

An Approach to Enhance the Architecture of Public Cloud System

Pankaj¹ Shalini² Kirti Bhatia³

^{1,2}M. Tech. Scholar ³Assistant Professor

^{1,2,3}Department of Computer Science & Engineering

^{1,2,3}SKITM, Haryana, India

Abstract— Cloud computing is a way of computing, where most of our data is stored in the cloud, i.e., the Internet. Cloud computing systems reduce the need for advanced hardware on the client side. In the paper, we will discuss the methods that will be used in the user friendly Interface for the development of “Custom interface for AWS public cloud communication”. Through this interface user(s) can create machines on AWS Public Cloud using AWS API.

Keywords: Cloud, AWS, Python, Django, API, AWS EC2, AWS IAM

I. INTRODUCTION

Cloud computing is the latest effort in delivering computing resources as a service. In this paper, we will simplify the problem faced by accessing the services of AWS. We will create an Interface that will give a user-friendly environment to the user. Amazon Web Service is a cloud provider which has many services such as AWS EC2. It has resources consist of multiple types (or dimensions) including CPU, memory, disk and network bandwidth. The resources supplied by cloud providers can vary over time due to highly dynamic workloads that require resizing, creating and (or) terminating Virtual Machines [1].

II. TECHNOLOGIES USED

We have used the following technologies for creating the custom interface of AWS:

A. Public Cloud

Public Cloud is a type of deployment model in Cloud. They deliver their computing resources like servers and storage over the Internet. Public Cloud is a pool of computing resources where the infrastructure is owned by the cloud service provider and the service is offered to a large group of users on the basis of utilization of the resources.

In this model, the resources are shared among the customers, and the service is offered through the internet. Amazon's Elastic Compute Cloud (EC2) is one such example of the public cloud [2]. Multiple enterprises can work on the infrastructure provided, at the same time [3].

Example: Amazon Web Services (AWS), Microsoft Azure.

B. Amazon Web Services (AWS)

Amazon Web Service is a collection of remote computing services (called web services) that create up a Cloud computing platform- is a subsidiary of Amazon. It provides on demand cloud computing platforms to individuals, companies, and organizations. It enabled business and developers to use web services to build scalable and sophisticated applications. AWS contains many services and the most well-known of AWS services AWS EC2. The service is much faster, scalable and cost-effective than building a physical server [4]. AWS helps companies to select the platform that is suitable according to the problem they

have and pay only for what they use. In addition, AWS applies advanced physical security and data privacy techniques to protect users' data [5]. AWS is the most trusted provider of cloud computing which provides the excellent cloud security and excellent cloud services [6].

C. EC2

The Amazon Elastic Compute Cloud (EC2) is an Infrastructure-as-a-Service cloud provider giving on-demand compute capacity. They provide virtual computing environments, known as instances. EC2 will provide service to launch instances. It can be accessed on hourly base [7]. In particular, each user is allowed to run any pre-installed virtual machine image called Amazon Machine Image. To simplify the setup of a server, Amazon offers an online catalogue where users can choose between large numbers of AMIs that are pre-installed with common services such as web servers, web applications, databases [8], secure login information using key pairs (public key and private key), various configurations of CPU, memory, storage and networking capacity of the instances are known as instance types. Amazon provides a number of different instance types that have varying performance characteristics. CPU capacity is defined in terms of an abstract Amazon EC2 Compute Unit. One EC2 Compute Unit is approximately equivalent to a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor. For example m1.large instances type. The m1.large instance type has four EC2 Compute Units, two virtual cores with two EC2 Compute Units each, and 7.5 GB of memory [9].

It has resizable compute capacity in the cloud, which can be customized according to the users need. Amazon EC2 is able to handle changes in requirements and also reduces the needs to forecast traffic. This service can be used in public cloud and it cannot be extended to private cloud. This service can be used in public cloud and it cannot be extended to private. Multicast, broadcast, and multiple Ethernet switching are not supported in EC2 [2].

D. AWS IAM: (Identify and Access Management)

AWS IAM manages the AWS services and specific resources securely. Using IAM, we can create and manage AWS users and groups, and also allow and deny their access to AWS resources. IAM controls authentication and authorization as shown in Fig. 1. AWS IAM provides options to configure granular permissions in AWS environments. It is recommended to give the least amount of permissions to manage AWS resources required for performing the job function. [10]



Fig. 1: AWS IAM

E. Python

It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. Python is a powerful, high level and easy-to-learn programming language based on traditional languages.

Computer science is constantly advancing with newer, faster hardware, better techniques. Earlier, a new language called C as the language of choice in the mainframe world; it was trying to replace Assembler, COBOL, and Fortran. Today, Python is one of several modern languages that use these venerable languages as a foundation while it is better suited to current operating systems, networks, and hardware [11]. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code. It provides constructs that enable clear programming on both small and large scales [12]. We have designed the interface by using Python and its framework Django.

F. Django

Django is a free and open source web application framework which is written in Python. A web framework is a set of components that help to develop websites faster and easier. It is an advantage that Django is built using Python. This is because it can create more functions with fewer lines of code. Django packages and libraries simplify the development process that gives the developer saving time in delivering the final product. It has a rich library of pre-made tools so the developer doesn't have to reinvent the wheel [13].

G. Visual Studio Code

Visual Studio Code is a source code editor which is developed by Microsoft for Windows, Linux and macOS Operating System. It supports macOS, Linux, and Windows - so we can run any program no matter what the platform is. We have designed the interface in Visual Studio Code by using Python and Django.

III. INSTANCE LAUNCH PROCESS OF EC2

AWS EC2 is a service of AWS which has many features and option. If we want to launch instance(s), we need to understand the whole working of the EC2 service like selecting of AMI, CPU etc.

The following are some steps for selecting the resources of AWS EC2.

A. Select AMI in AWS EC2

AMI is a template that contains the software configurations like (operating system, application server, and applications) required to launch our instance.

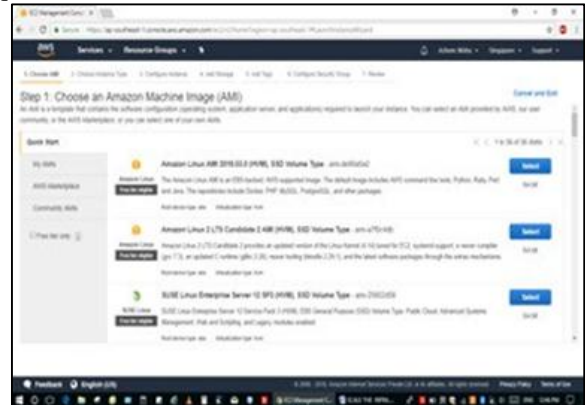


Fig. 2: Snapshot of selecting AMI (Amazon Machine Image)

B. Select Configure Security Group

A security group is a set of firewall rules that control the traffic for the instance. On this page, we can add rules to allow specific traffic to reach the instance as shown in Fig. 3.

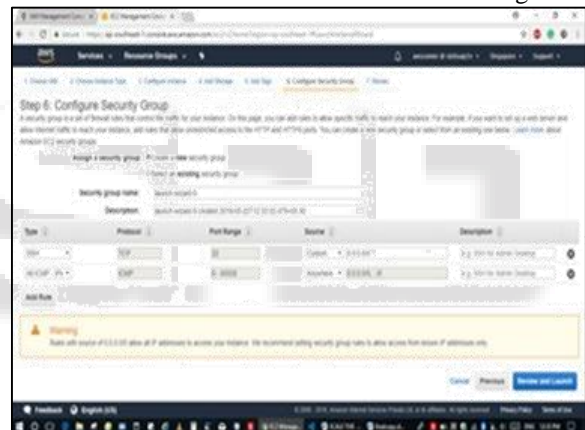


Fig. 3: Snapshot of selecting Configure Security Group

IV. IMPLEMENTATION

It is difficult for the user to choose different steps in EC2 dashboard. EC2 has many resources which are complex to understand for nontechnical users.

To simplify the problem, user friendly custom interface is created and it will be connected to the AWS. Fig.4 is the reference image of the Custom Software.

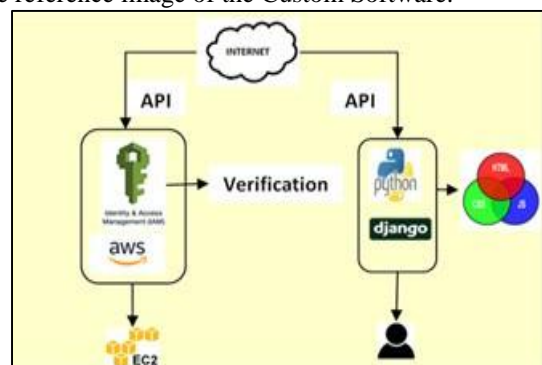


Fig. 4: An outline diagram of the Project.

The project gives us an Interface which will be connected to the AWS Cloud API by authenticating AWS IAM service. AWS IAM (Identity and Access Management) is a service which we can use to isolate the other AWS services on users and groups basis. Once the credentials are created using AWS IAM service then we have to put those in the project so that it can connect to AWS to create the instance (virtual machines) on AWS EC2. EC2 is one of the many services which gives compute to run the virtual machine(s)

AWS developed EC2 services in a generic way so that it can be used anywhere in the world to create the machine. So many features of EC2 increase complexity for nontechnical users.

This project is a tailor made solution for the companies which only use EC2 service to create a virtual machine. As this interface only have the relevant features hence it make easy for the nontechnical user to work on.

V. STEPS TO LAUNCH INSTANCE IN THE CUSTOM INTERFACE

A. Step 1

The following image (Fig. 5) is the reference image to launch EC2 of the interface and we can select different types of fields to launch instances i.e. Region, Availability zone, Operating System, Keys, Machine configuration.



Fig. 5: Snapshot of the interface.

B. Step 2: If we want to select the region then Fig. 6 is the reference image:



Fig. 6: Selecting of Region from the interface.

C. Step 3: Select Availability zone of the region:

After Step1 and Step2, select the availability zone of the region as shown in Fig. 7.

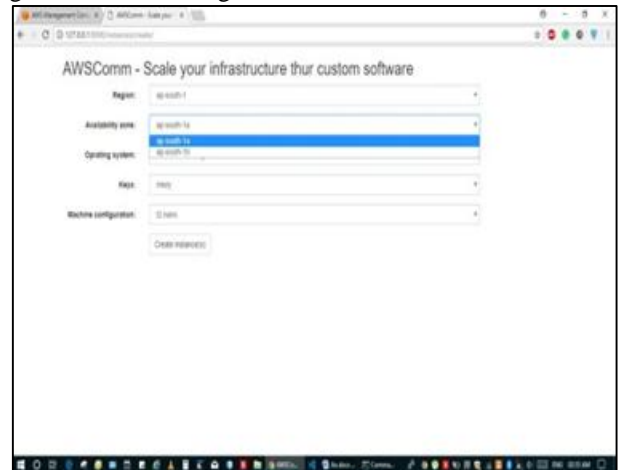


Fig. 7: Selecting availability zone of the region.

D. Step 4: Select Operating System:

In this Step, select the Operating System such as Windows, Linux, Ubuntu (Fig. 8).

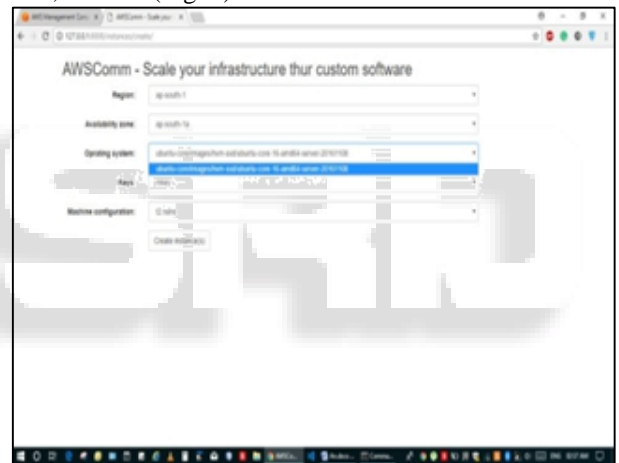


Fig. 8: Snapshot of selecting Operating System.

E. Step 5: Select key:

Amazon EC2 uses public and private keys which are known as a key pair. The public-key cryptography is used to encrypt and decrypt login information. Public-key cryptography uses a public key to encrypt a piece of data, such as a password and the recipient use the private key to decrypt the data.

To log in to the instance, we must create a key pair, specify the name of the key pair when it launch the instance and provide the private key when we connect to the instance.

SSH (Secure Shell) is a cryptographic network protocol which allows users to securely perform a number of network services over an unsecured network. SSH uses key pairs and can be generated with or without a password for the key. Password keys are more secure than no password keys. It can create multiple key pairs as shown in Fig. 9. The keys which we created are mkey, new and mymnewprivatekey.



Fig. 9: Snapshot of selecting Key.

F. Step 6: Machine Configuration:

Amazon EC2 provides a wide selection of instance types optimized to fit different use cases. They have varying combinations of CPU, memory, storage, and networking capacity. In this Step, it can select an appropriate mix of resources for the applications.

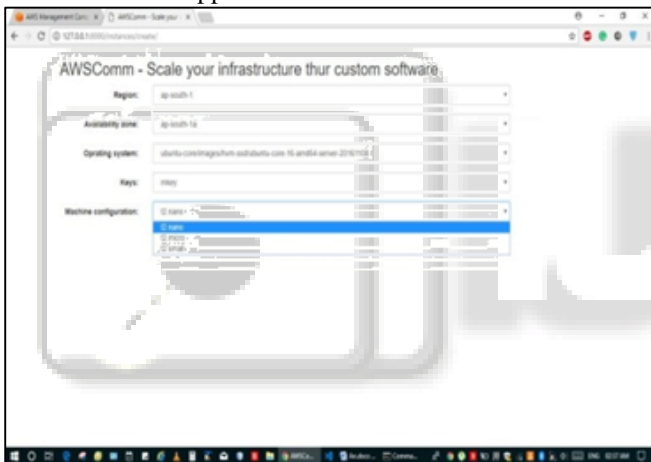


Fig. 10: Snapshot of selecting Machine configuration.

There are many instance types of EC2; some are t2.nano, t2.medium, t2.large as shown in Fig.10. The t2.nano is the lowest priced Amazon EC2 instance. Each t2.nano instance has 512 MiB of memory and 1 vCPU and can run 32 or 64 bit operating systems and applications. They support EBS encryption and up to two Elastic Network Interfaces per instance.

G. Step 7: Launch instance (RESULTS):

After complete selecting resources of what we need, now it can launch instance(s) just by click on the create instance(s) button in the interface. When an instance is launch, a congratulation message will come out in a Message Box as shown in Fig. 11.

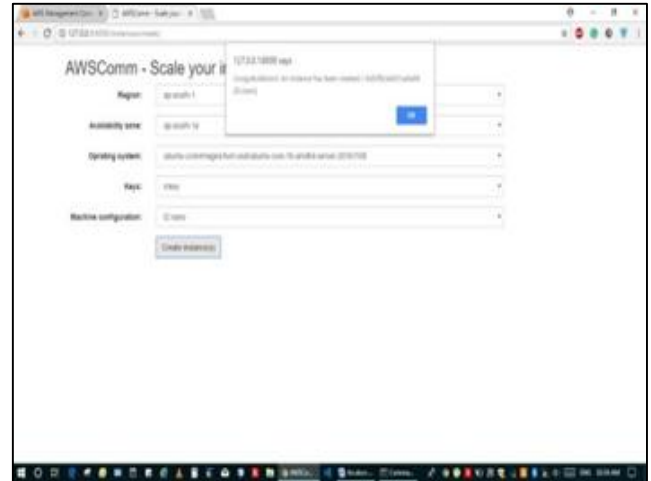


Fig. 11: Snapshot of Launching instance(s): an instance has been created.

Also, it can be seen in the AWS EC2 service that the successful in creating one instance shown in Fig. 12. Figure 12 Snapshot of launch instance in AWS EC2 service that is created from the interface.

VI. BENEFITS OF THE CUSTOM INTERFACE

A. Usability

The interface is very simple and attractive to use.

B. Flexibility

It enables to select the operating system, CPU, web application platform, memory storage, and other resources of EC2 that we need.

Allowed to use only a particular EC2 service without associating with other services.

C. Availability

It can be accessed anywhere and anytime. It is accessible for cloud services as a whole. The AWS Cloud spans 54 Availability Zones within 18 geographic Regions and 1 Local Region around the world, with announced plans for 12 more Availability Zones and four more Regions in Bahrain, Hong Kong SAR, Sweden, and a second AWS GovCloud Region in the US.



Fig. 13: Global Infrastructure

D. Auto Scalability

It has the capacity to get changed in size or scale and according to varying demand.

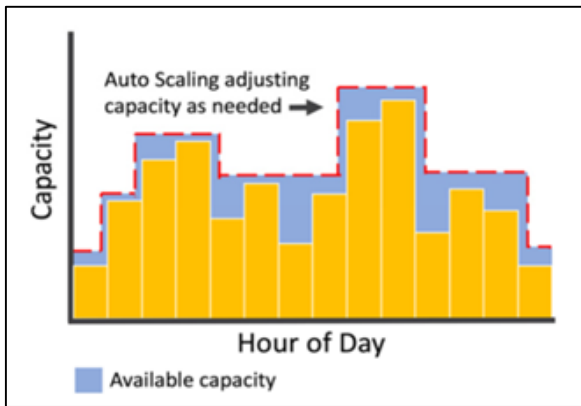


Fig. 14: shows the capability of scaling up and down whenever we change.

E. Increase Productivity

In past, most of the time we were spent time on an installation of software, maintenance, and backup and others IT problems. Now we don't need to install any software on our system and we don't need to set up the virtual private network manually. We can directly use without installing any software.

F. Security

Security is an important parameter in the cloud environment. As we work on AWS public cloud, the AWS infrastructure puts strong safeguards in place to help protect customer privacy. All data is stored in highly secure AWS data centers.

G. Maintenance

The interface is easier to maintain since they don't have to be installed on each user's computer. They are easier to support and to improve since the changes reach the clients instantly.

H. Complex Decrease

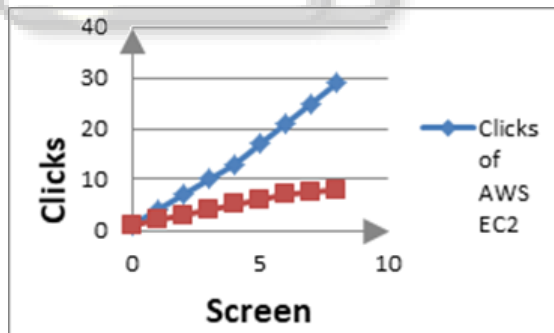


Fig. 15: it shows the number of clicks reduces while working on the interface than working on the AWS EC2.

To access the services of AWS we need to follow complex steps which is difficult for the user. So by creating the user-friendly custom interface, the number of steps has been decreased as shown in Fig. 15.

I. Disaster recovery

Cloud storage is a better, faster and secure method. Cloud storage is a method that can back up the data and information. Even stored data in the cloud is not only on a single server but it is multi-servers as shown in Fig 16. So don't need to worry about the data security.

Cloud disaster recovery is a backup and strategy that involves storing and maintaining copies of electronic records in a cloud computing environment as a measure.

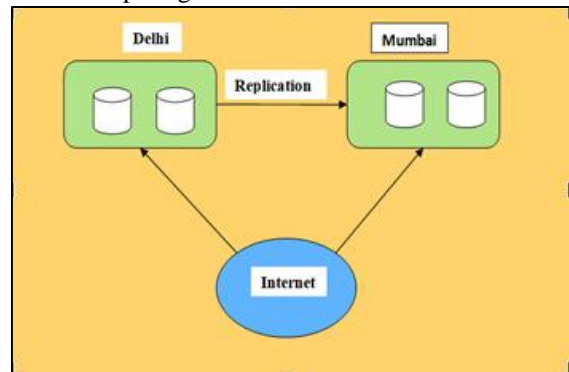


Fig. 16: shows multi-servers in different areas i.e. Delhi and Mumbai which they have same data copy.

VII. FUTURE SCOPE AND CONCLUSION

We conclude that EC2 has so many features which makes difficult to launch instance(s) for non-technical users. The interface in the project has only appropriate features to the users need. Because this interface has limited features hence non-technical users can also launch instances(s) without any hassle.

There are so many companies which are only using AWS Public cloud for the virtual machines. To create machines, they give access to the AWS account which is not very secure. This interface resolves the security issue as this will act as the communication between AWS and Non-technical user.

REFERENCES

- [1] J. Preethi; N. Aswathy, "A Survey on Cloud Applications" International Journal of Scientific Research in Computer Science, Engineering and Information Technology, IJSRCSEIT, Volume 2 Issue 2 ,pp.10-17, March-April-2017.
- [2] Sagar Sapkota; Khawar Shehzad, "Cloud Computing for telecom Systems", Electrical Engineering, School of Engineering Blekinge Institute of Technology 371 79 Karlskrona, Sweden, September 2011.
- [3] Nitesh Kumar Namdeo; Sachin Choudhari, "Privacy Preservation and Auditing in Public Cloud: A Review", International Journal of Computer Sciences and Engineering, Volume-4, Issue-3, pp. 124-127, March 31 2016.
- [4] V.D. Ambeth Kumar; V.D. Ashok Kumar; H. Divakar; R. Gokul, "Cloud enabled media streaming using Amazon Web Services", IEEE International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), Date of Conference: 2-4 Aug. 2017, Chennai, India, 2017.
- [5] B. Sai Karthik; M. Nalini Sri, "Cloud Computing Services and Applications", International Journal of Advanced Research in Computer Science and Software Engineering Research, ISSN: 2277 128X, Volume 3, Issue 4, April 2013 pp. 963-967, 2013.

- [6] Saakshi Narula; Arushi Jain, Prachi, “Cloud Computing Security: Amazon Web Service“, IEEE Fifth International Conference on Advanced Computing & Communication Technologies, IEEE, Date of Conference: 21-22 Feb. 2015, Date Added to IEEE Xplore: 06 April 2015, pp.501- 505, Haryana, India, 2015.
- [7] Amazon Web Services, “Amazon Web Services: Overview of Security Processes, “pp.1- 75, August 2016.
- [8] Marco Balduzzi; Jonas Zaddach ;Davide Balzarotti ; Engin ; Sergio Loureiro SecludIT;. “A Security Analysis of Amazon’s Elastic Compute Cloud -Service – Long Version”,
- [9] Keith R. Jackson; Lavanya Ramakrishnan; Krishna Muriki; Shane Canon; Shreyas Cholia; John Shalf; Harvey J. Wasserman; Nicholas J. Wright, “Performance Analysis of High Performance” , Conference: Cloud Computing, Second International Conference, CloudCom 2010, November 30 - December 3, 2010, Indianapolis, Indiana, USA,2010.
- [10]Balaji Balakrishnan; Rajat Ravinder Varuni, “Cloud Security Monitoring”, SANS Institute InfoSec Reading Room, Mar 8th, 2017.
- [11]G. Lindstrom, “Programming with Python”, IEEE Computer Society, published in IT Professional IT Volume: 7, Issue: 5, Sept.-Oct. 2005, pp.10 – 16, 10 October 2005
- [12]Dave Kuhlman, “A Python Book: Beginning Python, Advanced Python, and Python Exercises”, April 22, 2012.
- [13]Moore; Jonathan Ian,” Building a reusable application with Django”, Laurea University of Applied Sciences, Business Information Technology, Thesis, July 2016.