Lock the Doc

### Introduction

University Name:

<http://www.sjsu.edu/>

Course: [Cloud Technologies](http://info.sjsu.edu/web-dbgen/catalog/courses/CMPE281.html)

Professor [Sanjay Garje](https://www.linkedin.com/in/sanjaygarje/)

ISA: [Anushri Srinath Aithal](https://www.linkedin.com/in/anushri-aithal/)

Student : [Saylee Vyawahare](https://www.linkedin.com/in/saylee-vyawahare)

Demo

[Lock The Doc](https://youtu.be/8lPc9LzJvHs)

### Lock the Doc Introduction

Lock the Doc is a web application hosted on AWS cloud. The app helps to provide authorized users a portal to securely manage their docs/files on cloud. The project illustrates the use of various AWS cloud components in developing a 3 Tier Web Application. The application manages various components to provide a highly available, scalable, cost effective solution to securely back up data on to Amazon S3. The application leverages AWS auto scaling functionality to provide seamless experience during peak load times. The application also monitors the health of the EC2 associated with auto scale group using Cloud Watch, AWS Lambda and SNS

Application provides with two views one for admin and other for users.

Admin View

Admin can perform following activities: -

1. View all files or docs of all users from S3
2. Delete the files for all users from S3

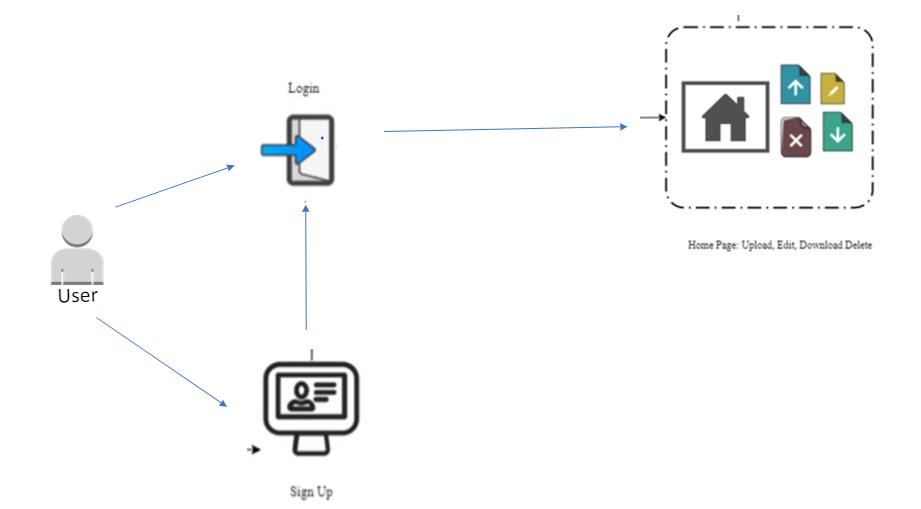
User View

User can perform following activities: -

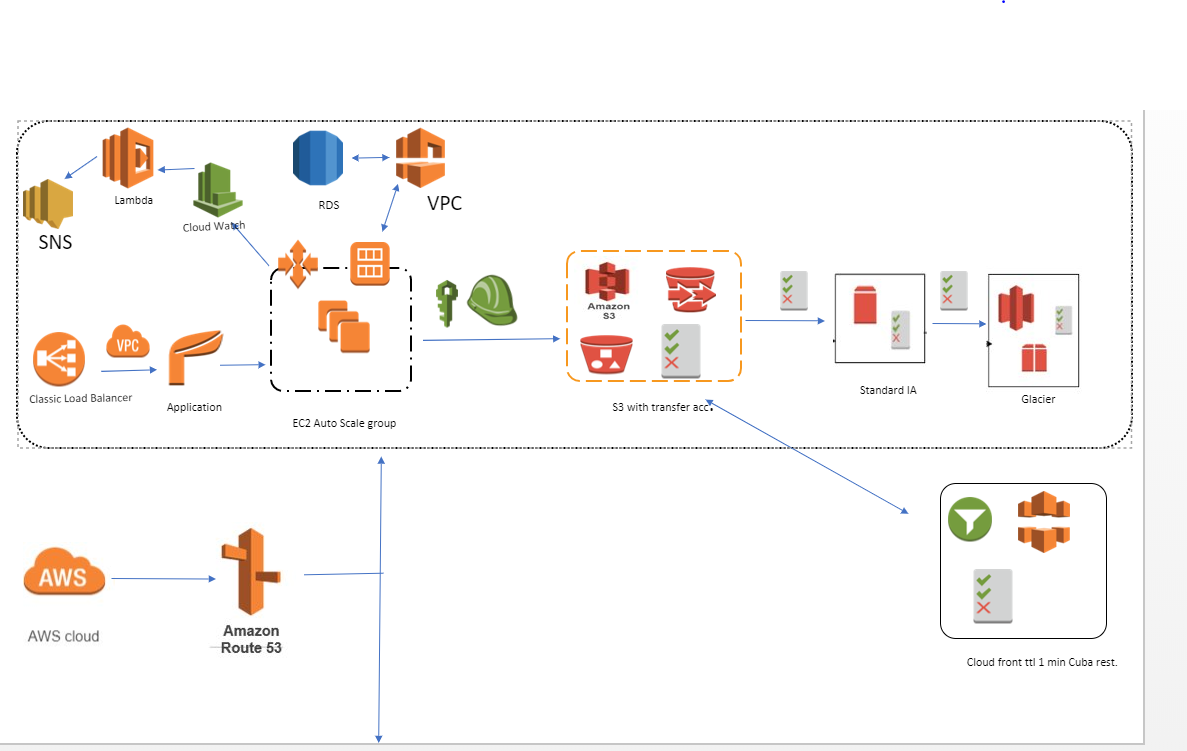
1. Upload file to S3
2. Download files from S3 via Cloud Front
3. Delete files on S3
4. Edit File

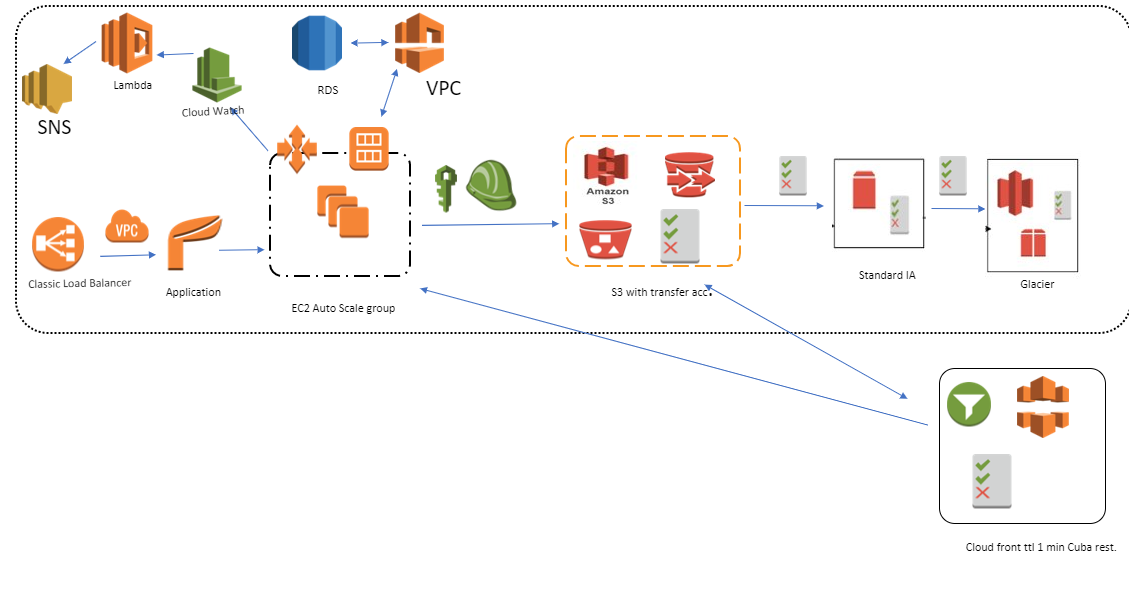
All these operations are handled securely through AWS VPC and S3 Transfer Acceleration. Appropriate I Am User roles are implemented to ensure secure cloud experience.

The general workflow to upload, edit, download, delete file is as below



Architecture Diagram





### Feature List

1. Sign up form for new user to create an account. A new user record is created in AWS RDS MySQL. If an already existing user tries to sign up, he is prevented in doing the same.
2. Login Page to allow only authorized users to login. Performs validation for username and password match.
3. User can log in google log in also
4. Allow authorized users to upload file on AWS S3. The file to be uploaded cannot exceed a file size of 10MB.
5. Allow authorized users to download file from S3 via Cloud Front they have previously uploaded
6. Allow users to delete files.

7. Allow authorized users to update their existing file. Update works as download the file, make changes and reupload the file without changing the file name

8.Admin can log in view all files from all users and delete them

### AWS Components Leveraged

1. EC2: Create an EC2 instance with all project artifacts and use that to obtain an AMI for the AutoScale Group deployment.
2. AutoScaling Group: To achieve high-available and scalable solution configure auto scale group with a desired instance of 1 and max instance of 2. This is configurable based on requirement and traffic the website attracts.
3. Classic Load Balancer: Load Balancer point to autoscale group so that it handles optimal load on all the EC2 instances associated with the group.
4. S3: Used to upload and maintain user files.
5. S3 Transfer Acceleration: S3 bucket is enabled with Transfer Acceleration to enable faster and secure transfer of files to S3.
6. Standard Infrequent Access (IA): Lifecycle policies are updated on S3 to move files to IA after 75 days.
7. Amazon Glacier: Lifecycle policies are enabled on S3 bucket to move files to Amazon Glacier after 365 days.
8. CloudFront: File download is done using CloudFront. The minimum TTL for CloudFront is setup as 1minute.
9. RDS: MySQL instance is created to maintain user data and file metadata.
10. CloudWatch: Cloud watch is used to monitor any additional spin up or termination of EC2 in the autoscale group. It triggers Lambda when such a event occurs. Cloudwatch alarms are configured to send notifications via SNS on S3 and ELB healthchecks.
11. SNS: Configured to send email to all the subscribers for the topic.
12. Route53: Domain Name Server that resolves IP address for the application domain www.saylee-sjsu.info

### Deployment Instructions

1. Prerequisite Softwares: Java8 Runtime, Maven, Apache Tomcat 8.5, Eclipse, MySQL 5.7.
2. Download code from Github.
3. Run DBScript.sql on MySQL instance.
4. Update applications.properties with MySQL URL, Username and Password.
5. Update application.properties with S3 access key and secret.
6. Deploy the cloud.jar folder in Tomcat webapps folder.
7. Run the startup.sh/startup.bat in bin folder
8. You can now access the application at <http://localhost:8080/>

Snapshots

