International Journal of Research in Engineering, Science and Management
Volume 5, Issue 3, March 2022
https://www.ijresm.com | ISSN (Online): 2581-5792

Cloud Storage and Sharing Service

Tanmay Anand[1*], Abhinav Singh[2], Amansingh Panwar[3]

***Abstract*: The idea of this project is to create a web application that sends files from one email to another email using the SMTP protocol, which is handled in a server-based application. The main advantage of the project in this paper is that it provides a safe, reliable, and excellent tool for sharing files in any format. Also, it has infinite scaling capabilities. With a bit of tweak in the code, it can be scaled to handle heavy file loads.**

***Keywords*: Cloud computing, cloud storage, file sharing, web application, SMTP.**

1. Introduction

In recent years especially when humankind is stuck in a pandemic, Online working has established a new place in human history which helps millions of people to do their jobs from remote locations, so in modern times like these, file sharing over the internet is increasing day by day. The data owner loses control of the data, which is why a new security risk arises regarding data security and integrity. CSSS, an abbreviation of our project that is “Cloud storage and sharing service,” helps remove this particular difficulty from the lives of millions of users who tend to use different cloud platforms, which tend to compromise the security of their data while sharing it. CSSS is a cloud-based application that can be used to send files of any size and extensions from one user to another user using SMTP protocol which is handled in an application hosted over a cloud platform, i.e., AWS Beanstalk(for Production) and Heroku(For Development).

2. Studies and Findings

In cloud computing, shared resource are provided over the Internet. Various cloud storage threats are Data Leakage, Snooping, Data Loss, Business Risks in Shared Technology and Key Management. In data leakage Resources are shared in cloud, a multi-tenant environment which provides access to a customers data. Sharing storage hardware and migrating private or confidential data in the cloud seems to be risky. There are number of threats which leads to data leakage, including unauthorized access of cloud user accounts or hacks of cloud providers. The tenant cannot thrust the cloud service provider with their data; the best strategy is to depend on stronger passwords and file encryption. The length of the key used to protect data in cloud is conventionally co- related to the time required to break down an encryption algorithm.In Snooping, files without security measures in the cloud are most susceptible of being hacked. Even if the cloud service provides encryption for files, on route to its destination data can still be cut-off. Security against this threat would is to ensure that the data is encrypted and transmitted over a secure connection, as it will prevent unauthorized users from accessing the clouds data. In Data Loss, some of the cloud services like Microsoft Azure, Dropbox and Google Drive has become a part of various business processes it has to deal with new security issues such as loss of control over confidential data. Data loss can be costly for an enterprise. A lot of data that are not meant to be shared can end up being viewed by unauthorized user; user need to backup their data in real-time. The Business Risks in Shared Technology cloud computing such as Infrastructure, platforms, and applications are shared by cloud service providers. The entire environment of the system can be exposed by a single vulnerable activity. In Key Management, the management of cryptographic keys has become huge security issues after the introduction of the cloud. It can be done by securing the key management process by being automated, inconspicuous, and active.

3. Methodology

By looking at the limitations described in a review of texts, it is necessary to have a novel and a general process to overcome those limitations. As with almost every large organization web-based and works with a cloud or cloud environment, organizational knowledge becomes an important factor. Even after the cloud provides the industry with a variety of security as well as a program free from malicious software virus attacks, the data is not safe for malicious users having administrative rights. Someone with superuser rights can access data from cloud storage for the wrong purposes. After passing various solutions to reduce the attacks of malicious users found to store data against the user's will makes data more secure than any other method, however, we cannot forget the fact that we must keep the data safe. Therefore the proposed system is built to keep these conditions inside consideration we have built a system that every individual or a company can deploy easily on their own cloud platforms and share files easily without having any fear of data leak and losing control of data.

Send Grid is an emailing service developed by Twilio for sending emails. Instead of configuring your email server to send an email with your apps, we use SendGrid to do a lot of hard work for us. It also reduces the chances of email being spam as it is a well-known, trusted service. It is also straightforward to use libraries in various email forums. Node.js is one of the supported platforms.

*Corresponding author: agrawalatharva0777@gmail.com

POST template route creates our template with creative work. The second argument has our chosen status. Route the Delete template removes it by checking the entry with the ID to delete with the destruction function.

The sending route calls the SendGrid API to send an email with the variables specified in an email template filled with user-set values. The application body has a variable field to send variables and values, where the key is a variable word, and the matter has a value. Sequencing provides for creating, finding everything, updating and destroying tasks as part of a model.

Node (or additionally formally Node.js) is an open source, multi-platform that allows everyone to build all kinds of server-side tools and applications in TypeScript and JavaScript. Operating time is intended for use outside the context of the browser (i.e., working directly on a computer or OS server). Thus, the environment leaves the JavaScript APIs browser and adds support for traditional OS APIs including HTTP and file system libraries.From a web server development perspective Node has many advantages.

Excellent performance: Node is designed to improve performance and consistency in web applications and is a good solution to many common web development issues (e.g., real-time web applications).

The code is written with "old plain JavaScript", which means less time spent dealing with "content changes" between languages when typing both client side and server side code.

JavaScript is a new programming language and benefits from improved language structure compared to other common web server languages (e.g. Python, PHP, etc.) Many other new and popular languages integrate/translate into JavaScript so you can use TypeScript, CoffeeScript, ClojureScript, Scala, LiveScript, etc.

Node Package Manager (NPM) provides access to hundreds of thousands of reusable packages. It also has a high reliability and can be used to automate most of the series of building tools.

Node.js is portable. Available on Microsoft Windows, macOS, Linux, Solaris, FreeBSD, OpenBSD, WebOS, and NonStop OS. In addition, It has an ecosystem of a foreign company that works closely with the developer community, with many people willing to help.

You can use Node.js to build a simple web server using the Node HTTP package.

On a standard data-driven website, a web application awaits HTTP requests from a web browser (or another client). When the application is received the application works using any action required based on the URL pattern and information that may be associated with it contained in POST data or GET data. Depending on what is required they may read or write the information on the website or perform other functions required to process the request. The application will then return the response to a web browser, usually creating an HTML page for the browser to display by retrieving the retrieved data in the HTML template.

Express provides ways to specify which function is called a specific HTTP action (GET, POST, SET, etc.) and URL pattern ("Route"), and ways to specify which template engine ("view") is used, where template files available, and what template you can use to provide feedback. You can use Express middleware to add support for cookies, sessions, and users, get POST / GET parameters, etc. You can use any Node-based data storage method (Express does not define any site- related behavior).

We used all the above tools to create an experience which is safe and keeps the control of data in the application or the servers of the user only hence providing extra safety for sharing of files.

Whenever a user wants to send a file he can send it by firstly hitting the POST API which is used for uploading the file to the server and then the sending api which sends the uploaded file to the email sent in the payload by the user.

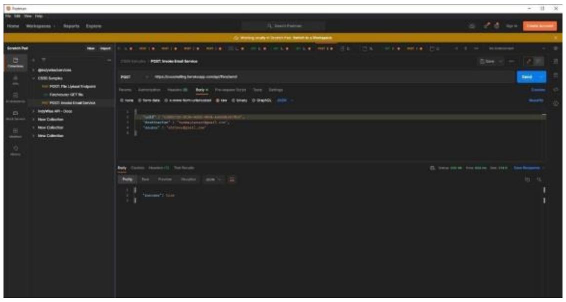

Fig. 1.

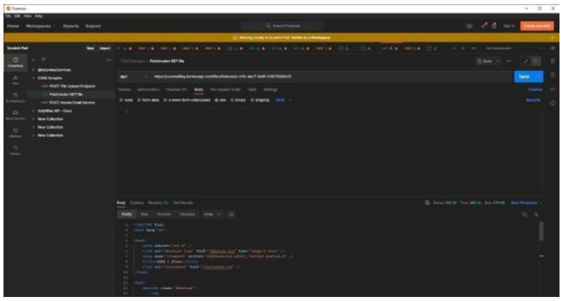

Fig. 2.

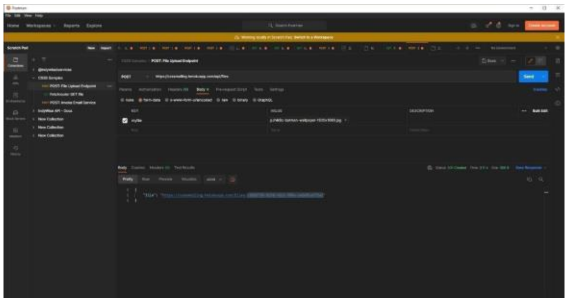

Fig. 3.

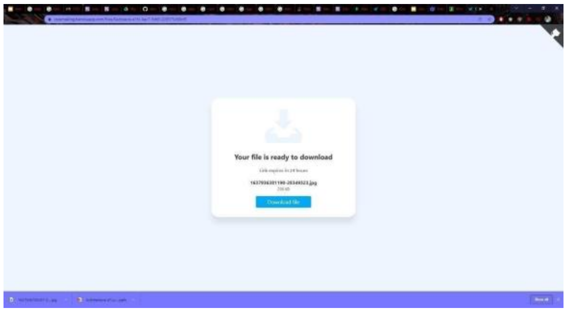

Fig. 4.

The above figures show the working of APIs from uploading a file to sending a file to an email address and the last screenshot shows how users will receive and download the file on their email.

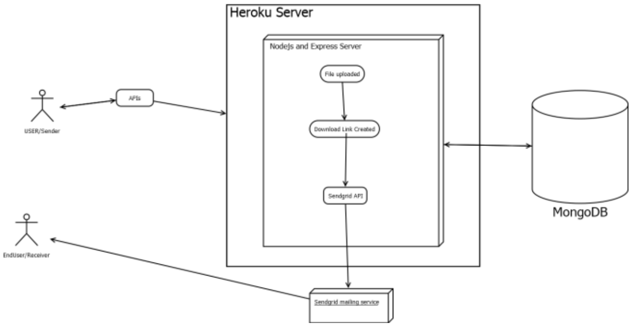

Fig. 5. UML diagram

4. Conclusion

The Cloud-based file sharing approach is proposed to provide the following services for external data confidentiality, secure data sharing within the group, protect data from unauthorized access of officials within the group and provide time and number of file access to users. Whenever information sharing among a bunch arises, the file owner sends the user uploads the file on the application and then shares it using the send API.This creates a safe medium of sharing of files and user in control of the data in the whole process of sharing the files.

References

[1] Archana K. Rajan, Surya Babu, "Privacy and Authenticity for Cloud Data using Attribute-Based Encryption and Digital Signature" 2017 Unpublished work.

[2] M. Ali et al., "SeDaSC: Secure Data Sharing in Clouds," in IEEE Systems Journal, vol. 11, no. 2, pp. 395-404, June 2017

[3] Pragathi Shetty, Rajkumari Sunanda, Sowmya Shree K. S, Swathi N. G, Hemanth Kumar N. P. "Cloud-based File Sharing", 2017.

[4] J. Wei, W. Liu and X. Hu, "Secure Data Sharing in Cloud Computing Using Revocable-Storage Identity-Based Encryption," in IEEE Transactions on Cloud Computing, vol. 6, no. 4, pp. 1136-1148, 1 Oct.-Dec. 2018.

[5] Hua Deng, Zheng Qin, Qianhong Wu, Zhenyu Guan, Robert H. Deng, Yujue Wang, Yunya Zhou, "Identity-Based Encryption Transformation for Flexible Sharing of Encrypted Data in Public Cloud", Information Forensics and Security IEEE Transactions on, vol. 15, pp. 3168-3180, 2020

[6] Q. Xu, C. Tan, Z. Fan, W. Zhu, Y. Xiao, and F. Cheng, ''Secure multi-authority data access control scheme in cloud storage system based on attribute-based signcryption,'' IEEE Access, vol. 6, pp. 34051–34074,2018.

[7] H. He, R. Li, X. Dong, and Z. Zhang, ''Secure, efficient and fine-grained data access control mechanism for P2P storage cloud,'' IEEE Trans. Cloud Comput., vol. 2, no. 4, pp. 471–484, Oct./Dec. 2014.

[8] P.-W. Chi and C.-L. Lei, ''Audit–free cloud storage via deniable attribute–based encryption,'' IEEE Trans. Cloud Comput., vol. 6, no. 2, pp. 414–427, Apr. 2018.