

Scalable Cloud Computing Platform

Muskan Pawar¹, Siddesh Naikare², Ritika Jagtap³, Dnyaneshwar Bavkar⁴

^{1, 2, 3, 4} Dept of Computer Engineering

^{1, 2, 3, 4} Mumbai University/Terna Engineering College, Navi Mumbai`

Abstract- In simple terms, cloud computing means storing data, accessing data over the Internet rather than a computer's hardware. In this project, we will be making a Cloud Computing Platform almost like Amazon Web Services. Our main aim is to enable the user to create a virtual machine using Virtual Environment. The user has got to launch an instance through the front end and a virtual machine is created at the back end. The virtual machine created provides a machine almost like one's PC with manageable RAM and CPU. With Virtual Environment, one can create, terminate and delete an instance. For the purpose of creating a virtual machine, we will be using docker. Since the scope of the project seems to be quite large, we will be using Computer A as our cloud platform and the user will be connected to it through the web interface and will be able to use that Computer A as their cloud. And now the user can use it for whatever purpose it wants.

Keywords-Cloud Computing, Docker, Virtual Environment.

I. INTRODUCTION

The world is rapidly moving toward cloud technology for easy access of data. Many big companies such as Instagram, Pinterest, Netflix, Amazon have already adopted this technology and shown us how valuable cloud can be. Cloud computing is majorly used to share data and computer resources without direct involvement of the user. The primary aim of cloud is to scale back costs, and helps the users specialize on their actual work instead of being interfered by some technology related obstacles [1]. We have a plethora of options while choosing our own cloud provider. Some popular ones include Amazon AWS, Microsoft Azure, Google Cloud, IBM Cloud, etc.

Hence, this project is made to understand how this technology works. The main objective of this project is to launch an instance, edit the instance and to delete an instance.

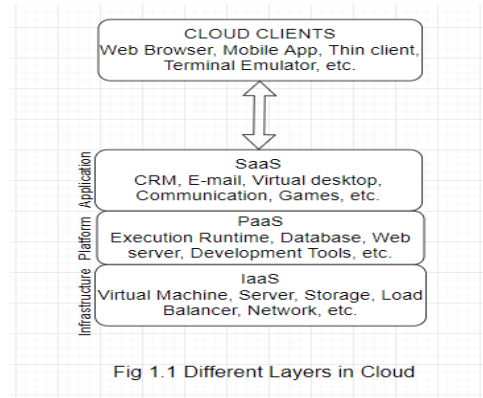


Fig 1: Different layers in cloud

II. LITERATURE SURVEY

With the increasing popularity of container technologies, many research efforts have been made to explore the advantages of containers, improve container performance and security, as well as to compare the containers with the VMs.

Advantages of using container technology in cloud computing:

- easy deployment, configuration, as well as convenient management of the applications in the cloud.
- a very low overhead HPC environment
- enhanced the container security and prevent outside attacks
- better for disk and network I/O intensive applications
- low power consumption for network-intensive applications.

Following papers were considered for the research of the project-

- Introduction to Docker for reproducible research – C Boettiger
- Docker: Lightweight Linux containers for consistent development and deployment – D Merkel

III. METHODOLOGY

In simple terms, Docker is a tool which makes it easy for user to install, test and implement any application faster than ever. Docker has two backbones: Images and Containers [4]. Container is like one box that in holds all the dependencies any application needs whereas Image can be interpreted as blueprint of the container. The prime purpose of selecting docker was the concept of containerization. Even a minute amount of storage becomes important at higher organizational level, we made sure of not wasting any of it and hence used docker.

Working of Docker:

In simple words, Docker user types the commands required within the chosen instruction interface/ user interface and these commands are directed to the Docker Daemon located within the Docker Server through the remainder API. Rest API uses HTTP requests to urge, PUT, POST and DELETE data. The Docker Daemon then searches for the images mentioned within the commands for it to run, in the local computer and if it fails to seek out, it goes to the Docker registry and installs the image required by the user in the local computer. All the images installed will have containers. Now the command given by the user runs successfully.

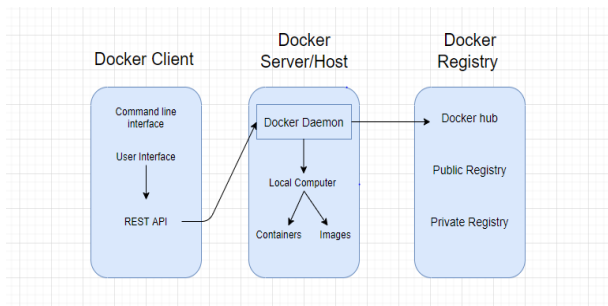


Fig 2: Docker Architecture

Virtualization v/s Containerization

Containers consists of libraries and other dependencies of various applications that runs under an equivalent operating system which is that the host operating system. All of the items that a container consists of, have size of up to tens of MBs and hence requires less space comparatively. In virtualization, every application has its own virtual environment and hence its own operating system which is guest operating system. So, for every application there is operating system, libraries and other dependencies resulting in space requirement of tens of GBs which is quite larger than containers. This is the primary reason of selecting the concept of containerization over virtualization for our project and

hence to supply the mechanism of containerization, docker platform is employed.

Containerization:

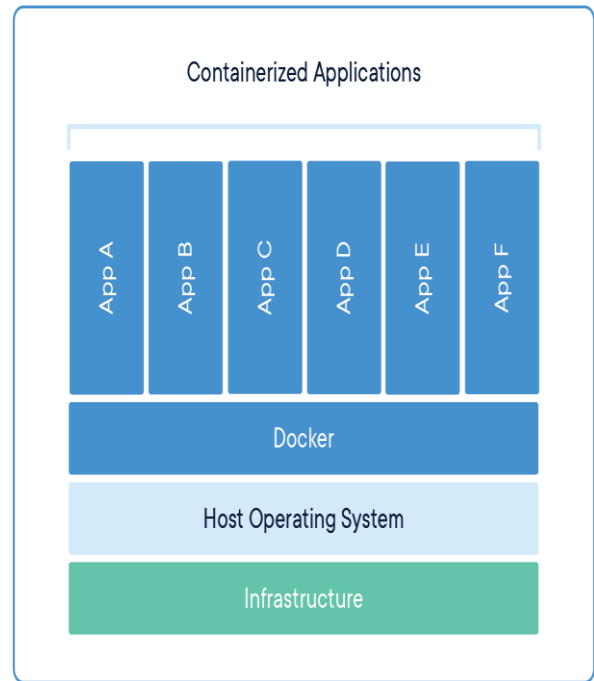


Fig 3: Containerization

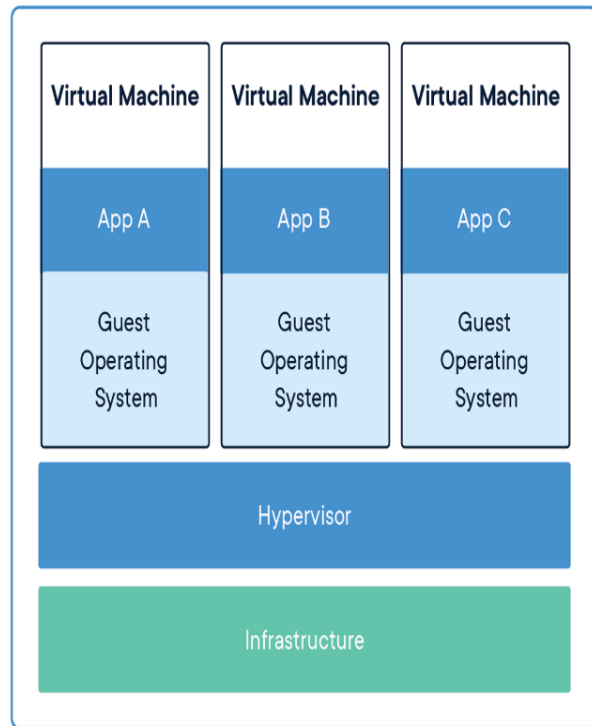


Fig 4: Virtualization

IV. SOFTWARE DESIGN

Understanding System Architecture

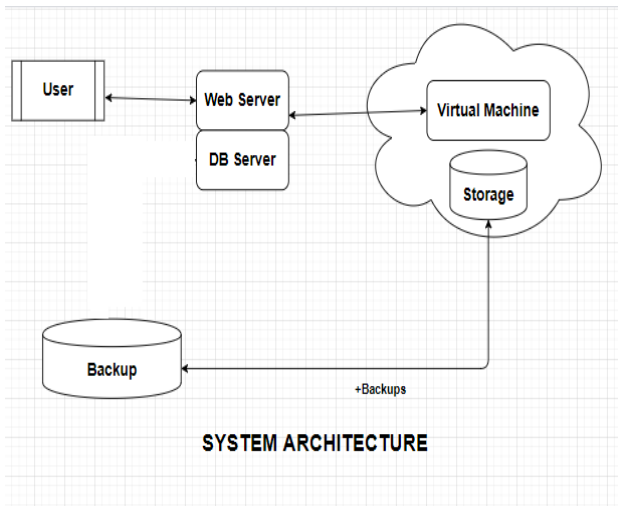


Fig 5: System Architecture

The user logs into his account with the assistance of a login page made using HTML+Java at the front end and his credentials are checked at the back end through DB Server made using MySQL. The user creates a Virtual Machine. The work user will be doing on the virtual environment will be saved in the backup in case the system crashes. As soon as the user is granted access in to his account, all these tasks that the user was working on, are displayed. User even has the authority to switch the instances he has created. This is the essential architecture of our project. To fire a virtual environment there would be an on-click button that says 'create' and docker will be used to process the virtual machine at the back end. There would be just one server but we can configure multiple servers even, but for that the user should have multiple machine that acts as a server. Once the user has created the instances, he can manage it through putty and hence can use it using the terminal access of the putty. In AWS, the user can access putty by providing the private key but we will be doing it by the user credentials. The user will have to enter his details and also the detail of the instance created like its IP address and port number.

Use Case diagram

The user will register with username and password he had used. The admin will create cloud server for the user. The user can define functionalities of virtual environment like its RAM, storage capacity, operating system etc. after virtual environment has been updated by the user. It can now use it for any purpose through the terminal access provided. The

virtual environment created often modified, terminated or stopped.

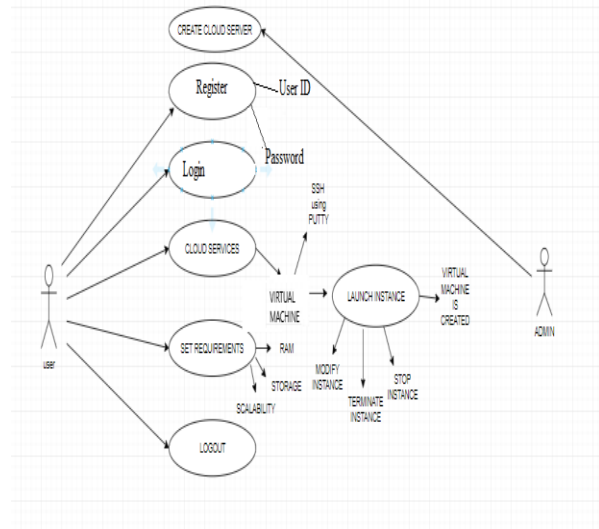


Fig 6: Use Case Diagram

- To manage the instances putty is employed and therefore the user has access to it by providing its credentials and hence ensuring security.
- Backup is made of each instance the user has created so if just in case there is any hardware failure/software error user has to not worry about the data lost hence ensuring reliability
- As virtual environment is provided there is a growth in ram, CPU w.r.t.size and hence providing scalability.
- As the system would work over internet and the user will certainly have internet access, the system is available 24/7 to the user.

V. RESULTS

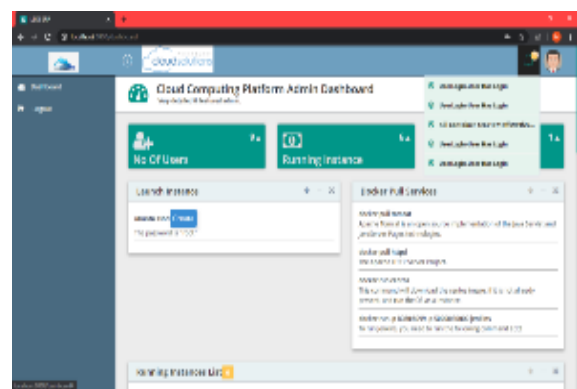


Fig 7: user launches an instance via the “create” button through web interface

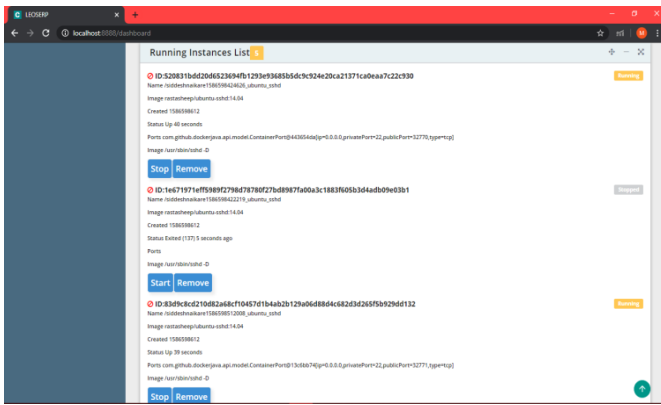


Fig8 : Instances launched are displayed in a separate section where modification options are available

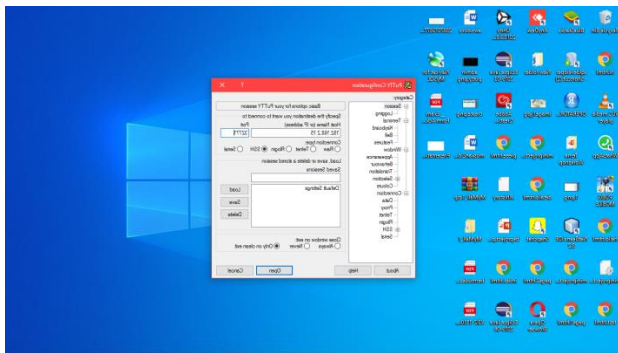


Fig9: To access the virtual machine created, putty is used

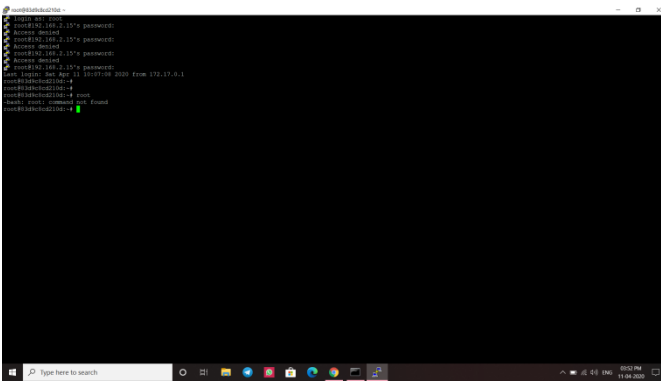


Figure 10: Putty connection succeeded by providing credentials.

VI. LIMITATIONS

While there are numerous advantages of cloud platform it comes with a touch limitation too.

- As these services require internet, the most important drawback is downtime. So it is important to stay this in mind and work accordingly.
- Storing and accessing data and files on external service providers will always have risk of hacking. There has already been an attack on AWS EC2 console that led to data deletion. Hence it is

important to see and verify your security details every now and then[7].

- Since the cloud infrastructure is totally owned and managed by the service provider, it gives minimal control to the user.

Drawbacks of Docker:

- By design, all of the data inside a container leaves forever when it closes down except you reserve it somewhere else first. You would like to feature additional tool: Docker Data Volumes.
- As docker containers share access to one host operating system, there are high chances of any malicious code to get access to your computer memory.
- It has given less attention to rich GUI application. So if you are planning to develop a desktop GUI application, you got to reconsider it.

VII. CONCLUSION

Hence, the project primarily focuses on providing storage to the user in simplest way but not on a large scale. We made sure of using the concept of containerization that will help user to efficiently use every percent of their storage with not much manual attention. Though this project does not aim t create a betterment of other cloud computing platforms, it was just reflection of them in the simplest form. This project can further be extended in the future by providing load balancing and a database infrastructure within the system quite like leading platforms like AWS, just to make oneself aware of the insights of it.

VIII. ACKNOWLEDGEMENT

We would wish to take this chance to precise our gratitude towards all the people that have in various ways, help in the successful completion of our project. We must convey our gratitude to our project guide Prof. Dyaneshvar Bavkar for being a continuous source of inspiration and helped in preparing the project. Our parents who always bear with us in every critical situation and supported us as and when required.

As we give expression to our love and appreciation our heart is fill. And we will always appreciate your valuable help.

REFERENCES

- [1] https://www.researchgate.net/publication/273393727_A_Study_On_Virtualization_Techniques_And_Challenges_In_Cloud_Computing&ved=2ahUKEwjC5pblgofqAhWQSH0KHR0QCxkQFjABegQIBRAC&usg=AOvVaw2CPYb-ZgLyIUICY4GSO3zp
- [2] https://www.researchgate.net/publication/309537961_An_Efficient_Cloud_Model_with_integrated_Services_by_addressing_Major_Security_Challenges
- [3] https://en.wikipedia.org/wiki/Cloud_computing#Deployment_models
- [4] [https://en.wikipedia.org/wiki/Docker_\(software\)](https://en.wikipedia.org/wiki/Docker_(software))
- [5] https://subscription.packtpub.com/book/application_development/9781788992329/1/ch011v11sec13/docker-container-architecture
- [6] <https://www.docker.com/resources/what-container>
- [7] <https://cloudacademy.com/>
- [8] M. G. Xavier, M. V. Neves, F. D. Rossi, T. C. Ferreto, T. Lange, and C. A. De Rose. Performance evaluation of container-based virtualization for high-performance computing environments. In Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on, pages 233–240. IEEE, 2013.