IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

EzShare: An Android File Sharing System Using BlockChain & IPFS

Devesh Solanki[1] , Dev Mehta[2] , Kaushal Modani[3] , Sashank R M[4]

Dr Rashmi S[5]

[1,2,3,4] *Students, Department of Information Science and Engineering, Dayanandasagar College of Engineering, Bangalore, Karnataka, India*

[5] *Associate Professor, Department of Information Science and Engineering, Dayanandasagar College of Engineering Bangalore, Karnataka, India*

Abstract - *The widespread adoption of smartphones has replaced desktop computers and laptops as the first component of the computer, due to mobility, regular communication and application versatility. Mobile devices include extensive data storage including sensitive objects such as verification credentials, photos, videos, personal information, activity details, and much more. Therefore, protecting data stored on mobile devices becomes a serious problem. In this update, we are investigating the safety of the Android storage model between 2013 and 2018. Several threats are found in literature that can be classified as physical or software threats. Interest in the blockchain has hit the top levels recently. While most of the buzz has been around for blockchain use like cryptocurrencies and ICOs, the technology itself is exciting. The blockchain provides democratized trust and authentication protocols that have already disrupted banking and are at risk of adding health care, financial services, social apps and more. We use BlockChain and IPFS to store Hash values and files, secure file transfers can also be obtained.*

Key Words: **Securing Data, BlockChain, Etherium-Blockchain, IPFS, Hash Values, SHA 258**

1. INTRODUCTION

This paper aims to complement previous reviews about insecure data storage on Android smartphones by increasing coverage of security threats and solutions. We believe that in‑depth testing of the Android data storage model is necessary. Therefore, we are reviewing Android attacks, threats, and their solutions for the period 2013-2018. Additionally, we propose a phased division of the Android data storage threat model based on physical and software threats and review a few tasks for each class. In addition, solutions to reduce each category are being investigated. The Android App is therefore designed for secure storage and protection using BlockChain and IPFS.

1.1 Registration:

The user can register using the Android app. While the registering user can provide name, email, mobile number, password etc. For =firebase references we import the DatabaseReference and Firebase Database from com.google.firebase: firebase database: 19.3.0 package. While the login user can log in via email and password, and verified by firebase. HTTP protocol is used for communication between the mobile application and firebase.

1.2 File Creation

User can create a pdf file using the mobile application. To create a file, the user can enter a file name, take pictures and save them to external storage. The user can generate a pdf file from the captured images using the itextpdf package and upload to the server. When the server receives the pdf file, it produces a hash file using the SHA 256 algorithm. Once the file has been uploaded to ipfs, ipfs reverts the ipfs hash file, using that file where it can no longer be found on ipfs. The file name, hash and ipfs file hash are stored in the etherium blockchain.

1.3 View and download file

When a user wants to view their file, they can send a request to the server. The server pulls the file details from the etherium blockchain and sends the list back to the user's mobile app. The user can select the file they want to download. Once selected file, the server collects the file from ipf using the ipfs file hash it found in the blockchain. Once the server has received the file, generate a hash value using SHA 256, then compare the hash file stored in the blockchain while uploading. Once verified, send the file to the mobile app. The user can view the file and interact with others as well.

1.4 Sharing

When the files are retrieved from the etherium blockchain, the user can select the files he/she wants to share.

2. Literature Review :

In the last few years, there has been a dramatic shift in information technology. This includes various ways in which files can be shared and stored. Android OS is a the latest mobile OS that has been gradually taking over the ever-expanding market share.

In the last few years, there has been a dramatic shift in information technology. Easy to use and easy to develop for and open-source, it has picked up a following of developers who want to create content for the masses. Cloud computing is known as the next big step for all forms of standard technology use. From businesses, to non-profit organisations, to individual users, there seems to be a variety of programs that can use cloud computing to provide a better, faster, and smarter computations. This paper aims to combine the two, build a cloud-based Android system, and give users the power of cloud computing in the palm of their hand. [1]

The performance of mobile devices, mainly smartphones, has improved rapidly over the past years. Many users use high-performance smartphones, and use content on smart phones longer than other devices. As a result, users are constantly sharing content and file sharing requirements through enhanced calls have increased dramatically. In order to overcome such problems, we bring an application for seamless file sharing for the Android devices. We anticipate that the proposed application could be reliable file sharing and cost effective solution between mobile devices. [2]

The centralized storage model is currently used for storing sensitive information. The main disadvantage of the centralized model is the difficulty in maintaining user privacy. Threats related to user (patient) privacy include unauthorized access to sensitive information such as identity details and diseases from which the patient suffers, and misuse of patients' data and their medical reports. To address this issue, we propose a distributed off-chain storage of medical data using IPFS (Interplanetary File System) and blockchain technology. The proposed framework while maintaining patient confidentiality facilitates easy access to medical information by authorized organizations such as health care providers (e.g., physicians and nurses). In addition, it achieves consistency, integrity, and availability. [3]

Android is the world's most widely used mobile operating system that dominates the smartphone market by 82.8% in 2015. Many techniques are used to protect data stored on mobile phones, especially based on password-based data encryption. Therefore, we are reviewing Android attacks, threats, and their solutions for the period 2013-2018. Additionally, we propose a phased division of the Android data storage threat model based on physical and software threats and review a few tasks for each class. [4]

2.1 Existing System :

System storage is a catalog where all the Android OS is available and protected by the Linux access control component. Application specific storage is controlled by a specific application and can only be read and written by that application, usually included in / data /. Another way to store shared storage (internal SD card), mounted on / sdcard / or / mnt / sdcard /, which is used to share information between applications. Additionally, some Android devices contain an external, removable SD card partitioned under shared storage. However, primary and secondary SD cards are sometimes referred to as external storage. To prevent this split of storage, Android relies on the Discretionary Access Control (DAC) process provided by the Linux sub-program to use system access control and system-specific storage. [4]

Discretionary Access Control (DAC) identifies threats and threats in Android Data storage, divided into visual threats and software threats. Factory reset can be separated into both software and physical threats Researchers' sets focus on Android smartphones with visible threats because private data can be stored in memory on mobile devices long after use. [4]

2.3 Disadvantages of Existing System

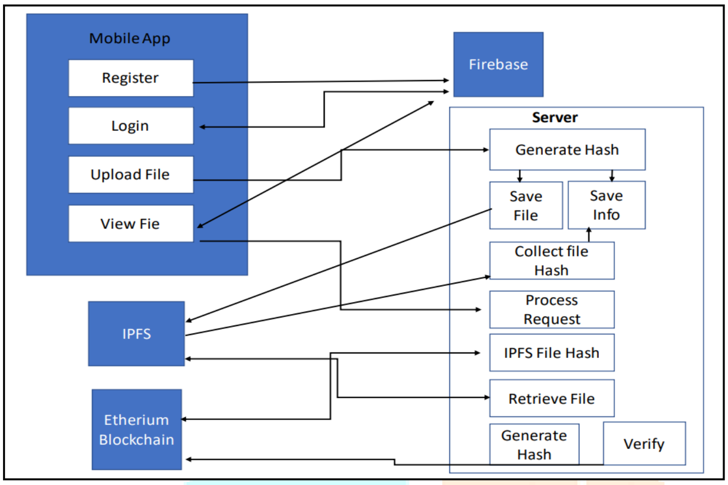
- Data security issues
- Many different types of attacks on the sensitive data

→ High risk to store the data into the phone

3. PROPOSED MODEL

The proposed system looks at how the application works with the algorithms used. Also summaries about app maintenance, performance, the storage and working of the application.

3.1 Architecture



A. <u>Peers Verification in Consortium Network</u>

Algorithm 1 is proposed to verify the peers in the consortium network for document sharing and uploading. The proposed algorithm prevents malicious peers from accessing the shared documents. The Proof-of-Identity ensures security in the model. Once peers gets registered in the network, they are provided with a unique Proof-of-Identity (http ://< url : port >/nodes/register). The Proof-of-Identity (PoI) gets verified for authentication of peers.

Algorithm 1: Algorithm for verification of peers

Input: Registration-Id
Output: document upload with verification
// checking authorization of peers while document upload//
if (document.sender is not authorized) then return false;
else
    Share/upload document from/to IPFS off-chain storage
    return true;
end

B. <u>Data Storage in Consortium Network</u>

Once the upload action is initiated by user, the generated files will be added to the IPFS.
At the same time, IPFS Hash (content-addressed hash) will be added to the consortium chain.
In addition, mapping is performed to identify the user using the Registration ID and their corresponding Block Id details which is shown in **Algorithm 2.**

Algorithm 2: Algorithm for off-chain and on-chain data storage

---

Input: Registration-Id, file
Output: off-chain and on-chain storage of file (file)
    // Adding the file to IPFS storage & collecting it's Hash //
IPFS Hash = file | ipfs add
    // Add IPFS Hash into consortium network//
Block Id = add Hash(IPFS Hash)
    //Mapping the Registration-Id and IPFS Hash of uploaded file //
map user(Registration-Id, Block Id, IPFS Hash, timestamp, PoW, Hash of previous Block)

As mentioned before, only hashes of actual reports are stored (on-chain storage) on the blockchain network in the proposed way for the following reasons. If both diagnostic reports and their hashes are embedded in the blockchain, for the first, as more & more diagnostic reports are loaded, the size of the blockchain will grow exponentially. Second, whenever a new peer arrives, they should copy the entire series containing details of all the files. This method is definitely not measurable, especially since the size of the diagnostic report can go up to several megabytes. For off-chain storage of diagnostic chains for diagnostic reports and hashes, IPFS is a P2P-enabled hypermedia protocol. IPFS generates a fingerprint of a file by inserting a hash based on cryptographic content. These unique hassles help to eliminate inefficiency across the network. In addition, they are used to retrieve diagnostic report. By keeping these hashes in the blockchain network, we are saving a lot in size. Additionally, it helps to maintain privacy in transaction (files).

3.2 BlockChain :

Blockchain is just a bunch of blocks. Each block contains pieces of information. Each block is connected to the previous one. To date, the main use of the blockchain is cryptocurrencies. Cryptocurrency is an electronic currency system that operates without central authority. Cryptocurrencies are gaining acceptance because they solve the problem of double spending - a situation where an unscrupulous user of a decentralized cash system uses the same currency twice. For Bitcoin, Ethereum, and other cryptocurrencies using the same currency twice is almost impossible. A blockchain is a way to organize and store data. There are alternatives, of course, though the blockchain has a set of different features.

3.3 Features :

Why use blockchain? Why not use traditional storage methods, such as a SQL database or keyword storage? Blockchain offers 2 features that make it a good use of cryptocurrensets: distributed and trust-less. Even if a company uses data replication, it's usually 2-5 copies.

In a blockchain, the number of copies can be hundreds or thousands. Thus, in that way blockchain is **distributed**. [5]
The number of copies of Bitcoin ledger is equal to the number of full nodes. To clear the transaction information from the presence, you need to delete it from each complete node around the world. To set up a blockchain, you can't just hack one computer or the whole world. You need to break into computers in different locations, use different operating systems and take different security measures. Stopping a blockchain by hacking each complete/full nodes is not possible. [5]

In blockchain, you do not need to trust any organization. You do not need to trust any particular user. Instead, you believe most users will behave honestly. In that sense, a blockchain is an trust-less system. One of the features of the trust-less system is that fair rules for everyone. [5]

3.4 IPFS :

The Inter Planetary File System (IPFS) is a network and peer-to-peer network for storing and sharing data on a distributed file system. IPFS uses content to identify separately each file in a global namespace that connects all computer devices. [6]
IPFS allows users to host and receive content in a manner similar to BitTorrent. As opposed to a centrally located server, IPFS is built in a decentralized system of user-operators who hold a portion of the overall data, creating a resilient system of file storage and sharing. Any user on the network can serve a file with the content address, and other peers on the network can find and request that content on any node they have using the distributed hash table (DHT). [6]
Unlike BitTorrent, IPFS aims to create a single global network. This means that if Alice and Bob publish the data block with the same hash, peers who download content from Alice will exchange data with those who download it from Bob. IPFS aims to replace the protocols used for static web page delivery through HTTP-accessible gateways. Users may choose not to install the IPFS client on their devices and instead use a public gateway. A list of these gateways is stored on the IPFS GitHub page. [6]

3.5 Here's how IPFS works :

→ Your file, and all of the blocks within it, is given a unique fingerprint called a cryptographic hash. [7]
→ IPFS removes duplicates across the network. [7]
→ Each network node stores only content it is interested in, plus some indexing information that helps figure out which node is storing what. [7]
→ When you look up a file to view or download, you're asking the network to find the nodes that are storing the content behind that file's hash. [7]
→ You do not need to remember the hash, though — every file can be found by human-readable names using a decentralized naming system called IPNS. [7]

3.6 Nodes :

Nodes are computers that read and write to the blockchain. Different types of nodes have different roles and abilities.

	Makes new blocks	Sends new transactions	Knows the balance of his wallet	Knows all transactions
Mining nodes	+	-	-	-
Full nodes	-	+	+	+
Light nodes	-	+	+	-
"Web" nodes	-	+ (via 3rd party services)	+ (via 3rd party services)	-

3.7 CONCUSSES IN THE BLOCKCHAIN :

With decentralized systems, and especially with our e-Voting Blockchain-based system, conflict issues can arise. This happens when different votes are casted by different voters at about the same time.

As mentioned earlier in this paper, when a voter casts a vote, it will be linked to the previous vote to create a chain that neither corruptible nor changeable. In the case of concusses, our solution is to use the Longest Chain Rule, which is used by Bitcoin to resolve the same issue. Let's assume all the blocks in the system are votes that have just been cast at the same time and they were all assigned the number 1002 in the chain. We will call these three new blocks 1001-A, 1001-B, and 1001-C. [8]

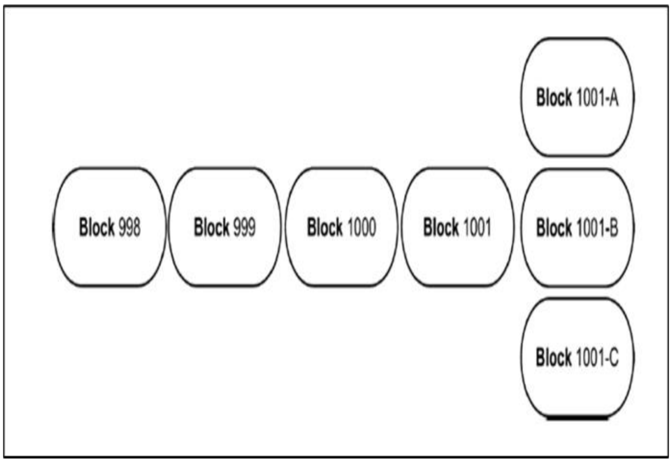


Fig: Concusses in the Blockchain

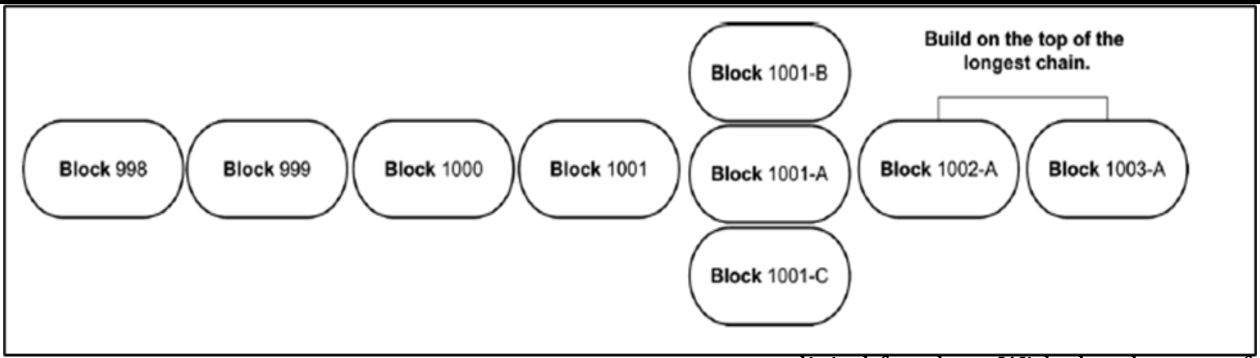


Fig: Longest Chain Rule

Let us assume that Block 1001-A was first introduced in Blockchain, so the system will be added to the chain as it follows Block 1001. Later, Block 1001-B was introduced in this series. The system will hold on to it and wait until another block arrives. When Block 1002-A is introduced into the system, Blockchain will assume that Block 1001-A is a valid block and will continue to build on a longer chain. Block 1001- B and 1001-C will be considered orphan blocks. [8]

Because we have a different chain for each candidate, orphan blocks will no longer be a problem because they contain the same information (vote) as other blocks, and will be considered when counting votes. [8]

3.8 Importance of BlockChain :

- ✓ **Immutability :** Immutability enables companies to ensure that there is no disruption done to the packages in transit. As blockchain is immutable, it is not possible to alter the package information in any way. Any alteration will alarm the system. [9]

- ✓ **Transparency :** Companies can also utilize it to ensure that the end-user can interact with the processes with full/partial transparency. [9]

get true digital freedom. With the absence of any central authority, you are the sole owner and person responsible for your assets. It gives you the digital freedom that relies heavily on the backbone of blockchain technology. [9]

- ✓ **Truly Decentralized Services :** Decentralized services are the backbone of our futuristic society. This will give users unprecedented access to the options that are currently not available in the market. [9]

- ✓ **Better Security :** Blockchain uses cryptography to add a layer of security to the data stored on the network. The decentralization feature, on top of the cryptography, makes blockchain provide better security than other systems. [9]

- ✓ **Improved Efficiency :** The cause is better security, intermediary removal, and overall better processes. Transactions also take seconds rather than a week to complete, especially international transactions. [9]

4. EXPERIMENTAL RESULTS

Table 1 Unit Test Case 1

S2 # 3Test 3Case	UTC-*1
Name3of3Test	Registration
Expected3Result	User should be able to register using the mobile app and the details should be saved in the google firebase.
Actual3output	Same3as3expected.
Remarks3	Successful

Table 2 Unit Test Case 2

S1 # 3Test 3Case	UTC-*2
Name3of3Test	User login.
Expected3Result	User can login by entering the user id & password and if firebase cloud should verify
Actual3output	Same3as3expected.
Remarks3	Successful

Table 3 Unit Test Case 3

S3 # 3Test 3Case	UTC-*3
Name3of3Test	Capture Image
Expected3Result	User should be able to capture images using the app and should get saved in the external storage
Actual3output	Same3as3expected.
Remarks3	Successful

Table 4 Unit Test Case 4

S4 # 3Test 3Case	UTC-*4
Name3of3Test	Create PDF File
Expected3Result	User should be able to create pdf file using the images
Actual3output	Same3as3expected.
Remarks3	Successful

Table 5 Unit Test Case 5

S5 # 3Test 3Case	UTC-*5
Name3of3Test	Save PDF
Expected3Result	User should be able to upload the pdf file into the server
Actual3output	Same3as3expected.
Remarks3	Successful

Table 6 Unit Test Case 6

S6 # 3Test 3Case	UTC-*6
Name3of3Test	File hash generation
Expected3Result	Server should be able to generate the hash value of the file using SHA 256
Actual3output	Same3as3expected.
Remarks3	Successful

Table 7 Unit Test Case 7

S7 # 3Test 3Case	UTC-*7
Name3of3Test	Save File
Expected3Result	Server should be able to store the file into IPFS and receive the ipfs file hash
Actual3output	Same3as3expected.
Remarks3	Successful

Table 8 Unit Test Case 8

S8 # 3Test 3Case	UTC-*8
Name3of3Test	Save file details
Expected3Result	Server should be able to save the file details like hash value, file name and ipfs hash value into the blockchain
Actual3output	Same3as3expected.
Remarks3	Successful

Table 9 Unit Test Case 9

S9 # 3Test 3Case	UTC-*9
Name3of3Test	Download File
Expected3Result	Server should be able to get the file from ipfs
Actual3output	Same3as3expected.
Remarks3	Successful

Table 10 Unit Test Case 10

S10 # 3Test 3Case	UTC-*10
Name3of3Test	File verification
Expected3Result	Server should be able to retrieve the file details from the blockchain and verify the hash value of the file
Actual3output	Same3as3expected.
Remarks3	Successful

Table 11 Unit Test Case 11

S11 # 3Test 3Case	UTC-*11
Name3of3Test	View and Share
Expected3Result	User should be able to view the file and should be able to share using other platform
Actual3output	Same3as3expected.
Remarks3	Successful

5. CONCLUSIONS

A secured file storage and sharing application is developed using blockchain and google firebase. The file is stored in the distributed IPFS and details are stored in the Etherium Blockchain. The file can be shared with the other users. It ensures the data integrity to be maintained and fend off data loss and network attacks.

6. ACKNOWLEDGEMENT

The authors would like to thank Dayananda Sagar College of Engineering for providing an opportunity to deploy the evolving technology. We also express sincere gratitude to our guide for providing their valuable guidance and necessary facilities needed for the successful completion of this paper throughout

7. REFERENCES

[1] Design And Implementation Of A File Sharing Application For Android(2013)
Authors: Alatishe A.A, Adegbola M.A
[http://www.ijcse.net/docs/IJCSE13-02-05-055.pdf]

[2] Seamless file sharing for Android devices
Authors: MinSeok Jeon, Sun-Kyum Kim
[https://ieeexplore.ieee.org/document/6803153]

[3] Distributed Off-chain Storage of Patient Diagnostic Reports in Healthcare System using IPFS and Blockchain (2020)
Authors: Randhir Kumar, Ningrinla Marchang
[https://ieeexplore.ieee.org/document/9027313]

[4] Android Data Storage Security: A Review
Authors: Haya Altuwaijri Sanaa Ghouzali
[https://bit.ly/3eHFdi8]

[5] BlockChain Explained
Authors: Timur Badretdinov
[https://bit.ly/33JbIGp]

[6] IPFS - A Peer-To-Peer Hypermedia Protocol
[https://github.com/ipfs]

[7] IPFS Stream
[https://ipfs-stream.ga]

[8] A Conceptual Secure BlockChain-Based Electronic Voting Systen
Authors: Ahmed Ben Ayed
[researchgate.net/publication/341498272]

[9] 101 BlockChains
[https://bit.ly/3tJscJg]