Establishment of Internal Cloud in a Cluster-based Environment

M.Hemalatha #1, C.Agjelia lydia *2

#1 PG IN COMPUTER SCIENCE AND ENGINEERING
NPR COLLEGE OF ENGINEERING AND TECH
NATHAM DINDIGUL,INDIA
mshema90@gmail.com

*2PG IN COMPUTER SCIENCE AND ENGINEERING
NPR COLLEGE OF ENGINEERING AND TECH
NATHAM DINDIGUL,INDIA

agjelia@gmail.com

Abstract— **Cloud computing is a computing model, where resources such as computing strengthen, storage, network and software are abstracted and provided as services on the internet in a remotely accessible fashion. On-demand availability, ease of provisioning, dynamic and virtually infinite scalability are some of the key attributes of cloud computing. This work aims to set up a fully operational internal cloud in a cluster-based environment. The information on the various infrastructures, components and middleware technologies that help in deploying this cloud setup are discussed in this paper.**

The internal cloud is implemented with Ubuntu Server Edition strengthen through Eucalyptus. Various platforms and infrastructure are being provided to the users upon request. Thus a cloud model similar to Amazon Web Services is established which is functioning with in a small cluster by efficiently utilizing the available computing resources.

I.   *INTRODUCTION*

A new paradigm in information technology, the cloud computing is the catchword that describes the direction in which the information infrastructure seems to be moving. It is the ability to utilize scalable, distributed computing environments within the confines of the internet. Cloud computing is emerging as a model in support of "everything-as-a-service" (XaaS). As the name suggests, the physical resources are on a 'cloud' whose physical infrastructure is unknown to the users. Cloud computing is an intersection of already existing concepts such as distributed computing, virtualization, grid computing and enterprise IT management.

1. *Distributed Computing*: Refers to the use of distributed systems, which consists of multiple autonomous computers that communicate through a computer network in order to achieve a common goal and to solve computational problems

2. *Virtualization*: An abstraction of computing resources and it hides physical characteristics of a computing platform from users, instead showing another abstract computing platform. Virtualization technique provides a certain kind of virtual machine environment which is a complete simulation of underlying hardware.

3. *Grid Computing*: A combination of computer resources from multiple administrative domains for common goal. The main strategy is to use middleware that divide and apportion pieces of a program among several computers, sometimes up to many thousands.

4. *Enterprise IT management*: It refers to the technologies, strategies, methods and tools to capture, manage, store, preserve and deliver content and documents related to an organization and its processes.

Internet-based utility computing using clouds enables the concept of pay-as-you-use the cloud. As a result, capital expenditure is reduced to operational expenditure. The consumer just needs a PC and internet connection. The key driving forces behind cloud computing are the ubiquity of broadband and wireless networking, falling storage costs, and progressive improvements in Internet computing software.

The main objective of the work is to set up a internal

cloud in a cluster-based environment with the help of different open source platforms and virtualization technologies in order to maintain the computing resources in a cost effective and efficient manner. The project aims to efficiently provision and utilize the resources within a small cluster setup which can be extended to a campus wide cloud. It is also possible to integrate the internal cloud with existing public cloud infrastructures to make a hybrid cloud. The third party client tools used along with public cloud are made compatible with the internal cloud for easy user interaction.

In the next section we describe about internal cloud and its various implementations .Then, we describe about virtualization and introduce the Eucalyptus architecture and describe the overall design and implementation of UEC. Finally, we discuss related work and conclude with a discussion of our work and future scope.

II. INTERNAL CLOUD

A internal cloud is a cloud computing infrastructure created by an organization for its own internal use, rather than using someone else's infrastructure like Amazon EC2 . The main thing that sets a internal cloud apart from a commercially used public cloud is where the hardware is kept and how it is maintained.

A internal cloud is typically hosted on the company's own servers, within their own network infrastructure. The main advantage one has with a internally-managed cloud is direct control over every aspect of the cloud's implementation: the hardware, the networking, the operating system and other software used to create the cloud itself; the way security is implemented and the APIs are used. Another advantage of a internal cloud is that it can generally be built from reasonably current commodity hardware. The most stringent requirements, apart from disk space and memory, are processors that support virtualization -- e.g., the Intel VT-x or AMD's AMD-V extensions. A third advantage is locality. A cloud hosted in our own datacenter is far easier to move data into and out of than a cloud hosting elsewhere. A fourth advantage is security. If we are hosting our own cloud infrastructure on a internal LAN, with no connections to the outside world, it is easier to secure. One major disadvantage of a internal cloud is the work involved in creating and maintaining the cloud. It becomes more difficult if we want to securely access the cloud from outside our corporate LAN.

A virtual internal cloud is a portion of a public cloud infrastructure that has been set aside for use by an organization, and is only accessible via an IPSec virtual internal network connection. One example of VPC technology is Amazon's Virtual Internal Cloud. We can make

the most of our existing hardware and network infrastructure by building a internal cloud. Some of the common cloud technology examples are discussed below.

A. Eucalyptus

Eucalyptus – Elastic Utility Computing Architecture Linking Your Programs To Useful Systems - is an open source software infrastructure for implementing on-premise clouds on existing enterprise IT and service provider infrastructure. It enables hybrid cloud and internal cloud deployments for enterprise datacenters and requires no special purpose hardware or reconfiguration. Leveraging Linux and web service technologies that commonly exist in today's IT infrastructure, Eucalyptus allows customers quickly and easily to create computing clouds on premise that are tailored to their specific application needs. At the same time, Eucalyptus supports the popular AWS cloud interface allowing these on-premise clouds to interact with public clouds using a common programming interface. Along with managing virtual machines, the technology supports secure virtualization of the network as well as the storage infrastructure within the cloud environment. Eucalyptus is compatible with and packaged for multiple distributions of Linux including Ubuntu, RHEL, OpenSuse, Debian, Fedora, and CentOS and will work with a variety of hypervisors and virtualization technologies.

B. VMWare

VMware, the global leader in virtualization and cloud infrastructure, builds on virtualization to deliver cloud infrastructure and management solutions, cloud application platform solutions and end-user computing solutions that significantly reduce IT complexity. VMware vCloud Datacenter Services are a new breed of enterprise-class hybrid cloud services, offering agility coupled with the security, interoperability and control that enterprises demand. A internal cloud built on VMware technology enables an IT department to transform into an efficient, agile and user-friendly internal service provider. Deploying a internal cloud on the VMware platform provides a practical path to the highly scalable, high-performance public clouds being built by recognized service providers leveraging the VMware platform.

Different hypervisors:
1. ESX –mainline product; commercial license
2. ESXi –mainline product, free (not OSS); boot from flash cards supported
3. Server –free (not OSS), installs on Linux & Windows
4. Workstation/Player –virtualizaton on user PC's

C. *Microsoft Windows Azure*

It is Microsoft's platform as a service that provides a wide range of internet services that can be consumed from both on-premises environments and the Internet. Azure Services Platform is an application platform in the cloud that allows applications to be hosted and run at Microsoft datacenters. It provides a cloud operating system called Windows Azure that serves as a runtime for the applications and provides a set of services that allows development, management and hosting of applications off-premises. Windows Azure has three core components: Compute, Storage and Fabric. Compute provides computation environment with Web Role and Worker Role while Storage focuses on providing scalable storage (Blobs, Tables, Queue, Drives) for large scale needs. Windows Azure Fabric makes up the physical underpinnings of the Windows Azure platform as the network of interconnected nodes consisting of servers, high-speed connections, and switches. The Azure Services Platform can Framework applications compiled currently run .Net for the CLR, while supporting the ASP.NET application frame work and associated deployment methods to deploy the applications onto the cloud platform. It can also support PHP websites.

III. *VIRTUALIZATION*

Virtualization is a cornerstone design technique for all cloud architectures. In cloud computing it refers primarily to platform virtualization or the abstraction of physical IT resources from the people and applications using them. Virtualization allows servers, storage devices, and other hardware to be treated as a pool of resources rather than individual systems, so that these resources can be allocated on demand. In cloud computing, two techniques such as para virtualization, which allows a single server to be treated as multiple virtual servers, and clustering, which allows multiple servers to be treated as a single server are used. There are several virtual machine monitors (VMM) available which differ in various aspects like scope, motivation, and method of implementation. A few types of monitor software are:

1) Native hypervisors: These VMMs have an OS associated with them. A complete software-based implementation will need a scheduler, a memory management subsystem and an IO device model to be exported to the guest OS. Examples are VMWare ESX server, Xen, KVM, and IBM mainframes

2) *Containers*: In this type of virtualization, the guest OS

and the host OS share the same kernel. Different namespaces are allocated for different guests. For example, the process identifiers, file descriptors are virtualized; in the sense a PID obtained for a process in the guest OS will only be valid within that guest. Examples are OpenVZ, FreeVPS, and Linux-Vserver.

3) Emulation: Each instruction in the guest is emulated. It is possible to run code compiled for different architectures on a computer, like running ARM code on a StrengthenPC machine. Examples are qemu, pearpc etc.

IV. *EUCALYPTUS ARCHITECTURE*

Amazon Web Services (AWS) have two popular services - Elastic Compute Cloud (EC2) and Simple Storage Service (S3). These services are available through web services interfaces. The client tools can use EC2 and S3 APIs to communicate with these services. Eucalyptus is a software available under GPL that helps in creating and managing a internal or even a publicly accessible cloud. It provides an EC2-compatible cloud computing platform and S3-compatible cloud storage platform. Eucalyptus has become very popular and is seen as one of the key open source cloud platforms. Since Eucalyptus makes its services available through EC2/S3 compatible APIs, the client tools written for AWS can be used with Eucalyptus as well. Currently VM's that run on top of Xen and KVM/QEMU hypervisor are supported by Eucalyptus.Fig.1 reveals the similar services of AWS and Eucalyptus.
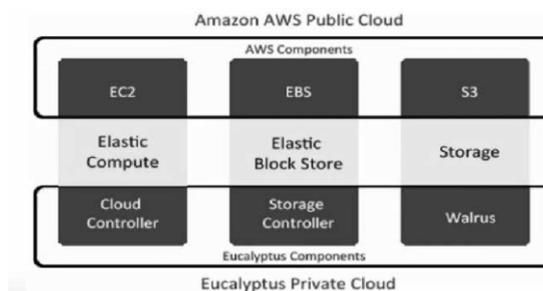
Fig. 1 Services from AWS and Eucalyptus

Each high-level system in Eucalyptus is implemented as a stand-alone web service. It employs a hierarchical design to reflect underlying resource topologies. The high level components of Eucalyptus, each with its own web-service Interfaces are described below.

A) Node Controller

The Node Controller (NC) is the component that executes on the physical resources that host VM instances and is responsible for instance start up, inspection, shutdown, and cleanup. The main functions of NC can be described as

- Collection of data related to the resource availability and utilization on the node and reporting the data to CC
- Instance life cycle management

B) Cluster Controller

A collection of NCs that logically belongs together report to a single Cluster Controller (CC) that typically executes on a cluster head node or server that has access to both internal and public networks. The CC is responsible for gathering state information from its collection of NCs, scheduling incoming VM instance execution requests to individual NCs, and managing the configuration of public and internal instance networks. The major functions of CC can be summarized as

- To receive requests from CLC to deploy instances
- To decide which NCs to use for deploying the instances on
- To control the virtual network available to the instances
- To collect information about the NCs registered with

and report it to the CLC

C) Cloud Controller

The Cloud Controller (CLC) is the front end to the entire cloud infrastructure. CLC provides an EC2/S3 compliant web services interface to the client tools on one side and interacts with the rest of the components of the Eucalyptus on the other side. The CLC is responsible for processing incoming user-initiated or administrative requests, making high-level VM instance scheduling decisions, processing service-level agreements (SLAs) and maintaining persistent system and user metadata.CLC also provides a web interface to users for managing certain aspects of the Eucalyptus infrastructure. Some of the duties of CLC are

- Monitor the availability of resources on various components of the cloud infrastructure, including hypervisor nodes that are used to actually provision the instances and the cluster controllers that manage the hypervisor nodes
- Resource arbitration – Deciding which clusters will be used for provisioning the instances
- Monitoring the running instances

D) Walrus Storage Controller

Walrus Storage Controller (WS3) provides a persistent simple storage service using REST and SOAP APIs compatible with S3 APIs.WS3 should be considered as a simple file storage system. Some of its functions are

- Storing the machine images
- Storing snapshots
- Storing and serving files using S3 API

E) Storage Controller

Storage Controller (SC) provides persistent block storage for use by the instances. This is similar to the Elastic Block Storage (EBS) service from AWS. The functions of SC include

- Creation of persistent EBS devices
- Providing the block storage over AoE or iSCSI protocol to the instances
  Allowing creation of snapshots of volumes

F) Administrative Interface

A cloud infrastructure must also support administrative tasks, such as adding and removing users and disk images. Eucalyptus supports such tasks though a Web-based interface, implemented by the cloud controller, and command-line tools. Using a Web browser, users can sign up for cloud access, download the cryptographic credentials needed for the programmatic interface, and query the system, e.g., about available disk images. The administrators can, additionally, manage user accounts, inspect the availability of system components. Users are added to a Eucalyptus installation either through the action of an administrator or by filling out an online form that is sent to the administrator for approval. Once added, a user account can be temporarily disabled or permanently removed by an administrator. At any point, the administrator can find out which instances a user is executing and terminate them. Currently, disk images in Eucalyptus can be added to the system only by an administrator. An image consists of a KVM-compatible guest OS kernel, a root file system image, and, optionally, a RAM disk image. Adding an image constitutes uploading these three components into the system and naming the image. After a image is added, any user can run instances of that image. Finally, the administrator is in charge of adding and removing nodes

from cluster controller's configuration. The Management Platform also exports various interfaces for the administrator, project manager, developer, and other users, with customizable levels of access and privileges.

G) Dependent Tools

Eucalyptus requires several supporting tools and software for managing and interacting with the internal cloud. Several open source virtualization tools can be integrated with Eucalyptus which enables the creation of virtual machines. Some of the related modules are discussed below.

1) KVM

Kernel-based Virtual Machine (KVM) is a Linux kernel virtualization infrastructure. It supports a wide variety of guest operating-systems to work with

2) Euca2ools

Euca2ools are command-line tools for interacting with Web services that export a REST/Query-based API compatible with Amazon EC2 and S3 services. The tools can be used with both Amazon's services and with installations of the Eucalyptus open-source cloud-computing infrastructure. The summary of the tasks that can be accomplished with Euca2ools are:

- Query of availability zones (i.e. clusters in Eucalyptus)
- SSH key management (add, list, delete)
- VM control (start, list, stop, reboot)
- Security group management (firewall configuration)
- Volume and snapshot management (attach, list, detach, create, bundle, delete)
- Image management (bundle, upload, register, list, deregister)
- IP address management (allocate, associate, list, release)
- Console output retrieval

3) Hybridfox

Hybridfox tool is a modified or extended elasticfox that enables us to switch seamless between different cloud clusters in order to manage the overall cloud computing environment. Hybridfox can perform all the functions that can be done by elasticfox, on the Eucalyptus Computing environment like

- Manage Images
- Raise and Stop Instances
- Manage Instances
- Manage Elastic IPs

- Manage Security Groups
- Manage Key pairs
- Manage Elastic Block Storage

V. RELATED WORK

This work for implementation of internal cloud is achieved using Ubuntu Enterprise Cloud (UEC) UEC includes Eucalyptus along with a number of other open source software. UEC makes the installation and configuration of the cloud easier. The basic architecture of UEC consists of

- A front end which runs
  · Cloud Controller (CLC)
  · Cluster Controller (CC)
  · Walrus (WS3)
  · Storage Controller (SC)
- one or more

A CLC manages the whole cloud and includes multiple CC's. There will be a WS3 attached to a CLC.A CC can contain multiple NC's and SC's. Ultimately the VM's will be running in the NC making use of its physical resources
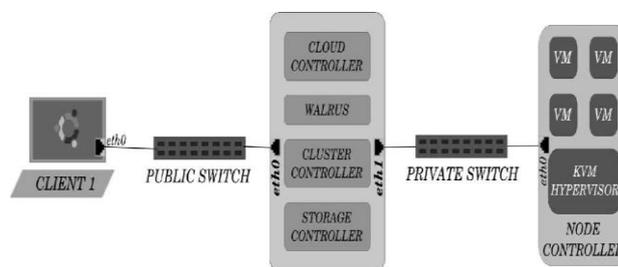
Fig. 2 UEC basic setup with three machines

The front end is set up in a machine with 1GHz CPU; 512MB RAM where as the nodes is running in processor with VT extensions and 1GB RAM. Both the front end and nodes have 5400 RPM IDE with 40GB disk space and connected to a network of 100 Mbps. A Eucalyptus Machine Image (EMI) is a combination of a virtual disk image, kernel and ramdisk images as well as an xml file containing metadata about the image. These images reside on WS3 and are used as templates for creating instances on UEC. Bundling an EMI is a multi-step process involving the following steps: creating a virtual disk image, installing the OS, installing required applications, making the OS ready to

run under UEC, registering the images with UEC and testing the image.

VI. *CONCLUSION AND FUTURE WORK*

This work focuses on harnessing the computing resources efficiently by making use of a internal cloud. The virtual machine images are available in the cloud and upon user request , its instances are created and run.

Eucalyptus can be integrated with Wavemaker , which is an open cloud development platform which is compatible with several working environments. It offers the advantage of visual development, one-click deployment and easy maintenance. After developing the application with Wavemaker, it can be deployed to Eucalyptus internal cloud. Thus Software-as-a-Service (SaaS) can also be provided with Eucalyptus which completes the full stack of cloud computing.

Eucalyptus was initially designed for the research purpose mainly. So it serves as a good framework for designing new scheduling policies and load balancing algorithms. The techniques for VM migration as well as replication can also be explored more.

REFERENCES

[1] Rajkumar Buyya,Chee Shin Yeo, Srikumar Venugopal, James Broberg, Ivona Brandic,, 2008, "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility", Elsevier Future Generation Computer Systems - Volume 25, Issue 6, June 2009, Pages 599-61

[2] Daniel Nurmi, Rich Wolski, Chris Grzegorczyk ,Graziano Obertelli, Sunil Soman, Lamia Youseff, Dmitrii Zagorodnov,2009, "The Eucalyptus Open-source Cloud-computing System", IEE E Computer Society

[3] Daniel Nurmi, Rich Wolski et al, 2009, "Eucalyptus: A Technical Report on an elastic utility computing Architecture", UCSB Computer Science Report

[4] Johnson D, Kiran Murari et al, 2010, "Eucalyptus Beginner's Guide-UEC Edition" CSS

[5] Sun Microsystems, 2009, "Introduction to Cloud Computing Architecture (White Paper)"

[6] Jinesh Varia, 2008," Cloud Architectures", In White Paper of Amazon jineshvaria.s3.amazonaws.com/public/cloudarchitectures -varia.pdf.

[7] Jeffrey Dean and Sanjay Ghemawat ,2008 , "MapReduce : Simplified Data Processing on Large Clusters"

[8] Eucalyptus Systems, Inc., 2009, "Eucalyptus Open-source Cloud Computing Infrastructure: An Overview", Eucalyptus Systems, Inc.

[9] "Setting up of an Open Source based Private Cloud" Dr.G.R.Karpagam1, J.Parkavi2 IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 1, May 2011