High Level Data Protection for Cloud

Nisarg Shah
M.E.Scholar
Hasmukh Goswami College of Engineering, GTU

Abstract - **Cloud computing is a computing paradigm shift where computing is moved away from personal computers or an individual application server to a "cloud" of computers. Users of the cloud only need to be concerned with the computing service being asked for, as the underlying details of how it is achieved are hidden. This method of distributed computing is done through pooling all computer resources together and being managed by software rather than a human. The services being requested of a cloud are not limited to using web applications, but can also be IT management tasks such as requesting of systems, a software stack or a specific web appliance.**

I. INTRODUCTION

1.1 Cloud computing models & services

The Cloud Computing[1] model has three service delivery models and main three deployment models models are:

- **Private cloud:** a cloud platform is dedicated for specific organization.
- **Public cloud:** available to public users to register and use the available infrastructure.
- **Hybrid cloud:** a private cloud that can extend to use resources in public clouds. Public cloud most vulnerable deployment model because for public users to host their services who may be malicious users.

Cloud service delivery models

Infrastructure-as-a-service (IaaS): provides virtual machines and other abstracted hardware and operating systems which may be controlled through a service API.

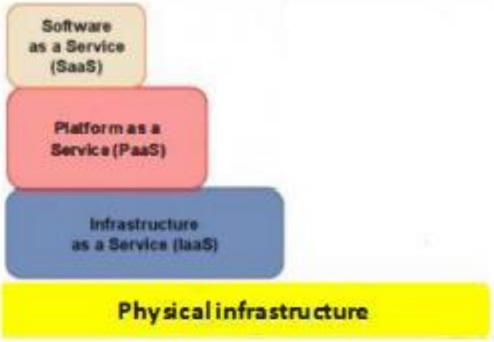

Software as a Service (SaaS)
Platform as a Service (PaaS)
Infrastructure as a Service (IaaS)
Physical infrastructure

Figure 1.1: cloud service delivery models [1]

Examples: include Amazon EC2 and S3, erremark Enterprise Cloud, Windows Live Skydrive and Rackspace Cloud.

Platform-as-a-service (PaaS): allows customers to develop new applications using APIs deployed and configurable remotely. The platforms offered include development tools, configuration management, and deployment platforms.

Examples: are Microsoft Azure, Force and Google App engine.

Software-as-a-service (SaaS): is software offered by a third party provider, available on demand, usually via the Internet configurable remotely.

Examples: include online word processing and spreadsheet tools, CRM services and web content delivery services (Salesforce CRM, Google Docs, etc).

1.2 Cloud computing architecture

Cloud has a four layered architecture [2] which identifies the system's fundamental components and specifies the function of these components as shown in figure 2. These four layers are:

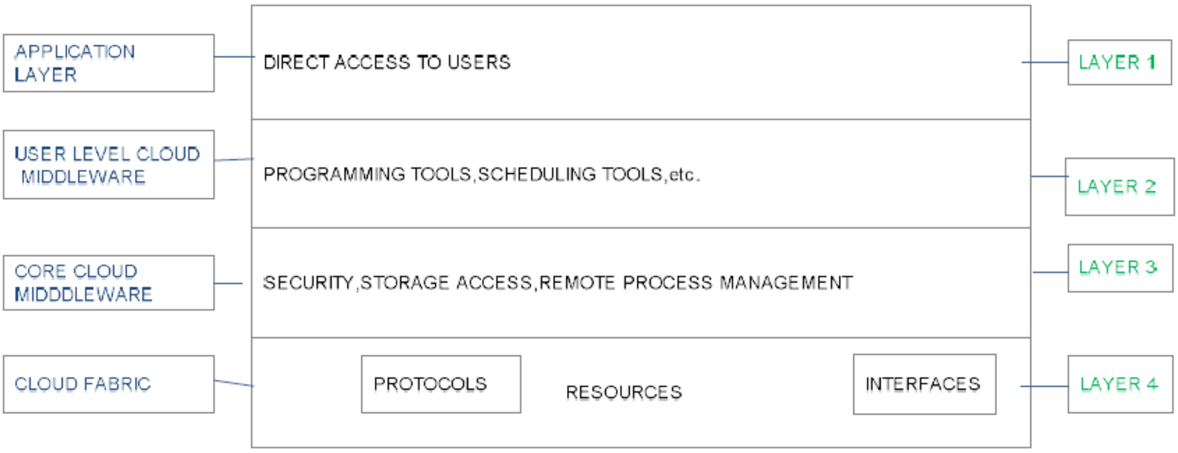


Figure 1.2: Cloud Architecture [2]

- **Cloud Fabric:** This layer comprises of all the resources accessible from the internet. It also includes the protocols and the interfaces accessing and managing these resources.
- **Core Cloud Middleware:** This layer offers the services like storage access, security, information registration and remote process management.
- **User Level Cloud Middleware**: Also named as Cloud tools, this layer involves programming tools, scheduling the task, managing the resources and developing the application environment.
- **Application Layer:** This is the highest layer of the architecture. Since it helps the cloud users to see and interact, it is also called as service ware.

II. LITERATURE SURVEY

Related to the cloud security there is lots of issue with their security and also with data storage. Here we look for how to protect data that store in to the un-trusted storage devices in to the cloud storage. For resolving the problem lots of the people has show their arrangement and analysis for their data protection. I have studied that arrangement one by one and try to understand the problem and related issue with cloud storage and data protection.

2.1 Security issue at different level [7]

The cloud service provider for cloud makes sure that the customer does not face any problem such as loss of data or data theft. There is also a possibility where a malicious user[5] can penetrate the cloud by impersonating a legitimate user, there by infecting the entire cloud. This leads to affects many customers who are sharing the infected cloud. There are five types of issues rise while discussing security of a cloud.

1. Data Issues
2. Privacy issues
3. Infected Application
4. Security issues
5. Trust Issues

2.2 Data protection for cloud using DFE and FHE[4]

Infrastructure-as-a-Service like Amazon Web Services provides virtual server instance API to start, stop, and access and configure their virtual servers and storage. In the enterprise, cloud computing allows a company to pay for only as much capacity as is needed, and bring more online as soon as required. Because this model resembles the way electricity, fuel and water are consumed; it's sometimes referred to as utility computing. Platform-as-a-service in the cloud is defined as a set of software and product development tools hosted on the provider's infrastructure.

- **Working model for Cloud storage**

Cloud customer can create applications on the provider's platform over the Internet. PaaS providers may use APIs, website portals or gateway software installed on the customer's computer. Force.com, (an outgrowth of Salesforce.com) and Google Apps are examples of PaaS. Developers need to know that currently, there are not standards for interoperability or data portability in the cloud. Some providers will not allow software created by their customers to be moved off the provider's platform.

In the software-as-a-service cloud model, the vendor supplies the hardware infrastructure, the software product and interacts with the user through a front-end portal. SaaS is a very broad market. Services can be anything from Web-based email and also database processing. Because the service provider hosts both the application and the data, the end user is free to use the service from anywhere.

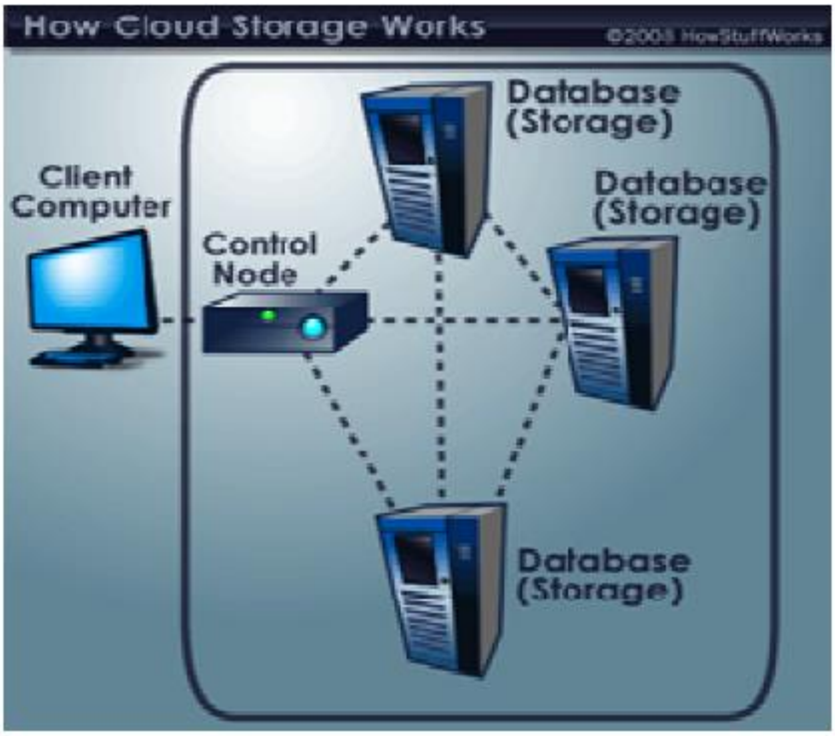


Figure 2.2: Working Model of Cloud Storage[4]

- **Full-disk encryption**

To protect the data at the hardware level there is mechanism provided called FDE. Which automatically convert the data on hard drive that cannot easily accessible by anyone. FDE can be installed on a computing device at the time of manufacturing or it can be added later on by installing a special software driver.

- **Requirement of FDE**

At the basic level of any cloud system to store valuable data in storage system. Mainly focus on the store e-mail data and digital picture. Some of the cloud system is using small operation and small system, where some organization use whole data warehouse. In cloud system data server is connected through the internet. The file is send from client is store to data server on cloud so customer can store data from anywhere and also access data from anywhere.

Even though there is using some complex algorithms use for encryption, But still they worry for their data protection and confidentiality. That’s why there is need to protect hard drive. Customer also worried that either storage device stolen or some third party can hacked that network then? That’s why there is need of FDE.

2.3 Cloud data protection for masses[5]

Umesh Shankar(google)[5] and his buddies who has define the difference between DFE and DHE with and enhance the data storage security with respect to data-protection as a service (DpaaS).

To challenge the issue of cloud computing they have define some issue and try to solve it. With the issue of cloud there is main problem with integrity, privacy, Access transparency, and rich computation. To give the answer of above problem there is define DpaaS (data protection as a service).

- **Architecture of DPAAS(data-protection as a service)**

Architecture of DPAAS is provide the security at the level of the integrity , access control as well it provide the security at the level of platform as a service. Fig.6 is totally illustration of the architecture.

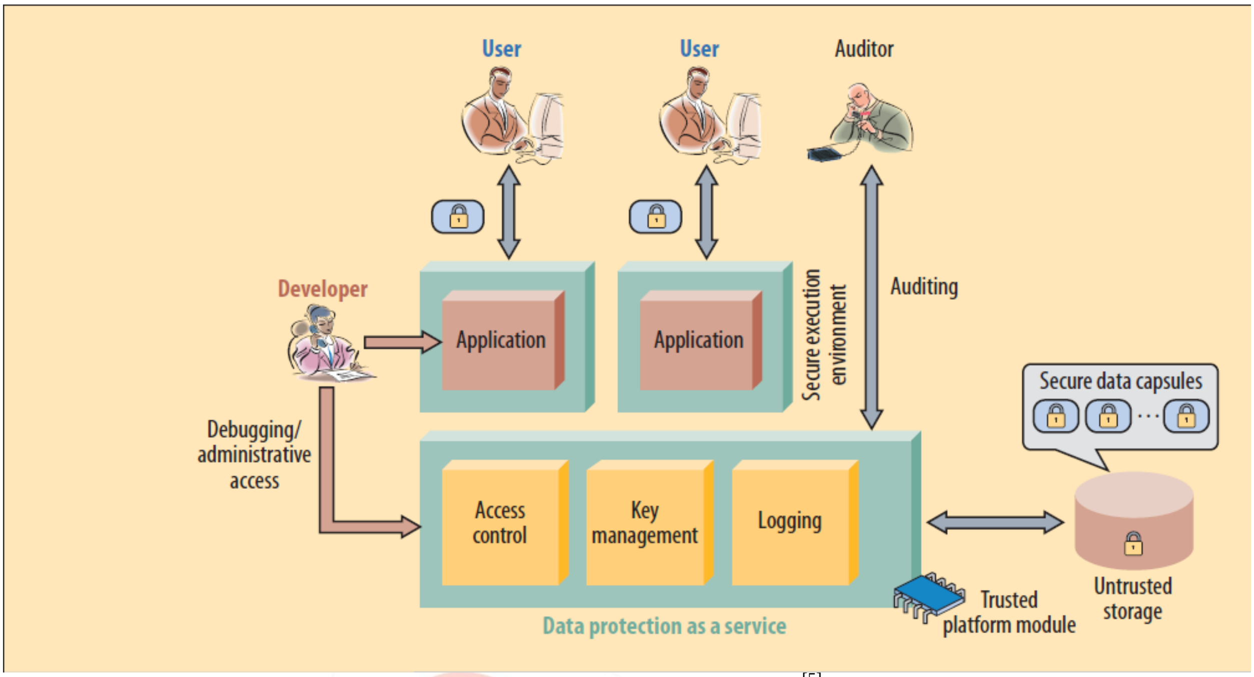


Figure. 2.3 Architecture of DPAAS[5].

Above secure structure of data protection as a service has consumer protection on their data , Because there is different customer has their own access control who can only use their own data as well own application installed on the cloud.

Developer who can go through the system by only administrative access and can manage the application or the key management. Without logging in unauthorized person cannot go through it. It is like as bottleneck structure that implements for provide security only. Also with different encryption algorithms and its key management this structure has iron-core steeled data.

As per the security issue there is all of the layers have been covered by this security mechanism but for the untrusted storage device can be stolen or that can be theft by physical unauthorized entry. This is hardly occur that the third person hack entire private network but in this case there is no backup plan for the cloud consumer.

2.4 High level security for scalable data[6]

High level security is provided by the Kirti Rao who has define by using FDE and FHE . By using both of the security mechanism here provide high level security, which is the most provide security to the hard drive.

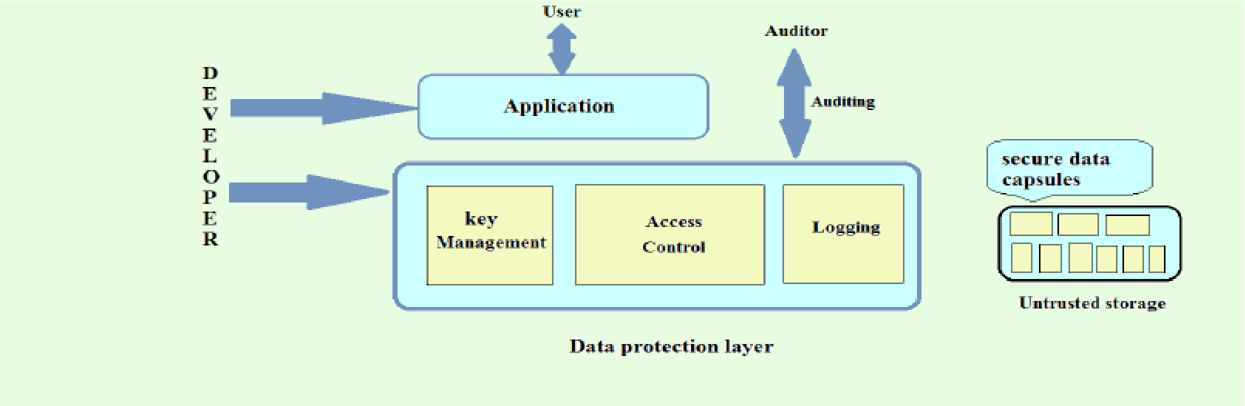


Figure 2.4 Architecture for security and privacy[6]

This mechanism is provide same security like FDE, but instead of giving key to whole disk, here branch out the whole disk in separate drive and then give different key to that separate drive and provide security.

Mechanism approach

Whole disk encryption indicates that every bit present in disk gets encrypted including the programs that encrypts the bootable operating system partitions.FileVault 2 encrypts the operating systems startup volume totally. Authorized users information is uploaded from an separate non encrypted boot volume. By using master boot record in systems,then that part of disk relics non encrypted. Based upon the full disk encryption some systems encrypts the total disk including the master boot record.

Creating a layer of security itself creates the data protection, hence this helps millions of cloud data users. By providing the protection to the data dramatically helps to reduce the per-application development effort. Encryption, logging, key management and access controls acts like barriers for secure storage of data. For the implementation of architecture offers evidence of privacy to the data owners, even in the presence of malicious applications.

III. RESEARCH WORK

3.1 Flow chart of algorithm

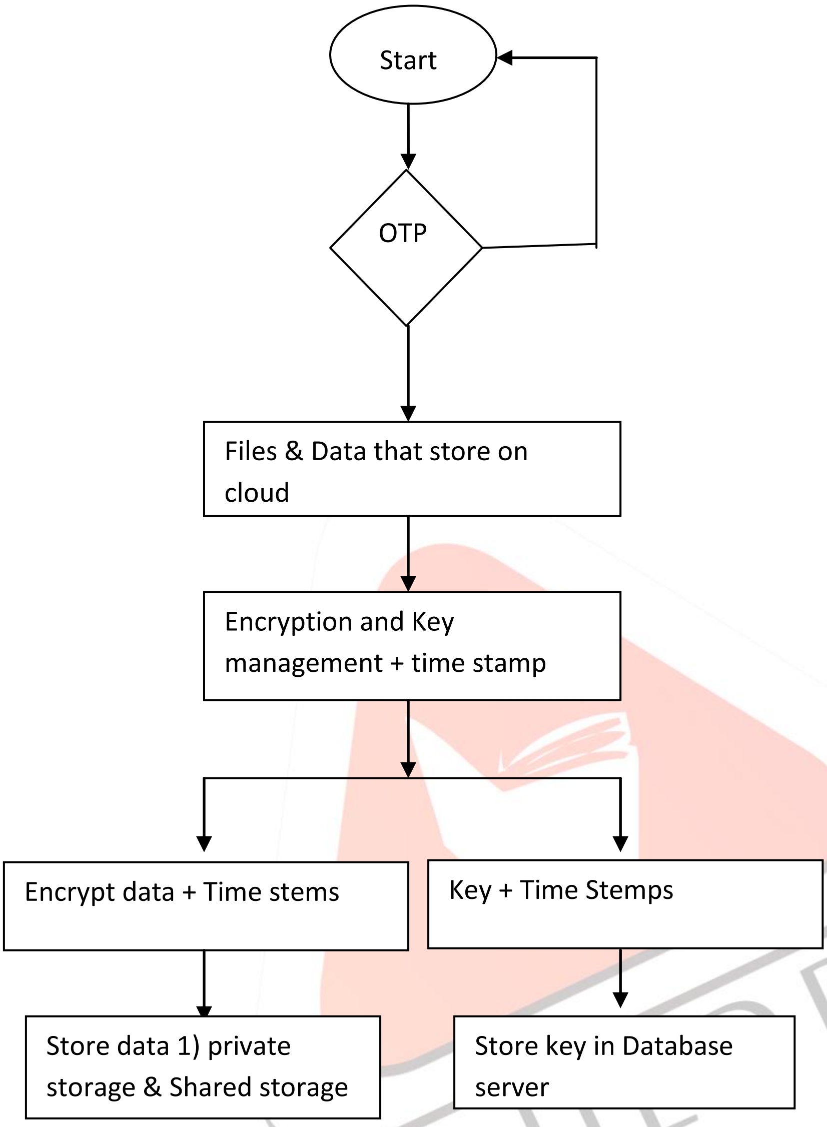

Start
OTP
Files & Data that store on cloud
Encryption and Key management + time stamp
Encrypt data + Time stems
Key + Time Stemps
Store data 1) private storage & Shared storage
Store key in Database server

Figure 3.1 Folw chart for proposed method

Above figure shows the flowchart for proposed method that to be implementing in next phase.

3.2 Proposed architecture

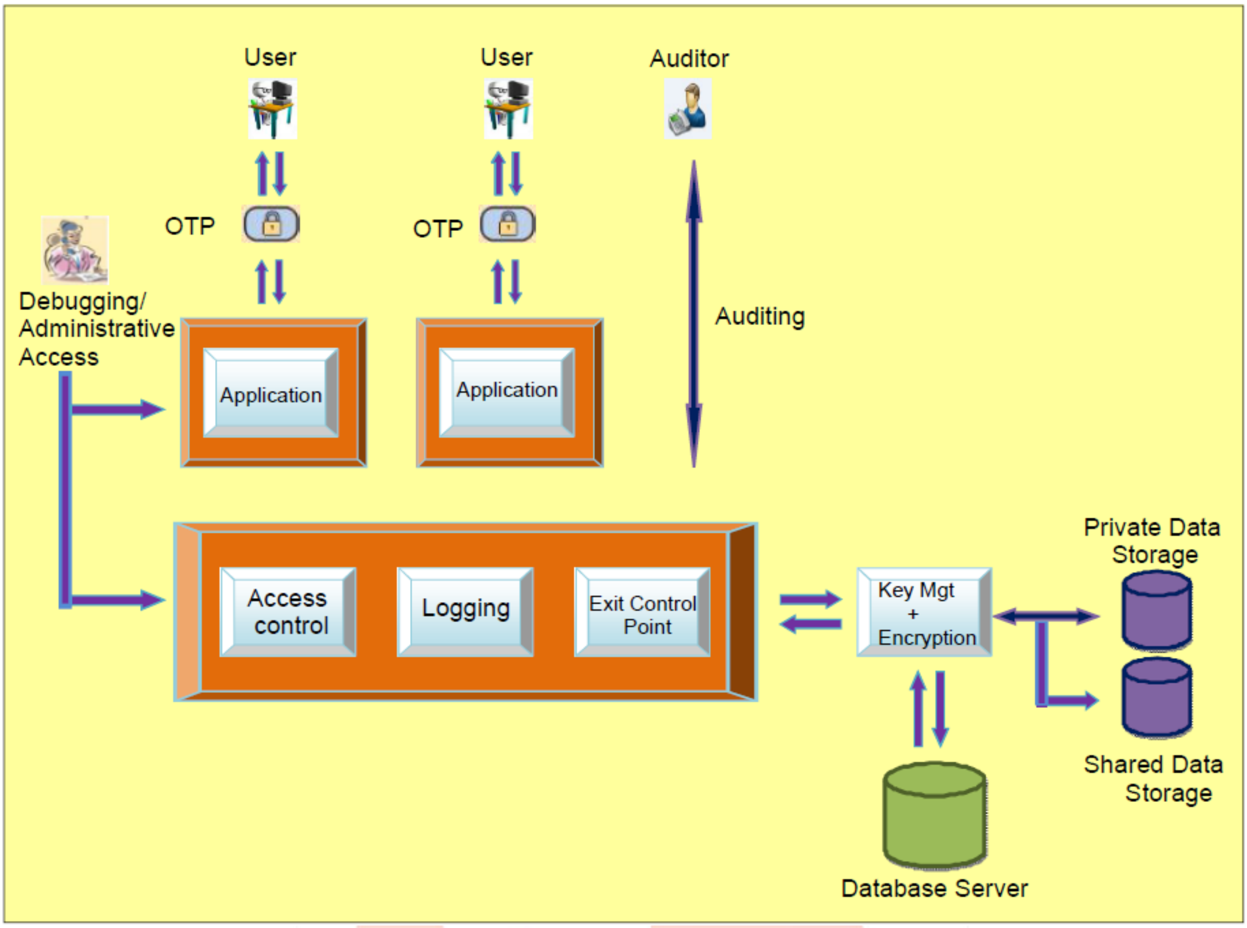


Figure 3.2 Architecture for Advanced DPAAS

3.2.1 Workflow of architecture

As per the architecture given above gives higher security at three level. We provide secure storage in private cloud through given mechanism. Here we provide security in three level,

- Access Control
- Middle layer
- Exit control

So, with this mechanism purpose to give secure data protection in cloud store is fulfill and there is higher protection for storage.

3.2.2 Security Mechanism

At the level of the security in proposed architecture there is defining different rules for the security purpose. In this mechanism used mainly different algorithms for different kinds of files. Basically it’s separated by its extension, if there is document file then there is use different type of algorithm when there is image file then there is differ security mechanism. Here I have use IDEA Encryption algoritham for text or documents where for images there is use Digital water marking.

- Random Number Generator for OTP
    - Multiplicative Congruential Generator
- IDEA algorithm
- Digital water marking

3.3 Implementation Methodology

Implementation of the private cloud for the is covering in this chepter. Implementation is the process of converting newly proposed system design into a real world working design.

3.3.1Implementation of cloud

In this section, steps required to set up the private cloud infrastructure using eucalyptus are explained. Eucalyptus is an acronym for “Elastic Utility Computing Architecture for Linking Your Programs to Useful Systems” and like other leading Cloud platforms, it is an open source application built from the ground up for the implementation of private cloud computing infrastructure thereby allowing full control over the cloud.

- Setup for controller node
- Node controller handling
- Administrative setup by using Eucalyptus Web Interface

Step 1 -Firstly the web page is browsed which tells the user to enter the user name and the password. The default user name and password is admin, admin. User name and password can be changed by user. New login details are provided with the email id of the administrator to set the new user name and password. The cloud will be registered with the new details.
Step 2 -Now the credentials are downloaded by clicking the credential tab shown by the web browser. These credentials will provide the administrative access to the cloud
Step 3 -The downloaded file euca2-admin-x509.zip is copied to /home/admin folder on the cloud controller. Now a folder is created on the cloud controller and zip file is extracted in this folder. The following code does the same thing which explained above.

```
root@ubuntunode:~# mkdir ~/.euca
root@ubuntunode:~# cd ~/.euca
root@ubuntunode:/.euca# unzip ../euca2-admin-x509.zip
root@ubuntunode:/.euca# chmod 0700 ~/.euca
root@ubuntunode:/.euca# chmod 0600 ~/.euca/*
```

- **Installing Cloud Images -** The images tab will list any images that have been registered with the cloud. Each instance or VM running in the cloud is based on an image. No images exist by default after installation, so images need to be installed them.
  Step 1-While it is possible to build custom images and bundle, upload and register them with the cloud. Clicking the “Store” tab in the web interface will show the images that are available from Canonical over the internet. Each instance running on the clod is based on these images.
  Step 2-After the image has been installed, images tab is clicked to confirm that the image has been registered with the cloud. A note of the emi-xxxxxx under the Id column is made as it will be used as the identifier to run an instance.

- **Steps to Run an Instance -** Before we run an instance, it is better to check that there are sufficient resources available in the cloud or not (e.g. the nodes). If sufficient resources are not available then cloud will not be started and errors will occur. So it is better to check availability of resources before running cloud.
  Step 1 Verifying Resources
  “euca-describe-availability-zones” command is used to show all the available resources on cloud nodes.
  ```
  root@ubuntunode:~# euca-describe-availability-zones verbose
  ```

  Step 2: Checking Images
  The command “euca-describe-images” is the command-line equivalent of clicking the “Images” tab in the Eucalyptus administrative web interface.
  ```
  root@ubuntunode:~# euca-describe-images
  ```

  Step 3: Checking Security Groups
  Security groups are basically sets of IP tables firewall rules that control connection requests originating from hosts outside the cloud and destined towards virtual instances running inside the cloud. Security groups within Eucalyptus can be viewed by issuing the following command.
  ```
  root@ubuntunode:~# euca-describe-groups
  ```

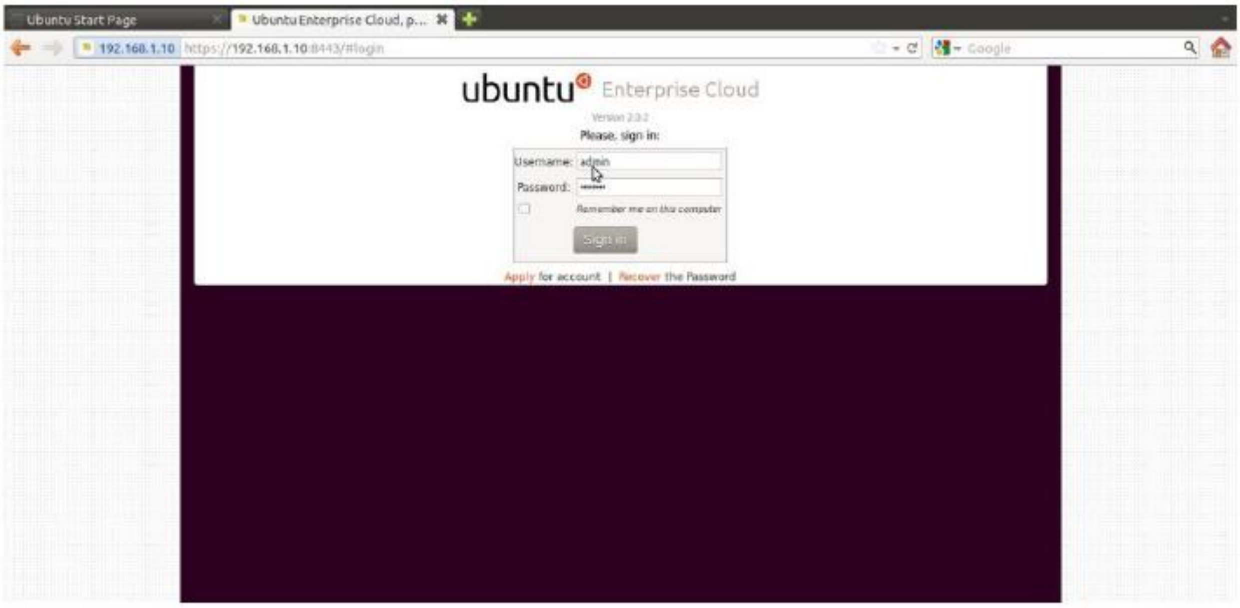
  Step 5: Running the Instance
  Now an instance has been created. If smaller availability zone was selected then it would automatically terminate because of insufficient of space.
  ```
  root@ubuntunode:~# euca-run-instances -g wiki -k mykey -t c1.medium emi-xxxxx
  ```

- **Monitoring and Accessing Instances -** After issuing the “euca-run-instances” command to run an instance, progress can be tracked by euca-describe-instances command. A note of the public IP assigned is made so that instances can be accessed from outside the cloud.
  ```
  root@ubuntunode:~# euca-describe-instances
  ```
  To ensure that the instance is in running state, euca- describe availability zones verbose command is used which shows the current utilization of resources by the running instance.

IV. Implementation Result

4.1 Private Cloud Setup

To develop a private cloud infrastructure we have used open source tool eucalyptus. We have installed Ubuntu Enterprise Cloud in the servers. In first server we installed cloud controller, walrus storage service, cluster controller, and storage controller. After this we create a node and install node controller on server 2. Once the installation is complete the following result appears. We have shown these results in graphical user interface forms.

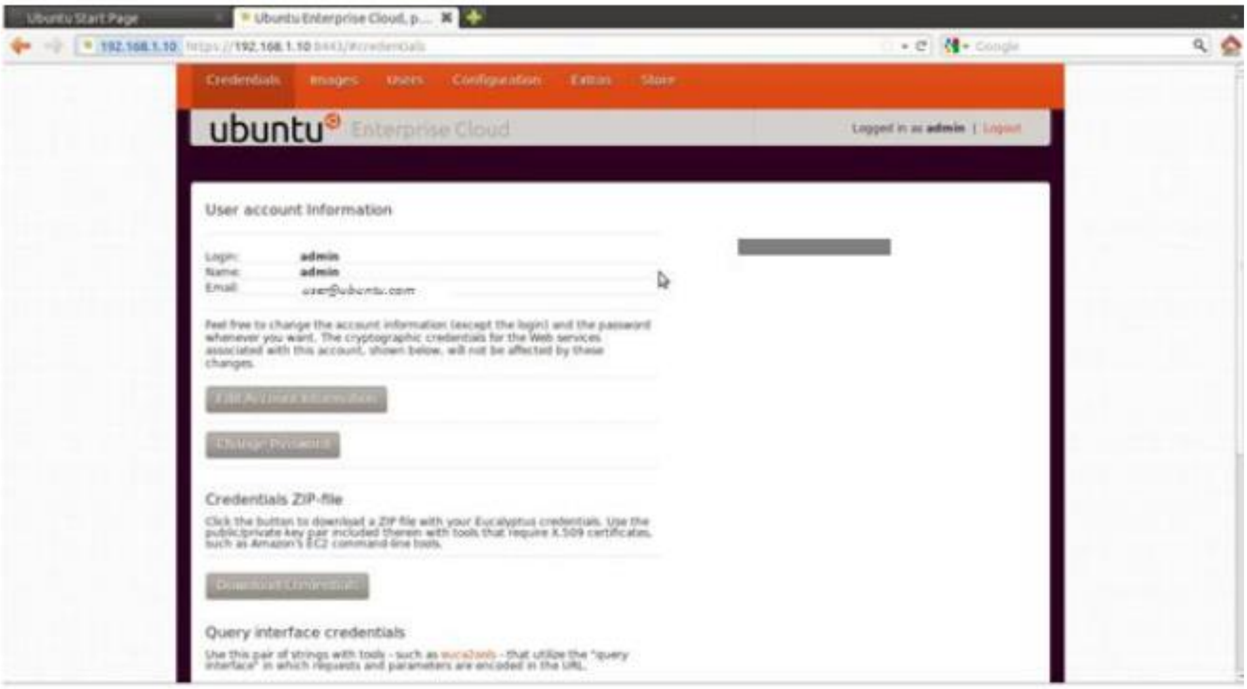


Ubuntu Start Page
Ubuntu Enterprise Cloud, p...
192.168.1.10 https://192.168.1.10:8443/#login

ubuntu Enterprise Cloud
Version 2.3.2
Please, sign in:
Username: admin
Password: ••••••
□ Remember me on this computer
Sign in
Apply for account | Recover the Password

Figure.4.3 Web interface for login

Once cloud gets installed first of all we have login with the default user name and password admin, admin. This step is shown in figure-4. Once user logged in the cloud a home page for the admin will be appear. User has to download and install credentials which consist of certificates and environment variables. This is most important step and it is necessary to use command line utilities which are required to communicate with the cloud. This credential file contains the necessary information needed to allow the administrative access to the cloud. To download the credentials, click on the credential button and zip file can be downloaded from the web interface. This step is shown in figure.



Ubuntu Start Page
Ubuntu Enterprise Cloud, p...
192.168.1.10 https://192.168.1.10:8443/#credentials

Credentials Images Users Configuration Extras Store

ubuntu Enterprise Cloud
Logged in as admin | Logout

User account Information

Login: admin
Name: admin
Email: user@ubuntu.com

Feel free to change the account information (except the login) and the password whenever you want. The cryptographic credentials for the Web services associated with this account, shown below, will not be affected by these changes.

Edit Account Information

Change Password

Credentials ZIP-file

Click the button to download a ZIP file with your Eucalyptus credentials. Use the public/private key pair included therein with tools that require X.509 certificates, such as Amazon's EC2 command-line tools.

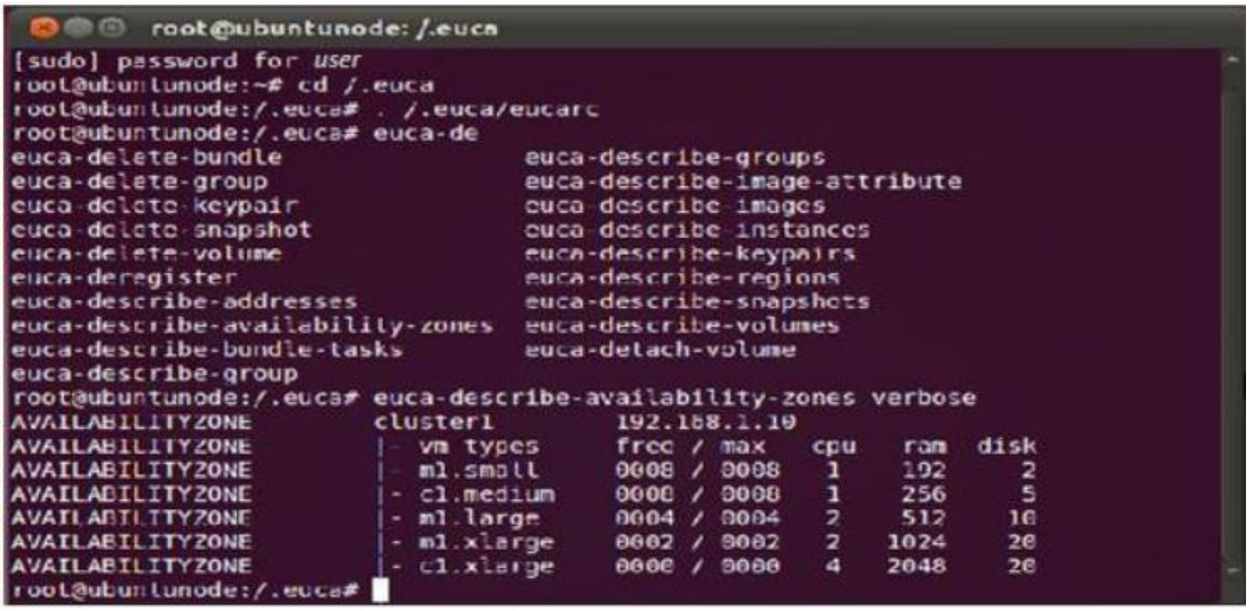
Download Credentials

Query interface credentials

Use this pair of strings with tools - such as euca2ools - that utilize the "query interface" in which requests and parameters are encoded in the URL.

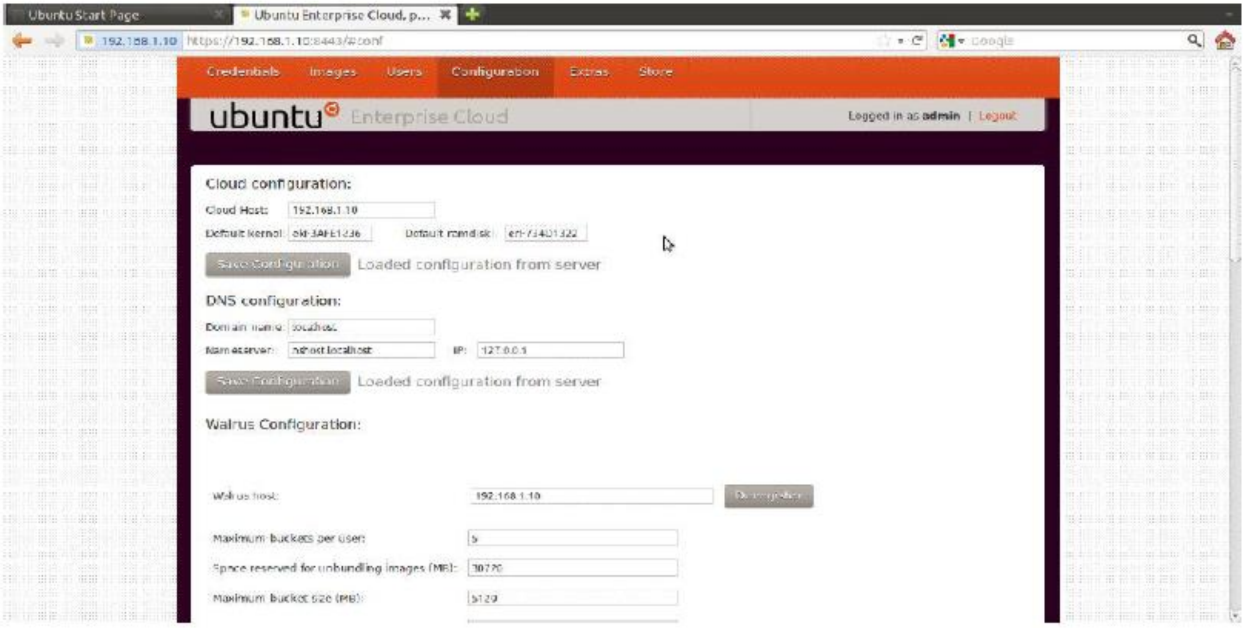
Figure.4.4 Web interface to change account information and to download credentials

When download gets complete, .zip file needs to be extracted for cloud registration. After that Ubuntu Enterprise cloud provides some default images with it. User can use those images by configuring images unique emi as show in fig.



```
[sudo] password for user
root@ubuntunode:~# cd /.euca
root@ubuntunode:/.euca# . /.euca/eucarc
root@ubuntunode:/.euca# euca-de
euca-delete-bundle                     euca-describe-groups
euca-delete-group                      euca-describe-image-attribute
euca-delete-keypair                    euca-describe-images
euca-delete-snapshot                   euca-describe-instances
euca-delete-volume                     euca-describe-keypairs
euca-deregister                        euca-describe-regions
euca-describe-addresses                euca-describe-snapshots
euca-describe-availability-zones       euca-describe-volumes
euca-describe-bundle-tasks             euca-detach-volume
euca-describe-group
root@ubuntunode:/.euca# euca-describe-availability-zones verbose
AVAILABILITYZONE         cluster1         192.168.1.10
AVAILABILITYZONE         |- vm types      free / max   cpu   ram  disk
AVAILABILITYZONE         |- m1.small      0008 / 0008   1    192     2
AVAILABILITYZONE         |- c1.medium     0008 / 0008   1    256     5
AVAILABILITYZONE         |- m1.large      0004 / 0004   2    512    10
AVAILABILITYZONE         |- m1.xlarge     0002 / 0002   2   1024    20
AVAILABILITYZONE         |- c1.xlarge     0000 / 0000   4   2048    20
root@ubuntunode:/.euca# 
```

Figure.4.5 Web interface for configuration



Cloud configuration:

Cloud Host: 192.168.1.10

Default kernel: eki-3AFE1236 Default ramdisk: eri-73401322

Loaded configuration from server

DNS configuration:

Domain name: localhost

Nameserver: nshost.localhost IP: 127.0.0.1

Loaded configuration from server

Walrus Configuration:

Walrus host: 192.168.1.10

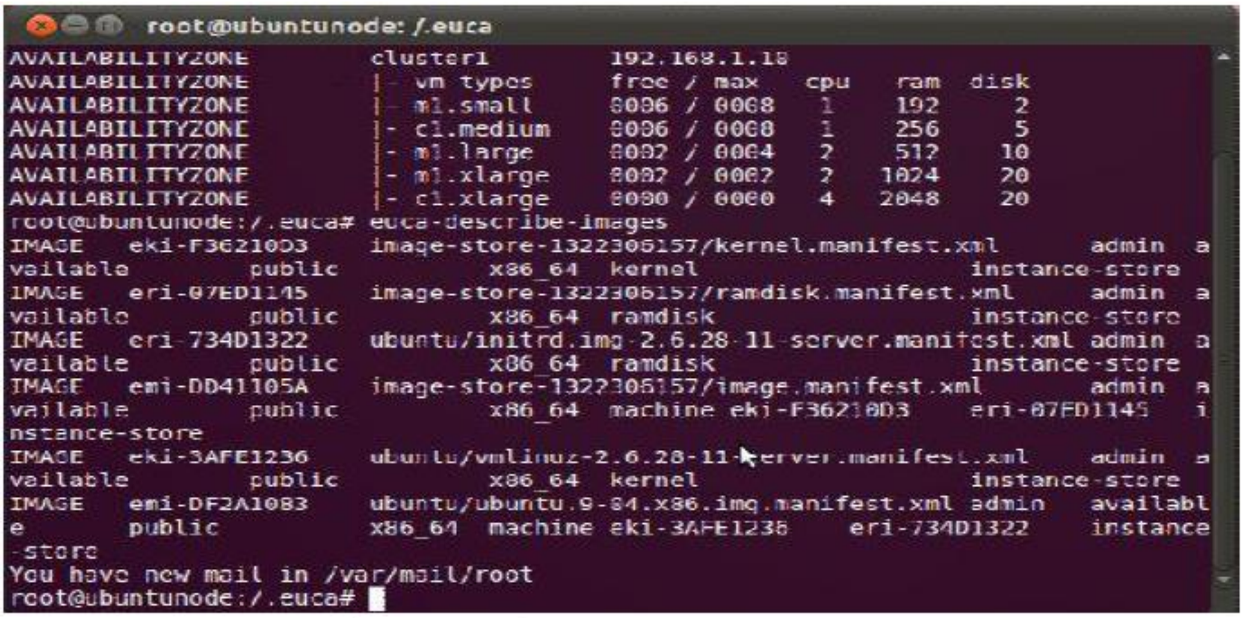
Maximum buckets per user: 5

Space reserved for unbundling images (MB): 30720

Maximum bucket size (MB): 5120

Figure.4.6 Availability zones without any instance

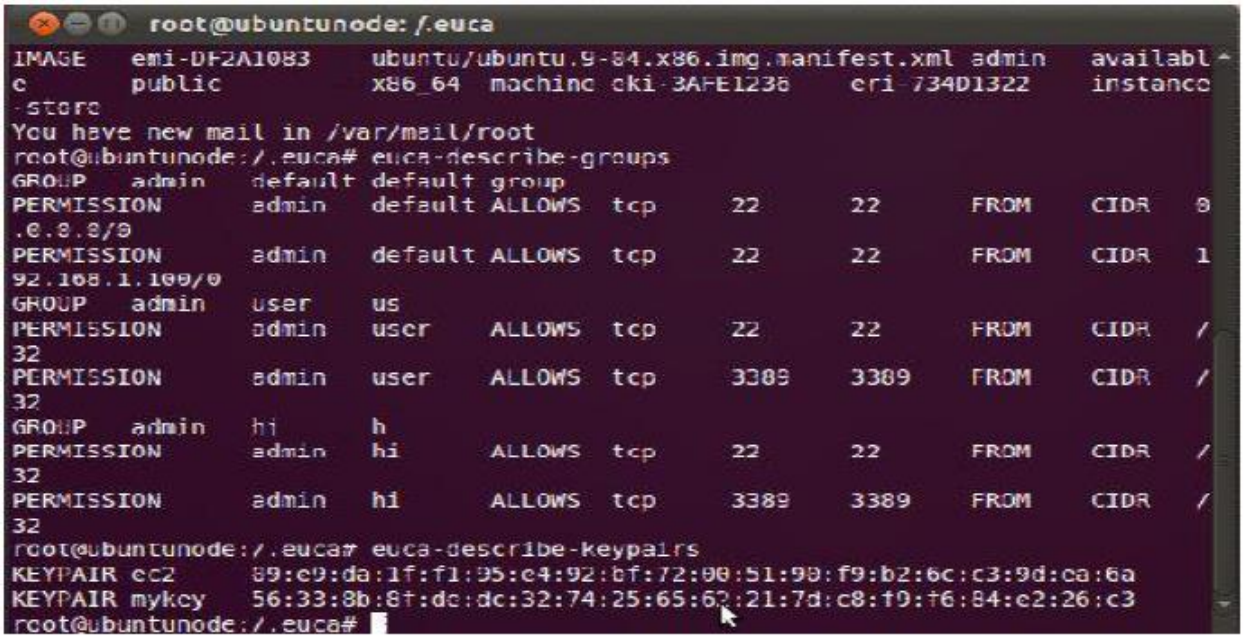
It is necessary to make sure that there are sufficient resources available in the cloud (e.g. the nodes) before creating any node or instance. Because if there is not enough resources, instance will not be generate. Command “euca- oud nodes. describe-availability-zones” is used to show all the available resources on cl Figure 7 shows the availability zones after running the command.



```
root@ubuntunode: /.euca
AVAILABILITYZONE        cluster1        192.168.1.18
AVAILABILITYZONE        |- vm types     free / max   cpu   ram  disk
AVAILABILITYZONE        |- m1.small     0006 / 0008   1    192     2
AVAILABILITYZONE        |- c1.medium    0006 / 0008   1    256     5
AVAILABILITYZONE        |- m1.large     0002 / 0004   2    512    10
AVAILABILITYZONE        |- m1.xlarge    0002 / 0002   2   1024    20
AVAILABILITYZONE        |- c1.xlarge    0000 / 0000   4   2048    20
root@ubuntunode:/.euca# euca-describe-images
IMAGE   eki-F36210D3    image-store-1322306157/kernel.manifest.xml      admin  a
vailable        public          x86_64  kernel                  instance-store
IMAGE   eri-07ED1145    image-store-1322306157/ramdisk.manifest.xml     admin  a
vailable        public          x86_64  ramdisk                 instance-store
IMAGE   eri-734D1322    ubuntu/initrd.img-2.6.28-11-server.manifest.xml admin  a
vailable        public          x86_64  ramdisk                 instance-store
IMAGE   emi-DD41105A    image-store-1322306157/image.manifest.xml       admin  a
vailable        public          x86_64  machine eki-F36210D3    eri-07ED1145   i
nstance-store
IMAGE   eki-3AFE1236    ubuntu/vmlinuz-2.6.28-11-server.manifest.xml    admin  a
vailable        public          x86_64  kernel                  instance-store
IMAGE   emi-DF2A1083    ubuntu/ubuntu.9-04.x86.img.manifest.xml admin   availabl
e       public          x86_64  machine eki-3AFE1236    eri-734D1322    instance
-store
You have new mail in /var/mail/root
root@ubuntunode:/.euca# 
```

Figure.4.7 List of Images registered in the cloud

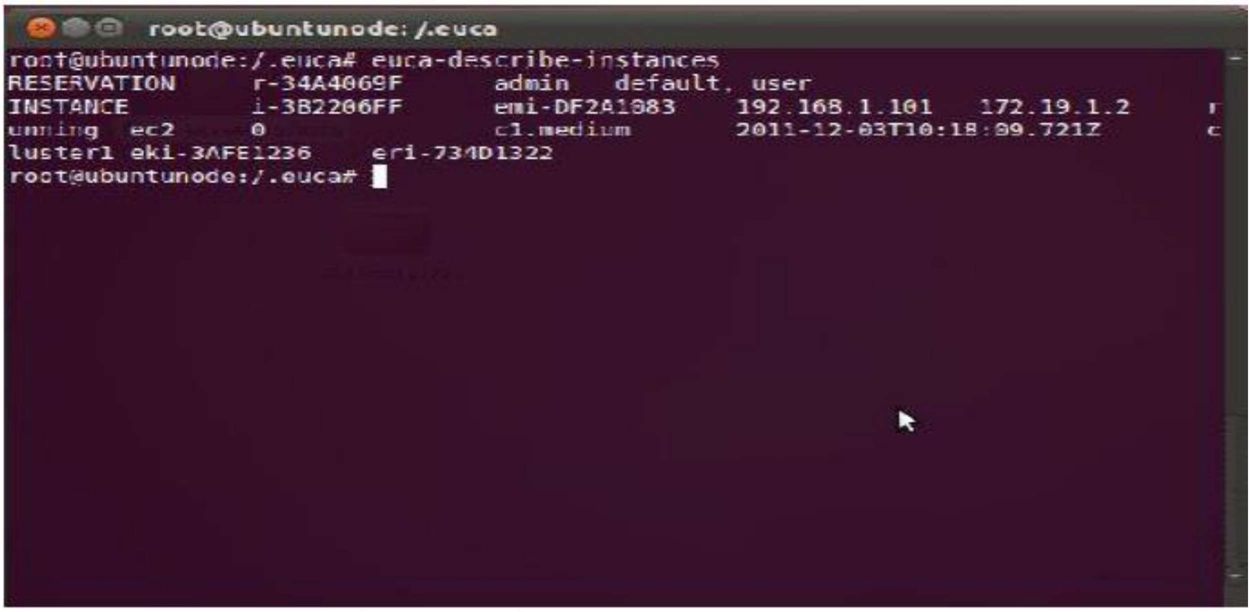
The Ubuntu Enterprise Cloud provide some default images within it but still user can upload and configure their own images. The command "euca-describe-images" is the command-line. This shows the emi identifier for each image that will be used to run an instance. Output of the command is shown in figure.



```
root@ubuntunode: /.euca
IMAGE   emi-DF2A1083    ubuntu/ubuntu.9-04.x86.img.manifest.xml admin   availabl
e       public          x86_64  machine eki-3AFE1236    eri-734D1322    instance
-store
You have new mail in /var/mail/root
root@ubuntunode:/.euca# euca-describe-groups
GROUP   admin   default default group
PERMISSION      admin   default ALLOWS  tcp     22      22      FROM    CIDR    0
.0.0.0/0
PERMISSION      admin   default ALLOWS  tcp     22      22      FROM    CIDR    1
92.168.1.100/0
GROUP   admin   user    us
PERMISSION      admin   user    ALLOWS  tcp     22      22      FROM    CIDR    /
32
PERMISSION      admin   user    ALLOWS  tcp     3389    3389    FROM    CIDR    /
32
GROUP   admin   hi      h
PERMISSION      admin   hi      ALLOWS  tcp     22      22      FROM    CIDR    /
32
PERMISSION      admin   hi      ALLOWS  tcp     3389    3389    FROM    CIDR    /
32
root@ubuntunode:/.euca# euca-describe-keypairs
KEYPAIR ec2     89:e9:da:1f:f1:95:e4:92:bf:72:00:51:90:f9:b2:6c:c3:9d:ca:6a
KEYPAIR mykey   56:33:8b:8f:dc:dc:32:74:25:65:62:21:7d:c8:f9:f6:84:e2:26:c3
root@ubuntunode:/.euca# 
```

Figure.4.8 Available Security groups and Key pairs to access instance

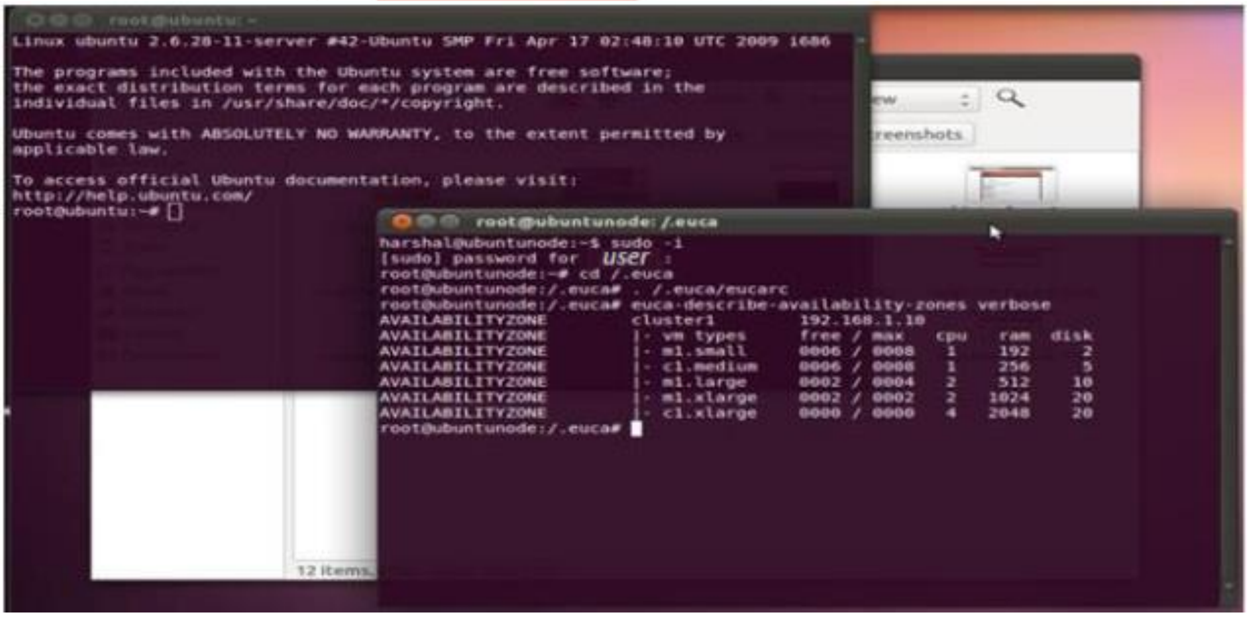
Security groups are basically sets of iptables firewall rules that control connection requests originating from hosts outside the cloud and destined towards virtual instances running inside the cloud. Key pairs are required to be injected into instances. So, that the instantiated VM is accessible. "Euca-describe-groups"used command is used to obtain groups & "Euca-describe-keypairs" command is used to obtain the key pairs as shown in figure.



```
root@ubuntunode:/.euca# euca-describe-instances
RESERVATION      r-34A4069F       admin   default, user
INSTANCE         i-3B2206FF       emi-DF2A1083     192.168.1.101    172.19.1.2      r
unning  ec2      0                c1.medium        2011-12-03T10:18:09.721Z         c
luster1 eki-3AFE1236    eri-734D1322
root@ubuntunode:/.euca# 
```

Figure.4.8 Running instance

After issuing the “euca-run-instances” command to run an instance, its progress can be tracked from pending to running state by using the “euca-describe-instances” command. Public IP assigned to the instance is also given by this command so instance can be accessed from outside the cloud.



```
Linux ubuntu 2.6.28-11-server #42-Ubuntu SMP Fri Apr 17 02:48:10 UTC 2009 i686

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 access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
root@ubuntu:~# 
```

```
harshal@ubuntunode:~$ sudo -i
[sudo] password for  user :
root@ubuntunode:~# cd /.euca
root@ubuntunode:/.euca# . /.euca/eucarc
root@ubuntunode:/.euca# euca-describe-availability-zones verbose
AVAILABILITYZONE        cluster1        192.168.1.10
AVAILABILITYZONE        |- vm types     free / max   cpu   ram  disk
AVAILABILITYZONE        |- m1.small     0006 / 0008   1    192     2
AVAILABILITYZONE        |- c1.medium    0006 / 0008   1    256     5
AVAILABILITYZONE        |- m1.large     0002 / 0004   2    512    10
AVAILABILITYZONE        |- m1.xlarge    0002 / 0002   2   1024    20
AVAILABILITYZONE        |- c1.xlarge    0000 / 0000   4   2048    20
root@ubuntunode:/.euca# 
```

Figure.4.9 Availability zones after creating an instance

Once instance start running the number of availability zones will automatically reduced we can see the reduced availability zone by typing Command “euca-describe-availability-zones”. It will show all the available resources on cloud nodes.

4.2 comparisons for best multimedia encryption algorithm

- DES is the old "data encryption standard" from the seventies. Its key size is too short for proper security (56 effective bits; this can be brute-forced, as has been demonstrated more than ten years ago). Also, DES uses 64-bit blocks, which raises some potential issues when encrypting several gigabytes of data with the same key (a gigabyte is not that big nowadays).
- 3DES is a trick to reuse DES implementations, by cascading three instances of DES (with distinct keys). 3DES is believed to be secure up to at least "2^{112}" security (which is quite a lot, and quite far in the realm of "not breakable with today's technology"). But it is slow, especially in software (DES was designed for efficient hardware implementation, but it sucks in software; and 3DES sucks three times as much).
- Blowfish is a block cipher proposed by Bruce Schneier, and deployed in some softwares. Blowfish can use huge keys and is believed secure, except with regards to its block size, which is 64 bits, just like DES and 3DES. Blowfish is efficient in software, at least on some software platforms (it uses key-dependent lookup tables, hence performance depends on how the platform handles memory and caches).

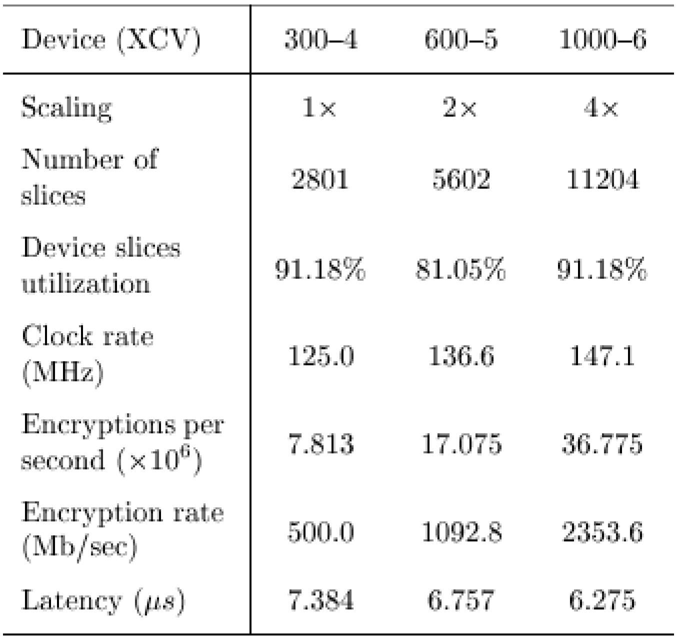
- AES is the successor of DES as standard symmetric encryption algorithm for US federal organizations (and as standard for pretty much everybody else, too). AES accepts keys of 128, 192 or 256 bits (128 bits is already very unbreakable), uses 128-bit blocks (so no issue there), and is efficient in both software and hardware. It was selected through an open competition involving hundreds of cryptographers during several years. Basically, you cannot have better than that.
- Here is new IDEA algorithm that used for encryption techniques because its suitable for easy and secure encryption for media file. Here we only think for text document files. Below the results and implementation for IDEA.

V. Implementation

Although IDEA involves only simple 16-bit operations, software implementations of this algorithm still cannot offer the encryption rate required for on-line encryption in high-speed networks. Software implementation running on a Sun Enterprise E4500 machine with twelve 400MHz Ultra-Hi processor, performs 2.30 x 106 encryptions per second or a equivalent encryption rate of 147.13Mb/sec, still cannot be applied to applications such as encryption for 155Mb/sec Asynchronous Transfer Mode (ATM) networks.

Hardware implementations offer significant speed improvements over software implementations by exploiting parallelism among operators. In addition, they are likely to be cheaper, have lower power consumption and smaller footprint than a high speed software implementation. The first VLSI implementation of IDEA was developed and verified by Bonnenberg et. al. in 1992 using a 1.5 μm CMOS technology [7]. This implementation had an encryption rate of 44Mb/sec. In 1994, VINCI, a 177Mb/sec VLSI implementation of the IDEA algorithm in 1.2 μm CMOS technology, was reported by Curiger et. al. [5, 11]. A 355Mb/sec implementation in 0.8 μm technology of IDEA was reported in 1995 by Wolter et. al. [10]. The fastest single chip implementation of which we are aware is a 424Mb/sec implementation of 0.7 μm technology by Salomao et. al. [9]. A commercial implementation of IDEA called the IDEACrypt coprocessor, developed by Ascom achieves 300Mb/sec [2].

A high performance implementation of the IDEA presented by Leong [8] uses a novel bit-serial architecture to perform multiplication modulo 216 + 1; the implementation occupies a minimal amount of hardware. The bit-serial architecture enabled the algorithm to be deeply pipelined to achieve a system clock rate of 125MHz. An implementation on a Xilinx Virtex X CV300-4 was successfully tested, delivering a throughput of 500Mb/sec. With a X CV1000-6 device, the estimated performance is 2.35Gb/sec, three orders of magnitude faster than a software implementation on a 450MHz Intel Pentium II. This design is suitable for applications in online encryption for high-speed networks. The results of Leong's experiment are summarized in Table 3.



Device (XCV)	300–4	600–5	1000–6
Scaling	1×	2×	4×
Number of slices	2801	5602	11204
Device slices utilization	91.18%	81.05%	91.18%
Clock rate (MHz)	125.0	136.6	147.1
Encryptions per second ($\times 10^6$)	7.813	17.075	36.775
Encryption rate (Mb/sec)	500.0	1092.8	2353.6
Latency (μs)	7.384	6.757	6.275

Fig:4.10. Results of experiment

Applications

Today, there are hundreds of IDEA-based security solutions available in many market areas, ranging from Financial Services, and Broadcasting to Government. IDEA is the name of a proven, secure, and universally applicable block encryption algorithm, which permits effective protection of transmitted and stored data against unauthorized access by third parties. The fundamental criteria for the development of IDEA were highest security requirements along with easy hardware and software implementation for fast execution.

The IDEA algorithm can easily be embedded in any encryption software. Data encryption can be used to protect data transmission and storage. Typical fields are:

– Audio and video data for cable TV, pay TV, video conferencing, distance learning, business TV, VoIP
– Sensitive financial and commercial data

– Email via public networks
– Transmission links via modem, router or ATM link, GSM technology
– Smart cards

VI. CONCLUSION

Summarization of the report is conveying the higher security paradigms. Such as a layer that include between the middle ware and entrusted storage device that make cloud storage system very much secure. With this architecture we can also band the unauthorized entry to access the important and private data of the cloud users. Behalf of this architecture key management and key usage is easy to manage. At the end we can conclude that proposed architecture increase the security and high performance of the cloud.

REFERENCE

[1] Anas BOUAYAD, Asmae BLILAT, Nour el houda MEJHED Mohammed EL GHAZI. “Preventing Cloud computing : security challenges”. In IEEE 2012, Seattle, WA, USA.

[2] Upma Goyal, and Gayatri Bhatti “A Dual Mechanism for defeating DDoS Attacks in Cloud Computing Model” In International Journal of Application or Innovation in Engineering & Management (IJAIEM), ,vol-2, pages 34-39 March-2013.

[3] Chun-Ting Huang,; A.; Zhongyuan Qin “Multimedia Storage Security in Cloud Computing: An Overview”. IEEE , Year-2011.

[4] T. Shashi Kumar.; Mrs. J. Deepthi; “Data Protection for Cloud Using Homomorphic Mechanism”. In Proceedings of the International Journal of Computer Science and Mobile Computing. Vol-2,Aug-2013.

[5] Dawn Song.; Elaine Shi Umesh Shankar, Google. “Cloud Data Protection for the Masses”. In IEEE. 2012.

[6] P. Kiran Rao; V. Lakshmi Sailaja; Alfisha Khan;. “High level security in cloud for scalable data”. International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, March, 2013

[7] P.Subhasri;, Dr.A.Padmapriya. "Multilevel Encryption for Ensuring Public Cloud,” in International Journal of Advanced Research in Computer Science and Software Engineering, july, 2013.

[8] P.Subhasri;, Dr.A.Padmapriya. "Multilevel Encryption for Ensuring Public Cloud,” in International Journal of Advanced Research in Computer Science and Software Engineering, july, 2013.

[9] How-Shen Chang. " International Data Encryption Algorithm,” CS-627-1, july, 2004.

Websites

1. International Data Encryption Algorithm - Wikipedia, the free encyclopedia
2. http://stackoverflow.com/questions/5554526/comparison-of-des-triple-des-aes-blowfish-encryption-for-data