

# Comparison of Open Source Cloud Platforms

Aleksandar Donevski<sup>1</sup>, Sasko Ristov<sup>2</sup> and Marjan Gusev<sup>3</sup>

**Abstract** – Cloud computing tends to be the next generation in ICT hosting platform. Customers can choose between many commercial and open source cloud platforms that exist on the market today. We installed and observed all of the possible features/configurations of one of the most common open source cloud platforms on the market today, i.e. the OpenStack, Eucalyptus, and OpenNebula. In this paper we present a methodology and evaluation of their capabilities.

**Keywords** – Cloud Computing, Open Source, Virtualization

## I. INTRODUCTION

In this paper we focus on the open source IaaS cloud platforms, i.e. the OpenStack [1], Eucalyptus [2], and OpenNebula [3] cloud platforms. These cloud platforms are made mainly for building private and hybrid clouds. Also, all of them are compatible with most Amazon Web Services (AWS) [4] and are made to be deployed on custom IT infrastructures.

We deployed the newest versions of Eucalyptus (v3.2.2), OpenStack (release:Grizzly), and OpenNebula (v3.8.3) cloud platforms on three, one, and one physical server, respectively. All of the five physical servers are installed with the Ubuntu Server 12.04 LTS [5] (kernel version: 3.2.0-23) operating system and KVM [6] hypervisor for virtualization. All of the installed cloud platforms are observed in details from features to all different possible configurations (custom and generic) and the opportunities for interactions with other systems.

The paper is organized in the following sections. Section 2 briefly describes the open source IaaS cloud platforms, i.e. their most common features, while the next 3 sections briefly describes the software architecture, their components, features and latest improvements of the Eucalyptus, OpenStack, and OpenNebula cloud platforms. The comparison of the cloud platforms is performed in Section 6. And finally, the Section 7 presents the conclusion and the future work.

## II. OPEN SOURCE IAAS CLOUD PLATFORMS

In this section we briefly describe the most common features of most of the open source IaaS cloud platforms that are available on the market today. Although there are many open source IaaS cloud platforms on the market today, all of

them offer same features such as:

*Custom Deployment and Allocation of Virtualized Hardware Resources, Managed of Virtual LANs, IP Addresses, Security Groups, Storage, Virtual Machines Images and Instances, Cloud Users, and Dashboard.*

## III. OPENSTACK CLOUD PLATFORM

The OpenStack Cloud is founded by Rackspace Hosting and NASA with the main goal of creating a massively scalable open source cloud and today this project is supported by more than 150 organizations.

In this section we briefly describe the architecture of the latest release “Grizzly” of the OpenStack Cloud, focusing on the OpenStack components and latest features.

### A. Components of the OpenStack Cloud Platform

Figure 1 depicts the architecture of the OpenStack Cloud. The OpenStack Cloud consists of the following components:

*OpenStack Compute* is the main core component of the OpenStack Cloud with the main purpose of managing of large numbers of virtual machine instances in the cloud. This component can interact with most of the common hypervisors on the market today, such as KVM, VMware [7], Xen [8], LXC [9], Hyper-V [10], UML [11] and etc. The main features of this component are: custom allocation of hardware resources on the available compute nodes in the cloud, management of virtual machine images and running virtual machine instances, VNC proxy, allocation of VLANs and IPs and etc.

*OpenStack Storage* is a component which consists of two types of storages such as:

*Block Storage* - this system manages the volumes (block devices) in the OpenStack Cloud. With the usage of this system, the cloud users can create volumes with custom size for their needs, attach and detach volumes on their running virtual machine instances in the cloud, and a feature for creation and deletion snapshots from volumes. This system offers a powerful mechanism for backup of users data.

*Object Storage* - this distributed system which is made to prevent failures (fault tolerance) is used for storing and fetching data such as virtual machines images, archives, backups, users data and etc. Also, this component replicates its stored data to other available nodes in the cloud.

*OpenStack Networking* is a component which is made to manage the networks and IP addresses over the datacenter.

The main features of this component are: elastic data scaling unlike RAID sizing, distributed database unlike central database which is sometimes a bottleneck, 3x data replication, data redundancy, creation of custom sized volumes, deletion of volumes, making snapshots of volumes. It is fully integrated with OpenStack Compute component.

<sup>1</sup>Aleksandar Donevski is with the University Sts Cyril and Methodius, Faculty of Computer Sciences and Engineering, Rugjer Boshkovikj, Skopje, Macedonia, aleksandar.donevski@outlook.com.

<sup>2</sup>Sasko Ristov is with the University Sts Cyril and Methodius, Faculty of Computer Sciences and Engineering, 16 Rugjer Boshkovikj, Skopje, Macedonia, sashko.ristov@finki.ukim.mk.

<sup>3</sup>Marjan Gusev is with the University Sts Cyril and Methodius, Faculty of Computer Sciences and Engineering, 16 Rugjer Boshkovikj, Skopje, Macedonia, marjan.gushev@finki.ukim.mk.

*OpenStack Dashboard* is a component which is made to manage and observe the whole cloud through a web page. This component is a high level abstraction of all components of the OpenStack Cloud which allows an option for limiting the use of hardware resources by administrators.

*OpenStack Shared Services* is a component which consists of two types of services such as:

*Image Service* is a service that allows manipulation (storing, fetching, and listing) with virtual machine images in the cloud. This service acts as a central catalog of virtual machine images in the cloud. It supports the following formats of virtual machine images: RAW, AMI, VHD, VDI, QCOW2, VMDK, and OVF.

*Identity Service* is a service which is a central point for authentication in the whole cloud. This component maps the existing users with the existing components that they can use in the OpenStack Cloud. Also, this component supports different forms of authentication such as token-based, AWS-style and standard (username-password) authentication. The users can get (programmatically/via Web) the list of components that they can use, while the administrators can integrate existing LDAP and configure the policies and permissions with the RBAC feature.

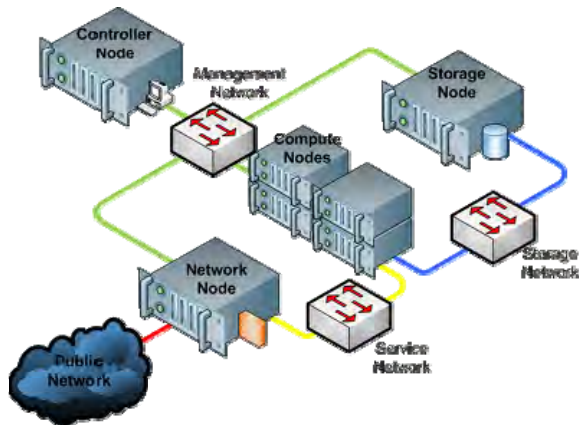


Figure 1. Architecture of the OpenStack Cloud Platform [18]

#### IV. EUCALYPTUS CLOUD PLATFORM

The Eucalyptus Cloud started as a research project at the University of California (Santa Barbara) in the department of Computer Science. Today, the Eucalyptus Cloud which is supported by Eucalyptus Systems is one of the best open source clouds that are available on the market today. NASA, Nist, US Department of Defense, Sony, Electronic Arts and many more world-wide organizations are customers of this cloud.

In this section we briefly describe the architecture of the latest release “3.2.2” of the Eucalyptus Cloud, focusing on the Eucalyptus components and latest features.

##### A. Components of the Eucalyptus Cloud Platform

Figure 2 depicts the architecture of the Eucalyptus Cloud. The Eucalyptus Cloud consists of six components (one optional component) which are separated in three logical layers:

*Cloud Layer* which consists of the Cloud Controller (CLC) and Walrus components.

*Cluster Layer* which consists of the Cluster Controller (CC), Storage Controller (SC) and VMware Broker components. The VMware Broker is an optional component.

*Node Layer* which consists of the Node Controller (NC) component.

*Cloud Controller* is a program written in Java, which acts as a demarcation point between the existing cloud and appropriate cloud users. This component offers a web-based administrative interface for management of the whole cloud, i.e. administrators, developers and users interacts with the cloud through this web interface. This component queries the existing available CCs in the cloud (the managers of the NCs) for available hardware resources, performs high-level scheduling operations and makes request to specific CCs in the cloud. This component manages the virtualized hardware resources and offers EC2-compatible SOAP and Query interfaces. There is no direct communication between the NCs and CLC in the cloud, i.e. the communication goes through the appropriate CCs.

*Walrus* is a program written in Java, which offers persistent storage to all of the virtual machines in the cloud. This component did not have data type restrictions, i.e. it can contain virtual machine images, cloud users data, snapshots of volumes, application data and etc. This component is equivalent to AWS S3 public service.

*Cluster Controller* is a program written in C, which manages the execution of virtual machine instances on specific NCs in its own cluster and networking of virtual machine instances. This component must have network connectivity to the CLC and appropriate NCs.

*Storage Controller* is a program written in Java, which offers block-accessed network storage and manages the virtual machines volumes and snapshots from volumes in the cloud. This component participates only in one cluster in the cloud. This component is equivalent to AWS EBS public service.

*Node Controller* is a program written in C, which hosts the virtual machine instances in the cloud. This component also manages the networks of virtual machine instances, allows caching of virtual machine images (download from Walrus) for better performances and interacts with the installed hypervisor and host OS.

*VMware Broker* is a component which allows interactions between the CC and existing ESX/ESXi hosts or vCenter server. This component converts existing EMIs (Eucalyptus Machine Images) to VMware virtual disks. There is no need for NCs, when we want VMware environment, because VMware Broker acts as a bridge between the CC and VMware hosts.

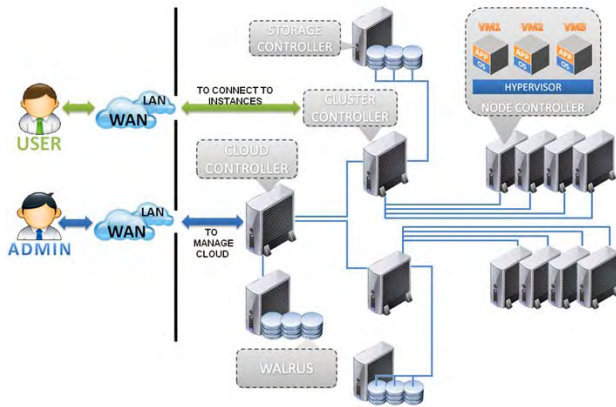


Figure 2. Architecture of the Eucalyptus Cloud Platform [19]

## V. OPENNEBULA CLOUD PLATFORM

The OpenNebula Cloud started as a research project back in 2005 and today is managed by C12G Labs. The OpenNebula Cloud platform is developed to be most-advanced, highly-scalable and IT infrastructure independent cloud solution for building and managing virtualized data centers and IaaS clouds.

In this section we briefly describe the architecture of the latest release “3.8.3” of the OpenNebula Cloud, focusing on the OpenNebula components and latest features.

### A. Components of the OpenNebula Cloud Platform

Figure 3 depicts the architecture of the OpenNebula Cloud. The OpenNebula Cloud consists of the following nine components:

*Front-end* is a component which runs all of the OpenNebula services.

*Host* is a component which runs on a server with installed hypervisor. This component interacts with the hypervisor, hosts virtual machine instances and manages the networks of the virtual machine instances in the cloud. This hosts must have network connectivity to the Front-end. The OpenNebula can work with heterogeneous environments, i.e. hosts in the cloud can have different operating systems or different hypervisors.

*Cluster* is a component which acts as a pool of hosts that share the same datastores and networks. With the forming of a clusters of hosts, performances such as load-balancing, high-availability and high-performance computing can be reached.

*Image Repository* is a component which acts as a repository of registered virtual machine images in the cloud. The OpenNebula Cloud can use multiple types of datastores for storing the virtual machine images and uses a dedicated datastore for the storing the virtual machine images of the running virtual machine instances.

*Sunstone* is a component which acts as a web-administrative interface for the OpenNebula Cloud. With the use of this administrative interface, the whole cloud can be managed and also observed. This interface also supports the RBAC principle.

*OCCI Service*, *Self-Service*, *EC2-Service* are components which enable management of the OpenNebula Cloud with different interfaces, i.e. management from OCCI Interface, end-user interface, EC2 interface, respectively.

*OCA* is a component which eases the communication with the management interface XML-RPC.

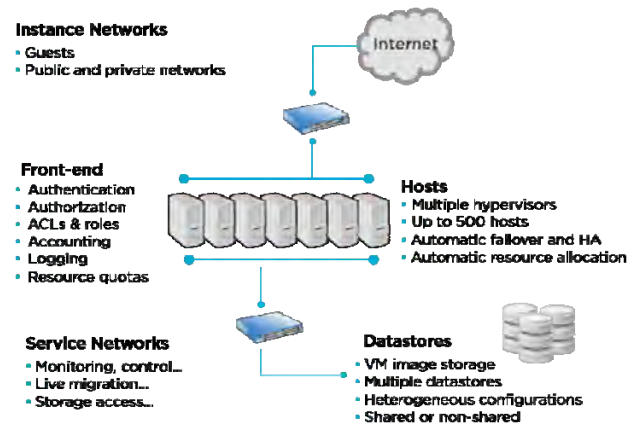


Figure 3. Architecture of the OpenNebula Cloud Platform [20]

## VI. COMPARISON OF THE CLOUD PLATFORMS

In this section we briefly describe the results of the performed comparison of the observed cloud platforms. Table 1 clearly shows the differences between newest versions of the observed open source cloud platforms.

Supports	OpenStack	Eucalyptus	OpenNebula
Service Model	IaaS	IaaS	IaaS
Deployment Model	Public Hybrid Private	Private Hybrid	Private Hybrid
AWS Compatibility	EC2 S3	EC2 S3 EBS IAM AMI	EC2 EBS AMI
Deployment OS	Linux	Linux	Linux
Available Packages	RedHat [12] CentOS [13] Debian [15] OpenSUSE [16] Ubuntu	RedHat CentOS	RedHat CentOS Fedora [14] Ubuntu

<i>Hypervisor</i>	KVM Xen VMware Hyper-V LXC QEMU UML PowerVM Bare Metal	KVM Xen VMware	KVM Xen VMware
<i>Virtual Machines</i>	Linux Windows [17]	Linux Windows	Linux Windows
<i>Networking</i>	VLAN NO VLAN Public IP's Private IP's SDN IDS Load-balance Firewalls VPN	VLAN NO VLAN Public IP's Private IP's	VLAN NO VLAN Public IP's Private IP's Ebttables OVSwitch
<i>Web-Management</i>	Yes	Yes	Yes
<i>Ease of Use</i>	Easy	Easy	Easy

Table 1. Comparison between the Cloud Platforms

## VII. CONCLUSION AND FUTURE WORK

We performed an evaluation of the three most common open source cloud platforms that are in use today, i.e. the newest versions of the OpenStack, Eucalyptus, and OpenNebula cloud platforms.

From the results of the performed evaluation, we can conclude that OpenStack cloud platform offers much more features than other two cloud platforms. Also, the OpenStack cloud platform offers much more compatibility with external systems and services which make this cloud platform a better IaaS cloud platform than both OpenNebula and Eucalyptus. We can conclude that the Eucalyptus cloud platform offers less features compared to the other two cloud platforms.

We will continue our work in evaluating other popular open source cloud platforms with different methodologies.

## REFERENCES

- [1] OpenStack, "Openstack cloud software," Apr. 2013. [Online]. Available: <http://openstack.org>
- [2] Eucalyptus, "Eucalyptus cloud software," Apr. 2013. [Online]. Available: <http://www.eucalyptus.com/>
- [3] OpenNebula, "Opennebula cloud software," Apr. 2013. [Online]. Available: <http://opennebula.org>
- [4] Amazon, "Amazon web services," Mar. 2012. [Online]. Available: <http://aws.amazon.com/what-is-aws/>
- [5] Ubuntu, "Ubuntu linux-based operating system," Apr. 2013. [Online]. Available: <http://www.ubuntu.com/>
- [6] KVM, "Kernel Based Virtual Machine," Apr 2013. [Online]. Available: [http://www.linux-kvm.org/page/Main\\_Page](http://www.linux-kvm.org/page/Main_Page)
- [7] VMware, "VMware enterprise virtualization," Apr 2013. [Online]. Available: <http://www.vmware.com/>
- [8] Xen, "The Xen Project," Apr 2013. [Online]. Available: <http://www.xen.org/>
- [9] LXC, "Lightweight virtual system mechanism," Apr. 2013. [Online]. Available: <http://lxc.sourceforge.net/>
- [10] Microsoft, "Microsoft Hyper-V Server 2012," Apr 2013. [Online]. Available: <http://www.microsoft.com/en-us/server-cloud/hyper-v-server/default.aspx>
- [11] UML, "User-mode linux kernel," Apr. 2013. [Online]. Available: <http://user-mode-linux.sourceforge.net/>
- [12] RedHat, "RedHat linux-based operating system," Apr. 2013. [Online]. Available: <http://www.redhat.com/>
- [13] CentOS, "CentOS linux-based operating system," Apr. 2013. [Online]. Available: <http://www.centos.org/>
- [14] Fedora, "Fedora linux-based operating system," Apr. 2013. [Online]. Available: <http://fedoraproject.org/>
- [15] Debian, "Debian linux-based operating system," Apr. 2013. [Online]. Available: <http://www.debian.org/>
- [16] OpenSUSE, "OpenSUSE linux-based operating system," Apr. 2013. [Online]. Available: <http://www.opensuse.org/en/>
- [17] Microsoft, "Microsoft Windows operating system," Apr. 2013. [Online]. Available: <http://windows.microsoft.com/en-us/windows/home>
- [18] OpenStack Cloud Platform, Available: <http://docs.stackops.org/display/documentation/Install+and+Configure+a+Multinode+Architecture>
- [19] Eucalyptus Cloud Platform, Available: <http://yoyoclouds.files.wordpress.com/2011/12/eucalyptus-cloud-setup.png>
- [20] OpenNebula Cloud Platform, Available: <http://opennebula.org/documentation:rel4.0:intro>