EasyChair Preprint

№ 8078

Cloud Based File Sharing Using BlockChain

Pabba Sumanth, Popuri Poojitha Chowdary, Ponnam Bharani, Thokala Gopal Krishna and Sriramulu Bojjagani

May 24, 2022

Cloud based file sharing using BlockChain

Pabba Sumanth
pabba_sumanth@srmap.edu.in

Popuri Poojitha
popuri_poojitha@srmap.edu.in

Ponnam Bharani
ponnam_bharani@srmap.edu.in

Thokala Gopal Krishna
thokala_gopal@srmap.edu.in

Sriramulu Bojjagani
sriramulubojjagani@gmail.com

Department of Computer Science and Engineering
Srm University AP

Abstract: Cloud computing is a relatively new technological advancement that has steadily increased market share over the last three years. In this paper, we will present a new methodology for securely sharing files via the cloud, combining block chain for secure transactions and viewing the shared files. Many individuals utilise file shredding software. Some people utilise these apps to make money by sharing their work in the form of files. We've all heard of Chegg, Scribd, and other similar services that provide a platform for students , researchers, freelancers to publish their work. However, in order to use these programmes, we must have a membership. Even if a user wants to access the files, he or she must pay a fee to the application rather than the author. So, in order to eliminate third parties, we devised a new approach known as "Cloud-based file sharing using Blockchain." By using our application users can easily share files with one another using the cloud-based file sharing method. As a result, this system enables users to store and share files via cloud networks in a simple and effective manner. Generally when we upload data to the cloud, we lose control of it, which introduces new security risks to the integrity and confidentiality of our data. So, in order to avoid this,In this paper, we will discuss a secure file sharing mechanism for the cloud that uses encryption. In this paper, we introduced a new method of file sharing. The transaction to view the files is carried out via a block chain from one user to the another.

Keywords: Cloud,Azure,BlockChain,Encryption

I. INTRODUCTION

With the advent of the internet, learning has become much more accessible in today's world. There are numerous applications that students, researchers, and working professionals use to share their work for money. However, all of these applications require a subscription. The files are directly uploaded into the application, with no security measures in place. If a user wants to see their work, they must subscribe to their application, and only they can view the files. In this process, the author of resources are paid less because they use third-party applications to share their work. To eliminate the need for third parties and sharing files securely we devised a new method known as a cloud-based file sharing system based on Block Chain. Using this method, users can easily and securely share their files.

The cloud-based file sharing system based on Block Chain makes use of the cloud platform to deliver solutions that are efficient and dependable. Cloud computing has changed the game in today's world. Cloud computing enables consumers and businesses to access applications without having to invest in hardware. Cloud computing can now perform any computing task, from creating a virtual image to providing artificial intelligence services. Cloud sharing is a relatively new technological advancement that has steadily gained market share over the last three years. The cloud based file sharing method allows users to easily share files with one another. As a result, this system gives users the ability to easily and effectively store and share files via cloud networks. We lose control of our data once we upload it to the cloud, which introduces new security risks to the integrity and confidentiality of our data. We discuss a secure file sharing mechanism for the cloud using encryption in this paper. We introduced a new way of sharing files in this paper. The transaction to view the files is done using a block chain from one user to another.

It can securely transfer files between multiple users while preventing intruders or unauthorised users. The most secure method of file sharing is to encrypt a file before sending it over the cloud. This is accomplished by employing an encryption algorithm.The vast majority of file sharing services or software implement secure file sharing by restricting file access, such as allowing only authorised personnel to access, view, and download data. As a result, we must ensure the confidentiality, integrity, availability, and validity of shared data in order to ensure its security.

This methodology allows users to upload any type of resource, such as documents, video lectures, and so on. This method can be used to encrypt any type of file. Our main goal is to eliminate the use of third parties when sharing files with author of the files. This can be accomplished using block chain technology. A distributed digital ledger, or blockchain, is a network of computer systems that stores transactional data. Furthermore, no one can attack or hack the network, and no one can alter data that has been entered.The user who wishes to use the resources can do so by transferring funds directly through the block chain. The author of the files can set their own price for the file during this process. The transaction is carried out on a peer-to-peer basis.

II. RELATED WORK

There are innumerable applications available for file sharing. All of these applications prioritise sharing over security and efficiency. These applications share files from user to user; therefore, in our project, we are looking for a centralised application in which all files can be viewed by any user who registers or uses the application. As a result, I'd like to highlight a few related applications:

For encryption, we used the aes algorithm in our application. There is a previous work that used AES to encrypt files. Aditya [1] They primarily focused on how files are encrypted and stored on disc using a secret key and then decrypted using the same secret key. This system utilizes the Advance Encryption Standard algorithm (AES).AES-128, AES-192, and AES-256 are the encryption key sizes (128 bits, 192 bits, and 256 bits, respectively) and the number of rounds (10, 12, and 14, respectively) required to open the vault that seals around the data.Encryption is accomplished in this algorithm by exchanging some of the characters containing the key and data. The encrypted files are set to read-only mode, which ensures that the data in the files cannot be tampered with. The system's main feature is that it disables the delete option in the right-click menu for encrypted files. This increases the security of the files on the disc. Amin et al [2] They primarily focused on two main concepts in this paper: user cooperation and resource sharing. They proposed a centralised P2P network based on a directory server that organises peer-to-peer connections. They used the directory server to help with the overlay network's formation and to ensure a quick and efficient search. They devised a method for building and maintaining the overlay network that maximises resource efficiency while minimising delay. Finally, simulation is used to evaluate the application's performance. Oliveira et al [3] They put a lot of emphasis on the access control capabilities of a few cloud storage services that allow users to share data in an unmodified cloud directly in a pay-as-you-go model. They used Amazon S3, Google Storage, HP Public Cloud, RackSpace Cloud Files, Windows Azure Storage, and Luna Cloud as cloud services. They described the services' permissions, their semantics, and the various access-granting techniques used to apply these permissions to specific users. A set of protocols for securely sharing data in several public storage clouds was also presented. These protocols were developed by extending an ideal set of properties needed for data sharing between different cloud service users. Using this as a guide, we stored our files in Azure using Azure Storage Services. We can ensure the security of our files by using Azure services.We used Mohinish et al [4] model for incorporating a block chain into a transaction where the user sends a small amount of crypto to the file's author. Smart contracts are used to accomplish this. [4] used block chain to reward waste management system participants in their paper. They proposed a paper for a new smart management system based on block chain and the Internet of Things. The system uses block chain to reward users for putting their trash in smart bins and using smart contracts.

III. PROPOSED WORK-APPLICATION

To protect files and data in the cloud we proposed the Secure File Sharing solution to protect files and data on the cloud. At the same time, the suggested System user can exchange files with many users. In this application, users will have the atmost level of security. Before uploading a file to the server, the user can encrypt it. After uploading, all users who have registered in this application can view files as well as information about the file, such as who uploaded, title,size,etc. If a user wishes to see a file, they must request for downloading file from the specific user who posted the content. While downloading the file the user is redirected to the crypto payment gatway.After successful transcation of funds the user receives an acknowledgement email containing the secret key.with the secret key the user be able to view or download files.

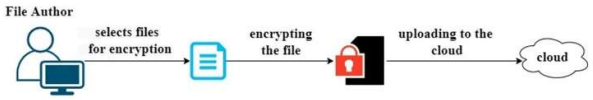


Fig.1. System Architecture Encrypting & uploading files to cloud

To encrypt files in this proposed system, we used the AES algorithm. While encrypting the files, a secret key is generated. This application can encrypt any type of file, such as docx,pdf,mp4, and so on; the secret key must meet the criteria for key generation, such as length and complexity. The filename will be changed to filename.txt after encryption, indicating that the encrypted file can be viewed in text format with cipher text. No one can tamper with the data in the file if it is encrypted.

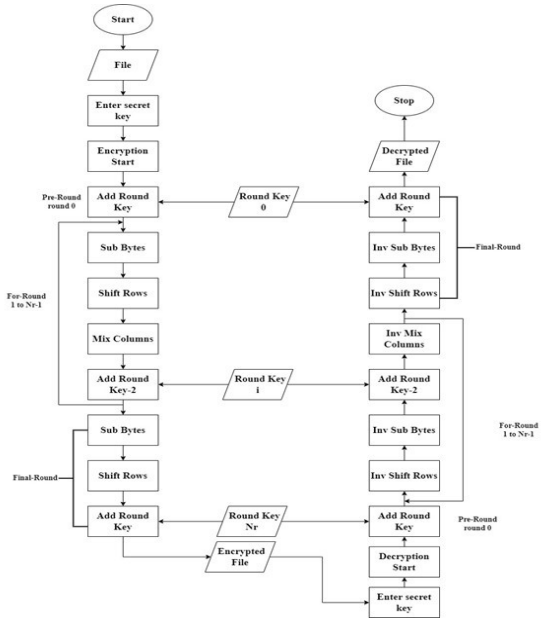


Fig.2. Flowchart of the encryption and decryption process

As we developed a web-based application for securely sharing files. In order to use the application, the user must first register with it. After completing the registration process, the user is redirected to the login page and logged in. The user can now navigate all of the application's options, which include viewing files, uploading files, encrypting files, and decrypting files. If a user wants to share a file, he or she must first encrypt it using the encrypt function. The user creates a private key to secure the files during the encryption process. After the encryption is complete, the user can proceed to upload files and upload the file along with the file information.

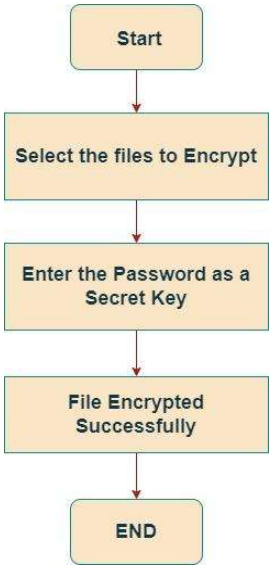


Fig. 3. Encryption Function

Following the upload of files, any user who wishes to view the file must have the secret key generated by the uploader. When decrypting the file, the user must provide the same secret key that was specified by the uploader. If the key is not matched, the user will be unable to decrypt the file and will receive an error message stating "file cannot be decrypted." If the secret key matches, the user can view and download the file. By following this methodology, we can ensure that our files on the server are secure.

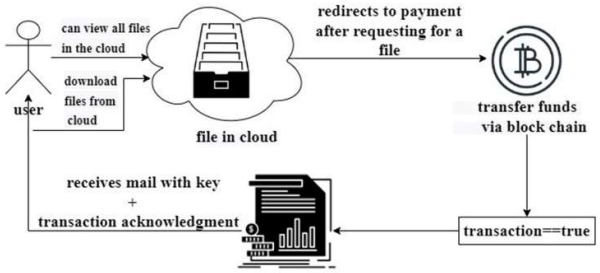


Fig.4. System architecture for retriving the files

The viewfiles page displays all of the files that have been uploaded. The uploaded files are all saved on cloud. All encrypted files can be viewed by users who have registered with the application. Now, if a user wants to view encrypted files, they must first download the file. When a user clicks on the download button, they are redirected to the funds transfer page, where they can download the files if they approve the funds. After downloading the files, the user receives an email with a secure pdf containing the key and acknowledgement information.

Fig.5. Flow Chart for decrypting the file

Then selects file for decryption process.uploads the file to the decryption function. To view the file during the decryption process, we must enter the key generated by the author. The user can view the file after successful decryption.

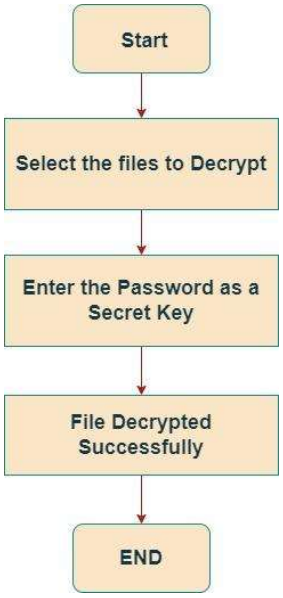


Fig.6. Decryption Function

As previously stated, transferring funds shall be carried out using block chain technology.We use blockchain technology to record data of file upload and download requests. Cryptocurrency concept is used from blockchain to also carry out the feature of Pay to view. Smart contract is responsible for carrying out these different operations. Blockchain is a shared, immutable ledger that enables the manner of recording transactions and monitoring belongings in a commercial enterprise network. Blockchain consists of several blocks that are connected and distributed over a network. the blockchain is divided into small blocks which will store the data of the cryptographic hash of the previous block, timestamp, and transaction data. The bitcoin blockchain was developed as an alternative to the present financial system, where banks act as trusted intermediaries for conducting varied valid transactions and preventing frauds. Banks maintain a personal database to store information of their customers and charge high transact their fees for the provide. Today, blockchain technology contains various other crypto currencies like Ether, Ripple, and Lite-coin, etc and it is an open-source network and provides transactions. A Smart Contract-based fund transfer system can ensure people get their funds securely without any delays .The entire process is run on top of Blockchain which gives the benefit of transparency.The main goal of using block chain for transferring funds is to make the application more transparent so that the author can see who made the transfer and who downloaded the files.All transaction details are recorded in the block chain.

Smart contract is an agreement between the parties on a blockchain, which is used to transfer money or any other things completely in a digitally and transparent manner. Ethereum is a very popular decentralized open-source blockchain platform with smart contract functionality. Smart Contracts are small programs that are stored on a blockchain that runs when the conditions are met. It is a self-executing process. They are used to automate the tactic of executing agreements, so that all participants are often immediately certain of the result, with no intermediary's involvement or time loss. Smart contracts allow the performance of all the transactions with no one's involvement of the third parties. Smart contracts not only define penalties around an agreement but within the same way that a standard contract does, but also automatically enforce those processes with no middle-person involvement.

Cloud based file sharing using blockchain uses blockchain technology to record data of file upload and download requests. Cryptocurrency concept is used from blockchain to also carry out the feature of Pay to view. Smart contract is responsible for carrying out these different operations. Uploader can upload his file to application for which he will sign a Blockchain transaction. He will not be charged any amount for this but instead all of the required details like Wallet address(to which he intends to receive his crypto tokens from downloaders), amount of royalty, data and time are recorded in this initial transaction. Later the encrypted file is uploaded to Azure DB and will be ready to download for others.

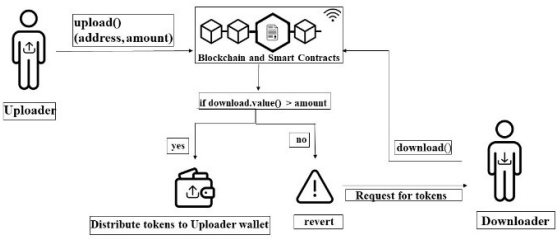


Fig.7. Flowchart of processing Pay to View

Downloaders can check for the file they need from a list of available files. They need to pay the necessary amount specified(royalty) by File owner/uploader in cryptocurrency. A blockchain transaction is recorded upon successful payment along with details of downloader and date,time of download. This creates an open ledger of uploads, downloads details in blockchain.Blockchain technology is thus used effectively to carry out Payments and maintain record of users who upload and retrieve files.

As previously stated, we developed our application on the Azure cloud. Microsoft Azure is a cloud computing service created and run by Microsoft. Azure offers a plethora of cloud and related services. Like most Microsoft products, Azure is an open source platform, but the client SDK is open source. Azure provides "Software as a Service (SAAS)", "Platform as a Service (PaaS)" and "Infrastructure as a Service (IAAS)" architectures for cloud computing. Azure provides a collection of cloud services that are heavily used in day-to-day business operations. Azure mainly provides products and services related to Microsoft, but also provides services related to Linux. Azure can run email servers, web

servers, applications, Active Directory, virtual machines, remote desktop management, CDNs, messaging services, data management and big data tools, etc. without having to install any software locally using only the relevant Azure services[5].

We initially developed the app locally, and after testing it locally, we deployed it to the cloud for hosting and production. However, deploying an app is not so simple; there are numerous processes that must be carried out in the backend**.** As we used Django framework we had to use the Azure App Service to deploy our code to the cloud, which requires Python 3.7 or higher in a Linux server environment. The azure app service allows you to create and host web apps, backend jobs, and REST APIs in any programming language or framework. The Azure app service enables automatic scaling and high availability of our choice without the need for infrastructure management. I chose Azure App Service for hosting because, while maintaining my application on-premise, I am responsible for many things such as:

- we need to specify and purchase physical servers, storage, networking equipment, and all other necessary hardware.
- Check that the main power supply, backup power supply, cooling system, and so on are all in place.
- Install and configure the network
- Install and configure any required virtualization software, operating system, middleware, or runtime components for your application.
- Install and configure a web server such as IIS, Apache, or Nginx.

Because Azure is a Platform as a Service[6], we can eliminate all of these jobs by deploying applications to App Service. We can only focus on the management of our application and its data. Azure handles everything else. You don't have to worry about things like network management or underlying infrastructure. Installing operating system updates, critical patches, runtime components, or middleware. Azure is in charge of all of this. This gives you even more time to focus on what is important to our application.

To host our app in Azure, you must first create an Azure App Service web app. In the Azure portal, we can create an Azure app. We must configure all of the required settings, such as resource group selection, web app naming, application environment selection, app service plan, and so on. We can then create the web app if all of the settings have been validated.Then our web app will be created.After creating web app we need to create PosgreSQL database in Azure to migrate our database from on-premise to azure. After the Azure Database for PostgreSQL is created, we configure access to the server from the web app by adding a firewall rule. With the web app and PostgreSQL database we connect the web app to the PostgreSQL database in Azure.

The web app code uses database information in four environment variables named DBHOST, DBNAME, DBUSER, and DBPASS to connect to the PostgresSQL server.After connecting to PostgreSQL database we deploy our application code to Azure. We can connect to the application from any device after successful deployment because Azure generates a DNS name for connecting via public internet. [7].

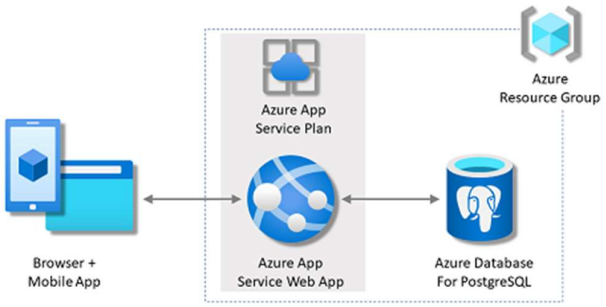
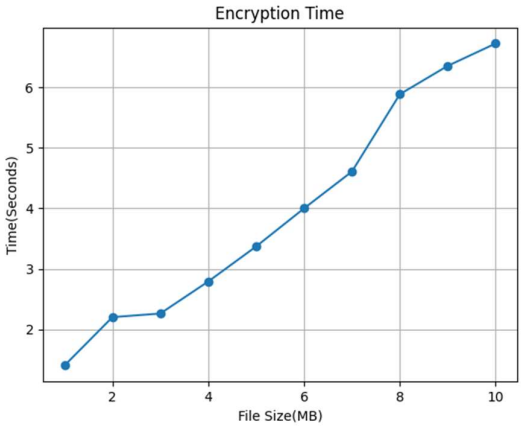


Fig.7. System Architecture after deploying application in Azure

IV. RESULTS AND DISCUSSION

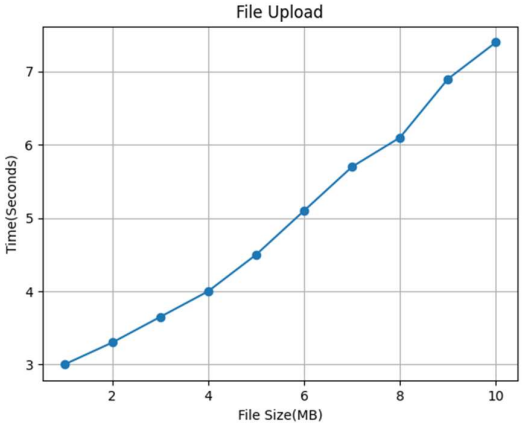
We used Azure App Service to deploy our application in the cloud. After deployment, Azure creates a public URL that may be used to access the application from anywhere. To interface with the Blockchain network during the deployment, we used the Remix platform and the web3.js API. We utilised the metamask plugin wallet and test accounts in that wallet for transactions. We deployed our system on matic networks.

Figure 1 shows that we must first encrypt the file before transferring it to the cloud. So we experimented with various file sizes to see how long the function takes for encryption.



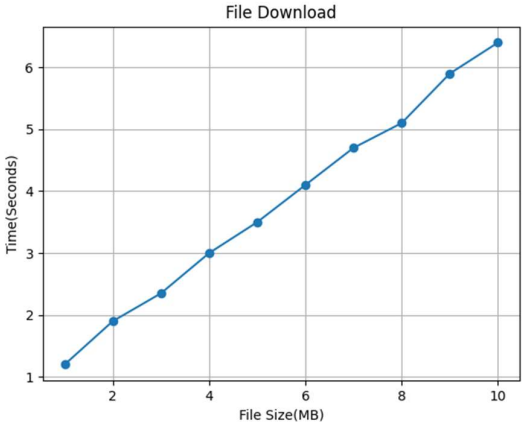
File Size vs Time

So, after encrypting the data, we must upload them to the cloud. The file size will grow after encryption. So we uploaded all of the encrypted files and computed how long it would take to upload the data to the cloud.



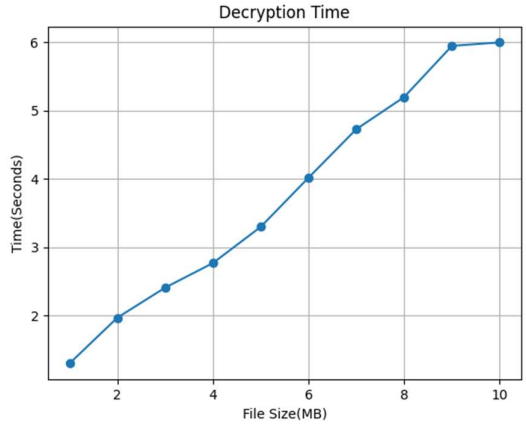
File Size vs Time

So, according to Figure 4, after uploading data to the cloud, all registered users may read the files. As a result, if a user intends to download the file, they must first send the crypto. Following the transaction, the user can download the file. As a result, several file sizes have altered. So we measured how long it takes to download the files.



File Size vs Time

As a result, after obtaining the data, we must decrypt them. Figure 6 depicts the decryption procedure. So, in order to calculate how much time is necessary for decryption, we tested the function with various sizes. Despite the fact that the file size has grown after encryption, the decryption procedure is much faster than the encryption. We obtained that information from the graph.



V. CONCLUSION AND FUTURE WORK

Sharing data securely is a challenging task. We can be certain that we will eliminate third parties while safely sharing their files if we use this strategy. We built this application on a very modest scale by concentrating on innovative architecture and methods. To get this app into production, we need to integrate it with a CDN (Content Delivery Network). Because file sharing is utilised by millions of people, we want an efficient programme that can satisfy all users' requests. As a result, in the future, we would like to combine our application with a CDN.

VI. REFERENCES

[1] Aditya Rayarapu et al., “Securing Files Using AES Algorithm”,2013 (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 4 (3) , 2013, 433-435

[2] H. Amin, M. K. Chahine and G. Mazzini, "P2P application for file sharing," 2012 19th International Conference on Telecommunications (ICT), 2012, pp. 1-4, doi: 10.1109/ICTEL.2012.6221249.

[3] Oliveira, T., Mendes, R., Bessani, A. (2014). Sharing Files Using Cloud Storage Services. In: , *et al.* Euro-Par 2014: Parallel Processing Workshops. Euro-Par 2014. Lecture Notes in Computer Science, vol 8806. Springer, Cham. https://doi.org/10.1007/978-3-319-14313-2_2

[4] M. Paturi, S. Puvvada, B. S. Ponnuru, M. Simhadri, B. S.Egala and A. K. Pradhan, "Smart Solid Waste Management System Using Blockchain and IoT for Smart Cities," 2021 IEEE International Symposium on Smart Electronic Systems (iSES),2021,pp.456-459,doi: 10.1109/iSES52644.2021.00107.

[5] https://docs.microsoft.com/en-us/azure/app-service/overview#app-service-on-linux

[6] https://azure.microsoft.com/en-in/overview/what-is-paas/

[7] https://docs.microsoft.com/en-us/azure/app-service/tutorial-python-postgresql-app?tabs=flask%2Cwindows%2Cazure-portal%2Cterminal-bash%2Cazure-portal-access%2Cvscode-aztools-deploy%2Cdeploy-instructions-azportal%2Cdeploy-instructions--zip-azcli%2Cdeploy-instructions-curl-bash