using the following commands. sudo pvcreate /dev/sda6. sudo vgcreate nova-volumes /dev/sda6.
9. Installed OpenStack Dashboard by installing the package openstack-dashboard.
10. Installed swift. The primary components of swift are the proxy, account, container and objectservers.

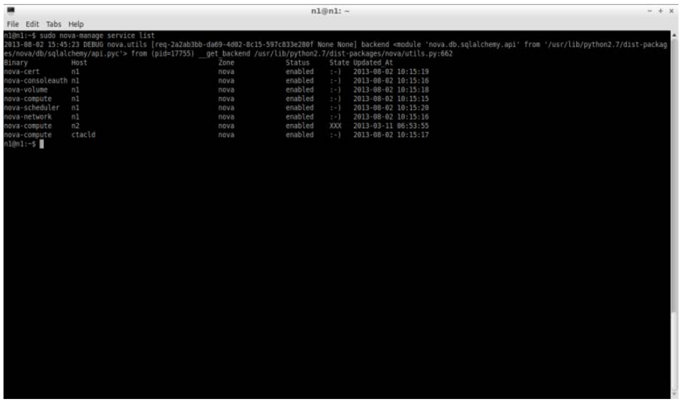
4.2 Compute Node Installation

1. In the compute nodes (S2, S3, S4) Ubuntu 12.04 version has been installed as the base operating system. This node contains only the nova- services including nova-compute, nova-api, nova-volume, nova-scheduler, nova-network. It contains two network interface cards (NICs).
2. Set the network configurations like public ip and private ip. Restart the network services.
3. Install the NTP package and sync this compute node with the cloud controller by editing the ntp.conf file. Then restart the ntp client.

4. Install the nova-compute and restart it.

Once Server1 has been installed, the "nova-manage service list" command lists all the nova services. Once compute nodes are added up, it shows the compute node details also. By issuing the following command, we could detect the compute nodes available in the cloud.

```
sudo nova-manage service list
```



n1@n1: ~

File Edit Tabs Help

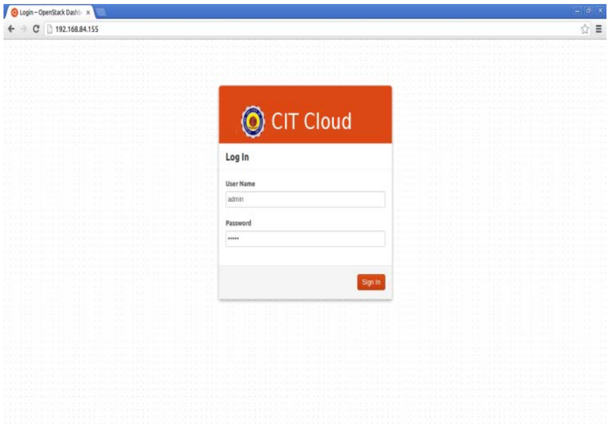
```
n1@n1:~$ sudo nova-manage service list
2013-08-02 15:45:23 DEBUG nova.utils [req-2a2ab3bb-da69-4d02-8c15-597c833e280f None None] backend <module 'nova.db.sqlalchemy.api' from '/usr/lib/python2.7/dist-packag
es/nova/db/sqlalchemy/api.pyc'> from (pid=17755) __get_backend /usr/lib/python2.7/dist-packages/nova/utils.py:662
Binary           Host                                 Zone             Status     State Updated_At
nova-cert        n1                                   nova             enabled    :-)   2013-08-02 10:15:19
nova-consoleauth n1                                   nova             enabled    :-)   2013-08-02 10:15:16
nova-volume      n1                                   nova             enabled    :-)   2013-08-02 10:15:18
nova-compute     n1                                   nova             enabled    :-)   2013-08-02 10:15:15
nova-scheduler   n1                                   nova             enabled    :-)   2013-08-02 10:15:20
nova-network     n1                                   nova             enabled    :-)   2013-08-02 10:15:16
nova-compute     n2                                   nova             enabled    XXX   2013-03-11 06:53:55
nova-compute     ctacld                               nova             enabled    :-)   2013-08-02 10:15:17
n1@n1:~$
```

Fig. 4. List of all nova services

5 Leveraging the Services of Openstack

Using the private cloud setup, we have provided database as a service (Oracle), web service as service (Xamp server), Compiler as a service (C compiler), Platform as a service (Ubuntu OS). All these software can be accessed by the authorized person from anywhere within the block using internal network. This eliminates the need for installing the software in each machine or in each lab and cut down the money spends for licensing of software.

This section describes about utilizing the Openstack by creating virtual machine and access it using the Horizon web interface.



Login - OpenStack Dash

192.168.84.155

CIT Cloud

Log In

User Name

admin

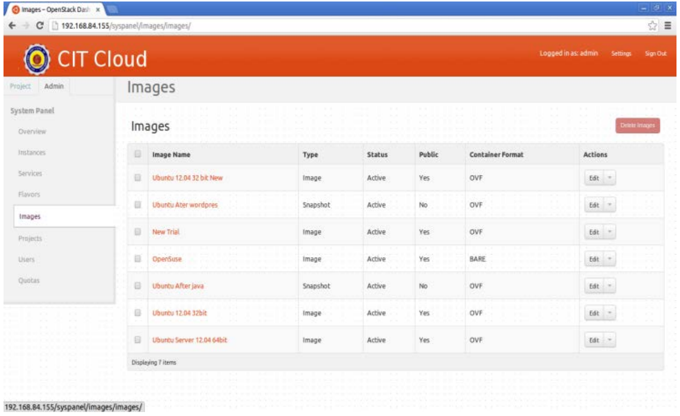
Password

•••••

Sign In

Fig. 5. Log in of OpenStack

Fig. 5. shows the dashboard Horizon. Horizon is the default user interface for Openstack. It asks for authentication. Authenticcation deals with the component Keystone. Keystone validates the user details using the database and allows the user to log in after validated.



Images - OpenStack Dash

192.168.84.155/syspanel/images/images/

CIT Cloud

Logged in as: admin Settings Sign Out

Project Admin

System Panel

Overview

Instances

Services

Flavors

Images

Projects

Users

Quotas

Images

Images

Delete Images

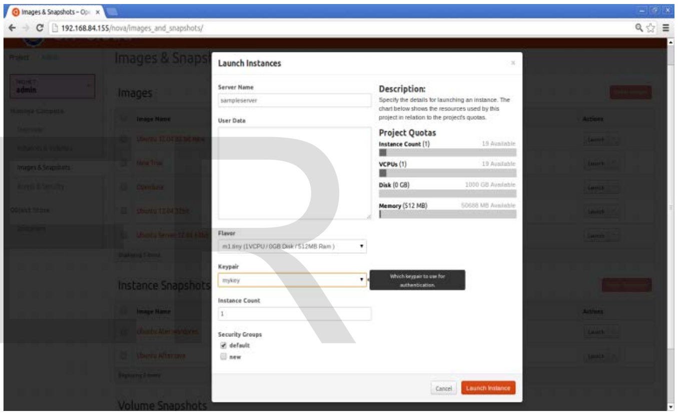
□	Image Name	Type	Status	Public	Container Format	Actions
□	Ubuntu 12.04 32 bit New	Image	Active	Yes	OVF	Edit
□	Ubuntu Ater wordpres	Snapshot	Active	No	OVF	Edit
□	New Trial	Image	Active	Yes	OVF	Edit
□	OpenSuse	Image	Active	Yes	BARE	Edit
□	Ubuntu After java	Snapshot	Active	No	OVF	Edit
□	Ubuntu 12.04 32bit	Image	Active	Yes	OVF	Edit
□	Ubuntu Server 12.04 64bit	Image	Active	Yes	OVF	Edit

Displaying 7 items

192.168.84.155/syspanel/images/images/

Fig. 6. Images in OpenStack

Fig. 6 shows the list of images registered in Openstack cloud. Image registration is done using the component glance.



Images & Snapshots - Op

192.168.84.155/nova/images_and_snapshots/

Launch Instances

Server Name

sampleserver

User Data

Description:

Specify the details for launching an instance. The chart below shows the resources used by this project in relation to the project's quotas.

Project Quotas

Instance Count (1) 19 Available

VCPUs (1) 19 Available

Disk (0 GB) 1000 GB Available

Memory (512 MB) 50688 MB Available

Flavor

m1.tiny (1VCPU / 0GB Disk / 512MB Ram)

Keypair

mykey

Which keypair to use for authentication.

Instance Count

1

Security Groups

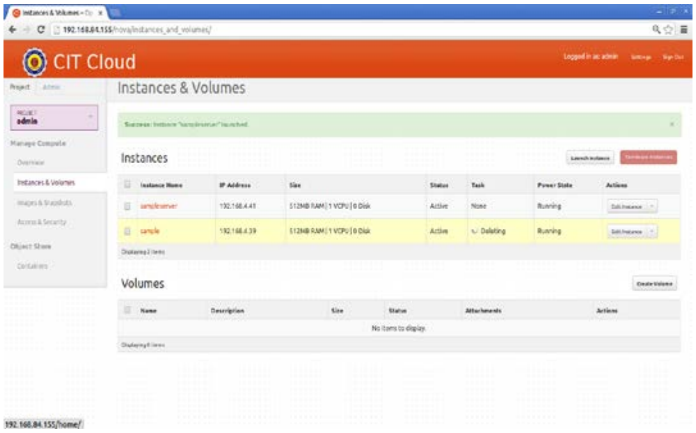
☑ default

□ new

Cancel Launch Instance

Fig. 7. Infrastucture allocated to Virtual Machine

Fig. 7. illustrates infrastucture like vcpu, memory, hard disk alloted to a virtual machine. There are some predefined flavors available in openstack, yet it allows the user to create their custom falvors. Before creating instances, keypair should be generated which contains public key and private key. The private key is stored in the local directory. These key are injected while the instace is being launched.



192.168.84.155/nova/instances_and_volumes/

CIT Cloud

Instances & Volumes

PROJECT admin

Instances

□	Instance Name	IP Address	Size	Status	Task	Power State	Actions
□	sampleserver	192.168.4.41	512MB RAM | 1 VCPU | 0 Disk	Active	None	Running	Edit Instance
□	sample	192.168.4.39	512MB RAM | 1 VCPU | 0 Disk	Active	Deleting	Running	Edit Instance

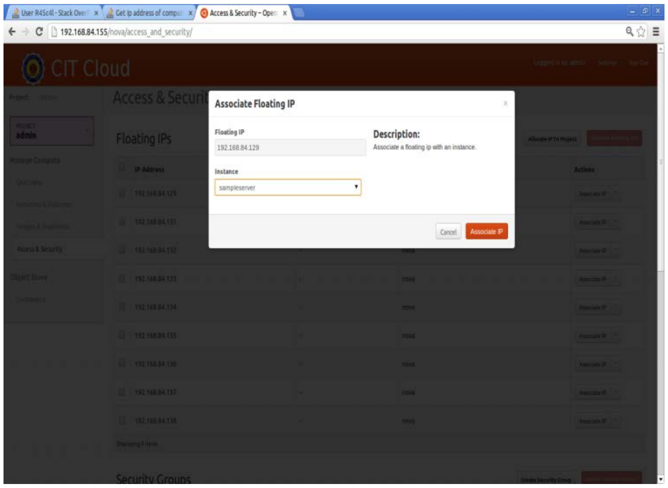
Volumes

□	Name	Description	Size	Status	Attachments	Actions
No items to display.						

192.168.84.155/home/

Fig. 8.Instances available in the cloud environment

Figure 8 depicts the list of created instances available in the Openstack cloud environment.



Associate Floating IP

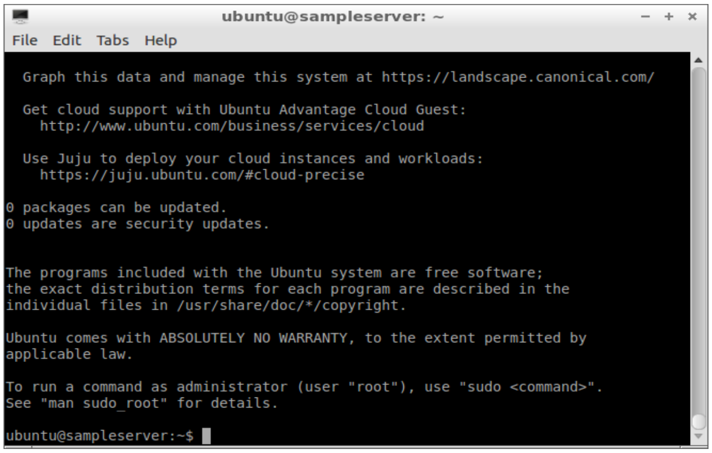
Floating IP: 192.168.84.129

Instance: sampleserver

Description:
Associate a floating ip with an instance.

Fig. 9. Assigning floating IP to Virtual Machine

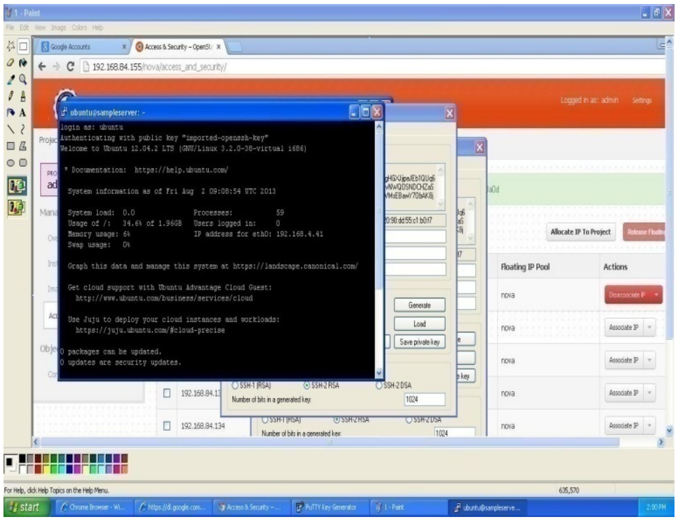
Every instance is automatically assigned a private IP address. The user may optionally assign public IP addresses to instances. OpenStack uses the term "floating IP" to refer to an IP address (typically public) that can be dynamically added to a running instance. OpenStack Compute uses Network Address Translation (NAT) to assign floating IPs to virtual instances.



```
ubuntu@sampleserver: ~

 Graph this data and manage this system at https://landscape.canonical.com/

 Get cloud support with Ubuntu Advantage Cloud Guest:
   http://www.ubuntu.com/business/services/cloud

 Use Juju to deploy your cloud instances and workloads:
   https://juju.ubuntu.com/#cloud-precise

0 packages can be updated.
0 updates are security updates.


The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@sampleserver:~$
```

Fig. 10. Virtual Machine accessed from Linux

Fig. 10. depicts accessing the virtual machine from the ubuntu platform using the command $ssh –i privatekey username@ipaddress.



```
ubuntu@sampleserver: ~
login as: ubuntu
Authenticating with public key "imported-openssh-key"
Welcome to Ubuntu 12.04.2 LTS (GNU/Linux 3.2.0-38-virtual i686)

 * Documentation:  https://help.ubuntu.com/

 System information as of Fri Aug  2 09:08:54 UTC 2013

 System load:  0.0               Processes:           59
 Usage of /:   34.6% of 1.96GB   Users logged in:     0
 Memory usage: 6%                IP address for eth0: 192.168.4.41
 Swap usage:   0%

 Graph this data and manage this system at https://landscape.canonical.com/

 Get cloud support with Ubuntu Advantage Cloud Guest:
   http://www.ubuntu.com/business/services/cloud

 Use Juju to deploy your cloud instances and workloads:
   https://juju.ubuntu.com/#cloud-precise

0 packages can be updated.
0 updates are security updates.
```

Fig. 11. Accessing the virtual machine in the windows environment

Fig. 11. depicts accessing the virtual machine from the windows platform using the putty. The private key should be converted from .ppk to .pem using the puttygen. Then using the putty by specifying the ip address of the virtual machine with auth as the .pem file the virtual machine can be accessed.

6 Conclusion

In the current economic climate where the expectations of efficiencies and cost savings are growing from IT organizations, private clouds provide a good opportunity to get started with cloud computing and reap the associated benefits of agility, cost savings and on-demand services while meeting the stringent enterprise security, performance and reliability requirements. Private clouds can be used to emulate a public cloud environment and can be used as a development and test platform while developing the applications to be deployed on the particular public cloud. Further, the same environment can also be used to test the developed applications for functionality as well as validations before it moves to the production at a public cloud.

The basic requirement for setting up cloud environment is not at all expensive. Cloud computing has the potential to revolutionize the computing facilities institutions provided to their students in a cost effective way. There are different open source technologies which are available to build private cloud within college, school campuses. This Private Cloud can offer students with desired computational facilities on demand without any expense. Now a day's most of the institutes are connected through high speed internet with each other. Thus centralized cloud infrastructure can satisfy all infrastructures related needs of those institutes on demand. It will create a common platform to be shared among faculties; students belonging to different institutions in reusable fashion.

REFERENCES

[1] Zoran Pantic and Muhammad Ali Babar, “Guidelines for building a Private Cloud Infrastructure”, Tech Report TR-2012-153, IT University of Copenhagen, 2012.

[2] Amazon Web Services(AWS), http://aws.amazon.com/ec2

[3] Google AppEngine, http://googcloudlabs.appspot.com

[4] Microsoft Azure, www.windowsazure.com

[5] NIST Definition of cloud computing http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

[6] OpenStack. OpenStack Compute – Adminstration Manual 2012 [On line]. Available : http://docs.openstack.org

[7] Yamuna Devi.L, Aruna.P, Sudha Devi. D, Priya.N, “Security in Virtual Machine Live Migration for KVM”, International Conference on Process Automation, Control and Computing (PACC), 2011, IEEE Conference Proceedings.

- *Aruna P., PH-091-9629306467. The corresponding author is Assistant Professor at: Department of Computer Technology and Applications, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu, India.. E-mail: aruna2023@gmail.com ;*
- *Yamuna Devi L, Sudha Devi D, Priya N : The authors are Assistant Professors at: Department of Computer Technology and Applications, Coimbatore Institute of Technology, Coimbatore, Tamil Nadu, India. E-mail: yamunalogan@gmail.com, ; sudhadevi_cit@yahoo.com, npriya_99@yahoo.com*
- *Dr. S. Vasantha, Dr. K. Thilagavathy, Associate Professors, Dept. of Physics, Coimbatore Institute of Technolgy, Coimbatore. E-Mail: vasantha@cit.edu.in, thilagavathy@cit.edu.in*