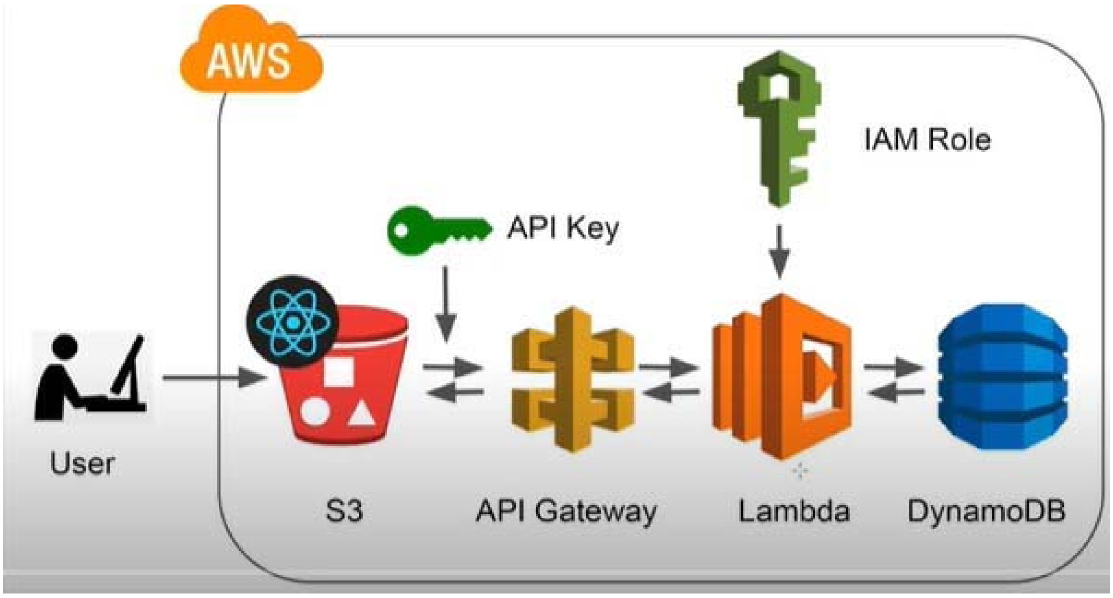
Review - 2

Name: Saptajit Banerjee
Registration Number: 20BCE1513

Abstract

In this project, an authentication system will be built on AWS from scratch. This project will use cloud computing concepts. The back-end is a node that uses node.js serverless API with DynamoDB for information storage. The serverless API will use the serverless computing concept which will make the system faster and more secure. It uses bcrypt to encrypt passwords and uses jwt to generate tokens. The front-end will be built using React.js and will be hosted on the AWS S3 bucket.
The system will authorize and register users so that only the registered users can access the premium content which is secured by the system. The non-registered users cannot view the premium content but they can make an account in the system to access it.



Introduction

A system that transmits information wirelessly is vulnerable to cyber attacks like ‘sniffing’, and ‘man in the middle’ which will compromise the security of the system as well as the data of the system’s users. To protect the system from such cyber attacks, the information transmitted needs to be encrypted using a private key which only the user’s device and the system will have access to. To further reduce the chances of cyber attacks, the session of a user’s account must be activated when the user has logged in and the session must log the user out of the system after some time when no user activity is detected. Thus, these cyber security strategies make the system more secure and enable the system to protect the privacy of its users. The maintenance and acquisition of the data, the servers, and the security of the system require a lot of money and a highly skilled workforce. To make the construction of secure online infrastructure/systems cheap and reduce the need for a skilled workforce, making the system on a cloud platform is the best choice. Cloud computing platform providers like AWS, Microsoft Azure, etc maintain the servers to deploy the systems as well as provide security to the deployed systems which saves a lot of money and makes the construction of secure online systems possible at low costs without any compromise on quality. The systems deployed on the cloud use serverless APIs which makes the system, even more, faster and increases the overall efficiency of the system. The project discussed in this paper is deployed on AWS Cloud. The project uses bcrypt.js to encrypt the passwords of the users which is transmitted and jwt is used to generate tokens. The cloud services like DynamoDB, Lambda, API Gateway, IAM, and S3 cloud storage have been used. DynamoDB has been used to store the data of the users of the project. Lambda contains the codes according to which the serverless API should operate. API Gateway is used to construct the API by creating 'methods' and 'resources' for every 'method' as well as setting the private key for the API which will be used to encrypt the data transmitted to the user and decrypt data received from the user. IAM is used to create the roles by selecting certain policies suitable to manage the AWS resources for the project. S3 is used to house/contain the frontend part of the system thus making the system available to anyone with the link to the S3 bucket. Any information transmitted from both the user's side and the server's side is always encrypted.

The link to the project:
http://jcomponent20bce1513.s3-website.ap-south-1.amazonaws.com/login
(If copying and posting the link in the browser is not opening the website then, kindly, type the link in the browser to access)

Literature Review

- **Analysis of Web Authentication Methods Using Amazon Web Services**

Single Sign on is a session which allows user to be authenticated using only one set of login credentials. This paper analyses the various authentication methods that can be used to ensure security of the same. The methods, HTTP basic authentication and OTP based authentication are implemented and compared to get a consolidated result, which will help in guiding the usage of them in different scenarios. These are deployed and tested in a Virtual Private Network using Amazon AWS. This paper also compares the features of Amazon AWS over OpenStack. With respect to time, HTTP Basic authentication proves to be almost twice as fast as OTP based authentication. This application is tested in Java and hosted in Virtual Private Network with a set of subnets using Amazon Web Service (AWS). Here, AWS is used, instead of software like OpenStack, since testing and deployment of applications is more easier comparatively.

Link:
https://www.researchgate.net/publication/328765590_Analysis_of_Web_Authentication_Methods_Using_Amazon_Web_Services

- **REST APIs Using Amazon AWS API Gateway**

APIs (Application Programming Interfaces) are becoming increasingly important for businesses in a variety of industries, as they provide the way by which two different applications may connect. APIs empower organizations to develop their organizations more rapidly than any time recently. APIs offer organizations the chance to scale, cultivate development and contact a more extensive crowd.In future APIs have potential to transform businesses as it makes ease integration of backend data and applications.Cloud Computer is the most common computing model nowadays, and most ISPs (Internet Service Providers) have launched cloud offerings.REST interfaces are commonly used to expose cloud services on the web, but also do not provide a formal grounding that can be used to semantically characterize the accessible services.In this paper, this paper presents an analysis of amazon aws api gateway and general methodology for developing RESTful api using aws api gateway

Link: https://www.jetir.org/view?paper=JETIR2107029

- **Cloud Computing and Security Issues—A Review of Amazon Web Services**

Nowadays, new business jargon has witnessed an immense success by deploying their services and data on web without depending on any of the physical maneuver. This independence and trend have driven many renowned companies such as Netflix, Salesforce and Amazon towards cloud-based infrastructure Amazon web services (AWS) overshadows the market for offering cloud-based services with top metrics like huge volume, flexibility, availability and large number of customers. However, in addition to several benefits offered by the cloud infrastructure of Amazon (AWS), cloud security remains as the major point of concern for Amazon. In this paper, some of the common security concerns faced by a common cloud infrastructure have been described. But this paper focuses on the security vulnerabilities of Amazon Web Services, which are proved to be a kind of barrier for widespread use of Amazon Web Services (AWS). The world congress on Internet Security survey in 2013 indicated that there is always going to be a high demand for products providing security management. Since AWS gives its clients full control of an instance, it can be concluded that security is not only the responsibility of the cloud provider, but also of the client. Human beings are the weakest link, as they say. Another possible solution to the above-mentioned security issues can be that AWS should not allow services and clients to share account login information with each other.

Link: https://www.ripublication.com/ijaer18/ijaerv13n22_87.pdf

- **Security Pattern for Cloud SaaS: From System and Data Security to Privacy Case Study in AWS and Azure**

This is because the Cloud has many advantages over the traditional private infrastructure, such as increased flexibility, no maintenance, less management burden, easy access and easy to share information. However, there are many concerns around issues like system security, communication security, data security, privacy, latency and availability. In addition, when designing and developing Cloud

SaaS application, these security issues need to be addressed in order to ensure regulatory compliance, security and trusted environment for Cloud SaaS users.This paper identifies identify the Cloud SaaS security patterns for different security aspects, from data and system security to privacy. This paper provides a complete list of patterns and their solutions applicable to Cloud computing environment, and to the best of our knowledge, there is no attempt so far to fully study and document them. In addition, this paper produces an official security best practices and security knowledge documentation that SaaS developer can use as a guideline for developing Cloud SaaS application from the ground up. Another contribution that also distinguishes this paper is the study of AWS and Azure security solutions. The main goal of this paper is to map each pattern to the solutions available in AWS and Azure.

Link: https://www.mdpi.com/2073-431X/8/2/34/htm

- **Analysis of Cloud Security Controls in AWS, Azure, and Google Cloud**

This research paper aims to identify the gaps in information related to mapping the cloud security alliance top 20 critical controls against cloud security services provided by the major cloud providers. This paper will be reviewing the security controls against the cloud security applications and services provided by major cloud providers. Most organizations are adopting the cloud for their business-critical applications. Organizations need to be compliant with various frameworks relevant to their industries. Along with cloud security controls, organizations also need to perform an audit that measures the organization's security policies to maintain compliance. Although a vast amount of information on cloud security is available, we still hear about cloud systems attacks. This paper focuses on providing baseline information on cloud security controls published by Cloud Security Alliance (Cloud Security Alliance, 2019) and map them to cloud services. Information technology professionals need to review the cloud security measures in AWS, Google Cloud, Azure against Cloud Security Alliance top 20 controls, which will help the cloud user make an informed decision. This paper assists as a decision support document for the cloud user who wants to understand the role of security controls in a cloud environment and address the cloud security risks. Cloud users, cloud architects, and cloud consumers will understand how various cloud providers offer tools that assist in maintaining the security controls. This research paper provides the base layer information and aims to help future research in cloud security controls.

Link:
https://repository.stcloudstate.edu/cgi/viewcontent.cgi?article=1149&context=msia_etds

- **Three factor authentication system with modified ECC based secured data transfer: untrusted cloud environment**

Cloud computing (CC) is a technology that delivers its service by means of the internet. In the modern scenario, cloud storage services have gained attention. The cloud environment confronts data breaches expansively in cloud storage, which might bring about the disclosure of personal in addition to corporate data. Thus, the requirement arises for the creation of a more foremost authentication system. Customary authentication schemes depend on techniques, like Password Authentications Protocol (PAP), Challenge Handshakes Authentication Protocols (CHAP), as well as One-Time Pads (OTP), which are often susceptible to malevolent attacks as well as security threats. To shun such issues, this paper proposed a Modified ECC centred secure data transfer and a '3'-factor authentication scheme in the untrusted cloud environment. The proposed work comprises '3' steps: authentication, data compression, and safe data transfer. In the authentication phase, the SHA-512 algorithm along with CCP is utilized. After that, the user-uploaded data is compressed utilizing CHA on the server-side. Next, MECC encrypts the compressed data, and then, safely uploaded it to the cloud server (CS). In the investigational appraisal, the proposed work is contrasted with the prevailing methods. The outcomes proved that the proposed work renders better security than the prevailing methods.

Link: https://link.springer.com/article/10.1007/s40747-021-00305-0

- **On the Network Performance of Amazon S3 Cloud-Storage Service**

The advances in networking technologies and the increase in the need for storage resources have prompted many companies to outsource their storage needs. Cloud-storage providers offer clean and simple file-system interfaces, abstracting away the complexities of direct hardware management. At the same time, however, such services eliminate the direct oversight of performance that final users with high service-level requirements traditionally expect. While several works in literature have addressed security-related issues (such as privacy,integrity, availability, etc.) few of them have targeted the network performance of this kind of services.
This paper proposes the analysis of the performance of the network associated to the storage service offered by Amazon: S3. Thanks to a large-scale distributed campaign performed by leveraging the Bismark measurement platform, the publishers of this paper have characterized how the performance of the network may impact the quality of service experienced by final users on the basis of their location and the configuration of services. The publishers found how performance heavily changes (up to 1553 KiB/s) according to the location of the customers and the cloud region they rely on (up to 2117 KiB/s), also deriving a number of usage guidelines for the customers. In addition we characterize the impact of leveraging the Amazon CDN service to distribute contents, finding that while it guarantees up to a 275-percent performance improvement, cases exist for which additional costs may lead to worse performance.

Link:
https://www.researchgate.net/publication/311530769_On_the_Network_Performance_of_Amazon_S3_Cloud-Storage_Service

- **Construct a Serverless Web Application with AWS Lambda, Amazon API Gateway, AWS Amplify, Amazon DynamoDB, and Amazon Cognito**

Serverless is a new innovation that empowers organizations to lessen the overhead for provisioning, scaling and overall dealing with the framework. Organizations are progressively receiving Serverless, by moving existing applications to this new worldview. Various specialists proposed designs for making and overseeing serverless capacities. Nonetheless, a portion of these examples offer various answers for take care of a similar issue, which makes it difficult to choose the most appropriate answer for every issue. [Goal] In this work, the paper has targeted supporting specialists in understanding the various examples, by grouping them and revealing potential advantages and issues. [Method] A multivocal writing audit measure, looking over peer-inspected and dark writing and grouping designs (basic answers for take care of basic issues) was received , along with advantages and issues.

Link: https://ijirt.org/Article?manuscript=151299

- **Implementation Of chat bot using AWS and GUPSHUP API**

A chatbot can be defined as a program developed to carry out conversations with a human using either audio or text. There exist numerous chatbots which are used for various purposes such as e-commerce, customer support, design, communication, finance, education, analytics, and so on. Furthermore, many companies use chatbots for their internal operations, for human resources, for customer support and more recently, support for Internet-of-Things (IoT) operations has also been added. Bearing in mind the existing chatbot applications with respect to productivity, the aim is to develop a chatbot for various operations related to productivity and project analysis within an organization, such that it can be integrated with CA Technologies Rally (Agile Central). It can be used for checking tasks and defects, generating reports and obtaining notifications. In the proposed work, the chatbot is built using Gupshup Bot Builder API which deploys it on to Amazon Web Services (AWS) Cloud, and then, it is integrated with Rally. Natural language processing (NLP) is used by the chatbot in general command interactions with the user, thereby eliminating the need for a fixed database of interaction commands.

Link: https://journal.scsa.ge/papers/implementation-of-chatbot-using-aws-and-gupshup-api/

- **Build A Serverless Website Using AWS Cloud**

AWS provides different services that allows to build full application or a website stacks without managing any server. Serverless denotes to an application framework for building application or a website without servers. The server is managed by cloud provider and takes care of its allocation. This makes application or a website to run in a stateless compute container. In this paper discusses a project which is a simple serverless website that helps user to request the feedback form for college survey. Serverless is a process which shows services, practices and strategies which is used to build a website so as to innovate and develop for faster changes. Serverless computing contains infrastructure management tasks as capacity provisioning and patching. Developers can easily focus on developing code that serves the customer. serverless services like AWS lambda has a quality of auto scaling and pay as per use billing model. Serverless infrastructure does not require any servers. Developers can focus on core product and a business logic. Serverless applications or website doesn't require you to manage any different servers.

Link: https://ijcrt.org/papers/IJCRT2107116.pdf

- **Use of AWS Lambda for Building a Serverless Chat Application**

Out of various cloud services and techniques, serverless computing represents an evolution of cloud based programming technologies, and is an attestation to the growth and large scale adoption of cloud concepts. In this era of increasing technological advancements large internet companies like Amazon, Netflix, and LinkedIn deploy big multistage applications in the cloud which can be developed, tested, deployed, scaled, operated and upgraded independently. However, aside from gaining agility and scalability, infrastructure costs are a major hindrance for companies following this pattern due to the increasing server loads and the need to increase server space. This is where serveless computing and services like AWS Lambda comes into picture. The paper delves into the functioning of AWS Lambda along with other existing AWS services through the development of a serverless chat application which supports scalability without the addition of new servers. This paper further explores the connection of different existing services in AWS like S3, DynamoDB, CloudWatch etc. with serverless technologies like Lambda. This paper has proposed the use of AWS Lambda for serverless Applications and merits of serverless computations. The research is supported by a case study of a serverless chat application built using AWS Lambda and other AWS cloud services. The architecture of the messenger application requires very less management and is cost efficient. Implementing the messenger application using the above mentioned technologies helped in grasping the usefulness and relevance of serverless computing and 'FaaS' in hosting dynamic applications with lots of user interaction. The future work of this messenger application includes creation of personalized groups and advanced encryption methods to enhance cyber security. Since Lambda cost model is based on the time required for the code execution, future scope also includes cost and pricing optimizations. Use of other NoSQL databases instead of DynamoDB may be tested in the future implementations of the application. Cross platform integration of AWS Lambda with other prominent serverless technology providers like Google Cloud and Microsoft Azure Functions, etc. Serverless computing with the help of AWS and other similar cloud services and their decreasing costs for the resources is hereto stay for a long time changing the way we develop, test, deploy and maintain the applications.

Link:
https://www.researchgate.net/publication/338391132_Case_Study_Use_of_AWS_Lambda_for_Building_a_Serverless_Chat_Application

References:

- https://www.researchgate.net/publication/328765590_Analysis_of_Web_Authentication_Methods_Using_Amazon_Web_Services
- https://www.jetir.org/view?paper=JETIR2107029
- https://www.ripublication.com/ijaer18/ijaerv13n22_87.pdf
- https://www.mdpi.com/2073-431X/8/2/34/htm
- https://repository.stcloudstate.edu/cgi/viewcontent.cgi?article=1149&context=msia_etds
- https://link.springer.com/article/10.1007/s40747-021-00305-0
- https://www.researchgate.net/publication/311530769_On_the_Network_Performance_of_Amazon_S3_Cloud-Storage_Service
- https://ijirt.org/Article?manuscript=151299
- https://journal.scsa.ge/papers/implementation-of-chatbot-using-aws-and-gupshup-api/
- https://ijcrt.org/papers/IJCRT2107116.pdf
- https://www.researchgate.net/publication/338391132_Case_Study_Use_of_AWS_Lambda_for_Building_a_Serverless_Chat_Application