Implementing a Open Source Based Private cloud
Kalpana Deorukhkar
Computer Engg., Fr. CRCE, Mumbai, India
Raj Kansara, Tejal Narkar, Amit Wali
Computer Engg., Fr. CRCE, Bandra, India

Abstract— Cloud computing is an evolving paradigm in the computing environment, which empowers the users to share resources, computing power, software and other devices through a virtual machine. In today’s world technology is the fodder for the masses where cloud computing turns to be one of the frontrunners as the most decisive technology. This paper explains how to configure a private cloud using Eucalyptus .Eucalyptus is an open source software for building private and hybrid clouds. Being an Open Source Cloud service it provides access to a large pool of development resources, quicker updates and no vendor lock-in.

Keywords— Cloud Computing, Private Cloud

I. INTRODUCTION

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. On-demand availability, ease of provisioning, dynamic and virtually infinite scalability are some of the key attributes of cloud computing [3][6].

Cloud computing entrusts remote services with a user’s data, software and computation. The broad categories of services available on the cloud are: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). Software as a Service (SaaS) is defined as software that is deployed over the internet. With SaaS, a provider licenses an application to customers either as a service on demand. PaaS can be defined as a computing platform that allows the creation of web applications quickly and easily and without the complexity of buying and maintaining the software and infrastructure underneath it. Infrastructure as a Service (IaaS) is a way of delivering Cloud Computing infrastructure – servers, storage, network and operating systems – as an on-demand service. Rather than purchasing servers, software, data center space or network equipment, clients instead buy those resources as a fully outsourced service on demand [2]. This paper describes about configuring Cloud Using Eucalyptus, which allows users to provision their own collections of resources (hardware, storage, and network) using a self-service interface on an as-needed basis.

II. EUCALYPTUS

Eucalyptus is open source computer software for building Amazon Web Service (AWS)-compatible private and hybrid cloud computing environments marketed by the company Eucalyptus Systems. Eucalyptus empowers organizations to create self-service, elastic clouds inside their datacenter using existing IT infrastructure. It also enables a seamless path to hybrid cloud to deliver services both on-premises and through public clouds for greater business agility.

A. Amazon web Service(AWS) Compatibility

Fig. 1 AWS and Eucalyptus Compatibility
As shown in Fig. 1, Eucalyptus maintains high fidelity with AWS APIs, which means users can use or reuse many existing AWS-compatible tools, images (AMIs), and scripts to manage their own private cloud environment. AWS APIs are implemented on top of Eucalyptus, so tools in the cloud ecosystem that can communicate with AWS can use the same API with Eucalyptus. Eucalyptus provides compatibility with these AWS features: Amazon Elastic Compute Cloud (EC2), Amazon Elastic Block Storage (EBS), Amazon Machine Image (AMI), Amazon Simple Storage Service (S3), Amazon Identity and Access Management (IAM), Auto Scaling, Elastic Load Balancing and Amazon CloudWatch. The Eucalyptus components are open source and communicate with each other using well-defined web service definitions, with an additional communication layer that exposes the Amazon-compatible interface [4].

B. Eucalyptus and its Components

Fig. 2 Eucalyptus Fundamental Architecture

The major components of Eucalyptus are as shown in Fig. 2, described as:

- Cluster Controller (CC) Cluster Controller manages the one or more Node controller and responsible for deploying and managing instances on them. It communicates with Node Controller and Cloud Controller simultaneously. CC also manages the networking for the running instances under certain types of networking modes available in Eucalyptus.
- Cloud Controller (CLC) Cloud Controller is front end for the entire ecosystem. CLC provides an Amazon EC2/S3 compliant web services interface to the client tools on one side and interacts with the rest of the components of the Eucalyptus infrastructure on the other side.
- Node Controller (NC) It is the basic component for Nodes. Node controller maintains the life cycle of the instances running on each node. Node Controller interacts with the OS, hypervisor and the Cluster Controller simultaneously.
- Walrus Storage Controller (WS3) Walrus Storage Controller is a simple file storage system. WS3 stores the machine images and snapshots. It also stores and serves files using S3 APIs.
- Storage Controller (SC) allows the creation of snapshots of volumes. It provides persistent block storage over AoE or iSCSI to the instances [5].

C. Setting up Private Cloud

Private cloud is cloud infrastructure operated solely for a single organization, whether managed internally or by a third-party and hosted internally or externally [6].

The following sections describe how to install and configure a basic Ubuntu Enterprise Cloud (UEC) setup using three servers. Two Servers (Server1 and Server2) will run a 64-bit Lucid server version and the third server will run a Lucid Desktop 64-bit version (Client1) [3]. This section will also describe the steps used for creating virtual machine images and uploading those images onto the cloud which can then be accessed from Client1 through a Browser.
Fig. 3 UEC Basic Setup

1) Installation and Configuration

The Fig. 3 shows the basic setup of Ubuntu Enterprise Cloud using 2 Servers (Server1 and Server2) and 1 Client (Client1), although Client1 shown above need not be a dedicated machine.

```
<table>
<thead>
<tr>
<th>Hardware</th>
<th>Server1</th>
<th>Server2</th>
<th>Server3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>1 GHZ</td>
<td>1 GB</td>
<td>1 GB</td>
</tr>
<tr>
<td>Suggested</td>
<td>2x2GHZ</td>
<td>2GB</td>
<td>2GB</td>
</tr>
<tr>
<td>CPU</td>
<td>VT</td>
<td>VT, 64Bit, Multicore</td>
<td>VT, 64Bit, Multicore</td>
</tr>
<tr>
<td>Memory</td>
<td>1 GiB</td>
<td>4GiB</td>
<td>1 GB</td>
</tr>
<tr>
<td></td>
<td>2GiB</td>
<td></td>
<td>2GiB</td>
</tr>
<tr>
<td>Disk</td>
<td>5400rpm IDE</td>
<td>7200rpm SATA or SCSI</td>
<td>5400rpm IDE</td>
</tr>
<tr>
<td>Disk Space</td>
<td>40 GiB</td>
<td>100GiB</td>
<td>40 GiB</td>
</tr>
<tr>
<td>Networking</td>
<td>100Mbps</td>
<td>100Mbps</td>
<td>100Mbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>1 GHZ</td>
<td>1 GB</td>
<td>1 GB</td>
</tr>
<tr>
<td>Memory</td>
<td>2GiB</td>
<td>4GiB</td>
<td>1 GB</td>
</tr>
<tr>
<td>Disk</td>
<td>5400rpm IDE</td>
<td>7200rpm SATA or SCSI</td>
<td>5400rpm IDE</td>
</tr>
<tr>
<td>Disk Space</td>
<td>40 GiB</td>
<td>40 GiB</td>
<td>100GiB</td>
</tr>
<tr>
<td>Networking</td>
<td>100Mbps</td>
<td>100Mbps</td>
<td>100Mbps</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU</td>
<td>1 GHZ</td>
<td>1 GB</td>
<td>1 GB</td>
</tr>
<tr>
<td>Memory</td>
<td>2GiB</td>
<td>4GiB</td>
<td>1 GB</td>
</tr>
<tr>
<td>Disk</td>
<td>5400rpm IDE</td>
<td>7200rpm SATA or SCSI</td>
<td>5400rpm IDE</td>
</tr>
<tr>
<td>Disk Space</td>
<td>40 GiB</td>
<td>40 GiB</td>
<td>100GiB</td>
</tr>
<tr>
<td>Networking</td>
<td>100Mbps</td>
<td>100Mbps</td>
<td>100Mbps</td>
</tr>
</tbody>
</table>

2) Steps For Installation

2.1) Server1 Installation

- Boot the server off the Ubuntu Server 10.04 CD. At the graphical boot menu, select Install Ubuntu Enterprise Cloud and proceed with the basic installation steps.
- Installation only lets you set up the IP address details for one interface. Please do that for eth0.
- You will need to choose certain configuration options for your UEC, during the course of the install:
- Cloud Controller Address - Leave this blank as Server1 is the Cloud Controller in this setup.
- Cloud Installation Mode - Select “Cloud controller”, “Walrus storageservice”, “Cluster controller” and “Storage controller”.

© 2014, IJARCSSE All Rights Reserved
Network interface for communication with nodes - eth1
- Eucalyptus cluster name - myueccluster
- Eucalyptus IP range - 192.168.10.200-192.168.10.220

2.1.1) Post Installation Steps:
- Setup static ip for network interface eth1:
  - auto eth1
  - iface eth1 inet static
  - address 192.168.20.1
  - netmask 255.255.255.0
  - network 192.168.20.0
  - broadcast 192.168.20.255
- Restart the networking using the command:
- $ sudo /etc/init.d.networking restart
- Install the NTP package:
- $ sudo apt-get install ntp
- Open the file /etc/ntp.conf and add the following two lines to make sure the server serves time even when its connectivity to the internet is down.
  - $ server 127.127.1.0
  - $ fudge 127.127.1.0 stratum 10
- Save the changes and restart CC:
  - $ sudo start eucalyptus-cc CLEAN=1

2.2) Server 2 Installation
- Boot the server off the Ubuntu Server 10.04 CD. At the graphical boot menu, select 'Install Ubuntu Enterprise Cloud' and proceed with the basic installation steps.
  - Installation only lets you set up the IP address for one interface.
  - Please do that for eth0 by setting up the private IP - 192.168.20.2.
  - You will need to choose certain configuration options for your UEC, during the course of the install.
  - You can ignore all the settings, except the following:
    - Cloud Controller Address - 192.168.20.1
    - Cloud Installation Mode - Select 'Node Controller'
    - Gateway - 192.168.20.1 (IP of the CC)

2.2.1) Post Installation Steps:
- Add your DNS server details to /etc/resolv.conf
  - nameserver 192.168.10.2
  - nameserver 192.168.10.3
- Install the NTP package to synchronize with the NTP server
- Open the file /etc/ntp.conf and append the following line:
  - server 192.168.20.1
- Restart NTP service to make the changes effective
- Open the file /etc/eucalyptus/eucalyptus.conf and make the following changes:
  - VNET_PUBINTERFACE="br0"
  - VNET_PRIVINTERFACE="br0"
  - VNET_BRIDGE="br0"
  - VNET_DHCPDAEMON="/usr/sbin/dhcpd3"
  - VNET_DHCPUSER="dhcpd"
  - VNET_MODE="MANAGEDNOVLA"
- After making the above changes, run the following commands:
  - $ sudo restart eucalyptusncpublication
  - $ sudo restart eucalyptusnc

2.3) Install CC's ssh public key to NC
- On the Node Controller, temporarily set a password for the 'eucalyptus' user:
  - $ sudo passwd eucalyptus
- On the Cluster Controller:
  - $ sudo u eucalyptus sshcopyid -i eucalyptus/.ssh/id_rsa.pub eucalyptus@192.168.20.2
  - You can now remove the password of the 'eucalyptus' account on the Node:
  - $ sudo passwd d eucalyptus
2.4) Client1

Client1 Machine is used to interact with the cloud setup and to bundle and register Eucalyptus Machine Images (EMI).

Installation:
- Boot the Desktop off the Ubuntu Desktop 10.04 CD and install. The Desktop will be on the enterprise network and will obtain an IP address through DHCP.
- Install KVM to help us install images on KVM platform and bundle them:
  
  ```
  $ aptget install qemukvm
  ```

2.5) Post Install Configuration

Install euca2ools to be able to manage the cloud from it:

```
$ sudo aptget install euca2ools
```

D. Invoking the web Interface

- Login to the web interface of CLC by using the following link: https://192.168.10.121:8443 The default username is `admin` and the default password is `admin`
- Download the credentials archive from https://192.168.10.121:8443/# credentials and save it in the ~/.euca directory.
- Extract the credentials archive:
  ```
  $ cd .euca
  $ unzip mycreds.zip
  ```
- Source eucarc:
  ```
  $ . ~/.euca/eucarc
  ```
- Verify euca2ools communication with UEC:
  ```
  $ eucadescribe availability zones verbose
  ```

E. Running Instances

The web interface is accessible at https://192.168.10.121:8443/.

1) Installing Cloud Images

Although it’s possible to make your own custom images to run on your cloud, it’s far easier to get one from the UEC “store” shown in Fig. 4. Simply access the cloud controller at the URL https://<cloud-controller-ip-address>:8443/, enter your login and password, click the Store tab, and you’ll be presented with the UEC Store. [8]

![Fig.4 UEC Store](image)

2) Checking Images

The command “euca-describe-images” which is equivalent to clicking the “Images” tab in the Eucalyptus administrative web interface shows the emi-xxxxxx identifier for each image/bundle that will be used to run an instance.

```
$ euca-describe-images
```

3) Installing a Key Pair

A Keypair which allows us to access the instance through ssh must be introduced in the instance.

```
$ euca-add-keypair mykey > ~/.euca/mykey.priv
$ chmod 0600 ~/.euca/mykey.priv [3]
```

4) Running Instances

Now, you can fire up your first image using the following commands:
The first time you instantiate a particular image, it'll be slow to start. Eucalyptus caches the image on the node controller, so there's a sizable amount of data that's got to move to the node. You can keep tabs on the status of your image by running: `watch -n5 euca-describe-instances`

5) **HybridFox**

HybridFox is an addon for the Firefox web browser that provides a graphical interface to an EC2-compatible cloud. HybridFox has a wide featureset, allowing for full control of instances, images, keypairs, security groups, volumes and others that OpenStack does not yet support. The plugin is based on ElasticFox, which is in wide-ranging usage by users of the Amazon EC2 cloud. It is actively developed by Amazon staff, then modified by the Eucalyptus team[9]. HybridOx is shown in Fig. 5.

```
S euca-run-instances emi-DF841070 -k mykey -t c1.medium
```

HybridFox is shown in Fig. 5.

6) **Life Cycle of Instances**

As shown in Fig. 6, when you invoke `euca-run-instances` command (or when you choose to run an instance from Hybridfox/Elasticfox), here are a few things that happen on various components of UEC:

- Authentication/Authorization of the user request to ensure you have permission to launch the instance
- Identification of CC to take responsibility for deploying the instance and identification of the NC for running the instance.
- Downloading the image from WS3 to NC (images are cached so that starting multiple instances of the same machine image downloads that image only once)[3].

### III. CONCLUSIONS AND FUTURE SCOPE

Cloud computing is spreading through enterprises as it enables the agility sought by global organizations. The rapid adoption of cloud computing can be attributed to a latent need in both the end user organizations as well as the IT industry.[10] This Paper describes installation and Configuration of an OpenSource Private Cloud using technologies like Eucalyptus, euca2ools, KVM and HybridFox. The services can be provided to the user on launching instances as and when required. The existing implementation of the paper provides Infrastructure as a Service(IaaS) and Software as a Service(SaaS).

### REFERENCES